Endbericht

PG555: Cloud-basiertes Internetbanking System

30. März 2012

Autoren		
Wei	Cai	(CW)
Richard	Hellwig	(RH)
Johann	Kexel	(JK)
Viktor	Mucha	(VM)
Thorben	Seeland	(TS)
Daniel	Spasovski	(DS)
Viktor	Stoklossa	(VS)
Anna	Vasileva	(AV)
Muhammad	Waqas	(MW)
Dong	Yang	(YD)

Inhaltsverzeichnis

0. Einleitung **1**
0.1. Problembeschreibung . . . 1
0.2. Aufbau des Dokuments . . . 2

I. Dokumentationen und Compliance **3**

1. Technische Dokumentation **5**
1.1. Cloud . . . 5
1.1.1. Verwendete Cloudtechnologie . . . 5
1.1.2. Banking-Umgebung . . . 6
1.1.3. Anwendung . . . 7
1.2. Datenbank . . . 10
1.2.1. Vorbetrachtung . . . 10
1.2.2. Zugriff auf die Datenbank . . . 12
1.2.3. Aufbau . . . 14
1.3. Glassfish . . . 20
1.3.1. Java Database Connectivity(JDBC) . . . 20
1.3.2. Sicherheitsrealm . . . 22
1.4. Komponenten . . . 24
1.4.1. Architektur . . . 24
1.5. Sicherheit . . . 28
1.5.1. Aufbau der Dokumentation . . . 28
1.5.2. Planung der Sicherheit . . . 28
1.5.3. Implementierung der Sicherheitsmaßnahmen . . . 30
1.5.4. Verwaltung der RSA-Schlüssel . . . 38
1.6. GUI Testen durch Selenium . . . 39
1.6.1. Einführung . . . 39
1.6.2. Selenium IDE . . . 39
1.6.3. Selenium RC . . . 39
1.6.4. Problem mit SSL-Zertifikate . . . 39
1.6.5. Selenium Testen in Form von JUnit . . . 41
1.7. Log-System mit Log4J . . . 42
1.7.1. Komponenten . . . 43
1.7.2. Besondere Konfiguration mit JSF und Spring unter Glassfish . . . 45
1.7.3. LogDaten . . . 45

2. Compliance **47**
2.1. Gesetze . . . 47
2.1.1. Behandlung der Gesetze nach dem Projektplan . . . 47
2.1.2. Betrachtete Gesetze . . . 48
2.1.3. Analyse und Dokumentation der Gesetze . . . 48
2.1.4. Anforderungen durch Gesetze . . . 49
2.2. Security Policy . . . 49
2.2.1. IT-Sicherheitsleitlinie (Security Policy Schicht 1, BSI 100-1) . . . 50
2.2.2. BSI-Grundschutz (Security Policy Schicht 2, BSI 100-2) . . . 50
2.2.3. Handbücher (Security Policy Schicht 3, BSI 100-3) . . . 51

II. Abweichungen vom Pflichtenheft **53**

3. Abweichungen vom Pflichtenheft **55**
3.1. Kommunikationsschnittstellen . . . 55
3.1.1. Kommunikation mit der Schufa . . . 55
3.1.2. Börse und Bafin . . . 56
3.1.3. Interbank . . . 56
3.2. Rollen und deren Funktionalitäten . . . 56
3.2.1. Kundenberater . . . 56
3.2.2. Kunde . . . 58
3.2.3. Geschäftsführer . . . 59
3.2.4. Produktentwickler . . . 59
3.2.5. Controlling-Mitarbeiter . . . 59
3.2.6. Kassierer . . . 60
3.2.7. Jurist . . . 60
3.2.8. System . . . 61
3.2.9. Systemadministrator . . . 62
3.2.10. Eigenhändler . . . 62
3.2.11. Marketing . . . 63
3.3. Automatisierte Prozesse . . . 63

III. Maßnahmen und Erfahrungen **65**

4. Maßnahmen des Managements **67**
4.1. Projektmanagement . . . 67
4.1.1. Zusammenfassung des Projektmanagements im Projektplan . . . 67
4.1.2. Vorgehen in der zweiten Projektphase . . . 68
4.2. Teammanagement . . . 71
4.2.1. Zusammenfassung des Teammanagements im Projektplan . . . 71
4.3. Qualitätsmanagement . . . 72
4.3.1. Zusammenfassung des Qualitätsmanagements im Projektplan . . . 72

4.3.2. Vorgehensweise im Qualitätsmanagement während des Projektes . 74
4.4. Risikomanagement 77
4.4.1. Zusammenfassung des Risikomanagements im Projektplan 77
4.5. Bewertung der Maßnahmen 79
4.5.1. Projektmanagement 79
4.5.2. Teammanagement 79
4.5.3. Qualtiätsmanagement 81
4.5.4. Risikomanagement 81

5. Erfahrungen **85**

IV. Anhang **91**

A. Anwender Dokumentation **93**
A.1. Einleitung 93
A.2. Vorausetzungen zur Nutzung 93
A.3. Benutzeroberfläche 93
A.4. Gemeinsame Funktionen 94
A.4.1. Ein- und Ausloggen 94
A.4.2. Rollen-Auswahl 95
A.4.3. Nachrichtenverkehr 96
A.5. Kunde 97
A.5.1. Banking 99
A.5.2. Börse 103
A.5.3. Persönliche Daten ansehen 105
A.5.4. Nachrichten 106
A.5.5. Kurse 106
A.6. Kundenberater 107
A.6.1. Aufgaben 108
A.6.2. Kundenverwaltung 108
A.6.3. Nachrichten 117
A.6.4. Kurse 117
A.7. Kassierer 119
A.8. Produktentwickler 120
A.8.1. Aufgaben 120
A.8.2. Statistiche Daten 120
A.8.3. Nachrichten 120
A.9. Marketing-Mitarbeiter 121
A.10. Controlling-Mitarbeiter 121
A.10.1. Aufgaben 121
A.10.2. Kundenverwaltung 122
A.10.3. Logs einsehen 122
A.10.4. Nachrichten 122

A.10.5. Kurse einsehen . . . 123
A.11. Jurist . . . 124
A.11.1. Aufgaben einsehen . . . 124
A.11.2. Kundenverwaltung . . . 124
A.11.3. Kontodatenliste beantragen . . . 125
A.11.4. Nachrichtenverkehr . . . 125
A.12. Geschäftsführer . . . 126
A.12.1. Limits setzen . . . 126
A.12.2. Kundenverwaltung . . . 127
A.12.3. Log einsehen . . . 127
A.12.4. Kontodatenliste beantragen . . . 128
A.12.5. Statistische Daten . . . 128
A.12.6. Einstellung der Verdächtigkeitsprüfung für Überweisung . . . 128
A.12.7. Nachrichtenverkehr . . . 128
A.12.8. Kurse einsehen . . . 128
A.13. Eigenhändler . . . 128
A.13.1. Börseangelegenheit . . . 129
A.13.2. Kontostand einsehen . . . 129
A.13.3. Nachrichtenverkehr . . . 129
A.14. Systemadministrator . . . 129
A.14.1. Systemstatus einsehen . . . 130
A.14.2. Mitarbeiterverwaltung . . . 130
A.14.3. Log einsehen . . . 131
A.14.4. Log auswerten . . . 132
A.14.5. Backup erstellen . . . 133
A.14.6. RSA Schlüssel verwalten . . . 133
A.14.7. Nachrichtenverkehr . . . 135
A.14.8. Runterfahren . . . 135

B. Datenbanktabellen **137**
B.1. Personen . . . 137
B.2. Interne Konten . . . 138
B.3. Transaktionen . . . 139
B.4. Börse und Nachrichten . . . 140
B.5. Vorgänge . . . 141
B.6. Stammdaten . . . 142

C. Security Policy **143**
C.1. IT-Sicherheitsleitlinie . . . 144
C.2. BSI-Grundschutzkataloge . . . 153

D. Technologieauswahl **221**
D.1. Technologieauswahl . . . 222

Literaturverzeichnis **240**

0. Einleitung

Autor: VM

Das Onlinebanking ist für die Banken eine wichtige Komponente zur Abwicklung ihrer Geschäftsprozesse. Die Entwicklung einer Software zur Umsetzung des Onlinebankings ist sehr komplex und stellt hohe Anforderungen an die Entwickler der Software. Im Rahmen der Projektgruppe *PG555* ist ein Bankensystem mit Online-Funktionalität von den Projektteilnehmern zu entwerfen. Das vorliegende Dokument beinhaltet den Endbericht des Projekts zur Entwicklung des Bankensystems.

0.1. Problembeschreibung

Die Aufgabe der Projektgruppe *PG555* ist die *Konzeption und Entwicklung eines sicheren Cloud-basierten Internetbanking-Systems mit anschließender Sicherheitsanalyse auf Basis von Business Process Mining*. Die Durchführung des Projekts lässt sich in zwei zeitliche Phasen unterteilen.

In der ersten Phase ist im Rahmen der Konzeption eine Spezifikation der Funktionen des Bankensystems zu entwerfen, die im Pflichtenheft [32] definiert ist und die Umsetzung der Anforderungen an das System widerspiegelt. Die Anforderungen sind in dem von dem Kunden der Projektgruppe verfassten Lastenheft [36] beschrieben. Neben der Spezifikation der reinen Funktionalität beinhaltet die Konzeption die Berücksichtigung von Themen, die im Kontext der Entwicklung eines Cloud-basierten Bankensystems relevant sind. Relevante Themen sind unter anderem die Compliance-Anforderungen der Bank, die Auswahl einer geeigneten Cloud-Umgebung und die Planung der IT-Sicherheit für das System.

Die zweite Phase hat das Ziel das Bankensystem zu entwickeln. Dazu sind die in der Spezifikation definierten Funktionen als Web-Applikation zu entwerfen und zu testen. Wichtige Aufgaben sind hierbei die Integration eines geeigneten Datenbanksystems zur Speicherung von Daten, die Implementierung der Geschäftsprozessverwaltung und der Systemschnittstellen, sowie die Erstellung von Webseiten für die Benutzeroberfläche. Das Testen der Funktionen dient der Qualitätssicherung.

0.2. Aufbau des Dokuments

Der Endbericht besteht aus drei Teilen. Im ersten Teil ist die technische Dokumentation des Bankensystems und die Umsetzung der Compliance-Anforderungen beschrieben. Im zweiten Teil wird auf die Unterschiede der Funktionen des Systems in Bezug zu den Funktionen, die im Pflichtenheft [32] definiert sind, eingegangen. Der dritte Teil enthält die Beschreibung der Maßnahmen des Managements und die Erfahrungen der Projektteilnehmer während des Projekts. Im Anhang befindet sich schließlich eine Anwenderdokumentation, in der die Nutzung des Bankensystems vorgestellt wird.

Im ersten Teil umfasst die technische Dokumentation die Beschreibung der Cloud, der Datenbank, des Servers, der Komponenten des Bankensystems und der Sicherheit. Der Abschnitt über die Cloud stellt die verwendete Technologie, die Banking-Umgebung und die Anwendung der Cloud vor. In der Dokumentation der Datenbank wird nach einer Vorbetrachtung, der Zugriff zu der Datenbank und ihr Aufbau vorgestellt. Im Abschnitt über den Server wird auf die Datenbankschnittstelle und eine sicherheitsspezifische Einrichtung des Servers eingegangen. Im Abschnitt über die Komponenten des Systems wird die Architektur und der Aufbau des Bankensystems beschrieben. Der Abschnitt über die Sicherheit stellt die Umsetzung der Sicherheit im System vor. In der Beschreibung der Compliance wird auf die Gesetze und die Security Policy eingegangen. Im Unterabschnitt über die Gesetze werden die für das Bankensystem relevanten Gesetze und die sich daraus ergebenden Anforderungen an das System vorgestellt. Der Unterabschnitt über die Security Policy fasst die Dokumente über die Security Policy des Bankensystems zusammen, die während des Projekts erarbeitet wurden.

Die Beschreibung der Unterschiede der Funktionen im zweiten Teil beginnt mit einem Abschnitt über die Kommunikationsschnittstellen des Systems. Daraufhin folgen Abschnitte, die zu je einer Rolle des Systems Bezug nehmen. Der Abschnitt über die Kommunikationsschnittstellen beschreibt die Unterschiede der Funktionen bezüglich der Interbank, der Schufa, der Börse und der BaFin. Jeder der rollenspezifischen Abschnitte beinhaltet eine Auflistung der im Pflichtenheft definierten und nach der Spezifikation implementierten Funktionen. Außerdem beinhalten diese Abschnitte eine Auflistung der Funktionen, die implementiert wurden und Abweichungen zu der Spezifikation aufweisen. Schließlich wird auf die Funktionen eingegangen, die im Pflichtenheft definiert und nicht implementiert wurden.

Der dritte Teil beinhaltet zum einen die Dokumentation der Maßnahmen des Managements und zum anderen Erfahrungsberichte der Projektmitglieder. Im Abschnitt über die Maßnahmen werden die im Projektplan [33] definierten Maßnahmen und ihre Umsetzung während des Projekts beschrieben. Die Maßnahmen beziehen sich auf das Projektmanagement, das Teammanagement, das Qualitätsmanagement und das Risikomanagement für die Projektgruppe. Die Erfahrungsberichte fassen die während des gesamten Projekts gemachten Erfahrungen der Projektteilnehmer zusammen.

Teil I.

Dokumentationen und Compliance

1. Technische Dokumentation

1.1. Cloud

Autor: VS

1.1.1. Verwendete Cloudtechnologie

Die Cloud wurde als eine *Private Cloud* mit der Open-Source Cloud Computing Software *Eucalyptus Community Cloud*[1] (ECC) in der Version 2.0.3 realisiert. Dies ist eine Infrastructure as a Service-Cloud (IaaS) und bietet für den Anwender einen Zugriff auf das System ab der Betriebsystemebene. Das bedeutet, dass der Anwender beliebig virtuelle Server starten kann auf denen ein Betriebsystem läuft, auf welches er vollen Zugriff hat. Die ECC besteht aus den folgenden Komponenten:

- Cloud Controller (CLC)
- Walrus
- Storage Controller (SC)
- Cluster Controller (CC)
- Node Controller (NC)

Der Cloud Controller ist für die Verwaltung der Ressourcen zuständig und kontrolliert die gesamte Cloud. Es gibt deshalb genau einen CLC in einer ECC. Die Cloud kann aus mehreren *Clustern* bestehen, die sich an verschiedenen Standorten befinden um eine höhere Verfügbarkeit zu gewährleisten. Ein *Cluster* wiederum besteht aus mehreren Knoten *(Nodes)*, die über ein lokales Netzwerk verbunden sind. Auf jedem Knoten ist ein Node-Controller und ein Hypervisor installiert. Der Node-Controller steuert den Hypervisor mit dessen Hilfe virtuelle Maschinen auf einem Knoten gestartet werden können. Für diese virtuellen Maschinen muss ein Image erstellt werden, von dem gebootet werden kann. Dieses Image besteht aus drei Teilen: Dem Betriebsystemkernel, der Ramdisk und dem *Eucalyptus Machine Image* (EMI), das aus der System-Partition einer Betriebssystem-Installation besteht. Über den Objekt-Speicher *Walrus*, welches Objekte in sogenannten *Buckets* speichert, müssen diese drei Teile in die Cloud geladen und dann beim CLC registriert werden. Danach kann der Benutzer über den CLC Instanzen eines virtuellen Servers starten, der diese dann auf die vorhandenen Cluster verteilt.

[1] http://open.eucalyptus.com/

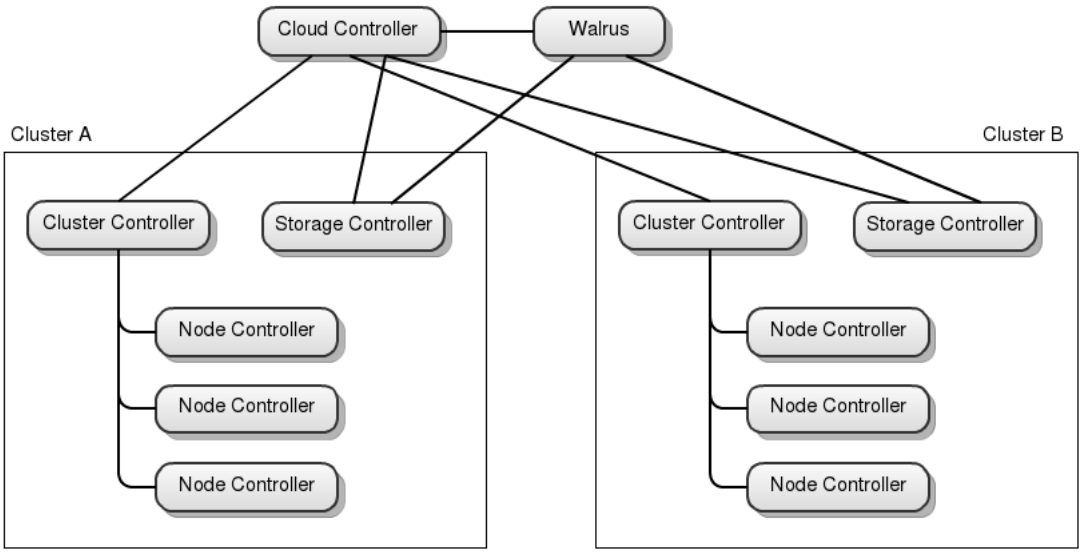


Abbildung 1.1.: Eucalyptus Architektur [20]

Dazu sendet jeder NC Informationen über seine freien Ressourcen an seinen CC, der diese dann an den CLC weiterreicht. Um Daten persistent zu speichern, wird ein *Elastic Block Storage* (EBS) verwendet, das einer Instanz dynamisch zugeordnet werden kann und welches vom *Storage Controller* verwaltet wird. Ein EBS kann wie eine Festplatte in einer Instanz verwendet werden, indem auf ihm ein Dateisystem erzeugt wird um es dann in der virtuellen Maschine einzubinden. Außerdem ist es möglich den aktuellen Zustand eines EBS zu sichern, indem von ihm ein sogenannter Snapshot erstellt wird. Dieser Snapshot kann dann wieder als eigenes EBS verwendet und zu einer beliebigen Instanz hinzugefügt werden.

1.1.2. Banking-Umgebung

Die Hardware für die Cloud-Umgebung besteht aus drei Servern auf denen ein Standard Debian-Linux [2] (Version 6) sowie die ECC-Software installiert ist. Einer der Server dient als Cloud- sowie als Cluster-Controller. Da nur ein Cluster verwendet wird, ist es nicht notwendig, einen separaten Server als Cluster Controller einzusetzten. Auf diesem Server laufen ebenso die Walrus-Komponente und der Storage Controller. Die beiden anderen Rechner werden als Node-Controller verwendet. Als EMI wird ein Ubuntu-Server Image eingesetzt, da für dieses regelmäßig Sicherheitsupdates erscheinen.

Die Banking-Software und die dazugehörigen Komponenten wurden auf einem EBS installiert, so dass auch alle Einstellungen und Daten, die bei der Ausführung entstehen,

[2] http://www.debian.org/

auf das EBS geschrieben werden. Dies hat den Grund, dass die Daten einer Instanz bei ihrer Terminierung oder bei einem Absturz verloren gehen. Das EBS existiert aber unabhänging von der Instanz weiter und somit auch die Daten der Banking-Software. Außerdem hat es den Vorteil, dass die cloudeigene Backup-Funktion für die Sicherung des EBS genutzt werden kann. Dazu wird automatisch täglich ein Snapshot des EBS erstellt. Es besteht aber auch die Möglichkeit ein Backup manuell anzustoßen, dies kann über den Cloud-Controller sowie über das Banking-System geschehen.

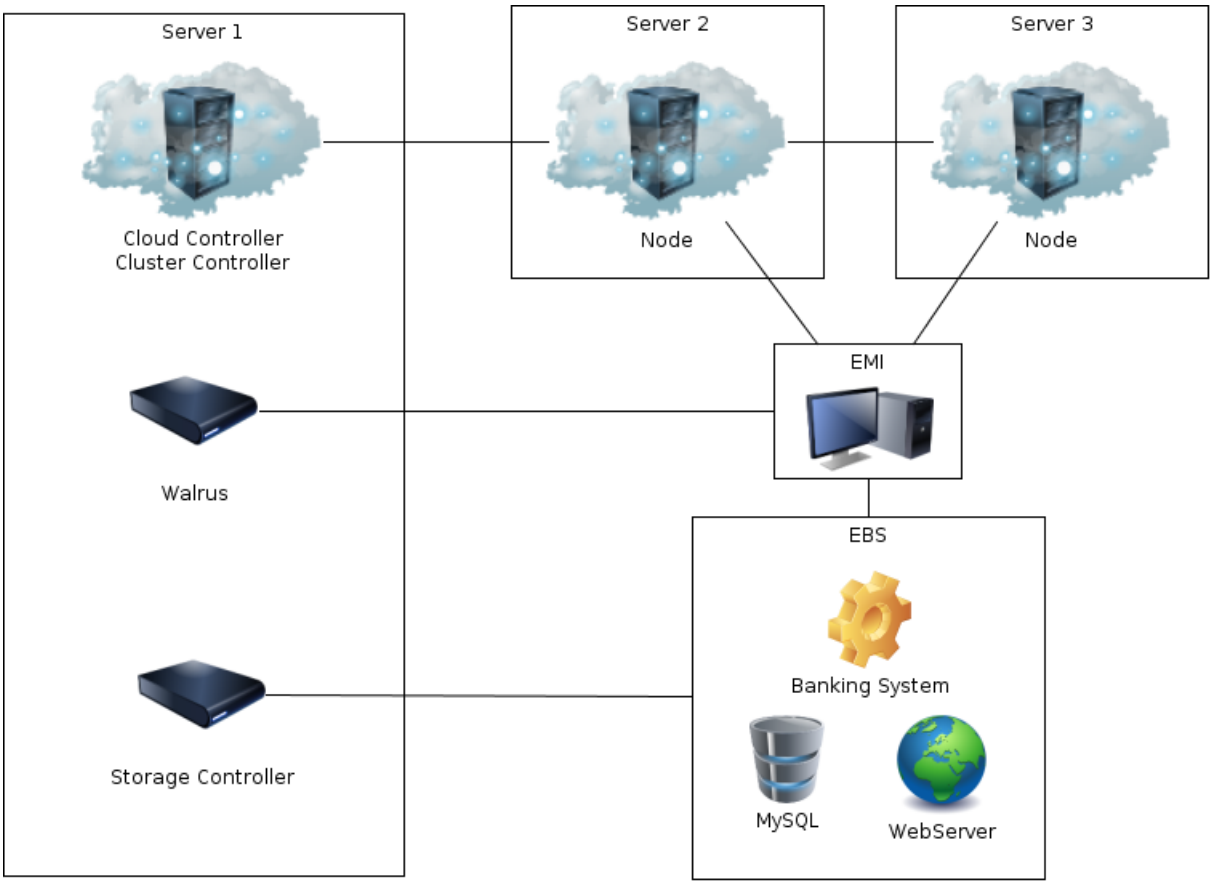


Abbildung 1.2.: Konkrete Umsetzung der Banksystemumgebung

1.1.3. Anwendung

Um mit der Cloud zu interagieren wurden die *Euca2ools*[19] eingesetzt. Dies ist eine Sammlung von Komandozeilenwerkzeugen, die vom Eucalyptus Team angeboten werden. Für den Zugriff auf die Cloud, ist eine Registrierung beim Cloud-Controller notwendig. Dazu besucht man die URL des Cloudservers (https://129.217.47.70:8443/) und klickt auf “Apply for Account” und gibt seine Daten ein. Daraufhin erhält der Administrator

eine Meldung und kann den Benutzer freischalten, welcher sich dann in der Weboberfläche einloggen kann. Danach kann unter dem Punkt “Credentials” ein Zertifikat für den Benutzer heruntergeladen werden, das von den Euca2ools verwendet wird, um sich bei der Cloud zu authorisieren und über eine verschlüsselte Verbindung mit ihr zu kommunizieren. Die Euca2ools können dann verwendet werden um Ressourcen abzufragen und Instanzen hochzuladen und laufen zu lassen.

1.1.3.1. Informationen abfragen

Mit den folgenden Befehlen lassen sich Informationen über laufende Instanzen, verfügbare Images, verfügbare Ressourcen und vorhandene Backups abfragen:

- euca-describe-instances: Zeigt Informationen wie Instanz-ID, IP-Adressen und Status an.
- euca-describe-images: Zeigt verfügbare Images, Image-IDs, Bucket- und Objektnamen an.
- euca-describe-availabitlity-zones: Zeigt die verfügbaren Cluster und deren freie Ressourcen an.
- euca-describe-snapshots: Zeigt die vorhandenen Backups, sowie deren IDs, Erstellungszeitpunkt und Status an.

1.1.3.2. Instanzen verwalten

Mit den folgenden Befehlen werden Instanzen gestartet, gestoppt und Adressen reserviert und zugeteilt:

- euca-run-instances: Startet eine Instanz eines Images bei der Parameter wie Größe, Cluster, IP-Adresse, Kernel und Ramdisk spezifiziert werden können.
- euca-terminate-instance: Stoppt laufende Instanzen.
- euca-allocate-address: Reserviert eine öffentliche IP-Adresse für den Benutzer.
- euca-associate-address: Weist einer Instanz eine IP-Adresse zu.

1.1.3.3. Images verwalten

Mit den folgenden Befehle werden Images in die Cloud hochgeladen und beim Cloud-Controller registriert:

- euca-bundle-image: Verbindet ein Image mit einem Benutzet und einem Zertifikat.
- euca-upload-bundle: Lädt ein gebündeltes Image in die Cloud.
- euca-register: Registriert ein gebündeltes Image bei der Cloud.

1.1.3.4. EBS verwalten

Mit den folgenden Befehlen wird ein EBS-Volume erstellt, gelöscht, einer Instanz zugwiesen und entfernt sowie Backups davon erstellt und gelöscht:

- euca-create-volume: Erstellt ein EBS-Volume in dem spezifizierten Cluster.
- euca-delete-volume: Löscht ein EBS-Volume.
- euca-attach-volume: Bindet ein EBS-Volume in einer Instanz ein.
- euca-detach-volume: Entfernt ein EBS-Volume aus einer Instanz.
- euca-create-snapshot: Erstellt einen Snapshot eines Volumes.
- euca-delete-snapshot: Löscht einen Snapshot.

Weitere Kommandos und Informationen zur Verwendung der Eucalyptus-Cloud finden sich im Eucalyptus User's Guide [21].

1.2. Datenbank

Autor: JK

1.2.1. Vorbetrachtung

In einer Cloud-Umgebung ist die Verfügbarkeit eines Systems nicht vollständig gewährleistet. Fällt das System aus, so muss sichergestellt sein, dass alle vor dem Ausfall existierende Informationen gespeichert wurden und durchgeführte Aktionen nachvollzogen werden können. Bei dem entwickelten Internetbanking System liegt zusätzlich das Problem vor, dass bei einem Datenverlust oder bei einer Dateninkonsistenz ein finanzieller Nachteil entstehen kann. Aus diesen Gründen ist eine persistente Datenhaltung unerlässlich und wurde durch eine relationale Datenbank realisiert. Die Wahl eines relationalen Datenbankmanagementsystems (DBMS) fiel auf das *MySQL*-DBMS, da es unter einer GPL-Lizenz verbreitet wird und dazu deshalb sehr viele frei verfügbare Dokumentationen und Anleitungen existieren.

1.2.1.1. Begriffe

In diesem Unterabschnitt sollen die wichtigsten Begriffe zu der verwendeten Datenbank vorgestellt werden, um das Verständnis des Abschnitts zu fördern. Es sei angemerkt, dass einige Begriffe sich mit denen in weiteren Kapiteln überschneiden , aber eine unterschiedliche Bedeutung besitzen können [35].

Datenbank Eine Ansammlung von Tabellen und Views, die miteinander in Beziehungen stehen.

Datensatz Daten einer Tabelle, die zusammen eine Einheit bilden.

Fremdschlüssel Eine Referenz von einer Tabelle auf einen Datensatz einer anderen Tabelle mit dem Ziel eine Beziehung zwischen den Tabellen sicherzustellen.

Normalisierung Methoden, die auf eine Datenbank und ihre Tabellen angewendet werden, um eine einfache Handhabung zu gewährleisten. So können z.B. doppelte Einträge ausgeschlossen werden.

Primärschlüssel Eine Menge von Feldern, die den Datensatz eindeutig identifiziert.

Tabelle Eine Ansammlung von Daten, die gleich aufgebaut sind und zeilenweise angeordnet werden.

Transaktion Eine atomare Aktion innerhalb der Datenbank, die aus mehreren Anfragen bestehen kann.

Schema Die Festlegung, welche Form die Daten in einer Datenbank haben müssen und welche Beziehungen zwischen den Daten bestehen.

Sicht oder View Eine Sammlung von Datensätzen bezüglich einer bestimmten Anfrage, die gespeichert und wie in einer Tabelle eingesehen werden kann. Der Zweck ist die Wiederverwendung von Anfragen und die Einsparung häufiger Datenbankzugriffe.

1.2.1.2. Einordnung ins System

Für das System wurden zwei Datenbankschemas vorgesehen, die zwei unterschiedlichen Zwecken dienen sollen. Die erste Datenbank (`banksystem`) ist ein wichtiger Teil des Systems. Alle Aktionen, die durch Benutzer des Systems durchgeführt werden, spiegeln sich in der Datenbank wider. Das Datenmodell des Systems wird dabei auf die Tabellen abgebildet. Die zweite Datenbank (`stammdaten`) dient ausschließlich der Kommunikation zu der BaFin und BZSt. Diese sollen über eine sichere SQL-Schnittstelle auf die Datenbank zugreifen und so unbemerkt bestimmte Daten einsehen [32]. Beide Datenbanken befinden sich auf einem Datenbankserver, wobei durch geeignete Authentifizierungs- und Autorisierungsmaßnahmen (siehe Abschnitt 1.2.2) die einzelnen Datenbanken voneinander getrennt werden. Diese Vorgehensweise hat den entscheidenden Vorteil, dass zwei physikalisch isolierte Datenbankserver die doppelte Speicherkapazität benötigen würden. Außerdem wird die Wartung der Server erschwert, was gleichzeitig zu Inkonsistenzen zwischen den beiden Datenbanken führen könnte.

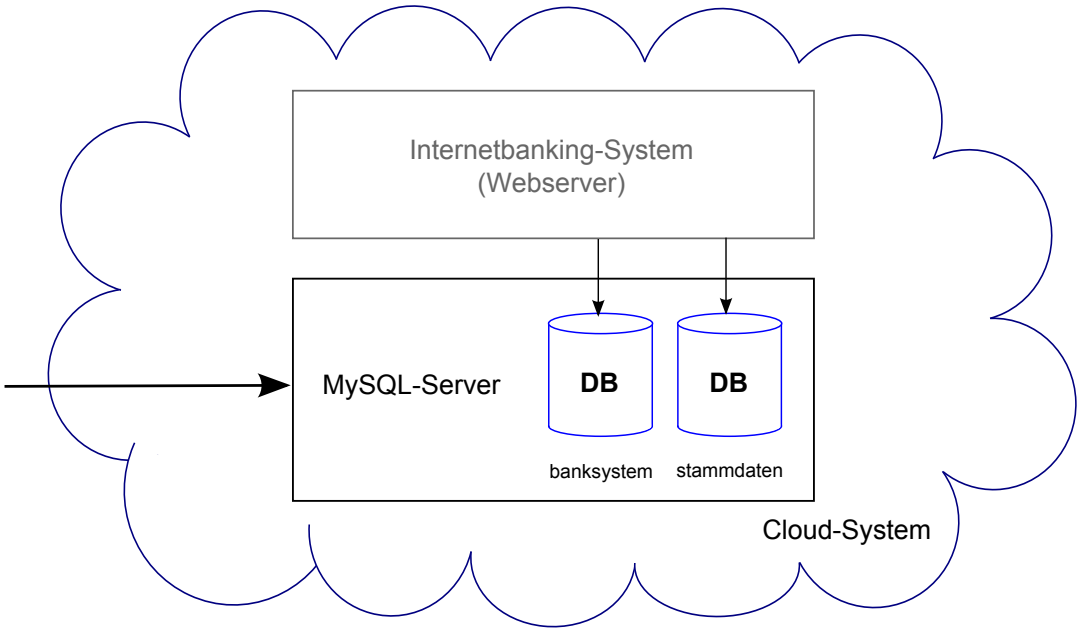


Abbildung 1.3.: Einordnung der Datenbank innerhalb des Cloud-Systems

Abbildung 1.3 stellt das Cloud-System abstrakt dar. Man erkennt, dass innerhalb der

Cloud die Anwendung ausgeführt wird, wobei diese auf zwei voneinander unabhängige Ebenen verteilt wird, die von zwei logischen Servern verwaltet werden, dem Webserver (siehe Abschnitt 1.3) und auf den MySQL-Server bzw. Datenbankserver. Die beiden Datenbanken `banksystem` und `stammdaten` wurden auf dem MySQL-Server aufgesetzt und können über eine Schnittstelle sowohl vom System, als auch von außen zugegriffen werden. An dieser Stelle wird der Datenbankserver mit den vorhandenen Datenbanken betrachtet und auf ihre Eigenschaften näher eingegangen.

1.2.2. Zugriff auf die Datenbank

Dieser Unterabschnitt behandelt die wesentlichen Vorbedingungen, um mit der Datenbank arbeiten zu können. Dazu zählt die Verbindung zur Datenbank, die Authentifizierung anhand von Rollen und die Autorisierung der einzelnen Rollen. Aus Sicherheitsgründen werden keine Passwörter angegeben. Diese müssen bei den Mitarbeitern der PG555 angefragt und über einen sicheren Kanal übertragen werden.

1.2.2.1. Verbindung zur Datenbank

Wie in Abbildung 1.3 dargestellt, existieren eine Möglichkeit der Verbindung zur Datenbank von außen. Dies ist vor allem für den Datenbankadministrator notwendig, da dieser unabhängig vom Internetbankingsystem verschiedene Arbeiten durchführen soll. Um die Datenbank also von außen ansprechen zu können benötigt man die folgenden Verbindungsdaten:

Host: `129.217.47.73`
Port: `3306`

Als graphische Werkzeuge für die Verwaltung der Datenbank von außen haben sich MySQL Workbench [25] und MySQL GUI Tools [23] als geeignet erwiesen. Es kann aber auch eine Konsolen-Verbindung über MySQL-Client hergestellt werden.

1.2.2.2. Rollen in der Datenbank

Es existieren insgesamt drei Rollen, die auf die Datenbank zugreifen können, um mit dieser zu interagieren. Diese sind:

- BaFin und BZSt
- Banksystem
- Datenbank-Administrator

BaFin und BZSt benötigen einen Zugriff auf die Kundenstammdaten der Bank [32] in `stammdaten` über eine SQL-Schnittstelle. Der Zugriff soll für den Kunden unbemerkt

sein und passiert außerhalb des Internetbankingsystems direkt auf den Datenbankserver. Die Authentifizierung erfolgt passwortbasiert, muss aber für einen sichere Verbindung zertifikatbasiert sein.

Das **Banksystem** befindet sich auf einem Webserver *Glassfish* (siehe Abschnitt 1.3). Dieser stellt eine sichere Verbindung zu der Datenbank her und authentifiziert sich mit Hilfe eines Benutzernamens und eines Passworts (siehe Abschnitt 1.3.1). Dabei muss das System sowohl auf die Datenbank `stammdaten`, als auch auf `banksystem` einen Zugriff erhalten, um die Daten aktuell und konsistent zu halten. Die Rolle *Banksystem* ist also vor allem für die Verwaltung der Daten innerhalb der Datenbanken zuständig.

Datenbank-Administrator hat die Aufgabe die Wartung durchzuführen und damit die Verwaltung des gesamten Datenbankservers zu übernehmen. Dazu gehört nicht nur Benutzerverwaltung und Rechtevergabe, sondern auch das Anlegen neuer Tabellen oder Datenbanken, falls diese im weiteren Betrieb des Systems benötigt werden. Dazu verbindet er sich von außen direkt auf den Datenbank-Server und kann die Aufgaben unabhängig vom Banksystem erledigen.

Tabelle 1.1 stellt die Rollen in einer Übersicht dar, indem die Benuzter innerhalb von MySQL-DMBS und ihre Authentifizierungsart aufgelistet werden. Aus Vertraulichkeitsgründen wird bei passwortbasierter Authentifizierung das dazugehörige Passwort nicht angegeben. Diese werden den Parteien über einen sicheren Kanal übermittelt.

Tabelle 1.1.: Übersicht über die Rollen und die Authentifizierungsart

Benutzer	**Rolle**	**Authentifizierung**
`bafinbzst`	BaFin und BZSt	passwortbasiert
`system`	Banksystem	passwortbasiert
`root`	Datenbank-Administrator	passwortbasiert

1.2.2.3. Rechte der Benutzer

Sind die Benutzer authentifiziert, erfolgt die Autorisierung nach einem rollenbasierten Verfahren. Tabelle 1.2 stellt dazu die Benutzer mit ihren Rechten dar. Die Rechte richten sich nach den SQL-Anfragen, die der jeweilige Benutzer ausführen darf. Es gilt zudem das Prinzip, dass alle Aktionen, die nicht explizit erlaubt, verboten sind.

Der Benutzer `bafinbzst` darf demzufolge auf die Daten in der Datenbank `stammdaten` nur lesend zugreifen, `system` erhält Schreib- und Leserechte an `stammdaten` und `banksystem`, während `root` einen Vollzugriff auf den Datenbankserver bekommt und alle verfügbaren Aktionen auf allen Datenbanken (auch Verwaltungsdatenbanken) bearbeiten darf.

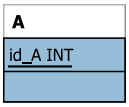
Tabelle 1.2.: Übersicht über die Autorisierung der Benutzer auf verschiedenen Datenbanken

Benutzer	**Datenbanken**	**Rechte**
`bafinbzst`	`stammdaten`	`SELECT`
`system`	`stammdaten`, `banksystem`	`SELECT, INSERT, UPDATE, DELETE`
`root`	alle	alle

An dieser Stelle soll erwähnt sein, dass obwohl der Datenbank-Administrator (`root`) alle Rechte hat, dennoch das BDSG und LDSG gilt [9][5]. Der Administrator unterliegt einer Schweigepflicht nach §5 BDSG und muss im Falle einer Aufragsdatenverarbeitung nach §11 BDSG einem schriftlichen Vertrag zustimmen (siehe Kapitel 2).

1.2.3. Aufbau

Im Folgenden wird der Aufbau der beiden Datenbanken `banksystem` und `stammdaten` beschrieben. Dazu wurden bestehende Tabellen der Datenbanken in ein Diagramm eingefügt und die Beziehungen durch Referenzierung der Schlüssel hergestellt. Die Diagramme wurden aus dem MySQL Workbench [25] exportiert und haben die folgende Syntax.



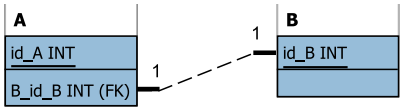
Eine Tabelle bzw. Entität, die Datenfelder, Primärschlüssel oder Fremdschlüssel enthält.

id_A

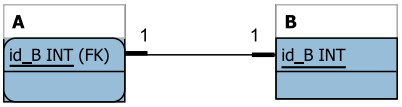
Datenfeld, das als Primärschlüssel oder als Teil des Primärschlüssels dient.

(FK)

Fremdschlüssel, der nicht Teil des Primärschlüssels ist.



Beziehung zwischen Tabelle A und Tabelle B. Fremdschlüssel aus A referenziert den Primärschlüssel oder Index der Tabelle B, ist aber **kein Teil des Primärschlüssels** in Tabelle A.



Beziehung zwischen Tabelle A und Tabelle B. Fremdschlüssel aus A referenziert den Primärschlüssel oder Index der Tabelle B und ist **ein Teil des Primärschlüssels** in Tabelle A.

1.2.3.1. Datenbank: banksystem

Die Datenbank `banksystem` bietet die Grundlage für das gesamte System. Darin werden alle Daten des Systems abgelegt, die persistent gehalten werden müssen, um diese bei einem Systemabsturz wiederherstellen zu können. Die Tabellen basieren auf dem verwendeten Modell, so dass eine Abbildung der Objekte im Programm unkompliziert auf die Tabellen durchgeführt werden kann. Da es sich um relationale Datenbanken[59] handelt, wurden die Tabellen einer Normalisierung bis zur dritten Normalform unterzogen. Im Folgenden werden einige für wichtig erachtete Ausschnitte aus der Datenbank `banksystem` vorgestellt.

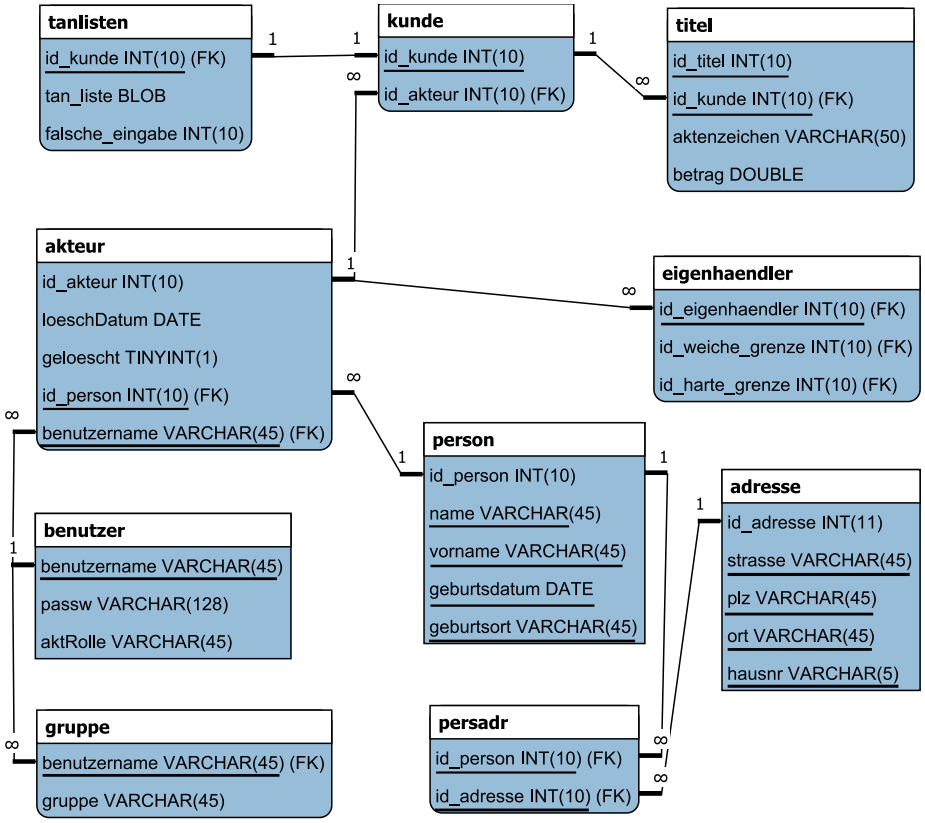


Abbildung 1.4.: Tabellenstruktur für die Akteure der Bank

In Abbildung 1.4 werden die Akteure der Bank betrachtet. Um einen Akteur im System

darstellen zu können, benötigt man persönliche Daten, die in der Tabelle `person` stehen. Um eine Person eindeutig zu identifizieren, wurde ein zusammengesetzter Schlüssel aus vier Attributen *name, vorname, geburtsdatum* und *geburtsort* prototypisch festgelegt. Da eine Person mehrere Adressen haben kann, wurde eine `n:m`-Beziehung zwischen der Tabelle `person` und der Tabelle `adresse` angenommen, die wiederum eine besondere Tabelle `persadr` benötigt, um die Personen mit den Adressen zu verknüpfen. Man beachte, dass in diesem Fall eindeutige Indizes *id_person* und *id_adresse* als Fremdschlüssel verwendet werden und nicht die Primärschlüssel selbst. Zusätzlich zu persönlichen Daten, muss sich ein Akteur im System anmelden können. Dazu benötigt er einen Benutzernamen, einen Passwort und eine Gruppenzugehörigkeit, wonach die Autorisierung erfolgt. Der Benutzername und das Passworthash wird in der Datenbank `benutzer` abgelegt, während sich die Gruppen eines Benutzers in der Tabelle `gruppe` befinden. Für die

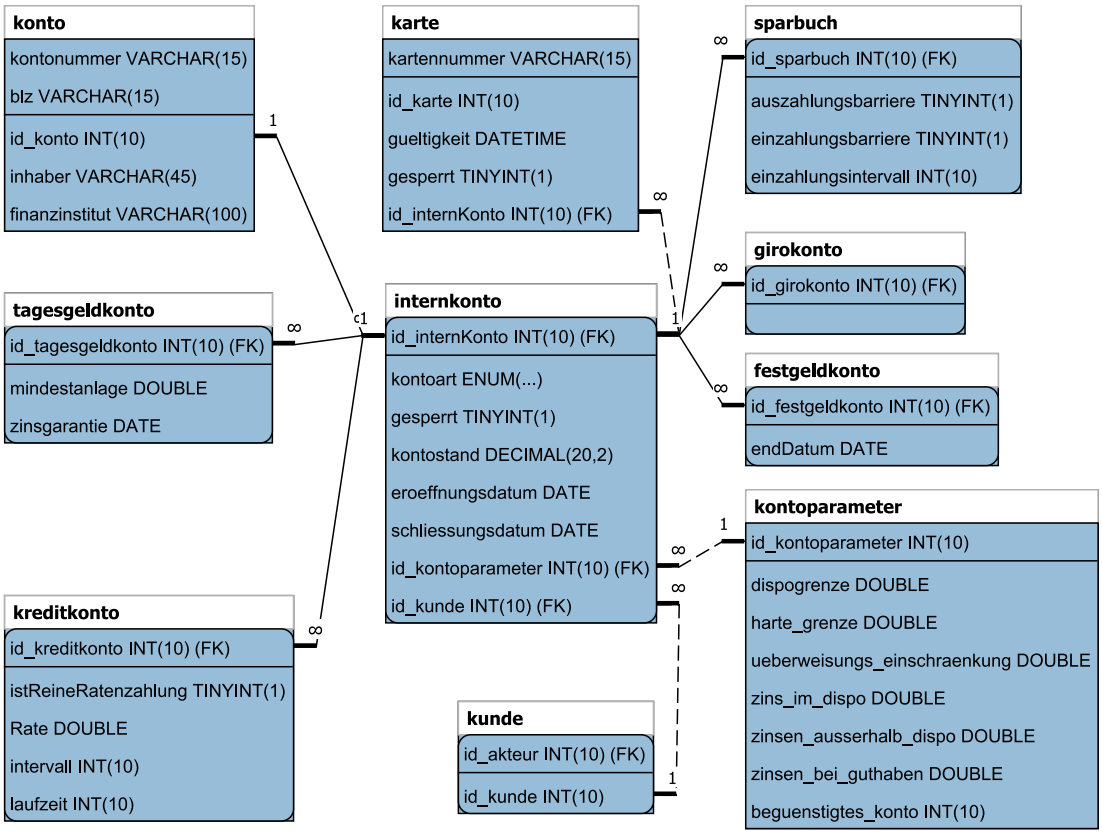


Abbildung 1.5.: Tabellenstruktur für die Konten der Bank

Identifikation eines Akteurs benötigt man also seine persönlichen Daten und den Benutzernamen. Dies liegt vor allem daran, dass die Mitarbeiter der Bank ebenfalls Kunden der Bank sein können und man gern eine Isolierung zwischen Kunden und Mitarbeitern hätte. Da Kunden zusätzlich noch eine Kundennummer in der Bank erhalten sollen,

wurde die Tabelle `kunde` angelegt, die alle Akteure aus der Tabelle `akteur` mit der Gruppe *Kunde* auf eine eindeutige Kundennummer abbildet. Um die Akteure der Bank mit wenigen Datenbankanfragen auslesen zu können, wurden die Views `view_kunde` und `view_akteur` angelegt.

Abbildung 1.5 stellt die Kontokomponente des Systems dar. Dabei ist das Hauptelement die Tabelle `konto` dieses ist vor allem dafür zuständig die wichtigsten Merkmale eines Kontos zu erfassen. Da sowohl interne, als auch externe Konten verarbeitet werden, unterscheidet man diese in der Datenbank, indem die Tabelle `internkonto` verwendet wird. Darin werden detaillierte Informationen zu Konten innerhalb der Bank gesammelt, während Konten anderer Banken in der Tabelle `konto` gespeichert werden. Insbesondere unterscheidet man bei internen Konten das Giro-, das Tagesgeld-, das Festgeld-, das Kreditkonto und das Sparbuch. Da diese verschiedene Eigenschaften aufweisen wird je eine Spezialisierungstabelle für diese Kontoarten angelegt z.B. Tabelle `sparbuch`. Dabei verweist der Primärschlüssel jeder dieser Spezialisierungstabelle auf den Primärschlüssel von `internkonto` *id_internkonto*.

Für den Geldtransfer von einem Konto auf ein anderes Konto sind die Tabellen in Abbildung 1.6 zuständig. Dabei wird die Tabelle `transaktion` betrachtet, die alle gemeinsamen Eigenschaften von allen Geldtransaktionen umfasst. Dazu zählen das Zielkonto, das Quellkonto, der Betrag, der Verwendungszweck und das Datum einer Geldtransaktion. Diese Attribute bilden auch den Primärschlüssel und identifizieren eine Transaktion, wobei das Datum mit der sekundengenauen Uhrzeit abgelegt wird. Zu den Geldtransaktionen gehören Überweisungen, Lastschriften, Daueraufträge und Barzahlungen, die je eine Spezialisierungstabellen besitzen. Zwischen Dauerauftrag und Überweisung existiert eine `n:m`-Beziehung und die zuständige Tabelle `dauerauftragzuueberweisung` enthält alle Überweisungen zu einem Dauerauftrag.

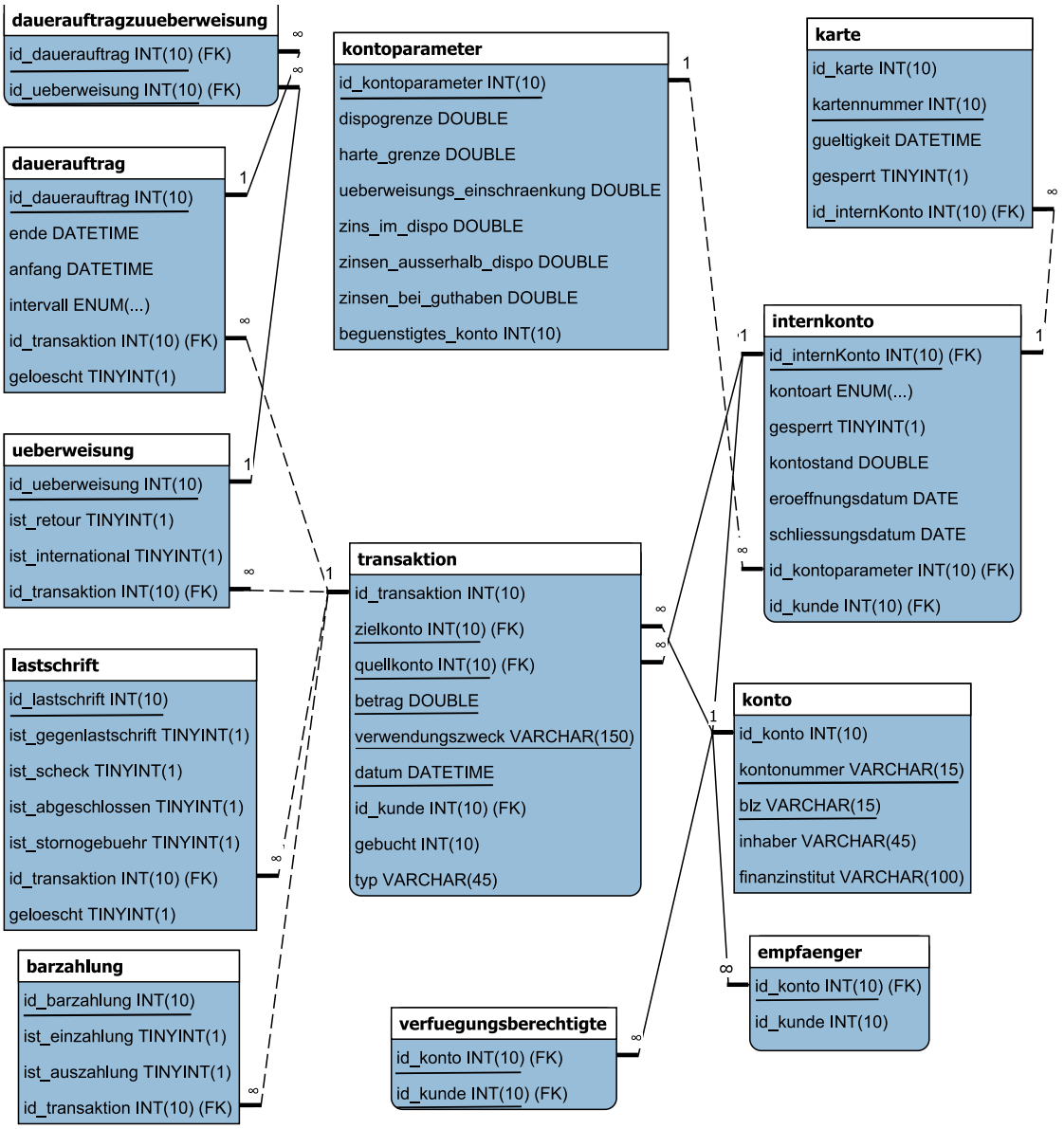


Abbildung 1.6.: Tabellenstruktur für die Geldtransaktionen der Bank

1.2.3.2. Datenbank: stammdaten

Die Datenbank für BaFin und BZSt wurde bereits von dem Auftraggeber vorgegeben und ist so auch im System hinzugefügt worden. Es werden dabei nur Stamminformationen der Konten in dieser Datenbank gespeichert, wobei nach dem Schließen eines Kontos die Daten noch drei Jahre in dieser Datenbank gespeichert bleiben bis sie gelöscht wer-

den müssen.

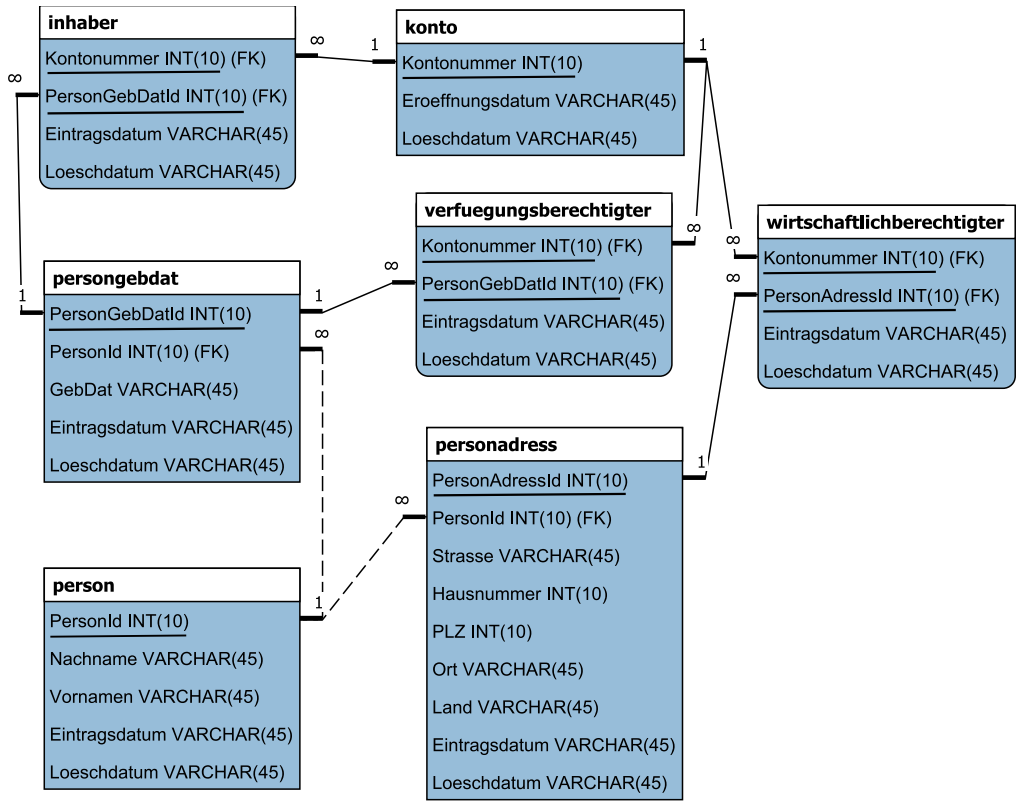


Abbildung 1.7.: Tabellenstruktur für die Datenbank stammdaten

Abbildung 1.7 zeigt die Tabellen der Datenbank `stammdaten`. In jeder Tabelle wird das Eintragsdatum und das Löschdatum des zugehörigen Wertes gespeichert. Die Person wird dabei durch einen Primärschlüssel *PersonId* identifiziert, der auch als Fremdschlüssel in den Tabellen `persongebdat` und `personadress` gespeichert wird. Eine besondere Unterscheidung gibt es zwischen einem wirtschaftlich Berechtigten und einem Verfügungsberechtigten eines Kontos, wobei die erste Gruppe den Inhaber und alle Personen beinhalten, die über das Konto verfügen können.

Eine vollständige Auflistung aller Tabellen der beiden Datenbanken befindet sich im Anhang B

1.3. Glassfish

Autor: JK

Bei der an dieser Stelle vorgestellten Internetbanking Software handelt es sich um eine *Java Enterprise Edition (J2EE)*-Anwendung. Jede solche Anwendung benötigt einen *Anwendungsserver* (engl. *Application Server*), auf dem diese ausgeführt wird. Dieser hat außerdem den Vorteil, dass er bereits einige Schnittstellen anbietet, die zum Beispiel den Datenbankzugriff oder die Sicherheit vereinfachen. Der hier verwendete Anwendungsserver ist *Glassfish* [42], wobei es sich um einen Open Source Server handelt, der bereits innerhalb der Entwicklungsumgebung *NetBeans* [26] integriert ist. Im weiteren Verlauf wird auf die Konfiguration des Servers eingegangen.

Um Glassfish konfigurieren zu können, muss dieser gestartet und die **Administrationskonsole** aufgerufen werden, die unter folgender Adresse im Browser zu erreichen ist:

```
http://129.217.47.73:4848
```

Die Konsole ist passwortgeschützt und die Zugangsdaten können bei dem zuständigen Administrator angefragt werden.

Die durchzuführenden Konfigurationen beziehen sich auf den Datenbankzugriff und Sicherheitsmechanismen, die angeboten werden.

1.3.1. Java Database Connectivity(JDBC)

Im Abschnitt 1.2 wurden Datenbanken auf einem MySQL-Server erstellt. Allerdings muss die Datenbank dem Webserver bekannt sein, um darauf zugreifen zu können. *Java Database Connectivity (JDBC)* ist eine Schnittstelle zwischen der SQL-Datenbank und der Anwendung, die sie benutzen möchte [52]. MySQL-Datenbank bietet bereits einen JDBC-Treiber, der auf der Herstellerseite heruntergeladen werden kann [24] und der ins Glassfish-Bibliotheksverzeichnis kopiert werden muss.

1.3.1.1. Verbindungspool erstellen

Zusätzlich dazu muss die Datenbank-Verbindung in der Administrationskonsole von Glassfish konfiguriert werden. Dazu öffnet man die Konsole und wählt in der Navigationsleiste links **Ressourcen** → **JDBC** → **JDBC-Verbindungspool** aus. Anschließend werden alle vorhandenen Pools im Hauptfenster aufgelistet. Falls der zu konfigurierende Pool noch nicht vorhanden ist, wählt man die Option *Neu* mit folgenden Daten aus :

Poolname `MySQLPool` (willkürlich wählbar)

Ressourcentyp `javax.sql.DataSource`

Hersteller `MySQL`

Im Fenster, das nach dem Auswählen von *Weiter* folgt, müssen folgende Eigenschaften eingetragen werden. Dabei muss auf die Groß- und Kleinschreibung geachtet werden.

Port `3306`

ServerName `129.217.47.73`

DatabaseName `banksystem`

User `system`

Password *Aus Sicherheitsgründen entfernt*

Url `jdbc:mysql://129.217.47.73:3306/banksystem`

URL `jdbc:mysql://129.217.47.73:3306/banksystem`

Sind alle Eigenschaften eingetragen, kann zusätzlich die Datenbankverbindung mit einem Ping getestet werden.

Zusätzlich muss die obere Prozedur für die weiteren, vorhandenen Datenbanken durchgeführt werden. Bevor man aber auf diese Datenbanken in der Anwendung zugreifen kann, muss je eine Datenquelle (engl. *Datasource*) angelegt und im System bekannt gemacht werden.

1.3.1.2. Datasource erstellen

In der Administrationskonsole von Glassfish wählt man **Ressourcen → JDBC → Datasource** aus. Bei der Auflistung im Hauptfenster legt man eine neue Datasource an mit folgenden Eigenschaften:

JNDI-Name `jdbc/MySQLDataSource`

Poolname `MySQLPool` (Name des Verbindungspools unter 1.3.1.1)

Aktiviert Häkchen setzen

Für jeden Pool muss eine eigene Datasource angelegt werden und unterschiedliche JNDI-Namen vergeben werden, um diese im System ansprechen zu können.

Zusätzlich zur Administrationskonsole müssen die Datasourcen im Deployment Descriptor `web.xml` des Projekts bekannt gemacht werden. Dazu fügt man die folgenden Zeilen (für jede Datasource) in die Datei ein.

```
<resource-ref>
   <res-ref-name>
    <!--JNDI-Name der Datasource z.B.-->
    jdbc/MySQLDataSource
    </res-ref-name>
   <res-type>javax.sql.DataSource</res-type>
   <res-auth>Container</res-auth>
   <res-sharing-scope>Shareable</res-sharing-scope>
</resource-ref>
```

Außerdem muss der Server-Deployment Descriptor `glassfish-web.xml` um die folgenden Zeilen erweitert werden:

```
<resource-ref>
   <res-ref-name>
    jdbc/MySQLDataSource
   </res-ref-name>
   <jndi-name>jdbc/MySQLDataSource</jndi-name>
</resource-ref>
```

Nun kann auf die Datasource innerhalb der Glassfish-Anwendung zugegriffen werden.

1.3.2. Sicherheitsrealm

Einer der Vorteile an den Webservern bzw. Web-Containern wie Glassfish ist, dass diese bereits mit einigen Sicherheitsmechanismen ausgestattet sind, die zertifiziert sind und deshalb keine Verifikation benötigen.

Diese Mechanismen wurden innerhalb der Projekts dazu verwendet, den Login in das System zu ermöglichen. Als die Authentifizierungsart wurde das passwortbasierte Verfahren angewendet, in dem ein Benutzername und Passwort eingegeben werden, um in das Internetbanking-System einloggen zu können. Die Authorisierung kann ebenfalls durch eine Sicherheitsfunktion durchgeführt werden. Die Login-Informationen werden dabei über die Datenbank `banksystem` verifiziert. Dazu werden in der Datenbank zwei Tabellen benötigt, die Tabelle `benutzer` und die Tabelle `gruppe`. Sie speichern die Benutzerdaten, wie den Benutzernamen, den Passworthash und die Gruppenzugehörigkeit. Auf diese Tabellen greift Glassfish über einen sogenannten *Sicherheitsrealm* während des Login-Vorgangs zu und prüft die Übereinstimmung der eingegebenen mit der gespeicherten Anmeldedaten. An dieser Stelle wird nicht näher auf die Funktionsweise des Sicherheitsrealms eingegangen, sondern nur die Konfiguration dessen vorgestellt. Interessierter Leser wird auf das Buch *Java EE 5 development using GlassFish Application Server* [42] verwiesen.

Die Konfiguration des Sicherheitsrealms erfolgt in der Administrationskonsole von Glassfish. Hier wählt man in der Navigationsleiste **Konfigurationen** → **server-config** → **Sicherheit** → **Bereiche** aus. An dieser Stelle muss ein neuer Realm angelegt werden. Folgende Daten werden dabei in die Felder eingetragen und die übrigen leergelassen:

Name `rollen-realm`

Klassenname `com.sun.enterprise.security.auth.realm.jdbc.JDBCRealm`

JAAS-Kontext `jdbcRealm`

JNDI `jdbc/MySQLDataSource` (Datasource aus Abschnitt 1.3.1.2)

Benutzertabelle `benutzer`

Benutzerspalte *benutzername*

Passwortspalte *passw*

Gruppentabelle `gruppe`

Gruppenspalte *gruppe*

Digest-Algorithmus `MD5`

Codierung `Hex`

Damit ist die Konfiguration des Sicherheitsrealms abgeschlossen. Die Authentifizierung und Authorisierung erfolgt über die Deployment Deskriptoren `web.xml` und `glassfish-web.xml` in Verbindung mit dem Formular auf der Seite `login.xhtml` (siehe Kapitel 1.5).

1.4. Komponenten

1.4.1. Architektur

Autor: JK

In Abbildung 1.3 wurde das Internetbanking-System innerhalb des Cloud-Systems in zwei Schichten dargestellt, die Internetbanking-Schicht auf dem Webserver und die Datenbankschicht. Die Benutzer greifen über einen Browser auf den Webserver zu, der die für den Benutzer wichtige Daten anzeigt. Es handelt sich also um eine Client-Server-Interaktion, weshalb sich als Architekturmodell das *Client-Server-Modell* anbietet. Abbildung 1.8 betrachtet das Internetbanking-System als Client-Server-Modell, wo-

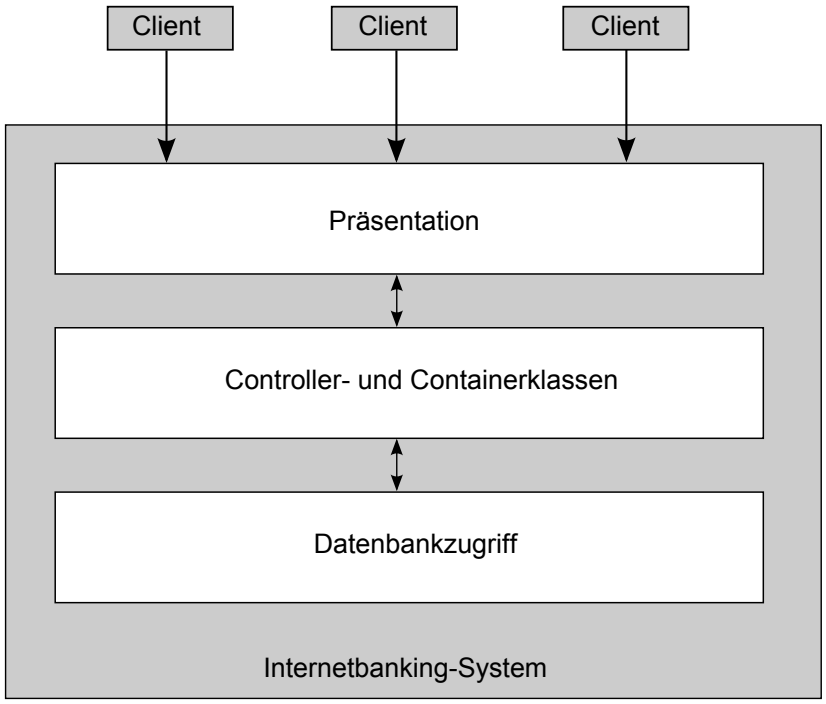


Abbildung 1.8.: Banksystem in einer Client-Server Architektur mit mehreren Schichten

bei die Server-Anwendung in Schichten aufgebaut ist. Die Clients greifen über den Browser auf die Präsentationsschicht und führen verschiedene Aktionen aus. Diese Aktionen werden von der Controller- und Containerschicht verarbeitet und falls notwendig in die Datenbank über die Datenzugriffsschicht gespeichert. Die Einzelnen Schichten werden im Folgenden genauer betrachtet und deren Umsetzung vorgestellt.

1.4.1.1. Präsentationsschicht

Bei der Präsentationsschicht handelt es sich um einen Teil des Entwurfsmusters *Model-View-Controller(MVC)* [53], wobei es sich um die View handelt [38]. Diese besteht aus zwei Teilen, den *JavaServer Faces* und den *ManagedBeans* [55].

JavaServer Faces

JavaServer Faces(JSF) sind ein Teil der J2EE [60] und werden für die Präsentation verwendet. Die früher benutzte Kombination aus Servlets und JavaServer Pages(JSP) wurde dabei erweitert und als eine übergeordnete Schicht die JSF eingeführt. JSF bietet viele Komponenten an, um die Benutzerschnittstelle zu erstellen, aber auch die Eingaben des Benutzers zu validieren oder in ein bestimmtes Format zu konvertieren ohne sich mit Java-Klassen zu beschäftigen. Die Implementierung von JSF erfolgt mit Hilfe von XHTML-Dateien, wobei es eigene Tags für JSF-Befehle existieren, die zur Darstellung von GUI-Elementen dienen. Außerdem wird eine *Expression Language (EL)* verwendet, die unter anderem dafür sorgt, dass Werte angezeigt, Methoden ausgeführt oder Berechnungen durchgeführt werden können. JavaServer Faces stehen in enger Verbindung mit den *ManagedBeans*, die vor allem für die Navigation unter den JSF-Seiten und für die Datenhaltung zuständig sind.

ManagedBeans

ManagedBeans sind normale Java-Klassen, die mit der Annotation `@ManagedBean` versehen werden. Damit dürfen alle JSF-Seiten auf die Methoden und Attribute der Bean zugreifen. Allerdings ist der Zugriff auf die Attribute nur über Getter- bzw. Setter-Methoden erlaubt, so dass eine Kapselung der Attribute möglich ist.

Innerhalb des Projekts wurde das PresentationModel- bzw. das ViewHelper-Muster angewandt [22], das eine Trennung von Daten, die innerhalb der View angezeigt werden und dem eigentlichen Modell (Containerklassen) fordert. Das PresentationModel (PM) besteht dabei aus mehreren ManagedBeans und befinden sich im Paket `de.layout`. Dazu zählen die Klassen:

- AufgabePM
- EingabeHandler
- KontoPM
- KundePM
- Navigation
- NavigationRollen
- NutzerPM
- PersonPM

- TransaktionPM

Als Beispiel soll die Klasse *EingabeHandler* betrachtet werden, dessen Methoden die komplexe Validierung der Eingabedaten übernehmen, die nicht innerhalb von JSF durchgeführt werden kann. Die Methode `validatePasswort` prüft zum Beispiel beim Anlegen eines Kunden oder eines Benutzers, ob das Passwort und seine Wiederholung übereinstimmen und die Mindestlänge von acht Zeichen besitzen.

1.4.1.2. Controllerschicht und Containerschicht

Die zweite Schicht beinhaltet die Controller und die Container des Systems, die gewöhnliche Java-Klassen sind, aber unterschiedliche Aufgaben haben. Beide dieser Klassenarten können sowohl mit der Präsentations-, als auch mit der Datenzugriffsschicht kommunizieren und sind deshalb auf der gleichen Ebene angesiedelt im Paket `de.modell`.

Controller
Das System besteht aus mehreren Controllern, die alle eine bestimmte Aufgabe haben. Der GeldtransferController zum Beispiel, der die Aufgabe hat aller Geldtransaktionen im System zu verwalten. Die verwendeten Controller sind:

- BoersenController
- GeldtransferController
- KontoController
- KommunikationsController
- KundenController
- Nutzerverwaltung
- SecurityController

Alle Anwendungsfälle, die das System enthalten muss [32], sind auf diese Controller verteilt und werden über die GUI und das PresentationModel ausgeführt.

Container
Die Container entsprechen dem Modell in dem MVC-Muster und dienen der Datenhaltung. Sie enthalten keine Programmlogik, sondern kapseln nur die Informationen zu verschiedenen Objekten im System. So stehen alle Daten eines Kunden in der Containerklasse `Kunde`. Muss ein Kunde eine Aktion ausführen, wird eine Instanz der Klasse mit allen wichtigen Informationen des Kunden erstellt und an die entsprechende Methode weitergeleitet.

1.4.1.3. Datanzugriffsschicht

Die Datenzugriffsschicht hat die Aufgabe eine Verbindung zur Datenbank aufzubauen und die notwendigen Informationen zu speichern oder auszulesen. Dabei wurde das DataAccessObject-Muster (DAO-Muster) verwendet [22], das die Datenbankanfragen in den entsprechenden DataAccess-Objekten umhüllt und eine Trennung zwischen der spezifischen Datenbankanfrage und dem System erreicht. Als Zugriff auf die DAOs wird eine MySQLDAOFabrik verwendet, die als Singleton [38] implementiert wurde. Die einzelnen DAOs sind keine Singleton-Objekte und können deshalb mehrmals im System vorkommen. Die implementierten DAOs befinden sich im Paket `de.dao` und sind im einzelnen folgende Klassen:

- AkteurDAO
- AufgabeDAO
- KontoDAO
- KundeDAO
- NutzerDAO
- PersonDAO

Da das System nicht komplett implementiert werden konnte, handelt es sich um eine unvollständige Liste der DAOs, die noch erweitert werden müsste.

1.5. Sicherheit

Autor: VM

1.5.1. Aufbau der Dokumentation

Die folgenden Abschnitte beinhalten die technische Dokumentation des Themas Sicherheit im Projekt. Die Verletzung von Sicherheitsinteressen wird dabei als Sicherheitsvorfall bezeichnet [39]. Es wird zunächst eine Zusammenfassung der Planung vorgestellt, die in dem Projektplan entwickelt wurde [33]. Es werden dazu die für das Bankensystem relevanten Sicherheitsinteressen, die identifizierten Risiken für Sicherheitsvorfälle und die Sicherheitsmaßnahmen zur Vermeidung dieser vorgestellt. Daraufhin wird die Umsetzung der Planung während des Projektes vorgestellt. Für jedes einzelne Sicherheitsinteresse werden die implementierten Maßnahmen beschrieben. Im Anschluss daran folgt ein Abschnitt über die technische Umsetzung der RSA-Schlüsselverwaltung [30], die für die Verwaltung der Sicherheitsmaßnahmen digitale Signierung und Verschlüsselung benötigt wird.

1.5.2. Planung der Sicherheit

Während der Planungsphase des Projektes wurden die folgenden Sicherheitsinteressen für das Bankensystem identifiziert, die bis auf die Anonymität, aus Biskup [30] entnommen sind:

- Anonymität
- Authentizität
- Integrität
- Nachweisbarkeit
- Verfügbarkeit
- Vertraulichkeit

Des weiteren wurden die folgenden Risiken für die Verletztung der Sicherheitsinteressen ermittelt:

- Die Anonymität kann durch die Weitergabe von personenbezogenen Daten (kurz: *Weitergabe*) verletzt werden
- Die Authentizität kann durch Vortäuschung einer falschen Identität (kurz: *Vortäuschung*) verletzt werden
- Die Integrität kann durch eine fehlerhafte Übertragung von Daten oder durch Manipulation dieser (kurz: *Manipulation*) verletzt werden

- Die Nachweisbarkeit kann durch den Verlust von Daten zum Nachhalten von fehlerhaften und spezifizierten Ereignissen (kurz: *Verlust der Nachvollziehbarkeit*) verletzt werden
- Die Verfügbarkeit kann durch den Ausfall von Funktionen des Systems oder des gesamten Systems (kurz: *Ausfall*) verletzt werden
- Die Vertraulichkeit kann durch das bereits erwähnte Risiko der Weitergabe personenbezogener Daten verletzt werden
- Das Risiko eines nicht-spezifizierten Verhaltens des Systems (kurz: *Nicht-spezifiziertes Verhalten*) hat nicht absehbare Folgen, weshalb die Betroffenheit aller Sicherheitsanforderungen bis auf die Nachweisbarkeit nicht ausgeschlossen werden kann

Während der Modellierung der Geschäftsprozesse des Bankensystems wurde auf die Modellierungsnotation EPK [50] zurückgegriffen, da dort die Modellierung von Risiken möglich ist. Durch eine Abbildung von den Sicherheitsanforderungen auf die Risiken, die in der Übergangsmatrix in Abbildung 1.9 zu sehen ist, konnten die Anforderungen implizit in die Prozessmodellierung integriert werden. Die Matrix beinhaltet größtenteils eine Eins-zu-Eins-Beziehung zwischen Sicherheitsanforderung und Risiko, so dass die Anforderungen direkt in die Prozessmodellierung einfließen konnten. Zwei Ausnahmen bilden die Risiken *Weitergabe* und *Nicht-spezifiziertes Verhalten*. Nach Erkennung der potenti-

Kürzel	Risiko	Sicherheitsanforderung	Sicherheitsmechanismus
Vortäuschung	Vortäuschung einer falschen Identität	Authentizität	Digitale Signierung, Benutzeranmeldung
Manipulation	Fehlerhafte Übertragung oder bewusste Manipulation von Daten	Integrität	Digitale Signierung
Weitergabe	Weitergabe von vertraulichen (z.B. personenbezogenen) Daten	Anonymität, Vertraulichkeit	Verschlüsselung
Ausfall	Ausfall von Funktionen des Systems, bzw. des kompletten Systems	Verfügbarkeit	Erstellung von Backups
Unautorisierte Nutzung	Unautorisierte Nutzung von Funktionen des Systems	Autorisierung	Rechtevergabe
Verlust der Nachvollziehbarkeit	Verlust der Nachvollziehbarkeit von fehlerhaften, bzw. unerlaubten, Ereignissen	Nachweisbarkeit	Event-Logging
Nicht-spezifiziertes Verhalten	Nicht-spezifiziertes Verhalten der Funktionen der Software	Anonymität, Authentizität, Autorisierung, Integrität, Verfügbarkeit, Vertraulichkeit	Testen (inkl. BPM)

Abbildung 1.9.: Übergangsmatrix

ellen Risiken für Sicherheitsvorfälle, wurden die Sicherheitsmechanismen zur Vermeidung dieser ermittelt. Die Sicherheitsmechanismen wurden in die Übergangsmatrix integriert. Die betrachteten Mechanismen sind digitale Signierung, Verschlüsselung, Erstellung von Backups, Rechtevergabe, Event-Logging und Business Process Mining, deren Referenzierung im Projektplan zu finden ist [33].

1.5.3. Implementierung der Sicherheitsmaßnahmen

In diesem Abschnitt wird die Implementierung der Sicherheitsmechanismen beschrieben. Die Beschreibung ist nach den Sicherheitsinteressen untergegliedert. Es werden somit die Sicherheitsmechanismen zu je einem Sicherheitsinteresse genannt. Im Anschluss der Beschreibung der Maßnahmen der einzelnen Sicherheitsinteressen wird die Vermeidung des Risikos *Nicht-spezifiziertes Verhalten* beschrieben, welches eine Sonderstellung einnimmt, da das Risiko fast alle Sicherheitsinteressen betreffen kann.

1.5.3.1. Wahrung der Anonymität

Die Notwendigkeit der Anonymität resultiert aus der Verarbeitung personenbezogener Daten durch das Bankensystem. Im Fokus stehen hierbei zum Beispiel die personenbezogenen Daten der Kunden der Bank. Die Compliance der Bank fordert aufgrund des Bundesdatenschutzgesetzes [9], dass personenbezogene Daten nicht an Unberechtigte weitergegeben werden können. Es wird somit gefordert, dass die Kunden in Bezug zu unberechtigten Personen anonym bleiben. Die Anonymität muss sowohl für die Daten gelten, die von einem Client zum Server-System geleitet werden, als auch für die im System persistent gehaltenen Daten. So soll es nicht möglich sein, dass die Informationen der personenbezogenen Daten, die über das Internet geschickt werden und somit durch beliebige Rechner verarbeitet werden können, Unberechtigten zugänglich gemacht werden. Die Personen, auf die sich die Daten beziehen, sollen also Unberechtigten gegenüber anonym bleiben. Außerdem soll das unberechtigte Lesen der persistent gehaltenen Daten in der Datenbank des Systems vermieden werden. Auch hier gilt die Forderung nach der Anonymität der Personen auf die sich die Daten beziehen. So ist es beispielsweise notwendig, dass ein Kunde, der das Bankensystem nutzt, gegenüber weiteren Akteuren des Systems unbekannt ist, sofern dies nicht explizit erlaubt ist. Die Person soll also nicht aus den persistent gehaltenen Daten ermittelbar sein.

Die Anonymität stellt ein Sicherheitsinteresse dar, welches über das Sicherheitsinteresse der Vertraulichkeit von Daten abgedeckt wird. Aus diesem Grund entsprechen die für die Wahrung der Anonymität genutzten Sicherheitsmechanismen den Sicherheitsmechanismen für die Wahrung der Vertraulichkeit. Diese Sicherheitsmechanismen werden im Abschnitt 1.5.3.7 beschrieben, worin die Wahrung der Vertraulichkeit erläutert ist.

1.5.3.2. Wahrung der Authentizität

Die Authentizität wird für das Bankensystem gefordert, da zum einen die Benutzer des Systems korrekt identifiziert werden müssen. Das bedeutet, dass für das System eine korrekte Zuordnung des Benutzers zu einer Person notwendig ist. Zum anderen ist es für die Akteure des Systems notwendig, dass der Zugang zum System über eine bereitgestellte Verbindung authentisch ist. Hierbei muss erstens sichergestellt werden, dass eine von einem Benutzer aufgerufene Internetseite über dem Webbrowser dem Banken-

system zugeordnet werden kann. Zweitens muss eine zum Bankensystem hergestellte Verbindung durch ein weiteres externes System dem Bankensystem zugeordnet werden können. Die Verbindung zu dem externen System muss ebenfalls authentisch sein. Die korrekte Identifizierung der Benutzer wird für die Autorisierung benötigt. Ein Beispiel ist, dass ein Kunde nicht die gleichen Rechte eines anderen Kunden besitzt, da ihm dessen Daten im Allgemeinen nicht zugänglich gemacht werden sollen. Somit müssen sich die Kunden authentifizieren, um den Zugang zu ihren Daten zu erlangen. Die Authentifizierung der zum Bankensystem gehöhrenden Internetseiten ist hingegen notwendig, da die Möglichkeit besteht, dass vertrauliche Daten in die Eingabefenster der Internetseiten durch die Benutzer eingegeben werden können. Diese Daten dürfen nur von dem Bankensystem weiterverarbeitet werden. Da nun die Möglichkeit besteht, dass die Internetseiten des Bankensystems mit anderen Internetseiten verwechselt werden können, wird eine Authentifizierung dieser Seiten benötigt. Die Authentifizierung der Verbindung zum Bankensystem ist notwendig, da ein externes System ebenfalls personenbezogene Daten besitzen kann, die zum Bankensystem gesendet werden müssen. Dies gilt auch für die Authentifizierung der Verbindung zu den externen Systemen von dem Bankensystem aus.

Zur Wahrung der Authentizität des Systems gegenüber den Akteuren, wird die digitale Signierung einer SSL-Verbindung eingesetzt [56]. Diese dient ebenfalls zur Authentifizierung der externen Systeme. Die SSL-Verbindung wird serverseitig durch den Container *Glassfish* [42] bereitgestellt. Dazu muss die für *Glassfish* vorhandene XML-Datei *web.xml* bearbeitet werden [57]. In dieser Datei muss ein *transport-guarantee*-Tag mit dem Wert *CONFIDENTIAL* eingetragen werden, der in dem Tag *user-data-constraint* geschachtelt ist. Dieser ist wiederum in einem *Security Constraint*-Tag zu schachteln. Die Authentifizierung wird nun dadurch ermöglicht, dass Dateien über eine asymmetrische Verschlüsselung [30] derart übertragen werden, dass eine eindeutige Zuordnung einer verschlüsselten Datei zu dem Sender der Datei gegeben ist. Dazu wird für die Empfänger der Datei ein öffentlicher Schlüssel benötigt, der wiederum eine Signatur mit der Information über den Urheber des öffentlichen Schlüssels enthält. Als asymmetrische Verschlüsselung wurde das *RSA*-Verfahren eingesetzt [30]. Die Signatur wurde in dem *X.509*-Standard erstellt [34].

Die Authentifizierung der Benutzer des Systems erfolgt über eine Benutzeranmeldung. Hierbei werden den Benutzern des Systems über einen sicheren Kommunikationsweg ein Benutzername und ein Passwort zugeteilt. Durch den ausschließlichen Zugang zu dem System über ein Anmeldefenster können die Benutzer authentifiziert werden. Dazu werden die Eingaben eines Benutzers mit den in der Datenbank gespeicherten Daten verglichen. Zur Unterstützung der Anmeldung wurde in der *web.xml*-Datei von *Glassfish* ein *login-config*-Tag spezifiziert [63]. Der *login-config*-Tag ermöglicht unter anderem die Definition der Authentifizierungsmethode. Es wurde die Authentifizierungsmethode *FORM* ausgewählt, die festlegt, dass eine Authentifizierung über ein Formular, also ein Anmeldefenster, erfolgt. In Verbindung mit einem *HttpServletRequest*-Objekt wurde nun der *login-config*-Tag dazu verwendet, um die Benutzereingaben mit der Datenbank au-

tomatisch zu validieren [49]. Nach der Validierung erfolgt entweder eine Weiterleitung zu einer benutzerspezifischen Webseite des Systems oder zu einer Webseite mit dem Hinweis einer fehlgeschlagenen Authentifizierung.

Eine zusätzliche, kundenspezifische Authentifizierung ist im Bankenwesen bei Durchführung von Transaktionen durch Kunden über das Onlinebanking gegeben. In diesem speziellen Fall wird eine zusätzliche Authentifizierung mit Transaktionsnummern [61] gefordert, damit die Durchführung einer beabsichtigten Transaktion zusätzlich bestätigt wird. So soll die Durchführung von Transaktionen ohne Einwilligung des Kunden weniger wahrscheinlich sein. Im Bankensystem wurde das *iTAN*-Verfahren zur Authentifizierung der Transaktionen von Kunden eingesetzt [61]. Dazu wird jedem Kunden eine Liste von 50 Transaktionsnummern mit einer Länge von jeweils 5 Ziffern generiert. Die Generierung erfolgt mit Hilfe eines Objekts der Klasse *SecureRandom* des Frameworks *JCA* und dem Algorithmus *SHA-1* [64]. *SecureRandom* ist ein Pseudo-Zufallszahlengenerator, der Zufallszahlen kryptographisch sicher erzeugt. Diese Zufallszahlen sind für die kryptographische Nutzung geeignet [64]. Nach der Generierung der Zufallszahlen wurden zur Berechnung der Transaktionsnummern der Rest der Division der Zufallszahlen durch 100000 bestimmt. Somit konnten fünfstellige Zufallszahlen erzeugt werden, die kryptographischen Ansprüchen genügen. Zur Authentifizierung von Transaktionen wurde nun die Abfrage der Transaktionsnummern nach dem *iTAN*-Verfahren implementiert.

1.5.3.3. Wahrung der Autorisierung

Die Autorisierung ist notwendig, um den Benutzern des Bankensystems nach einer erfolgreichen Authentifizierung die benötigten Rechte zur Nutzung der Funktionen des Systems zu vergeben. Dazu wird in der Spezifikation zunächst eine Einschränkung der Möglichkeiten zur Nutzung der Funktionen definiert. Dies ist notwendig da mehrere Benutzer mit unterschiedlichen Berechtigungen Zugang zum Bankensystem haben. Ziel der Einschränkung der Rechte ist es also, dass die Benutzer keine Funktionen nutzen können zu denen sie nicht berechtigt sind. Es ist außerdem notwendig, dass in der Spezifikation Rollen definiert sind, die bestimmte Funktionen durchführen können. Eine Rolle definiert für einen Benutzer die Menge der zugelassenen Funktionen des Systems. Bei der Erstellung eines Nutzers wird diesem dann eine der spezifizierten Rollen zugewiesen. Wenn sich der Benutzer authentifiziert, ist es dem System dann möglich dem Benutzer die bei der Erstellung zugewiesene Rolle zu vergeben. Ein Beispiel für die Notwendigkeit der Autorisierung, also auch der Einschränkung der Rechte, ist, dass es zwei Benutzer gibt, die in der Bank unterschiedlichen Tätigkeiten nachgehen. So kann ein Benutzer ein Eigenhändler der Bank sein und ein anderer Benutzer kann der Systemadministrator der Bank sein. Es ist hierbei offensichtlich, dass dem Eigenhändler nicht die gleichen Rechte zur Verfügung gestellt werden sollen wie dem Systemadministrator. In diesem speziellen Fall dient die Einschränkung der Rechte des Eigenhändlers beispielsweise dazu, dass keine falschen, beabsichtigten oder nicht beabsichtigten Einstellungen des Systems gemacht werden.

Die Wahrung der Autorisierung wurde im Bankensystem durch eine rollenbasierte Zugriffskontrolle auf die Webseiten des Systems umgesetzt. Da die Webseiten die Benutzeroberfläche und damit den einzigen Zugang eines Benutzers zu den Funktionen des System darstellen, können die einzelnen Funktionen zum Zwecke der Einschränkung der Funktionen auf die Webseiten verteilt werden. Somit kann eine Einschränkung des Zugriffs auf die Webseiten gleichzeitig die Möglichkeiten zur Nutzung von Funktionen einschränken, da der Zugang zu den Webseiten die Voraussetzung zur Nutzung der Funktionen ist. Um nun die Zugriffskontrolle umzusetzen, wurden in der *web.xml*-Datei von *Glassfish* *security-role*-Tags definiert, mit denen Rollen mit eingeschränkter Zugriffskontrolle deklariert werden können [63]. Um nun den Zugriff einer Rolle auf bestimmte Webseiten zu beschränken, wurden *security-constraint*-Tags für jede deklarierte Rolle definiert [63]. In jedem dieser Tags wurden die für eine Rolle zugänglichen Webseiten aufgelistet. Die Auflistung der erlaubten Zugriffe ist gegenüber einer alternativen Auflistung der nicht erlaubten Zugriffe vorteilhaft, da für den Fall, dass die erlaubten Zugriffe nicht vollständig aufgelistet wurden, keine Sicherheitslücke entstehen kann. Durch eine Zugriffskontrolle auf die Webseiten des Bankensystems wurde implizit die Vergabe von Rechten für die Benutzer und somit die Wahrung der Autorisierung zur Nutzung der Funktionen im System umgesetzt.

Zur konkreten Vergabe der Nutzerrechte wurde ein Anlegen von Benutzern für das Bankensystem implementiert. Das Anlegen erfolgt dabei mit einer wiederholten Eingabe eines Passworts für die korrekte Authentifizierung. Ein Mangel dieser Funktionalität ist, dass der Abgleich der beiden Passwörter nicht immer korrekt funktioniert. So kann es passieren, dass die gleiche Eingabe des Passwortes nicht erkannt wird. Dieser Mangel konnte aus zeitlichen Gründen nicht behoben werden. Aufgrund eines weiteren Fehlverhaltens des implementierten Bankensystems existieren nicht spezifizierte Zugriffsbeschränkungen für die Rolle *Kunde*. Dieses Fehlverhalten konnte aufgrund von Zeitmangel nicht behoben werden. Ein zusätzlicher Mangel des Systems bezüglich der Rechtevergabe besteht darin, dass viele Seiten rollenübergreifend genutzt werden, obwohl nicht alle Rollen die gleichen Rechte bezüglich der durch die Seiten zugänglichen Funktionen besitzen. Dies führt dazu, dass ein Benutzer, der sich mit einer bestimmten Rolle anmelden kann, Kenntnis über das Vorhandensein von Funktionen gewinnt, die er nicht nutzen kann. Dieser Mangel ist entstanden, da die Erstellung von rollenspezifischen Seiten, in denen ausschließlich die für eine Rolle spezifizierten Funktionen beinhaltet sind, einen erhöhten Arbeitsaufwand erfordert. Dieser konnte wiederum aus Zeitgründen nicht aufgebracht werden.

1.5.3.4. Wahrung der Integrität

Die Wahrung der Integrität der Daten wird zum Schutz vor Manipulation der Daten gefordert. Dadurch sollen die Prozesse des Bankensystems nach der Spezifikation entsprechend umgesetzt werden, die in dem Pflichtenheft definiert [32] ist. Grundsätzlich

können zwei verschiedene Arten der Manipulation der systembezogenen Daten genannt werden. Zum einen können die Daten, die für die Verarbeitung an das System gesendet werden, manipuliert werden. Zum anderen können die Daten in dem System manipuliert werden. Ein Beispiel für die Notwendigkeit der Wahrung der Integrität beim Senden von Daten ist, dass die Durchführung einer Überweisung, die von einem Kunden mittels eines Client-Systems ausgelöst wurde, in der vom Kunden vorgesehenen Form durchgeführt werden soll. Es sollen dazu die vom Kunden eingegebenen Daten, wie das Quell-, das Zielkonto, der Verwendungszweck, sowie der zu überweisende Betrag unverändert zum Server-System, also zum Bankensystem, übertragen werden. Bei diesem Vorgang ist zu beachten, dass aufgrund der Übertragung der Daten über das Internet, die Daten durch beliebige Rechner zur Weiterleitung vom Client zum Server verarbeitet werden können. Somit besteht potentiell die Gefahr, dass die Daten an diesen Rechnern manipuliert werden. Zur Wahrung der Integrität sollen die Daten in der vorgesehen Form verarbeitet werden. Dies bedeutet, dass die bei der Eingabe spezifizierten Konten durch den eingegebenen Betrag aktualisiert werden und der Vorgang in der Datenbank gespeichert wird. Ein Beispiel zur Wahrung der Integrität bezüglich der Daten des Systems ist, dass der Kontostand eines Kunden, der in der Datenbank gespeichert ist, nicht auf unberechtigte Weise verändert werden soll.

Die Wahrung der Integrität gesendeter Daten wurde mittels der digitalen Signierung der SSL-Verbindung [56] umgesetzt. Die digitale Signierung kann die Manipulation von Daten bei der Übertragung über das Internet erkennbar machen. Die Wahrung der Integrität kann nun dadurch hergestellt werden, dass bei der Übertragung gefälschte Daten für die Weiterverarbeitung an einem Rechner nicht genutzt, sondern verworfen werden. Ein erneutes Senden der Daten wird daraufhin angefordert. Zur Erkennung der Integrität wird ein Hash-Wert der übertragenen Daten bestimmt. Hierbei gilt, dass zwei verschiedene Daten immer zwei verschiedene Hash-Werte besitzen. Der Sender einer Nachricht schickt die Daten mit dem berechneten Hash-Wert an den Empfänger. Der Empfänger kann den Hash-Wert aus der Nachricht berechnen und dann mit dem übersendeten Hash-Wert vergleichen. Zur Berechnung des Hash-Wertes muss ein Algorithmus verwendet werden, der eine Hash-Funktion realisiert. In der SSL-Verbindung des Bankensystems wird dazu der Algorithmus *SHA-256* [54] verwendet.

Zur Wahrung der Integrität bezüglich der Daten des Systems muss die Möglichkeit der Manipulation der Daten des Bankensystems durch einen SQL-Injection-Angriff [51] betrachtet werden. Bei einem SQL-Injection-Angriff werden SQL-Befehle über ein Client-System in die Benutzeroberfläche des Server-Systems eingegeben, die an die Kommunikationsschnittstelle des Server-Systems mit der Datenbank weitergegeben werden. Mit den SQL-Befehlen können so Datenbankeinträge in einer nicht vorgesehenen Form verändert werden. Die Verwaltung der Datenbank in dem Bankensystem wurde mit dem Java-Framework JDBC [43] umgesetzt. Um sich vor den SQL-Injection-Angriffen zu schützen, werden die SQL-Befehle ausschließlich mit einem *PrepareStatement*-Befehl [51] des JDBC-Frameworks durchgeführt. Bei diesem Befehl werden die von Benutzern eingegebenen Daten nicht direkt an die Datenbank weitergeleitet und dort als SQL-Befehl interpre-

tiert. Ein Prepare-Statement-Befehl hat stattdessen Platzhalter für die Benutzereingabe und die Interpretation des Befehls erfolgt mit den Platzhaltern. Die Ausführung des Befehls erfolgt im Nachhinein mit den Benutzereingaben. Dies verhindert, dass Benutzereingaben als SQL-Befehle interpretiert werden können [51].

Eine weitere Möglichkeit der Manipulation der Daten des Bankensystems ist es die Benutzeroberfläche des Systems durch *Cross Site Scripting*-Angriffe [41] zu verändern. Hierbei werden in den Webseiten *JavaScript*-Befehle [37] eingegeben und gespeichert, sodass dadurch die Syntax der Webseite verändert wird. Um sich auch vor dieser Art der Manipulation zu schützen, wurde das Framework *Java Server Faces* [29] zur Erstellung der Webseiten des Bankensystems verwendet. Mit Hilfe von *Java Server Faces* lässt sich die Anzeige von Inhalten in Webseiten mittels einem *Output*-Tag realisieren. Unter Nutzung dieses Tags werden *HTML*-Befehle bei der Anzeige ignoriert [29]. Die Anzeige von Inhalten, die von Benutzern eingegeben werden können, wurden ausschliesslich mit dem *Output*-Tag realisiert. Dadurch werden *Cross Site Scripting*-Angriffe verhindert.

1.5.3.5. Wahrung der Nachweisbarkeit

Die Nachweisbarkeit wird benötigt, um alle durchgeführten Aktionen des Systems und der Benutzer aufzeichnen und überprüfen zu können. Dies bedeutet, dass es für berechtigte Personen möglich gemacht werden muss, dass Aufzeichnungen der Aktionen des Systems und der Benutzer eingesehen werden können. So ist es beispielsweise notwendig, die Aktionen eines Benutzers einzusehen, wenn sich herausgestellt hat, dass das Verhalten des Benutzers nicht der vorgesehenen Form entsprochen hat. Ein Beispiel ist, dass ein Benutzer eine Straftat begangen haben kann. Der Zweck des Einsehens wäre in diesem Fall die Nachweisbarkeit der Straftat. Eine weitere Möglichkeit für die Notwendigkeit der Nachweisbarkeit ist nach einem unerwarteten und nicht-spezifizierten Verhalten des Systems, wie zum Beispiel einem Absturz, gegeben. In diesem Fall ist es für die Personen, die für die Wartung des Systems zuständig sind, wichtig zu erfahren aus welchem Grund das nicht-spezifizierte Verhalten aufgetreten ist. Die Einsicht auf die Dokumentation der Aktionen stellt die Möglichkeit bereit, die Aktionen nachzuvollziehen.

Die Wahrung der Nachweisbarkeit kann mittels dem Loggen von Aktionen realisiert werden. Dies bedeutet, dass alle Aktionen aufgezeichnet werden. Hierzu wurde eine Logging-Funktionalität implementiert, die das Framework *OpenXES* verwendet [62].

Die Einsicht der erstellten Log-Dateien in der Benutzeroberfläche konnte allerdings aus zeitlichen Gründen nicht der Spezifikation entsprechend umgesetzt werden. Beispielsweise konnte für die Rolle *Geschäftsführer* die Einsicht der Log-Daten nicht implementiert werden.

1.5.3.6. Wahrung der Verfügbarkeit

Verfügbarkeit bedeutet, dass das Bankensystem zu den spezifizierten Zeiten für die Benutzer genutzt werden kann. Die Nutzung bezieht sich auf die Funktionen des Systems, weshalb die Verfügbarkeit fordert, dass die Funktionen ausgeführt werden können und dass diese der Spezifikation entsprechend umgesetzt werden können. Die Funktionen müssen also nicht nur ausführbar sein, sondern müssen zudem korrekte Ergebnisse liefern. Für das Bankensystem ist die Verfügbarkeit ein sehr wichtiges Interesse, da die Geschäfte einer Bank häufig über das Internet getätigt werden. Fallen die Funktionen des Systems aus, so fallen auch potentielle Einnahmen aus Geschäften für die Bank weg. Außerdem kann durch den Ausfall der Funktionen die Reputation der Bank geschädigt werden, weshalb auch hier mit negativen finanziellen Folgen zu rechnen ist.

Die Wahrung der Verfügbarkeit des Systems wurde in dem Bankensystem mit Hilfe der Erstellung von Backups realisiert. Die Erstellung von Backups ermöglicht die Aufnahme des Zustandes des Bankensystems zu einem bestimmten Zeitpunkt. Somit kann der Ausfall des Systems, beziehungsweise einzelner Funktionen des Systems, durch das Einspielen des erstellten Backups rückgängig gemacht werden. Dies verhindert, dass das System für einen langen Zeitraum nicht zur Verfügung steht. Das Einspielen führt dazu, dass die Funktionen, wie sie zu dem Zeitpunkt der Erstellung verfügbar waren, wiederhergestellt werden. Dazu wurde die Möglichkeit geschaffen, dass der Systemadministrator der Bank die Backups manuell erstellen kann. Außerdem wurde eine automatische Erstellung der Backups implementiert. Das automatische Erstellen erfolgt einmal pro Stunde. Die Daten werden dann für insgesamt 10 Tage gespeichert. Das Erstellen der Backups erfolgt mit Hilfe der Ausführung eines *Shell-Scripts* [58] auf dem Betriebssystem der Cloud.

1.5.3.7. Wahrung der Vertraulichkeit

Vertraulichkeit bedeutet, dass Informationen nur einem begrenzten Kreis von Personen zugänglich gemacht werden. Die Wahrung der Vertraulichkeit ist notwendig, da in dem Bankensystem personenbezogene Daten verarbeitet werden. Ein Beispiel für die Notwendigkeit der Vertraulichkeit ist, dass die Daten eines Kunden, wie die Kontonummer oder der Kontostand eines Kontos des Kunden, nur dem Kunden und seinem Kundenberater zur Verfügung gestellt werden. Es soll somit verhindert werden, dass nicht berechtigte Personen, wie ein anderer Kunde, Zugang zu den Daten des Kunden haben. Da die Daten sowohl im Bankensystem gespeichert werden, als auch über das Internet übertragen werden können, ist es notwendig den Zugang zu den persistent gehaltenen Daten und die Übertragung der Daten zu kontrollieren.

Die Gewährleistung der Vertraulichkeit während der Übertragung der Daten erfolgt mit Hilfe einer Verschlüsselung der Daten in der bereits im Abschnitt 1.5.3.2 beschriebenen SSL-Verbindung [56]. Hierbei wird die symmetrische und asymmetrische Verschlüsselung in der hybriden Verschlüsselung kombiniert. Zur asymmetrischen Verschlüsselung wurde

der *RSA*-Algorithmus verwendet [30]. Die Schlüssellänge wurde auf 2048 Bits festgelegt. Die Verschlüsselung kann die Vertraulichkeit während der Übertragung der Daten im Internet gewährleisten. Eine verschlüsselte Datei wirkt wie die zufällige Anordnung von Zeichen ohne Möglichkeit auf eine Interpretation der Zeichen. Wenn der Zugang zum Entschlüsselungsschlüssel nicht gegeben ist, können somit die Informationen der übertragenen Dateien nicht ermittelt werden.

Zur Wahrung der Vertraulichkeit für die persistent gehaltenen Daten, wurde die in Abschnitt 1.5.3.3 beschriebene rollenbezogene Rechtevergabe eingesetzt. Somit lässt sich die Einsicht der Daten derart kontrollieren, dass nur berechtigte Personen Zugang zu diesen Daten haben. Eine weitere Maßnahme zur Wahrung der Vertraulichkeit betrifft den im Abschnitt 1.5.3.4 beschriebenen *SQL-Injection*-Angriff. Da ein *SQL-Injection*-Angriff nicht nur die Daten des Systems manipulieren, sondern auch den unberechtigten Zugang zu den Daten ermöglichen kann, ist die im Abschnitt 1.5.3.4 beschriebene Maßnahme zum Schutz vor *SQL-Injection*-Angriffen auch für die Wahrung der Vertraulichkeit relevant. Die nicht berechtigte Einsicht der Daten wird deshalb möglich, weil bei einem Angriff nicht nur SQL-Befehle zum Schreiben, sondern auch zum Lesen von Daten eingegeben werden können.

1.5.3.8. Vermeidung von nicht-spezfiziertem Verhalten

Nicht-spezifiziertes Verhalten der Funktionen des Bankensystems kann auftreten, wenn ein Fehler bei der Berechnung der Daten auftritt, oder ein Angreifer versucht, das System anzugreifen. In der Planungsphase wurde das Testen des Bankensystems zur Vermeidung von nicht-spezifiziertem Verhalten, sowie die Überwachung der Prozessabläufe mittels *Business Process Mining* (kurz: *BPM*) [62] zur Erkennung von nicht-spezifiertem Verhalten geplant.

Nicht-spezifiziertes Verhalten kann vermieden werden, wenn das System ausreichend getestet wird. Zweck des Testens ist die Erkennung von Fehlverhalten, die aufgrund von fehlerhafter Implementierung entstehen. Somit werden Fehler bereits in der Implementierungsphase erkannt und behoben. Das Testen der Software wurde mit dem Framework *J-Unit* umgesetzt [45].

Mit Hilfe von *BPM* kann das Verhalten des Systems, das durch Log-Dateien aufgezeichnet wird, mit einem spezifizierten Verhalten verglichen werden. Das Verhalten entspricht dem Prozessablauf des Systems. Wird eine Abweichung des aktuell durchgeführten Prozessablaufs zu dem spezifizierten Prozessablauf festgestellt, können Log-Dateien mit Einträgen über die aufgetretenen Anomalien erstellt werden. Das *Business Process Mining* wurde mit Hilfe der Software *ProM* [62] umgesetzt. In *ProM* können Petri-Netze [33] aus Log-Dateien durch Importieren dieser in *ProM* erzeugt werden. Außerdem können die erstellten Petri-Netze mit weiteren Petri-Netzen zur Erkennung der Anomalien verglichen werden. Zur Erstellung der Log-Daten wurde, wie bereits im Abschnitt über die

Nachweisbarkeit erwähnt, die *OpenXES Library* verwendet. Die Log-Daten werden bei Aufruf jeder Methode des Systems erstellt. Anschließend werden sie in *ProM* importiert und das daraus erstellte Petri-Netz wird mit dem gespeicherten Petri-Netz verglichen, das im Voraus erstellt wurde und dem spezifizierten Verhalten entspricht. Somit konnte die Erkennung von nicht-spezifiziertem Verhalten umgesetzt werden. Die Erkennung kann im weiteren zur Vermeidung darauffolgender, nicht-spezifizierter Prozessabläufe genutzt werden.

1.5.4. Verwaltung der RSA-Schlüssel

Im Folgenden wird die Verwaltung der *RSA*-Schlüssel beschrieben, die für die Verwaltung der in Abschnitt 1.5.3 beschriebenen Sicherheitsmaßnahmen digitale Signierung und Verschlüsselung notwendig ist. Da der *RSA*-Algorithmus für beide Maßnahmen verwendet wird, werden in beiden Maßnahmen *RSA*-Schlüssel eingesetzt.

Die SSL-Verschlüsselung setzt ein Schlüsselpaar für die asymmetrische *RSA*-Verschlüsselung ein. Nach einer gewissen Zeit kann das Schlüsselpaar als nicht mehr sicher angesehen werden, da beispielsweise Schlüssel mit einer größeren Anzahl an Bits zur sicheren Kommunikation benötigt werden. Aufgrund dessen wurde die Verwaltung der *RSA*-Schlüssel implementiert. Die Verwaltung ermöglicht das Einsehen der vorhandenen Schlüssel, das Generieren von neuen Schlüsseln, das Löschen vorhandener Schlüssel, sowie das Ersetzen des aktuell genutzten Schlüssels für die SSL-Verbindung. Zur Ersetzung des genutzten Schlüssels muss der *Glassfish*-Server neu gestartet werden. Der *Glassfish*-Server nutzt zur Speicherung der Schlüssel die Datei *keystore.jks* [48]. Ein Objekt der Klasse *Keystore* des Frameworks *JCA* [64] kann nun dazu genutzt werden, um zur Laufzeit Zugang zu der *keystore.jks*-Datei zu erlangen und die zuvor beschriebenen Funktionen durchzuführen. Durch die Nutzung der Klasse *Keystore* konnte somit die Verwaltung der Schlüssel des Bankensystems implementiert werden.

Damit eine Verbindung von Clients zum Bankensystem aufgebaut werden kann, muss das bei der Authentifizierung ausgetauschte Zertifikat für beide Seiten vertrauenswürdig sein. Ein Client kann hierbei ein externes System, wie die Interbank, sein. Unter der Annahme, dass die Zertifikate über einen sicheren Kommunikationsweg ausgetauscht wurden, muss es möglich sein, dass das Zertifikat der Gegenstelle für den *Glassfish*-Server bekannt gemacht wird. Durch das Bekanntmachen wird ein Zertifikat als vertrauenswürdig eingestuft. Die vertrauenswürdigen Zertifikate werden für den *Glassfish*-Server in der *truststore.jks*-Datei [48] gespeichert. Sobald also ein Zertifikat in dieser Datei gespeichert ist, wird die Verbindung zum Client ermöglicht. Zur Verwaltung der Einträge der *truststore.jks*-Datei wurde wieder ein Objekt der Klasse *Keystore* genutzt. Es wurden mit Hilfe dieser Klasse die Funktionen zum Auslesen der vertrauenswürdigen Zertifikate, zum Hinzufügen neuer Zertifikate und zum Löschen vorhandener Zertifikate implementiert.

1.6. GUI Testen durch Selenium

Autor: CW

1.6.1. Einführung

Unser Projekt basiert auf dem JSF-Framework[55], welches ein Bestandteil von JavaEE ist. Dadurch, dass es bei dieser Technik hauptsächlich um die Erstellung von Webpräsenzen geht, muss die Oberfläche dieser entsprechend getestet werden. Es gibt neben dem Selenium-Framework[7] noch einige weitere Frameworks zum Testen solchen Oberflächen. Auf diese musste jedoch aufgrund der Inkompatibilität mit ***Glassfish*** verzichtet werden. Selenium[55][7] ist ein Testwerkzeug für Webanwendung und unterstützt die Programmier-/Skriptsprachen C#, Python, Javascript, JSP, JSF. Mit diesem Framework ist es möglich Aktionen im Brwoser aufzuzeichnen und später zum Testen wieder abzuspielen (Dies wird auch *Benutzeroberflächetest* genannt).

1.6.2. Selenium IDE

Selenium IDE (siehe Abbildung 1.10) ist ein Plugin für den Webbrowser. Mit diesem können Bewegungen und Klicks auf der GUI-Anwendung nachverfolgt werden. Die Oberfläche ist intuitiv und sieht aus wie ein Video-Rekorder. In der Base URL Zeile wird die URL der zu testenden Seite eingegeben. Danach werden die nachfolgenden Benutzeraktionen im Browser automatisiert aufgezeichnet. Es ist möglich , die aufgezeichneten Aktionen durch dieses Tool abzuspielen. Als Alternative dazu, kann Selenium auch mit JUnit[3] zusammenarbeiten. Mit Hilfe der Selenium IDE kann die Suche nach möglichen Programmfehler erfolgen.

1.6.3. Selenium RC

Mit Hilfe von Selenium RC(Remote Control Server) kann auch ein Test auf einer entfernten Maschine erfolgen. Die Architektur der Anwendung Selenium RCs ist in Abb. 1.11 dargestellt. Um die Oberfläche zu testen muss ein bestimmter Selenium-Core in den Browser eingebunden werden.

1.6.4. Problem mit SSL-Zertifikate

Bei den Tests mit Selenium kamen unerwartete NULL-POINTER-Exceptionen vor. Das wesentliche Problem lag darin, dass unser Server eine verschlüsselte Verbindung aufbaut. Dadurch, dass das verwendete Zertifikat in unserer Anwendung nicht zertifiziert

[3] siehe den Abschnitt 1.6.5

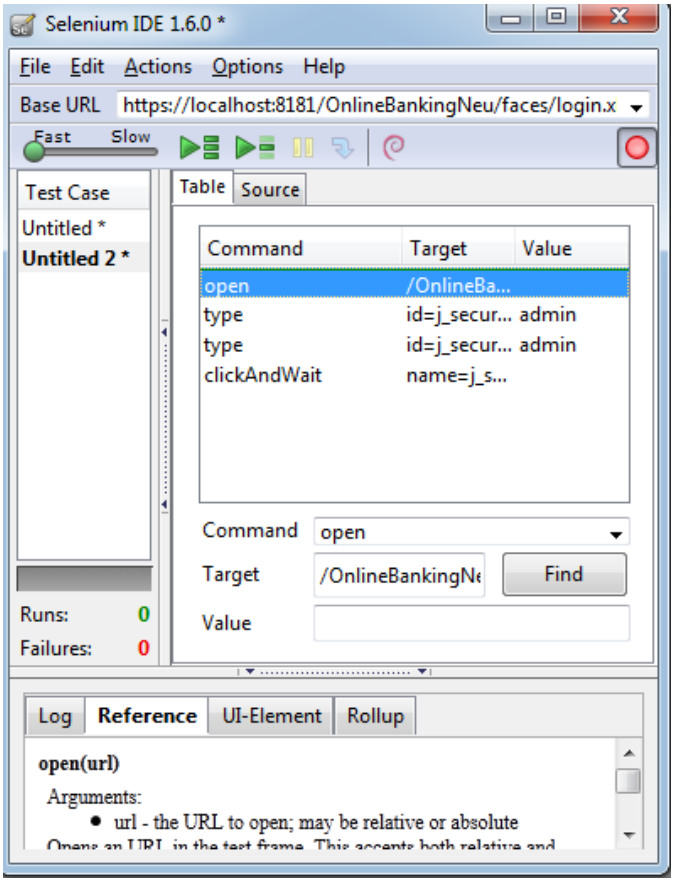


Abbildung 1.10.: Selenium RC

war, zeigte der Browser eine entsprechende Fehlermeldung. Diese führte letztlich dazu, dass der Test mit Selenium abgebrochen wurde[8]. Um dieses Problem zu lösen, musste innerhalb von Firefox das verwendete Browserprofil geändert oder ein Neues erstellt werden. Im Folgenden wurde immer die Kombination aus Selenium und Firefox verwendet.

1.6.4.1. Einstellungsvorgang

Durch den Befehl “firefox.exe -ProfileManager“ ist ein neues Firefoxprofil in einem bestimmten Verzeichnis anzulegen, so dass beim nächste Mal der Selenium-Test mit dem neuen Profil gestartet wird. Dafür ist es nötig mit diesen Befehlen aus der Selenium API das neue Firefox-Profil zu konfigurieren[8]:

“java -jar selenium-server-2.19.0.jar -trustAllSSLCertificates -firefoxProfileTemplate [Profilspeicherpfad]“

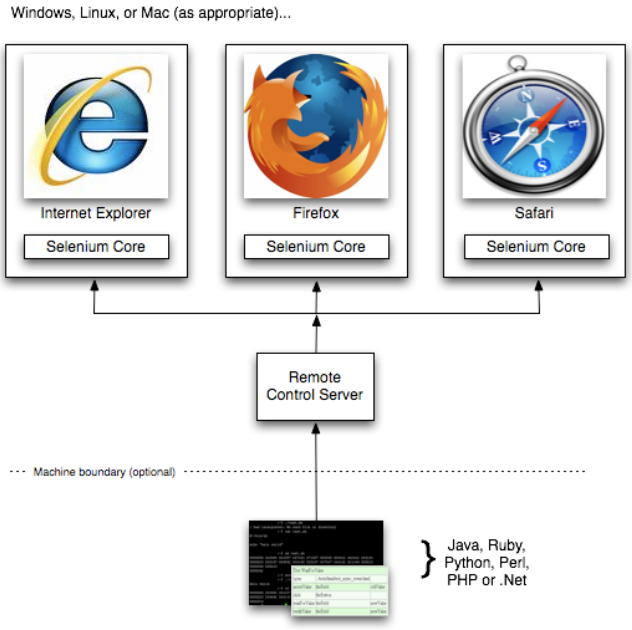


Abbildung 1.11.: Selenium RC[7]

Das Argument "-trustAllSSLCertificates" spielt dabei eine große Rolle. Mit diesem Attribut wird auch der Zugang zu Webseiten ohne gültiges Zertifikat erlaubt.

1.6.5. Selenium Testen in Form von JUnit

Der Test der GUI der Webanwendung kann auch ohne die Selenium IDE durchgeführt werden. Dazu muss die entsprechende Library (Selenium-JAVA-API) im Projekt aufgenommen werden. Anschließend können die JUnit-Codes auch manuell erstellt werden. Dieser Code besteht aus drei Teilen, die mit der Annotation @ gekennzeichnet werden:

@Before Zuerst werden alle für den Test nötigen Vorbereitungen getroffen werden. Darunter fällt z.B. die Instanziierung des Browser, sowie die Eingabe der Ziel-URL.

@Test Hierin befindet sich der eigentliche Test. In diesem Teil werden die aufzurufenden GUI-Seiten sowie Klicks programmiert.

@After Im After-Teil findet die Verifiktation des Tests statt. Darunter fällt z.B. die Prüfung, ob die korrekte Seite aufgerufen wurde, oder ob Layout-Fehler während des Test aufgetreten sind (siehe Abbildung 1.13). Tritt ein solcher Fehler auf, wird der laufenden Test unterbrochen und die entsprechende Fehlermeldung wird in der Konsole ausgegeben. Tritt kein Fehler auf, läuft der Test bis zum Ende durch (siehe Abbildung 1.12).

Abbildung 1.12.: Selenium Testen mit JUnit

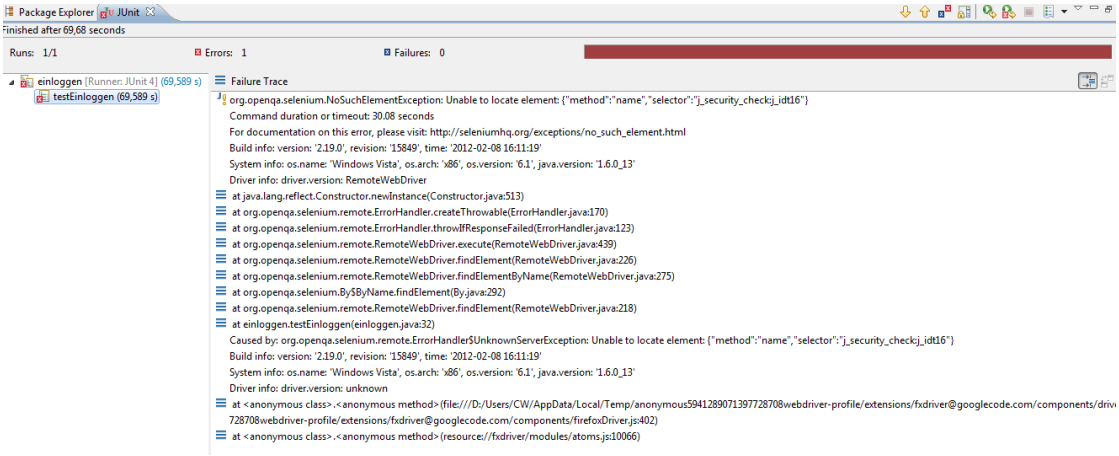


Abbildung 1.13.: fehlschlägtes Testen

1.7. Log-System mit Log4J

Autor: YD

Log4j ist ein Loggingtool in Java, dass es auf einfache und komfortable Art ermöglicht, Meldungen auf verschiedenste Art auszugeben.[6]
Log4j wird im Bankensystem dazu genutzt, um Log-Dateien im Hintergrund des Systems zu generieren und in der Benutzeroberfläche anzuzeigen. Durch die Anzeige der Log-Dateien können die Mitarbeiter den Zustand des Systems einsehen. Es ist unter anderem möglich eine Ausgabe von Log-Dateien mit Informationen über den abgearbeiteten Quellcode zu generieren. Dies erleichtert unter anderem die Überprüfung und Wartung des Systems.

1.7.1. Komponenten

In *Log4j* gibt es die drei Hauptkomponenten *Logger*, *Layout* und *Appender*. Um die Komponenten benutzen zu können, wird die Library *org.apache.log4j* benötigt. In den folgenden Abschnitten werden die Komponenten beschrieben. Die Log-Dateien werden im Folgenden als Meldungen bezeichnet.

1.7.1.1. Logger

Logger ermöglicht die Ausgabe der erstellten Log-Dateien. Die Meldungen können nach 5 Stufen priorisiert werden. Die einzelnen Stufen sind *DEBUG*, *INFO*, *WARN*, *ERROR* und *FATAL*. Jede Meldung muss einem der Stufen zugeordnet werden. Durch die Priorisierung kann der Benutzer erkennen, wie schwerwiegend eine generierte Meldung ist. Falls beispielsweise eine Meldung der Stufe *FATAL* zugeordnet wird, handelt es sich sehr wahrscheinlich um eine wichtige Meldung [6].

Beipiel-Code:

```
Logger l = Logger.getLogger("wecker");
l.setLevel(Level.ERROR);
l.debug("Jetzt wird geweckt.");
l.info("Guten Morgen.");
l.warn("Du solltest langsam aufstehen!");
l.error("Es ist ein unerwarteter Fehler aufgetreten!");
l.fatal("Das war's. Du kommst zu spaet zur Arbeit!");
```

In dem oben vorgestellten Beispiel-Code wird eine Meldung mit den 5 genannten Stufen zur Priorisierung definiert. Die Meldung in der Stufe *DEBUG* bezeichnet die niedrigste Priorität, während die Meldung mit *FATAL* höchste Priorität hat.

1.7.1.2. Appender

Appender ermöglicht die Ausgabe der Meldungen auf verschiedene Arten. Beispielsweise kann definiert werden, dass die Meldungen in der Konsole der Entwicklungsumgebung oder in Log-Daten ausgegeben werden. An einen Logger können mehrere Appender angehängt werden, um verschiedene Situationen zu unterscheiden. Ein Beispiel ist, dass drei *Appender* mit den folgenden Funktionen verfasst werden. Der erste *Appender* ist für die Ausgabe einer Meldung mit niedriger Priorität in der Benutzeroberfläche zuständig. Der zweite *Appender* ermöglicht das Schreiben von Meldungen in Log-Daten, die Informatioenn über die Datenbank beinhalten und mit höherer Priorität eingestuft wurden. Der dritte *Appender* ermöglicht hingegen die Ausgabe von Meldungen in der Konsole der Entwicklungsumgebung, die ebenfalls Informationen über die Datenbank enthalten.

Beispiel-Code:

```
log4j.rootLogger=DEBUG,myapp,A1

log4j.appender.myapp=org.apache.log4j.RollingFileAppender
log4j.appender.myapp.file=/home/LogDatei.log

log4j.appender.myapp.maxFileSize=10MB
log4j.appender.myapp.maxBackupIndex=10
log4j.appender.myapp.layout.ConversionPattern=\%5p | \%d | \%F:\%L | \%m\%n

log4j.appender.A1=org.apache.log4j.ConsoleAppender
log4j.appender.A1.layout=org.apache.log4j.PatternLayout
log4j.appender.A1.layout.ConversionPattern=\%5p | \%d | \%F:\%L | \%m\%n
```

In dem zuvor dargelegten Beispiel-Code können die Meldungen in einer Log-Datei geschrieben werden. Dazu wird eine Log-Datei mit Namen *Logfile.log* generiert. Eine detaillierte Beschreibung von Log-Dateien ist in Abschnitt 1.7.3.1 zu finden. Es wird festgelegt, dass die Meldungen auf der Konsole der Entwicklungsumgebung angezeigt werden. Weiterhin wird festgelegt, dass die Log-Datei maximal 10 MB groß sein kann. Außerdem wird die Anzahl der Meldungen auf höchstens 10 Stück begrenzt. Zusätzlich wird die Anzeige von Informationen über den Quellcode definiert. Die Informationen beziehen sich auf den Quellcode-Abschnitt, der die Erstellung der Meldung auslöst. In dem Beispiel werden dazu der Methodenname, das Objekt, sowie die Zeit in der die betroffene Methode aufgerufen wurde, angezeigt und in die Log-Datei geschrieben.

1.7.1.3. Layouts

Die Komponente *Layout* bestimmt die Form der zu erstellenden Meldungen. Dazu wird vor Anzeige der Meldungen durch den *Appender* die Form dieser durch eine *Layout*-Komponente zugeordnet. Die Zuordnung wird in Konfigurationsdateien definiert. Beispiele:

% -5p linksbündig mit fünf Zeichen die Priorität der Meldung ausgegeben.

[% t] Ausgeben von welchem Thread aufgerufen ist.

(% F:% L) Name und Zeilenummer der Methode.

1.7.1.4. log4j.properties

In jedem Projekt, in dem *Log4j* verwenden werden soll, muss zuerst eine *properties*-Datei erstellt werden. Diese ist für die Einstellungen von *Log4j* notwendig. Die *properties*-Datei definiert unter anderem die Komponenten, die von *Log4j* verwendet werden sollen, und wie die Meldungen anzuzeigen sind (1.7.1.2, 1.7.1.3).

1.7.2. Besondere Konfiguration mit JSF und Spring unter Glassfish

Ein wichtiger Punkt ist die Kombination von *Log4j* mit *JSF* [29] und *Spring* unter *Glassfish* [42]. Da *Log4j* wenig in der Praxis mit *JSF* und Spring unter *Glassfish* verwendet wird, existiert lediglich wenig Literatur zu diesem Thema. Zur Erkennung von *Log4j* in einem Projekt mit *JSF* müssen einige Schlüsselwörter in der *properties*-Datei angegeben werden. Außerdem müssen in dieser Datei die Pfade für die Log-Dateien angegeben werden.

Beispile:

```
log4j.logger.com.acme=DEBUG
log4j.logger.org.springframework=DEBUG
log4j.logger.javax.faces=DEBUG
log4j.logger.org.apache.myfaces=DEBUG
log4j.logger.com.sun.faces=DEBUG
```

1.7.3. LogDaten

Die Meldungen, die von *Log4j* für das System erstellt werden, können sowohl in der Konsole der Entwicklungsumgebung angezeigt werden als auch in einer erstellten Log-Datei geschrieben werden. Die Einstellungen diesbezüglich werden ebenfalls in der *properties*-Datei 1.7.1.4 definiert.

1.7.3.1. LogDatei.log

Die Datei *LogDatei.log* wird standardmäßig unter Linux-Systemen im Ordner */home/LogDatei.log* und unter Windows-System unter *C:\home\LogDatei.log* angelegt. Im Folgenden werden beispielhaft zwei Meldungen angezeigt. Eine Meldung zeigt in diesem Beispiel die Stufe der Priorisierung, die Zeit der Erstellung der Meldung, die Klasse des Quellcodes, die die Erstellung der Meldung angestoßen hat und die Meldung selbst. Die Meldung hat hierbei ein bestimmtes Protokoll, welches zuvor festgelegt wurde.

```
DEBUG | 2012-02-22 14:27:28,637 | NavigationRollen.java:62
| Benutzer: testperson hat sich eingeloggt
DEBUG | 2012-03-13 13:48:14,105 | NavigationRollen.java:57
| Benutzer: admin hat sich eingeloggt
```

1.7.3.2. Log-Datei in Konsole

Ein Beispiel für Meldungen, die in der Konsole der Entwicklungsumgebung angezeigt werden, ist in Abbildung 1.14 dargestellt.

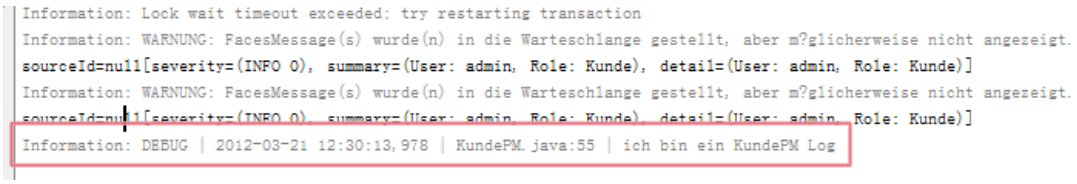


Abbildung 1.14.: Anzeige von Meldungen in der Konsole der Entwicklungsumgebung

2. Compliance

Autor: TS

2.1. Gesetze

Um die korrekte Abbildung und vollständige Erhaltung der Compliance im Projekt zu gewährleisten, wurde inspiriert durch [28] und [46] ein Verfahren entwickelt und angewedet. Zuerst wird im Unterabschnitt 2.1.1 das ursprünglich geplante Verfahren vorgestellt, danach wird in den Unterabschnitten 2.1.3 und 2.1.4 das Vorgehen nach diesem Verfahren erläutert und Abweichungen aufgezeigt.

2.1.1. Behandlung der Gesetze nach dem Projektplan

Das urprüngliche Vorgehen war folgendermaßen geplant:

1. Extraktion der wichtigsten Artikel mit Bezug zum Online-Banking-System aus diesen Texten.
2. Strukturierung der Gesetze nach Rechten, Pflichten und Verboten, und Bestimmung des handelnden Akteurs, der Aktion, des von der Aktion betroffene Objekt und zusätzlicher passiver Personen für jeden Artikel.
3. Modellierung der Rechte und Pflichten mit Secure Tropos [33].
4. Ermitteln der Anforderungen aus den Gesetzen, Extraktion der Rollen für die jeweilige Anforderung und Modellierung der Aktivitäten im System, die sich für jede Rolle ergeben, durch Anwendungsfalldiagramme in Secure Tropos.
5. Ergänzung der Anwendungsfalldiagramme durch zusätzliche Metainformationen in Form von Tabellen (eine je Anwendungsfall).
6. Zusammenführung der Aktivitäten mit den funktionalen Aktivitäten des Systems.
7. Um die geltenden Gesetze im System zurückverfolgen zu können, wurde eine Notation der Gesetzesanforderungen für jeden modellierten Prozess festgelegt. Dann sollte die Notation der Prozesse in dem Architekturentwurf die Zurückverfolgbarkeit herstellen.

2.1.2. Betrachtete Gesetze

In Anbetracht der sehr begrenzten Zeit wurden *BDSG* [9] und *MaRisk* [11] ausführlich analysiert und aus allen übrigen Gesetzen die wichtigsten Artikel berücksichtigt.

Die übrigen Gesetze sind:

- KonTraG [2]
- AktG [10]
- KWG [3]
- WpHG [4]
- HGB [15]
- AO [12]
- LDSG [5]
- TMG [1]

2.1.3. Analyse und Dokumentation der Gesetze

Während der Anforderungsanalyse wurden die zuvor aufgelisteten Gesetzestexte untersucht. Die relevanten Paragrafen wurden in Form einer Tabelle aufgelistet. Die Notationselemente [P], [V] und [R] wurden den Paragrafen hinzugefügt, um die Strukturierung in Pflichten, Verbote und Rechte umzusetzen. Außerdem wurden Paragrafen, die Begriffe definieren, in die Auflistung mit aufgenommen (Notationselement [D]). Die Ergänzung der Definitionen wurde festgelegt, da eine Beschreibung wichtiger Begriffe als sinnvoll angesehen wurde. Die im Projektplan definierten Eigenschaften, die für jeden Paragrafen angegeben werden sollten, wurden als Spaltenelemente in die Tabelle eingefügt. Die Eigenschaften unterscheiden sich hierbei zum Teil von den Eigenschaften, die im Projektplan definiert wurden. Die Eigenschaften sind:

- Gesetz (entspricht dem Kürzel des jeweiligen Paragrafen)
- Was (entspricht dem betroffenen Objekt)
- Wer (entspricht dem handelnden Akteur)
- Anforderung (entspricht den zusätzlichen Anforderungen an das System)
- Gesetzestext (Kurze Zusammenfassung des Paragrafen)

Ebenfalls abweichend vom Projektplan wurde auf die Notation mit SecureTropos verzichtet, da keine brauchbare Entwicklungsumgebung gefunden werden konnte.

Parallel zur Erstellung der Tabelle wurde das Pflichtenheft erstellt. Dort sind die Anwendungsfalldiagramme [17] für die Rollen des Systems modelliert. Für jeden Anwendungsfall wurde eine zusätzliche Tabelle erstellt, die unter anderem die Zeile “Gesetze“

beinhaltet. In dieser Zeile sind die relevanten Gesetze und Paragrafen für den jeweils betrachteten Anwendungsfall aufgelistet. Somit konnte eine Referenzierung der Gesetze und Paragrafen in die tabellarischen Anwendungsfallbeschreibungen integriert werden.

Zu der tabellarischen Beschreibung wurde im Pflichtenheft die Prozessmodellierung ergänzt, sodass ein Prozess pro Anwendungsfall vorhanden ist. Mit Hilfe der Referenzierung auf die relevanten Gesetze in der Anwendungsfallbeschreibung ist also zusätzlich eine Referenzierung in der Prozessmodellierung vorhanden.

2.1.4. Anforderungen durch Gesetze

Durch einige Gesetze ergaben sich zusätzliche Anforderungen an das System.

So fordert das *HGB* bestimmte Aufbewahrungsfristen für Unterlagen, die bestimmte Vorgänge oder deren Zustandekommen dokumentieren.

Das *BDSG* fordert, dass personenbezogene Daten wie Name, Adresse, Geburtsdatum und Bankverbindungen nur erhoben werden dürfen, wenn ein Grund dafür vorliegt und auch nur solange gespeichert werden dürfen, wie dieser Grund besteht.
Durch das *TMG* wurden insbesondere Anforderungen an die Sicherheit der Datenübertragung und die Lagerung von Daten definiert.

Die *MaRisk* fordern das erstellen einer Security Policy, sowie im Zusammenspiel mit dem *KWG* einen Zugang für BaFin und BZSt.

WpHG und *AO* definieren bestimmte Meldepflichten für Abwickelungen von Geschäften und Gesetzesverstöße, wobei das eher *WpHG* auf den Wertpapiere im Sinne von Aktien und die *AO* eher auf (insbesondere steuerlich relevante) Finanztransaktionen ausgelegt sind. Aus diesen Texten resultiert u.a. die Prüfung auf Verdächtigkeit bei Überweisungen und Wertpapiertransaktionen.

AktG und *KonTraG* richten sich auf die Struktur von Aktiengesellschaften aus und hatten somit keine Auswirkungen auf unser System.

2.2. Security Policy

Der Projektplan sah vor, dass eine *Security Policy* für das Online-Banking-System erstellt werden sollte, um den richtigen Umgang mit dem System zu fördern. Die *Security Policy* sollte nach dem BSI-Standard [13] entworfen werden.

Dazu war vorgesehen sie in drei Schichten zu entwerfen:

1. Die erste Schicht (BSI 100-1) sollte Management-Prinzipien, Ressourcen, Mitarbeiter und den Sicherheitsprozess beinhalten.
2. In der zweiten Schicht (BSI 100-2) sollte die Vorgehensweise für den Betrieb der Software beschrieben werden.

3. Die dritte Schicht (BSI 100-3) sollte aus den Handbüchern für die vorgesehenen Nutzer des Systems bestehen.

In den folgenden Unterabschnitten werden die Ergebnisse zusammengefasst. Die Dokumente der *Security Policy* sind diesem Dokument im Anhang beigefügt.

2.2.1. IT-Sicherheitsleitlinie (Security Policy Schicht 1, BSI 100-1)

Die *IT-Sicherheitsleitlinie* (Anhang C.1) beschreibt, wie im Allgemeinen sichergestellt werden soll, dass im Betrieb alle nötigen Gesetze und Standards eingehalten werden. Sie behandelt unter anderem die Themen:

1. Allgemeine Ziele
2. Geltungsbereich
3. Verantwortlichkeiten und Zuständigkeiten
4. Sicherheitsmanagement
5. Sicherheitsmaßnahmen
6. Richtlinien für Mindeststandards

In dem Abschnitt *Allgemeine Ziele* werden die grundlegenden Ziele der Sicherheitsleitlinie beschrieben, während im Abschnitt *Geltungsbereich* die Gültigkeit der Leitlinie für alle Mitarbeiter der Bank definiert wird. Im darauffolgenden Abschnitt werden die verantwortlichen Rollen Datenschutzbeauftragter, IT-Sicherheitsbeauftragter und Vertreter der Rechtsabteilung genannt, um die Zuständigkeiten für den Datenschutz, die Informationssicherheit und die Leitlinie zu beschreiben. Für das *Sicherheitsmanagement* wurden unter anderem die Einrichtung einer Sicherheitsorganisation und die Nennung eines Sicherheitsbeauftragten genannt. Die *Sicherheitsmaßnahmen* bestimmen unter anderem Verantwortlichkeiten für alle Verfahren, Informationen, IT-Anwendungen und IT-Systeme. Die Richtlinien gelten für die Themen Internet- und E-Mail-Nutzung, IT-Nutzung, Outsourcing, Archivierung, Datensicherung und Notfallvorsorge.

2.2.2. BSI-Grundschutz (Security Policy Schicht 2, BSI 100-2)

Die zweite Schicht der Security Policy (Anhang C.2) beinhaltet Bausteine, Gefährdungen und Maßnahmen des BSI-Grundschutzkataloges zugeschnitten auf die *Magnus Monetarus Bank*. Hierbei werden für jeden Baustein die Gefährdungen und zu jeder Gefährdung die zugehörigen Maßnahmen aufgelistet.

Die Bausteine sind für die folgenden Themen beschrieben:

1. Übergreifende Aspekte
2. Infrastruktur

3. IT-Systeme
4. Netze
5. Anwendungen

Die *übergreifenden Aspekte* behandeln unter anderem Gefährdungen und Maßnahmen für das Sicherheitsmanagement, die Organisation, das Datensicherungskonzept, den Datenschutz und Outsourcing. Die *Infrastruktur* beschäftigt sich mit dem Gebäude, dem Serverraum und dem Rechenzentrum. Das Thema *Netze* beschäftigt sich mit dem Netzmanagement, während *Anwendungen* Bausteine für das Sicherheitsmanagement beschreibt.

2.2.3. Handbücher (Security Policy Schicht 3, BSI 100-3)

Das Ziel der dritten Schicht der Security Policy ist, den Benutzern rollenbezogene Beschreibungen des Umgangs mit dem System zu bieten. Diese Handbücher sollen die Einarbeitung der Mitarbeiter in das System erleichtern und Fehler im Umgang mit dem System verhindern indem Abläufe ausführlich beschrieben werden.

Insbesondere soll die dritte Schicht der Security Policy solchem Fehlverhalten vorbeugen, das zu Sicherheitslücken oder Gesetzeswidrigkeiten führt. Die Handbücher sind deshalb so ausgelegt, dass sie auch als Nachschlagewerk dienen können.

Die Umsetzung der dritten Schicht findet sich in Kapitel 1 und Anhang A.

Teil II.

Abweichungen vom Pflichtenheft

3. Abweichungen vom Pflichtenheft

Autor: AV, MW, DS

In diesem Kapitel geht es um die Abweichungen zwischen dem implementierten System und den Zielen, die sich die PG555 in dem Pflichtenheft vorgenommen hat. Es wird erklärt, was bis jetzt erreicht wurde. Die Rollen, die im Pflichtenheft beschrieben wurden, werden lediglich mit deren Funktionalitäten aufgelistet. Insbesondere werden wir darauf eingehen, welche Unterschiede es zwischen der Spezifikation und der Implementierung gibt, mit Begründung, warum es zu diesen Abweichungen gekommen ist. Schließlich werden die Funktionalitäten aufgelistet, die nicht implementiert worden sind. Anschließend werden auch die automatisierten Prozesse dargestellt und zwar die, die nicht als Anwendungsfall bei den Rollen vorgekommen und erklärt worden sind.

Ein Grund dafür, dass wir manche Funktionen nicht implementieren konnten, ist, dass sehr viel Zeit in der Entwurfsphase aufgewendet werden musste. Weiterer Zeitverzug entstand dadurch, dass viel Zeit in die Einarbeitung in verschiedene Technologien gebraucht wurde. Diverse externe Absprachen aufgrund verschiedener Probleme, wie z.b. Interbank-Kommunikation und Schufa haben auch die PG im massiven Zeitverzug gebracht. Die somit verlorene Zeit konnte auch nicht durch Gegenmaßnahmen, wie sie in Kapitel 4.4 erläutert wurden, kompensiert werden. Aufgrund dessen konnten diverse, nachfolgend erläuterte Funktionen, nicht bis zum Abschluss des Projektes umgesetzt werden.

3.1. Kommunikationsschnittstellen

3.1.1. Kommunikation mit der Schufa

Die Anbindung an die Schufa wurde im Laufe der ersten Phase spezifiziert und konzipiert. Leider konnte aufgrund juristischer Probleme eine Anbindung an den Schufa-Testserver nicht erfolgen. Aufgrund dessen wurde, in Absprache mit den Kunden, zu einer Alternativlösung gegriffen. Diese Alternativlösung sieht vor, dass lediglich eine Dummy-Klasse implementiert wurde. Diese Dummy-Klasse bildet zufällige Antworten, die vorsehen, dass 95 % aller Anfragen positiv beantwortet werden.
Die Schnittstellen, seitens des von uns entworfenen Systems, sind für eine Verbindung mit der realen Schufa-Auskunft über die entsprechenden Server vorbereitet. Somit wäre eine nachträgliche Anbindung an diese jederzeit möglich.

3.1.2. Börse und Bafin

Diese Komponente wurde in der Entwurfsphase komplett geplant. Allerdings konnte sie aufgrund der oben genannten Gründe nicht mehr implementiert werden.

3.1.3. Interbank

Die Kommunikation mit der Interbank, also das Versenden von Überweisungen und Lastschriften an andere Banken, wurde in der ersten Phase der PG geplant. Die Spezifikationen der Schnittstelle [40] wurden erst zu Beginn der zweiten Phase der PG bekannt gegeben. Danach wurden bis Ende Januar die Spezifikationen in Zusammenarbeit mit den Betreuern auf die Anforderungen des Pflichtenhefts reduziert und angepasst.

Die Spezifikationen und die ursprüngliche Planung waren leider widersprüchlich, so dass erneute Planung notwendig wurde, zudem ging der Interbank-Server, der vom Lehrstuhl bereitgestellt werden sollte, erst in der Mitte der zweiten Phase in Betrieb. Es gab zudem Probleme den Cloud der *PG555* aufzusetzen, da durch lange Verzögerung die ursprünglich ausgewählte IRB für die Bereitstellung der Server (Hardware) nicht mehr in Frage kam, wurden die Server (Hardware) vom Lehrstuhl bereitgestellt. Hieraus entstanden große Verzögerungen in der Implementierung der Interbank-Komponente.

Der letzte Stand der Interbank-Kommunikation ist, dass die EBICS-Nachrichten [40] mit den abgesprochenen Zustands-Tags gelesen und erstellt werden können. Das Versenden und Empfangen der Nachrichten ist auch implementiert, jedoch traten immer wieder Probleme mit den Zertifikaten auf, die nicht behoben werden konnten.

Die Interbank-Nachrichten für Lastschriften (FIToFICstmrDrctDbt), Überweisungen (FIToFICstmrCdtTrf), Retourüberweisungen und Rücklastschriften (PmtRtr) sowie Statusreports (FIToFIPmtStsRpt)[47] können in vereinfachter Form auch gelesen, evaluiert, verarbeitet und versendet werden.

3.2. Rollen und deren Funktionalitäten

3.2.1. Kundenberater

Autor: MW

In diesem Unterkapitel werden alle Funktionen, die einem Kundenberater erlaubt sind, mit den im Pflichtenheft spezifizierten Funktionen verglichen. Wenn es eine Funktion gibt, die sowohl von Kundenberater als auch von anderen Akteuren benutzt werden kann, wird so eine Funktion nur in dem Unterkapitel beschrieben.

Dieses Unterkapitel besteht aus drei Teilen. Im ersten Teil werden die Funktionen aufgelistet, die entsprechend der Beschreibung im Pflichtenheft implementiert wurden.

Im zweiten Teil werden die Funktionen erläutert, die von der Spezifikation abweichen. In diesem Teil wird es auch begründet, wie diese Abweichungen zustande gekommen sind. Im letzten Teil werden Funktionen aufgelistet, die aus den oben genannten Gründen nicht implementiert werden können.

- Bereits implementierte Funktionen ohne Abweichung:
  - Aufgaben im Workflowsystem bearbeiten
  - Ausloggen
  - Daueauftrag ändern
  - Daueauftrag löschen
  - Daueauftrag tätigen
  - Einloggen
  - Geld abbuchen
  - Geld einzahlen
  - Kontaktdaten ändern
  - Konto entsperren
  - Kontoentsperrung beantragen
  - Konto eröffnen
  - Konto schließen
  - Kontosperrung ausführen
  - Kontoparameter ändern
  - Kunden anlegen
  - Kunde löschen
  - Lastschrift stornieren
  - Lastschrift tätigen
  - Nachricht verschicken
  - Schecks einreichen
  - Überweisung tätigen
- Bereits implementierte Funktionenen mit Abweichung:

  - Buchungen einsehen:
    Der Akteur kann alle verfügbaren Buchungen einsehen, aber hat keine Möglichkeit

Buchungen aus einem bestimmten Zeitraum abzurufen.

  - Kontostand einsehen:
    Kontostand weicht von der Beschreibung im Pflichtenheft ab, indem nach dem Einloggen kein Kontostand im Startfenster angezeigt wird. Der Grund dafür ist, dass ein Kunde mehrere Konten besitzen kann. Deswegen kann der Akteur den Kontostand einsehen, wenn ein bestimmtes Konto ausgewählt wurde.

- Nicht implementierte Funktionen – Banking:
  - Internationale Überweisung tätigen:
    Da es noch keine Möglichkeit gibt, eine Transaktion mit einer internationalen Bank auszuführen, wird diese Funktion nicht implementiert.
  - Konto sperren:
    Kundenberater als Akteur kann diese Funktion ausführen, aber das System nicht.
  - Karte beantragen:
    Da dieses System nur prototypisch implementiert ist und es keine Möglichkeit Karte einzusetzen gibt, wurde diese Funktion nicht implementiert.
  - Karte sperren:
    Da keine Karte angelegt wird, wird diese Funktion auch nicht implementiert.
- Nicht implementierte Funktionen – Börse (Begründung siehe Kapitel 3.1.2)
  - Automatische Order
  - Depotinformationen einsehen
  - Finanzinstrumente kaufen
  - Finanzinstrumente verkaufen
  - Kurse einsehen
  - Order löschen
- Nicht implementierte Funktionen (Begründung siehe Kapitel 3.1.2)
  - Nachricht lesen

3.2.2. Kunde

Autor: MW

Laut Pflichtenheft gibt es eine Spezialisierung unter den Akteuren, die bewirkt, dass der spezielle Akteur (Kundenberater) alle Anwendungsfälle ausführen kann, die der generelle Akteur (Kunde) auch ausführen kann. Zusätzlich gibt es zwei Anwendungsfälle, die der Kunde ausführen kann. [32]

- Nicht implementierte Funktionen:

  - Empfängerdaten auswählen
  - Empfängerdaten speichern

3.2.3. Geschäftsführer

Autor: AV

Im Folgenden wird auf die Funktionen, die ein Geschäftsführer bedienen darf, eingegangen. Wie es im Pflichtenheft definiert ist, übernimmt ein Geschäftsführer alle Funktionen, die auch von einem Kundenberater, Jurist, Eigenhändler, Controlling-Mitarbeiter oder Produktentwickler ausgeführt werden. Deswegen werden die Anwendungsfälle, die bereits bei der Rolle Kundenberater (siehe Kapitel 3.2.1) betrachtet werden, nicht berücksichtigt, da sie genau so für die Rolle Geschäftsführer funktionieren. Es gibt einen Anwendungsfall, der nur von einem Geschäftsführer ausgeführt werden darf.

- Nicht implementierte Funktionen:
  - Limits setzen

3.2.4. Produktentwickler

Autor: AV

Die folgenden Anwendungsfälle – *Ausloggen*, *Einloggen*, *Nachrichten lesen* und *Nachrichten verschicken* – sind im Abschnitt 3.2.1 beschrieben und funktionieren genau so auch für die Rolle Produktentwickler. Demnach werden diese hier nicht beschrieben. Außerdem darf laut Pflichtenheft ein Produktentwickler:

- Nicht implementierte Funktionen
  - Anonymisierte Daten einsehen

3.2.5. Controlling-Mitarbeiter

Autor: AV

Die folgenden Anwendungsfälle – *Aufgaben im Workflowsystem bearbeiten*, *Ausloggen*, *Einloggen*, *Kontostand einsehen*, *Nachrichten lesen*, *Nachrichten verschicken* und *Depotinformationen einsehen* – sind im Abschnitt 3.2.1 beschrieben und funktionieren identisch wie für den Controlling-Mitarbeiter. Aus diesem Grund wird an dieser Stelle auf dieser Funktionen nicht eingegangen. Folgende Anwendungsfälle dürfen von einem Controlling-Mitarbeiter ausgeführt werden:

- Bereits implementierte Funktionen ohne Abweichung:
  - Logs einsehen

- Bereits implementierte Funktionen mit Abweichung:
  - Fachabteilung stellt Daten bereit:
    Die Daten, die von dem Jurist angefragt wurden, werden ausgewählt, extrahiert und per Post an den Juristen geschickt.
- Nicht implementierte Funktionen:
  - Liste mit verdächtigen Finanzinstrumentstransaktionen prüfen

3.2.6. Kassierer

Autor: AV

Laut Pflichtenheft darf ein Kassierer sich einloggen und ausloggen, Geld auszahlen, Geld einzahlen und den Kontostand einsehen. Diese Anwendungsfälle werden bereits im Abschnitt 3.2.1 berücksichtigt und funktionieren ohne einen Unterschied. Aufgrund dessen werden sie in diesem Abschnitt nicht weiterverfolgt.

3.2.7. Jurist

Autor: DS

Die Anwendungsfälle *Nachrichten lesen* und *Nachrichten verschicken* funktionieren genauso wie sie bei dem Kundenberater, Kapitel 3.2.1, beschrieben worden sind. Weiter folgt eine Auflistung der

- bereits implementierten Funktionen ohne Abweichung:
  - Kontosperrung beantragen
  - Konto entsperren
  - Titel mit Pfändung und Vollstreckung eintragen
  - Liste mit verdächtigen Überweisungen prüfen
  - Antrag auf Entsperrung bestätigen bzw. ablehnen
  - Aufgaben im Workflowsystem bearbeiten
- bereits implementierten Funktionen mit Abweichung:
  - Bonität des Kunden von der Schufa einholen
    Hierfür wurde eine Dummy-Klasse erstellt, die mit einer Wahrscheinlichkeit von 99 % eine positive Auskunft liefert.
  - Informationen über große Kredite erhalten (siehe Bonität)
- nicht implementierten Funktionen:
  - Kontodatenliste beantragen

Dieser Anwendungsfall wurde durch die Aufgaben im Workflowsystem gelöst. Ein Jurist darf nur dann die für ihn relevanten Kunden- und Kontendaten einsehen, wenn er vorher eine Aufgabe erstellt hat und diese von einem Kunden genehmigt wurde.
Der andere Fall ist, wenn vom Kundenberater eine Aufgabe, wie z.B. Bonität prüfen, erstellt wurde. Im diesem Fall werden die Daten des Kunden mitgeschickt und dem Jurist zur Einsicht freigegeben.

3.2.8. System

Autor: DS

In diesem Unterkapitel werden die von System automatisiert ausgeführten, bereits implementieren, Funktionen aufgelistet, bzw. die Funktionen die laut Spezifikation automatisiert ausgeführt werden sollten, nur as Gründe, die in der Einleitung (vgl. Kapitel 3) angegeben wurden, nicht oder mit kleinen Abweichungen, umgesetzt worden sind.

- Bereits implementierte Funktionen ohne Abweichung:
  - Logs erstellen
  - Überweisung auf Gültigkeit prüfen
- Bereits implementierte Funktionen mit Abweichung:
  - Block von Lastschrift erhalten
  - Block von Lastschrift verschicken
  - Block von Überweisung erhalten
  - Block von Überweisung verschicken
  - Internationale Überweisung auf Gültigkeit prüfen
  - Lastschrift prüfen
    Zu den oberen sechs Anwendungesfällen siehe Kapitel 3.1.3
  - Zinsen buchen
    Zinsen werden einmal pro Tag berechnet aber nicht abgebucht. Grund dafür ist, dass es kein Dispoüberziehung geprüft wird.
- Nicht implementierte Funktionen:
  (vgl. Kapitel 3 und 3.1.2)
  - Börsentransaktionen der Bank / des Kunden durchführen
  - Dispoüberziehung Prüfen
  - Gegenlastschrift Veranlassen
  - Limits prüfen

- Überweisung auf Verdächtigkeit prüfen
- Kontostammdaten historisieren

3.2.9. Systemadministrator

Autor: DS

Bei dem Systemadministrator gibt es nur bei zwei der für ihn verfügbaren Funktionen kleine Abweichungen vom Pflichtenheft. Als erstes werden die Funktionen aufgelistet, bei denen es keine Unterschiede zu der Spezifikation gibt.

- Bereits implementierte Funktionen ohne Abweichung:
  - Konfiguration des Systems
  - Starten
  - Nutzerrechte/Nutzer anlegen, löschen, vergeben
  - RSA Schlüssel ändern
  - Backup einspielen
  - Aktuellen Systemstatus einsehen
  - Logs einsehen
  - Backup erstellen
- Bereits implementierte Funktionen mit Abweichung:
  - Logs auswerten:
    Bei dieser Funktionalität findet eine Abweichung von Pflichtenheft statt. Das Auswerten der Logdateien geschieht nicht automatisch sondern manuell. Dieses Verfahren wird näher im Anhang beim Kapitel D im Unterkapitel BPM beschrieben.

3.2.10. Eigenhändler

Autor: DS

Diese Rolle besitzt die Funktionen, die der Kundenberater auch hat (vgl. Kapitel 3.2.1). Deswegen werden sie hier nicht detailliert beschrieben.

Die Funktionen *Einloggen* und *Ausloggen* sowie *Nachrichten lesen* und *Nachrichten verschicken* sind implementiert und es kann einen Nutzer mit der Rolle Eigenhändler angelegt werden, der diese Funktionen tätigen darf. Da aber die Hauptfunktionalitäten aus der Börse-Komponente (vgl. Kapitel 3.1.2) nicht vorhanden sind, gibt es in der Endversion dieses Banksystems keine Möglichkeit einen Nutzer dieser Rolle anzulegen.

3.2.11. Marketing

Autor: DS

Bei dieser Rolle kommen Anwendungsfälle – *Einloggen, Ausloggen, Aufgaben in Workflowsystem bearbeiten, Nachrichten lesen, Nachrichten verschicken* – die in Abschnitt 3.2.1 schon beschrieben wurden. Zusätzlich gibt es einen Anwendungsfall – *Anonymisierte Daten einsehen* – der im Kapitel 3.2.4 erklïrt wird.

Aufgrund dessen wird bei der Marketing-Abteilung nicht auf die oben gennanten Funktionen weiter eingegangen.

3.3. Automatisierte Prozesse

Autor: DS

Bei den bankinternen Prozesse gibt es eine Abweichungen. Diese werden automatisch durch das System ausgeführt. Das System führt die entsprechenden Aktionen durch und informiert die involvierten Akteure über die Vorgänge. So erfolgt beispielsweise das Überprüfen von Überweisungen auf Validität automatisch, wenn diese getätigt oder empfangen werden. Wenn sich hierbei die Überweisung als verdächtig herausstellt, erfolgt eine Benachrichtigung der Rechtsabteilung.
Die bankinternen automatisierten Prozesse sind:

- bereits implementierte Funktionen ohne Abweichung:
  - Retourüberweisung eintragen
  - Stornogebühren abbuchen
  - Zinsen und Kontoführungsgebühren verbuchen
- bereits implementierte Funktionen mit Abweichung:
  - System Testen:
    Anfangs war es so gedacht, dass der Anwendungsfall *System testen* möglichst automatisiert ablaufen soll. Das bedeutet, dass die erstellten Logs dazu genutzt werden sollten, um mit Business Process Mining (BPM) die Abläufe der Bank automatisch zu analysieren und bei Abweichungen vom spezifizierten Verhalten Anomaliemeldungen an den Systemadministrator zu senden. Die bisherige Vorgehensweise der Mitarbeiter Fehler im Ablauf manuell zu finden wäre somit durch diesen automatischen Test unterstützt. Das ist uns leider nicht gelungen. Der Grund dazu war, dass wir kein passendes und gut dokumentiertes Tool gefunden haben, dass uns bei dieser Aufgabe helfen konnte. So ein Tool oder Plug-In in der für dieser PG geplanten Zeit selber zu entwickeln, hätte die Zeit- und Aufgabenstellungsrahmen dieser PG gesprengt.

Teil III.

Maßnahmen und Erfahrungen

4. Maßnahmen des Managements

Autor: RH

4.1. Projektmanagement

Dieses Kapitel befasst sich mit allgemeinen Problemen in einem Projekt. Entsprechende Lösungen, die während des Projektes ausgearbeitet wurden, werden vorgestellt.

4.1.1. Zusammenfassung des Projektmanagements im Projektplan

Im Projektplan werden zwei Phasen des Projektes beschrieben, wobei sich die Phasen an die Semester anordnen, in denen das Projekt stattfand. Die erste Phase betrifft die Planung des zu erstellenden Banksystems, während die zweite Phase dessen Umsetzung, also Implementierung, Test und Dokumentation berücksichtigt. Die erste Phase wurde in folgende Arbeitspakete (abgekürzt AP) eingeteilt, wobei die einzelnen Arbeitspakete voneinander abhängen [33]:

1. AP Projektplan
2. AP Anforderungsanalyse
3. AP Spezifikation
4. AP Pflichtenheft
5. AP GUI-Entwurf
6. AP Technologieauswahl
7. AP Architekturentwurf

Die Vorgehensweise in der zweiten Phase unterscheidet sich von der Vorgehensweise in der ersten Phase dadurch, dass die Methode *Extreme Programming(XP)* [27] verwendet wurde, wobei nicht alle Praktiken davon vollständig durchgeführt wurden. Dies liegt daran, dass die Möglichkeit einer räumlichen Nähe fehlte und fachliche Qualifikationen nicht hinreichend vorhanden waren. Deshalb wurde vor allem auf die Praktiken *Räumlich Zusammensitzen* und *Komplettes Team* verzichtet und die Praktik *Wochenzyklus* etwas abgewandelt (siehe Kapitel 4.3).
Die Vorteile von *Extreme Programming* liegen vor allem in der inkrementellen und in der testgetriebenen Entwicklung, die die Qualität des Systems steigern sollen. Außer-

dem wird dadurch im Gegensatz zu anderen Entwicklungsmodellen, wie zum Beispiel dem Spiralmodell [31], mehr Zeit eingespart. Die Mitglieder suchen sich die Aufgaben selbstständig aus, so dass das Zuteilen von Aufgaben nicht notwendig ist. Im Falle, dass ein Mitglied keine Aufgaben bearbeitet, wird ihm eine der vorhandenen Aufgaben zugeteilt. Falls es mehr Aufgaben als Mitglieder geben sollte, werden Prioritäten festgelegt bzw. nach Möglichkeit Aufgaben gruppiert, so dass die Schwierigkeit der einzelnen Aufgaben auf einem ähnlich hohen Niveau ist [33].

4.1.2. Vorgehen in der zweiten Projektphase

Da die zweite Phase mithilfe einer agilen Methode durchgeführt werden sollte, musste diese ebenfalls geplant werden. *Extreme Programming* wurde als geeignetes Entwicklungsmodell ausgewählt (siehe Abschnitt 4.1.1) und die Implementierungsphase in Teilphasen aufgeteilt, in denen das System schrittweise erweitert wurde. Zusätzlich zu den einzelnen Implementierungsarbeiten wurden Tests und Dokumentationen angelegt, um die Qualität des Systems sicherzustellen.

4.1.2.1. 1. Teilphase

1.1 GUI
1.2 Programmskelett
1.3 Datenbank
1.4 Schnittstellen

In dieser Teilphase wurde nach dem Top-Down Vorgehen implementiert. Hierbei stelle man sich das System in einem 3-Schichten-Modell vor. Die obere Schicht dient der Präsentation und der Ausgabe, die Mittlere der Geschäftslogik und die Untere der Infrastruktur. Die Präsentation dient der Interaktion mit dem Benutzer des System und hat deshalb als Hauptbestandteil die GUI. Die Geschäftslogik beinhaltet die ablaufenden Prozesse und damit auch das Programmskelett in Form von Klassen. Die Infrastrukturschicht soll hauptsächlich die Persistenz der Daten sicherstellen und die Kommunikation mit externen Diensten bewerkstelligen. Für die Top-Down Implementierung hatte dies zur Folge, dass die GUI am Anfang erstellt wurde. Danach wurde das Programmskelett mit den einzelnen Klassen und Beziehungen angelegt, wobei diese noch nicht vollständig sein mussten und auch keine Funktionalität boten. Abschließend wurde bei der Teilphase eine Datenbank erstellt und die Schnittstellen zu externen Kommunikationspartnern angelegt. Dies konnte parallel stattfinden, da sich diese Schritte nicht gegenseitig beeinflussen.

Nach der Fertigstellung der ersten Teilphase wurde folglich ein vollständiges Gerüst des Systems angelegt. Die restlichen Teilphasen erweitern die leeren Klassen um Funktionalitäten.

4.1.2.2. 2. Teilphase

2.1 Authentifikation und Authorisierung
2.2 Kontodatenkomponente
2.3 Kundendatenkomponente

In der zweiten Teilphase wurden die zentralen Komponenten des Systems integriert. Unter Authentifikation und Authorisierung sollte nicht nur die Seite des Benutzers, sondern auch die Seite der Bank betrachtet werden. Beide Seiten müssen sich gegenseitig authentifizieren. Dazu muss der Nutzer sich über eine Passworteingabe beim Onlinebanking-System anmelden. Um sicherzustellen, dass es sich wirklich um das Onlinebanking-System der Bank handelt, stellt dieses ein Zertifikat bereit, mit dem sich dies prüfen lässt. Zusätzlich wurde die Nutzerverwaltung eingeführt, die alle Akteure des Systems betrifft und das Anlegen ihrer Rechte und Eigenschaften erlaubt.

Kontodatenkomponente beinhaltet die Implementierung der Kontoverwaltung. Dazu gehören die unterschiedlichen Kontoarten und ihr Zusammenspiel. Die Kundendatenkomponente ist das Gegenstück zur Kontodatenkomponente und verwaltet die Kundendaten. Hierzu gehört zum Beispiel das Anlegen eines neuen Kunden, die Änderung der persönlichen Informationen oder der Zugriff durch den Kunden auf das System. Durch die frühe Integration dieser wichtigen Komponenten wurde sichergestellt, dass diese in den weiteren Integrationstests einbezogen und damit am meisten getestet werden.

4.1.2.3. 3. Teilphase

3.1 Transaktionskomponente
3.3 Zinskomponente
3.4 Kreditkomponente

Die dritte Phase beinhaltet die Integration der einzelnen Komponenten für den Geldtransfer. Dazu gehört die Transaktionskomponente, mit allen im System vorhandenen Geldtransaktionen, wie zum Beispiel Überweisungen, Lastschriften, Ein- und Auszahlungen oder Daueraufträgen. Des Weiteren wurde in dieser Teilphase weitere Konto- und Kundenfunktionen integriert, die in der zweiten Teilphase nicht vollständig fertiggestellt werden konnten. Dazu zählt zum Beispiel das Verwalten von Buchungen zu jedem Konto oder das Speichern und Verwalten der Empfänger der jeweiligen Transaktion.

Die Zinskomponente hat die Aufgabe die Zinsen von den einzelnen Konten zu berechnen und zu buchen. Dies soll zu bestimmten Zeiten und bestimmten Intervallen geschehen und wird von einem Prozess angestoßen und durchgeführt.

Bei der Kreditkomponente handelt es sich um die Verwaltung der Kredite, die als Konto im System angelegt werden. Hierbei prüft das System die Kreditkonten auf die Einhaltung der Konditionen und meldet notfalls Unstimmigkeiten an eine bestimmte Rolle [32].

4.1.2.4. 4. Teilphase

4.1 Börsenkomponente

Bei der vierten Teilphase wird die Börsenkomponente erstellt. Hierbei handelt es sich um die Funktionen, die den Handel und die Verwaltung von Finanzinstrumenten betreffen. Dazu gehört auch das Einsehen von aktuellen Kursen eines Finanzinstrumentes oder das Anlegen einer automatischen Order, die beim Erreichen eines bestimmten Kurses eine automatische Order anstößt. Zusätzlich sollten die Funktionen und Eigenschaften des Eigenhändlers implementiert werden.

4.1.2.5. 5. Teilphase

5.1 Backup
5.2 Replikationsserver

Die fünfte Teilphase beinhaltet die Komponente, die für das Backup des Systems verantwortlich ist. Dazu zählt sowohl das Erstellen, als auch das Einspielen des Backups auf den Spiegelserver [32]. Der Replikationsserver, der Kontostammdaten für die BaFin und BZSt bereitstellt wird ebenfalls in dieser Teilphase erstellt und vor allem das Übertragen der Kontostammdaten auf diesen Server verwaltet.

4.1.2.6. 6. Teilphase

6.1 Systemangelegenheiten

Unter Systemangelegenheiten fallen vor allem die Funktionen, die der Systemadministrator erledigt. Außerdem beinhaltet die Teilphase das Einbinden von *Business Process Mining*[32] und das automatische Erkennen von Anomalien im System.

4.2. Teammanagement

Im Folgenden wird auf das Thema Teammanagement während der gesamten Projekts eingegangen. Hierzu werden die im Projektplan ([33]) genannten Maßnahmen erwähnt, die im Projekt durchgeführten Maßnahmen vorgestellt und schließlich einen Rückblick, der den Erfolg der Methoden bewertet.

4.2.1. Zusammenfassung des Teammanagements im Projektplan

Im Projektplan wurde für das Teammanagement ein Konfliktmanagement innerhalb der Gruppe geplant. Dieses beinhaltet die Maßnahmen Konflikte rechtzeitig zu erkennen, richtig zu analysieren und konstruktiv zu bearbeiten. Ziel dieser Maßnahme ist vor allem das Arbeitsklima zu verbessern und die Zufriedenheit innerhalb der Gruppe zu erhalten.

Außerdem wurde im Projektplan eine Maßnahme zum strukturierten Diskutieren bei Meinungsverschiedenheiten bestimmt. Für die Diskussion wird hierbei angenommen, dass zwei verschiedene Meinungen über einen Sachverhalt bezüglich des Projekts innerhalb der Gruppe vorhanden sind. Die Maßnahme legt nun fest, dass bei Meinungsverschiedenheiten zwei gleichgroße Gruppen, deren Zusammensetzung zufällig gewählt ist, gebildet werden. Die Gruppen überlegen sich innerhalb von festgelegten Zeiten, Argumente über den Sachverhalt, stellen diese der anderen Gruppe vor und nehmen zu den Argumenten der Gegengruppe jeweils Stellung. Ein Moderator des Treffens achtet auf Einhaltung von Zeiten und weiteren Regeln. Zur Lösung der Meinungsverschiedenheit dient daraufhin ein Mehrheitsentscheid. Falls bei der Abstimmung keine Mehrheit zustande kommt, entscheidet der Projektleiter über den Sachverhalt.

Eine weitere Maßnahme wurde im Projektplan bezüglich des Gesamtkontexts des Projekts definiert. Die Maßnahme sieht vor, dass der Projektmanager und der Moderator bei jedem Treffen den Zusammenhang der einzelne Arbeitsphasen betrachten und auf Unstimmigkeiten überprüfen. Eine Unstimmigkeit kann zum Beispiel Zeitverzug sein. Ein anderes Beispiel ist, dass die Auslastung innerhalb einer Arbeitsgruppe zu hoch ist. Im Falle eines Zeitverzuges wird versucht die Parallelität des Arbeitens zu maximieren. Dies ist jedoch nur Möglich, wenn dies die Abhängigkeiten der AP's zulassen. Im Falle der überhöhten Auslastung ist die Gegenmaßnahme die Aufgabe durch zusätzliche Teammitglieder bearbeiten zu lassen.

Der Projektplan sah weiterhin Konflikttypen bezüglich eines möglichen Vergehens einzelner Projektmitglieder vor, um mögliche Konsequenzen zu ermitteln. Die Konflikttypen sind:

1. Leichte Vergehen (z.B. Verspätungen zu Treffen) ohne weitere Konsequenzen, außer bei wiederholten Vergehen mit der Konsequenz einer zusätzlichen Arbeitszuteilung
2. Mittlere Vergehen (z.B. Arbeitsverweigerung) mit der Konsequenz einer Abmahnung, wobei drei Abmahnungen einem schweren Vergehen entsprechen

3. Schwere Vergehen (z.B. bei dauerhafter Arbeitsverweigerung) mit der Konsequenz, dass ein Ausschluss der Person aus der PG angestrebt wird

Um die Kommunikation innerhalb der Gruppe zu verwalten, wurde im Projektplan die Unterstützungsplattform Redmine ([16]) ausgesucht, um das dort vorhandene Wiki als Wissensverwaltung zu nutzen. Das Wiki ermöglicht das Lesen und Schreiben von Texten auf einer Webseite. Außerdem wurde im Projektplan ein strukturiertes Vorgehen bei den regelmäßigen Treffen der Gruppe festgelegt, um die Treffen für eine möglichst optimale Absprache unter den Teammitgliedern zu nutzen. Zunächst wurde dazu geplant, ein Protokoll des Treffens durch eine Person aus der Gruppe schreiben zu lassen. Der Protokollant wechselt beim darauffolgenden Treffen und der vorherige Protokollant wird zum Moderator des Treffens. Dieser plant und leitet das Treffen. Beim Treffen werden die bis zum Treffen zu bearbeitenden Aufgaben, deren aktueller Stand und die Aufgaben für das darauffolgende Treffen besprochen.

4.3. Qualitätsmanagement

Qualitätsmanagement sollte in der gesamten Laufzeit des Projekts angewandt werden, um so nicht nur die Qualität des zu erstellenden Systems, sondern auch die Qualität der herangezogenen Prozesse sicherzustellen.

4.3.1. Zusammenfassung des Qualitätsmanagements im Projektplan

Innerhalb des Projektplans[33] ist ein Qualitätsplan einbezogen, der vor allem Qualitätsziele und -mechanismen beinhaltet. Zusätzlich sollte der Qualitätsplan Konventionen und Standards enthalten. Allerdings wurden diese nicht in den Projektplan aufgenommen, da sie im Laufe des Projekts erstellt bzw. ergänzt wurden. Der zentrale Aspekt ist dabei, dass die genauen Anforderungen während der Erstellung des Projektplans noch nicht vorlagen und mehrere Entscheidungen bezüglich Technologien und Architektur noch bevorstanden. Der im Projektplan enthaltene Qualitätsplan ist eine generische Zusammenfassung von Qualitätszielen zu jedem Arbeitspaket und von möglichen, zu verwendenden Qualitätsmechanismen.

4.3.1.1. Qualitätsziele

Die Qualitätsziele sollen für jedes Arbeitspaket (AP) die Zeitpunkte festmachen, an denen eine erfolgreiche Erreichung des Ziels vom Paket erkennbar ist. Es wurden folgende Qualitätsziele bezüglich der Arbeitspakete festgelegt [33]:

1. AP Projektplan: Vollständigkeit; Korrektheit
2. AP Anforderungsanalyse: Vollständigkeit; Einhaltung der Compliance; Betrachtung aller Sicherheitsrisiken

3. AP Spezifikation: Vollständigkeit; Korrektheit der Diagramme; Einhaltung der Compliance
4. AP Prozessmodellierung: Vollständigkeit; Korrektheit der Diagramme; Schutz der Daten; Schutz der Personen
5. AP GUI erstellen: Benutzbarkeit der GUI; Vollständigkeit; Korrektheit; Robustheit
6. AP Technologieauswahl: Vollständigkeit; Notwendigkeit; Konsistenz; Einfachheit
7. AP Architekturentwurf: Vollständigkeit; Korrektheit; Robustheit; Zuverlässigkeit; Schutz
8. AP Zwischenbericht: Vollständigkeit; Korrektheit
9. AP Implementierung und Test:Vollständigkeit der Tests, der Komponenten, der Dokumentation; Korrektheit; Verständlichkeit des Codes; Zuverlässigkeit; Schutz; Effizienz; Robustheit gegen Angriffe; Robustheit unter Last
10. AP Endbericht: Vollständigkeit des Endberichts; Korrektheit

In jedem Arbeitspaket wird das Qualitätsziel *Vollständigkeit* geprüft, wobei dieses Ziel für jedes Paket einzeln definiert werden musste, indem man einen Sollwert vorgibt. Dieser Sollwert wird beim Messen mit dem gemessenen Istwert verglichen, um den Grad der Vollständigkeit zu ermitteln.

4.3.1.2. Qualitätsmechanismen

Als Qualitätsmechanismen zur Sicherstellung und Überprüfung der Qualitätsziele wurden folgende Verfahren herausgestellt [33] [44]:

- Review: Eine visuelle Prüfung eines Dokuments oder eines Diagramms mit dem Ziel Fehler und Unstimmigkeiten zu erkennen. Wird von mehreren, meist unabhängigen Personen getrennt durchgeführt und die Anmerkungen an den Autor in Form von Verbesserungsvorschlägen oder Fragen weitergeleitet.
- Inspektion: Eine Weiterentwicklung des Reviews. Dabei versammeln sich die Reviewer, der Autor und ein Moderator, um die Fehler innerhalb des Dokuments mit Hilfe einer gemeinsamen Prüfung zu unterziehen. Fragen und Unklarheiten können sofort geklärt werden. Allerdings bedeutet dieser Schritt mehr Zeitaufwand als bei einem Review.
- Goal-Question-Metric-Ansatz: Anhand von Qualitätszielen, werden Fragen herausgearbeitet, die an die jeweilige Phase gestellt werden, um schließlich daraus Metriken abzuleiten und zu messen. Der Vorteil liegt in der Messung nur sinnvoller Metriken und soll Zeit und Kosten sparen.
- Statische Codeanalyse: Ein Programmcode wird mit Hilfe eines automatischen Tools einer Prüfung unterzogen, in der vor allem syntaktische Fehler gefunden

werden. Außerdem werden Merkmale gefunden, die häufig einen Fehler verursachen oder die Verständlichkeit des Codes beeinträchtigen.

- Test: Ein Test prüft das System dynamisch auf die Einhaltung der Anforderungen. Dabei werden zu verschiedenen Eingaben Soll-Ausgabewerte vorgegeben und die Ist-Werte mit diesen verglichen. Man unterscheidet den Unit-, Integrations- und Systemtest. Zusätzlich wird der Penetrationstest ausgeführt, der das System auf seine Sicherheit überprüft.
- Style Check: Ein *Style Checker* soll die Verständlichkeit des Programmcodes verbessern, indem die für das Projekt definierten Konventionen auf ihre Einhaltung geprüft werden.

4.3.2. Vorgehensweise im Qualitätsmanagement während des Projektes

In der ersten Phase der Entwicklung, die alle Arbeitspakete vor der Implementierung enthält, wurden drei Qualitätsmechanismen Review, Inspektion und Goal-Question-Metric-Ansatz(GQM) angewandt.

4.3.2.1. AP Projektplan

Bei der Erstellung des Projektplans wurde in Gruppen von bis zu drei Personen gearbeitet, die jeweils einen Teilbereich des Plans bearbeitet haben. Diese Gruppen behielten wir auch bei den Reviews, wobei jede Gruppe den Text einer anderen Gruppe korrigiert hat. Somit waren die Autoren und die Reviewer unabhängig voneinander. Der GQM-Ansatz wurde in diesem Paket nicht angewendet, da unserer Ansicht nach bei Dokumenten Reviews mit anschließender Fehlerkorrektur ausreichen, um die Qualitätsziele sicherzustellen.

Die Vollständigkeit des Projektplans wurde allerdings nicht erreicht, da die angenommenen Sollwerte falsch waren, was dazu führte, dass der Projektplan dreimal überarbeitet werden musste.

4.3.2.2. AP Anforderungsanalyse

Bei der Anforderungsanalyse gab es die Schwierigkeit, dass die Vollständigkeit und auch die Korrektheit der herausgestellten Anforderungen nicht klar überprüft werden konnten. Dies liegt vor allem daran, dass es wesentlich von der Eigeninterpretation abhängt.

Die durchgeführten Reviews richteten sich nach dem Lastenheft und es wurde bezüglich folgender Qualitätsziele der GQM-Ansatz verwendet. Allerdings ist es fragwürdig, ob bei der Anforderungsanalyse sinnvoll zum Einsatz gekommen ist.

Qualitätsziel: Vollständigkeit

Frage 1: Gibt es Unklarheiten, die im Lastenheft auftauchen?
Metrik 1: Anzahl Unklarheiten pro Anforderung
Falls Unklarheiten bei einer gefundenen Anforderung existieren, bedeutet das, dass eine Anforderung möglicherweise unvollständig ist. Dies hat zur Folge, dass Fragen an den Kunden formuliert werden müssen.

Qualitätsziel: Korrektheit
Frage 1: Sind die extrahierten Anforderungen auch so im Lastenheft vorhanden?
Metrik 1: Anzahl überflüssiger Anforderungen
Geht eine Anforderung weder aus dem Lastenheft, noch aus den Antworten des Kunden hervor, kann es sein, dass diese überflüssig ist. Die Anzahl der überflüssigen Anforderungen liefert den Grad der Korrektheit der Anforderungsanalyse.

Frage 2: Sind Anforderungen in sich stimmig? Gibt es keine Konflikte?
Metrik 1: Anzahl Anforderungen insgesamt
Metrik 2: Anzahl Inkonsistenzen zwischen den Anforderungen

Bei diesen Metriken war das Ziel die Konsistenz zu messen. Wurde mindestens eine Inkonsistenz gefunden, so mussten die Anforderungen überarbeitet und einem erneuten Inkonsistenzcheck unterzogen werden.

4.3.2.3. AP Spezifikation

Die Spezifikation beinhaltete vor allem die Erstellung des Pflichtenheftes, das aus mehreren Teilen besteht. Zum einen mussten Use-Case-Diagramme [18] zu den Anforderungen erstellt, die Spezifikationstabellen zu jedem Use-Case angelegt und die Prozesse dazu modelliert werden. Zusätzlich mussten Texte, wie Beschreibungen der Diagramme und der Tabellen oder Begriffsdefinitionen geschrieben werden. Da innerhalb der PG555 mehr als die Hälfte der Teilnehmer einen Migrationshintergrund aufweisen, konnten diese Reviews nur von wenigen Personen durchgeführt werden.

Wegen Zeitmangel konnte das GQM-Verfahren nicht durchgeführt werden, was auch an der Qualität des Pflichtenheftes bemerkbar war. Die Qualitätsziele Vollständigkeit und Korrektheit konnten beim Pflichtenheft erst nach mehreren Review- und Inspektionsdurchläufen sichergestellt werden.

4.3.2.4. AP GUI-Entwurf

Beim GUI-Entwurf wurden mehrere Reviews durchgeführt und somit versucht die Qualität der GUI, vor allem im Hinblick auf die Benutzbarkeit zu optimieren. GQM-Ansatz konnte bei diesem Qualitätsziel sinnvoll zum Einsatz gebracht werden, indem die Elemente und ihre Anordnung betrachtet wurden. Dabei wurden Sollwerte für die Anzahl

von Elementen angenommen oder einheitliche Regelungen für die Anordnung von Elementen festgelegt. So kann eine sinnvoller Sollwert für die Anzahl der Elemente auf 5 festgelegt werden. Somit dürften zum Erhalt der Übersicht nicht mehr als 5 Elemente auf einer Teilseite zu sehen sein. Wurden diese Werte überschritten, musste die GUI überarbeitet und überflüssige Elemente verschoben oder weggelassen werden.

Qualitätsziel: Benutzbarkeit
Frage 1: Ist die GUI intuitiv zu bedienen?
Metrik 1: Anzahl Buttons pro Seite
Metrik 2: Anzahl Textfelder pro Seite
Metrik 3: Anzahl Menüeinträge in der Navigation
Metrik 4: Anzahl Beschreibungen pro Seite

Bei diesem Qualitätsziel gab es die Idee, dass zu viele Elemente auf einer Seite den Benutzer überfordern würden. Um dies zu verhindern messen die Metriken die Anzahl der Elemente pro Seite. Die Schwellen für die Anzahl an Elementen weichen bei unterschiedlichen Seiten von einander ab.

4.3.2.5. AP Zwischenbericht

Beim Zwischenbericht wurden Reviews von mindestens drei unabhängigen Personen durchgeführt, um so die Vollständigkeit und Korrektheit gewährleisten zu können. In besonders schwierigen Fällen, wie z.B. wenn ein Teiltext durch jemanden mit Migrationshintergrund geschrieben wurde, erfolgte meist ein weiteres Review durch eine weitere Person. Diese stand dann entsprechend auch bei Fragen zur deutschen Sprache bereit.

Da auch die Abgabe des Zwischenberichtes erst nach mehrmaligen hin und her schicken zwischen Projektgruppe und Betreuer erfolgen konnte, wurden die Anmerkungen seitens der Betreuer protokolliert und korrigiert. Bei den darauffolgenden erneuten Reviews wurde insbesondere auf die Einhaltung der Korrekturwünsche der Betreuer geachtet, um eine möglichst zufriedenstellende Version des Zwischenbericht abzugeben.

4.3.2.6. Qualitätssicherung während der Entwicklung

Die zweite Phase wurde nach dem Vorgehensmodell *eXtreme Programming (XP)* [27] organisiert, wobei sie die Implementierung, die Dokumentation und den Test des Systems beinhaltet. Da nach *XP* der Ansatz *Test First* [27] vorgeschlagen wird, wurden die BlackBox-Tests [33],die über Schnittstellen die Anforderungen testen, vor dem Code erstellt. Zusätzlich wurden Reviews durch das vier-Augen Prinzip durchgeführt. Dabei wurden die Paare, die eine Programmieraufgabe zusammen bearbeiteten, am Anfang eines Programmierabschnitts festgelegt. So könnte z.B. eine Person den Test erstellen und die Andere die Implementierung. Allerdings wurden die Reviews gegenseitig ganz nach dem Motto *check before check-in* durchgeführt, was bedeutet, dass die gröbsten Fehler

beseitigt wurden, bevor eine Integration ins Hauptsystem erfolgte. Eine Abwandlung ist, dass die Iterationen nicht in einem Wochenzyklus angesetzt sind, da eine Woche für die meisten Iterationen zu kurz zu sein schien. Stattdessen wurden zwei- oder dreiwöchige Iterationen durchgeführt.

Nach dem lokalen Check des Programmcodes, konnte ein zentraler Integrationstest durchgeführt werden, wobei für noch unfertige Komponenten *Mock-Up*-Objekte erstellt wurden. Dabei dienten diese Objekte als Stellvertreter für unfertige Klassen und boten nur sehr begrenzte Funktionalität, so dass das System immer lauffähig blieb und so eine *Continuous Integration*[27] durchgeführt werden konnte.

4.4. Risikomanagement

Bei Gruppen, die neu zusammengesetzt werden, wie bei einer Projektgruppe, kann es zu sehr vielen unvorhergesehenen Ereignissen kommen, die ein Risiko für die Erreichung des festgelegten Ziels darstellen. Diese Risiken zu erkennen und Strategien für die Beherrschung dieser zu entwickeln, ist das Ziel des Risikomanagements.

4.4.1. Zusammenfassung des Risikomanagements im Projektplan

Im Projektplan wurden die folgenden vier Stufen des Risikomanagements vorgestellt [33].

1. Risikoerkennung
2. Risikoanalyse
3. Risikoplanung
4. Risikoüberwachung

Diese einzelnen Phasen sollen im Weiteren Verlauf genauer betrachtet werden.

4.4.1.1. Risikoerkennung

Bei der Risikoerkennung ist die hauptsächliche Aufgabe, mögliche Risiken vor dem Eintreten zugehöriger Störungen zu erkennen. Im Projektplan wurden folgende Risiken als relevant für die Projektgruppe identifiziert.

- Personalausfall
- Konflikte zwischen den Mitgliedern
- Schlechte Moral bei den Mitgliedern
- Sprachliche Barrieren

Tabelle 4.1.: Risikoanalyse: Je nach Schaden, den der Eintritt eines Risikos mit sich bringt und dessen Eintrittswahrscheinlichkeit, wird die Gefahr des Risikos durch ein Ampelsystems angezeigt.

	Eintrittswahrscheinlichkeit				
Schaden bei Eintritt			Niedrig (wenig wahrscheinlich <25%)	Mittel (wahrscheinlich 25%-75%)	Hoch (Sehr wahrscheinlich >75%)
			1	2	3
	Tolerierbar (weniger relevante Ergebnisse gefährdet)	1	Grün	Grün	Grün
	Ernst (Teilergebnisse gefährdet)	2	Grün	Gelb	Gelb
	Katastrophal (Erfolg des gesamten Projektes gefährdet)	3	Gelb	Rot	Rot

- Zeitverzug
- Technische Risiken

4.4.1.2. Risikoanalyse

Bei der Risikoanalyse werden die im ersten Schritt identifizierten Risiken bezüglich ihrer Eintrittswahrscheinlichkeit und bezüglich des Schadens, der durch eine risikobasierte Störung entsteht, bewertet. Dabei zog man Tabelle 4.1 heran, die ein Ampelsystem verwendet, um die Bedeutung der Risiken für den Projekterfolg zu veranschaulichen [33].

Die geschätzte Eintrittswahrscheinlichkeit eines Risikos wird in eins der drei dargestellten Intervalle (wenig wahrscheinlich, wahrscheinlich, sehr wahrscheinlich) eingeordnet. Der Schaden wird parallel dazu betrachtet und in drei Stufen (tolerierbar, ernst, katastrophal) eingeordnet, wobei diese Einordnung vom Risiko abhängt. Die Eintrittswahrscheinlichkeit und der mögliche Schaden eines Risikos liefern anhand der Tabelle 4.1 dessen Bewertung für das Projekt. Zusätzlich werden die Eintrittswahrscheinlichkeiten und die Schäden jeweils mit den Ziffern 1-3 kategorisiert.

4.4.1.3. Risikoplanung

Bei der Risikoplanung ist das Ziel die erkannten und analysierten Risiken, so zu behandeln, dass die Eintrittswahrscheinlichkeit und auch die Schwere des Schadens beim Eintritt reduziert wird. Es mussten im Voraus Risikobehandlungsmaßnahmen[33] getroffen werden. Diese Maßnahmen sind:

- Risikovermeidung: Ein bestimmtes Projektrisiko beseitigen.
- Risikoverminderung: Die Eintrittswahrscheinlichkeit eines Risikos verringern.
- Risikobegrenzung: Den Schaden eines Risikos verringern.
- Risikoverlagerung: Projektrisiken auf Dritte übertragen.
- Risikoakzeptanz: Projektrisiko in Kauf nehmen und keine Maßnahmen ergreifen.

4.4.1.4. Risikoüberwachung

Zusätzlich zu den drei vorherigen Schritten wurde eine ständige Überwachung des Risikos durchgeführt und bei Veränderungen eine weitere Analyse- und Planungsphase angestoßen. So konnte immer schnell auf ein neues Risiko reagiert werden um entsprechende Gegenmaßnamen einzuleiten. Allerdings war es nur selten notwendig auf ein neues Risiko zu reagieren, da wir die meisten bereits bei den Überlegungen beachtet hatten. Andererseits gab es Risiken, die anfangs bedacht wurden, aber nie aufgetreten sind.

4.5. Bewertung der Maßnahmen

4.5.1. Projektmanagement

Prinzipiell war die Einteilung in kleinere Teilphasen ein gutes Vorgehen. Auf diese Art und Weise wurden einmal am Anfang der Planung die Abhängigkeiten zwischen den einzelnen Teilphasen bestimmt und so die AP‘s aufgeteilt. Befürchtungen, dass es Abhängigkeiten gibt ,die nicht betrachtet wurden und so später zu Problemen führen könnten, bewahrheiteten sich nicht. Letztlich konnten leider nicht alle Arbeitspakete abgearbeitet werden. Was von den oben genannten Arbeitspaketen realisiert wurde, kann aus Teil II *Abweichungen vom Pflichtenheft* entnommen werden.

4.5.2. Teammanagement

Während der gesamten Zeit des Projektes wurden grundsätzlich in jedem Treffen offene Diskussion geführt. Dadurch, dass schnell klar wurde, dass kein Mitglied der Projektgruppe ein Problem damit hat Probleme offen anzusprechen, konnte ein durchweg positives Arbeitsklima erhalten werden. Alternative Maßnahmen, wie z.B. eine anonyme Bewertung der anderen Mitglieder durch jeden einzelnen, waren durch die offene Kommunikation untereinander nicht notwendig und wurden daher in Hinblick auf den zeitlichen Aspekt nicht angewandt.

Auf das strukturierte Diskutieren mit der Bildung von zwei Gruppen wurde bei Meinungsverschiedenheiten verzichtet, da sich die Diskussionen immer in einem zeitlich akzeptablen Rahmen bewegten. Als einziger Punkt der Maßnahme zum strukturierten

Diskutieren aus dem Projektplan wurde der Mehrheitsentscheid angewandt, da es sich als eine gute Entscheidungshilfe nach einer offen geführten Diskussion herausgestellt hat. Durch die völlige Akzeptanz durch die Mitglieder wurden aufwendigere Verfahren zur Abstimmung und Diskussionsführung nicht angewandt.

Die Maßnahme zum Überblicken des Gesamtkontexts des Projekts wurde während der Treffen durch die Mitglieder praktiziert. So konnte die erste Phase des Projekts in den geplanten Arbeitspaketen abgearbeitet werden, auch wenn Änderungen der Arbeitspakete nicht verhindert werden konnten.

Auch in der zweiten Phase wurden Arbeitspakete gebildet. Zusätzlich wurden diese durch Redmine in Form eines Gantt-Diagramms visualisiert. Diese Visualisierung erwies sich als durchaus nützlich um den Gesamtkontext des Projektes abzuschätzen. Dennoch mussten wir feststellen, dass wir unsere zeitliche Planung immer wieder anpassen mussten.

Diese Anpassungen resultierten im Wesentlichen daraus, dass wir einerseits keinerlei Erfahrung hatten, um solche Abschätzungen realistisch zu erstellen und andererseits daraus, dass wir immer wieder auf Probleme gestoßen waren, die nicht vorhergesehen werden konnten. Zu diesem Problemen zählen z.B. dass mit Techniken gearbeitet wurde, die keiner der Mitglieder der Projektgruppe kannte bzw. mit denen in der Vergangenheit keiner gearbeitet hatte. Hinzu kamen diverse Verzögerungen, die nicht durch die Projektgruppe bedingt waren. So kam es häufig vor, dass Informationen durch die Betreuer erfragt wurden, jedoch leider die Antworten häufig nicht zufriedenstellend waren, da sie Beispielsweise zu ungenau waren. Dies führte wiederum zu häufigem Schriftwechsel und verschlimmerte so den Zeitverzug. Worin genau der Ursprung dieses Problem lag, konnte nicht ermittelt werden und soll auch nicht weiter Bestandteil dieses Endbericht sein.

Die Anwendung der Konflikttypen hat während der ersten Phase eine klare Vorgabe zum Einhalten von Regeln gegeben. Die Anwendung von Konsequenzen wurde allerdings nicht nötig, da sich die Vergehen, wie Verspätungen zu Treffen, in einem vertretbaren Rahmen bewegten. Auch innerhalb der zweiten Projektphase musste nie einer der Konflikttypen angewandt werden. Wenn Probleme wie z.B. Verspätungen auftraten löste sich das Problem automatisch dadurch, dass die Akzeptanz der betreffenden Person drastisch innerhalb der Projektgruppe sank. Dies führte wiederum zu mehr Anstrengung dieser Person. Somit haben sich kleinere Probleme selbstständig reguliert.

Die Unterstützungsplattform Redmine wurde in der ersten Phase sehr häufig genutzt, wobei sich vor allem das Wiki als geeignetes Werkzeug zum Zusammentragen von Ergebnissen herausgestellt hat. Dies zeigte sich beispielsweise in der Phase der Anforderungsanalyse. Dort wurden Ergebnisse von Reviews von Anwendungsfällen in das Wiki eingetragen, um die Qualitätssicherung gruppenintern zu verwalten. Außerdem wurde das Wiki dazu genutzt, um Ausfälle von Mitgliedern während Klausurphasen, Urlaub oder anderen Vorfällen terminlich festzuhalten. So hatte die Gruppe immer einen Überblick über den bevorstehenden Ausfall von Mitgliedern.

Die Maßnahme zur Strukturierung der regelmäßigen Treffen wurde vom Projektplan angewandt. Es hat sich dabei herausgestellt, dass durch die Strukturierung eine effiziente

Absprache zur weiteren Bearbeitung von Aufgaben ermöglicht wurde. Das Protokollieren von Treffen war ebenfalls sehr hilfreich, um in der Gruppe gemachte Beschlüsse zu dokumentieren und im Nachhinein nachzuschlagen.

Innerhalb der zweiten Phase stellte sich heraus, dass die Protokollierung durch das bisher angewandte Verfahren keinen großen Nutzen mehr hatte. Über die Zeit der Entwurfsphase wurden in jedem Treffen Protokolle erstellt, die festhielt, was beschlossen wurde, und wie welche Arbeiten verteilt wurden. Dies brachte in der Phase der Implementierung nur wenig, da sich jeder selbst Aufgaben aus einem Aufgabenpool nahm und diese bearbeitete. Somit wandelten sich die Ansprüche an die Protokolle, denn es mussten auch eine technische Dokumentation erfolgen. Aus diesem Grund wurde diverse andere Protokollierungsverfahren angewandt und die bisherigen Protokolle lediglich bei Treffen mit den Betreuern weitergeführt.

4.5.3. Qualtiätsmanagement

Im Verlauf des Projektes stellte sich heraus, dass es an einigen Stellen durchaus sinnvoll sein kann, ein gutes und gewissenhaftes Qualtiätsmanagement anzuwenden. Dagegen gab es allerdings auch Punkte an denen sich herausstellte, dass ein übermäßiges Qualtiätsmanagement störend und zeitraubend wirken kann. So wurde insbesondere bei den Dokumenten, die im Verlauf der Projektgruppe erstellt werden mussten, eine hohe Qualität durch das mehrmalige zurückschicken durch die Betreuer erzwungen. Dies wirkte sich sehr zeitraubend aus und zerstörte jegliche zeitliche Planung, die zuvor erstellt wurde. Es gibt sicherlich geteilte Meinungen darüber, ob ein von vornherein besseres Qualitätsmanagement einen zeitlichen Vorteil bewirkt hätte. Das genaue Gegenteil stellt ein gutes Qualitätsmanagement in der Implementierungsphase dar. Ausführliche Tests, sowie eine ausführliche Dokumentation kann sehr viel Aufwand einsparen. Die Erstellung der Tests war für die meisten Mitglieder der Projektgruppe eine eher störend wirkende Sache, aber letztlich hat es auch seine Vorteile gehabt. Dagegen wurden beispielsweise die Metriken zum Erhalt der Benutzerfreundlichkeit (z.B. durch Begrenzung der Anzahl der Buttons auf einer Seite) als eher nutzlos erachtet. Das wesentliche Problem bestand vor allem darin, dass sich z.B. eine sinnvolle Anzahl an Buttons nicht für alle Seiten gleich festsetzen ließ. Dadurch musste diese Metrik für nahezu jede Seite angepasst werden und verlor somit auch an Bedeutung und Nutzen.

4.5.4. Risikomanagement

Das Risikomanagement erwies sich als eine der schwierigsten Aufgaben, die in einem Projektalltag durchgeführt werden müssen. Dies lag vor allem daran, dass keiner von den Teilnehmern der Projektgruppe jemals damit Erfahrungen gemacht hat.

Die Risiken wurden zwar, soweit es geht, erkannt, die Analyse und die Planung der Behandlungsmaßnahmen waren aber eine ziemlich große Herausforderung. Im Folgenden sollen die einzelnen Risiken und ihre Behandlung vorgestellt werden.

4.5.4.1. Personalausfall

Auf der Grundlage von Anwesenheitslisten, konnte die Eintrittswahrscheinlichkeit innerhalb des Semesters auf etwa 20% geschätzt werden. Je nachdem welche Teilnehmer ausfallen, ist der Schaden vom Personalausfall als *ernst* einzustufen. Nach Tabelle 4.1 war damit der Personalausfall innerhalb des Semesters im grünen Bereich. Die Überwachung des Risikos ergab auch, dass die Ausfälle sich in Grenzen hielten bis die Prüfungsphase am Ende der vorlesungsfreien Zeit anfing. Hier häuften sich die Ausfälle, so dass die geschätzte Ausfallwahrscheinlichkeit auf etwa 50% angestiegen ist. Da der Schaden durch die Ausfälle bei *ernst* geblieben ist, stieg die Gefahr des Risikos in den gelben Bereich. Dazu wurde als Behandlungsmaßnahme die Risikoverminderung angewandt, indem einzelne Personen ihre Prüfungstermine und Urlaub, im Rahmen des Möglichen angepasst haben, so dass in der zweiten Hälfte der vorlesungsfreien Zeit die Ausfälle sich wieder unter 25% fingen. Auch während des zweiten Semesters lag die Eintrittswahrscheinlichkeit bei etwa 20%.

4.5.4.2. Konflikte zwischen den Mitgliedern

Bei Konflikten zwischen den Mitgliedern sehen wir uns im grünen Bereich. Da es bisher auch nur wenige Konflikte gab, ist die Eintrittswahrscheinlichkeit auf unter 10% geschätzt worden. Als Risikoverminderung werden gemeinsame Aktivitäten und Diskussionsrunden durchgeführt.

4.5.4.3. Sprachliche Barrieren

Da von den zehn Teilnehmern der Projektgruppe, sieben einen Migrationshintergrund aufweisen und fünf davon sprachliche Probleme haben, ist die Wahrscheinlichkeit von sprachlichen Schwierigkeiten in der Kommunikation und Erstellung von Dokumenten bei etwa 50%. Der Schaden, der dadurch entsteht ist bezüglich Tabelle 4.1 *katastrophal*, was auch direkte Auswirkungen auf den *Zeitverzug* hat. Deshalb wird dieses Risiko im roten Bereich eingeschätzt. Die einzige Möglichkeit, die wir sehen ist eine Risikoverlagerung durchzuführen, wobei sprachintensive Aufgaben, wie z.B. das Verfassen von Texten, von den übrigen Teilnehmern erledigt werden.

4.5.4.4. Zeitverzug

Auch bei diesem Risiko sehen wir uns im roten Bereich. Dies führt dazu, dass wichtige Dokumente meistens mit Wochen Verspätung abgegeben werden. Die Risikobegrenzung

war die Nutzung der vorlesungsfreien Zeit als Puffer und die Einführung von *Powertagen*, an denen die Gruppe sich ganztägig trifft. Damit wird Zeit eingespart, was daran liegt, dass Fragen in der Gruppe persönlich geklärt wurden und langwieriger E-Mail-Austausch vermieden wurde. Außerdem konnten Entscheidungen so viel schneller und effizienter getroffen werden.

5. Erfahrungen

Autor: VM

Ich habe während der Projektgruppe gelernt, dass die Konzeption und Entwicklung eines sicheren cloudbasierten Onlinebanking-Systems sehr komplex ist. Die Komplexität ergab sich aus den vielen Themengebieten, die bei der Entwicklung eines Onlinebanking-Systems detailliert zu berücksichtigen sind. Während der ersten Phase des Projekts, in der die Konzeption des Onlinebanking-Systems umgesetzt wurde, habe ich festgestellt, dass die Koordination der Projektteilnehmer entscheidend ist, um die geforderten Aufgaben termingerecht zu bearbeiten. Eine Maßnahme, die sich hierbei als sehr effektiv erwiesen hat, ist die Zusammenarbeit der Teilnehmer in räumlicher Nähe. Eine weitere, wichtige Maßnahme für die Koordination war das Wissensmanagement, welches jedoch aus mangelnder Erfahrung der Teilnehmer nicht reibungslos funktionierte. Während der zweiten Phase des Projekts, in der die Implementierung des Onlinebanking-Systems umgesetzt werden sollte, konnte ich viele Erfahrungen mit dem Einsatz der für das System notwendigen Techniken sammeln. In der Implementierung der Techniken kamen jedoch viele Probleme auf, da zu Beginn der Implementierung die Erfahrungen im Umgang mit den Techniken oft gering oder gar nicht vorhanden waren. Dies führte dazu, dass eine Einarbeitung in die Techniken notwendig wurde und dadurch zeitliche Verzögerungen bezüglich der einzuhaltenden Termine entstanden sind.

Autor: CW

Als Student mit ausländischer Herkunft durfte ich mich nicht nur damit auseinandersetzen, wie ein sicheres cloudbasiertes Onlinebanking-System auf technischer Ebene erstellt wird, sondern ich durfte auch neue Kommilitonen und neue internationale Kulturen kennenlernen. Das hat mir sehr viel Spaß gemacht. Ich werde nun meine Erfahrungen, die ich innerhalb der Projektgruppe gesammelt habe, kurz zusammenfassen. Durch die PG habe ich einen Eindruck davon gewonnen, wie ein kompletter Softwareentwicklungsvorgang abläuft und was alles dazugehört. Dabei konnte ich viele neue technische und organisatorische Kenntnisse gewinnen, wie z.B. die Bedeutung der Compliance für die Softwareentwicklung und wie Sicherheitsmaßnahmen in der Realität angewendet werden können. Auf technischer Ebene ist das zu erstellende System, aus meiner Sicht, nicht leicht zu implementieren gewesen, da die technologischen Anforderungen auf Software- und Hardware-Ebene komplex ist und alles voneinander abhängt. In der zweiten Phase habe ich vieles über unterschiedliche Frameworks gelernt und ich denke die Erfahrungen sind hilfreich für meine Zukunft. Im Bezug auf die Organisation, habe ich, als eher unerfahrene Person, gelernt, dass es sehr wichtig ist einen entsprechenden Arbeitsplan zu verfassen. Deshalb haben wir uns an “XP Programming“ orientiert, damit wir unsere

Arbeit möglichst flexibel halten zu können. Trotz unserer Bemühungen, konnten wir den zeitlichen Problemen nicht aus dem Weg gehen. Aber trotz Zeitverzug waren wir stets bemüht unseren Zeitplan einzuhalten. Dazu sind einige Maßnahmen wie z.B. Powertage oder das Ticketsystem eingeführt worden. Unser größtes Problem lag jedoch in der mangelnden Erfahrung, da sich unser Thema auf einem sehr umfangreichen Gebiet befindet. Die Auswahl der relevanten Informationen war daher schwer zu treffen und der daraus resultierende Aufwand ließ sich nicht abschätzen. Auch der Erfolg der herangezogenen Techniken war aufgrund der mangelnden Erfahrung unklar. Neben den Problemen, die aus unsere mangelnden Erfahrung resultierte, gab es zu einigen Techniken kaum oder keine Informationen, die beispielsweise bei Problemlösungen hätten helfen können. Dies führte zu weiteren Problemen beim Systementwurf, der Implementierung und der Erstellung der Compliance.

Autor: RH

Durch die Projektgruppe konnte ich meine Erfahrungen in vielen Bereichen weiter ausbauen. Darunter fallen z.B. Erfahrungen in der Entwicklung einer cloudbasierten Software inklusive der dazugehörigen Techniken. Die für mich wertvollste Erfahrung war jedoch ein Team zu leiten. Daraus ergaben sich Aufgaben, die für mich vollkommenes Neuland waren. Unter diese Aufgaben fiel nicht nur die Koordinierung der Gruppe sondern auch den Überblick über das gesamte Projekt zu behalten. Auch der Erhalt der Motivation innerhalb der Gruppe und das ergreifen von Mitteln gegen Risiken gehörte klar zu meinen Aufgaben. All diese Erfahrungen sind sicherlich nützlich im weiteren Verlauf meines Berufslebens. Daher kann ich durchaus sagen, dass diese Projektgruppe mir, auch wenn nicht alles immer nach Plan lief, neue Erkenntnisse beschert hat.

Autor: TS

Während der Arbeit an dem umfangreichen Projekt, ein cloudbasiertes Internetbankinsystem zu planen und zu implementieren, habe ich für einiges zur langfristigen Arbeit innerhalb einer Gruppe feststellen können. Besonders auffällig war in diesem Projekt die Planung, da hiermit noch niemand Erfahrungen gesammelt hatte und sie die Hälfte des Aufwandes für das Projekt ausmachte. Es kam immer wieder die Situation auf, dass etwas modelliert werden musste, von dem zu dem Zeitpunkt niemand so wirklich wusste was es war, so wurde z.B. die Struktur der Konto- und Transaktions-Klassen stark verändert, nachdem bekannt wurde, wie die Transaktionen übermittelt werden müssen. Oder dass etwas ausgewählt werden musste, das trotz hohem Zeitaufwand nicht ausreichend gestestet werden konnte. Einige der Tools, die allen unbekannt waren stellten sich als nicht oder zumindest nicht für Arbeit im Team brauchbar heraus und mussten durch andere Tools oder geschickte Organisation ersetzt werden. Insbesondere das Einarbeiten in die Gesetzestexte und die Modellierung der Gesetze war sehr aufwendig, da wir zur Extraktion wichtiger Passagen keine Hilfsmittel oder für uns verständliche Vorgehensweisen finden konnten und für deren Darstellung keine für die Bearbeitung in einer Gruppe geeignete Software vorhanden war.

Eine zweite große Problematik innerhalb der Planung bestand aus der Kommunikation

und Aufgabenverteilung, die sich leicht bewältigen lassen, wenn alle im gleichen Raum sind. Doch gerade in der Vorlesungszeit war diese Zeit recht knapp und es hat sich bewährt Besprechungsinhalte schriftlich festzuhalten (z.B. Protokolle oder Wiki). Allerdings bedurfte auch dies einer Gewöhnungsphase. Und wir mussten feststellen, dass eine grobe Schätzung für eine Aufgabe so gut wie unmöglich ist, wenn die Einarbeitung noch stattfinden muss.

In der Implementierungsphase, die die zweite Hälfte des Projekts ausmachte, habe ich viel über das Programmieren an sich und den Einsatz von Frameworks gelernt. Das Lernen und Einarbeiten kostete viel Zeit, da zu den meisten Anforderungen und Frameworks wenig oder kein Vorwissen vorhanden war.

Autor: JK

Die Erfahrungen, die ich in diesem Projekt gesammelt habe, sind sehr unterschiedlich. In meinen Augen habe ich sehr viele neue Technologien kennengelernt und damit Erfahrungen gesammelt, was ich als sehr wertvoll für die praktische Tätigkeit im späteren Job empfinde. Vor allem die Arbeit mit der Cloud und dem Application Server im J2EE hat mir sehr viel Spass gemacht. Dennoch mussten wir sehr viele Rückschläge hinnehmen, da wir dabei auf uns allein gestellt waren. So kam es, dass man erst nach mehrtägigen Recherche ein Fehler beheben konnte.

Als Problem hat sich das verteilte Arbeiten herausgestellt. Das und die Komplexität des Systems stellte sich als eine große Hürde im Verlauf der Projektgruppe. Die räumliche Nähe war deshalb sehr wichtig, um Lösungen schnell ausdiskutieren zu können und Fragen in der Gruppe zu klären. Für mich hat sich herausgestellt, dass für verteiltes Arbeiten man ein sehr großes Vertrauen in die Fähigkeiten und Zuverlässigkeit der Teilnehmer benötigt, was bei einer wild zusammengewürfelten Gruppe nicht möglich ist.

Nach den Erfahrungen in der Projektgruppe bin ich zum Entschluss gekommen, dass eine lockerer und respektvoller Umgangston sehr wichtig für die Produktivität ist. Zusätzlich spielt die Erfahrung eine wichtige Rolle, denn theoretisches Wissen reicht oft nicht aus, da die Praxis oft ganz anders aussieht.

Autor: VS

In der PG habe ich viel über die Projektplanung und das Zeitmanagement gelernt. Circa die Hälfte der gesamten Projektzeit wurde für die Planung und Vorbereitung aufgewendet, was ich zuerst für übertrieben hielt. Im nachhinein denke ich aber, dass eine Planungsphase sehr wichtig ist, vor allem weil bei dem Projekt viele Bereiche berücksichtigt werden mussten, in denen alle PG-Teilnehmer noch keine Erfahrungen hatten, wie z.B. Compliance und Gesetzte in Bezug auf Cloud-Systeme und Online-Banking. Es fand zwar am Anfang des Projektes eine Seminarphase statt, in der solche Themen vorgestellt wurden, aber hier wurde nur ein Überblick gegeben und nicht zu sehr ins Detail gegangen. Deswegen waren lange Einarbeitungen in neue Gebiete während des Projektes immer wieder nötig, durch die ein Zeitverzug entstanden ist. Auch bei der Bearbeitung von Aufgaben sind Probleme entstanden, da häufig Fragen aufgetaucht sind, die trotz

der langen Planungsphase nicht berücksichtigt worden sind und die erst in der Gruppe geklärkt werden mussten. Aus diese Grund haben wir uns dazu entschlossen uns regelmäßig zu treffen und in räumlicher Nähe zu arbeiten, so dass eine direkte Absprache möglich ist. Dies hat den Arbeitsablauf merklich beschleunigt.

Autor: YD

Mein Erfahrungsbericht besteht aus zwei Teilen, dem Wintersemester (1. Semester der PG) und dem Sommersemester (2. Semester der PG). Im ersten Semester der PG habe ich meine ersten Eindrücke über ein richtiges Projekt erfahren dürften. Wie sich herausstellte war dieses Projekt viel komplexer, als das, welches man aus dem Softwarepraktikum kennt. Aus den Bereichen Entwurf, Compliance, Projektbeschreibung, Anforderungen-Analyse, Projekt-Management besonders auch Teamarbeit usw. konnte ich viel lernen und fachliches Wissen zugewinnen. Dies wird mir später sicher in realen Projekten im Berufsleben weiterhelfen.

In dem zweiten PG-Semester hatte unsere Arbeit meistens mit diversen Technologien zu tun. In diesem Semester habe ich gelernt, wie ein cloudbasiertes Projekt eingerichtet wird und in wie weit die Aspekte Sicherheit, Compliance, Konstruktion usw. berücksichtigt werden müssen, damit es zu keinen Problemen kommt. Besonders viele Erfahrungen habe ich mit dem Logsystem und dem Oberflächentest gesammelt. Besonders problematisch war die Implementierung des Logsystem unter der Verwendung von Techniken wie JSF, Richfaces und einer Cloud-Umgebung. Das Problem war, dass es kaum Literatur oder Informationen gab, die uns bei diesem Problem hätten helfen können. Letztlich wurde das Problem damit gelöst, dass Keywords verwendet wurden. Dies unterscheidet sich zu den Techniken Java SD und JSP. Außerdem haben wir die aktullen free-Tools für Oberflächentests gestestet und das beste davon ausgewählt. Auch dabei gab es immer wieder Probleme zwischen dem Test-Tool und der Sicherheitskofiguration unseres Projektes. Dieses Problem haben wir jedoch später behoben.

Als wichtigsten Punkt sehe ich die Teamarbeit. Ich habe viel Unterstützung durch unsere Betreuer und Kommilitonen bekommen. So habe ich rechtzeitige Antworten von Kommilitonen und Betreuer bekommen und die Kommunikation war stets positiv.

Autor: MW

Im Sommersemester 2011 und Wintersemester 2011/2012 habe ich mit Projektgruppe beschäftigt. Die Konzeption und Entwicklung eines Cloud basierten Onlinebanking Systems hat mich nicht leicht aufgefallen. Es würde in zwei Phasen gearbeitet. Während der Konzeption habe ich gelernt, dass die Koordination und Planung bei eine Gruppe, der so gross ist, sehr wichtig ist. Am Anfang der Semester haben wir uns zwei mal die Woche in der Vorlesung zeit getroffen, das hat aber nicht gereicht, und haben wir in der Vorlesung freie zeit Powertage geführt. Powertage haben unsere problem mit Koordination und Planug sehr gut geholfen, und in der zeit könnten wir mit Betreuern auch häufig kontaktieren. Ich habe gelernt, dass bei jeder Phase Kommunikation mit Betreuern sehr wichtig ist, und bevor man was tut, sollte man auch die Betreuern fragen, damit jeder die gleiche Meinung hat, und spart man auch doppelte Arbeit. Es kann aber auch zur

noch mehr zeitliche Verzögerung führen. In der Entwicklungsphase habe wir sehr viel mit neue Technologien zu tun gehabt. Es hat viel beigebracht aber hat auch zur Verzögerung geführt. Ohne irgendwelche Erfahrung mit jeweilige Technologien eine System wie dieses zu Implementieren ist sehr aufwädig gewesen, aber auf der Stand, wo wir jetzt bezüglich Implementierung als Gruppe sind, ist meiner Meinung nach sehr gut.

Autor: AV

Im Rahmen der Projektgruppe 555 hatte ich die Möglichkeit, viel über Teamarbeit und Projektplanung zu lernen. Bereits am Anfang habe ich festgestellt, dass ein gutes Zeitmanagement sehr wichtig ist, damit ein Projekt rechtzeitig und erfolgreich abgeschlossen werden kann. Das Problem des Zeitmanagements stellte innerhalb unserer Projektgruppe mit das größte Problem dar. Sowohl während de der ersten, als auch während der zweiten Phase des Projekts hatte wir Probleme mit der Zeitplanung. Ein Grund dafür waren vor allem die fehlenden technischen Erfahrungen und die mangelnde Koordination des Projektes, was mit vielen Recherche-Arbeiten und Einarbeitung verbunden war.

Die Recherchen sowie den Einarbeitungsaufwand sehe ich persönlich dennoch nicht als Nachteil, da auf diese Weise ich die Möglichkeit hatte, neue Technologien kennenzulernen, praktische Erfahrung zu sammeln und zu lernen, wie eine Projektgruppe funktioniert, wo Risiken auftreten können und was dagegen unternommen werden kann.

Autor: DS

Arbeiten in einer Gruppe ist meistens eine sehr große Herausforderung, da man mit vielen, in der Regel, unterschiedlichen Personentypen mit unterschiedlichen Erfahrungen, zusammen ein Projekt erfolgreich abschließen möchte. Während dieser Projektgruppe (PG) war dies einer der Hürden, die uns ab und an ins Schwierigkeiten gebracht hatte. Aber nichtsdestotrotz war dieses vergangene Jahr eine schöne und reichliche Erfahrung und Vorbereitung fürs Berufsleben.

Während dieser PG, hatte man, neben der obebgenannten nicht technischen Erfahrung, auch mit reichlich vielen technischen Neuigkeiten zu tun gehabt, die man sicherlich mit ins Berufsleben mitnehmen könnte. Puncto Sicherheit in einer Cloud z.B. und das aufsetzen dieser oder aber auch die Vielzahl an Gesetzen, die man sich in der Planungsphase aneignen musste.
Diese erste Phase, die ein wesentlicher Teil unserer für die PG verbrauchte Zeit ausmachte, wurde meiner Meinung nach, sehr ausführlich seitens meiner Kommilitonen verfasst und erklärt. Ich kann mich nur ihrer Meinung anschließen.

Während der Implementierungsphase könnte man zwar die Fehler, die man in der ersten Phase, aufgrund mangelnder Erfahrung, gemacht hatte, vermeiden, aber gegen die Zeit konnten wir leider nicht kämpfen und deshalb waren nicht in der Lage manche Sachen so umzusetzen, wie sie Anfangs geplant waren.

Schließlich waren die Organisation, die Koordination und die Kommunikation und die Schwierigkeit mit zehn Personen sie zu beherrschen und die Ruhe dabei zu bewahren, die drei weitere wesentliche Sachen, die diese PG ausgemacht haben.

Teil IV.

Anhang

A. Anwender Dokumentation

Autor: CW,DY

Korrektor: TS,DS

A.1. Einleitung

In den folgenden Kapiteln wird für die einzelnen Rollen beschrieben, welche Vorgänge das System unterstützt und wie der Nutzer mit unserem System arbeiten soll.

A.2. Vorausetzungen zur Nutzung

Die Online-Banking-Software ist im Internet unter der folgenden Adresse verfügbar: https://129.217.47.73:8181/OnlineBankingNeu/faces/login.xhtml. Um das System zu nutzen, benötigen Sie einen Internet-Browser, der JAVA, JavaScript und SSL unterstützt. Im Prinzip sollen alle modernen Browser, z.B. Microsoft Internet Explorer oder Mozilla Firefox, die genannten Anforderungen erfüllen. Um die Sicherheit zu erhalten, ist es notwendig den Browser durch regelmäßige Updates immer auf dem aktuellen Stand zu halten.

A.3. Benutzeroberfläche

Die Benutzeroberfläche, die man nach dem Login erreicht, ist in drei Teile unterteilt:

1. Rollenbereich: der rechte Teil der Seite zeigt die aktuell ausgewählte Rolle und bietet die Möglichkeit die Rolle zu wechseln. Ausserdem befindet sich der Logout-Link im Rollenbereich.
2. Navigationsbereich: die linke Teil der Seite beinhaltet ein Menü, mit dessen Hilfe man die unterschiedlichen Funktionen und Ansichten des Online-Banking-Systems erreichen kann. Das Menü im Navigationsbereich ist abhängig von der ausgewählten Rolle. Innerhalb des Navigationsbereichs sind die Funktionen und Ansichten nach Themen gruppiert.
3. Hauptbereich: der mittlere Teil der Seite zeigt wichtige Informationen und Funktionen an. Die Anzeige hängt von der ausgewählten Option im Navigationsbereich

ab.

A.4. Gemeinsame Funktionen

In diesem Abschnitt werden die Funktionen des Systems aufgeführt, die alle Nutzer ausführen können.

A.4.1. Ein- und Ausloggen

Nachdem Sie die Adresse des Online-Banking-Systems aufgerufen haben, kommen Sie auf die Login-Seite (s. Abb. A.2). Geben Sie Ihr Password und Ihren Benutzernamen in die dafür vorgesehenen Felder ein und bestätigen Sie Ihre Eingabe mit dem Login-Button.

Durch das Einloggen gelangen Sie in den gesicherten Bereich unseres Internet-Bankings und können Ihre Bankgeschäfte starten. Vergessen Sie nach der Nutzung des Online-Banking-Systems nicht, sich auszuloggen, damit Ihr Account nicht von Dritten unberechtigt genutzt wird. Das Ausloggen erfolgt durch Klick auf *Logout* in Abb. A.1, dann lenke die Webseite wieder zur Einloggenseite. In den folgenden Abschnitten werden die verschiedenen Rollen und ihre jeweiligen Funktionen vorgestellt.

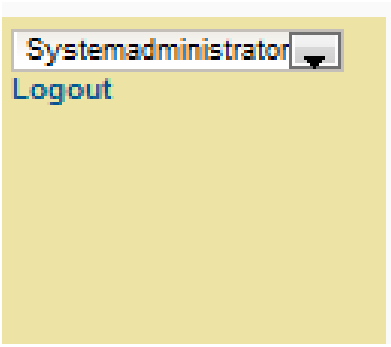


Abbildung A.1.: Ausloggen

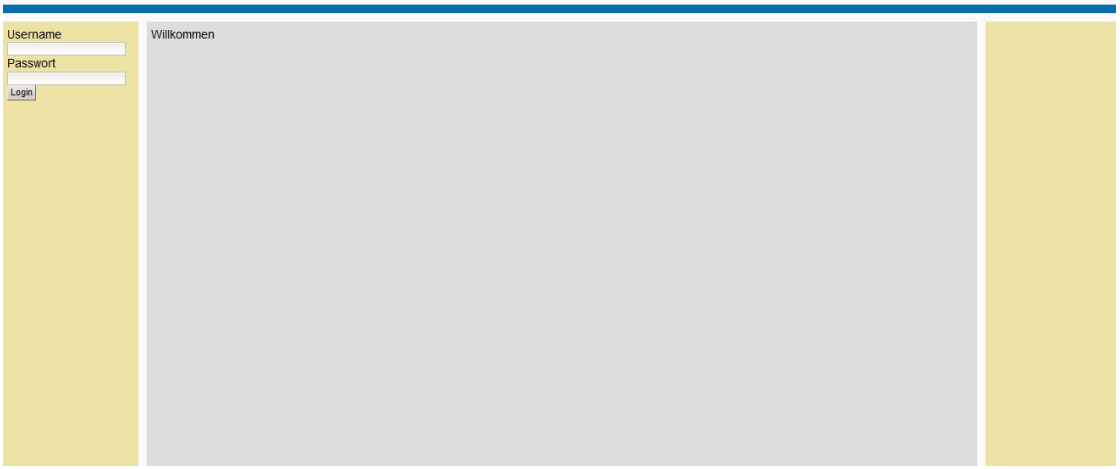


Abbildung A.2.: Einloggen

A.4.2. Rollen-Auswahl

Unser System ist rollenbasiert, das heißt, dass es mehrere Rollen gibt, die jeweils bestimmte Funktionalitäten nutzen können.

Nachdem Sie sich eingeloggt haben, müssen Sie zuerst eine Rolle auswählen, um die ihr zugeordneten Funktionen ausführen zu können. Insgesamt sind zehn verschiedene Rollen wie folgt klassifiziert [32].

1. Kunde: Kunde der Bank
2. Kundenberater: Mitarbeiter der Bank, die als Kundenbetreuer in einer Filiale arbeiten
3. Controlling-Mitarbeiter: Mitarbeiter der Bank, die in der Controllingabteilung arbeiten
4. Eigenhändler: Mitarbeiter der Bank, die in der Eigenhandelsabteilung arbeiten
5. Geschäftsführer: Der Geschätsführer der Bank.
6. Jurist: Mitarbeiter der Bank, die in der Rechtsabteilung arbeiten.
7. Kassierer: Mitarbeiter der Bank, die Dienste der Kasse für Kunden anbieten.
8. Marketing-Mitarbeiter: Mitarbeiter der Bank, die in der Marketingabteilung arbeiten.
9. Produktentwickler: Mitarbeiter der Bank, die neue Produkte erstellen.
10. Systemadministrator: Mitarbeiter der Bank, der als Systemadministrator arbeitet.

Das Rolle-Wechseln erfolgt durch Klick auf das Rollenmenü in der oberen rechten Ecke der Seite in Abb. A.3.

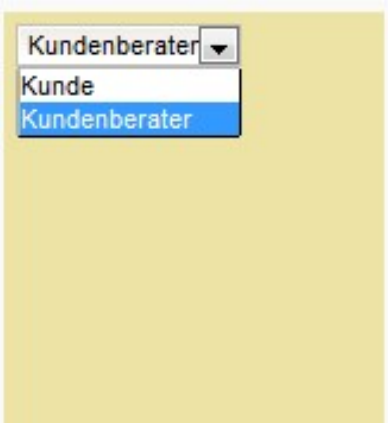


Abbildung A.3.: Rolle wechseln

A.4.3. Nachrichtenverkehr

Das Banking-System bietet die Möglichkeit zwischen Mitarbeitern oder zwischen Mitarbeiter und Kunde Nachrichten zu verschicken.

Klicken Sie den Button *Nachrichten*, werden wie in Abb. A.4 zwei Optionen *Nachrichten lesen*, *Nachrichten verschicken* angezeigt. Wählen Sie *Nachricht lesen*, können Sie empfangene Nachrichten anzeigen lassen und auf die angezeigte Nachricht direkt antworten. Wählen Sie *Nachricht verschicken* können Sie Nachrichten verfassen und versenden. Hierbei können Kunden wie in Abb. A.5 nur Nachrichten an ihre Kundenberater versenden, Mitarbeiter können wie in Abb. A.6 andere Mitarbeiter oder Kunden auswählen, an die sie ihre Nachricht verschicken.

Abbildung A.4.: Nachrichten im Navigationsbereich und Nachricht lesen

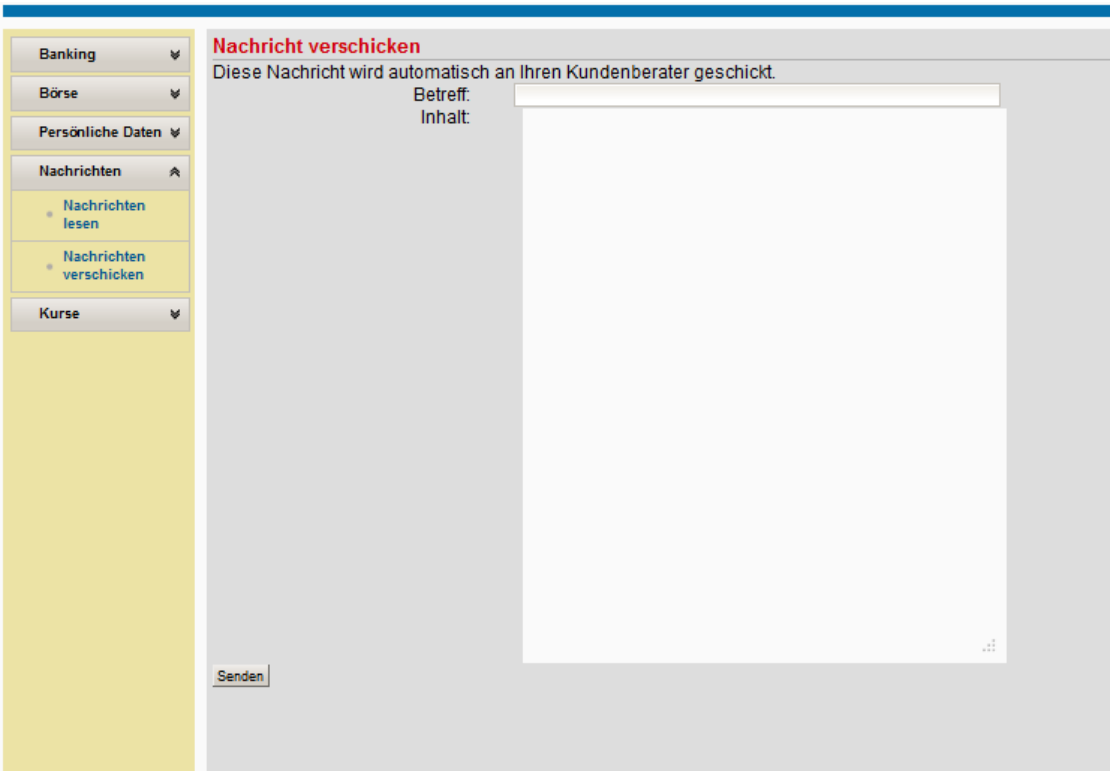


Abbildung A.5.: Nachricht verschicken für den Kunden

A.5. Kunde

Autor: CW

Als Kunde sehen Sie nach dem Einloggen Ihre Benutzeroberfläche wie in Abb. A.7. Der Navigationsbereich zeigt die folgenden Einträge:

1. Online-Banking
    - Überweisung
    - Lastschift
    - Dauerauftrag
    - Kontostand einsehen
    - Buchungen einsehen
2. Börseangelegenheiten
    - Finanzinstrument handeln
    - Depot einsehen

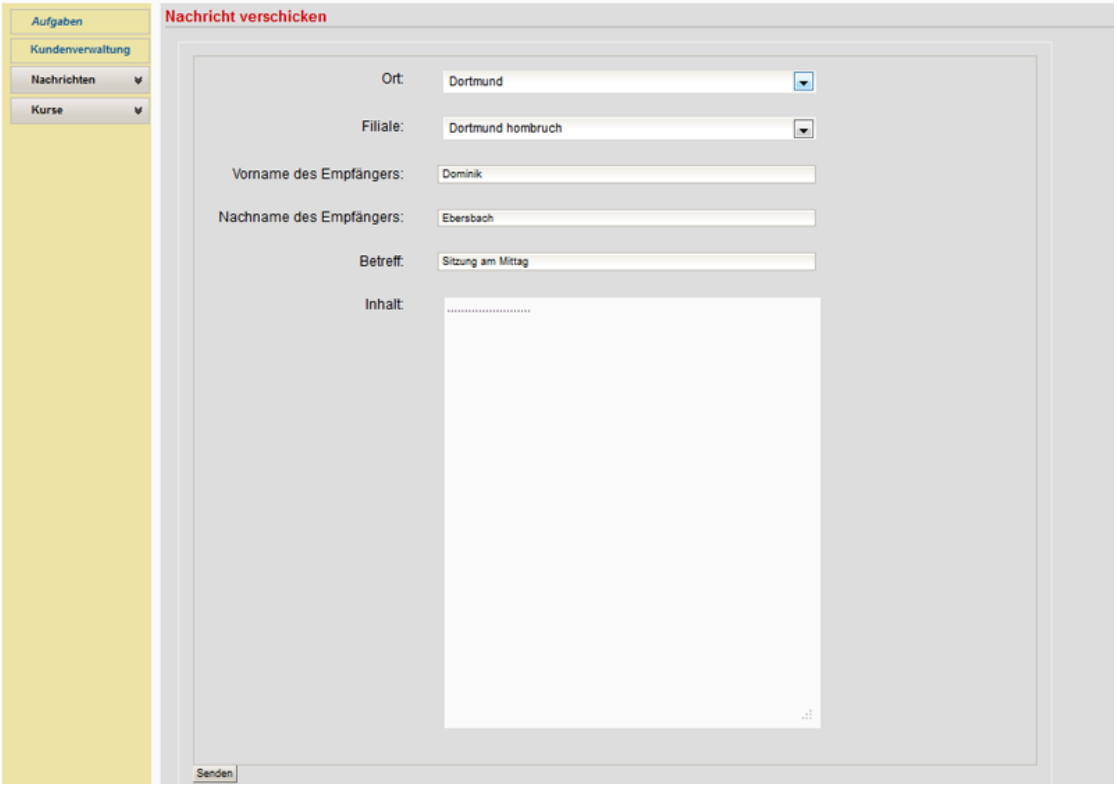


Abbildung A.6.: Nachrichten verschicken für Mitarbeiter

Abbildung A.7.: Kunden-Portal

- Orderbuch

3. Persönliche Daten
4. Nachrichten
   - Nachrichten lesen
   - Nachrichten verschicken
5. Kurse
   - Kurse einsehen

Die Unterpunkte in der Aufzählung sind die Unterpunkte der entsprechenden Einträge im Navigationsbereich.

Im folgenden werden die Funktionen an die Navigation angelehnt beschrieben.

A.5.1. Banking

In den folgenden Unterabschnitten werden alle Funktionen beschrieben, die unter *Banking* eingeordnet sind.

A.5.1.1. Überweisung

Über den Menüpunkt *Überweisung* gelangen Sie zur Ansicht *Überweisung tätigen* (s. Abb. A.8).

Die Felder dieser Ansicht haben die folgenden Funktionen:

1. Empfängername: Hier tragen Sie den Namen des Empfängers ein.
2. Empfänger speichern: Wenn Sie hier das Häkchen setzen, wird der Empfänger für Ihren Account gespeichert.
3. Zielkonto/IBAN: Hier tragen Sie die IBAN des Zielkontos ein oder die Kontonummer für Inlandsüberweisungen.
4. BLZ/SWIFT: Hier tragen Sie die SWIFT zur IBAN oder die BLZ zur Kontonummer ein.
5. Verwendungszweck: Hier tragen Sie den Verwendungszweck ein.
6. Betrag: Hier tragen Sie den Betrag in EUR ein.
7. Zu belastendes Konto: Hier wählen Sie das Konto, von dem aus Sie überweisen wollen, aus.

A.5.1.2. Lastschrift

Über den Menüpunkt *Lastschrift* gelangen Sie zur Ansicht *Lastschrift tätigen* (s. Abb. A.9).

Die Felder dieser Ansicht haben folgende Funktionen:

1. Zahlungspflichtiger: Name des Zahlungspflichtigen
2. Zahlungspflichtigen speichern: Wenn Sie hier das Häkchen setzen, wird der Zahlungspflichtige für Ihren Account gespeichert.
3. Zielkonto/IBAN: Hier tragen Sie die IBAN des Zielkontos ein oder die Kontonummer für Inlandsüberweisungen.

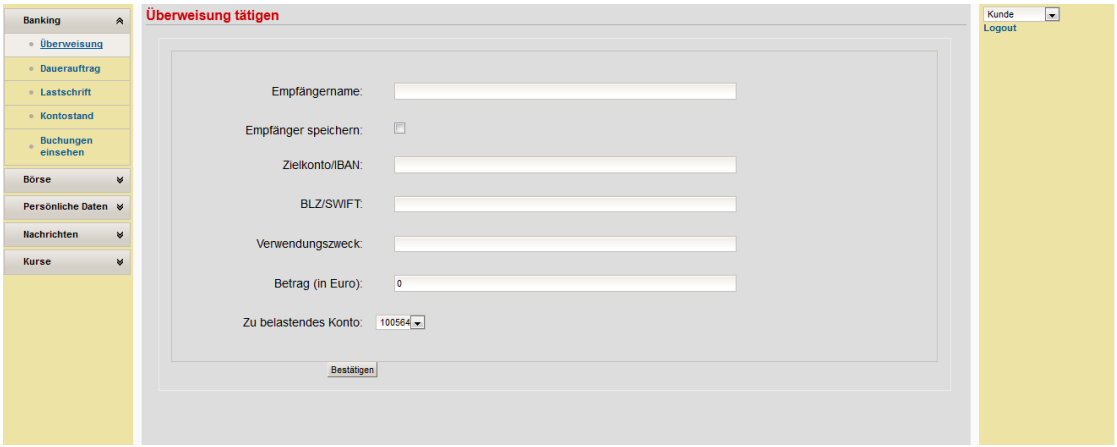


Abbildung A.8.: Überweisung

4. BLZ/SWIFT: Hier tragen Sie die SWIFT zur IBAN oder die BLZ zur Kontonummer ein.
5. Verwendungszweck: Hier tragen Sie den Verwendungszweck ein.
6. Betrag: Hier tragen Sie den Betrag in EUR ein.
7. Zu entlastendes Konto: Hier wählen Sie das Konto, auf dem der Betrag gutgeschrieben werden soll.

Abbildung A.9.: Lastschrift

A.5.1.3. Dauerauftrag

Über den Menüpunkt *Dauerauftrag* gelangen Sie zur Ansicht *Dauerauftrag* (s. Abb. A.10).

Die Felder dieser Ansicht haben die folgenden Funktionen:

1. Empfängername: Hier tragen Sie den Namen des Empfängers ein.
2. Empfänger speichern: Wenn Sie hier das Häkchen setzen, wird der Empfänger für Ihren Account gespeichert.
3. Zielkonto/IBAN: Hier tragen Sie die IBAN des Zielkontos ein oder die Kontonummer für Inlandsüberweisungen.
4. BLZ/SWIFT: Hier tragen Sie die SWIFT zur IBAN oder die BLZ zur Kontonummer ein.
5. Verwendungszweck: Hier tragen Sie den Verwendungszweck ein.
6. Betrag: Hier tragen Sie den Betrag in EUR ein.
7. Zu belastendes Konto: Hier wählen Sie das Konto, von dem aus Sie überweisen wollen, aus.
8. Gültig bis: hier können Sie ein Enddatum auswählen. Ist ein Enddatum gesetzt, werden nach dem Überschreiten des Enddatums keine Überweisungen mehr von diesem Dauerauftrag erstellt. Wenn Sie kein Enddatum auswählen, müss der Dauerauftrag bei Bedarf von Ihrem Kundenberater abgebrochen werden.
9. Startzeitpunkt: Der Startzeitpunkt ist das erste Datum an dem eine Buchung des Dauerauftrags erfolgt.
10. Ausführungsintervall: Das Ausführungsintervall legt fest in welchen Abständen der Dauerauftrag gebucht wird.

A.5.1.4. Kontostand

Über den Menüpunkt *Kontostand* gelangen Sie zur Ansicht *Kontostand einsehen* (s. Abb. A.11).

Ihnen werden im oberen Teil der Ansicht Ihre Konten in tabellarischer Form angezeigt. In der Tabelle gibt es die Spalten *Kontonummer*, *Eröffnungsdatum*, *Gesperrt* und *Kontostand*. In der Spalte *Gesperrt* ist ein Wahrheitswert eingetragen (*true* für wahr, das Konto ist gesperrt oder *false* für falsch, das Konto ist nicht gesperrt).

Indem Sie in der Tabelle ein Konto auswählen und danach auf den Button *Kontostand anzeigen* klicken, wird Ihnen der Kontostand des ausgewählten Kontos detailliert angezeigt.

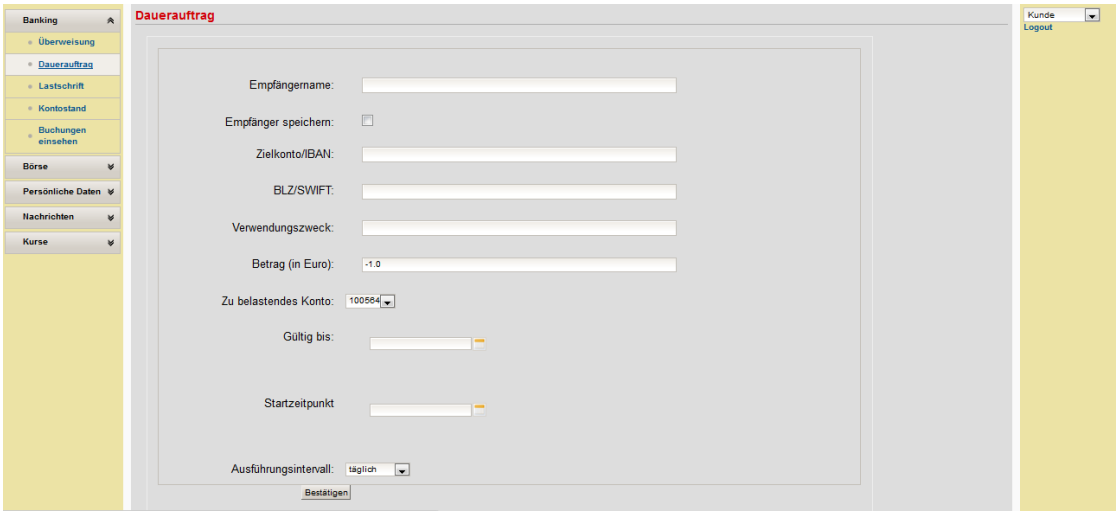


Abbildung A.10.: Daueauftrag

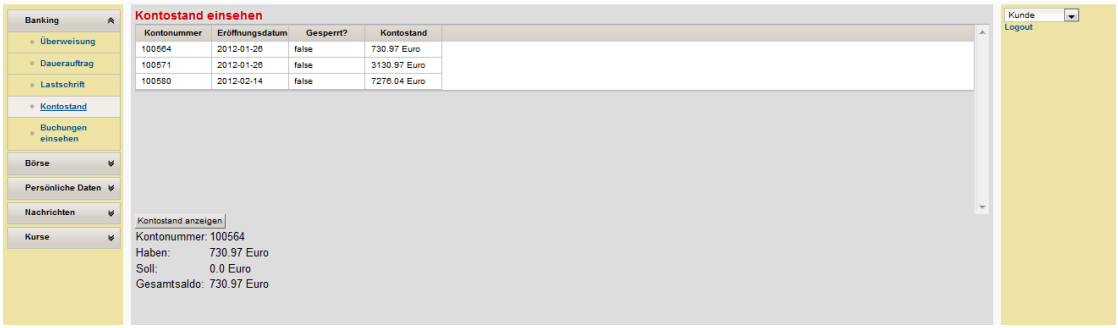


Abbildung A.11.: Kontostand informieren

A.5.1.5. Buchungen einsehen

Über den Menüpunkt *Buchungen einsehen* gelangen Sie zur Ansicht *Buchungen* (s. Abb. A.12).

Ihnen werden im oberen Teil der Ansicht Ihre Konten in tabellarischer Form angezeigt. In der Tabelle gibt es die Spalten *Kontoinhaber*, *Kontonummer* und *Kontostand*.

Indem Sie in der Tabelle ein Konto auswählen und danach auf den Button *Konto ausähnlen* klicken, werden in einer Tabelle im unteren Teil der Ansicht die zum ausgewählten Konto gehörigen Buchungen angezeigt.

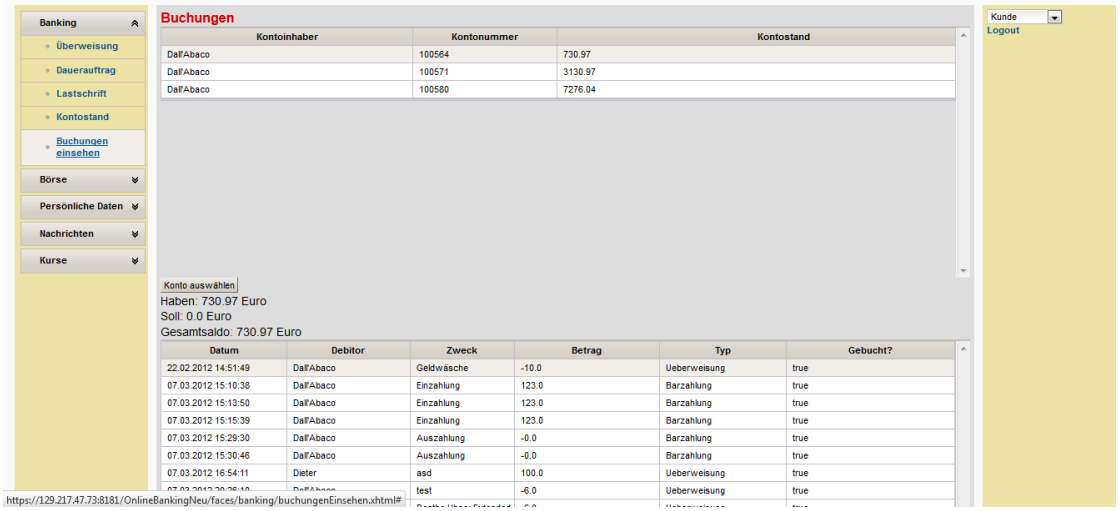


Abbildung A.12.: Buchungen informieren

A.5.2. Börse

A.5.2.1. Finanzinstrument handeln

Über den Menüpunkt *Finanzinstrument handeln* gelangen Sie zur Ansicht *Finanzinstrument handeln* (s. Abb. A.13).

Die Felder dieser Ansicht haben die folgenden Funktionen:

1. Konto: Hier wählen Sie das Konto aus, auf dem der Handel verbucht werden soll.
2. WKN/ISIN: .
3. Name: Hier wird der Name des Finanzinstruments angezeigt.
4. Kurswert: Hier wird der Kurswert des Finanzinstruments angezeigt.
5. Anzahl: Hier tragen Sie Anzahl der Einheiten ein, die Sie handeln möchten.
6. Orderzusatz: .
7. Limit: .
8. Gültig bis: Hier können Sie bestimmen, bis wann die Order bestehen soll.
9. Transaktionsart: Hier wählen Sie die Transaktionsart, also Verkauf oder Kauf.

Mit Betätigen des *Bestätigen* Buttons wird das Angebot an die Börse weitergeleitet. Um eine bestätigte Order zu stornieren lesen Sie im Abschnitt A.5.2.3 nach.

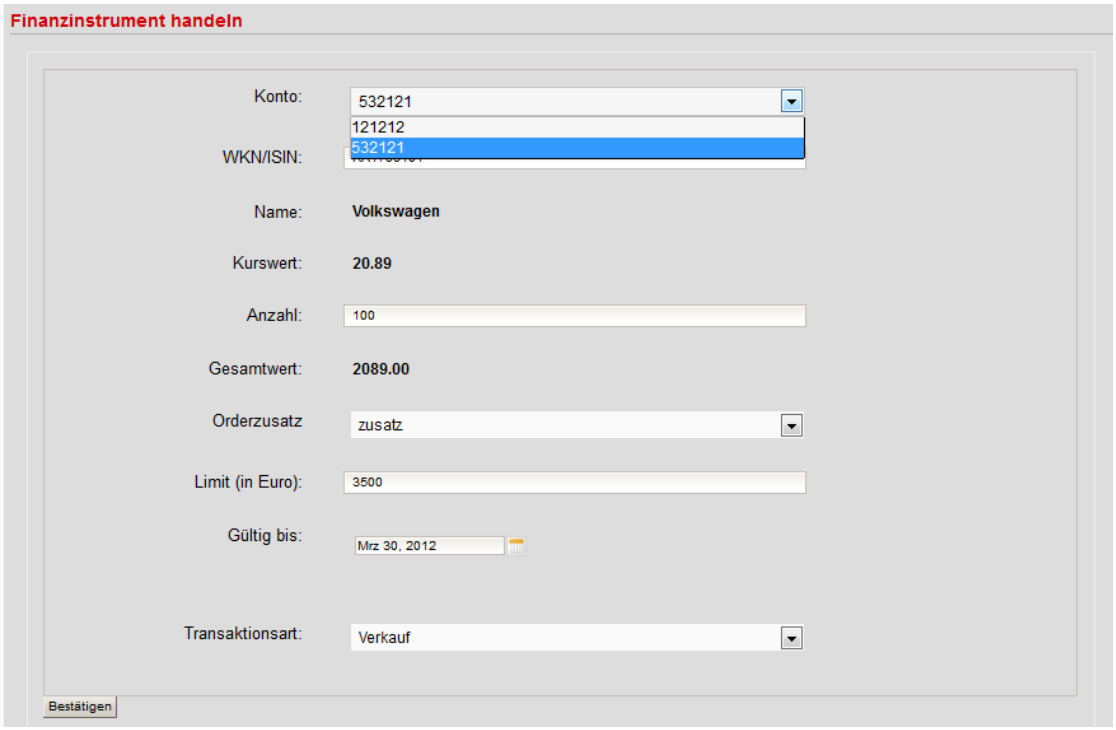


Abbildung A.13.: Finanzinstrument handeln

A.5.2.2. Depotinformationen einsehen

Über den Menüpunkt *Depotinformationen einsehen* gelangen Sie zur Ansicht *Depotinformationen einsehen* (s. Abb. A.14).

Im oberen Teil der Ansicht werden in einer Tabelle Ihre Finanzinstrumente aufgelistet. Durch das Auswählen eines Finanzinstruments und anschließender Betätigung des *Handeln* Buttons gelangen Sie zur Ansicht *Finanzinstrument handeln* (s. Abschnitt A.5.2.1).

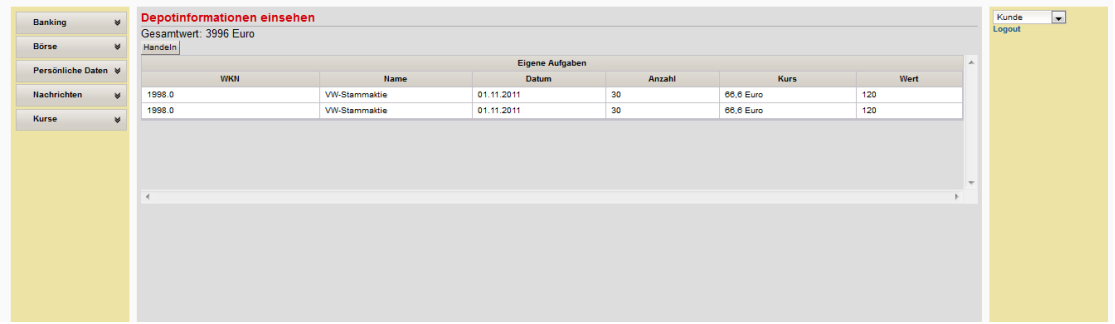


Abbildung A.14.: Depot einsehen

A.5.2.3. Orderbuch

Über den Menüpunkt *Orderbuch* gelangen Sie zur Ansicht *Orderbuch* (s. Abb. A.15).

Im oberen Teil der Ansicht werden Ihre aktuellen, offenen Order in einer Tabelle angezeigt.

Durch das Auswählen einer Order und anschließendes Betätigen des *Stornieren* Buttons können Sie eine Order stornieren.

Abbildung A.15.: Orderverwaltung

A.5.3. Persönliche Daten ansehen

Über den Menüpunkt *Persönliche Daten ansehen* gelangen Sie zur Ansicht *Kontaktdaten anzeigen* (s. Abb. A.16).

In dieser Ansicht werden Ihre Kundennummer, Ihr Vorname, Nachname, Geburtsdatum, Geburtsort und Ihre Adresse mit Straße, Hausnummer, Postleitzahl und Ort angezeigt.

Mit Betätigung des *Ändern* Buttons erhalten Sie in der Ansicht *Kontaktdaten ändern* die Möglichkeit ihre Adresse zu ändern (s. Abschnitt A.5.3.1).

A.5.3.1. Kontaktdaten ändern

In der Ansicht *Kontaktdaten ändern* befinden sich die Felder *Vorname*, *Nachname*, *Strasse*, *Hausnummer*, *PLZ* und *Ort* (s. Abb. A.17). Indem Sie die Einträge der Felder bearbeiten, ändern Sie die Kontaktdaten. Durch Betätigen des Buttons *Bestätigen* werden die Änderungen übernommen und gespeichert.

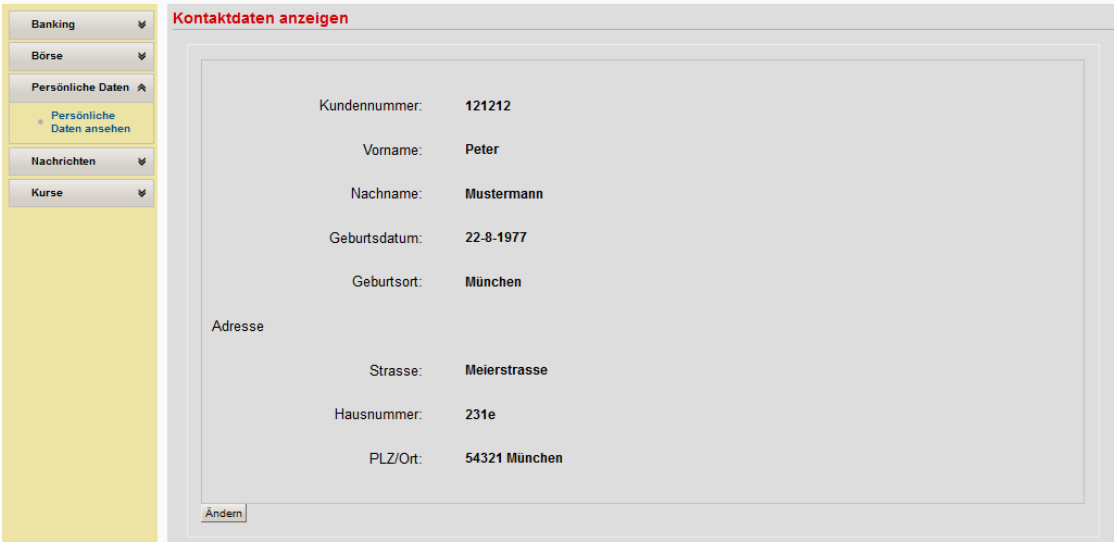


Abbildung A.16.: Anzeige von persönlichen Daten

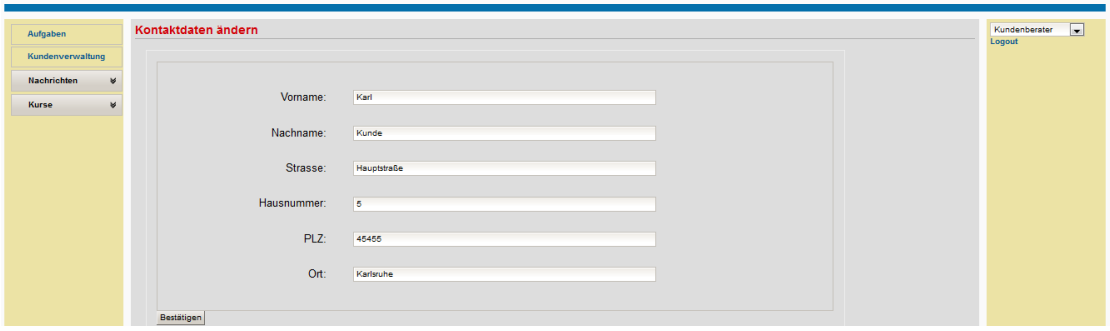


Abbildung A.17.: Persönlichedaten aendern

A.5.4. Nachrichten

Der Menüpunkt *Nachrichten* ist im Abschnitt A.4.3 beschrieben.

A.5.5. Kurse

Über den Menüpunkt *Kurse* im Navigationsbereich und den Unterpunkt *Kurse einsehen* gelangen Sie zur Ansicht *Kurse einsehen* (s. Abb. A.18).

In dieser Ansicht können Sie in dem Feld *WKN/ISIN* die WKN/ISIN eines Finanzinstruments eingeben. Daraufhin wird Ihnen der Name, der Typ, der aktuelle Kurswert, der mittlere Kurswert sowie die Durchschnittswerte der letzten Woche, des letzten Monats, des letzten Jahres und der durchschnittliche Wert seit dem ersten Kauf angezeigt.

Über den *Handeln* Button gelangen Sie zur Ansicht *Finanzinstrument handeln*, die in Abschnitt A.5.2.1 beschrieben ist.

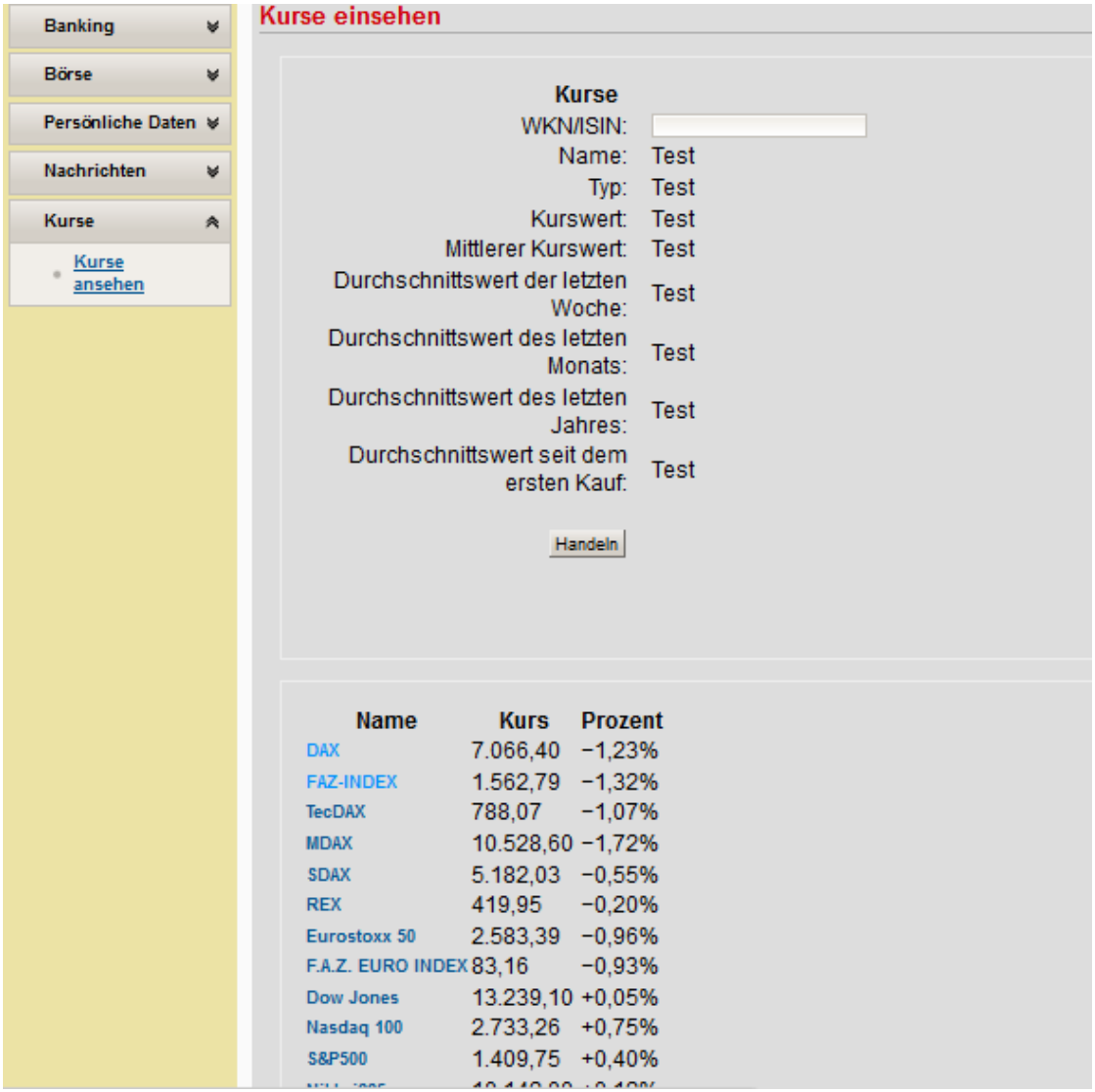


Abbildung A.18.: Kurse einsehen

A.6. Kundenberater

Autor: YD

Als Kundenberater sehen Sie nach dem Einloggen Ihre Benutzeroberfläche wie in Abb. A.19. Der Navigationsbereich zeigt die folgenden Einträge:

1. Aufgaben
2. Kundenverwaltung
3. Nachrichten
    - Nachrichten lesen
    - Nachrichten verschicken
4. Kurse

Die Unterpunkte in der Aufzählung sind die Unterpunkte der entsprechenden Einträge im Navigationsbereich.

Im folgenden werden die Funktionen an die Navigation angelehnt beschrieben.

Abbildung A.19.: Kundenberaterportal

A.6.1. Aufgaben

Über den Menüpunkt *Aufgaben* im Navigationsbereich gelangen Sie zur Ansicht *Aufgaben-Übersicht* (s. Abb. A.20).

Im oberen Teil der Ansicht werden unter *Im Auftrag gegebene Aufgaben*, in einer Tabelle die Aufgaben aufgelistet, die der Kundenberater erzeugt hat (z.B. die Überprüfung einer Person beim Eröffnen eines Kontos).

In der *Im Auftrag gegebene Aufgaben*-Tabelle können Aufgaben ausgewählt werden. Durch anschließendes betätigen des *Details* Buttons werden Ihnen die Details (*Name*, *Beschreibung*, *Weitere Daten*, *Status*, *Ersteller*, *Abteilung* und *Bearbeiter*) der ausgewählten Aufgabe in der Ansicht *Aufgaben-Details* (s. Abb. A.21) angezeigt. Durch betätigen des *Löschen* Buttons wird die ausgewählte Aufgabe gelöscht. Die gelöschten Aufgaben können in der Tabelle *Gelöschte Aufgaben* eingesehen werden.

A.6.2. Kundenverwaltung

Über den Menüpunkt *Kundenverwaltung* im Navigationsbereich gelangen Sie zur Ansicht *Kundenverwaltung Übersicht* (s. Abb. A.22).

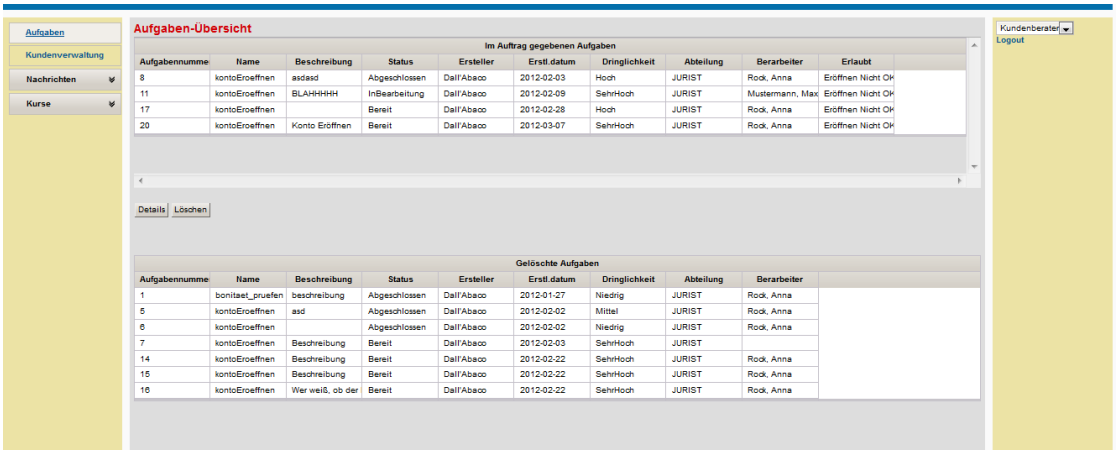


Abbildung A.20.: Aufgaben Übersicht

Abbildung A.21.: Aufgaben Details Übersicht

Im oberen Teil der Ansicht befinden sich die Felder *Kontonummer*, *Kundennummer*, *Name* und *Vorname* sowie der *Filtern* Button, dessen Funktion im Abschnitt A.6.2.1 beschrieben wird.

Im unteren Teil der Ansicht befindet sich eine Tabelle, in der die Kunden der *Magnus Monetarus Bank*aufgelistet sind und die Buttons *Ändern* (s. Abschnitt A.5.3.1) und *Details anzeigen* (s. Abschnitt A.6.2.4) in jeder Zeile und *Details anzeigen* (s. Abschnitt A.6.2.4), *Kunde anlegen* (s. Abschnitt A.6.2.2) und *Bonität prüfen lassen* (s. Abschnitt A.6.2.3).

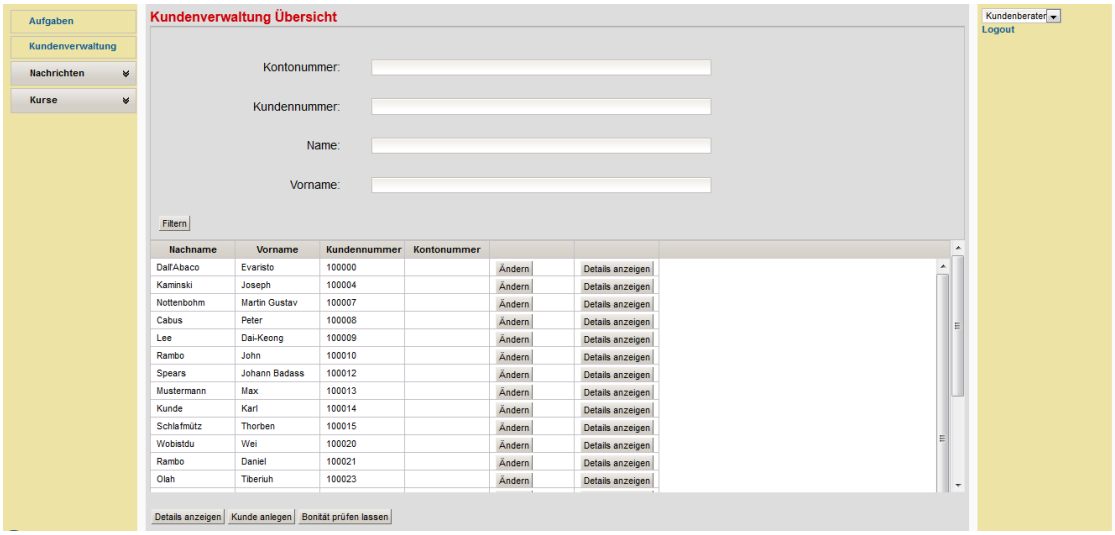


Abbildung A.22.: Kundenverwaltung-Übersicht

A.6.2.1. Filtern

Im unteren Teil der Ansicht *Kundenverwaltung Übersicht* befindet sich eine Tabelle, in der die Kunden der *Magnus Monetarus Bank*aufgelistet sind. Im oberen Teil der Ansicht können Sie die Einträge in der Tabelle filtern, indem Sie in den Feldern *Kontonummer*, *Kundennummer*, *Name* und *Vorname* die gewünschten Kriterien eingeben und auf den *Filtern* Button klicken. So können Sie beispielsweise mit den Einträgen *10000* im Feld *Kundennummer* nur die Kunden anzeigen lassen, deren Kundennummer *10000* enthält (Ergebnis: s. Abb. A.23).

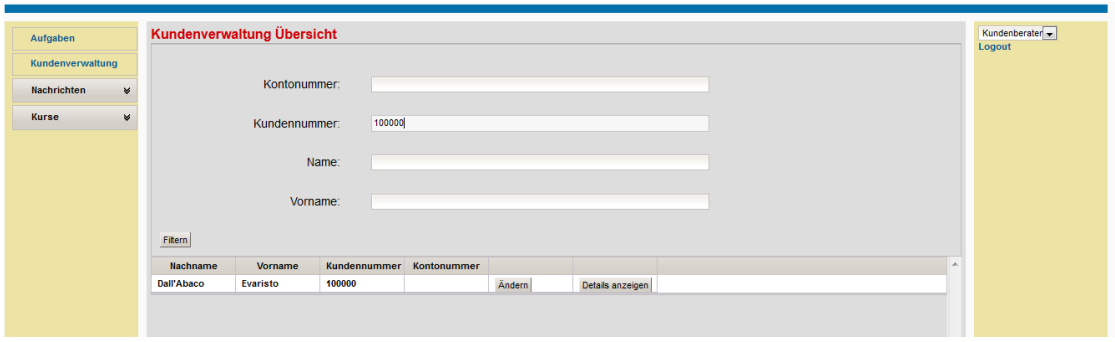


Abbildung A.23.: Kundenverwaltungmitfilter: ein Beispiel von Filtern-Funktion mit Kontonummer 100000

A.6.2.2. Kunden anlegen

Wenn Sie in der Ansicht *Kundenverwaltung Übersicht* den Button *Kunde anlegen* betätigt haben, gelangen Sie zur Ansicht *Kunden anlegen* (s. Abb. A.24).

Hier tragen Sie in die entsprechenden Felder *Vorname*, *Nachname*, *Titel*, *Geburtsdatum*, *Geburtsort*, *Straße*, *Hausnummer*, *PLZ*, *Ort* und zweimal das *Passwort* des Kunden ein und legen dann den Kunden mit Betätigen des *Bestätigen* Buttons an.

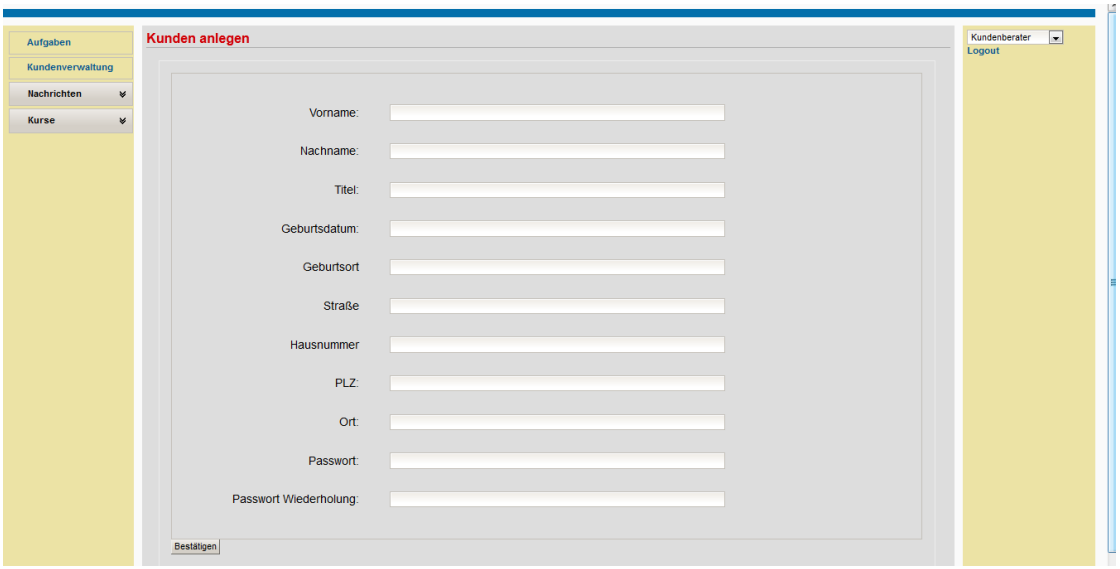


Abbildung A.24.: Beispiel: kundenanlegen

A.6.2.3. Bonität prüfen

Wenn Sie in der Ansicht *Kundenverwaltung Übersicht* einen Kunden ausgewählt haben und anschließend den Button *Bonität prüfen lassen* betätigt haben, gelangen Sie zur Ansicht *Bonität prüfen* (s. Abb. A.25).

Um eine Bonitätsprüfung des im oberen Teil der Ansicht angezeigten Kunden zu veranlassen, müssen Sie die folgenden Felder ausfüllen:

- Beschreibung: eine kurze Beschreibung des Grundes der Prüfung
- Dringlichkeit: die Dringlichkeit der Prüfung ist mit einem der vorgegebenen Werte *Niedrig*, *Mittel*, *Hoch* oder *Sehrhoch* anzugeben. Ist die Dringlichkeit *Sehrhoch*, wird die Prüfung dieses Kunden bei Möglichkeit bevorzugt.
- Frist: der späteste Zeitpunkt, an dem das Prüfungsergebniss vorliegen muss
- Weitere Informationen: sollten zu den vorangegangenen Punkten zusätzliche Informationen relevant sein, können und müssen diese unter *Weitere Informationen*

eingetragen werden.

Durch betätigen des Buttons *Absenden* wird die Aufgabe erstellt und an die juristische Abteilung gesendet.

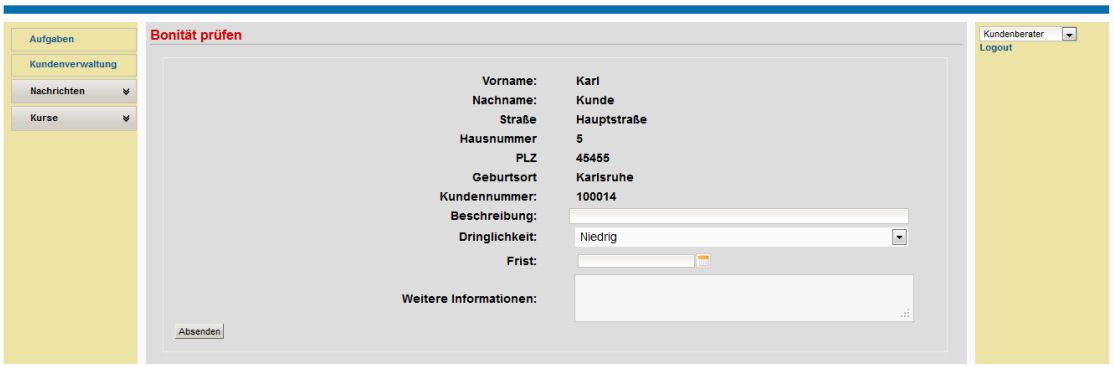


Abbildung A.25.: Beispiel: bonitätpruefen

A.6.2.4. Kundendetail-Ansicht

Wenn Sie in der Tabelle im unteren Teil der Ansicht *Kundenverwaltung Übersicht* einen Kunden auswählen und anschließend auf den Button *Details anzeigen* in der gleichen Zeile oder den gleichnamigen Button unterhalb der Tabelle betätigen, gelangen Sie in die Ansicht *Kundendetail-Ansicht*.

In der *Kundentetail-Ansicht* werden im oberen Teil der Ansicht die persönlichen Daten und die Kundennummer des Kunden angezeigt. Unter den persönlichen Daten befindet sich eine Tabelle mit den Konten des Kunden bei der *Magnus Monetarus Bank*. Durch das Auswählen eines Kontos in der Tabelle und das betätigen des Buttons *Details* in der gleichen Zeile oder des gleichnamigen Buttons unterhalb der Tabelle gelangen Sie zur *Konto Detailansicht* (s. Abschnitt A.6.2.5).

Weiter befinden sich folgende Buttons unterhalb der Konto-Tabelle:

- Durch das Betätigen des Buttons *Kontosperrung beantragen* wird eine Aufgabe erstellt und an die juristische Abteilung gesendet.
- Über den Button *Nachricht schicken* gelangen Sie zur Ansicht *Nachricht verschicken*, die im Abschnitt A.4.3 beschrieben wird.
- Über den Button *Konto eröffnen* gelangen Sie zur Ansicht *Konto eröffnen*, in der Sie die, für die Eröffnung eines Kontos notwendigen Daten eintragen und ein neues Konto für den Kunden eröffnen können.
- Über den Button *Kredit gewähren* gelangen Sie zur Ansicht *Kredit anlegen*, in der Sie dem Kunden einen Kredit in einer bestimmten Höhe und über eine bestimmte Laufzeit mit festgelegten Raten gewähren können.

- Durch das Betätigen des Buttons *Bonitätsprüfung* gelangen Sie zur Ansicht *Bonität prüfen*, die im Abschnitt A.6.2.3 beschrieben wird.
- Der Button *PuV* ist für den Kundenberater deaktiviert.
- Über den Button *Persönliche Daten ändern* gelangen Sie zur Ansicht *Kontaktdaten ändern*, die im Abschnitt A.5.3.1 beschrieben wird.
- Durch betätigen des Buttons *Kunde löschen* wird der Kunde aus dem System gelöscht. Dies geschieht jedoch nur, wenn alle Konten des Kunden ausgeglichen sind.
- Über den Button *Buchungen einsehen* gelangen Sie zur Ansicht *Buchungen*, die im Abschnitt A.5.1.5 beschrieben wird.
- Über den Button *Daueraufträge einsehen* gelangen Sie zur Ansicht *Daueraufträge einsehen*, die im Abschnitt A.6.2.9 beschrieben wird.
- Über den Button *Überweisung tätigen* gelangen Sie zur Ansicht *Überweisung tätigen*, die im Abschnitt A.5.1.1 beschrieben wird.
- Über den Button *Lastschrift tätigen* gelangen Sie zur Ansicht *Lastschrift tätigen*, die im Abschnitt A.5.1.2 beschrieben wird.
- Über den Button *Dauerauftrag tätigen* gelangen Sie zur Ansicht *Dauerauftrag*, die im Abschnitt A.5.1.3 beschrieben wird.

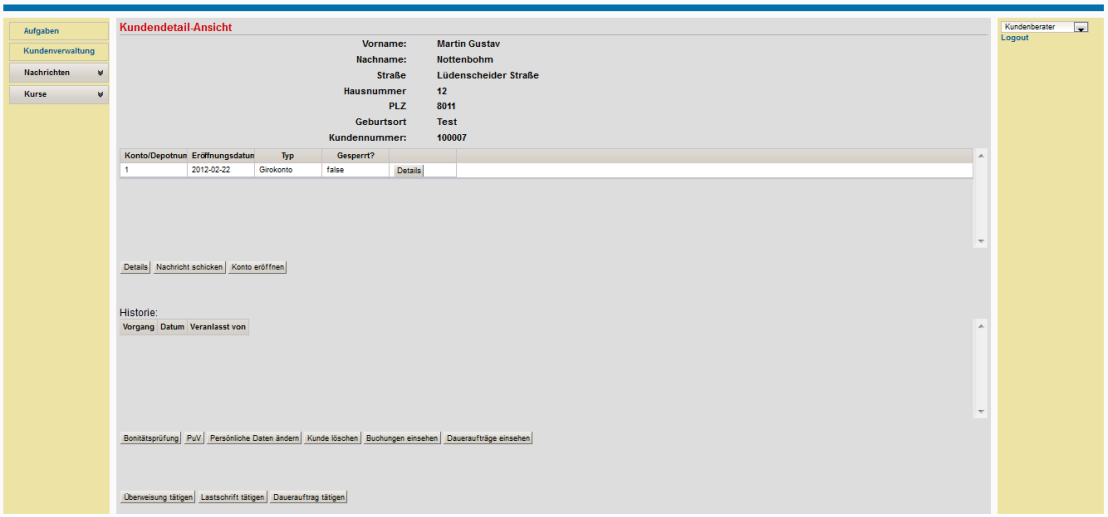


Abbildung A.26.: Beispiel: kundendetailsanzeigen

A.6.2.5. Konto Detailsansicht

Zur *Konto Detailansicht* gelangen Sie wie Abschnitt A.6.2.4 beschrieben.

Die der obere Teil der *Konto Detailansicht* besteht aus den zwei Unteransichten *Kundendaten* und *Zinsen*. Die Ansichten sind über die Reiter im oberen Teil der Ansicht durch Anclicken der Reiter erreichbar (s. Abb. A.28).

In der Unteransicht *Kundendaten* werden Ihnen *Vorname*, *Nachname* und *Kundennummer* des Kunden und *Kontonummer*, *Bankleitzahl*, *Kontotyp* und *Kontostand* des Kontos angezeigt (s. Abb. A.27).

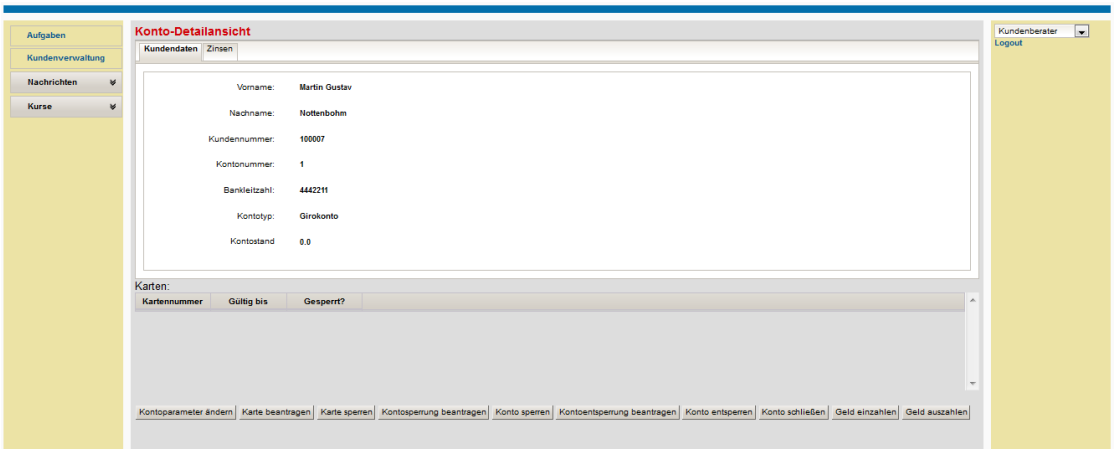


Abbildung A.27.: kundendatenansicht

In der Unteransicht *Zinsen* werden Ihnen die *Zinsen bei positivem Kontostand*, die *Zinsen innerhalb des Dispokredits*, die *Dispogrenze* und die *Harte Grenze* angezeigt (s. Abb. A.28). Die *Dispogrenze* ist der niedrigste Kontostand, für den der Dispokredit gewährt wird. Die *Harte Grenze* ist der niedrigste Kontostand, der dem Kunden zugestanden wird, bevor das Konto gesperrt wird.

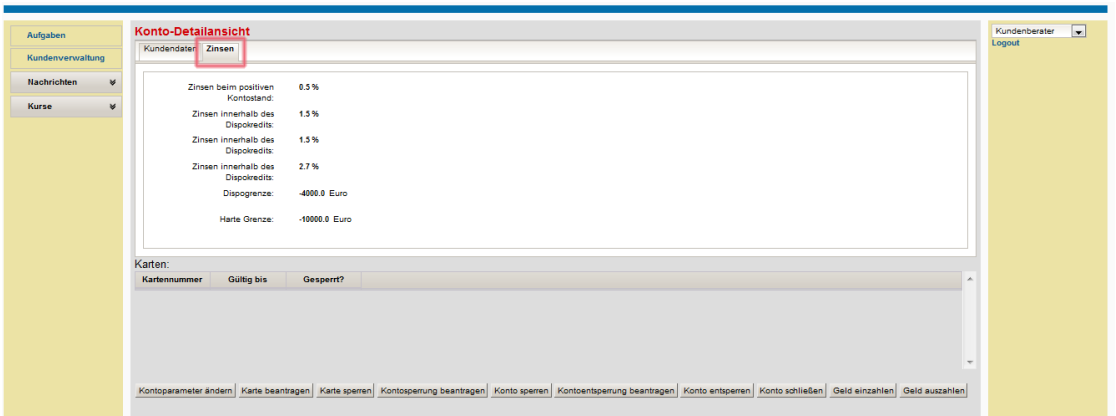


Abbildung A.28.: Zinsen-ansicht

Im unteren Teil der *Konto Detailansicht* befindet sich eine Tabelle, die mit *Karten*

überschrieben ist. In dieser Tabelle werden die Karten mit *Kartennummer*, dem Gültigkeitsdatum *Gültig bis* und dem Wahrheitswert *Gesperrt* aufgelistet, die dem Kunden für das Konto ausgestellt wurden.

Unter der Tabelle befinden sich die folgenden Buttons:

- Kontoparameter ändern: Über den Button *Kontoparameter ändern* gelangen Sie zur Ansicht *Kontoparameter ändern*, die in Abschnitt A.6.2.6 beschrieben wird.
- Karte beantragen: Über den Button *Karte beantragen* wird eine Aufgabe erstellt, die für das Konto das Beantragen der Produktion einer Karte anstößt.
- Karte sperren: Durch das Auswählen einer nicht gesperrten Karte in der Tabelle und das anschließende betätigen des Buttons *Karte sperren* wird die ausgewählte Karte gesperrt.
- Kontosperrung beantragen: Über den Button *Kontosperrung beantragen* wird eine Aufgabe erstellt, und an die juristische Abteilung übermittelt.
- Konto sperren: Wurde durch die juristische Abteilung das Sperren des Kontos autorisiert, kann durch betätigen des Buttons *Konto sperren* das Konto gesperrt werden.
- Konto schließen: Wenn das Konto ausgeglichen ist, kann es durch Betätigen des Buttons *Konto schließen* geschlossen werden.
- Geld einzahlen: Über den Button *Geld einzahlen* gelangen Sie zur Ansicht *Geld einzahlen*, die in Abschnitt A.6.2.7 beschrieben wird.
- Geld auszahlen:Über den Button *Geld auszahlen* gelangen Sie zur Ansicht *Geld auszahlen*, die in Abschnitt A.6.2.8 beschrieben wird.

A.6.2.6. Kontoparameter ändern

In der Ansicht *Kontoparameter ändern* können Sie das im oberen Teil der Ansicht mit den Feldern *Nachname*, *Vorname*, *Kontonummer*, *Kundennummer* und *Art des Kontos* beschriebene Konto verändern, indem Sie die Kontoparameter, die in Abschnitt A.6.2.5 erläutert werden, in den gleichbenannten Feldern neu setzten und die Eingaben mit dem Button *Bestätigen* bestätigen (s. Abb. A.29).

A.6.2.7. Geld einzahlen

In der Ansicht *Geld einzahlen* (s. Abb. A.30) können Sie einen Betrag in Euro in das Feld *Betrag* eingeben. Der Betrag muss zwischen 0 und 10000 Euro liegen. Durch betätigen des Buttons *Bestätigen* wird der eingegebene Betrag dem im Feld *Konto* angezeigten Konto gutgeschrieben.

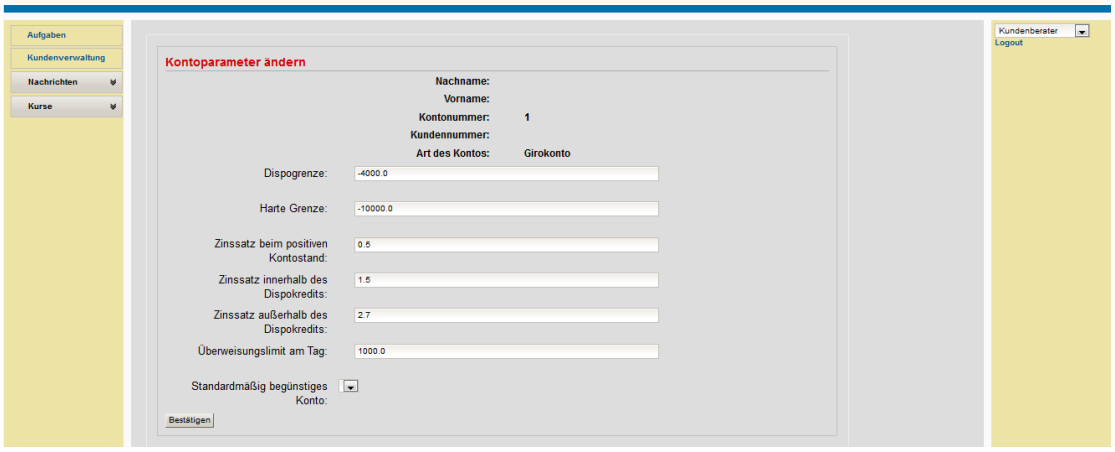


Abbildung A.29.: Kontoparameteraendern

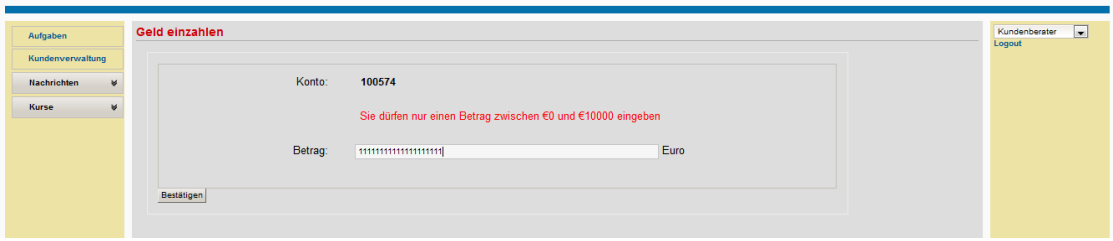


Abbildung A.30.: Beispiel: Überschreitung des Maxmums von dem Seite geldeinzahlen

A.6.2.8. Geld auszahlen

In der Ansicht *Geld auszahlen* (s. Abb. A.31) können Sie im Feld *Konto* ein Konto auswählen und einen Betrag in Euro in das Feld *Betrag* eingeben. Der Betrag muss zwischen 0 und 10000 Euro liegen. Durch betätigen des Buttons *Bestätigen* wird der eingegebene Betrag von dem im Feld *Konto* angezeigten Konto abgehoben.

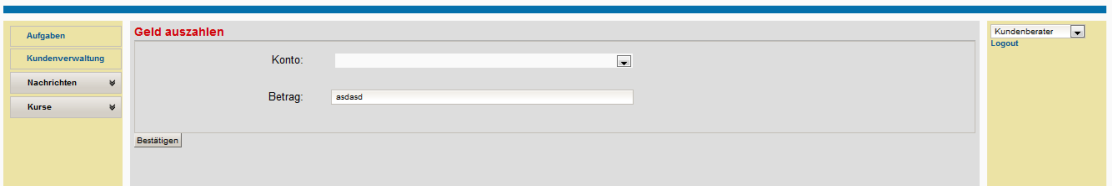


Abbildung A.31.: Geld auszahlen

A.6.2.9. Daueraufträge einsehen

In der Ansicht *Daueraufträge einsehen* befindet sich eine tabellarische Aufzählung der Daueraufträge des vorher ausgewählten Kunden und die Buttons *Bearbeiten* und *Löschen* (s. Abb. A.32).

Wenn Sie einen Dauerauftrag in der Tabelle auswählen und anschließend den Button *Bearbeiten* betätigen, gelangen Sie in die Ansicht *Dauerauftrag bearbeiten*, die in Abschnitt A.6.2.10 beschrieben wird.

Wenn Sie einen Dauerauftrag in der Tabelle auswählen und anschließend den Button *Löschen* betätigen, wird der ausgewählte Dauerauftrag gelöscht.

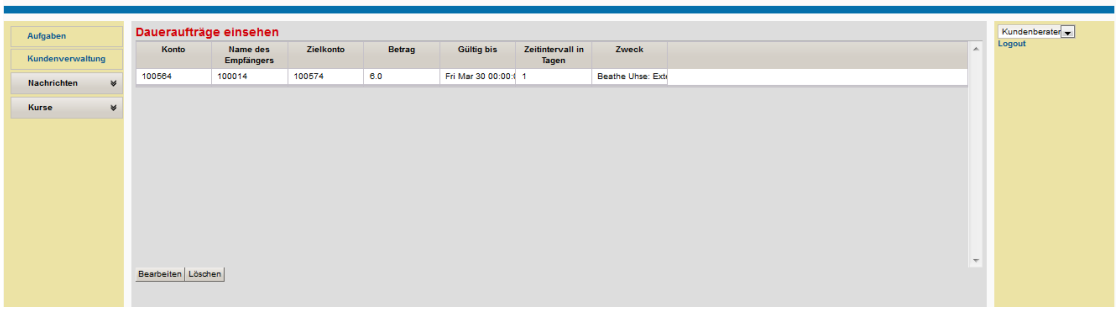


Abbildung A.32.: Daueraufträg einsehen

A.6.2.10. Dauerauftrag bearbeiten

In der Ansicht *Dauerauftrag bearbeiten* können Sie den im oberen Teil der Ansicht *Dauerauftrag bearbeiten* den ausgewählten Dauerauftrag verändern.

A.6.3. Nachrichten

Der Menüpunkt *Nachrichten* entspricht Abschnitt A.4.3 für Mitarbeiter.

A.6.4. Kurse

In der Ansicht *Kurse* können die aktuellen Kursstände eingesehen werden (s. Abb. A.33).

Im oberen Teil der Ansicht können Sie in dem Feld *WKN/ISIN* die WKN (Wertpapierkennnummer) oder ISIN (International Securities Identifications Number) eingeben, woraufhin Ihnen Details zu dem entsprechenden Wertpapier angezeigt werden. Wenn sie den Button *Handeln* betätigen, gelangen sie zur Ansicht *Finanzinstrument handeln* (s. Abschnitt A.5.2.1).

Im unteren Teil der Ansicht werden Ihnen bekannte Wertpapiere und deren aktueller Kurs aufgelistet.

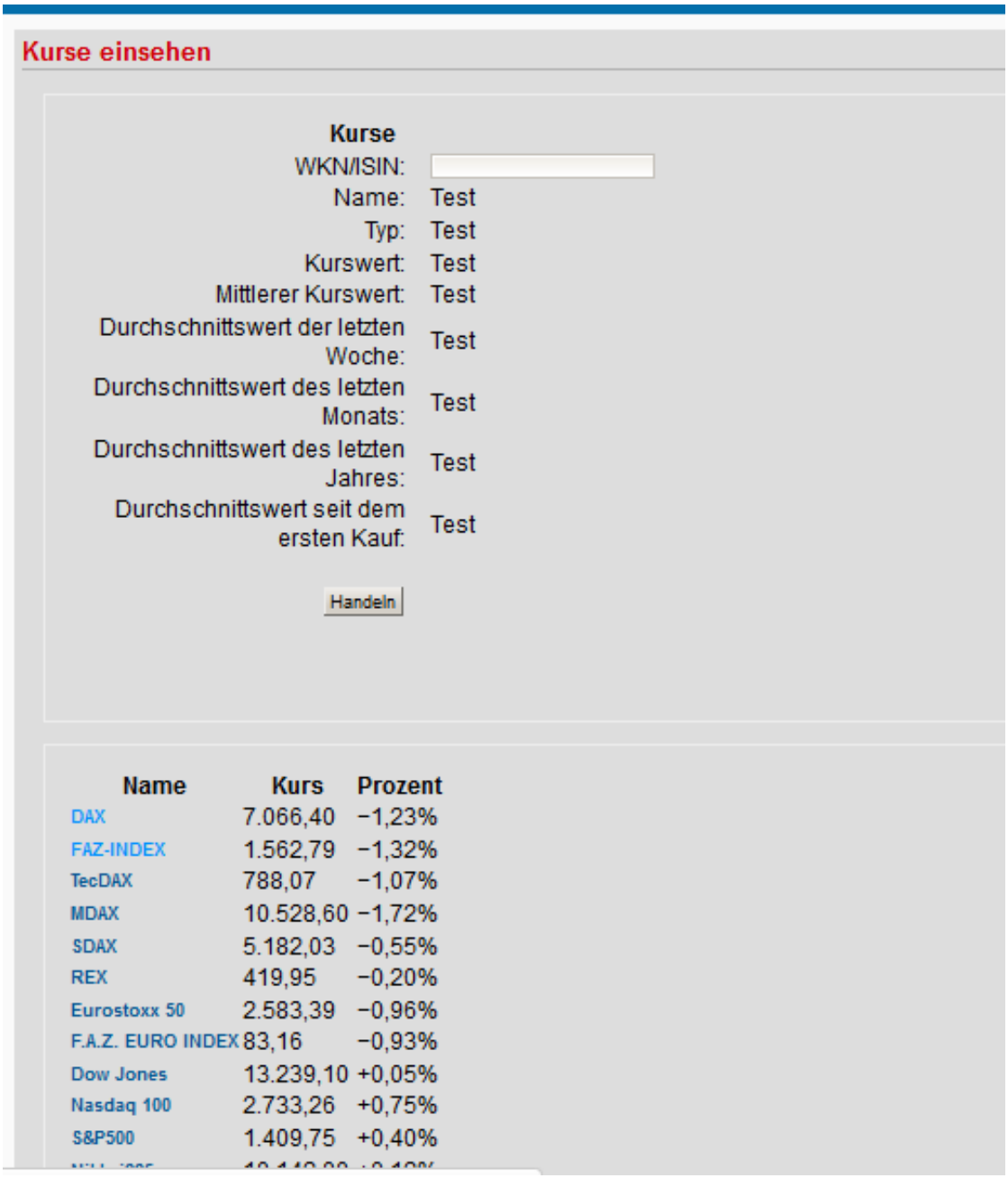


Abbildung A.33.: kurseeinsehen

A.7. Kassierer

Als Kassierer sehen Sie nach dem Einloggen befindet sich im Navigationsmenü nur der Eintrag *Kundenverwaltung* (s. Abb. A.34), der in Abschnitt A.6.2 beschrieben wird. Jedoch ist die Ansicht des Kassierers auf die Funktionen *Filtern* (s. Abschnitt A.6.2.1) und *Details anzeigen* (s. Abschnitt A.6.2.4) eingeschränkt.

Die Ansicht *Kundendetail-Ansicht* ist für den Kassierer auf die Funktion der Buttons, die mit *Details* beschrieben sind und zur *Konto Detailansicht* (s. Abschnitt A.6.2.5) führen.

Die *Konto Detailansicht* ist auf die Funktionen *Geld einzahlen* (s. Abschnitt A.6.2.7) und *Geld auszahlen* (s. Abschnitt A.6.2.8) beschränkt.

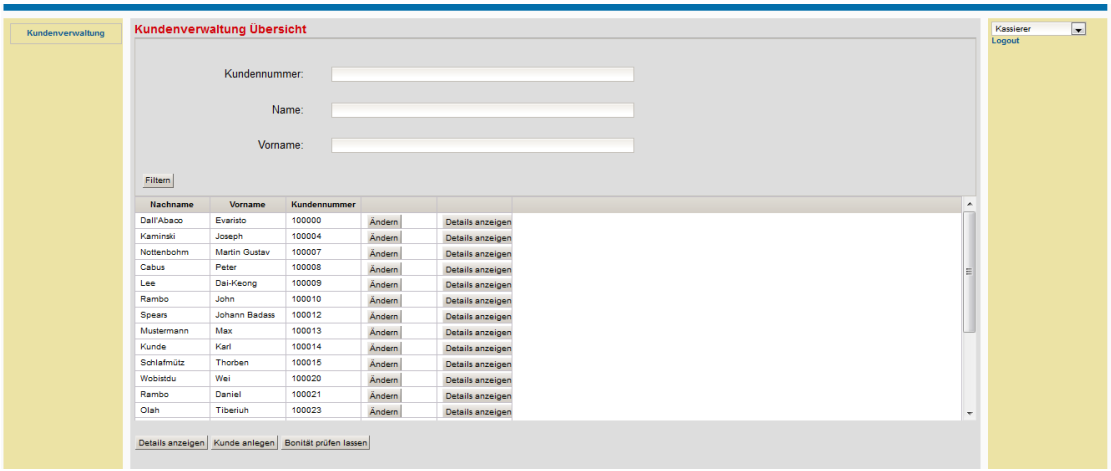


Abbildung A.34.: kundenverwaltung

A.8. Produktentwickler

Autor: YD

Als Produktentwickler und Marketing-Mitarbeiter sehen Sie nach dem Einloggen Ihre Benutzeroberfläche wie in Abb. A.35. Der Navigationsbereich zeigt die folgenden Einträge:

1. Aufgaben
2. Statistische Daten
3. Nachrichten

Abbildung A.35.: Produktentwickler-Portal

A.8.1. Aufgaben

Die Ansicht *Aufgaben* des Produktentwicklers und des Marketing-Mitarbeiters sind identisch mit der Ansicht *Aufgaben* des Kundenberaters (s. Abschnitt A.6.1).

A.8.2. Statistiche Daten

Über den Menüpunkt *statistische Daten* im Navigationsmenü gelangen Sie zur Ansicht *Statistische Daten* (s. Abb. A.36). Hier können Sie eine Aufgabe auswählen, die Sie reserviert haben und die geforderte Anfrage übertragen. Das Ergebnis können Sie durch betätigen des Buttons *Nachricht verschicken* an den Anfragenden schicken oder durch betätigen des Buttons *Extrahieren* als PDF-Datei aus dem System extrahieren.

A.8.3. Nachrichten

Die Funktionen unter dem Menüpunkt *Nachrichten* in der Navigationsleiste für Produktentwickler und Marketing-Mitarbeiter werden im Abschnitt A.4.3 beschrieben.

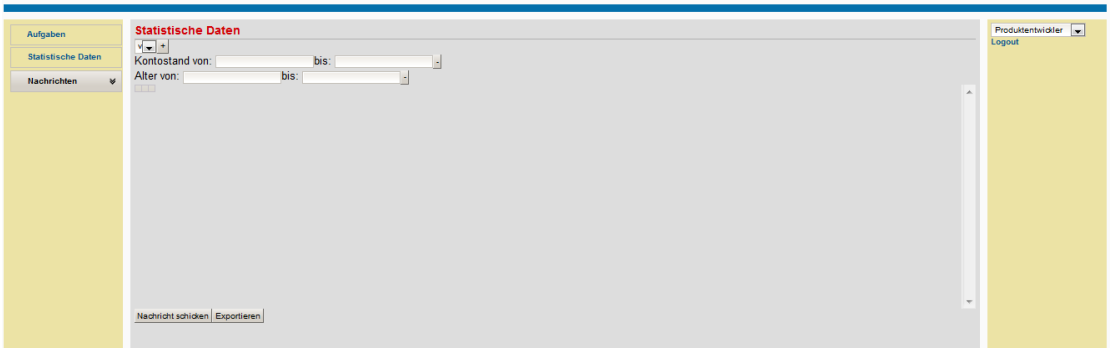


Abbildung A.36.: Statischedaten

A.9. Marketing-Mitarbeiter

Alle Funktionen, die Ihnen als Marketing-Mitarbeiter zur Verfügung stehen, entsprechen den Funktionen, die im Kapitel A.8 *Produktentwickler* beschrieben werden.

A.10. Controlling-Mitarbeiter

Autor: YD

Als Controlling-Mitarbeiter sehen Sie nach dem Einloggen Ihre Benutzeroberfläche wie in Abb. A.37. Der Navigationsbereich zeigt die folgenden Einträge:

1. Aufgaben
2. Kundenverwaltung
3. Logs einsehen
4. Nachrichten
5. Kurs einsehen

Controlling-Mitarbeit: Mitarbeiter der Bank, die in der Controllingabteilung arbeiten. [32][Pflichtenheft]

Unter der Rolle Controlling-Mitarbeiter bietet das System folgende Service an, *Aufgaben*, *Kundenverwaltung*, *logs einsehen*, *Nachrichten lesen* und *kurse einsehen* (s. Abb. A.37).

A.10.1. Aufgaben

Das Aufgabe-Workflow unter Controlling-Mitarbeiter sind identisch wie das Aufgabe-Workflow unter Kundenberater. (siehe Abschnitt Kundenberater A.6.1)

Abbildung A.37.: Controlling-portal

A.10.2. Kundenverwaltung

Die Kundenverwaltung unter Controlling-Mitarbeiter sind identisch wie die Kundenverwaltung unter Kundenberater. (siehe Abschnitt Kundenberater A.6.2)

A.10.3. Logs einsehen

Im System werden alle Ereignisse und das Fehlverhalten des Systems geloggt. Diese Logdateien kann sich der Akteur einsehen und analysieren.[32][Pflichtenheft]

Wenn der Akteur auf dem Menüleiste „*Logs einsehen* "drückt, wird Logs angezeigt, die nach dem Datum, Uhrzeit und Aktivität sortiert sind. Da die Logs nach langer Zeit automatisch viel generiert werden, kann der Akteur die Logs in einer bestimmte Zeitraum aufrufen.(Abb. A.38)

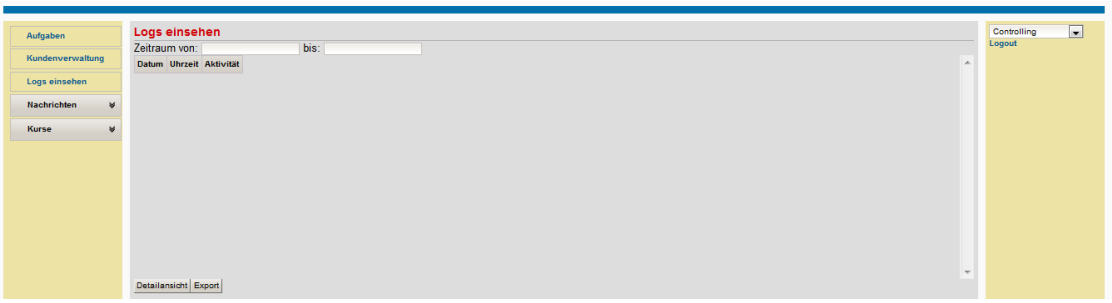


Abbildung A.38.: logseinsehen

A.10.4. Nachrichten

Das Nachrichtensystem unter Controlling-Mitarbeiter sind identisch wie Nachrichtensystem unter Kundenberater. (siehe Kapitel A.4.3)

A.10.5. Kurse einsehen

Die Funktion „*Kurse einsehen* “unter Controlling-Mitarbeiter sind identisch wie Nachrichtensystem unter Kundenberater. (siehe Kapitel A.6.4)

A.11. Jurist

Autor: CW

In Abb. A.39 wird die Start Seite des Juristen angezeigt. Der Navigationsbereich zeigt die folgenden Einträge:

1. Aufgabenverwaltung
2. Kundenverwaltung
3. Kontodatenliste beantragen
4. Nachrichten
    - Nachrichten lesen
    - Nachrichten verschicken

Der Jurist-Mitarbeiter befasst sich mit der Überbrückungsfunktion zwischen Bank und staatlichen Behörden wie Bafin und BZST bei Anfrage von Kontodaten. Darüber hinaus lässt sich die Kundenverwaltung anhand der Sachverhalten durchführen oder mit den anderen Abteilungen(Controlling, Kundenberater) zusammenarbeiten.

Abbildung A.39.: Portal für Jurist-Mitarbeiter

A.11.1. Aufgaben einsehen

Der Jurist-Mitarbeiter kann sich durch Klick des *Aufgaben* Buttons sofort über juristische Aufgaben wie in Abb. A.40 informieren (siehe auch Abschnitt A.6.1).

A.11.2. Kundenverwaltung

Um juristisch relevante Aufgaben zu erledigen, braucht der Jurist-Mitarbeiter auch die Funktionen von Kundenverwaltung. Sie bietet die Möglichkeiten, eine Übersicht über Kunden bzw. Konten zu anzuzeigen. Wenn es nötig ist, können die kunden- oder konto-bezogene Daten ebenfalls verarbeitet werden. Die Kundenverwaltung für die juristische Abteilung ist gleich wie beim Kundenberater in Abschnitt A.6.2.

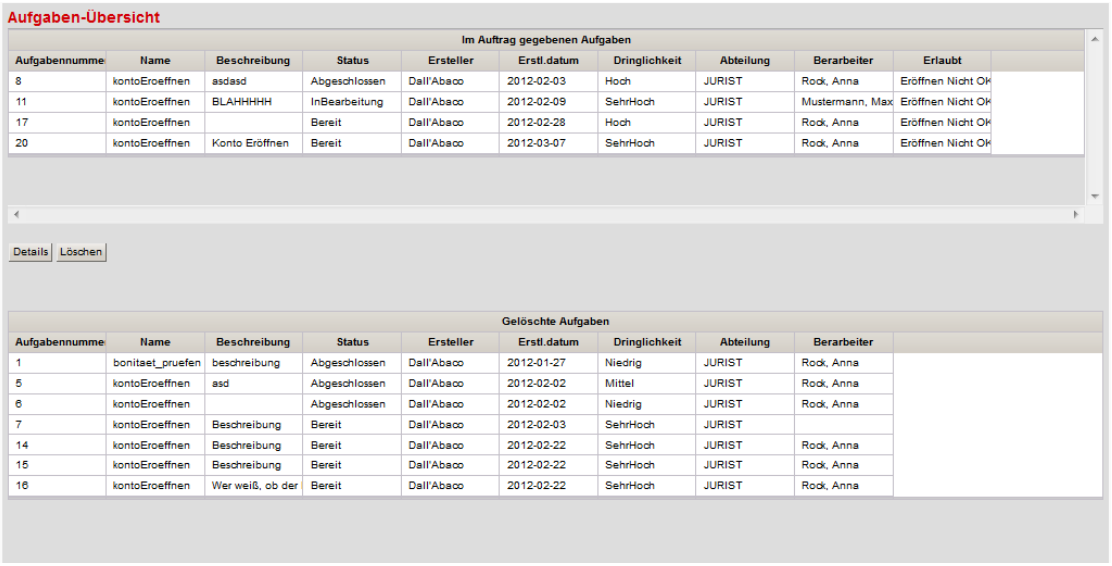


Abbildung A.40.: Anzeige der juristischen Aufgaben

A.11.3. Kontodatenliste beantragen

Solange der Jurist-Mitarbeiter von Bafin oder BZSt eine Anfrage zu kundenbezogenen Information von gewissen Konten bekommt, erhält er dabei eine Vorgangsnummer, um die nachfolgende Aktion *Kundendatenliste beantragen* zu bestätigen und identifizieren. Es gibt zwei Möglichkeiten die Kontodatenliste zu erstellen. Entweder auf gewisse (z.B. zeitliche, zahlungsmäßige) Bedingungen gerichtet die Buchungen eines Kontos auszusortieren oder direkt anhand der geforderten Kontonummer abzufragen und eine dementsprechende Kontodatenliste zu erstellen.

Abbildung A.41.: Kontodatenliste beantragen

A.11.4. Nachrichtenverkehr

Sehen Sie bitte den Abschnitt A.4.3.

A.12. Geschäftsführer

Autor: CW

Für Geschäftsführer hat das Online System auch die dafür benötigten Funktionen bereitgestellt. Die Ansicht von dem Geschäftsführer Portal ist wie in Abb. A.42 gezeigt. Der Navigationsbereich zeigt die folgenden Einträge:

1. Limits setzen
2. Aufgabenverwaltung
3. Kundenverwaltung
4. Logs einsehen
5. Kontodatenliste beantragen
6. Statistische Daten
7. Einstellung für verdächtige Überweisungen
8. Nachrichten
    - Nachrichten lesen
    - Nachrichten verschicken
9. Kurse
    - Kurse einsehen

Abbildung A.42.: Start Seite für Geschäftsführer

A.12.1. Limits setzen

Limitsystem dient dazu, die potenziellen Widerrechtlichkeiten aus dem Eigenhandel zu vermeiden. In Bezug auf das Limitsystem sollen die in Abb. A.43 aufgelisteten Parameter(Overnight-, Intraday-, Stop-Loss-, Volumen-Limit und Maximale Verlust) in

zwei Arten (weiche und harte Grenze) ausgefüllt werden. Das Überschreiten der weichen Grenze erzeugt eine Warnung, das Überschreiten der harten Grenze wird durch das System verhindert.

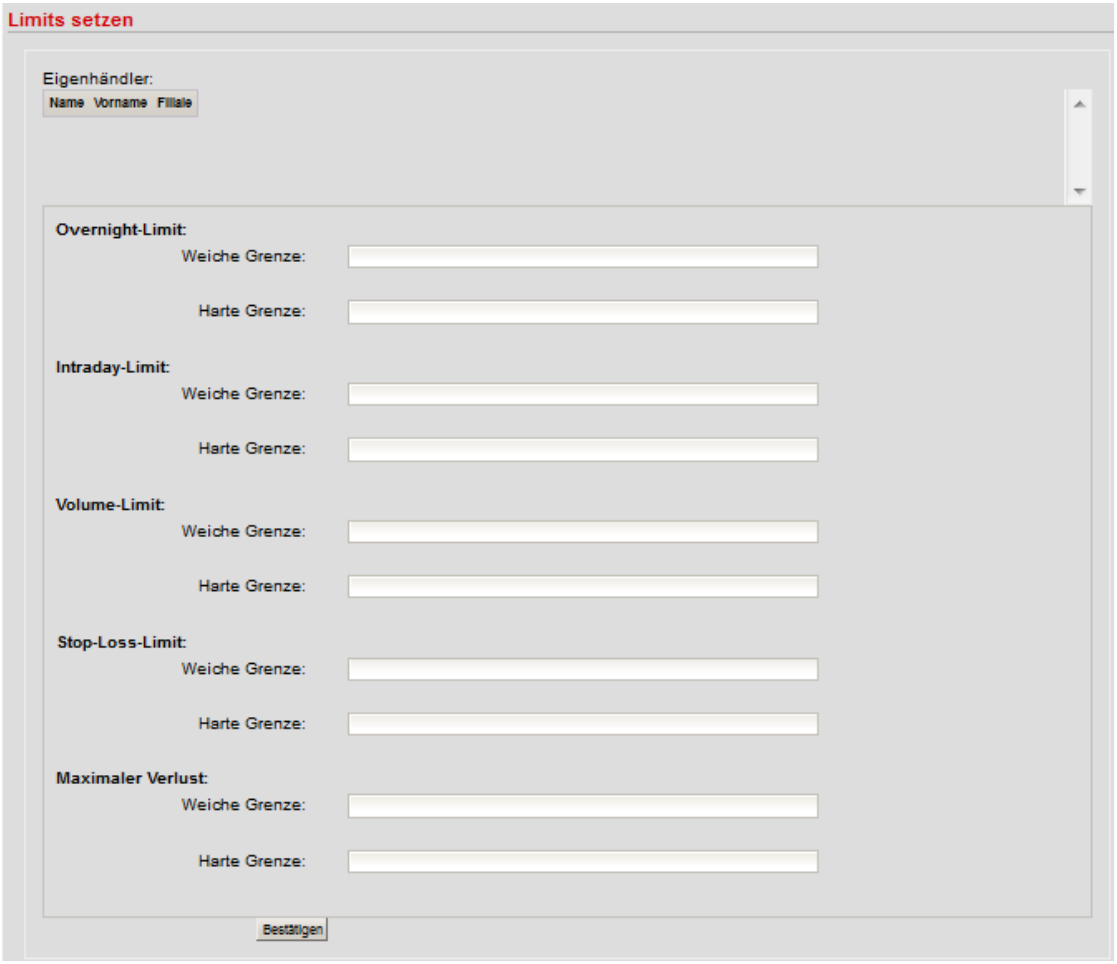


Abbildung A.43.: Limits setzen

A.12.2. Kundenverwaltung

Sehen Sie bitte den Abschnitt A.6.2

A.12.3. Log einsehen

Sehen Sie bitte den Abschnitt A.14.3

A.12.4. Kontodatenliste beantragen

Sehen Sie bitte den Abschnitt A.11.3

A.12.5. Statistische Daten

Sehen Sie bitte den Abschnitt A.8.2

A.12.6. Einstellung der Verdächtigkeitsprüfung für Überweisung

Diese Einstellung hilft unser System bei der Verdächtigkeitsprüfung für Überweisung, indem ein Grenzwert wie in Abb. A.44 eingetragen wird. Dies erlaubt, die Verdächtigkeitsprüfung für Überweisung automatisiert durchzuführen.

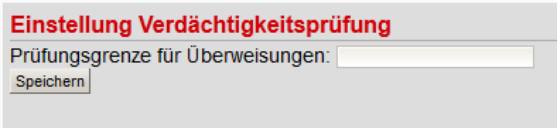


Abbildung A.44.: Parametrisierte Einstellung der Überweisungsverdächtigkeit

A.12.7. Nachrichtenverkehr

Sehen Sie bitte den Abschnitt A.4.3

A.12.8. Kurse einsehen

Sehen Sie bitte den Abschnitt A.5.5

A.13. Eigenhändler

Autor: CW

Als Eigenhändler können Sie Ihre Finanzinstrumente online an- oder verkaufen. Zusätzlich dazusind die Funktionen für Eigenhandel in Zusammenhang mit Kundenangelegenheiten verfügbar, um die aktuellen Zustände von Finanzinstrumenten oder den aktuellen Kontostand nachzufragen . In Abb. A.45 wird die Start Seite für den Eigenhändler nach Einloggen angezeigt. Der Navigationsbereich zeigt die folgenden Einträge:

1. Börseangelegenheiten
    - Order erstellen

- Depot einsehen
- Orderverwaltung

2. Kontostand
3. Nachrichten
   - Nachrichten lesen
   - Nachrichten verschicken
4. Kurse
   - Kurse einsehen

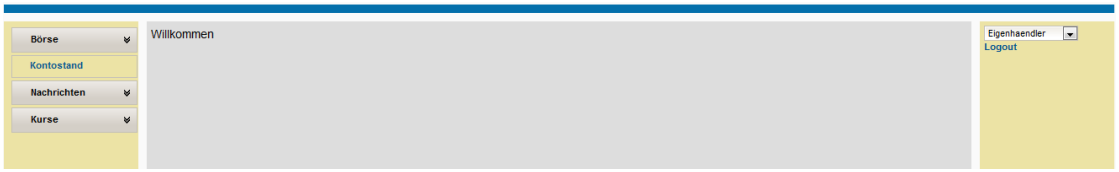


Abbildung A.45.: Start Seite für Eigenhändler

A.13.1. Börseangelegenheit

Sehen Sie bitte den Abschnitt A.5.2.

A.13.2. Kontostand einsehen

Sehen Sie bitte den Abschnitt A.5.1.4.

A.13.3. Nachrichtenverkehr

Sehen Sie bitte den Abschnitt A.4.3.

A.14. Systemadministrator

Autor: CW

Als Kundenberater sehen Sie nach dem Einloggen Ihre Benutzeroberfläche wie in Abb. A.46. Der Navigationsbereich zeigt die folgenden Einträge:

1. Systemstatus einsehen
2. Mitarbeiterverwaltung

3. Logs einsehen
4. Logs auswerten
5. Backup erstellen
6. RSA-Schlüssel-Verwaltung
7. Nachrichten
    - Nachrichten lesen
    - Nachrichten verschicken
8. Runterfahren

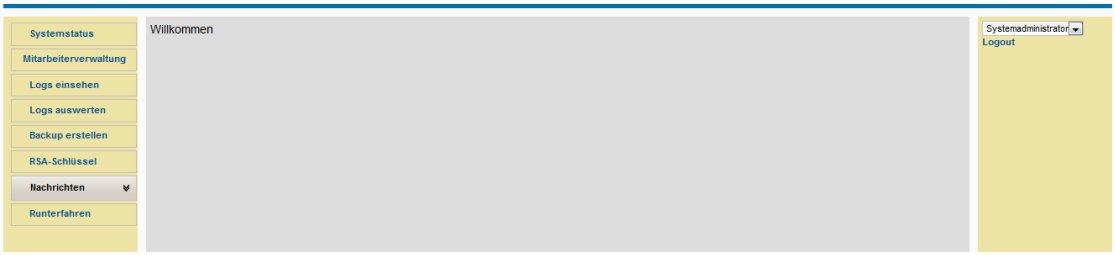


Abbildung A.46.: Start Seite für Systemadministrator

A.14.1. Systemstatus einsehen

Klicken Sie den Button *Systemstatus*, so wird der aktuelle Systemstatus, der die Informationen über Laufzeit Speicher/CPU-Auslastung des Cloudservers enthält, wie in Abb. A.47 angezeigt wird.

A.14.2. Mitarbeiterverwaltung

Die Mitarbeiterverwaltung ist von Systemadministrator zu übernehmen. Die Ansicht *Mitarbeiterverwaltung* (s. Abb. A.48) ist ähnlich wie Kundenverwaltung. In der Ansicht *Mitarbeiterverwaltung* befindet sich eine Tabelle, in der alle aktuellen Mitarbeiter mit ihren Namen, Benutzernamen und Rollen eingetragen sind. Wählen Sie einen bestimmten davon aus, gelangen Sie durch anschließendes Betätigen des Buttons *Nutzer bearbeiten* gelangen Sie zur Ansicht *Nutzerrechte*, die in Abb. A.49 gezeigt wird. In dieser Ansicht können Sie die persönlichen Daten und Nutzerrechte des ausgewählten Mitarbeiters bearbeiten.

Abbildung A.47.: aktueller Serverstatus anzeigen

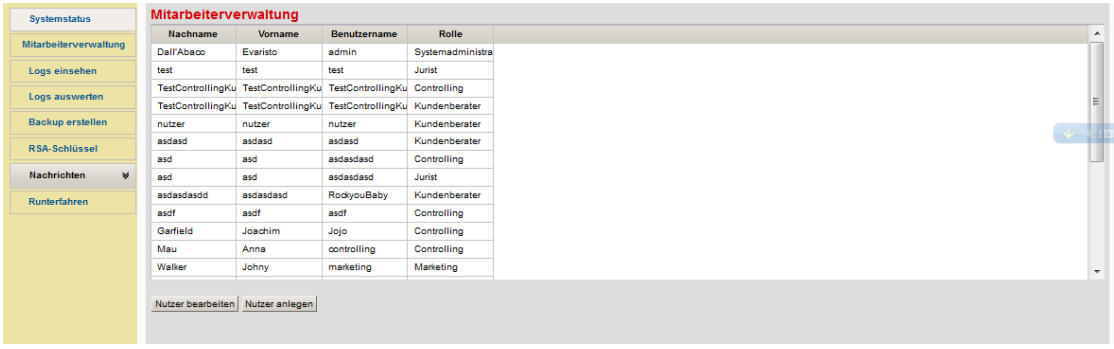


Abbildung A.48.: Mitarbeiterverwaltung

A.14.3. Log einsehen

Um Systemanomalien rechtzeitig zu informieren bzw. die Systemlogs anzuschauen ist dazu benötigt. Die Logdaten, welche unter einem vordefinierten Verzeichnis abgelegt sind, werden aufgelistet. Wählen Sie einer von denen aus. Dann durch Klick des “Detailansicht“ Buttons wird dies Logfile auf der Benutzeroberfläche tabellarisch angezeigt. Wenn Sie den Button “Export“ drücken, wird die ausgewählten Logs in File-Format ausgegeben.

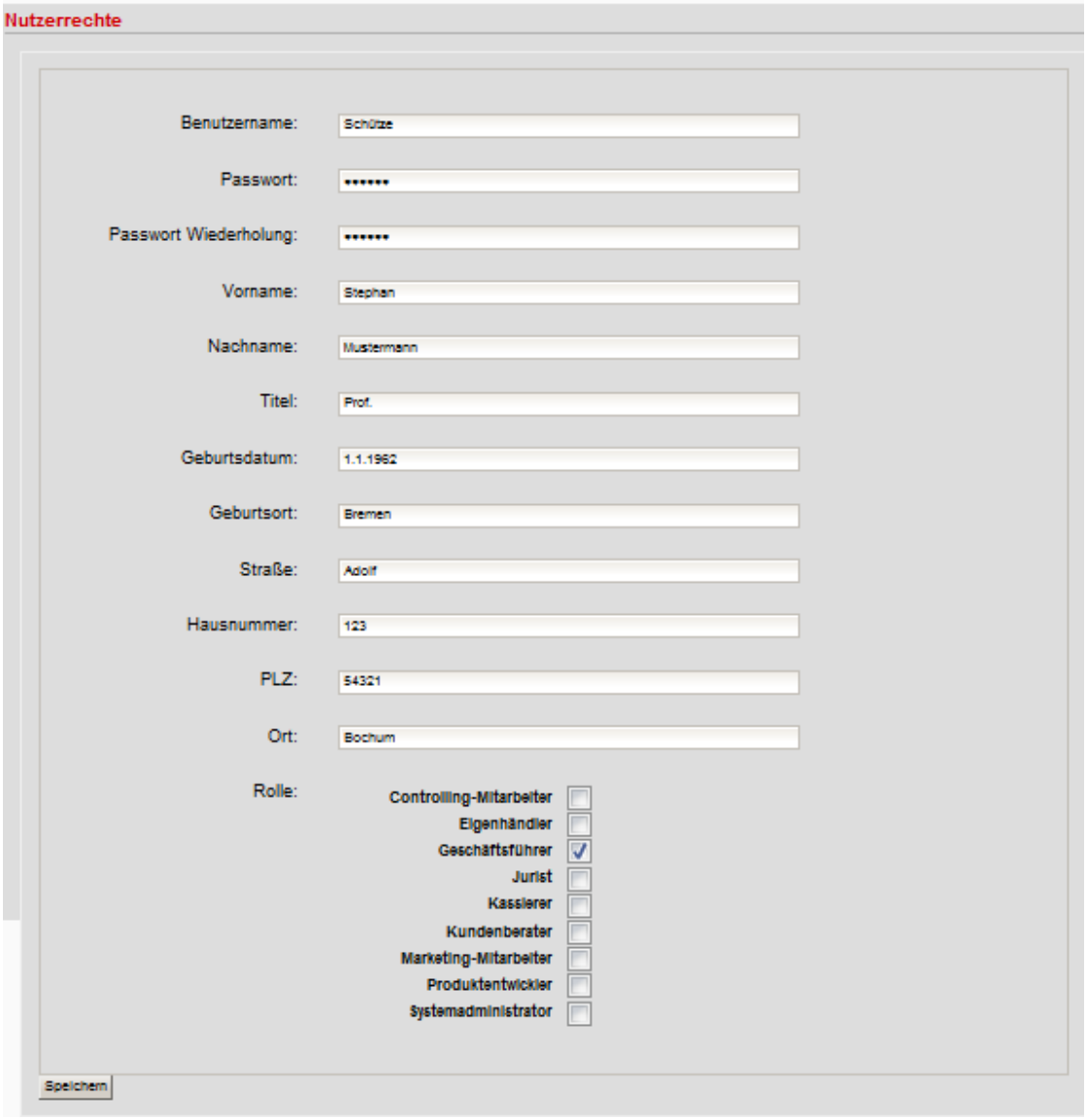


Abbildung A.49.: Persondaten von Mitarbeiter editieren

A.14.4. Log auswerten

Diese Funktion dient dazu, Systemlogs bei Bedarf katalogisch sortiert darzustellen (s. Abb. A.51). Hier haben Sie die Möglichkeit sich die Logs aus einem Bestimmten Zeitraum in verkürzter Form anzeigen zu lassen. Der Button *Exportieren* ermöglicht es Ihnen die aktuell angezeigten Logs in eine separate Log-Datei zu exportieren. Durch Auswählen eines Log-Eintrags und Betätigen des Buttons *Details anzeigen* wird der gesamte Log-Eintrag angezeigt.

Abbildung A.50.: Systemlog einsehen

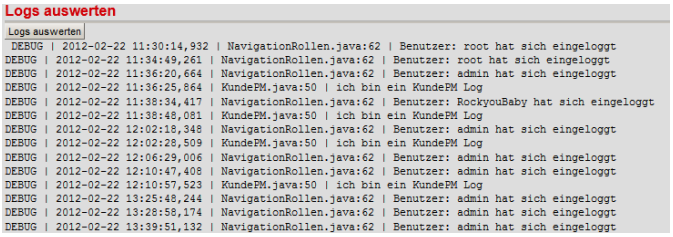


Abbildung A.51.: Logs auswerten

A.14.5. Backup erstellen

Durch Betätigen des Menüeintrags *Backup erstellen* wird ein Backup des Systems erstellt. Das Erstellen eines Backups benötigt Ressourcen und kann dadurch das System verlangsamen.

A.14.6. RSA Schlüssel verwalten

In der Ansicht *RSA Schlüssel verwalten* können Sie den RSA Schlüssel 1.5.4 des Banking-Systems ändern und neue RSA Schlüssel generieren (s. Abb. A.53) oder fremde Zertifikate importieren, denen die *Magnus Monetarus Bank*vertraut (s. Abb. A.54).

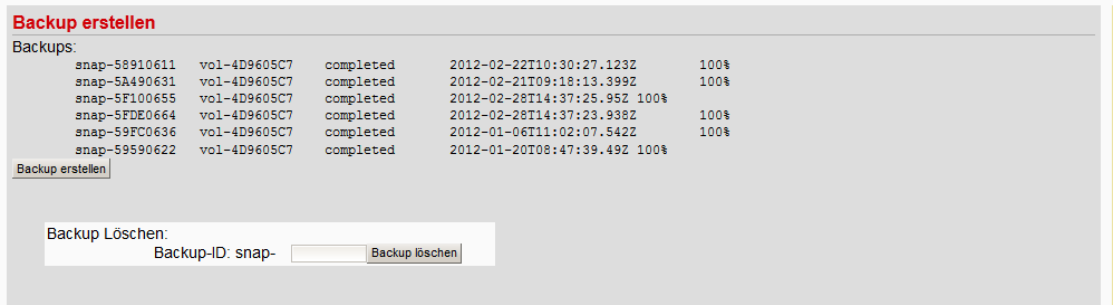


Abbildung A.52.: Backup erstellen

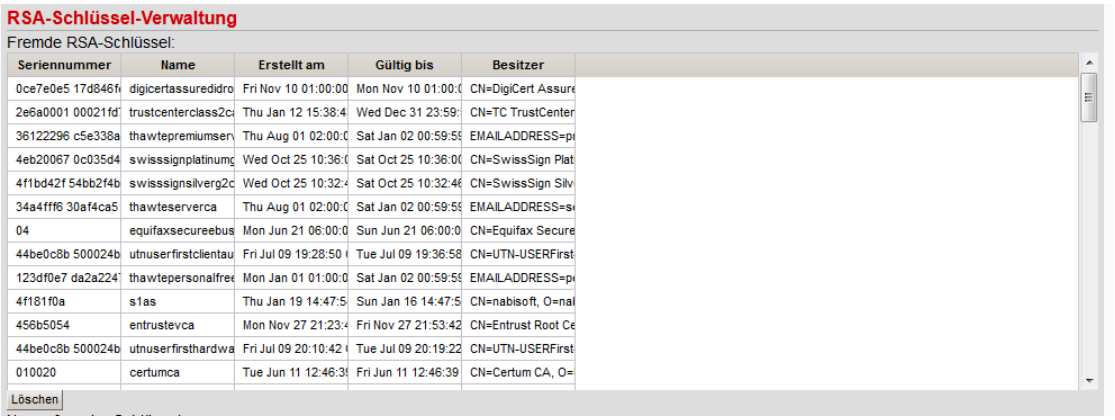


Abbildung A.53.: RSA-Schlüssel verwalten

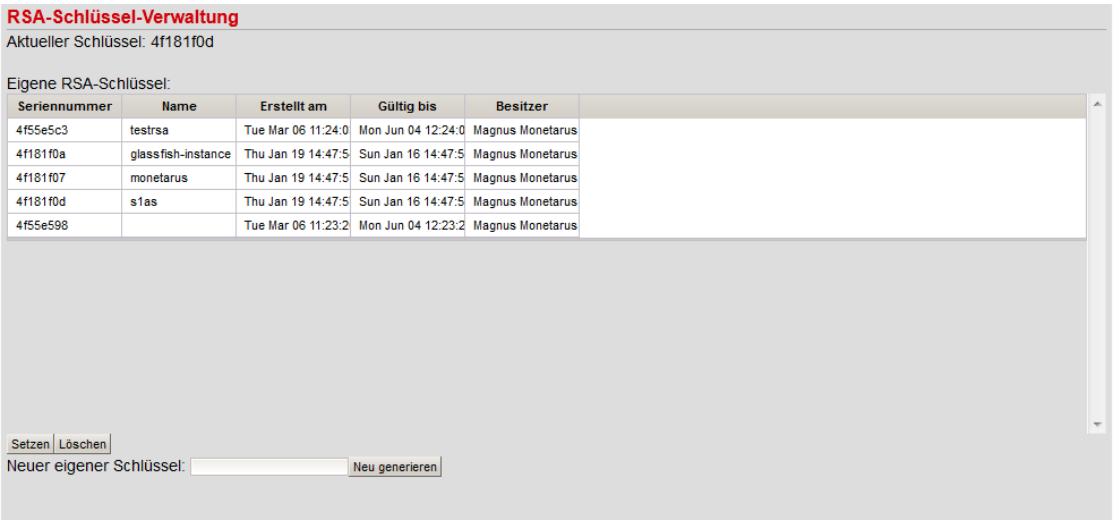


Abbildung A.54.: RSA-Schlüssel verwalten

A.14.7. Nachrichtenverkehr

Sehen Sie bitte den Abschnitt A.4.3.

A.14.8. Runterfahren

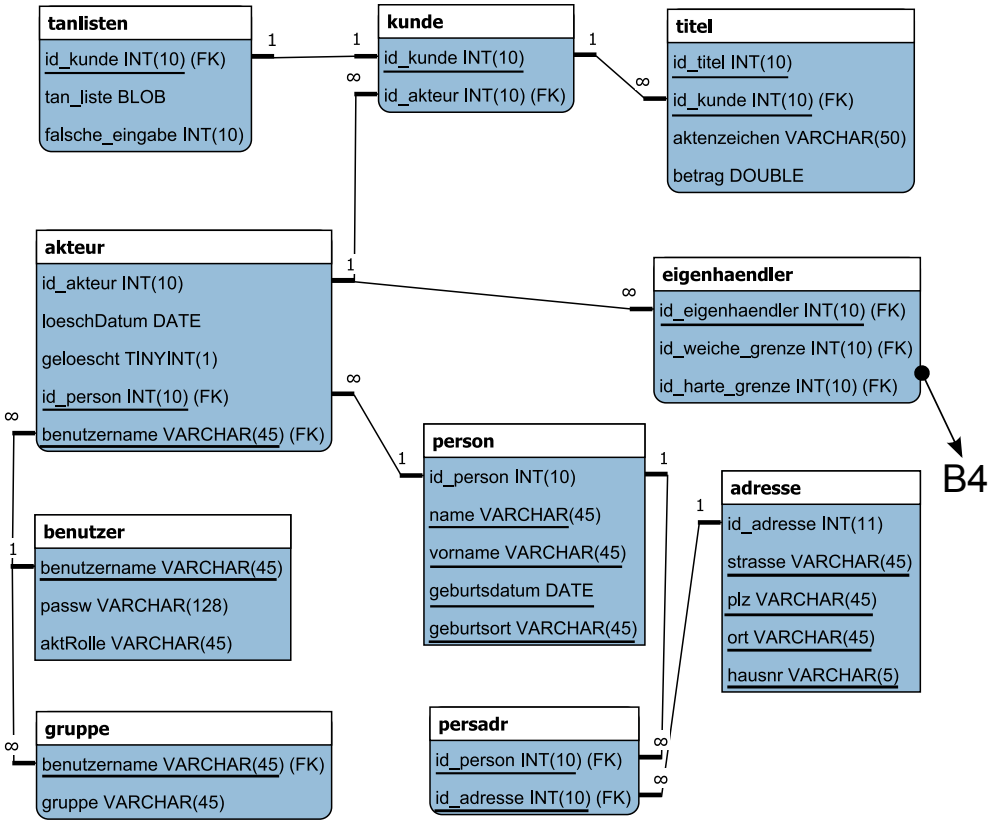
Durch den Menüeintrag *Runterfahren* kann der Server heruntergefahren werden.

B. Datenbanktabellen

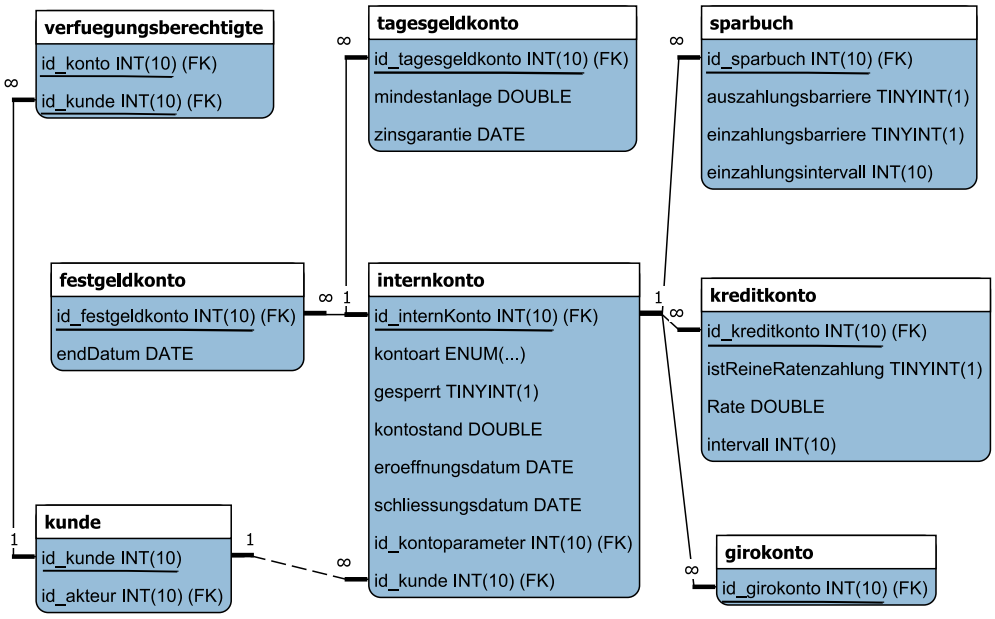
Folgende Tabellen werden in den beiden Datenbanken `banksystem` und `stammdaten` verwendet. Der Übersicht halber wurden sie nach unterschiedlichen Bereichen aufgeteilt, wobei einige Tabellen in mehreren Bereichen vorhanden sind und als Schnittstelle zwischen diesen Bereichen dienen sollen.

Die Syntax der Tabellen wird in Abschnitt 1.2.3 vorgestellt.

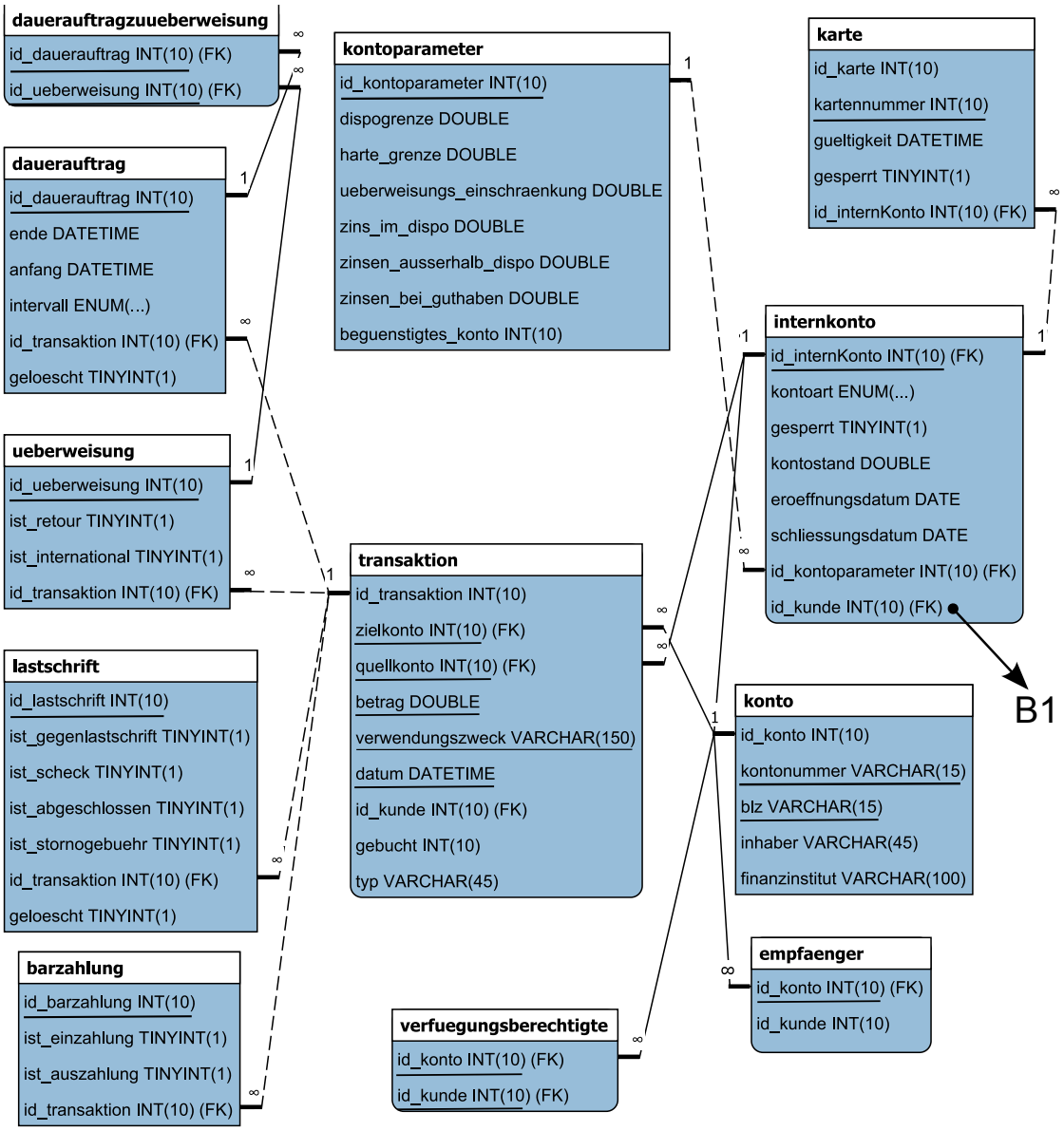
B.1. Personen



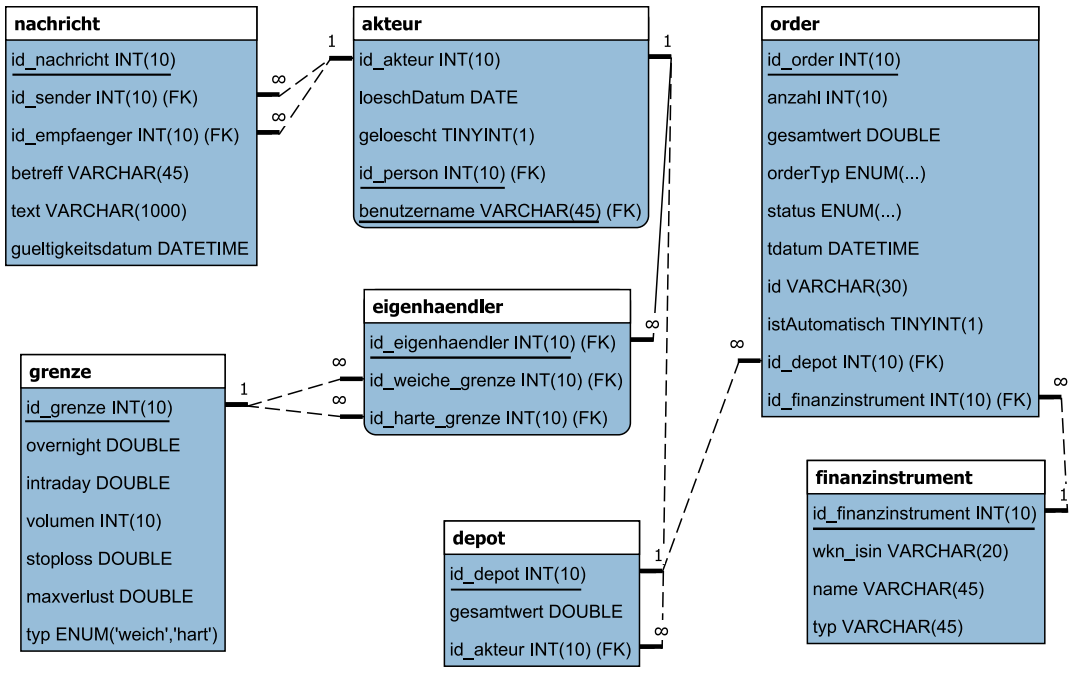
B.2. Interne Konten



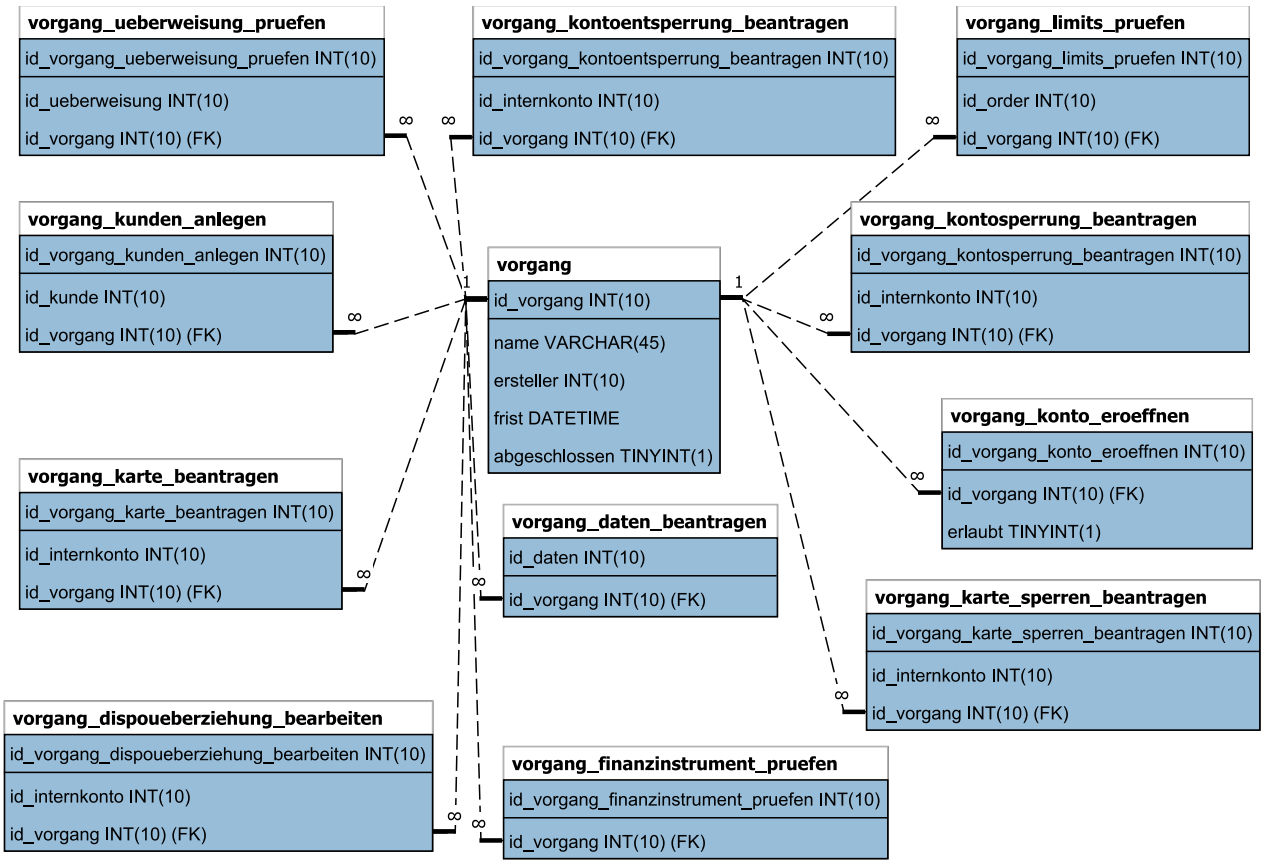
B.3. Transaktionen



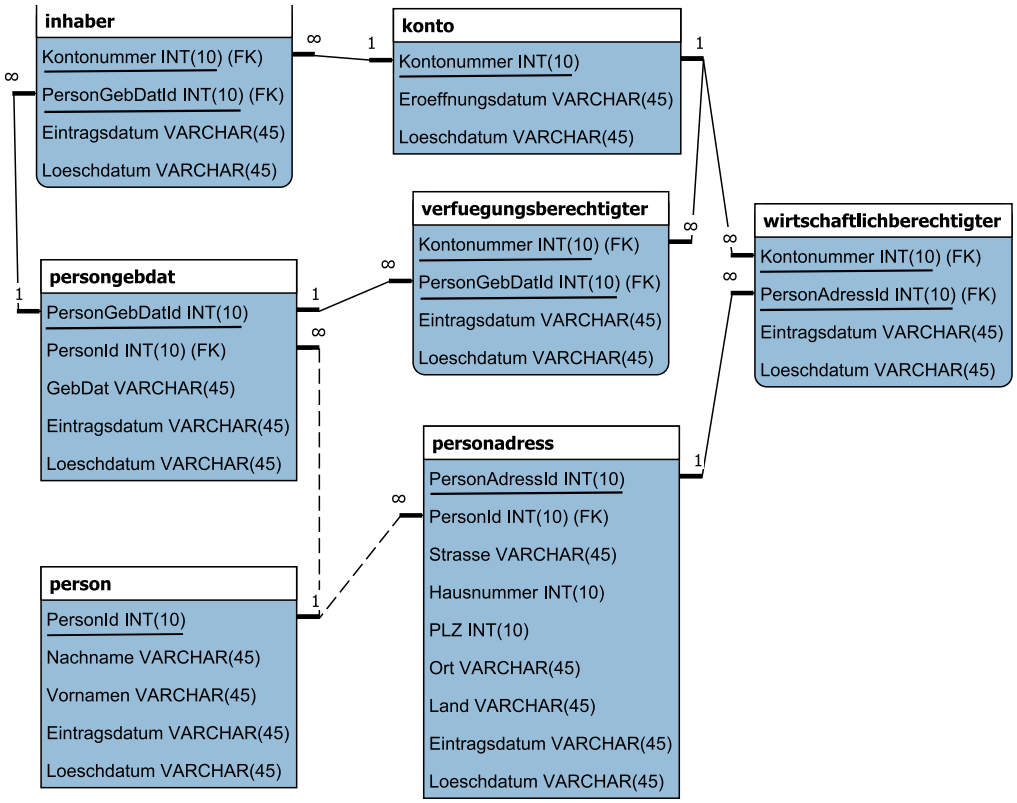
B.4. Börse und Nachrichten



B.5. Vorgänge


vorgang_ueberweisung_pruefen
id_vorgang_ueberweisung_pruefen INT(10)
id_ueberweisung INT(10)
id_vorgang INT(10) (FK)
vorgang_kontoentsperrung_beantragen
id_vorgang_kontoentsperrung_beantragen INT(10)
id_internkonto INT(10)
id_vorgang INT(10) (FK)
vorgang_limits_pruefen
id_vorgang_limits_pruefen INT(10)
id_order INT(10)
id_vorgang INT(10) (FK)
vorgang_kunden_anlegen
id_vorgang_kunden_anlegen INT(10)
id_kunde INT(10)
id_vorgang INT(10) (FK)
vorgang
id_vorgang INT(10)
name VARCHAR(45)
ersteller INT(10)
frist DATETIME
abgeschlossen TINYINT(1)
vorgang_kontosperrung_beantragen
id_vorgang_kontosperrung_beantragen INT(10)
id_internkonto INT(10)
id_vorgang INT(10) (FK)
vorgang_konto_eroeffnen
id_vorgang_konto_eroeffnen INT(10)
id_vorgang INT(10) (FK)
erlaubt TINYINT(1)
vorgang_karte_beantragen
id_vorgang_karte_beantragen INT(10)
id_internkonto INT(10)
id_vorgang INT(10) (FK)
vorgang_daten_beantragen
id_daten INT(10)
id_vorgang INT(10) (FK)
vorgang_karte_sperren_beantragen
id_vorgang_karte_sperren_beantragen INT(10)
id_internkonto INT(10)
id_vorgang INT(10) (FK)
vorgang_dispoueberziehung_bearbeiten
id_vorgang_dispoueberziehung_bearbeiten INT(10)
id_internkonto INT(10)
id_vorgang INT(10) (FK)
vorgang_finanzinstrument_pruefen
id_vorgang_finanzinstrument_pruefen INT(10)
id_vorgang INT(10) (FK)

B.6. Stammdaten



C. Security Policy

IT-Sicherheitsleitlinie

Magnus Monetarus Bank
Cloud-basiertes Internetbanking System

19. Dezember 2011

Inhaltsverzeichnis

1. Ziele **2**

2. Adressatenkreis **3**

3. Verantwortlichkeiten und Zuständigkeiten **3**

4. Verantwortlichkeit Dritter **4**

5. Sicherheitsmanagement **4**

6. Sicherheitsmaßnahmen **4**

7. Richtlinien für Mindeststandards **5**

8. Schulung und Sensibilisierung **6**

9. Prüfung **6**

10. Gesetze und Vorschriften **6**

11. Verstöße und Sanktionierung **6**

12. Inkrafttreten **7**

A. Glossar **7**

Literaturverzeichnis **8**

Dokumentenhistorie

Version	Datum	Editor	Änderung
1.0	12.09.2011	DS, TS, VS	Initiales Dokument
1.1	25.11.2011	TS, VS (PG555)	Zertifizierung referenziert
1.2	19.12.2011	TS (PG555)	Orthographische Korrekturen

Präambel

Informationen und die zu ihrer Übertragung und Verarbeitung eingesetzten Prozesse und IT-Systeme stellen grundlegende Werte für die Magnus Monetarus Bank dar. Alle wesentlichen strategischen und operativen Funktionen und Aufgaben werden durch Informationstechnologie (IT) maßgeblich unterstützt. Die Informationssicherheit ist ein zunehmend wichtiger Faktor geworden, weshalb insbesondere das Sicherheitsbewusstsein hinsichtlich der vorhandenen Informationen einer der entscheidenden Erfolgsfaktoren für die Bank ist.

Der Schutz dieser Werte vor Verlust, unberechtigtem Zugriff und vor unerlaubter Änderung ist unverzichtbar, um die Leistungsfähigkeit und Wettbewerbspositionen des Unternehmens, sowie das Vertrauen bei Geschäftspartnern und Kunden zu erhalten und zu verbessern.

1. Ziele

1.1 Das oberste Ziel der IT-Sicherheitsleitlinie der Magnus Monetarus Bank ist es, die Verfügbarkeit, Vertraulichkeit und Integrität der eigenen und der ihr von Kunden und Geschäftspartnern anvertrauten Informationen und Ressourcen zu schützen. Dabei ist die Nachvollziehbarkeit, Verbindlichkeit und Ordnungsmäßigkeit von Prozessen zu garantieren, um das Erreichen der Unternehmensziele zu gewährleisten und Schaden durch den Eintritt unerwünschter Ereignisse zu verhindern oder zu begrenzen.

1.2 Ein weiteres Ziel der Security Policy ist es, die Zertifizierbarkeit der Sicherheit der Systeme nach den Sicherheitskriterien des BSI zu ermöglichen. Aufgrund dessen wird die Security Policy nach der IT-Grundschutz-Vorgehensweise des BSI-Standards 100 erstellt. Das IT-Grundschutz-Zertifikat stellt einen anerkannten Standard für IT-Sicherheit dar [1].

1.3 Verspätete oder fehlerhafte Managemententscheidungen können weitreichende Folgen nach sich ziehen. Daher ist für das Management bei wichtigen Entscheidungen der Zugriff auf aktuelle Steuerungsdaten wichtig. Für diese Informationen ist ein hohes Sicherheitsniveau in Bezug auf Verfügbarkeit und Integrität sicherzustellen.

1.4 Die Aktualität der Börsendaten ist für die Eigenhändler essentiell. Es muss folglich sichergestellt werden, dass der Zugriff auf die Börsendaten schnell und zuverlässig verfügbar ist.

1.5 Die Verfügbarkeit der Kommunikation mit anderen Banken ist für die Ausübung der Geschäfte von höchster Bedeutung und ist in jedem Fall sicher zu stellen.

1.6 Der Zugriff auf das Internetbanking-System muss für den Kunden zu jeder Zeit möglich sein. Dies liegt im Interesse der Magnus Monetarus Bank, da dies für das Kern-

geschäft notwendig ist.

1.7 Ein Ausfall des Nachrichtensystems oder des Workflowsystems kann zu Verzögerungen im Arbeitsablauf führen, daher ist die Verfügbarkeit des Nachrichtensystems und des Workflowsystems sicherzustellen. Im Falle eines Ausfalls muss der Betrieb der Kerngeschäfte weiterhin möglich sein.

1.8 Die Datenschutzgesetze und die Interessen unserer Mitarbeiter verlangen eine Sicherstellung der Vertraulichkeit der Mitarbeiterdaten. Die Daten und die IT-Anwendungen der Personalabteilung werden daher einem hohen Vertraulichkeitsschutz unterzogen. Gleiches gilt für die Daten unserer Kunden und Geschäftspartner.

1.9 Die Nutzung des Internets zur Informationsbeschaffung und zur Kommunikation ist für uns selbstverständlich. E-Mail dient als Ersatz oder als Ergänzung von anderen Bürokommunikationswegen. Durch entsprechende Maßnahmen wird sichergestellt, dass die Risiken der Internetnutzung möglichst gering bleiben.

2. Adressatenkreis

Diese Sicherheitsleitlinie gilt für die Magnus Moneraturs Bank sowie für ihre sämtlichen Mitarbeiter. Außerdem muss die Leitlinie für alle Geschäftsprozesse, die durch das Bankingsystem durchgeführt und unterstützt werden, umgesetzt werden.

3. Verantwortlichkeiten und Zuständigkeiten

Verantwortlich für die Einhaltung dieser Sicherheitsleitlinie und damit für die Informationssicherheit ist neben der Geschäftsleitung jede Abteilung und jeder Mitarbeiter der Magnus Monetarus Bank. Jede Abteilung und jeder Mitarbeiter ist insbesondere für die Sicherheit und einen angemessenen Schutz ihrer/seiner Informationen entsprechend ihres Wertes und des Risikos für das betreffende Geschäfts- und technische Umfeld verantwortlich.

Vor diesem Hintergrund muss es für alle Informationen einen benannten Verantwortlichen geben. Insbesondere müssen folgende Verantwortliche benannt werden:

- *Datenschutzbeauftragter*

  Verantwortlich für die Einhaltung des Datenschutzes und die Weiterentwicklung der Datenschutzrichtlinie

- *IT-Sicherheitsbeauftragter*

  Verantwortlich für das Erreichen und die Einhaltung der Informationssicherheit, sowie die Erstellung und Weiterentwicklung der IT-Sicherheitsleitlinie

- *Vertreter der Rechtsabteilung*

Berät das Unternehmen bei der Erstellung und Weiterentwicklung von Sicherheits- und Datenschutzleitlinien.

4. Verantwortlichkeit Dritter

Sofern IT-Dienstleistungen an externe Stellen ausgelagert werden, werden von uns konkrete Sicherheitsanforderungen in den Service Level Agreements vorgegeben. Das Recht auf Kontrolle wird festgelegt. Für umfangreiche oder komplexe Outsourcing-Vorhaben erstellen wir ein detailliertes Sicherheitskonzept mit konkreten Maßnahmenvorgaben.

5. Sicherheitsmanagement

5.1 Zur Erreichung der Informationssicherheitsziele wurde eine Sicherheitsorganisation eingerichtet. Es ist ein IT-Sicherheitsbeauftragter benannt worden. Der IT-Sicherheitsbeauftragte berichtet in seiner Funktion direkt an den IT-Direktor, der Mitglied der Geschäftsführung ist.

5.2 Dem IT-Sicherheitsbeauftragten und den Administratoren werden von der Leitung ausreichende finanzielle und zeitliche Ressourcen zur Verfügung gestellt, um sich regelmäßig weiter zu bilden und zu informieren und die vom Management festgelegten Informationssicherheitsziele zu erreichen.

5.3 Die Administratoren und der IT-Sicherheitsbeauftragte sind durch die IT-Benutzer ausreichend in ihrer Arbeit zu unterstützen.

5.4 Der IT-Sicherheitsbeauftragte ist frühzeitig in alle Projekte einzubinden, um schon in der Planungsphase sicherheitsrelevante Aspekte zu berücksichtigen. Sofern personenbezogene Daten betroffen sind, gilt Gleiches für den Datenschutzbeauftragten.

5.5 Die IT-Benutzer haben sich in sicherheitsrelevanten Fragestellungen an die Anweisungen des IT Sicherheitsbeauftragten zu halten.

5.6 Es wurde ein Datenschutzbeauftragter bestellt. Der Datenschutzbeauftragte hat ein ausreichend bemessenes Zeitbudget für die Erfüllung seiner Pflichten zur Verfügung. Der Datenschutzbeauftragte ist angehalten, sich regelmäßig weiterzubilden.

6. Sicherheitsmaßnahmen

6.1 Für alle Verfahren, Informationen, IT-Anwendungen und IT-Systeme wird eine verantwortliche Person benannt, die den jeweiligen Schutzbedarf bestimmt und Zugriffsbe-

rechtigungen vergibt.

6.2 Für alle verantwortlichen Funktionen sind Vertretungen einzurichten. Es muss durch Unterweisungen und ausreichende Dokumentationen sichergestellt werden, dass Vertreter in der Lage sind, ihre Aufgaben zu erfüllen.

6.3 Gebäude und Räumlichkeiten werden durch ausreichende Zutrittskontrollen geschützt. Der Zugang zu IT-Systemen wird durch angemessene Zugangskontrollen und der Zugriff auf die Daten durch ein restriktives Berechtigungskonzept geschützt.

6.4 Datenverluste können nie vollkommen ausgeschlossen werden. Durch eine umfassende Datensicherung wird daher gewährleistet, dass der IT-Betrieb kurzfristig wiederaufgenommen werden kann, wenn Teile des operativen Datenbestandes verloren gehen oder offensichtlich fehlerhaft sind. Informationen werden einheitlich gekennzeichnet und so aufbewahrt, dass sie schnell auffindbar sind.

6.5 Um größere Schäden in Folge von Notfällen zu begrenzen bzw. diesen vorzubeugen, muss auf Sicherheitsvorfälle zügig und konsequent reagiert werden. Maßnahmen für den Notfall werden in einem separaten Notfallvorsorgekonzept zusammengestellt. Unser Ziel ist, auch bei einem Systemausfall kritische Geschäftsprozesse aufrecht zu erhalten und die Verfügbarkeit der ausgefallenen Systeme innerhalb einer tolerierbaren Zeitspanne wiederherzustellen.

7. Richtlinien für Mindeststandards

7.1 Die Grundsätze der IT-Sicherheit der Magnus Monetarus Bank werden in Richtlinien zur IT-Sicherheit und entsprechenden Standards und Anweisungen detailliert, konkretisiert und auf die Anforderungen der Prozesse, Informationen, Systeme und organisatorischen Einheiten des Unternehmens zugeschnitten. Durch die Aufstellung, Umsetzung und Einhaltung von Mindeststandards wird bei der Magnus Monetarus Bank eine Grundsicherheit hergestellt, auf deren Basis ein vertrauensvolles Arbeiten im Unternehmen und mit dem Kunden und dessen Gütern sichergestellt wird. Richtlinien [2] für Mindeststandards existieren für die folgenden Themengebiete:

- Internet- und E-Mail-Nutzung
- IT-Nutzung
- Outsourcing
- Archivierung
- Datensicherung
- Notfallvorsorge

7.2 Alle aus den Sicherheitsgrundsätzen abgeleiteten Regelwerke und Anweisungen werden in regelmäßigen Abständen auf ihre Aktualität, Korrektheit, Angemessenheit und Umsetzbarkeit überprüft und bei Bedarf angepasst oder durch Folgedokumente ersetzt.

Für die Koordination dieser Überprüfungen ist das Sicherheitsmanangement verantwortlich.

8. Schulung und Sensibilisierung

IT-Benutzer nehmen regelmäßig an Schulungen zur korrekten Nutzung der IT-Dienste und den hiermit verbundenen Sicherheitsmaßnahmen und ihrer Anwendung teil. Auch erfahrene IT-Benutzer nehmen an den Schulungen teil, um ihr Wissen aufzufrischen und zu ergänzen. Mitarbeiter, die neu eingestellt oder denen eine neue Aufgabe zugewiesen wurden, werden gründlich eingearbeitet und ausgebildet. Die Unternehmensleitung unterstützt dabei die bedarfsgerechte Fort- und Weiterbildung.

9. Prüfung

Die IT-Sicherheitsleitlinie sowie die auf ihr beruhenden Richtlinien werden durch das Sicherheitsmanagement ständig aktualisiert und regelmäßig auf ihre Wirksamkeit und Einhaltung überprüft. Die auf diesen Regelungen beruhenden Maßnahmen werden regelmäßig daraufhin untersucht, ob sie den betroffenen Mitarbeitern bekannt sind, ob sie (noch) umsetzbar und in den (aktuellen) Betriebsablauf integrierbar sind.

Die Leitung unterstützt die ständige Verbesserung des Sicherheitsniveaus. Mitarbeiter sind angehalten, mögliche Verbesserungen oder Schwachstellen an die entsprechenden Stellen weiterzugeben.

Durch eine kontinuierliche Revision der Regelungen und deren Einhaltung wird das angestrebte Sicherheits- und Datenschutzniveau sichergestellt. Abweichungen werden mit dem Ziel analysiert, die Sicherheitssituation zu verbessern und ständig auf dem aktuellen Stand der IT-Sicherheitstechnik zu halten.

10. Gesetze und Vorschriften

Beim Einsatz der IT sind einschlägige Gesetze, Vorschriften sowie vertragliche und aufsichtsrechtliche Verpflichtungen und Regelungen einzuhalten.

11. Verstöße und Sanktionierung

11.1 Als Verstöße werden Handlungen gegen das Regelwerk der IT-Sicherheit verstanden. Dazu zählen insbesondere:

- die Kompromittierung der Sicherheit von Informationen,

- der unberechtigte Zugriff auf Informationen,
- die unberechtigte Änderung, Nutzung und/oder Veröffentlichung von Informationen.

11.2 Verstöße gegen die IT-Sicherheit werden nach den geltenden Regelungen und gesetzlichen Bestimmungen geahndet.

12. Inkrafttreten

Diese Richtlinie wird von der Unternehmensleitung der Magnus Monetarus Bank verabschiedet. Sie tritt am Tag nach ihrer Bekanntmachung durch den IT-Sicherheitsbeauftragten in Kraft.

A. Glossar

- **Administrator:** Ein Administrator ist zuständig für Einrichtung, Betrieb, Überwachung und Wartung eines IT-Systems.
- **Datenschutzbeauftragter:** Ein Datenschutzbeauftragter ist eine von der Unternehmensleitung bestellte Person, die für den datenschutzrechtlich korrekten bzw. gesetzeskonformen Umgang mit personenbezogenen Daten im Unternehmen verantwortlich ist.
- **IT-Anwendung:** Software, die angewendet wird, um eine nützliche oder gewünschte Funktionalität zu erfüllen
- **IT-Direktor:** Mitglied der Geschäftsführung, das für die Organisation der IT-Systeme verwantwortlich ist
- **IT-Sicherheitsbeauftragter:** Ein IT-Sicherheitsbeauftragter ist eine von der Unternehmensleitung ernannte Person, die im Auftrag der Leitungsebene die Aufgabe Informationssicherheit koordiniert und innerhalb des Unternehmens vorantreibt.
- **IT-System:** Jegliche Art informationsverarbeitender elektronischer Systeme, z.B.: Computer, Serversysteme, Datenbanksysteme
- **Outsourcing:** Abgabe von Unternehmensaufgaben und -strukturen an Drittunternehmen
- **Service Level Agreement (SLA):** Bezeichnet einen Vertrag bzw. die Schnittstelle, zwischen Auftraggeber und Dienstleistung für wiederkehrende Dienstleistungen
- **Sicherheitsmanagement:** Bezeichnet die Planung, Steuerung und Kontrolle der Sicherheit in einem Unternehmen

- **Verfügbarkeit:** Eigenschaft, die die korrekte Nutzung einer Funktion des Systems garantiert
- **Verlässlichkeit:** Eigenschaft einer Funktion, die die korrekte Ausführung garantiert
- **Vertraulichkeit:** Eigenschaft, die die kontrollierte Nutzung (Erhebung / Bearbeitung) einer Datei des Systems durch vorgesehene Akteure garantiert. Dies impliziert auch die Nicht-Nutzung durch unberechtigte Akteure
- **Vertreter der Rechtsabteilung:** Berater des Unternehmens bei der Erstellung und Weiterentwicklung von Sicherheits- und Datenschutzrichtlinien

Literatur

[1] BSI. https://www.bsi.bund.de/DE/Themen/ZertifizierungundAnerkennung/zertifizierungundanerk 2011. [Online; accessed 18-November-2011].

[2] Magnus Monetarus Bank. Bausteine, gefährdungen und maßnahmen des bsi-grundschutzkataloges. Technical report, 2011.

Bausteine, Gefährdungen und Maßnahmen des BSI-Grundschutzkataloges

Magnus Monetarus Bank
Cloud-basiertes Internetbanking System

25. November 2011

Inhaltsverzeichnis

1 Bausteine für Übergreifende Aspekte 2

1.1 Sicherheitsmanagement (B 1.0) . . . 2
1.1.1 Beschreibung . . . 2
1.1.2 Gefährdungslage . . . 2
1.1.3 Maßnahmen . . . 3
1.2 Organisation (B 1.1) . . . 4
1.2.1 Beschreibung . . . 4
1.2.2 Gefährdungslage . . . 4
1.2.3 Maßnahmen . . . 5
1.3 Personal (B 1.2) . . . 6
1.3.1 Beschreibung . . . 6
1.3.2 Gefährdungslage . . . 6
1.3.3 Maßnahmen . . . 7
1.4 Notfallmanagement (B 1.3) . . . 8
1.5 Datensicherungskonzept (B 1.4) . . . 8
1.5.1 Beschreibung . . . 8
1.5.2 Gefährdungslage . . . 9
1.5.3 Maßnahmen . . . 9
1.6 Datenschutz (B 1.5) . . . 10
1.6.1 Beschreibung . . . 10
1.6.2 Gefährdungslage . . . 10
1.6.3 Maßnahmen . . . 11
1.7 Schutz vor Schadprogrammen (B 1.6) . . . 12
1.7.1 Beschreibung . . . 12
1.7.2 Gefährdungslage . . . 12
1.7.3 Maßnahmen . . . 13
1.8 Kryptokonzept (B 1.7) . . . 15
1.8.1 Beschreibung . . . 15
1.8.2 Gefährdungslage . . . 15
1.8.3 Maßnahmen . . . 16
1.9 Behandlung von Sicherheitsvorfällen (B 1.8) . . . 18
1.10 Hard- und Software-Management (B 1.9) . . . 18
1.11 Standardsoftware (B 1.10) . . . 18
1.12 Outsourcing (B 1.11) . . . 18
1.12.1 Beschreibung . . . 18
1.12.2 Gefährdungslage . . . 20
1.12.3 Maßnahmen . . . 21
1.13 Archivierung (B 1.12) . . . 22
1.13.1 Beschreibung . . . 22
1.13.2 Gefährdungslage . . . 24
1.13.3 Maßnahmen . . . 25
1.14 Sensibilisierung und Schulung zur Informationssicherheit (B 1.13) . . . 27

1.15 Patch- und Änderungsmanagement (B 1.14) 27
1.16 Löschen und Vernichten von Daten (B 1.15) 27
1.17 Anforderungsmanagement (B 1.16) 27

2 Bausteine für Infrastruktur 28
2.1 Gebäude (B 2.1) 28
2.1.1 Beschreibung 28
2.1.2 Gefährdungslage 28
2.1.3 Maßnahmen 29
2.2 Serverraum (B 2.4) 30
2.2.1 Beschreibung 30
2.2.2 Gefährdungslage 30
2.2.3 Maßnahmen 31
2.3 Rechenzentrum (B 2.9) 32
2.3.1 Beschreibung 32
2.3.2 Gefährdungslage 34
2.3.3 Maßnahmen 35

3 Bausteine für IT-Systeme 37
3.1 Allgemeiner Server (B 3.101) 37
3.1.1 Beschreibung 37
3.1.2 Gefährdungslage 37
3.1.3 Maßnahmen 38
3.2 Server unter Unix (B 3.102) 40
3.2.1 Beschreibung 40
3.2.2 Gefährdungslage 41
3.2.3 Maßnahmen 42
3.3 Sicherheitsgateway (Firewall) (B 3.301) 43
3.3.1 Beschreibung 43
3.3.2 Gefährdungslage 44
3.3.3 Maßnahmen 45
3.4 Router und Switches (B 3.302) 47
3.4.1 Beschreibung 47
3.4.2 Gefährdungslage 47
3.4.3 Maßnahmen 48

4 Bausteine für Netze 51
4.1 Heterogene Netze (B 4.1) 51
4.1.1 Beschreibung 51
4.1.2 Gefährdungslage 51
4.1.3 Maßnahmen 52
4.2 Netz- und Systemmanagement (B 4.2) 53
4.2.1 Beschreibung 53
4.2.2 Gefährdungslage 54

4.2.3 Maßnahmen 55

5 Bausteine für Anwendungen **57**
5.1 Webserver (B 5.4) 57
5.1.1 Beschreibung 57
5.1.2 Gefährdungslage 57
5.1.3 Maßnahmen 58
5.2 Datenbanken (B 5.7) 60
5.2.1 Beschreibung 60
5.2.2 Gefährdungslage 61
5.2.3 Maßnahmen 62

Literaturverzeichnis **64**

Dokumentenhistorie

Version	Datum	Editor	Änderung
0.1	12.09.2011	TS, VS (PG555)	Initiales Dokument
0.2	25.11.2011	TS, VS (PG555)	Präambel hinzugefügt

Präambel

Aufbau des Dokuments

Das Dokument enthält die für die Bausteine des BSI-Grundschutzkataloges [1] Gefährdungen und Maßnahmen, die für den Betrieb und die Weiterentwicklung der Bankingsoftware relevant sind. Alle vom BSI aufgeführten Bausteine werden hier in einem Dokument zusammengefasst.

Ziel des Dokuments

Das Dokument führt die Gefährdungen und Maßnahmen auf, die für den Betrieb und die Weiterentwicklung der Bankingsoftware relevant sind, um die IT-Sicherheitsleitlinie zu vervollständigen. Weiter soll durch dieses Dokument die Zertifizierung der Bankingsoftware mit dem BSI-Grundschutz-Zertifikat (BS 7799-2) vorbereiten, die eine Zertifizierung nach ISO 27001 einschließt.

Für die Bausteine, deren Gefährdungslage durch die PG555 nicht festgestellt werden kann, werden in dieser Version (0.2) keine Gefahren und auch keine Maßnahmen aufgeführt.

1 Bausteine für Übergreifende Aspekte

1.1 Sicherheitsmanagement (B 1.0)

1.1.1 Beschreibung

Die sichere Verarbeitung von Informationen ist heutzutage für nahezu alle Unternehmen und Behörden von existenzieller Bedeutung. Dabei können Informationen sowohl auf Papier, in Rechnern oder auch in Köpfen gespeichert sein. Für den Schutz der Informationen reicht es nicht aus, nur technische Sicherheitslösungen einzusetzen. Ein angemessenes Sicherheitsniveau kann nur durch geplantes und organisiertes Vorgehen aller Beteiligten erreicht und aufrechterhalten werden. Voraussetzung für die sinnvolle Umsetzung und Erfolgskontrolle von Sicherheitsmaßnahmen ist eine systematische Vorgehensweise. Diese Planungs-, Lenkungs- und Kontrollaufgabe wird als Informationssicherheitsmanagement oder auch kurz als IS-Management bezeichnet.

Der Begriff Informationssicherheit ist umfassender als der Begriff IT-Sicherheit und wird aufgrund dessen zunehmend verwendet. Da aber in der Literatur noch überwiegend der Begriff *IT-Sicherheit* zu finden ist, wird er auch in dieser sowie in anderen Publikationen des IT-Grundschutzes weiterhin verwendet, allerdings werden die Texte sukzessive stärker auf die Betrachtung von Informationssicherheit ausgerichtet.

Ein funktionierendes Sicherheitsmanagement muss in die existierenden Managementstrukturen einer jeden Institution eingebettet werden. Daher ist es praktisch nicht möglich, eine für jede Institution unmittelbar anwendbare Organisationsstruktur für das Sicherheitsmanagement anzugeben. Vielmehr werden häufig Anpassungen an spezifische Gegebenheiten erforderlich sein.

Dieser Baustein soll aufzeigen, wie ein funktionierendes Informationssicherheitsmanagement eingerichtet und im laufenden Betrieb weiterentwickelt werden kann. Er beschreibt dazu sinnvolle Schritte eines systematischen Sicherheitsprozesses und gibt Anleitungen zur Erstellung eines umfassenden Sicherheitskonzeptes. Der Baustein baut auf dem BSI-Standard 100-1 Managementsysteme für Informationssicherheit und BSI-Standard 100-2 Vorgehensweise nach IT-Grundschutz auf und fasst die wichtigsten Aspekte zum Sicherheitsmanagement hieraus zusammen.

1.1.2 Gefährdungslage

Gefährdungen im Umfeld des Sicherheitsmanagements können vielfältiger Natur sein. Stellvertretend für diese Vielzahl der Gefährdungen werden in diesem Baustein die folgenden typischen Gefährdungen betrachtet:

Organisatorische Mängel

G 2.66	Unzureichendes Sicherheitsmanagement

G 2.105	Verstoß gegen gesetzliche Regelungen und vertragliche Vereinbarungen
G 2.106	Störung der Geschäftsabläufe aufgrund von Sicherheitsvorfällen
G 2.107	Unwirtschaftlicher Umgang mit Ressourcen durch unzureichendes Sicherheitsmanagement

1.1.3 Maßnahmen

Im Rahmen des Sicherheitsmanagements sind eine Reihe von Maßnahmen umzusetzen, beginnend mit der Konzeption über den Aufbau geeigneter Organisationsstrukturen bis hin zur regelmäßigen Revision. Die Schritte, die dabei zu durchlaufen sind, sowie die Maßnahmen, die in den jeweiligen Schritten beachtet werden sollten, sind im Folgenden aufgeführt. Einer der Grundpfeiler zur Erreichung eines angemessenen Sicherheitsniveaus ist, dass die Leitungsebene hinter den Sicherheitszielen steht und sich ihrer Verantwortung für Informationssicherheit bewusst ist. Die Leitungsebene muss den Sicherheitsprozess initiieren, steuern und kontrollieren, damit dieser in der Institution auch in allen Bereichen umgesetzt wird (siehe M 2.336 Übernahme der Gesamtverantwortung für Informationssicherheit durch die Leitungsebene).

Informationssicherheit muss in allen Bereichen der Institution gelebt werden (siehe M 2.337 Integration der Informationssicherheit in organisationsweite Abläufe und Prozesse). Dazu gehört neben der Erarbeitung eines Sicherheitskonzepts (siehe M 2.195 Erstellung eines Sicherheitskonzepts) auch die Integration der Mitarbeiter in den Sicherheitsprozess (siehe M 2.197 Integration der Mitarbeiter in den Sicherheitsprozess) sowie die Erstellung von zielgruppengerechten Sicherheitsrichtlinien (siehe M 2.338 Erstellung von zielgruppengerechten Sicherheitsrichtlinien).

Nachfolgend wird das Maßnahmenbündel für den Bereich *Sicherheitsmanagement* vorgestellt.

Planung und Konzeption

M 2.192	(A)	Erstellung einer Leitlinie zur Informationssicherheit
M 2.335	(A)	Festlegung der Sicherheitsziele und -strategie
M 2.336	(A)	Übernahme der Gesamtverantwortung für Informationssicherheit durch die Leitungsebene

Umsetzung

M 2.193	(A)	Aufbau einer geeigneten Organisationsstruktur für Informationssicherheit
M 2.195	(A)	Erstellung eines Sicherheitskonzepts

M 2.197	(A)	Integration der Mitarbeiter in den Sicherheitsprozess
M 2.337	(A)	Integration der Informationssicherheit in organisationsweite Abläufe und Prozesse
M 2.338	(Z)	Erstellung von zielgruppengerechten Sicherheitsrichtlinien
M 2.339	(Z)	Wirtschaftlicher Einsatz von Ressourcen für Informationssicherheit

Betrieb

M 2.199	(A)	Aufrechterhaltung der Informationssicherheit
M 2.200	(C)	Management-Berichte zur Informationssicherheit
M 2.201	(C)	Dokumentation des Sicherheitsprozesses

Notfallvorsorge

M 6.16	(Z)	Abschließen von Versicherungen

1.2 Organisation (B 1.1)

1.2.1 Beschreibung

In diesem Baustein werden allgemeine und übergreifende Maßnahmen im Organisationsbereich aufgeführt, die als organisatorische Standardmaßnahmen zur Erreichung eines Mindestschutzniveaus erforderlich sind. Spezielle Maßnahmen organisatorischer Art, die in unmittelbarem Zusammenhang mit anderen Maßnahmen stehen (z. B. LAN-Administration), werden in den entsprechenden Bausteinen aufgeführt.

1.2.2 Gefährdungslage

In diesem Baustein werden für den IT-Grundschutz die folgenden typischen Gefährdungen betrachtet:

Organisatorische Mängel

G 2.1	Fehlende oder unzureichende Regelungen
G 2.2	Unzureichende Kenntnis über Regelungen
G 2.3	Fehlende, ungeeignete, inkompatible Betriebsmittel
G 2.5	Fehlende oder unzureichende Wartung
G 2.6	Unbefugter Zutritt zu schutzbedürftigen Räumen
G 2.7	Unerlaubte Ausübung von Rechten

G 2.8	Unkontrollierter Einsatz von Betriebsmitteln

Menschliche Fehlhandlungen

G 3.6	Gefährdung durch Reinigungs- oder Fremdpersonal

Vorsätzliche Handlungen

G 5.1	Manipulation oder Zerstörung von Geräten oder Zubehör
G 5.2	Manipulation an Informationen oder Software
G 5.3	Unbefugtes Eindringen in ein Gebäude
G 5.4	Diebstahl
G 5.5	Vandalismus
G 5.6	Anschlag
G 5.16	Manipulation bei Wartungsarbeiten
G 5.68	Unberechtigter Zugang zu den aktiven Netzkomponenten
G 5.102	Sabotage

1.2.3 Maßnahmen

Ein Mindestschutzniveau kann in einer Institution nur erreicht werden, wenn übergreifende Regelungen zur Informationssicherheit verbindlich festgelegt werden. Hierzu sind eine Reihe von Maßnahmen umzusetzen, beginnend mit Festlegung und Zuweisung von verantwortlichen Personen für einzelne Objekte (z. B. Informationen, Geschäftsprozesse, Anwendungen, IT-Komponenten) über entsprechende organisatorische Handlungsanweisungen bis hin zur Behandlung von schützenswerten Betriebsmitteln. Die Schritte, die dabei im Sinne eines kontinuierlichen Informationssicherheitsprozesses durchlaufen werden sollten, sowie die Maßnahmen, die in den jeweiligen Schritten beachtet werden sollten, sind im Folgenden aufgeführt.

Planung und Konzeption

M 2.1	(A)	Festlegung von Verantwortlichkeiten und Regelungen
M 2.2	(C)	Betriebsmittelverwaltung
M 2.4	(B)	Regelungen für Wartungs- und Reparaturarbeiten
M 2.5	(A)	Aufgabenverteilung und Funktionstrennung
M 2.40	(A)	Rechtzeitige Beteiligung des Personal-/Betriebsrates
M 2.225	(B)	Zuweisung der Verantwortung für Informationen, Anwendungen und IT-Komponenten
M 2.393	(A)	Regelung des Informationsaustausches

Betrieb

M 2.6	(A)	Vergabe von Zutrittsberechtigungen
M 2.7	(A)	Vergabe von Zugangsberechtigungen
M 2.8	(A)	Vergabe von Zugriffsrechten
M 2.16	(B)	Beaufsichtigung oder Begleitung von Fremdpersonen
M 2.18	(Z)	Kontrollgänge
M 2.37	(C)	Der aufgeräumte Arbeitsplatz
M 2.39	(B)	Reaktion auf Verletzungen der Sicherheitsvorgaben
M 2.177	(Z)	Sicherheit bei Umzügen
M 5.33	(B)	Absicherung von Fernwartung

Aussonderung

M 2.13	(A)	Ordnungsgemäße Entsorgung von schützenswerten Betriebsmitteln

1.3 Personal (B 1.2)

1.3.1 Beschreibung

In diesem Baustein werden die übergeordneten IT-Grundschutz-Maßnahmen erläutert, die im Bereich Personalwesen standardmäßig durchgeführt werden sollten. Beginnend mit der Einstellung von Mitarbeitern bis hin zu deren Weggang ist eine Vielzahl von Maßnahmen erforderlich. Auch für den Umgang mit Externen, wie z. B. Besuchern oder Wartungstechnikern, müssen angemessene Sicherheitsmaßnahmen vorhanden sein. Personelle Empfehlungen, die an eine bestimmte Funktion gebunden sind, wie z. B. die Ernennung des Systemadministrators eines LAN, werden in den Bausteinen angeführt, die sich mit dem jeweiligen Themengebiet beschäftigen.

1.3.2 Gefährdungslage

In diesem Baustein werden für den IT-Grundschutz die folgenden typischen Gefährdungen betrachtet:

Höhere Gewalt

G 1.1	Personalausfall
G 1.2	Ausfall von IT-Systemen

Organisatorische Mängel

G 2.2	Unzureichende Kenntnis über Regelungen
G 2.7	Unerlaubte Ausübung von Rechten

Menschliche Fehlhandlungen

G 3.1	Vertraulichkeits- oder Integritätsverlust von Daten durch Fehlverhalten
G 3.2	Fahrlässige Zerstörung von Gerät oder Daten
G 3.3	Nichtbeachtung von Sicherheitsmaßnahmen
G 3.8	Fehlerhafte Nutzung von IT-Systemen
G 3.9	Fehlerhafte Administration von IT-Systemen
G 3.36	Fehlinterpretation von Ereignissen
G 3.37	Unproduktive Suchzeiten
G 3.43	Ungeeigneter Umgang mit Passwörtern
G 3.44	Sorglosigkeit im Umgang mit Informationen

Vorsätzliche Handlungen

G 5.1	Manipulation oder Zerstörung von Geräten oder Zubehör
G 5.2	Manipulation an Informationen oder Software
G 5.20	Missbrauch von Administratorrechten
G 5.23	Schadprogramme
G 5.42	Social Engineering
G 5.80	Hoax
G 5.104	Ausspähen von Informationen

1.3.3 Maßnahmen

Für das in einem Unternehmen oder einer Behörde tätige Personal sind eine Reihe von Maßnahmen umzusetzen, beginnend mit einer geregelten Einarbeitung neuer Mitarbeiter, über Schulungen, bis hin zu einem geregelten Ausscheiden eines Mitarbeiters. Die Schritte, die dabei durchlaufen werden sollten, sowie die Maßnahmen, die in den jeweiligen Schritten beachtet werden sollten, sind im Folgenden aufgeführt.

Planung und Konzeption

M 2.226	(A)	Regelungen für den Einsatz von Fremdpersonal
M 3.51	(Z)	Geeignetes Konzept für Personaleinsatz und -qualifizierung

Beschaffung

M 3.50	(Z)	Auswahl von Personal

Umsetzung

M 3.1	(A)	Geregelte Einarbeitung/Einweisung neuer Mitarbeiter
M 3.10	(A)	Auswahl eines vertrauenswürdigen Administrators und Vertreters
M 3.33	(Z)	Sicherheitsüberprüfung von Mitarbeitern
M 3.55	(C)	Vertraulichkeitsvereinbarungen

Betrieb

M 3.3	(A)	Vertretungsregelungen
M 3.4	(A)	Schulung vor Programmnutzung
M 3.5	(A)	Schulung zu Sicherheitsmaßnahmen
M 3.7	(Z)	Anlaufstelle bei persönlichen Problemen
M 3.8	(Z)	Vermeidung von Störungen des Betriebsklimas
M 3.11	(A)	Schulung des Wartungs- und Administrationspersonals

Aussonderung

M 3.6	(A)	Geregelte Verfahrensweise beim Ausscheiden von Mitarbeitern

1.4 Notfallmanagement (B 1.3)

Keine ausreichende Einschätzung möglich.

1.5 Datensicherungskonzept (B 1.4)

1.5.1 Beschreibung

Durch technisches Versagen, versehentliches Löschen oder durch Manipulation können gespeicherte Daten unbrauchbar werden bzw. verloren gehen. Eine Datensicherung soll gewährleisten, dass durch einen redundanten Datenbestand der IT-Betrieb kurzfristig wiederaufgenommen werden kann, wenn Teile des operativen Datenbestandes verloren gehen.

Die Konzeption einer angemessenen und funktionstüchtigen Datensicherung bedarf allerdings aufgrund der Komplexität einer geordneten Vorgehensweise. In diesem Baustein

wird ein Weg beschrieben, wie für ein IT-System ein Datensicherungskonzept erstellt werden kann.

1.5.2 Gefährdungslage

Für die mittels eines Datensicherungskonzepts zu schützenden Daten wird für den IT-Grundschutz folgende typische Gefährdung angenommen:

Technisches Versagen

G 4.13	Verlust gespeicherter Daten

1.5.3 Maßnahmen

Um eine effektive Datensicherung einzurichten, sind eine Reihe von Schritten zu durchlaufen. Diese sind in der Maßnahme M 6.33 Entwicklung eines Datensicherungskonzepts beschrieben und werden durch die dort aufgeführten Maßnahmen erläutert. Daher sollte mit der Umsetzung der Maßnahme M 6.33 begonnen werden.

Nachfolgend wird das Maßnahmenbündel für den Bereich "Datensicherungskonzept" vorgestellt, das vor allem für größere IT-Systeme oder IT-Systeme mit großem Datenvolumen sinnvoll ist. Die Bearbeitung der Maßnahmen sollte in der angegebenen Reihenfolge geschehen, um systematisch ein Datensicherungskonzept zu erarbeiten.

Planung und Konzeption

M 6.33	(B)	Entwicklung eines Datensicherungskonzepts
M 6.34	(B)	Erhebung der Einflussfaktoren der Datensicherung
M 6.35	(B)	Festlegung der Verfahrensweise für die Datensicherung
M 6.36	(A)	Festlegung des Minimaldatensicherungskonzeptes

Beschaffung

M 2.137	(A)	Beschaffung eines geeigneten Datensicherungssystems

Umsetzung

M 2.41	(A)	Verpflichtung der Mitarbeiter zur Datensicherung
M 6.21	(C)	Sicherungskopie der eingesetzten Software
M 6.37	(A)	Dokumentation der Datensicherung

Betrieb

M 6.20	(A)	Geeignete Aufbewahrung der Backup-Datenträger
M 6.22	(A)	Sporadische Überprüfung auf Wiederherstellbarkeit von Datensicherungen
M 6.32	(A)	Regelmäßige Datensicherung

Notfallvorsorge

M 6.41	(A)	Übungen zur Datenrekonstruktion

1.6 Datenschutz (B 1.5)

1.6.1 Beschreibung

Aufgabe des Datenschutzes ist es, den Einzelnen davor zu schützen, dass er durch den Umgang mit seinen personenbezogenen Daten in seinem Recht beeinträchtigt wird, selbst über die Preisgabe und Verwendung seiner Daten zu bestimmen (informationelles Selbstbestimmungsrecht).

Aufgrund der engen Verflechtung von Datenschutz und IT-Sicherheit werden in diesem IT-Grundschutz-Baustein zum Thema *Datenschutz* einerseits die Rahmenbedingungen für den Datenschutz praxisgerecht aufbereitet und andererseits die Verbindung zur IT-Sicherheit im IT-Grundschutz aufgezeigt.

Der IT-Grundschutz-Baustein "Datenschutz"wurde vom Bundesbeauftragten für den Datenschutz und Informationsfreiheit gemeinsam mit dem Arbeitskreis Technik der Datenschutzbeauftragten des Bundes und der Länder sowie den Datenschutzaufsichtsbehörden der Länder erstellt. Er richtet sich an die privaten und öffentlichen Anwender für den IT-Grundschutz in Deutschland.

1.6.2 Gefährdungslage

G 6.1	Fehlende Zulässigkeit der Verarbeitung personenbezogener Daten
G 6.2	Nichteinhaltung der Zweckbindung bei der Verarbeitung personen-bezogener Daten
G 6.3	Überschreitung des Erforderlichkeitsgrundsatzes bei der Verarbeitung personenbezogener Daten
G 6.4	Fehlende oder unzureichende Datenvermeidung und Datensparsamkeit bei der Verarbeitung personenbezogener Daten

G 6.5	Verletzung des Datengeheimnisses bei der Verarbeitung personenbezogener Daten
G 6.6	Fehlende oder nicht ausreichende Vorabkontrolle
G 6.7	Gefährdung der Rechte Betroffener bei der Verarbeitung personenbezogener Daten
G 6.8	Fehlende oder unzureichende Absicherung der Datenverarbeitung im Auftrag bei der Verarbeitung personenbezogener Daten
G 6.9	Fehlende Transparenz für den Betroffenen und die Datenschutz-Kontrollinstanzen
G 6.10	Gefährdung vorgegebener Kontrollziele bei der Verarbeitung personenbezogener Daten
G 6.11	Fehlende oder unzureichende Absicherung der Verarbeitung personenbezogener Daten im Ausland
G 6.12	Unzulässige automatisierten Einzelfallentscheidungen oder Abrufe bei der Verarbeitung personenbezogener Daten
G 6.13	Fehlende oder unzureichende Datenschutzkontrolle

1.6.3 Maßnahmen

M 2.110	Datenschutzaspekte bei der Protokollierung
M 7.1	Datenschutzmanagement
M 7.2	Regelung der Verantwortlichkeiten im Bereich Datenschutz
M 7.3	Aspekte eines Datenschutzkonzeptes
M 7.4	Prüfung rechtlicher Rahmenbedingungen und Vorabkontrolle bei der Verarbeitung personenbezogener Daten
M 7.5	Festlegung von technisch-organisatorischen Maßnahmen entsprechend dem Stand der Technik bei der Verarbeitung personenbezogener Daten
M 7.6	Verpflichtung/Unterrichtung der Mitarbeiter bei der Verarbeitung personenbezogener Daten
M 7.7	Organisatorische Verfahren zur Sicherstellung der Rechte der Betroffenen bei der Verarbeitung personenbezogener Daten
M 7.8	Führung von Verfahrensverzeichnissen und Erfüllung der Meldepflichten bei der Verarbeitung personenbezogener Daten
M 7.9	Datenschutzrechtliche Freigabe
M 7.10	Meldung und Regelung von Abrufverfahren bei der Verarbeitung personenbezogener Daten

M 7.11	Regelung der Auftragsdatenverarbeitung bei der Verarbeitung personenbezogener Daten
M 7.12	Regelung der Verknüpfung und Verwendung von Daten bei der Verarbeitung personenbezogener Daten
M 7.13	Dokumentation der datenschutzrechtlichen Zulässigkeit
M 7.14	Aufrechterhaltung des Datenschutzes im laufenden Betrieb
M 7.15	Datenschutzgerechte Löschung/Vernichtung

1.7 Schutz vor Schadprogrammen (B 1.6)

1.7.1 Beschreibung

Jede Institution sollte geeignete vorbeugende Maßnahmen gegen Schadprogramme zusammenstellen sowie das Vorgehen im Fall einer Infektion mit Schadprogrammen regeln. Unter Schadprogrammen werden neben den klassischen Computer-Viren auch Trojanische Pferde, Computer-Würmer und weitere Schaden verursachende Software verstanden. Als Grundlage, um das Eindringen von Schadprogrammen in IT-Systeme zu verhindern, sollte ein Sicherheitskonzept gegen Schadprogramme entwickelt werden. Eine hundertprozentige Sicherheit ist auch beim Schutz vor Schadprogrammen nicht möglich. Im Bewusstsein des Restrisikos müssen Maßnahmen ergriffen werden, einem Eindringen von Schadprogrammen vorzubeugen. Ist eine vorbeugende Abwehr nicht gelungen, soll das Eindringen von Schadprogrammen zumindest aber so früh wie möglich entdeckt werden. Darüber hinaus werden in diesem Baustein Maßnahmen benannt, die der Schadensminderung dienen, wenn ein Schadprogramm nicht rechtzeitig entdeckt werden konnte. Wesentlich ist die konsequente Anwendung der Maßnahmen und die ständige Aktualisierung der eingesetzten technischen Methoden. Diese Forderung begründet sich durch die täglich neu auftretenden Schadprogramme bzw. durch ständig neue Variationen schon bekannter Schadprogramme. Durch die Weiterentwicklung von Betriebssystemen, Programmiersprachen und Anwendungsprogrammen entstehen regelmäßig neue Angriffspotentiale für Schadprogramme, so dass rechtzeitig geeignete Gegenmaßnahmen eingeleitet werden müssen.

Um für eine Gesamtorganisation einen effektiven Schutz gegen Schadprogramme zu erreichen, wird in diesem Baustein die Vorgehensweise zur Erstellung und Realisierung eines entsprechenden Sicherheitskonzeptes erläutert. Konkrete Maßnahmenempfehlungen zum Schutz vor Schadprogrammen für einzelne IT-Systeme finden sich in den systemspezifischen Bausteinen.

1.7.2 Gefährdungslage

Für den IT-Grundschutz werden bezüglich Schadprogramme die folgenden typischen Gefährdungen betrachtet:

Organisatorische Mängel

G 2.1	Fehlende oder unzureichende Regelungen
G 2.2	Unzureichende Kenntnis über Regelungen
G 2.3	Fehlende, ungeeignete, inkompatible Betriebsmittel
G 2.4	Unzureichende Kontrolle der Sicherheitsmaßnahmen
G 2.8	Unkontrollierter Einsatz von Betriebsmitteln
G 2.9	Mangelhafte Anpassung an Veränderungen beim IT-Einsatz
G 2.136	Fehlende Übersicht über den Informationsverbund

Technisches Versagen

G 4.13	Verlust gespeicherter Daten
G 4.22	Software-Schwachstellen oder -Fehler

Vorsätzliche Handlungen

G 5.2	Manipulation an Informationen oder Software
G 5.23	Schadprogramme
G 5.28	Verhinderung von Diensten
G 5.42	Social Engineering
G 5.71	Vertraulichkeitsverlust schützenswerter Informationen
G 5.85	Integritätsverlust schützenswerter Informationen
G 5.142	Verbreitung von Schadprogrammen über mobile Datenträger

1.7.3 Maßnahmen

Bei der Erstellung eines Sicherheitskonzeptes gegen Schadprogramme (siehe M 2.154 Erstellung eines Sicherheitskonzeptes gegen Schadprogramme) muss zunächst ermittelt werden, welche der vorhandenen oder geplanten IT-Systeme in das Sicherheitskonzept einzubeziehen sind. Für diese IT-Systeme müssen die für die Umsetzung von Sicherheitsmaßnahmen relevanten Einflussfaktoren betrachtet werden. Darauf aufbauend können dann die technischen und organisatorischen Maßnahmen ausgewählt werden. Hierzu ist insbesondere die Auswahl geeigneter technischer Gegenmaßnahmen wie der Einsatz von Viren-Schutzprogrammen zu beachten (siehe M 2.157 Auswahl eines geeigneten Viren-Schutzprogramms). Neben der Einrichtung eines Meldewesens (siehe M 2.158 Meldung von Schadprogramm-Infektionen) und einer Koordinierung der Aktualisierung eingesetzter Schutzprodukte (siehe M 2.159 Aktualisierung der eingesetzten Viren-Schutzprogramme und Signaturen) sind für die Umsetzung des Konzeptes eine Reihe von Regelungen zu vereinbaren.

Die wichtigsten vorbeugenden Maßnahmen gegen Schäden durch Schadsoftware sind der Einsatz von Viren-Schutzprogrammen sowie regelmäßige Datensicherungen (siehe M 6.32 Regelmäßige Datensicherung).

Nachfolgend wird das Maßnahmenbündel für den Bereich SSchutz vor Schadprogrammen"vorgestellt:

Planung und Konzeption

M 2.154	(A)	Erstellung eines Sicherheitskonzeptes gegen Schadprogramme
M 2.160	(A)	Regelungen zum Schutz vor Schadprogrammen
M 3.69	(W)	Einführung in die Bedrohung durch Schadprogramme

Beschaffung

M 2.157	(A)	Auswahl eines geeigneten Viren-Schutzprogramms

Umsetzung

M 4.84	(A)	Nutzung der BIOS-Sicherheitsmechanismen

Betrieb

M 2.34	(A)	Dokumentation der Veränderungen an einem bestehenden System
M 2.158	(A)	Meldung von Schadprogramm-Infektionen
M 2.159	(A)	Aktualisierung der eingesetzten Viren-Schutzprogramme und Signaturen
M 2.224	(A)	Vorbeugung gegen Schadprogramme
M 4.3	(A)	Einsatz von Viren-Schutzprogrammen

Notfallvorsorge

M 6.23	(A)	Verhaltensregeln bei Auftreten von Schadprogrammen
M 6.24	(A)	Erstellen eines Notfall-Bootmediums
M 6.32	(A)	Regelmäßige Datensicherung

1.8 Kryptokonzept (B 1.7)

1.8.1 Beschreibung

Dieser Baustein beschreibt eine Vorgehensweise, wie in einer heterogenen Umgebung sowohl die lokal gespeicherten Daten als auch die zu übertragenen Daten wirkungsvoll durch kryptographische Verfahren und Techniken geschützt werden können. Dazu wird beschrieben, wie und wo in einer heterogenen Umgebung kryptographische Verfahren und die entsprechenden Komponenten eingesetzt werden können. Da beim Einsatz kryptographischer Verfahren sehr viele komplexe Einflussfaktoren zu betrachten sind, sollte hierfür ein Kryptokonzept erstellt werden.

In diesem Baustein wird daher beschrieben, wie ein Kryptokonzept erstellt werden kann. Beginnend mit der Bedarfsermittlung und der Erhebung der Einflussfaktoren geht es über die Auswahl geeigneter kryptographischer Lösungen und Produkte bis hin zur Sensibilisierung und Schulung der Anwender und zur Krypto-Notfallvorsorge.

Für die Umsetzung dieses Bausteins sollte ein elementares Verständnis der grundlegenden kryptographischen Mechanismen vorhanden sein. Ein Überblick über kryptographische Grundbegriffe findet sich in M 3.23 Einführung in kryptographische Grundbegriffe .

1.8.2 Gefährdungslage

Kryptographische Verfahren werden eingesetzt zur Gewährleistung von
Vertraulichkeit,
Integrität,
Authentizität und
Nichtabstreitbarkeit.

Daher werden für den IT-Grundschutz primär die folgenden Gefährdungen für kryptographische Verfahren betrachtet:

Organisatorische Mängel

G 2.1	Fehlende oder unzureichende Regelungen
G 2.2	Unzureichende Kenntnis über Regelungen
G 2.4	Unzureichende Kontrolle der Sicherheitsmaßnahmen
G 2.19	Unzureichendes Schlüsselmanagement bei Verschlüsselung

Menschliche Fehlhandlungen

G 3.1	Vertraulichkeits- oder Integritätsverlust von Daten durch Fehlverhalten

G 3.32	Verstoß gegen rechtliche Rahmenbedingungen beim Einsatz von kryptographischen Verfahren
G 3.33	Fehlbedienung von Kryptomodulen

Technisches Versagen

G 4.22	Software-Schwachstellen oder -Fehler
G 4.33	Schlechte oder fehlende Authentikation
G 4.34	Ausfall eines Kryptomoduls
G 4.35	Unsichere kryptographische Algorithmen
G 4.36	Fehler in verschlüsselten Daten

Vorsätzliche Handlungen

G 5.27	Nichtanerkennung einer Nachricht
G 5.71	Vertraulichkeitsverlust schützenswerter Informationen
G 5.81	Unautorisierte Benutzung eines Kryptomoduls
G 5.82	Manipulation eines Kryptomoduls
G 5.83	Kompromittierung kryptographischer Schlüssel
G 5.84	Gefälschte Zertifikate
G 5.85	Integritätsverlust schützenswerter Informationen

1.8.3 Maßnahmen

Entwicklung eines Kryptokonzepts (siehe M 2.161 Entwicklung eines Kryptokonzepts) Der Einsatz kryptographischer Verfahren wird von einer großen Zahl von Einflussfaktoren bestimmt. Das IT-System, das Datenvolumen, das angestrebte Sicherheitsniveau und die Verfügbarkeitsanforderungen sind einige dieser Faktoren. Daher sollte zunächst ein Konzept entwickelt werden, in dem alle Einflussgrößen und Entscheidungskriterien für die Wahl eines konkreten kryptographischen Verfahrens und der entsprechenden Produkte berücksichtigt werden und das gleichzeitig unter Kostengesichtspunkten wirtschaftlich vertretbar ist.

Ermittlung der Anforderungen an die kryptographischen Verfahren Es muss ein Anforderungskatalog erstellt werden, in dem die Einflussgrößen und die Entscheidungskriterien beschrieben werden, die einem Einsatz von kryptographischen Verfahren zugrunde liegen (siehe M 2.162 Bedarfserhebung für den Einsatz kryptographischer Verfahren und Produkte und M 2.163 Erhebung der Einflussfaktoren für kryptographische Verfahren und Produkte). Kryptographische Verfahren können auf den verschiedenen Schichten des ISO/OSI-Schichtenmodells eingesetzt werden. Je nach den festgestellten Anforderungen oder Gefährdungen ist der Einsatz auf bestimmten Schichten zu empfehlen (siehe auch

M 4.90 Einsatz von kryptographischen Verfahren auf den verschiedenen Schichten des ISO/OSI-Referenzmodells .

Auswahl geeigneter kryptographischer Verfahren (siehe M 2.164 Auswahl eines geeigneten kryptographischen Verfahrens) Bei der Auswahl von kryptographischen Verfahren steht zunächst die Frage, ob symmetrische, asymmetrische oder hybride Algorithmen geeignet sind, im Vordergrund und dann die Mechanismenstärke. Anschließend sind geeignete Produkte zu bestimmen.

Auswahl eines geeigneten kryptographischen Produktes (siehe M 2.165 Auswahl eines geeigneten kryptographischen Produktes) Nachdem alle Rahmenbedingungen bestimmt worden sind, muss ein Produkt ausgewählt werden, das die im Kryptokonzept dargelegte Sicherheitsfunktionalität bietet. Ein solches Produkt, im folgenden kurz Kryptomodul genannt, kann dabei aus Hardware, Software, Firmware oder aus einer diesbezüglichen Kombination sowie der zur Durchführung der Kryptoprozesse notwendigen Bauteilen wie Speicher, Prozessoren, Busse, Stromversorgung etc. bestehen. Ein Kryptomodul kann zum Schutz von sensiblen Daten bzw. Informationen in unterschiedlichsten Rechner- oder Telekommunikationssystemen Verwendung finden.

Geeigneter Einsatz der Kryptomodule (siehe M 2.166 Regelung des Einsatzes von Kryptomodulen) Auch im laufenden Betrieb müssen eine Reihe von Sicherheitsanforderungen an ein Kryptomodul gestellt werden. Neben der Sicherheit der durch das Kryptomodul zu schützenden Daten geht es schwerpunktmäßig auch darum, das Kryptomodul selbst gegen unmittelbare Angriffe und Fremdeinwirkung zu schützen.

Notfallvorsorge, hierzu gehören die Datensicherung bei Einsatz kryptographischer Verfahren (siehe M 6.56 Datensicherung bei Einsatz kryptographischer Verfahren), also die Sicherung der Schlüssel, der Konfigurationsdaten der eingesetzten Produkte, der verschlüsselten Daten, die Informationsbeschaffung über sowie die Reaktion auf Sicherheitslücken.

Nachfolgend wird das Maßnahmenbündel für den Bereich "Kryptokonzept"vorgestellt. Auf eine Wiederholung von Maßnahmen anderer Bausteine wird hier verzichtet.

Planung und Konzeption

M 2.161	(A)	Entwicklung eines Kryptokonzepts
M 2.162	(A)	Bedarfserhebung für den Einsatz kryptographischer Verfahren und Produkte
M 2.163	(A)	Erhebung der Einflussfaktoren für kryptographische Verfahren und Produkte
M 2.164	(A)	Auswahl eines geeigneten kryptographischen Verfahrens
M 2.166	(A)	Regelung des Einsatzes von Kryptomodulen
M 3.23	(A)	Einführung in kryptographische Grundbegriffe
M 4.90	(A)	Einsatz von kryptographischen Verfahren auf den verschiedenen Schichten des ISO/OSI-Referenzmodells

Beschaffung

M 2.165	(A)	Auswahl eines geeigneten kryptographischen Produktes
M 4.85	(Z)	Geeignetes Schnittstellendesign bei Kryptomodulen
M 4.88	(A)	Anforderungen an die Betriebssystem-Sicherheit beim Einsatz von Kryptomodulen

Umsetzung

M 2.46	(A)	Geeignetes Schlüsselmanagement
M 4.86	(A)	Sichere Rollenteilung und Konfiguration der Kryptomodule
M 4.87	(Z)	Physikalische Sicherheit von Kryptomodulen
M 4.89	(Z)	Abstrahlsicherheit

Notfallvorsorge

M 6.56	(A)	Datensicherung bei Einsatz kryptographischer Verfahren

1.9 Behandlung von Sicherheitsvorfällen (B 1.8)

Keine ausreichende Einschätzung möglich.

1.10 Hard- und Software-Management (B 1.9)

Keine ausreichende Einschätzung möglich.

1.11 Standardsoftware (B 1.10)

Keine ausreichende Einschätzung möglich.

1.12 Outsourcing (B 1.11)

1.12.1 Beschreibung

Beim Outsourcing werden Arbeits- oder Geschäftsprozesse einer Organisation ganz oder teilweise zu externen Dienstleistern ausgelagert. Outsourcing kann sowohl Nutzung und Betrieb von Hardware und Software, aber auch Dienstleistungen betreffen. Dabei ist es unerheblich, ob die Leistung in den Räumlichkeiten des Auftraggebers oder in einer externen Betriebsstätte des Outsourcing-Dienstleisters erbracht wird. Typische Beispiele

sind der Betrieb eines Rechenzentrums, einer Applikation, einer Webseite oder des Wachdienstes. Outsourcing ist ein Oberbegriff, der oftmals durch weitere Begriffe ergänzt wird: Tasksourcing bezeichnet das Auslagern von Teilbereichen. Werden Dienstleistungen mit Bezug zur IT-Sicherheit ausgelagert, wird von Security Outsourcing oder Managed Security Services gesprochen. Beispiele sind die Auslagerung des Firewall-Betriebs, die Überwachung des Netzes, Virenschutz oder der Betrieb eines Virtual Private Networks (VPN). Unter Application Service Provider (ASP) versteht man einen Dienstleister, der auf seinen eigenen Systemen einzelne Anwendungen oder Software für seine Kunden betreibt (E-Mail, SAP-Anwendungen, Archivierung, Web-Shops, Beschaffung). Auftraggeber und Dienstleister sind dabei über das Internet oder ein VPN miteinander verbunden. Beim Application Hosting ist ebenfalls der Betrieb von Anwendungen an einen Dienstleister ausgelagert, jedoch gehören im Gegensatz zum ASP-Modell die Anwendungen noch dem jeweiligen Kunden. Da die Grenzen zwischen klassischem Outsourcing und reinem ASP in der Praxis zunehmend verschwimmen, wird im Folgenden nur noch der Oberbegriff Outsourcing verwendet.

Das Auslagern von Geschäfts- und Produktionsprozessen ist ein etablierter Bestandteil heutiger Organisationsstrategien. Speziell in den letzten beiden Jahrzehnten hat sich der Trend zum Outsourcing enorm verstärkt, und dieser scheint auch für die nächste Zukunft ungebrochen. Es gibt aber inzwischen auch publizierte Beispiele für gescheiterte Outsourcing-Projekte, wo der Auftraggeber den Outsourcing-Vertrag gekündigt hat und die ausgelagerten Geschäftsprozesse wieder in Eigenregie betreibt (Insourcing).

Die Gründe für Outsourcing sind vielfältig: die Konzentration einer Organisation auf ihre Kernkompetenzen, die Möglichkeit einer Kostenersparnis (z. B. keine Anschaffungs- oder Betriebskosten für IT-Systeme), der Zugriff auf spezialisierte Kenntnisse und Ressourcen, die Freisetzung interner Ressourcen für andere Aufgaben, die Straffung der internen Verwaltung, die verbesserte Skalierbarkeit der Geschäfts- und Produktionsprozesse, die Erhöhung der Flexibilität sowie der Wettbewerbsfähigkeit einer Organisation sind nur einige Beispiele.

Beim Auslagern von IT-gestützten Organisationsprozessen werden die IT-Systeme und Netze der auslagernden Organisation und ihres Outsourcing-Dienstleisters in der Regel eng miteinander verbunden, so dass Teile von internen Geschäftsprozessen unter Leitung und Kontrolle eines externen Dienstleisters ablaufen. Ebenso findet auf personeller Ebene ein intensiver Kontakt statt.

Durch die enge Verbindung zum Dienstleister und die entstehende Abhängigkeit von der Dienstleistungsqualität ergeben sich Risiken für den Auftraggeber, durch die im schlimmsten Fall sogar die Geschäftsgrundlage des Unternehmens oder der Behörde vital gefährdet werden können. (Beispielsweise könnten sensitive Organisationsinformationen gewollt oder ungewollt nach außen preisgegeben werden.) Der Betrachtung von Sicherheitsaspekten und der Gestaltung vertraglicher Regelungen zwischen Auftraggeber und Outsourcing-Dienstleister kommt im Rahmen eines Outsourcing-Vorhabens somit eine zentrale Rolle zu.

Den Schwerpunkt dieses Bausteins bilden daher Maßnahmen, die sich mit IT-Sicherheitsaspekten des Outsourcing beschäftigen. Dazu zählen ebenfalls geeignete Maßnahmen zur Kontrolle der vertraglich vereinbarten Ziele und Leistungen sowie der IT-Sicherheitsmaßnahmen.

1.12.2 Gefährdungslage

Die Gefährdungslage eines Outsourcing-Vorhabens ist ausgesprochen vielschichtig. Die Entscheidung über das Auslagern einer speziellen Aktivität beeinflusst nachhaltig die strategische Ausrichtung der Organisation, die Definition ihrer Kernkompetenzen, die Ausgestaltung der Wertschöpfungskette und betrifft viele weitere wesentliche Belange eines Organisationsmanagements. Es sollten daher alle Anstrengungen unternommen werden, um Fehlentwicklungen des Unternehmens oder der Behörde frühzeitig zu erkennen und zu verhindern.

Die Gefährdungen können parallel auf physikalischer, technischer und auch menschlicher Ebene existieren und sind nachfolgend in den einzelnen Gefährdungskatalogen aufgeführt. Um die jeweils existierenden Risiken quantitativ bewerten zu können, müssen zuvor die organisationseigenen Werte und Informationen entsprechend ihrer strategischen Bedeutung für die Organisation beurteilt und klassifiziert werden.

Höhere Gewalt

G 1.10	Ausfall eines Weitverkehrsnetzes

Organisatorische Mängel

G 2.1	Fehlende oder unzureichende Regelungen
G 2.7	Unerlaubte Ausübung von Rechten
G 2.26	Fehlendes oder unzureichendes Test- und Freigabeverfahren
G 2.47	Ungesicherter Akten- und Datenträgertransport
G 2.66	Unzureichendes Sicherheitsmanagement
G 2.67	Ungeeignete Verwaltung von Zugangs- und Zugriffsrechten
G 2.83	Fehlerhafte Outsourcing-Strategie
G 2.84	Unzulängliche vertragliche Regelungen mit einem externen Dienstleister
G 2.85	Unzureichende Regelungen für das Ende des Outsourcing-Vorhabens
G 2.86	Abhängigkeit von einem Outsourcing-Dienstleister
G 2.88	Störung des Betriebsklimas durch ein Outsourcing-Vorhaben
G 2.89	Mangelhafte IT-Sicherheit in der Outsourcing-Einführungsphase
G 2.90	Schwachstellen bei der Anbindung an einen Outsourcing-Dienstleister

G 2.93	Unzureichendes Notfallvorsorgekonzept beim Outsourcing

Menschliche Fehlhandlungen

G 3.1	Vertraulichkeits- oder Integritätsverlust von Daten durch Fehlverhalten

Technisches Versagen

G 4.33	Schlechte oder fehlende Authentikation
G 4.34	Ausfall eines Kryptomoduls
G 4.48	Ausfall der Systeme eines Outsourcing-Dienstleisters

Vorsätzliche Handlungen

G 5.10	Missbrauch von Fernwartungszugängen
G 5.20	Missbrauch von Administratorrechten
G 5.42	Social Engineering
G 5.71	Vertraulichkeitsverlust schützenswerter Informationen
G 5.85	Integritätsverlust schützenswerter Informationen
G 5.107	Weitergabe von Daten an Dritte durch den Outsourcing-Dienstleister

1.12.3 Maßnahmen

Ein ausgelagerter IT-Verbund kann sowohl aus Komponenten bestehen, die sich ausschließlich im Einflussbereich des Outsourcing-Dienstleisters befinden, als auch aus Komponenten beim Auftraggeber. In der Regel gibt es in diesem Fall Schnittstellen zur Verbindung der Systeme. Für jedes Teilsystem und für die Schnittstellenfunktionen muss IT-Grundschutz gewährleistet sein.

Planung und Konzeption

M 2.40	(A)	Rechtzeitige Beteiligung des Personal-/Betriebsrates
M 2.42	(B)	Festlegung der möglichen Kommunikationspartner
M 2.221	(A)	Änderungsmanagement
M 2.226	(A)	Regelungen für den Einsatz von Fremdpersonal
M 2.250	(A)	Festlegung einer Outsourcing-Strategie
M 2.251	(A)	Festlegung der Sicherheitsanforderungen für Outsourcing-Vorhaben

M 2.254	(A)	Erstellung eines IT-Sicherheitskonzepts für das Outsourcing-Vorhaben

Beschaffung

M 2.252	(A)	Wahl eines geeigneten Outsourcing-Dienstleisters

Umsetzung

M 2.253	(A)	Vertragsgestaltung mit dem Outsourcing-Dienstleister
M 2.255	(A)	Sichere Migration bei Outsourcing-Vorhaben
M 3.33	(Z)	Sicherheitsüberprüfung von Mitarbeitern
M 5.87	(C)	Vereinbarung über die Anbindung an Netze Dritter
M 5.88	(C)	Vereinbarung über Datenaustausch mit Dritten

Betrieb

M 2.256	(A)	Planung und Aufrechterhaltung der IT-Sicherheit im laufenden Outsourcing-Betrieb

Aussonderung

M 2.307	(A)	Geordnete Beendigung eines Outsourcing-Dienstleistungsverhältnisses

Notfallvorsorge

M 6.83	(A)	Notfallvorsorge beim Outsourcing
M 6.109	(A)	Notfallplan für den Ausfall eines VPNs

1.13 Archivierung (B 1.12)

1.13.1 Beschreibung

ie Abbildung von Geschäftsprozessen und -unterlagen in elektronische Dokumente erfordert eine geeignete Ablage der entstehenden Daten für die spätere Verwendung, deren Wiederfinden und Aufbereitung. Dies betrifft sowohl Datensätze als auch elektronische Repräsentationen papierner Geschäftsdokumente und Belege. Die dauerhafte und unveränderbare Speicherung von elektronischen Dokumenten und anderen Daten wird als Archivierung bezeichnet.

Die Archivierung ist als Teil eines Dokumentenmanagement-Prozesses zu sehen. Neben der Erzeugung, Bearbeitung und Verwaltung elektronischer Dokumente spielt die dauerhafte Speicherung (Archivierung) eine besondere Rolle, denn es wird üblicherweise erwartet, dass einerseits die Dokumente bis zum Ablauf einer vorgegebenen Aufbewahrungsfrist verfügbar sind und andererseits deren Vertraulichkeit- und Integrität gewahrt bleibt. Unter Umständen sollen elektronische Dokumente zeitlich unbegrenzt verfügbar sein.

Die Spannweite der Realisierungsmöglichkeiten eines solchen Archivsystems umfasst:

- kleine Archivsysteme, z. B. bestehend aus einen Archivserver mit angeschlossenem Massenspeicher (wie Festplatte oder Jukebox), bis hin zu
- komplexen, gegebenenfalls weltweit verteilten Archivsystemen zur organisationsweiten Archivierung von relevanten Geschäftsdaten, bestehend aus:
- zentralen Archivserver-Komponenten mit RAID-Systemen, Jukeboxen oder der Anbindung an Storage Area Networks (SAN) für das zentrale Speichern von Dateien,
- WORM-Medien für die revisionssichere, unveränderbare Speicherung von Daten,
- Komponenten zur Indizierung von Dateien, Recherche und zur Umwandlung von Speicherformaten (Rendition),
- dezentralen Cache-Servern für den schnellen Zugriff auf häufig benötigte Daten,
- Client-Software, die einen direkten Zugriff auf Daten des Archivs erlaubt (z. B. auch aus Office-Anwendungen heraus).

Es ist zweckmäßig, elektronische Archive gegenüber Systemen zur Datensicherung abzugrenzen. Bei einer Datensicherung werden Kopien der System- und Nutzdaten angelegt. Die gesicherten Daten werden hierbei physikalisch vom IT-System getrennt und gefahrengeschützt gelagert. Elektronische Archive dagegen sind regelmäßig in den laufenden Systembetrieb eingebunden. Dabei werden üblicherweise große Mengen von Nutzdaten (elektronischen Dokumenten) abgelegt, die aus dem elektronischen Archivsystem heraus jederzeit abgerufen werden können. Bei besonderem Aufbau (z. B. die redundante Auslegung der Speicherkomponenten und eine entsprechende räumliche Anordnung) können größere Archivsysteme teilweise die Funktionalität der Datensicherung (der Nutzdaten) übernehmen.

In diesem Baustein soll ein systematischer Weg aufgezeigt werden, wie ein Konzept zur elektronischen Archivierung erstellt und wie der Aufbau eines Archivsystems und dessen Einbettung innerhalb eines Unternehmens bzw. einer Behörde sichergestellt werden kann. Der Aufwand zur Erstellung und Umsetzung eines solchen Konzepts ist nicht gering. Dieser Baustein sollte immer dann angewandt werden, wenn die zu archivierenden Daten langfristig für die Behörde bzw. das Unternehmen relevant sind.

1.13.2 Gefährdungslage

Für die bei der elektronischen Archivierung zu betrachtenden Archivsysteme sowie die zugehörigen Organisationsprozesse werden im Rahmen des IT-Grundschutzes die folgenden typischen Gefährdungen angenommen:

Höhere Gewalt

G 1.2	Ausfall von IT-Systemen
G 1.7	Unzulässige Temperatur und Luftfeuchte
G 1.9	Datenverlust durch starke Magnetfelder
G 1.14	Datenverlust durch starkes Licht

Organisatorische Mängel

G 2.7	Unerlaubte Ausübung von Rechten
G 2.72	Unzureichende Migration von Archivsystemen
G 2.73	Fehlende Revisionsmöglichkeit von Archivsystemen
G 2.74	Unzureichende Ordnungskriterien für Archive
G 2.75	Mangelnde Kapazität von Archivdatenträgern
G 2.76	Unzureichende Dokumentation von Archivzugriffen
G 2.77	Unzulängliche Übertragung von Papierdaten in elektronische Archive
G 2.78	Unzulängliche Auffrischung von Datenbeständen bei der Archivierung
G 2.79	Unzureichende Erneuerung von digitalen Signaturen bei der Archivierung
G 2.80	Unzureichende Durchführung von Revisionen bei der Archivierung
G 2.81	Unzureichende Vernichtung von Datenträgern bei der Archivierung
G 2.82	Fehlerhafte Planung des Aufstellungsortes von Speicher- und Archivsystemen

Menschliche Fehlhandlungen

G 3.1	Vertraulichkeits- oder Integritätsverlust von Daten durch Fehlverhalten
G 3.16	Fehlerhafte Administration von Zugangs- und Zugriffsrechten
G 3.35	Server im laufenden Betrieb ausschalten
G 3.54	Verwendung ungeeigneter Datenträger bei der Archivierung

G 3.55	Verstoß gegen rechtliche Rahmenbedingungen beim Einsatz von Archivsystemen

Technisches Versagen

G 4.7	Defekte Datenträger
G 4.13	Verlust gespeicherter Daten
G 4.20	Datenverlust bei erschöpftem Speichermedium
G 4.26	Ausfall einer Datenbank
G 4.30	Verlust der Datenbankintegrität/-konsistenz
G 4.31	Ausfall oder Störung von Netzkomponenten
G 4.45	Verzögerte Archivauskunft
G 4.46	Fehlerhafte Synchronisierung von Indexdaten bei der Archivierung
G 4.47	Veralten von Kryptoverfahren

Vorsätzliche Handlungen

G 5.2	Manipulation an Informationen oder Software
G 5.6	Anschlag
G 5.29	Unberechtigtes Kopieren der Datenträger
G 5.82	Manipulation eines Kryptomoduls
G 5.83	Kompromittierung kryptographischer Schlüssel
G 5.85	Integritätsverlust schützenswerter Informationen
G 5.102	Sabotage
G 5.105	Verhinderung der Dienste von Archivsystemen
G 5.106	Unberechtigtes Überschreiben oder Löschen von Archivmedien

1.13.3 Maßnahmen

Die im Folgenden beschriebene Vorgehensweise für die Einführung und den Betrieb von elektronischen Archivsystemen werden empfohlen. Bereits bei der Planung ist zu berücksichtigen, dass die eingesetzten Archivsysteme und -medien im Lauf der Zeit technologisch und physikalisch veralten werden. Daher schließt sich an eine Planungs- und Einführungs-/Betriebsphase eine Migrationsphase an, in der das bestehende Archivsystem oder Teile davon durch neue Technologien und Komponenten ersetzt werden. Die Migrationsphase umfasst auch die Übertragung der archivierten Daten und Dokumente in zukünftig verwendete Datenformate.

Planung und Konzeption

M 2.242	(A)	Zielsetzung der elektronischen Archivierung
M 2.243	(A)	Entwicklung des Archivierungskonzepts
M 2.244	(A)	Ermittlung der technischen Einflussfaktoren für die elektronische Archivierung
M 2.245	(A)	Ermittlung der rechtlichen Einflussfaktoren für die elektronische Archivierung
M 2.246	(A)	Ermittlung der organisatorischen Einflussfaktoren für die elektronische Archivierung
M 2.259	(Z)	Einführung eines übergeordneten Dokumentenmanagements
M 2.262	(A)	Regelung der Nutzung von Archivsystemen
M 2.265	(Z)	Geeigneter Einsatz digitaler Signaturen bei der Archivierung

Beschaffung

M 4.168	(A)	Auswahl eines geeigneten Archivsystems
M 4.169	(A)	Verwendung geeigneter Archivmedien
M 4.170	(A)	Auswahl geeigneter Datenformate für die Archivierung von Dokumenten

Umsetzung

M 1.59	(A)	Geeignete Aufstellung von Speicher- und Archivsystemen
M 2.266	(C)	Regelmäßige Erneuerung technischer Archivsystem-Komponenten
M 3.34	(A)	Einweisung in die Administration des Archivsystems
M 3.35	(A)	Einweisung der Benutzer in die Bedienung des Archivsystems

Betrieb

M 1.60	(A)	Geeignete Lagerung von Archivmedien
M 2.257	(C)	Überwachung der Speicherressourcen von Archivmedien
M 2.258	(A)	Konsistente Indizierung von Dokumenten bei der Archivierung
M 2.260	(B)	Regelmäßige Revision des Archivierungsprozesses
M 2.261	(B)	Regelmäßige Marktbeobachtung von Archivsystemen
M 2.263	(A)	Regelmäßige Aufbereitung von archivierten Datenbeständen

M 2.264	(B)	Regelmäßige Aufbereitung von verschlüsselten Daten bei der Archivierung
M 4.171	(A)	Schutz der Integrität der Index-Datenbank von Archivsystemen
M 4.172	(C)	Protokollierung der Archivzugriffe
M 4.173	(B)	Regelmäßige Funktions- und Recoverytests bei der Archivierung

Notfallvorsorge

M 6.84	(A)	Regelmäßige Datensicherung der System- und Archivdaten

1.14 Sensibilisierung und Schulung zur Informationssicherheit (B 1.13)

Keine ausreichende Einschätzung möglich.

1.15 Patch- und Änderungsmanagement (B 1.14)

Keine ausreichende Einschätzung möglich.

1.16 Löschen und Vernichten von Daten (B 1.15)

Keine ausreichende Einschätzung möglich.

1.17 Anforderungsmanagement (B 1.16)

Keine ausreichende Einschätzung möglich.

2 Bausteine für Infrastruktur

2.1 Gebäude (B 2.1)

2.1.1 Beschreibung

Gebäude bilden den äußeren Rahmen, um Geschäftsprozesse durchführen zu können. Ein Gebäude umgibt die stationären Arbeitsplätze, die verarbeiteten Informationen sowie die aufgestellte Informationstechnik und gewährleistet für diese somit einen äußeren Schutz. Weiterhin ermöglichen die Infrastruktureinrichtungen eines Gebäudes häufig erst die Durchführung von Geschäftsprozessen und den IT-Betrieb. Daher ist einerseits das Bauwerk, also Wände, Decken, Böden, Dach, Fenster und Türen zu betrachten und andererseits alle gebäudeweiten Versorgungseinrichtungen wie Strom, Wasser, Gas, Heizung, Rohrpost etc. Spezielle Räumlichkeiten wie Serverräume werden in den jeweiligen Bausteinen der Schicht 2 betrachtet.

2.1.2 Gefährdungslage

Für den IT-Grundschutz eines Gebäudes werden folgende typische Gefährdungen angenommen:

Höhere Gewalt

G 1.3	Blitz
G 1.4	Feuer
G 1.5	Wasser

Organisatorische Mängel

G 2.1	Fehlende oder unzureichende Regelungen
G 2.6	Unbefugter Zutritt zu schutzbedürftigen Räumen

Menschliche Fehlhandlungen

G 3.85	Verletzung von Brandschottungen

Technisches Versagen

G 4.1	Ausfall der Stromversorgung
G 4.2	Ausfall interner Versorgungsnetze
G 4.3	Ausfall vorhandener Sicherungseinrichtungen

Vorsätzliche Handlungen

G 5.3	Unbefugtes Eindringen in ein Gebäude
G 5.4	Diebstahl
G 5.5	Vandalismus
G 5.6	Anschlag

2.1.3 Maßnahmen

Bei der Nutzung von Gebäuden für den Geschäftsbetrieb von Behörden oder Unternehmen sind hinsichtlich der Informationssicherheit bei bestimmten Maßnahmen unterschiedliche Vorgehensweisen zu verfolgen. Bei einem Neubau können erforderliche Maßnahmen zu einem großen Teil schon in der Planungsphase durchgeführt werden.

Wenn es sich dagegen um eine Anmietung oder die Nutzung eines bestehenden Gebäudes handelt, was eventuell mit Erweiterungs- bzw. Umbaumaßnahmen verbunden sein kann, sind die Möglichkeiten zur Realisierung einer adäquaten Informationssicherheit oft viel stärker eingeschränkt.

Planung und Konzeption

M 1.3	(A)	Angepasste Aufteilung der Stromkreise
M 1.4	(B)	Blitzschutzeinrichtungen
M 1.5	(Z)	Galvanische Trennung von Außenleitungen
M 1.7	(A)	Handfeuerlöscher
M 1.8	(A)	Raumbelegung unter Berücksichtigung von Brandlasten
M 1.10	(Z)	Verwendung von Sicherheitstüren und -fenstern
M 1.11	(A)	Lagepläne der Versorgungsleitungen
M 1.12	(A)	Vermeidung von Lagehinweisen auf schützenswerte Gebäudeteile
M 1.13	(Z)	Anordnung schützenswerter Gebäudeteile
M 1.14	(Z)	Selbsttätige Entwässerung
M 1.16	(Z)	Geeignete Standortauswahl
M 1.18	(Z)	Gefahrenmeldeanlage
M 1.19	(Z)	Einbruchsschutz
M 2.334	(Z)	Auswahl eines geeigneten Gebäudes

Umsetzung

M 1.1	(A)	Einhaltung einschlägiger DIN-Normen/VDE-Vorschriften
M 1.2	(A)	Regelungen für Zutritt zu Verteilern
M 1.6	(A)	Einhaltung von Brandschutzvorschriften

M 1.17	(Z)	Pförtnerdienst
M 1.51	(A)	Brandlastreduzierung
M 2.17	(A)	Zutrittsregelung und -kontrolle

Betrieb

M 1.15	(A)	Geschlossene Fenster und Türen
M 2.14	(A)	Schlüsselverwaltung
M 2.15	(B)	Brandschutzbegehungen
M 2.391	(B)	Frühzeitige Information des Brandschutzbeauftragten

Aussonderung

M 2.308	(Z)	Auszug aus Gebäuden

Notfallvorsorge

M 6.17	(A)	Alarmierungsplan und Brandschutzübungen

2.2 Serverraum (B 2.4)

2.2.1 Beschreibung

Der Serverraum dient in erster Linie zur Unterbringung von Servern, z. B. eines LAN-Servers, eines Unix-Zentralrechners oder eines Servers für eine TK-Anlage. Darüber hinaus können dort serverspezifische Unterlagen, Datenträger in kleinem Umfang oder weitere Hardware (Sternkoppler, Protokolldrucker, Klimatechnik) vorhanden sein.

In einem Serverraum ist kein ständig besetzter Arbeitsplatz eingerichtet, er wird nur sporadisch und zu kurzfristigen Arbeiten betreten. Zu beachten ist jedoch, dass im Serverraum aufgrund der Konzentration von IT-Geräten und Daten ein deutlich höherer Schaden eintreten kann als zum Beispiel in einem Büroraum.

2.2.2 Gefährdungslage

Für den IT-Grundschutz eines Serverraumes werden folgende typische Gefährdungen angenommen:

Höhere Gewalt

G 1.4	Feuer

G 1.5	Wasser
G 1.7	Unzulässige Temperatur und Luftfeuchte
G 1.16	Ausfall von Patchfeldern durch Brand

Organisatorische Mängel

G 2.1	Fehlende oder unzureichende Regelungen
G 2.6	Unbefugter Zutritt zu schutzbedürftigen Räumen

Technisches Versagen

G 4.1	Ausfall der Stromversorgung
G 4.2	Ausfall interner Versorgungsnetze
G 4.6	Spannungsschwankungen/Überspannung/Unterspannung

Vorsätzliche Handlungen

G 5.1	Manipulation oder Zerstörung von Geräten oder Zubehör
G 5.2	Manipulation an Informationen oder Software
G 5.3	Unbefugtes Eindringen in ein Gebäude
G 5.4	Diebstahl
G 5.5	Vandalismus

2.2.3 Maßnahmen

Um den betrachteten IT-Verbund abzusichern, müssen zusätzlich zu diesem Baustein noch weitere Bausteine umgesetzt werden, gemäß den Ergebnissen der Modellierung nach IT-Grundschutz.

Bei der Auswahl und Gestaltung eines Serverraums sind eine Reihe infrastruktureller und organisatorischer Maßnahmen umzusetzen, die in M 1.58 Technische und organisatorische Vorgaben für Serverräume beschrieben sind. Dabei sind bei bestimmten Maßnahmen unterschiedliche Vorgehensweisen zu verfolgen, je nachdem, ob ein Serverraum in einem neu zu errichtenden Gebäude eingerichtet werden soll oder ob es sich um eine Anmietung oder die Nutzung eines bestehenden Gebäudes handelt. In diesem zweiten Fall sind die Möglichkeiten zur Realisierung einer adäquaten IT-Sicherheit oft viel stärker eingeschränkt. Die Schritte, die bei der Gestaltung eines Serverraums durchlaufen werden sollten, sowie die Maßnahmen, die in den jeweiligen Schritten beachtet werden sollten, sind im folgenden aufgeführt.

Planung und Konzeption

M 1.3	(A)	Angepasste Aufteilung der Stromkreise
M 1.7	(A)	Handfeuerlöscher
M 1.10	(C)	Verwendung von Sicherheitstüren und -fenstern
M 1.18	(Z)	Gefahrenmeldeanlage
M 1.24	(C)	Vermeidung von wasserführenden Leitungen
M 1.26	(W)	Not-Aus-Schalter
M 1.27	(B)	Klimatisierung
M 1.28	(B)	Lokale unterbrechungsfreie Stromversorgung
M 1.31	(Z)	Fernanzeige von Störungen
M 1.52	(Z)	Redundanz, Modularität und Skalierbarkeit in der technischen Infrastruktur
M 1.58	(A)	Technische und organisatorische Vorgaben für Serverräume
M 1.62	(C)	Brandschutz von Patchfeldern

Umsetzung

M 2.17	(A)	Zutrittsregelung und -kontrolle
M 2.21	(A)	Rauchverbot

Betrieb

M 1.15	(A)	Geschlossene Fenster und Türen
M 1.23	(A)	Abgeschlossene Türen

2.3 Rechenzentrum (B 2.9)

2.3.1 Beschreibung

In den meisten Institutionen werden alle wesentlichen strategischen und operativen Funktionen und Aufgaben durch Informationstechnik (IT) maßgeblich unterstützt oder sind sogar ohne IT nicht auszuführen. Die IT-Systeme der Institution selbst und auch deren Anbindung an externe Netze müssen in einer angemessenen Umgebung und Infrastruktur betrieben werden. Nur so lässt sich die nötige Verfügbarkeit der IT sicherstellen. Die Anforderungen an die Leistungsfähigkeit dieser Systeme und der Netzumgebung steigen stetig an. Um diesem Leistungsbedarf gerecht zu werden, um entsprechende Reserven vorzuhalten und um die IT auch wirtschaftlich betreiben zu können, haben Behörden und Unternehmen jeglicher Größe ihre IT-Landschaft in Rechenzentren konzentriert.

Als Rechenzentrum werden die für den Betrieb von komplexen IT-Infrastrukturen (Server-

und Speichersysteme, Systeme zur Datensicherung, aktive Netzkomponenten und TK-Systeme, zentrale Drucksysteme usw.) erforderlichen Einrichtungen (Klimatechnik, Elektroversorgung, überwachende und alarmierende Technik) und Räumlichkeiten (z. B. Rechnersaal, Räume für die aktiven Netzkomponenten, Technikräume, Archiv, Lager, Aufenthaltsraum usw.) bezeichnet. Die Abgrenzung vom Rechenzentrum zum Serverraum besteht vor allem darin, dass in einem Rechenzentrum eine räumliche Trennung der IT-Systeme und der unterstützenden Infrastruktur (Elektroversorgung, Klimatechnik usw.) obligatorisch ist. Ein Rechenzentrum sollte insgesamt einen Sicherheitsbereich bilden, der in sich noch mindestens in die organisatorisch und physisch getrennten Sicherheitsbereiche "Infrastruktur" und "IT" aufgeteilt wird. Ein Rechenzentrum ist entweder ständig personell besetzt (Schichtdienst) oder es existiert in bedienerlosen Zeiten eine Rufbereitschaft (mit oder ohne Fernadministrationsmöglichkeit). In einem Rechenzentrum kann aufgrund der Konzentration von IT-Geräten und Daten ein deutlich höherer Schaden eintreten als bei dezentraler Datenverarbeitung. In jedem Fall ist beim Einsatz einer Großrechenanlage der Baustein Rechenzentrum anzuwenden.

Gegenstand dieses Bausteins ist ein Rechenzentrum mittlerer Art und Güte. Die Sicherheitsanforderungen liegen zwischen denen eines Serverraums oder "Serverparks" und denen von Hochsicherheitsrechenzentren, wie sie beispielsweise im Bankenbereich eingesetzt werden. Neben den hier aufgeführten Standard-Sicherheitsmaßnahmen, die sich in der Praxis bewährt haben, sind in den meisten Fällen jedoch weitere, individuelle Sicherheitsmaßnahmen erforderlich, die die konkreten Anforderungen und das jeweilige Umfeld berücksichtigen (hierzu kann beispielsweise die Risikoanalyse basierend auf IT-Grundschutz verwendet werden). Gefährdungen aus den Bereichen Terrorismus oder höhere Gewalt werden durch die hier beschriebenen Standard-Sicherheitsmaßnahmen nur begrenzt Rechnung getragen.

Der Baustein richtet sich einerseits an Anwender, die ein Rechenzentrum betreiben und im Rahmen einer Revision prüfen möchten, ob sie geeignete Standard-Sicherheitsmaßnahmen umgesetzt haben. Auf der anderen Seite kann der Baustein Rechenzentrum auch dazu verwendet werden, überblicksartig die Sicherheitsmaßnahmen abzuschätzen, die bei einer Zentralisierung der IT in einem mittleren Rechenzentrum für einen sicheren Betrieb umgesetzt werden müssen. Um den Baustein überschaubar zu halten, wurde bewusst auf technische Details und planerische Größen verzichtet. Der Neubau eines Rechenzentrums sollte auch von großen IT-Abteilungen nicht ohne Hilfe eines erfahrenen Planungsstabes bzw. einer versierten Planungs- und Beratungsfirma in Betracht gezogen werden. Beim Outsourcing von Rechenzentrumsleistungen kann dieser Baustein dazu benutzt werden, die angebotenen Leistungen im Hinblick auf deren Sicherheitsniveau zu prüfen.

Im Gegensatz zum Schutzbedarf eines Serverraums (siehe dort) sind viele Sicherheitsmaßnahmen für ein Rechenzentrum nicht optional, sondern obligatorisch. Dazu gehören beispielsweise eine angemessene Gefahrenmeldeanlage und eine alternative Stromversorgung. Üblich und bewährt für einen sicheren IT-Betrieb ist eine Brandfrühesterkennung in Raum und Doppelboden von Rechnersaal und Technikräumen und gegebenenfalls auch eine automatische Löschanlage.

2.3.2 Gefährdungslage

Für den IT-Grundschutz eines Rechenzentrums werden folgende typische Gefährdungen angenommen:

Höhere Gewalt

G 1.2	Ausfall von IT-Systemen
G 1.3	Blitz
G 1.4	Feuer
G 1.5	Wasser
G 1.6	Kabelbrand
G 1.7	Unzulässige Temperatur und Luftfeuchte
G 1.8	Staub, Verschmutzung
G 1.11	Technische Katastrophen im Umfeld
G 1.12	Beeinträchtigung durch Großveranstaltungen
G 1.13	Sturm
G 1.16	Ausfall von Patchfeldern durch Brand

Organisatorische Mängel

G 2.1	Fehlende oder unzureichende Regelungen
G 2.2	Unzureichende Kenntnis über Regelungen
G 2.4	Unzureichende Kontrolle der Sicherheitsmaßnahmen
G 2.6	Unbefugter Zutritt zu schutzbedürftigen Räumen
G 2.11	Unzureichende Trassendimensionierung
G 2.12	Unzureichende Dokumentation der Verkabelung

Technisches Versagen

G 4.1	Ausfall der Stromversorgung
G 4.2	Ausfall interner Versorgungsnetze
G 4.3	Ausfall vorhandener Sicherungseinrichtungen

Vorsätzliche Handlungen

G 5.3	Unbefugtes Eindringen in ein Gebäude
G 5.4	Diebstahl
G 5.5	Vandalismus
G 5.6	Anschlag
G 5.16	Manipulation bei Wartungsarbeiten

G 5.68	Unberechtigter Zugang zu den aktiven Netzkomponenten
G 5.102	Sabotage

2.3.3 Maßnahmen

Bei der Auswahl und Gestaltung eines Rechenzentrums sind eine Reihe infrastruktureller und organisatorischer Maßnahmen umzusetzen, die in M 1.49 Technische und organisatorische Vorgaben für das Rechenzentrum beschrieben sind. Dabei sind bei bestimmten Maßnahmen unterschiedliche Vorgehensweisen zu verfolgen, je nachdem, ob ein Rechenzentrum in einem neu zu errichtenden Gebäude eingerichtet werden soll oder ob es sich um eine Anmietung oder die Nutzung eines bestehenden Gebäudes handelt. In diesem zweiten Fall sind die Möglichkeiten zur Realisierung einer adäquaten Informationssicherheit oft viel stärker eingeschränkt. Die Schritte, die bei der Gestaltung eines Rechenzentrums durchlaufen werden sollten, sowie die Maßnahmen, die in den jeweiligen Schritten beachtet werden sollten, sind im Folgenden aufgeführt.

Planung und Konzeption

M 1.3	(A)	Angepasste Aufteilung der Stromkreise
M 1.7	(A)	Handfeuerlöscher
M 1.10	(C)	Verwendung von Sicherheitstüren und -fenstern
M 1.12	(A)	Vermeidung von Lagehinweisen auf schützenswerte Gebäudeteile
M 1.13	(Z)	Anordnung schützenswerter Gebäudeteile
M 1.18	(B)	Gefahrenmeldeanlage
M 1.24	(C)	Vermeidung von wasserführenden Leitungen
M 1.25	(B)	Überspannungsschutz
M 1.26	(W)	Not-Aus-Schalter
M 1.27	(B)	Klimatisierung
M 1.31	(Z)	Fernanzeige von Störungen
M 1.47	(B)	Eigener Brandabschnitt
M 1.48	(B)	Brandmeldeanlage
M 1.49	(A)	Technische und organisatorische Vorgaben für das Rechenzentrum
M 1.50	(C)	Rauchschutz
M 1.52	(W)	Redundanz, Modularität und Skalierbarkeit in der technischen Infrastruktur
M 1.53	(Z)	Videoüberwachung
M 1.54	(Z)	Brandfrühesterkennung / Löschtechnik
M 1.55	(Z)	Perimeterschutz

M 1.56	(A)	Netzersatzanlage
M 1.62	(C)	Brandschutz von Patchfeldern
M 1.70	(A)	Zentrale unterbrechungsfreie Stromversorgung

Umsetzung

M 1.57	(A)	Aktuelle Infrastruktur- und Baupläne
M 2.21	(A)	Rauchverbot
M 2.212	(B)	Organisatorische Vorgaben für die Gebäudereinigung
M 2.213	(A)	Inspektion und Wartung der technischen Infrastruktur

Betrieb

M 1.15	(A)	Geschlossene Fenster und Türen
M 1.23	(A)	Abgeschlossene Türen
M 1.71	(C)	Funktionstests der technischen Infrastruktur
M 1.72	(Z)	Baumaßnahmen während des laufenden Betriebs
M 1.73	(A)	Schutz eines Rechenzentrums gegen unbefugten Zutritt

Notfallvorsorge

M 6.17	(A)	Alarmierungsplan und Brandschutzübungen
M 6.74	(Z)	Notfallarchiv

3 Bausteine für IT-Systeme

3.1 Allgemeiner Server (B 3.101)

3.1.1 Beschreibung

Server sind IT-Systeme, die Dienste (Services) für andere IT-Systeme (Clients) im Netz anbieten. Sie werden typischerweise in zentralen, besonders gesicherten Räumlichkeiten betrieben, beispielsweise in einem Serverraum oder einem Rechenzentrum, und nicht als Arbeitsplatzrechner genutzt. Für Server stehen unterschiedliche Betriebssysteme zur Verfügung, unter anderem Unix bzw. Linux, Microsoft Windows und Novell Netware. Dieser Baustein betrachtet Sicherheitsaspekte, die unabhängig vom eingesetzten Betriebssystem für Server relevant sind. Für betriebssystemspezifische Sicherheitsaspekte existieren in den IT-Grundschutz-Katalogen eigenständige Bausteine, die zusätzlich auf die jeweils betroffenen Server anzuwenden sind. Die netzspezifischen Aspekte des Servereinsatzes werden im Baustein B 4.1 Heterogene Netze behandelt.

3.1.2 Gefährdungslage

Wie jedes IT-System ist auch ein Server vielfältigen Gefahren ausgesetzt. Generell gilt, dass die Gefährdungslage einzelner Rechner immer auch vom Einsatzszenario, beispielsweise der Nutzung als Dateiserver, Terminalserver bzw. Authentisierungsserver, abhängt und diese Einzelgefährdungen auch in die Gefährdung des Gesamtsystems eingehen.

Für den IT-Grundschutz eines Servers werden folgende typische Gefährdungen angenommen:

Höhere Gewalt

G 1.1	Personalausfall
G 1.2	Ausfall von IT-Systemen

Organisatorische Mängel

G 2.7	Unerlaubte Ausübung von Rechten
G 2.9	Mangelhafte Anpassung an Veränderungen beim IT-Einsatz
G 2.36	Ungeeignete Einschränkung der Benutzerumgebung

Menschliche Fehlhandlungen

G 3.2	Fahrlässige Zerstörung von Gerät oder Daten
G 3.3	Nichtbeachtung von Sicherheitsmaßnahmen
G 3.5	Unbeabsichtigte Leitungsbeschädigung

G 3.6	Gefährdung durch Reinigungs- oder Fremdpersonal
G 3.8	Fehlerhafte Nutzung von IT-Systemen
G 3.9	Fehlerhafte Administration von IT-Systemen

Technisches Versagen

G 4.1	Ausfall der Stromversorgung
G 4.6	Spannungsschwankungen/Überspannung/Unterspannung
G 4.7	Defekte Datenträger
G 4.10	Komplexität der Zugangsmöglichkeiten zu vernetzten IT-Systemen
G 4.13	Verlust gespeicherter Daten
G 4.20	Datenverlust bei erschöpftem Speichermedium
G 4.22	Software-Schwachstellen oder -Fehler
G 4.39	Software-Konzeptionsfehler

Vorsätzliche Handlungen

G 5.1	Manipulation oder Zerstörung von Geräten oder Zubehör
G 5.2	Manipulation an Informationen oder Software
G 5.7	Abhören von Leitungen
G 5.9	Unberechtigte IT-Nutzung
G 5.15	Neugierige Mitarbeiter
G 5.18	Systematisches Ausprobieren von Passwörtern
G 5.19	Missbrauch von Benutzerrechten
G 5.20	Missbrauch von Administratorrechten
G 5.21	Trojanische Pferde
G 5.23	Schadprogramme
G 5.26	Analyse des Nachrichtenflusses
G 5.40	Abhören von Räumen mittels Rechner mit Mikrofon
G 5.71	Vertraulichkeitsverlust schützenswerter Informationen
G 5.75	Überlastung durch eingehende E-Mails
G 5.85	Integritätsverlust schützenswerter Informationen

3.1.3 Maßnahmen

Um den betrachteten Informationsverbund abzusichern, müssen zusätzlich zu diesem Baustein noch weitere Bausteine umgesetzt werden, gemäß den Ergebnissen der Model-

lierung nach IT-Grundschutz.

Für den erfolgreichen Aufbau eines Servers sind eine Reihe von Maßnahmen umzusetzen, beginnend mit der Konzeption über die Installation bis zum Betrieb. Ein besonderes Gewicht ist dabei auf die konzeptionellen Planungsmaßnahmen zu legen, wenn der Server im Rahmen des Aufbaus eines neuen servergestützten Netzes installiert wird. Sofern die Installation dagegen als Ausbau eines schon existierenden Netzes erfolgt, können sich die Planungsmaßnahmen häufig darauf beschränken, auf die Konformität des neuen Servers mit den schon vorhandenen Strukturen zu achten. Die Maßnahmen zur Beschaffung und zum Betrieb des Servers sind dagegen in jedem Fall umzusetzen. Die Schritte, die zum Schutz eines Servers zu durchlaufen sind, sowie die Maßnahmen, die in den jeweiligen Schritten beachtet werden sollten, sind im Folgenden aufgeführt.

Planung und Konzeption

M 1.28	(B)	Lokale unterbrechungsfreie Stromversorgung
M 2.314	(Z)	Verwendung von hochverfügbaren Architekturen für Server
M 2.315	(A)	Planung des Servereinsatzes
M 2.316	(A)	Festlegen einer Sicherheitsrichtlinie für einen allgemeinen Server
M 4.250	(Z)	Auswahl eines zentralen, netzbasierten Authentisierungsdienstes
M 5.10	(A)	Restriktive Rechtevergabe
M 1.138	(Z)	Einsatz von RADIUS-Servern

Beschaffung

M 2.317	(C)	Beschaffungskriterien für einen Server

Umsetzung

M 2.32	(Z)	Einrichtung einer eingeschränkten Benutzerumgebung
M 2.204	(A)	Verhinderung ungesicherter Netzzugänge
M 2.318	(A)	Sichere Installation eines Servers
M 4.7	(A)	Änderung voreingestellter Passwörter
M 4.15	(A)	Gesichertes Login
M 4.16	(A)	Zugangsbeschränkungen für Accounts und / oder Terminals
M 4.17	(A)	Sperren und Löschen nicht benötigter Accounts und Terminals
M 4.40	(C)	Verhinderung der unautorisierten Nutzung des Rechnermikrofons
M 4.237	(A)	Sichere Grundkonfiguration eines IT-Systems

M 4.305	(B)	Einsatz von Speicherbeschränkungen (Quotas)

Betrieb

M 2.22	(A)	Hinterlegen des Passwortes
M 2.273	(A)	Zeitnahes Einspielen sicherheitsrelevanter Patches und Updates
M 4.24	(A)	Sicherstellung einer konsistenten Systemverwaltung
M 4.93	(Z)	Regelmäßige Integritätsprüfung
M 4.238	(A)	Einsatz eines lokalen Paketfilters
M 4.239	(A)	Sicherer Betrieb eines Servers
M 4.240	(Z)	Einrichten einer Testumgebung für einen Server
M 5.8	(B)	Regelmäßiger Sicherheitscheck des Netzes
M 5.9	(B)	Protokollierung am Server

Aussonderung

M 2.319	(C)	Migration eines Servers
M 2.320	(A)	Geregelte Außerbetriebnahme eines Servers

Notfallvorsorge

M 6.24	(A)	Erstellen eines Notfall-Bootmediums
M 6.96	(A)	Notfallvorsorge für einen Server

3.2 Server unter Unix (B 3.102)

3.2.1 Beschreibung

Unix-Server sind Rechner mit dem Betriebssystem Unix, die in einem Netz Dienste anbieten, die von anderen IT-Systemen in Anspruch genommen werden können. Das erste Unix-System wurde Anfang der 1970er Jahre entwickelt. Mittlerweile gibt es eine Vielzahl von Betriebssystemen, die der Unix-Familie zugeordnet werden. Hierbei wird zwischen

- klassischen Unix-Systemen oder Unix-Derivaten,
- zertifizierten UNIX-Systemen (UNIX ist ein Warenzeichen der Open Group, das nur zertifizierte Systeme tragen dürfen, die die entsprechende Spezifikation erfüllen) und
- funktionellen Unix-Systemen oder unix-ähnlichen Systemen.

Beispiele für klassische Unix-Systeme sind die BSD-Reihe (FreeBSD, OpenBSD und NetBSD), Solaris und AIX. Linux ist kein klassisches Unix (der Kernel basiert nicht auf dem ursprünglichen Quelltext, aus dem sich die verschiedenen Unix-Derivate entwickelt haben), sondern ein funktionelles Unix-System. In diesem Baustein werden alle Betriebssysteme der Unix-Familie betrachtet, also auch Linux als funktionelles Unix-System.

In diesem Baustein werden ausschließlich die für einen Unix-Server spezifischen Gefährdungen und Maßnahmen beschrieben, daher sind zusätzlich noch diejenigen für allgemeine Server aus Baustein B 3.101 zu betrachten.

3.2.2 Gefährdungslage

Für den IT-Grundschutz eines Unix-Servers werden folgende Gefährdungen angenommen:

Organisatorische Mängel

G 2.15	Vertraulichkeitsverlust schutzbedürftiger Daten im Unix-System
G 2.36	Ungeeignete Einschränkung der Benutzerumgebung

Menschliche Fehlhandlungen

G 3.10	Falsches Exportieren von Dateisystemen unter Unix
G 3.11	Fehlerhafte Konfiguration von sendmail

Technisches Versagen

G 4.11	Fehlende Authentisierungsmöglichkeit zwischen NIS-Server und NIS-Client
G 4.12	Fehlende Authentisierungsmöglichkeit zwischen X-Server und X-Client

Vorsätzliche Handlungen

G 5.41	Mißbräuchliche Nutzung eines Unix-Systems mit Hilfe von uucp
G 5.89	Hijacking von Netz-Verbindungen

3.2.3 Maßnahmen

Um den betrachteten Informationsverbund abzusichern, müssen zusätzlich zu diesem Baustein noch weitere Bausteine umgesetzt werden, gemäß den Ergebnissen der Modellierung nach IT-Grundschutz.

Für den erfolgreichen Aufbau eines Servers unter Unix sind eine Reihe von Maßnahmen umzusetzen, beginnend mit der Konzeption über die Beschaffung bis zum Betrieb dieses Servers. Die Schritte, die dabei durchlaufen werden sollten, sowie die Maßnahmen, die in den jeweiligen Schritten beachtet werden sollten, sind im folgenden aufgeführt.

Planung und Konzeption

M 2.33	(Z)	Aufteilung der Administrationstätigkeiten unter Unix
M 4.13	(A)	Sorgfältige Vergabe von IDs
M 4.18	(A)	Administrative und technische Absicherung des Zugangs zum Monitor- und Single-User-Modus
M 5.16	(B)	Übersicht über Netzdienste
M 5.34	(Z)	Einsatz von Einmalpasswörtern
M 5.64	(Z)	Secure Shell
M 5.83	(Z)	Sichere Anbindung eines externen Netzes mit Linux FreeS/WAN

Umsetzung

M 4.9	(A)	Einsatz der Sicherheitsmechanismen von X-Window
M 4.14	(A)	Obligatorischer Passwortschutz unter Unix
M 4.19	(A)	Restriktive Attributvergabe bei Unix-Systemdateien und -verzeichnissen
M 4.20	(B)	Restriktive Attributvergabe bei Unix-Benutzerdateien und -verzeichnissen
M 4.21	(A)	Verhinderung des unautorisierten Erlangens von Administratorrechten
M 4.22	(Z)	Verhinderung des Vertraulichkeitsverlusts schutzbedürftiger Daten im Unix-System
M 4.23	(B)	Sicherer Aufruf ausführbarer Dateien
M 4.105	(A)	Erste Maßnahmen nach einer Unix-Standardinstallation
M 4.106	(A)	Aktivieren der Systemprotokollierung
M 5.17	(A)	Einsatz der Sicherheitsmechanismen von NFS
M 5.18	(A)	Einsatz der Sicherheitsmechanismen von NIS
M 5.19	(A)	Einsatz der Sicherheitsmechanismen von sendmail
M 5.20	(A)	Einsatz der Sicherheitsmechanismen von rlogin, rsh und rcp
M 5.21	(A)	Sicherer Einsatz von telnet, ftp, tftp und rexec

M 5.35	(A)	Einsatz der Sicherheitsmechanismen von UUCP
M 5.72	(A)	Deaktivieren nicht benötigter Netzdienste

Betrieb

M 4.25	(A)	Einsatz der Protokollierung im Unix-System
M 4.26	(C)	Regelmäßiger Sicherheitscheck des Unix-Systems

Notfallvorsorge

M 6.31	(A)	Verhaltensregeln nach Verlust der Systemintegrität

3.3 Sicherheitsgateway (Firewall) (B 3.301)

3.3.1 Beschreibung

Ein Sicherheitsgateway (oft auch Firewall genannt) ist ein System aus soft- und hardwaretechnischen Komponenten, um IP-Netze sicher zu koppeln. Dazu wird die technisch mögliche auf die in einer IT-Sicherheitsleitlinie ordnungsgemäß definierte Kommunikation eingeschränkt. Sicherheit bei der Netzkopplung bedeutet hierbei die ausschließliche Zulassung erwünschter Zugriffe oder Datenströme zwischen verschiedenen Netzen.

Sicherheitsgateways werden am zentralen Übergang zwischen zwei unterschiedlich vertrauenswürdigen Netzen eingesetzt. Unterschiedlich vertrauenswürdige Netze stellen dabei nicht unbedingt nur die Kombination Internet-Intranet dar. Vielmehr können auch zwei organisationsinterne Netze unterschiedlich hohen Schutzbedarf besitzen, z. B. bei der Trennung des Bürokommunikationsnetzes vom Netz der Personalabteilung, in dem besonders schutzwürdige, personenbezogene Daten übertragen werden.

Die Verwendung des Begriffs Sicherheitsgateway anstatt des üblicherweise verwendeten Begriffs „Firewall“ soll verdeutlichen, dass zur Absicherung von Netzübergängen heute oft nicht mehr ein einzelnes Gerät verwendet wird, sondern eine ganze Reihe von IT-Systemen, die unterschiedliche Aufgaben übernehmen, z. B. Paketfilterung, Schutz vor Viren oder die Überwachung des Netzverkehrs („Intrusion Detection“).

In diesem Baustein werden ausschließlich die für ein Sicherheitsgateway spezifischen Gefährdungen und Maßnahmen beschrieben. Zusätzlich sind noch die Gefährdungen und Maßnahmen zu betrachten, die für das IT-System, mit dem das Sicherheitsgateway realisiert wird, spezifisch sind. Oftmals werden Komponenten von Sicherheitsgateways auf einem Unix-System implementiert. In diesem Fall sind zusätzlich zu den im Folgenden beschriebenen Gefährdungen und Maßnahmen die in Baustein B 3.102 Server unter Unix beschriebenen zu beachten.

3.3.2 Gefährdungslage

Für den IT-Grundschutz eines Sicherheitsgateways werden die folgenden Gefährdungen angenommen:

Organisatorische Mängel

G 2.24	Vertraulichkeitsverlust schutzbedürftiger Daten des zu schützenden Netzes
G 2.101	Unzureichende Notfallvorsorge bei einem Sicherheitsgateway

Menschliche Fehlhandlungen

G 3.3	Nichtbeachtung von Sicherheitsmaßnahmen
G 3.9	Fehlerhafte Administration von IT-Systemen
G 3.38	Konfigurations- und Bedienungsfehler

Technisches Versagen

G 4.8	Bekanntwerden von Softwareschwachstellen
G 4.10	Komplexität der Zugangsmöglichkeiten zu vernetzten IT-Systemen
G 4.11	Fehlende Authentisierungsmöglichkeit zwischen NIS-Server und NIS-Client
G 4.12	Fehlende Authentisierungsmöglichkeit zwischen X-Server und X-Client
G 4.20	Datenverlust bei erschöpftem Speichermedium
G 4.22	Software-Schwachstellen oder -Fehler
G 4.39	Software-Konzeptionsfehler

Vorsätzliche Handlungen

G 5.2	Manipulation an Informationen oder Software
G 5.9	Unberechtigte IT-Nutzung
G 5.18	Systematisches Ausprobieren von Passwörtern
G 5.24	Wiedereinspielen von Nachrichten
G 5.25	Maskerade
G 5.28	Verhinderung von Diensten
G 5.39	Eindringen in Rechnersysteme über Kommunikationskarten
G 5.48	IP-Spoofing
G 5.49	Missbrauch des Source-Routing

G 5.50	Missbrauch des ICMP-Protokolls
G 5.51	Missbrauch der Routing-Protokolle
G 5.78	DNS-Spoofing und Pharming
G 5.143	Man-in-the-Middle-Angriff

3.3.3 Maßnahmen

Um den betrachteten Informationsverbund abzusichern, müssen zusätzlich zu diesem Baustein noch weitere Bausteine umgesetzt werden, gemäß den Ergebnissen der Modellierung nach IT-Grundschutz.

Ein Sicherheitsgateway schützt nicht vor Angriffen, die innerhalb des internen Netzes erfolgen. Um das interne Netz gegen Angriffe von Innentätern zu schützen, müssen auch beim Einsatz eines Sicherheitsgateways alle erforderlichen Sicherheitsmaßnahmen umgesetzt sein. Wenn es sich bei dem internen Netz beispielsweise um ein Unix- bzw. PC-Netz handelt, sind die in den jeweiligen Bausteinen beschriebenen Sicherheitsmaßnahmen umzusetzen.

Das Sicherheitsgateway sollte in einem separaten Serverraum aufgestellt werden. Hierbei zu realisierende Maßnahmen sind in Baustein B 2.4 Serverraum beschrieben. Wenn kein Serverraum zur Verfügung steht, kann das Sicherheitsgateway alternativ in einem Serverschrank aufgestellt werden (siehe Baustein B 2.7 Schutzschränke). Soll das Sicherheitsgateway nicht in Eigenregie, sondern von einem Dienstleister betrieben werden, so ist der Baustein B 1.11 Outsourcing anzuwenden. Insbesondere sollten die Empfehlungen in M 5.116 Integration eines E-Mailservers in ein Sicherheitsgateway beachtet werden.

Planung und Konzeption

M 2.70	(A)	Entwicklung eines Konzepts für Sicherheitsgateways
M 2.71	(A)	Festlegung einer Policy für ein Sicherheitsgateway
M 2.301	(Z)	Outsourcing des Sicherheitsgateway

Beschaffung

M 2.73	(A)	Auswahl geeigneter Grundstrukturen für Sicherheitsgateways
M 2.74	(A)	Geeignete Auswahl eines Paketfilters
M 2.75	(A)	Geeignete Auswahl eines Application-Level-Gateways
M 2.299	(A)	Erstellung einer Sicherheitsrichtlinie für ein Sicherheitsgateway

Umsetzung

M 2.76	(A)	Auswahl und Einrichtung geeigneter Filterregeln
M 2.77	(A)	Integration von Servern in das Sicherheitsgateway
M 3.43	(C)	Schulung der Administratoren des Sicherheitsgateways
M 4.224	(Z)	Integration von VPN-Komponenten in ein Sicherheitsgateway

Betrieb

M 2.78	(A)	Sicherer Betrieb eines Sicherheitsgateways
M 2.302	(Z)	Sicherheitsgateways und Hochverfügbarkeit
M 4.47	(A)	Protokollierung der Sicherheitsgateway-Aktivitäten
M 4.100	(C)	Sicherheitsgateways und aktive Inhalte
M 4.101	(C)	Sicherheitsgateways und Verschlüsselung
M 4.222	(B)	Festlegung geeigneter Einstellungen von Sicherheitsproxies
M 4.223	(B)	Integration von Proxy-Servern in das Sicherheitsgateway
M 4.225	(Z)	Einsatz eines Protokollierungsservers in einem Sicherheitsgateway
M 4.226	(Z)	Integration von Virenscannern in ein Sicherheitsgateway
M 4.227	(C)	Einsatz eines lokalen NTP-Servers zur Zeitsynchronisation
M 5.39	(A)	Sicherer Einsatz der Protokolle und Dienste
M 5.46	(A)	Einsatz von Stand-alone-Systemen zur Nutzung des Internets
M 5.59	(A)	Schutz vor DNS-Spoofing
M 5.70	(A)	Adressumsetzung - NAT (Network Address Translation)
M 5.71	(Z)	Intrusion Detection und Intrusion Response Systeme
M 5.115	(Z)	Integration eines Webservers in ein Sicherheitsgateway
M 5.116	(Z)	Integration eines E-Mailservers in ein Sicherheitsgateway
M 5.117	(Z)	Integration eines Datenbank-Servers in ein Sicherheitsgateway
M 5.118	(Z)	Integration eines DNS-Servers in ein Sicherheitsgateway
M 5.119	(Z)	Integration einer Web-Anwendung mit Web-, Applikations- und Datenbank-Server in ein Sicherheitsgateway
M 5.120	(A)	Behandlung von ICMP am Sicherheitsgateway

Aussonderung

M 2.300	(C)	Sichere Außerbetriebnahme oder Ersatz von Komponenten eines Sicherheitsgateways

Notfallvorsorge

M 6.94	(C)	Notfallvorsorge bei Sicherheitsgateways

3.4 Router und Switches (B 3.302)

3.4.1 Beschreibung

Netze spielen eine immer wichtigere Rolle als Teile der IT-Infrastruktur, weil Anwendungen heutzutage vermehrt über lokale Netze oder Weitverkehrsnetze betrieben werden. Die Verfügbarkeit, Integrität und Vertraulichkeit der Netze muss sichergestellt sein und mindestens den Anforderungen der Anwendungen an den Schutz dieser drei Grundwerte der IT-Sicherheit entsprechen.

Ein Netz besteht aus aktiver und passiver Netztechnik. Als passive Netztechnik wird in erster Linie die strukturierte Verkabelung verstanden. Hierzu gehören Patch-Felder (über Steckfelder konfigurierbare Kabelverteiler), Schutzschränke und Anschlussdosen am Arbeitsplatz. Zur aktiven Netztechnik gehören beispielsweise Hubs, Bridges, Switches und Router. In modernen Netzen ersetzen Switches heutzutage vielfach Hubs sowie Bridges. Ein Ausfall einer oder mehrerer Komponenten der aktiven Netztechnik (Router und Switches) kann zum kompletten Stillstand der gesamten IT-Infrastruktur führen. Da diese Komponenten die Basis und das Rückgrat der IT-Infrastruktur bilden, müssen Router und Switches vor unerlaubten Zugriffen und Manipulationen geschützt werden.

3.4.2 Gefährdungslage

Neben den Gefährdungen, die generell für den Großteil der IT-Systeme gelten, existieren für aktive Netzkomponenten eine Reihe spezieller Gefährdungen.

Diese Gefährdungen basieren oft auf bekannten Schwachstellen in den verwendeten Protokollen, wie TCP, UDP, IP oder ICMP. Durch Schwachstellen in dynamischen Routing-Protokollen können beispielsweise Routing-Tabellen auf Routern modifiziert werden. Die oft fehlende oder unzureichende Möglichkeit zur Authentisierung auf aktiven Netzkomponenten ist als weitere Gefährdung anzufügen.

Aktive Netzkomponenten werden oft mit einer unsicheren Default-Konfiguration ausgeliefert (siehe G 4.49 Unsichere Default-Einstellungen auf Routern und Switches), die bei der Inbetriebnahme der Geräte geprüft werden sollte. Für die sichere Trennung von Teilnetzen mit unterschiedlichem Schutzbedarf wird gelegentlich die Nutzung von virtuellen Netzen (VLANs) vorgeschlagen. Es sind jedoch einige Angriffsmethoden bekannt, die es ermöglichen, die Grenzen zwischen VLANs zu überwinden und unberechtigt auf andere VLANs zuzugreifen (siehe G 5.115 Überwindung der Grenzen zwischen VLANs).

Nachfolgend ist die Gefährdungslage beim Einsatz von Routern und Switches als Übersicht dargestellt:

Organisatorische Mängel

G 2.1	Fehlende oder unzureichende Regelungen
G 2.3	Fehlende, ungeeignete, inkompatible Betriebsmittel
G 2.4	Unzureichende Kontrolle der Sicherheitsmaßnahmen
G 2.22	Fehlende Auswertung von Protokolldaten
G 2.27	Fehlende oder unzureichende Dokumentation
G 2.44	Inkompatible aktive und passive Netzkomponenten
G 2.54	Vertraulichkeitsverlust durch Restinformationen
G 2.98	Fehlerhafte Planung und Konzeption des Einsatzes von Routern und Switches

Menschliche Fehlhandlungen

G 3.64	Fehlerhafte Konfiguration von Routern und Switches
G 3.65	Fehlerhafte Administration von Routern und Switches

Technisches Versagen

G 4.8	Bekanntwerden von Softwareschwachstellen
G 4.49	Unsichere Default-Einstellungen auf Routern und Switches

Vorsätzliche Handlungen

G 5.4	Diebstahl
G 5.51	Missbrauch der Routing-Protokolle
G 5.66	Unberechtigter Anschluss von IT-Systemen an ein Netz
G 5.112	Manipulation von ARP-Tabellen
G 5.113	MAC-Spoofing
G 5.114	Missbrauch von Spanning Tree
G 5.115	Überwindung der Grenzen zwischen VLANs

3.4.3 Maßnahmen

Die diesem Baustein zugeordneten Sicherheitsmaßnahmen orientieren sich an dem Lebenszyklus der aktiven Netzkomponenten.

Planung und Konzeption

M 2.276	(Z)	Funktionsweise eines Routers
M 2.277	(Z)	Funktionsweise eines Switches
M 2.278	(Z)	Typische Einsatzszenarien von Routern und Switches
M 2.279	(A)	Erstellung einer Sicherheitsrichtlinie für Router und Switches

Beschaffung

M 2.280	(C)	Kriterien für die Beschaffung und geeignete Auswahl von Routern und Switches

Umsetzung

M 1.43	(A)	Gesicherte Aufstellung aktiver Netzkomponenten
M 3.38	(B)	Administratorenschulung für Router und Switches
M 4.201	(A)	Sichere lokale Grundkonfiguration von Routern und Switches
M 4.202	(A)	Sichere Netz-Grundkonfiguration von Routern und Switches
M 4.203	(A)	Konfigurations-Checkliste für Router und Switches
M 5.111	(C)	Einrichtung von Access Control Lists auf Routern

Betrieb

M 2.281	(A)	Dokumentation der Systemkonfiguration von Routern und Switches
M 2.282	(A)	Regelmäßige Kontrolle von Routern und Switches
M 2.283	(B)	Software-Pflege auf Routern und Switches
M 4.204	(C)	Sichere Administration von Routern und Switches
M 4.205	(C)	Protokollierung bei Routern und Switches
M 4.206	(C)	Sicherung von Switch-Ports
M 5.112	(C)	Sicherheitsaspekte von Routing-Protokollen

Aussonderung

M 2.284	(C)	Sichere Außerbetriebnahme von Routern und Switches

Notfallvorsorge

M 6.91	(C)	Datensicherung und Recovery bei Routern und Switches

M 6.92	(C)	Notfallvorsorge bei Routern und Switches

4 Bausteine für Netze

4.1 Heterogene Netze (B 4.1)

4.1.1 Beschreibung

Ein lokales Netz setzt sich aus der Verkabelung (d. h. den passiven Netzkomponenten Kabel und den Verbindungselementen) sowie den aktiven Netzkomponenten zur Netzkopplung zusammen. Generell können dabei unterschiedliche Verkabelungstypen wie auch unterschiedliche aktive Netzkomponenten in ein LAN integriert werden. Als aktive Netzkomponenten werden alle Netzkomponenten bezeichnet, die eine eigene (Netz-) Stromversorgung benötigen. Dazu gehören unter anderem Repeater, Brücken, Switches, Router, Gateways. Als passive Netzkomponenten werden alle Netzkomponenten betrachtet, die keine eigene Netzstrom-Versorgung benötigen. Dazu gehören z. B. Kabel, Verteilerschränke, Patchfelder, Steckverbinder.

4.1.2 Gefährdungslage

Für den IT-Grundschutz eines heterogenen Netzes werden die folgenden Gefährdungen angenommen:

Höhere Gewalt

G 1.2	Ausfall von IT-Systemen
G 1.3	Blitz
G 1.4	Feuer
G 1.5	Wasser
G 1.7	Unzulässige Temperatur und Luftfeuchte
G 1.8	Staub, Verschmutzung

Organisatorische Mängel

G 2.7	Unerlaubte Ausübung von Rechten
G 2.9	Mangelhafte Anpassung an Veränderungen beim IT-Einsatz
G 2.22	Fehlende Auswertung von Protokolldaten
G 2.27	Fehlende oder unzureichende Dokumentation
G 2.32	Unzureichende Leitungskapazitäten
G 2.44	Inkompatible aktive und passive Netzkomponenten
G 2.45	Konzeptionelle Schwächen des Netzes
G 2.46	Überschreiten der zulässigen Kabel- bzw. Buslänge oder der Ringgröße

Menschliche Fehlhandlungen

G 3.2	Fahrlässige Zerstörung von Gerät oder Daten
G 3.3	Nichtbeachtung von Sicherheitsmaßnahmen
G 3.5	Unbeabsichtigte Leitungsbeschädigung
G 3.6	Gefährdung durch Reinigungs- oder Fremdpersonal
G 3.8	Fehlerhafte Nutzung von IT-Systemen
G 3.9	Fehlerhafte Administration von IT-Systemen
G 3.28	Ungeeignete Konfiguration der aktiven Netzkomponenten
G 3.29	Fehlende oder ungeeignete Segmentierung

Technisches Versagen

G 4.1	Ausfall der Stromversorgung
G 4.10	Komplexität der Zugangsmöglichkeiten zu vernetzten IT-Systemen
G 4.31	Ausfall oder Störung von Netzkomponenten

Vorsätzliche Handlungen

G 5.1	Manipulation oder Zerstörung von Geräten oder Zubehör
G 5.2	Manipulation an Informationen oder Software
G 5.4	Diebstahl
G 5.5	Vandalismus
G 5.6	Anschlag
G 5.7	Abhören von Leitungen
G 5.8	Manipulation an Leitungen
G 5.9	Unberechtigte IT-Nutzung
G 5.18	Systematisches Ausprobieren von Passwörtern
G 5.20	Missbrauch von Administratorrechten
G 5.28	Verhinderung von Diensten
G 5.66	Unberechtigter Anschluss von IT-Systemen an ein Netz
G 5.67	Unberechtigte Ausführung von Netzmanagement-Funktionen
G 5.68	Unberechtigter Zugang zu den aktiven Netzkomponenten

4.1.3 Maßnahmen

Planung und Konzeption

M 2.139	(A)	Ist-Aufnahme der aktuellen Netzsituation
M 2.140	(Z)	Analyse der aktuellen Netzsituation
M 2.141	(B)	Entwicklung eines Netzkonzeptes
M 2.142	(B)	Entwicklung eines Netz-Realisierungsplans
M 4.79	(A)	Sichere Zugriffsmechanismen bei lokaler Administration
M 5.2	(A)	Auswahl einer geeigneten Netz-Topologie
M 5.13	(A)	Geeigneter Einsatz von Elementen zur Netzkopplung
M 5.60	(A)	Auswahl einer geeigneten Backbone-Technologie
M 5.61	(A)	Geeignete physikalische Segmentierung
M 5.62	(Z)	Geeignete logische Segmentierung
M 5.77	(Z)	Bildung von Teilnetzen

Umsetzung

M 4.7	(A)	Änderung voreingestellter Passwörter
M 4.80	(B)	Sichere Zugriffsmechanismen bei Fernadministration
M 4.82	(A)	Sichere Konfiguration der aktiven Netzkomponenten
M 5.7	(A)	Netzverwaltung

Betrieb

M 4.81	(B)	Audit und Protokollierung der Aktivitäten im Netz
M 4.83	(C)	Update/Upgrade von Soft- und Hardware im Netzbereich

Notfallvorsorge

M 6.52	(A)	Regelmäßige Sicherung der Konfigurationsdaten aktiver Netzkomponenten
M 6.53	(Z)	Redundante Auslegung der Netzkomponenten
M 6.54	(B)	Verhaltensregeln nach Verlust der Netzintegrität
M 6.75	(Z)	Redundante Kommunikationsverbindungen

4.2 Netz- und Systemmanagement (B 4.2)

4.2.1 Beschreibung

Ein Managementsystem für ein im Allgemeinen lokales Rechnernetz (LAN, VLAN) dient dazu, möglichst alle im lokale Netz angesiedelten Hard- und Software-Komponenten

zentral zu verwalten. Ein solches System soll den Systemverwalter maximal in seiner täglichen Arbeit unterstützen. Grundsätzlich kann zwischen Netzmanagement und Systemmanagement unterschieden werden. Die Unterschiede ergeben sich durch die jeweils verwalteten Komponenten.

Netzmanagement umfasst die Gesamtheit der Vorkehrungen und Aktivitäten zur Sicherstellung des effektiven Einsatzes eines Netzes. Hierzu gehört beispielsweise die Überwachung der Netzkomponenten auf ihre korrekte Funktion, das Monitoring der Netzperformance und die zentrale Konfiguration der Netzkomponenten. Netzmanagement ist in erster Linie eine organisatorische Problemstellung, deren Lösung lediglich mit technischen Mitteln, einem Netzmanagementsystem, unterstützt werden kann.

Systemmanagement befasst sich in erster Linie mit dem Management verteilter IT-Systeme. Hierzu gehören beispielsweise eine zentrale Verwaltung der Benutzer, Softwareverteilung, Management der Anwendungen usw. In einigen Bereichen, wie z. B. dem Konfigurationsmanagement (dem Überwachen und Konsolidieren von Konfigurationen eines Systems oder einer Netzkomponente), sind Netz- und Systemmanagement nicht klar zu trennen.

4.2.2 Gefährdungslage

Für den IT-Grundschutz eines Managementsystems werden die folgenden Gefährdungen angenommen:

Höhere Gewalt

G 1.1	Personalausfall
G 1.2	Ausfall von IT-Systemen

Organisatorische Mängel

G 2.27	Fehlende oder unzureichende Dokumentation
G 2.32	Unzureichende Leitungskapazitäten
G 2.59	Betreiben von nicht angemeldeten Komponenten
G 2.60	Fehlende oder unzureichende Strategie für das Netz- und Systemmanagement
G 2.61	Unberechtigte Sammlung personenbezogener Daten

Menschliche Fehlhandlungen

G 3.9	Fehlerhafte Administration von IT-Systemen
G 3.28	Ungeeignete Konfiguration der aktiven Netzkomponenten
G 3.34	Ungeeignete Konfiguration des Managementsystems
G 3.35	Server im laufenden Betrieb ausschalten

G 3.36	Fehlinterpretation von Ereignissen

Technisches Versagen

G 4.31	Ausfall oder Störung von Netzkomponenten
G 4.38	Ausfall von Komponenten eines Netz- und Systemmanagementsystems

Vorsätzliche Handlungen

G 5.2	Manipulation an Informationen oder Software
G 5.8	Manipulation an Leitungen
G 5.9	Unberechtigte IT-Nutzung
G 5.18	Systematisches Ausprobieren von Passwörtern
G 5.28	Verhinderung von Diensten
G 5.66	Unberechtigter Anschluss von IT-Systemen an ein Netz
G 5.67	Unberechtigte Ausführung von Netzmanagement-Funktionen
G 5.86	Manipulation von Managementparametern

4.2.3 Maßnahmen

Um den betrachteten IT-Verbund abzusichern, müssen zusätzlich zu diesem Baustein noch weitere Bausteine umgesetzt werden, gemäß den Ergebnissen der Modellierung nach IT-Grundschutz.

Das zu verwaltende System besteht aus einzelnen Rechnern, Netzkoppelelementen und dem physikalischen Netz. Jede dieser Komponenten ist ein potentielles Sicherheitsrisiko für das Gesamtsystem. Diese Risiken können im allgemeinen alleine durch die Einführung von Managementsoftware nicht vollständig beseitigt werden. Dies gilt schon deshalb, weil in der Regel nicht alle Systeme in gleichem Maße durch ein Managementsystem erfasst werden. Grundvoraussetzung für die Systemsicherheit ist hier einerseits die Definition und andererseits die Realisierung einer organisationsweiten Sicherheitsrichtlinie, die sich im betrachteten Fall insbesondere in der Konfiguration von Hard- und Software niederschlagen muss. Aus diesem Grund sollten insbesondere die Maßnahmen der Bausteine der Schicht 3 (IT-Systeme) betrachtet werden. Als Ausgangsbaustein kann der Baustein B 4.1 Heterogene Netze dienen.

Da Managementsysteme von einem zentralistischen Ansatz ausgehen, kommt der zentralen Managementstation eine besondere Bedeutung unter Sicherheitsgesichtspunkten zu und ist daher besonders zu schützen. Zentrale Komponenten eines Managementsystems

sollten daher in Räumen aufgestellt werden, die den Anforderungen an einen Serverraum (vergleiche Baustein B 2.4 Serverraum) entsprechen.

Planung und Konzeption

M 2.143	(A)	Entwicklung eines Netzmanagementkonzeptes
M 2.144	(A)	Geeignete Auswahl eines Netzmanagement-Protokolls
M 2.168	(A)	IT-System-Analyse vor Einführung eines Systemmanagement-systems
M 2.169	(A)	Entwickeln einer Systemmanagementstrategie

Beschaffung

M 2.145	(B)	Anforderungen an ein Netzmanagement-Tool
M 2.170	(A)	Anforderungen an ein Systemmanagementsystem
M 2.171	(A)	Geeignete Auswahl eines Systemmanagement-Produktes

Umsetzung

M 4.91	(A)	Sichere Installation eines Systemmanagementsystems

Betrieb

M 2.146	(A)	Sicherer Betrieb eines Netzmanagementsystems
M 4.92	(A)	Sicherer Betrieb eines Systemmanagementsystems

Notfallvorsorge

M 6.57	(C)	Erstellen eines Notfallplans für den Ausfall des Management-systems

5 Bausteine für Anwendungen

5.1 Webserver (B 5.4)

5.1.1 Beschreibung

Das Internet ist eines der zentralen Medien der heutigen Informationsgesellschaft. Die Informationsangebote im Internet werden von Servern bereitgestellt, die Daten, meist Dokumente in Form von HTML-Seiten, an entsprechende Clientprogramme ausliefern. Dies erfolgt typischerweise über die Protokolle HTTP (Hypertext Transfer Protocol) oder HTTPS (HTTP über SSL bzw. TLS, d. h. HTTP geschützt durch eine verschlüsselte Verbindung). Neben dem Einsatz im Internet werden Webserver auch in zunehmendem Maße für interne Informationen und Anwendungen in Firmennetzen (Intranet) eingesetzt. Ein Grund dafür ist, dass sie eine einfache und standardisierte Schnittstelle zwischen Server-Anwendungen und Benutzern bieten und entsprechende Client-Software (Webbrowser) für praktisch jede Betriebssystemumgebung kostenlos verfügbar ist.

Die Bezeichnung Webserver (oder auch WWW-Server) wird meist sowohl für das Programm benutzt, welches die HTTP-Anfragen beantwortet, als auch für den Rechner, auf dem dieses Programm läuft. Bei Webservern sind verschiedene Sicherheitsaspekte zu beachten.

Da ein Webserver ein öffentlich zugängliches System darstellt, sind eine sorgfältige Planung vor dem Aufbau eines Webservers und die sichere Installation und Konfiguration des Systems und seiner Netzumgebung von großer Bedeutung. Das Thema Sicherheit umfasst bei Webservern auch deswegen eine relativ große Anzahl von Gebieten, weil auf einem Webserver meist neben der reinen Webserver-Anwendung noch weitere Serveranwendungen vorhanden sind, die zum Betrieb des Webservers erforderlich sind und deren sicherer Betrieb ebenfalls gewährleistet sein muss. Beispielsweise werden die Daten meist über das Netz (etwa per ftp oder scp) auf den Server übertragen oder es wird Zugriff auf eine Datenbank benötigt.

5.1.2 Gefährdungslage

Für den IT-Grundschutz werden bezüglich Schadprogramme die folgenden typischen Gefährdungen betrachtet:

Organisatorische Mängel

G 2.1	Fehlende oder unzureichende Regelungen
G 2.4	Unzureichende Kontrolle der Sicherheitsmaßnahmen
G 2.7	Unerlaubte Ausübung von Rechten
G 2.9	Mangelhafte Anpassung an Veränderungen beim IT-Einsatz
G 2.28	Verstöße gegen das Urheberrecht

G 2.32	Unzureichende Leitungskapazitäten
G 2.37	Unkontrollierter Aufbau von Kommunikationsverbindungen
G 2.96	Veraltete oder falsche Informationen in einem Webangebot
G 2.100	Fehler bei der Beantragung und Verwaltung von Internet-Domainnamen

Menschliche Fehlhandlungen

G 3.1	Vertraulichkeits- oder Integritätsverlust von Daten durch Fehlverhalten
G 3.37	Unproduktive Suchzeiten
G 3.38	Konfigurations- und Bedienungsfehler

Technisches Versagen

G 4.10	Komplexität der Zugangsmöglichkeiten zu vernetzten IT-Systemen
G 4.22	Software-Schwachstellen oder -Fehler
G 4.39	Software-Konzeptionsfehler

Vorsätzliche Handlungen

G 5.2	Manipulation an Informationen oder Software
G 5.19	Missbrauch von Benutzerrechten
G 5.20	Missbrauch von Administratorrechten
G 5.21	Trojanische Pferde
G 5.23	Schadprogramme
G 5.28	Verhinderung von Diensten
G 5.43	Makro-Viren
G 5.48	IP-Spoofing
G 5.78	DNS-Spoofing und Pharming
G 5.87	Web-Spoofing
G 5.88	Missbrauch aktiver Inhalte

5.1.3 Maßnahmen

Um den betrachteten Informationsverbund abzusichern, müssen zusätzlich zu diesem Baustein noch weitere Bausteine umgesetzt werden, gemäß den Ergebnissen der Modellierung nach IT-Grundschutz.

In diesem Baustein werden die für einen Webserver spezifischen Gefährdungen und Maßnahmen beschrieben. Darüber hinaus muss für die Sicherheit des verwendeten Servers der Baustein B 3.101 Allgemeiner Server umgesetzt werden, sowie der Baustein B 3.102 Server unter Unix. Falls das Webangebot Inhalte enthält, die von einer Webanwendung dynamisch aus einer Datenbank erzeugt werden, ist auch der Baustein B 5.7 Datenbanken zu berücksichtigen. Insbesondere dann, wenn der Webserver aus dem Internet heraus angesprochen werden kann, sollte auch Baustein B 1.8 Behandlung von Sicherheitsvorfällen beachtet werden.

Für die sichere Anbindung eines Webservers an öffentliche Netze (z. B. das Internet) ist Baustein B 3.301 Sicherheitsgateway (Firewall) zu betrachten, ebenso wie für den Zusammenschluss mehrerer Intranets zu einem übergreifenden Intranet.

Ein Webserver sollte in einem separaten Serverraum aufgestellt werden. Hierbei zu realisierende Maßnahmen sind in Baustein B 2.4 Serverraum beschrieben.

Planung und Konzeption

M 2.172	(A)	Entwicklung eines Konzeptes für die Web-Nutzung
M 2.173	(A)	Festlegung einer Web-Sicherheitsstrategie
M 2.175	(A)	Aufbau eines Webservers
M 2.271	(A)	Festlegung einer Sicherheitsstrategie für den WWW-Zugang
M 2.272	(A)	Einrichtung eines WWW-Redaktionsteams
M 2.298	(Z)	Verwaltung von Internet-Domainnamen
M 4.34	(Z)	Einsatz von Verschlüsselung, Checksummen oder Digitalen Signaturen
M 4.176	(B)	Auswahl einer Authentisierungsmethode für Webangebote
M 5.64	(Z)	Secure Shell
M 5.66	(B)	Verwendung von TLS/SSL
M 5.69	(A)	Schutz vor aktiven Inhalten

Beschaffung

M 2.176	(B)	Geeignete Auswahl eines Internet Service Providers

Umsetzung

M 4.94	(A)	Schutz der WWW-Dateien
M 4.95	(A)	Minimales Betriebssystem
M 4.96	(Z)	Abschaltung von DNS
M 4.98	(A)	Kommunikation durch Paketfilter auf Minimum beschränken

Betrieb

M 2.174	(A)	Sicherer Betrieb eines Webservers
M 2.273	(A)	Zeitnahes Einspielen sicherheitsrelevanter Patches und Updates
M 4.33	(A)	Einsatz eines Viren-Suchprogramms bei Datenträgeraustausch und Datenübertragung
M 4.64	(C)	Verifizieren der zu übertragenden Daten vor Weitergabe / Beseitigung von Restinformationen
M 4.78	(A)	Sorgfältige Durchführung von Konfigurationsänderungen
M 4.177	(B)	Sicherstellung der Integrität und Authentizität von Softwarepaketen
M 5.59	(A)	Schutz vor DNS-Spoofing

Notfallvorsorge

M 6.88	(B)	Erstellen eines Notfallplans für den Webserver

5.2 Datenbanken (B 5.7)

5.2.1 Beschreibung

Datenbanksysteme (DBS) sind ein weithin genutztes Hilfsmittel zur rechnergestützten Organisation, Erzeugung, Veränderung und Verwaltung großer Datensammlungen und stellen in vielen Unternehmen und Organisationen die zentrale Informationsbasis zu ihrer Aufgabenerfüllung bereit. Ein DBS besteht aus dem so genannten Datenbankmanagement-System (DBMS) und einer oder mehrerer Datenbanken.

Eine Datenbank ist eine Zusammenstellung von Daten samt ihrer Beschreibung (Metadaten), die persistent im DBS abgelegt werden.

Das DBMS bildet die Schnittstelle zwischen den Datenbanken und dient den Benutzern zur Daten-Verwaltung und Veränderung. Die zentralen Aufgaben eines DBMS sind im Wesentlichen die Bereitstellung verschiedener Sichten auf die Daten (Views), die Konsistenzprüfung der Daten (Integritätssicherung), die Autorisationsprüfung, die Behandlung gleichzeitiger Zugriffe verschiedener Benutzer (Synchronisation) und das Bereitstellen einer Datensicherungsmöglichkeit, um im Falle eines Systemausfalls zeitnah Daten wiederherstellen zu können.

5.2.2 Gefährdungslage

Neben den grundlegenden Gefährdungen, die prinzipiell für IT-Systeme gelten, existieren Gefährdungen, die speziell die Verfügbarkeit von Datenbanken sowie die Vertraulichkeit oder die Integrität der gespeicherten Daten bedrohen.

Generell steht die Gefährdungslage in Abhängigkeit vom Einsatzszenario und berechtigten Benutzerkreis. Beispielsweise ergibt sich eine erhöhte Gefährdungslage, wenn, anders als gegenüber identifizierbaren Benutzerkreisen innerhalb des Unternehmens, Zugriffe anonymer Benutzern (z. B. Internet-Zugriffe) erlaubt werden.

Ein weiterer Aspekt ergibt sich aus der steigenden Komplexität des DBMS, der sich unter anderem auch in örtlich weit voneinander getrennter Datenhaltung und den damit einhergehenden Anforderungen an sichere Kommunikationswege und konsistente Daten-Synchronisation begründet.

Für den IT-Grundschutz von Datenbanken werden die folgenden Gefährdungen angenommen:

Organisatorische Mängel

G 2.22	Fehlende Auswertung von Protokolldaten
G 2.26	Fehlendes oder unzureichendes Test- und Freigabeverfahren
G 2.38	Fehlende oder unzureichende Aktivierung von Datenbank-Sicherheitsmechanismen
G 2.39	Mangelhafte Konzeption eines DBMS
G 2.40	Mangelhafte Konzeption des Datenbankzugriffs
G 2.41	Mangelhafte Organisation des Wechsels von Datenbank-Benutzern
G 2.57	Nicht ausreichende Speichermedien für den Notfall
G 2.110	Mangelhafte Organisation bei Versionswechsel und Migration von Datenbanken

Menschliche Fehlhandlungen

G 3.6	Gefährdung durch Reinigungs- oder Fremdpersonal
G 3.16	Fehlerhafte Administration von Zugangs- und Zugriffsrechten
G 3.23	Fehlerhafte Administration eines DBMS
G 3.24	Unbeabsichtigte Datenmanipulation
G 3.80	Fehler bei der Synchronisation von Datenbanken

Technisches Versagen

G 4.26	Ausfall einer Datenbank

G 4.27	Unterlaufen von Zugriffskontrollen über ODBC
G 4.28	Verlust von Daten einer Datenbank
G 4.30	Verlust der Datenbankintegrität/-konsistenz

Vorsätzliche Handlungen

G 5.9	Unberechtigte IT-Nutzung
G 5.10	Missbrauch von Fernwartungszugängen
G 5.18	Systematisches Ausprobieren von Passwörtern
G 5.64	Manipulation an Daten oder Software bei Datenbanksystemen
G 5.65	Verhinderung der Dienste eines Datenbanksystems
G 5.131	SQL-Injection

5.2.3 Maßnahmen

Um den betrachteten IT-Verbund abzusichern, müssen zusätzlich zu diesem Baustein noch weitere Bausteine umgesetzt werden, gemäß den Ergebnissen der Modellierung nach IT-Grundschutz.

Als zentraler Informationsspeicher einer Behörde oder eines Unternehmens empfiehlt es sich, den Datenbank-Server in einem separaten Serverraum aufzustellen oder in einem zentralen Rechenzentrum unterzubringen. Zu realisierende Maßnahmen sind in den Bausteinen B 2.4 Serverraum und B 2.9 Rechenzentrum beschrieben.

Planung und Konzeption

M 2.80	(A)	Erstellung eines Anforderungskatalogs für Standardsoftware
M 2.126	(A)	Erstellung eines Datenbanksicherheitskonzeptes
M 2.132	(A)	Regelung für die Einrichtung von Datenbankbenutzern/-benutzergruppen
M 2.134	(B)	Richtlinien für Datenbank-Anfragen
M 2.363	(B)	Schutz gegen SQL-Injection
M 5.58	(B)	Auswahl und Installation von Datenbankschnittstellen-Treibern

Beschaffung

M 2.124	(A)	Geeignete Auswahl einer Datenbank-Software

Umsetzung

M 2.125	(A)	Installation und Konfiguration einer Datenbank
M 2.135	(C)	Gesicherte Datenübernahme in eine Datenbank
M 4.7	(A)	Änderung voreingestellter Passwörter
M 4.71	(C)	Restriktive Handhabung von Datenbank-Links
M 4.73	(C)	Festlegung von Obergrenzen für selektierbare Datensätze

Betrieb

M 2.31	(A)	Dokumentation der zugelassenen Benutzer und Rechteprofile
M 2.34	(A)	Dokumentation der Veränderungen an einem bestehenden System
M 2.65	(B)	Kontrolle der Wirksamkeit der Benutzer-Trennung am IT-System
M 2.127	(B)	Inferenzprävention
M 2.128	(A)	Zugangskontrolle einer Datenbank
M 2.129	(A)	Zugriffskontrolle einer Datenbank
M 2.130	(A)	Gewährleistung der Datenbankintegrität
M 2.131	(C)	Aufteilung von Administrationstätigkeiten bei Datenbanksystemen
M 2.133	(A)	Kontrolle der Protokolldateien eines Datenbanksystems
M 3.18	(A)	Verpflichtung der Benutzer zum Abmelden nach Aufgabenerfüllung
M 4.67	(B)	Sperren und Löschen nicht benötigter Datenbank-Accounts
M 4.68	(A)	Sicherstellung einer konsistenten Datenbankverwaltung
M 4.69	(B)	Regelmäßiger Sicherheitscheck der Datenbank
M 4.70	(C)	Durchführung einer Datenbanküberwachung
M 4.72	(Z)	Datenbank-Verschlüsselung
M 5.117	(Z)	Integration eines Datenbank-Servers in ein Sicherheitsgateway

Notfallvorsorge

M 6.48	(A)	Verhaltensregeln nach Verlust der Datenbankintegrität
M 6.49	(A)	Datensicherung einer Datenbank
M 6.50	(Z)	Archivierung von Datenbeständen
M 6.51	(B)	Wiederherstellung einer Datenbank

Literatur

[1] BSI. https://www.bsi.bund.de/DE/Home/home_node.html/, 2011. [Online; accessed 18-November-2011].

D. Technologieauswahl

Technologieauswahl

Magnus Monetarus Bank
Cloud-basiertes Internetbanking System

12. März 2012

Inhaltsverzeichnis

1 Ziele **2**

2 Cloud-Umgebung **3**

3 Model-View-Control Prinzip **5**

4 Server-Software (Backend) **6**

5 Entwicklung und Darstellung des Frontend **7**

6 Sichere Kommunikation unter Servern und zu externen Services **8**

7 Frameworks und Extensions **10**

8 Business Process Mining **11**

9 Testumgebung **12**

10 Entwicklungsumgebung **13**

Dokumentenhistorie

Version	Datum	Editoren	Änderung
1.0	29.09.2011	AV, RH	Initiales Dokument
2.0	19.12.2011	AV, RH	Korrekturen aus dem externen Bericht

1 Ziele

Dieses Dokument beschreibt die Technologieauswahl der PG555 zur Umsetzung einer Internetbanking-Lösung für die Magnus Monetarus Bank.

Bei der Auswahl der Technologien wurde im Wesentlichen auf zwei Dinge geachtet. Zum einen sollten die verwendeten Technologien absolut sicher sein, da dies besonders im Rahmen von Bank-Angelegenheiten oberste Priorität hat. Zum anderen sollte mit den ausgewählten Technologien die Entwicklung einer Internetbanking-Lösung möglichst einfach und komfortabel sein. Bei der Priorisierung dieser beiden Aspekte soll darauf hingewiesen werden, dass die Sicherheit grundsätzlich dem Komfort übergeordnet ist.

Das Dokument ist wie folgt unterteilt. Zunächst wird auf die verwendete Cloud-Software eingegangen. Anschließend geht es um die verwendete Programmiersprache für das Server-Backend. Weiter wird auf die verwendete Technologie für das Frontend eingegangen. Abschließend geht es um die verwendeten Technologien zur Kommunikation unter den Servern und um die Kommunikation mit externen Services über diverse Schnittstellen.

2 Cloud-Umgebung

Für die Auswahl der Cloud-Software standen nach ausführlicher Recherche der PG555 drei Lösungen zur Auswahl:

- Eucalyptus (2.0.3) [1] , [2]
- OpenNebula (3.0) [3] , [4]
- OpenStack (Diablo) [5] , [6]

	Eucalyptus	**OpenStack**	**OpenNebula**
Kompatibilität	hervorragende Kompatibilität (Amazon EC, S3 etc.)	befriedigend (einige kleinere Anbieter)	gering
Updates	mehrmals pro Jahr	unregelmäßig	mehrmals pro Jahr
Community	sehr gut	sehr gut	gut
kommerzielle Version	verfügbar	nicht verfügbar	nicht verfügbar

Tabelle 2: Vergleich der Cloud-Software

Es existieren neben den hier genannten Lösungen eine Menge weiterer Lösungen, die weniger bekannt und weniger verbreitet sind. Diese werden hier nicht weiter betrachtet, da ein wesentlicher Aspekt, der der Kompatibilität zu externen Cloud-Anbietern, im Vordergrund steht. Durch diese Kompatibilität wäre es möglich, ohne große Änderungen der Software von einer private-Cloud zu einer public-Cloud zu wechseln. Eine solche zusätzliche Option liegt sicher im Interesse des Kunden. Neben der Kompatibilität spielt auch die Regelmäßigkeit von Updates eine wesentliche Rolle. Auch Cloud-Software kann, wie jede andere Software auch, nicht als vollkommen fehlerfrei angenommen werden. Daher spielen, insbesondere für sicherheitskritische Systeme, regelmäßige Updates eine wesentliche Rolle. Wie in Tabelle 2 zu sehen ist, werden sowohl Eucalytus, als auch OpenNebula häufig mit Updates versorgt. OpenStack scheint nur in unregelmäßigen Abständen mit Updates versorgt zu werden. Da auch in Betracht gezogen wird, dass die Magnus Monetarus Bank die private-Cloud selbst wartet, ist dies sicherlich ein wesentlicher Vorteil gegenüber anderen Cloud-Systemen. In dem Kontext spielt auch die Unterstützung durch eine Community eine Rolle. Insbesondere bei Problemen kann eine starke und aktive Community eine erste solide Hilfestellung bilden. Für Eucalyptus steht darüber hinaus eine kommerzielle Version zur Verfügung. Diese bietet u.a. den Vorteil

[1] http://open.eucalyptus.com/wiki/FAQ
[2] http://open.eucalyptus.com/participate/wiki/community
[3] http://wiki.openstack.org/Releases
[4] http://www.openstack.org/community/
[5] http://opennebula.org/software:release
[6] http://opennebula.org/community:community

des zusätzlichen Supports.
Aufgrund der hervorragenden Community, regelmäßigen Updates und die Kompatibilität zu den großen Public-Cloud-Anbietern ist Eucalyptus die beste, und daher die von der PG555 gewählte, Cloud-Umgebung.

3 Model-View-Control Prinzip

Das Model-View-Control (MVC) Prinzip [7] ist ein bereits bewährtes Architekturmuster in der Softwareentwicklung. Im Wesentlichen wird die Software in 3 Betandteile mit unterschiedlichen Aufgaben zerlegt:

- Model

  Das Model stellt das Datenmodell bereit. In den Klassen der Model-Komponente werden lediglich Daten gehalten und verwaltet. Entsprechend hängt diese Komponente von der Steuerung (Control) und der Ansicht (View) ab.

- View

  Die View-Komponente enthält lediglich das Frontend, also die Benutzeroberfläche, die der Nutzer beim endgültigen Produkt sieht und bedient. In dem Fall der durch die PG555 zu entwickelnden Software stellt die Weboberfläche des Systemes die View-Komponente dar. **Sowohl die Kunden, als auch die Mitarbeiter werden sich über diese Weboberfläche einloggen.**

- Control

  Der Controller ist zuständig für sämtliche Logik und Steuerung der Abläufe und Prozesse innerhalb der Software. Er wird unter anderem durch die GUI (View-Komponente) dazu instruiert diverse Abläufe zu steuern.

Das MVC Prinzip bietet einige Vorteile gegenüber anderen Architekturmustern.[8] Beispielsweise kann Software, die nach diesem Prinzip entworfen wurde, schneller angepasst werden. Zum Beispiel könnte die grafische Oberfläche einfach durch eine andere ersetzt werden, ohne dass die Model- oder Control-Komponente davon betroffen wäre. Ebenso lassen sich Änderungen am Datenmodell leichter integrieren.

[7] nähere Informationen sind nachzulesen unter: Praxisbuch Objektorientierung: Prinzipien, Design, Umsetzung: OOP mit C++, Java, Ruby und C#, inkl. Aspektorientierung, (Barnhard Lahres, Gregor Rayman, Galileo Computing Verlag, 2006)

[8] Die Vor- und Nachteile dieser Architekturmuster hier näher zu beleuchten, würde den Rahmen dieses Dokumentes sprengen. Daher wird an dieser Stelle darauf verzichtet.

4 Server-Software (Backend)

Die Server, die innerhalb der Cloud laufen, werden vor Angriffen sicher geschützt. Die Wahl der Programmiersprache zur Implementierung der Serversoftware spielt hier eine wesentliche Rolle.

Um diese absolute Sicherheit zu gewährleisten, wird für die Entwicklung der Server-Software Java (7.0.1) als Programmiersprache zum Einsatz kommen. Diese bietet gegenüber anderen Programmiersprachen, wie z.B. C++, den wesentlichen Vorteil, dass der kompilierte Code innerhalb einer eigenen Virtual Machine läuft. Außerdem kann es keine sicherheitskritischen Programmierfehler geben, die z.B. durch Unachtsamkeiten mit dem Umgang von Zeigern entstehen können, da es diese erst gar nicht in Java gibt. Ein weiterer Vorteil, der sich durch die Nutzung von Java ergibt, sind die verfügbaren Erweiterungen in puncto Sicherheit:

- Die Java Cryptography Extension (JCE) definiert Programmierschnittstellen für diverse Verschlüsselungsverfahren.
- Der Java Authentication and Authorization Service (JAAS) definiert Programmierschnittstellen für die sichere Feststellung der Identität eines Anwenders.
- Die Java Secure Socket Extension (JSSE) ermöglicht sichere Kommunikation über SSL.

Um die Daten sinnvoll und effizient verwalten zu können, wird auf den Servern eine Datenbank benötigt. Bei der Auswahl wurden lediglich kostenlose Open-Source Lösungen betrachtet, um keine unnötigen Kostenstellen zu eröffnen. Diese stehen kommerziellen Datenbank-Lösungen in nichts nach. Im Wesentlichen gibt es hier die großen und bekannten Datenbanken, die zur Wahl stehen:

- MySQL (5.5.19) [9]
- PostgreSQL (9.1.2) [10]

Hauptsächlich sticht MySQL durch einen erheblichen Geschwindigkeitsvorteil gegenüber PostgreSQL hervor. MySQL weicht zwar stärker vom SQL-Standard ab, ist dafür aber mit einer erheblich stärkeren Community und einer wesentlich besseren Verfügbarkeit bei externen Anbietern vertreten. (siehe 3) Da bei der zu entwickelnden Internetbanking-Software mit einem stetigen Wachstum der Kunden gerechnet werden muss, ist dies ein nicht zu vernachlässigender Vorteil. **Um die Daten des Internetbanking-Servers zu halten, wird eine MySQL-Datenbank verwendet.**

[9] http://www.mysql.de/
[10] http://www.postgresql.de/

	Performance	**Kompabilität**	**Community**	**kommerzielle Version**	**Verfügbarkeit bei externen Anbietern**
MySQL	schnell	teils starke Abweichungen vom ANSI-Standard	sehr groß	verfügbar	nahezu überall verfügbar
PostgreSQL	teilweise schnell, teilweise eher langsam	weitgehend konform mit dem SQL-Standard ANSI-SQL 92	eher klein	nicht verfügbar	seltener

Tabelle 3: Vergleich der Datenbank-Systeme

5 Entwicklung und Darstellung des Frontend

Eine Internetbanking-Anwendung verfügt neben der Server-Software auch über eine Weboberfläche, über die sich sowohl die Kunden einloggen können, als auch die Mitarbeiter ihrer Tätigkeit nachgehen können. Um eine reibungslose Anbindung an die Java-Server-Software zu gewährleisten, wird hier als Technologie das JavaServer Faces (JSF) Framework [11] verwendet. Mit Hilfe dieses Frameworks werden wiederum JavaServer Pages erzeugt, die zur dynamischen Generierung von HTML und XML Daten verwendet werden. Diese erzeugten Daten laufen als Java Bytecode innerhalb eines Servlets. Somit ist auch hier wieder eine hohe Sicherheit gewährleistet.

Um der Weboberfläche ein interessantes Design zu geben, werden übliche Techniken zum Design einer Webseite, wie z.B. CSS [12] verwendet. Dies dient lediglich zur Darstellung und stell somit kein Sicherheitsrisiko dar.

[11] http://www.oracle.com/technetwork/java/javaee/javaserverfaces-139869.html
[12] http://www.w3.org/Style/CSS/

6 Sichere Kommunikation unter Servern und zu externen Services

Ein wesentliches Sicherheitsrisiko stellt die Kommunikation unter den Servern untereinander und mit externen Services, wie z.B. der Schufa dar. Entsprechend werden alle Verbindungen, gegen Angriffe von Dritten, durch den Einsatz von hybrider Verschlüsselung, geschützt.

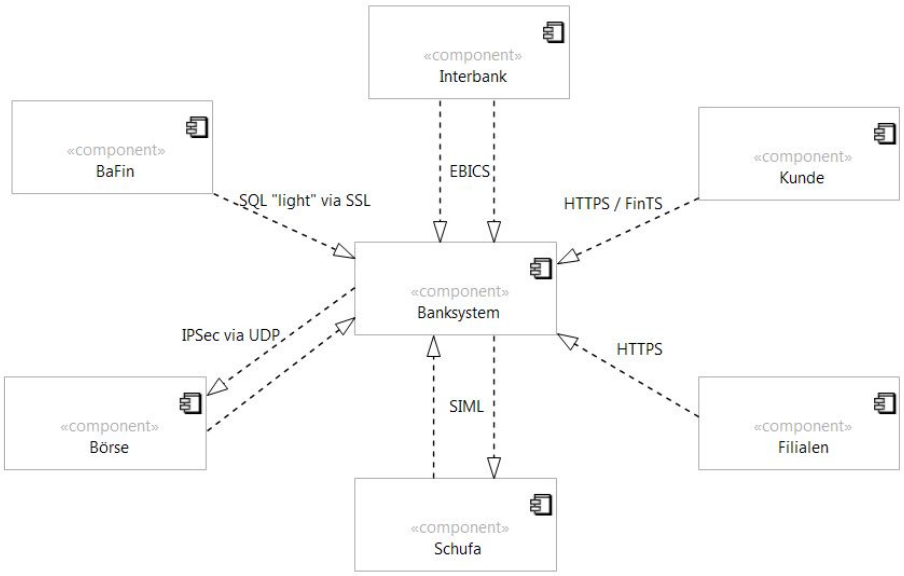


Abbildung 1: Übersicht über die externen Schnittstellen

In der Grafik (Abb. 1) ist eine Übersicht der Schnittstellen zu externen Services zu sehen. An den Kanten stehen jeweils die Verschlüsselungsverfahren, die genutzt werden, um eine sichere Kommunikation zu gewährleisten. Im Folgenden soll auf diese Kommunikationsmethoden näher eingegangen werden:

- **SSL (TLS)**

  SSL ist ein hybrides Verschlüsselungsprotokoll und basiert somit auf einem asymmetrischen und symmetrischen Verschlüsselungsverfahren. Seit der Version 3.0 wird SSL unter dem Namen Transport Layer Security (TLS) weiter entwickelt. Unter dem Aspekt der Sicherheit, stellt TLS das sicherste hybride Verschlüsselungsverfahren dar, das gegenwärtig verfügbar ist.

- **HTTPS**

  HTTPS ist die verschlüsselte Variante des HTTP-Protokolls. Das TLS-Verschlüsselungsprotokoll ist im TCP/IP-Modell oberhalb der Transportschicht und unterhalb der Anwendungsprotokolle, wie z.B. HTTP, angesiedelt (HTTP over SSL/TLS). HTTPS ist somit die zusammengefasste/kombinierte Version von HTTP und SSL/TLS.

Um die Verschlüsselung anzudeuten wird der Name HTTPS verwendet. Durch diese Verschlüsselung wird der Datenverkehr, der z.B. zwischen Kunde und Internetbanking-Server stattfindet, vor Angriffen (z.B. Abhören der Netzwerkdaten) Dritter geschützt.

- **IPSec**

  IPSec dient zur sicheren Kommunikation unter Gewährleistung von Vertraulichkeit, Authentizität und Integrität. Im Gegensatz zu SSL/TLS arbeitet IPSec direkt auf der Vermittlungsschicht der TCP/IP-Modells.

- **FinTS**

  Financial Transaction Services (FinTS) ist der Nachfolger des verbreiteten HBCI-Standards zur Kommunikation mit Banken.
  FinTS ist als Baukasten-System aufgebaut und unterstützt beispielsweise Verfahren, wie das PIN/TAN-Verfahren und die SECCOS-HBCI-Signaturkarte als mögliches einheitliches Sicherheitsmedium.

- **EBICS**

  EBICS steht für Electronic Banking Internet Communication Standard und ist ein Standard zur Übertragung von Zahlungsverkehrsdaten im Internet. Das Verfahren ist nach aktuellem Stand der Technik sicher, da die Daten in einem XML-Container über HTTP mit einer TLS-Verschlüsselung versendet werden.

7 Frameworks und Extensions

Zur Entwicklung des Bankingsystems werden mehrere Frameworks verwendet. Diese bieten den Vorteil, dass auf diverse bereits implementierte Funktionen zurückgegriffen werden kann. Abhängig davon, welches Framework man für einen bestimmten Zweck einsetzt, sind die enthaltenen Funktionen sehr weit entwickelt und mehrfach erprobt. Im Folgenden werden die von der PG555 verwendeten Frameworks vorgestellt:

- Spring (3.0.6) [13]

  Spring ist ein Framework für die Java-Plattform und bietet für diverse Problemstellungen, die während der Entwicklung von Software in Java auftreten können, Lösungen. Im Vordergrund steht hierbei die Vereinfachung der Entwicklung. Durch eine breit gefächerte Auswahl von Erweiterungen für das Spring Framework gibt es auch die Möglichkeit das Framework in Verbindung mit JavaServer Faces zur Entwicklung von Webapplikationen einzusetzen.

- Java Cryptography Extension (JCE)[14]

  Die JCE ist, wie der Name schon sagt, eine Extension für Java, welche diverse kryptographische Implementierungen beinhaltet. Im Wesentlichen bietet die JCE Lösungen für Aufgaben aus den Bereichen Verschlüsselung, Kommunikations-Authentifizierung und Schlüsselverwaltung. Genauer bietet die JCE folgende Dienste, die für die Entwicklung von sicherheitskritischen Systemen relevant sind:

  - **Cypher**

    Cypher bietet sowohl symmetrische, als auch asymmetrische kryptographische Algorithmen zur Verschlüsselung an.

  - **Key Management**

    Key Management bietet Lösungen für die Schlüsselgenerierung zum sicheren Aushandeln von Schlüsseln und zum Zerlegen der Schlüssel in ihre Bestandteile.

  - **Message Authentication Codes**

    Die Message Authentication Codes dienen zur Berechnung von Authentifizierungen für Kommunikationen

[13] http://www.springsource.org/
[14] http://www.oracle.com/technetwork/java/javase/tech/index-jsp-136007.html

8 Business Process Mining

Um den sicheren Ablauf des Systems zu gewährleisten, werden sämtliche Vorgänge und Prozesse protokolliert. Um die Prozesse innerhalb des Onlinebanking-System auf Anomalien zu prüfen, müssen die erhobenen Daten mit einem Referenzmodell verglichen werden.

Leider ist das Angebot an kostenlosen oder zumindest günstigen Lösungen zur Durchführung von Business Process Mining sehr begrenzt. Somit fällt beispielsweise die ARIS-Familie der Software AG aufgrund des hohen Kostenaufwands heraus. Als einzige anwendbare Lösung blieb lediglich ProM (Version 6.1) der TU Eindhoven. Mit Hilfe dieser Software kann aus Event-Log Daten ein Modell erstellt werden. In unserem Fall wird aus den Event-Logs ein Petri-Netz erzeugt. Um dieses erzeugte Petri-Netz auf Anomalien zu prüfen, muss es mit einem Referenz-Petri-Netz verglichen werden.
Dieses Referenz Petri-Netz kann sowohl manuell erstellt werden, als auch automatisch aus einem Event-Log aus korrekten Prozessen erzeugt werden.

Damit ProM die protokollierten Daten verarbeiten kann, müssen diese in dem XML XES Format vorliegen. Dazu wurde die OpenXES-Library verwendet um entsprechende Logging-Methoden zu implementieren.

9 Testumgebung

Das Testen der Software stellt einen wesentlichen Bestandteil des Entwicklungszyklus dar. Somit ist es unverzichtbar und insbesondere im Rahmen einer cloudbasierten Internetbankinglösung ein Aspekt mit sehr hoher Priorität.

Je nach Anwendungsdomäne werden unterschiedliche Testumgebungen und Werkzeuge benötigt:

- **Tests auf Korrektheit**

  Grundlegend eignet sich JUnit (4.10) [15] als Testumgebung für Java Projekte, in denen man Komponenten- und Integrationstests durchführt. Da wir zusätzlich Bestandteile von Java EE durch JSF verwenden, wird zusätzlich das Tool Cactus (1.8.1) eingesetzt. Bei Cactus [16] handelt es sich um eine speziell angepasste Version von JUnit für Java EE. Da auch die Oberfläche auf Korrektheit geprüft werden soll, kommt zusätzlich Selenium (2.11.0) [17] zum Einsatz. Selenium bietet z.B. die Möglichkeit das Durchklicken der Oberfläche (z.B. um die Korrektheit dieser zu prüfen) zu automatisieren.

- **Tests auf Sicherheit**

  Sicherheitstests sind sehr vielseitig. Daher werden auch eine Vielzahl von Werkzeugen eingesetzt, um eine bestehende Sicherheit zu garantieren und nachzuweisen.

  – Application Vulnerabilty Scanner

    Einen häufigen Schwachpunkt bei Webservices stellen Sicherheitslücken in der verwendeten Server-Software dar. Um mögliche Sicherheitslücken im von der PG555 erstellten System, ausfindig zu machen, werden Werkzeuge eingesetzt, mit denen Software auf bereits bekannte Sicherheitslücken geprüft werden kann. Das wohl bekannteste und von uns eingesetzte Werkzeug ist das Metasploit-Framework (4.1). [18]

  – Web Vulnerability Scanner

    Sensible Daten wie z.B. Kundendaten müssen optimal geschützt sein. Damit Dritte nicht unbefugt über Schwachstellen innerhalb der Webseite auf Datenbanken z.B. über SQL-Injection [19] zugreifen können, wird das Webportal der Internetbankinglösung manuell und unterstützt durch Tools, auf Schwachstellen untersucht. [20]

[15] http://www.junit.org/
[16] http://jakarta.apache.org/cactus/
[17] http://seleniumhq.org/
[18] http://metasploit.com/
[19] http://de.wikipedia.org/wiki/SQL-Injection
[20] w3af ist solch eine Tool: http://w3af.sourceforge.net/

10 Entwicklungsumgebung

Zur Entwicklung des Projektes wird die integrierte Entwicklungsumgebung (IDE) Eclipse (3.7) [21] verwendet. Der wesentliche Vorteil ist, dass diese Entwicklungsumgebung hauptsächlich zur Entwicklung von Java-Anwendungen entworfen wurde (inzwischen unterstützt diese diverse andere Programmiersprachen) und sie sich durch eine Vielzahl von Plugins ergänzen lässt. Im Wesentlichen soll die Entwicklungsumgebung folgende Punkte gewährleisten:

- **Maximale Unterstützung des Entwicklers**

  Dies wird bei Eclipse unter anderem durch die automatische Code-Vervollständigung geleistet. Eine weitere Erleichterung ist, dass Projekte direkt aus der IDE kompiliert und getestet werden können.

- **Anbindung an eine Versionsverwaltung**

  Durch diverse Plugins kann eine Anbindung an das SVN [22] der PG555 erfolgen. Somit werden aktuelle Entwicklungsfortschritte zentral gespeichert und dem Rest des Teams zugänglich gemacht. Dies bietet gleich mehrere Vorteile, wie die Funktion eines Backups oder des Wiederherstellens von älteren Versionen einer Datei. Darüber hinaus kann jederzeit nachvollzogen werden, was, wann, von wem geändert wurde.

- **Qualitätskontrolle**

  Um eine für alle Teammitglieder akzeptable Qualität des geschriebenen Quellcodes zu gewährleisten, ist eine Konvention erstellt worden, die beschreibt, wie geschriebener Quellcode auszusehen hat. Dies bringt diverse Vorteile, wie z.B. eine erhöhte Lesbarkeit und Wartbarkeit des Quellcodes, was das Arbeiten für andere Teammitglieder erheblich erleichtert. Auch dies kann durch ein Plugin für Eclipse sichergestellt werden. Sollte die Konvention nicht eingehalten worden sein, kann der Quellcode nicht ins SVN eingecheckt werden.

[21] http://www.eclipse.org/
[22] http://subversion.tigris.org/

Literaturverzeichnis

[1] TMG. http://www.gesetze-im-internet.de/tmg/, 1990. [Online; accessed 20-February-2012].

[2] KonTraG. http://www.wiwi.uni-regensburg.de/scherrer/edu/opi/kontrag.html, 1998. [Online; accessed 18-November-2011].

[3] KWG. www.gesetze-im-internet.de/bundesrecht/kredwg/gesamt.pdf/, 1998. [Online; accessed 18-November-2011].

[4] WpHG. http://www.gesetze-im-internet.de/wphg/, 1998. [Online; accessed 18-November-2011].

[5] LDSG. https://recht.nrw.de/, 2000. [Online; accessed 18-November-2011].

[6] Log4j. http://www.inf.fu-berlin.de/lehre/SS02/swp/swp2/vorbereitung/log4j.shtml, 2002. [Online; accessed 18-November-2011].

[7] Selenium offizielle Seite. http://seleniumhq.org/, 2006. [Online; accessed 25-Februar-2012].

[8] Problemlösung für ssl-empfändliches Selenium. http://kapanka.com/2008/12/selenium-rc-firefox-and-the-self-signed-ssl-cert/, 2008. [Online; accessed 25-Februar-2012].

[9] BDSG. http://www.gesetze-im-internet.de/bdsg_1990/, 2009. [Online; accessed 18-November-2011].

[10] AktG. http://www.gesetze-im-internet.de/aktg/, 2010. [Online; accessed 18-November-2011].

[11] MaRisk. http://www.bafin.de, 2010. [Online; accessed 18-November-2011].

[12] AO. http://www.gesetze-im-internet.de/ao_1977/index.html, 2011. [Online; accessed 18-November-2011].

[13] BSI. https://www.bsi.bund.de/DE/Home/home_node.html/, 2011. [Online; accessed 18-November-2011].

[14] BSI. https://www.bsi.bund.de/DE/Home/home_node.html/, 2011. [Online; accessed 18-November-2011].

[15] HGB. http://www.gesetze-im-internet.de/hgb/, 2011. [Online; accessed 18-November-2011].

[16] Redmine. http://www.redmine.org, 2011. [Online; accessed 18-November-2011].

[17] UML. http://www.uml.org/, 2011. [Online; accessed 18-November-2011].

[18] UML Superstructure specification. http://www.omg.org/spec/UML/2.4.1/, 2011. [Online;accessed 18-November-2011].

[19] Euca2ools. http://open.eucalyptus.com/wiki/Euca2oolsGuide, 2012. [Online; accessed 05-March-2012].

[20] Eucalyptus Architektur. http://open.eucalyptus.com, 2012. [Online; accessed 27-February-2012].

[21] Eucalyptus User's Guide. http://open.eucalyptus.com/wiki/EucalyptusUserGuide_v2.0, 2012. [Online; accessed 05-March-2012].

[22] J2EE Core Patterns. http://java.sun.com/blueprints/corej2eepatterns/Patterns/, 2012. [Online; accessed 25-Februar-2012].

[23] MySQL GUI Tools. http://dev.mysql.com/downloads/gui-tools/5.0.html, 2012. [Online; accessed 25-Februar-2012].

[24] MySQL JDBC-Treiber. http://www.mysql.de/downloads/connector/j/, 2012. [Online; accessed 25-Februar-2012].

[25] MySQL Workbench. http://dev.mysql.com/downloads/workbench/5.2.html, 2012. [Online; accessed 25-Februar-2012].

[26] Netbeans Developer Page. http://netbeans.org/features/index.html, 2012. [Online; accessed 20-March-2012].

[27] Kent Beck. *Extreme programming: Eine Einfï¿½hrung mit Empfehlungen und Erfahrungen aus der Praxis.* Addison-Wesley Verlag, 2003.

[28] K. Beckers, H. Schmidt, J. Kuster, and S. Fassbender. Pattern-based support for context establishment and asset identification of the iso 27000 in the field of cloud computing. In *Availability, Reliability and Security (ARES), 2011 Sixth International Conference on*, pages 329–333. Dept. of Comput. & Appl. Cognitive Sci., Univ. Duisburg-Essen, Duisburg, Germany, IEEE, 2011.

[29] Hans Bergsten. *Java Server Faces - Building Web-based User Interfaces.* O'Reilly, 2004.

[30] Joachim Biskup. *Security in Computing Systems - Challenges, Approaches and Solutions.* Springer, 2009.

[31] Barry W. Boehm. A Spiral Model of Software Development and Enhancement. *IEEE Computer. Vol. 21*, 1998.

[32] Wei Cai, Richard Hellwig, Johann Kexel, Viktor Mucha, Thorben Seeland, Daniel Spasovski, Viktor Stoklossa, Anna Vasileva, Muhammad Waqas, and Dong Yang. Pflichtenheft, 2011.

[33] Wei Cai, Richard Hellwig, Johann Kexel, Viktor Mucha, Thorben Seeland, Daniel Spasovski, Viktor Stoklossa, Anna Vasileva, Muhammad Waqas, and Dong Yang. Projektplan, 2011. PG 555.

[34] Claudia Eckert. *IT-Sicherheit - Konzepte - Verfahren - Protokolle*. Oldenbourg, 2009.

[35] Ramez A. Elmasri and Shamkant B. Navathe. *Grundlagen von Datenbanksystemen*. Pearson Studium, 2009.

[36] Stephan Faßbender and Sebastian Pape. Konzeption und Entwicklung eines sicheren Cloud-basierten Internetbanking-Systems mit anschließender Sicherheitsanalyse auf Basis von Business Process Mining - Ausschreibungsinformationen und Lastenheft, 2011.

[37] David Flanagan. *JavaScript - Das umfassende Referenzwerk*. O'Reilly, 2002.

[38] E. Freeman, E. Freeman, and B. Bates. *Entwurfsmuster von Kopf bis Fuß*. O'Reilly, 2006.

[39] Bundesamt für Sicherheit in der Informationstechnik. IT-Grundschutz-Kataloge, 2009.

[40] Ralph Peters GFT Technologies AG. Ebics detailed specification, 13.05.2011.

[41] Jeremiah Grossmann, Robert Hansen, Petko D. Petkov, Anton Rager, and Seth Fogie. *XSS Attacks - Cross Site Scripting: Exploits and Defense*. Elsevier Inc., 2007.

[42] David R. Heffelfinger. *Java EE 5 development using GlassFish Application Server*. PACKT Publ., 2007.

[43] Heinz Hille. *JDBC - Leitfaden zur Programmierung von Datenbankschnittstellen auf JAVA-Plattformen*. Europäischer Hochschulverlag GmbH & Co. KG, 2009.

[44] Klaus;Müller Markus;Schmied Jürgen Hindel, Bernd;Hörmann. *Basiswissen Software-Projektmanagement*. dpunkt Verlag, 2009.

[45] Andrew Hunt and David Thomas. *Unit Tests mit JUnit - Pragmatisch Programmieren*. Carl Hanser-Verlag München Wien, 2004.

[46] Shareeful Islam, Haralambos Mouratidis, and Jan Jürjens. A framework to support alignment of secure software engineering with legal regulations. *Software and Systems Modeling*, 10:369–394, 2011. 10.1007/s10270-010-0154-z.

[47] ISO - Universal financial industry message scheme. Payments - maintenance 2009 - message definition report, September 2009.

[48] Eric Jendrock, Jennifer Ball, Debbie Bode Carson, Ian Evans, Scott Fordin, and Kim Haase. *The Java EE 5 Tutorial - Third Edition.* Pearson Education, 2006.

[49] Elisabeth Jung. *Servlets und JavaServer Pages - Das Übungsbuch - Band 3.* mitp, 2010.

[50] G. Keller, M. Nüttgens, and Scheer A.-W. Semantische Prozeßmodellierung auf der Grundlage "Ereignisgesteuerter Prozeßketten (EPK)". Technical report, Institut für Wirtschaftsinformatik (IWi), Universität des Saarlandes, 1992.

[51] Tobias Kölligan. PHP/MySQL: Mehr Sicherheit und erhöhte Performance durch MySQLi und Prepared Statementst. *hakin9.org/de*, pages 7 – 10, 2011.

[52] Guido Krueger and Thomas Stark. *Handbuch der Java-Programmierung.* Pearson Deutschland, 2009.

[53] Bernhard Lahres and Gregor Rayman. *Praxisbuch Objektorientierung.* Galileo Computing, 2006.

[54] Nader F. Mir. *Computer and Communication Networks.* Pearson Education, 2007.

[55] Bernd Müller. *JavaServer Faces 2.0.* Hanser München, 2010.

[56] Rolf Oppliger. *SSL and TLS - Theorie and Practise.* Artech House, 2009.

[57] Erik T. Ray. *Learning XML.* O'Reilly, 2003.

[58] Arnold Robbins and Nelson H. F. Beebe. *Classic Shell Scripting.* O'Reilly, 2005.

[59] Matthias Schubert. *Datenbanken: Theorie, Entwurf und Programmierung relationaler Datenbanken.* Vieweg und Teubner, 2007.

[60] Thomas Stark. *Erfolgreich Java EE 6 programmieren.* Addison-Wesley, 2012.

[61] Alexandra Trefz and Marion Büttgen. *Digitalisierung von Dienstleistungen: Umsetzung und Potenziale im Bankensektor.* Logos Verlag Berlin, 2007.

[62] Wil M. P. Van der Aalst. *Process Mining - Discovery, Conformance and Enhancement of Business Processes.* Springer-Verlag Berlin Heidelberg, 2011.

[63] Juri Vasiliev. *Beginning Database-Driven Application Development in Java EE Using Glassfish.* Apress, 2008.

[64] Jason Weiss. *Java Cryptography Extensions - Practical Guide for Programmers.* Elsevier Inc., 2004.