FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO



# A Competence Management System towards Increased Corporate Success

Ana Gabriela Teixeira Soares

FINAL VERSION

Report of Dissertation Master in Informatics and Computing Engineering

Supervisor: Teresa Galvão Dias (PhD)

7<sup>th</sup> April 2010

### A Competence Management System towards Increased Corporate Success

Ana Gabriela Teixeira Soares

Report of Dissertation Master in Informatics and Computing Engineering

Approved in oral examination by the committee:

Chair: Luís Paulo Reis (PhD)

External Examiner: Paulo Novais (PhD) Internal Examiner: Teresa Galvão Dias (PhD)

7<sup>th</sup> April, 2010

## Abstract

In an organization, the careful management of intellectual capital is crucial for the nourishment of a sustained competitive advantage, hardly replicable by its competitors. Particularly, a Competence Management (CM) strategy transversal to the different corporate business units is of pivotal importance, as all knowledge domains are considered.

The main goal of this project – called iKnow - is to identify the requirements for the design of an effective Competence Management System (CMS) and to implement a working prototype. In particular, a study of CM concepts and key issues is intended as well as a state-of-the-art review of CMSs and main features. Additionally, the project contemplates the proposal and prototype deployment of a CMS for Critical Manufacturing (CMF), applying the acknowledged CM concepts to this corporate instance.

Created in 2009, CMF is an organization devoted to the delivery of *avant-garde* solutions for the high-tech manufacturing market, aiming at becoming the prime provider for that niche. In particular, its Services Unit is involved in multiple projects, whose engineers are constantly exposed to new technologies. In this scenario, the extension of skills, the growing number of projects and the associated increase in staff crave for a system capable of effectively managing the unit's competence pool. Although CMF possesses several platforms related to its employees' competences, none of them is specifically oriented towards CM nor deploys a CM strategy. To bridge this flaw, a CMS is imperative.

The project's objectives were successfully accomplished. Aside from the research effort on state-of-the-art CMSs, iKnow materialized in a fully documented software development cycle (requirements analysis, design engineering and prototype deployment and acceptance). Although a CM strategy has been instanced for CMF, iKnow's design is comprehensive and generic enough to be marketed for external corporations.

## Resumo

Numa organização, a gestão cuidada do capital intelectual é crucial para a criação de uma vantagem competitiva sustentada e dificilmente replicável pelos seus concorrentes. Em particular, uma estratégia de Gestão de Competências (CM) transversal às diferentes unidades de negócio empresarial é vital, uma vez que todos os domínios de conhecimento são contemplados.

O objectivo nuclear deste projecto – denominado iKnow - consiste na identificação dos requisitos para o desenho de um Sistema de Gestão de Competências (CMS) e implementação de um protótipo funcional. Concretamente, pretende-se um estudo dos conceitos e problemas chave de CM bem como uma revisão do estado da arte de CMSs e principais funcionalidades. O projecto contempla ainda a proposta e implementação de um protótipo funcional de um CMS para a Critical Manufacturing (CMF), aplicando os conceitos de CM estudados a esta instância organizacional.

Criada em 2009, a CMF é uma organização dedicada ao desenvolvimento de soluções de vanguarda para o mercado de fabrico de tecnologia de ponta, com o objectivo de se assumir como o principal fornecedor para esse nicho. Em particular, a sua Unidade de Serviços está envolvida em vários projectos, cujos engenheiros estão constantemente expostos a novas tecnologias. Neste cenário, a ampliação de competências, o crescente número de projectos bem como o associado aumento de trabalhadores motivam a necessidade de um sistema capaz de gerir eficazmente a *pool* de competências da unidade. Embora a CMF possua diversas plataformas relacionadas com as competências dos seus trabalhadores, nenhuma está especificamente orientada para CM nem implementa qualquer estratégia correspondente. Deste modo, é imperativo um CMS para colmatar esta falha.

Os objectivos do projecto foram cumpridos com sucesso. Para além da pesquisa relativa ao estado da arte de CMSs, iKnow materializou-se num ciclo de desenvolvimento de *software* fortemente documentado (análise de requisitos, engenharia de *design* bem como implementação do protótipo e aceitação). Embora uma estratégia de CM tenha sido instanciada para a CMF, é relevante frisar que o sistema foi desenhado por forma a ser suficientemente abrangente e genérico visando a comercialização para empresas externas.

## Acknowledgements

I can almost see it, that dream I'm dreaming. There's always going to be another mountain, I'm always going to want to make it move. Ain't about how fast I get there, ain't about what's waiting on the other side. It's the climb.

– Miley Cirus, in The Climb

This project would not have been possible without the essential and gracious support of many individuals. The professionalism and dedication of my tutor at FEUP – Professor Teresa Galvão – kept me focused and motivated, providing insightful remarks towards a coherent and consistent thesis. I'm thankful for the opportunity Critical Manufacturing has given me, allowing me to carry out this project and apply my academic training. The guidance, experience and work methodology from my supervisors at this company – Pedro Mêdas and Luis Pessoa – helped me improve my technical and soft skills, contributing to the successful completion of this project. I'd also like to acknowledge Professors Eugénio Oliveira, Nuno Flores, António Castro and João José Pinto Ferreira for sparing time in their agendas to discuss with me implementation approaches for this project.

This work is the culmination of a two year route. I would like to thank everyone who has helped me reach this stage. FEUP's staff, from teachers to colleagues, for making my experience at this institution indeed unforgettable and enriching. In particular, I'd like to thank Professors Rosaldo Rossetti, Ademar Aguiar, Nuno Flores and Hugo Ferreira for the remarkable opportunity to participate in CDVE '09 at Luxembourg. My colleague André Oliveira, for his support and sense of humour. I'd also wish to acknowledge INESC Porto, particularly Eng. Luís Carneiro and Professor José Soeiro Ferreira, for letting me take FEUP's courses in my first year, while working full-time as their researcher.

Last, but not least, I would like to thank my family, for their unconditional support and encouragement to pursue my dreams. My parents, for enriching me with solid educational bases, for showing me the four corners of the world while filling my heart with tender moments. My sister, for listening to my complaints and frustrations, and for believing in me. Luna, for 10 years of pure companionship. The Valadares *clan*, for their endearing support every step of the way. Pedro, for his generosity, patience and affection, pushing me forward and giving me strength to reach my goals. Raquel, for being the world's best godmother and a truly remarkable human being. I am, beyond doubt, blessed to be surrounded by such wonderful people.

I'd like to dedicate this Dissertation to my father Olivério and my uncle Rogério. To my father, for his unshakable faith in who I could be, even when I threw my school notebooks in the trash. To my uncle, for his quietly humble and kind spirit.

To all, thank you.

Ana Gabriela Teixeira Soares

# **Table of Contents**

1	Introduction	1
	1.1 Report Scope	1
	1.2 Motivation	2
	1.3 Goals	2
	1.4 Document Overview	2
2	Competence Management	4
	2.1 Competences in the Organization: Dawn of the Knowledge Era	5
	2.2 Why Manage Competences?	6
	2.2.1 Impact on the Organization Structure	6
	2.2.2 Corporate Benefits	7
	2.2.3 Application Areas	7
	2.3 Competence Management: Strategic Perspective	8
	2.4 Key Challenges	11
	2.4.1 Competence Modelling	11
	2.4.2 Competence Profiling	13
	2.5 Summary	14
3	Competence Management Systems	15
5	3.1 Competence Management Systems: Review	15
	3.2 Key Challenges and Literature Review	21
	3 2 1 Competence Modelling	21
	3.2.2 Competence Profiling	22
	3.3 Summary	
4	iKnow: a General Purpose CMS	
•	4.1 Project Proposal	
	4.2 Client Portraval: Critical Manufacturing	
	4.3 Motivation	
	4.4 Goals	
	4.5 Summary	
5	Requirements Analysis	
-	5.1 Users Characteristics	
	5.2 Restrictions	
	5.3 Assumptions and Dependencies	
	5.4 External Interface Requirements	
	5.5 Functional Requirements	
	5.5.1 User Stories	
	5.5.2 Use Case Models	
	5.6 Supplementary Requirements	64
	5.7 Summary	64

6	Solution Specification	65
	6.1 Architectural Requirements	65
	6.2 Main Design Decisions	66
	6.2.1 Model-View-Controller Pattern	
	6.2.2 Dependency Injection Pattern	67
	6.3 Technology Platform	68
	6.3.1 Preliminary Survey	69
	6.3.2 Final Decisions	75
	6.4 High Level Design	79
	6.4.1 System Components	79
	6.4.2 Physical Architecture	79
	6.4.3 Logical Architecture	81
	6.5 Data Model	84
	6.6 Summary	84
7	Prototyne Denlovment	
-	7.1 Approached Features	
	7.2 Implementation Details	87
	7 3 Testing	
	7 4 Summary	
0		100
8	Conclusions and Future Work	
	8.1 Goal Satisfaction	
	8.2 Future work	
Refe	ferences	
Glo	166911/	110
GIU	,55 <b>41</b> y	
Ann	nex A : Additional Use Case Models	
Ann	nex A : Additional Use Case Models A.1 Core Competence Management	<b>112</b>
Ann	nex A: Additional Use Case Models A.1 Core Competence Management A.1.1 Competence Derivation A.1.2 Competence Derivation Create	<b>112 112 112 114 115</b>
Ann	nex A: Additional Use Case Models A.1 Core Competence Management A.1.1 Competence Derivation A.1.2 Competence Derivation: Derivation Graph	<b>112</b> 112 114 115
Ann	nex A: Additional Use Case Models A.1 Core Competence Management A.1.1 Competence Derivation A.1.2 Competence Derivation: Derivation Graph A.1.3 Wishlist	<b>112</b> 112 114 114 115 117
Ann	nex A: Additional Use Case Models A.1 Core Competence Management A.1.1 Competence Derivation A.1.2 Competence Derivation: Derivation Graph A.1.3 Wishlist A.1.4 Basic Management	<b>112</b> 112 114 115 117 118
Ann	nex A: Additional Use Case Models A.1 Core Competence Management A.1.1 Competence Derivation A.1.2 Competence Derivation: Derivation Graph A.1.3 Wishlist A.1.4 Basic Management A.1.5 Hierarchy A.1.6 Competence Drafile	<b>112</b> 112 114 115 117 118 119
Ann	nex A: Additional Use Case Models A.1 Core Competence Management A.1.1 Competence Derivation A.1.2 Competence Derivation: Derivation Graph A.1.3 Wishlist A.1.4 Basic Management A.1.5 Hierarchy A.1.6 Competence Profile	<b>112</b> 112 114 115 117 118 119 121
Ann	nex A: Additional Use Case Models A.1 Core Competence Management A.1.1 Competence Derivation A.1.2 Competence Derivation: Derivation Graph A.1.3 Wishlist A.1.4 Basic Management A.1.5 Hierarchy A.1.6 Competence Profile A.1.7 Competence Grouping	112           112           114           115           117           118           119           121           123           126
Ann	nex A: Additional Use Case Models A.1 Core Competence Management A.1.1 Competence Derivation A.1.2 Competence Derivation: Derivation Graph A.1.3 Wishlist A.1.4 Basic Management A.1.5 Hierarchy A.1.6 Competence Profile A.1.7 Competence Grouping A.1.8 Import/Export	<b>112</b> 112 114 115 117 118 119 121 123 126
Ann	nex A: Additional Use Case Models A.1 Core Competence Management A.1.1 Competence Derivation A.1.2 Competence Derivation: Derivation Graph A.1.3 Wishlist A.1.4 Basic Management A.1.5 Hierarchy A.1.6 Competence Profile A.1.7 Competence Grouping A.1.8 Import/Export. A.2 Training	112           112           114           115           117           118           119           121           123           126           127
Ann	nex A: Additional Use Case Models A.1 Core Competence Management A.1.1 Competence Derivation A.1.2 Competence Derivation: Derivation Graph A.1.3 Wishlist A.1.4 Basic Management A.1.5 Hierarchy A.1.6 Competence Profile A.1.7 Competence Grouping A.1.8 Import/Export A.2 Training A.2.1 Gap Analysis	112           112           114           115           117           118           119           121           123           126           127           127           120
Ann	nex A: Additional Use Case Models A.1 Core Competence Management A.1.1 Competence Derivation A.1.2 Competence Derivation: Derivation Graph A.1.3 Wishlist A.1.4 Basic Management A.1.5 Hierarchy A.1.6 Competence Profile A.1.7 Competence Grouping A.1.8 Import/Export A.2 Training A.2.1 Gap Analysis A.2.2 Evaluation	112           112           114           115           117           118           119           121           123           126           127           130           122
Ann	nex A: Additional Use Case Models A.1 Core Competence Management A.1.1 Competence Derivation A.1.2 Competence Derivation: Derivation Graph A.1.3 Wishlist. A.1.4 Basic Management A.1.5 Hierarchy A.1.6 Competence Profile A.1.7 Competence Grouping A.1.8 Import/Export. A.2 Training. A.2.1 Gap Analysis A.2.2 Evaluation A.2 Concluster	112           112           114           115           117           118           119           121           123           126           127           130           132
Ann	nex A: Additional Use Case Models         A.1 Core Competence Management         A.1.1 Competence Derivation         A.1.2 Competence Derivation: Derivation Graph         A.1.3 Wishlist         A.1.4 Basic Management         A.1.5 Hierarchy         A.1.6 Competence Profile         A.1.7 Competence Grouping         A.1.8 Import/Export         A.2 Training         A.2.1 Gap Analysis         A.2.2 Evaluation         A.2.3 Lookup         A.3 Coaching	112           112           114           115           117           118           119           121           123           126           127           127           130           132           134
Ann	nex A: Additional Use Case Models A.1 Core Competence Management A.1.1 Competence Derivation A.1.2 Competence Derivation: Derivation Graph A.1.3 Wishlist A.1.4 Basic Management A.1.5 Hierarchy A.1.6 Competence Profile A.1.6 Competence Grouping A.1.7 Competence Grouping A.1.8 Import/Export A.2 Training A.2.1 Gap Analysis A.2.2 Evaluation A.2.3 Lookup A.4 Talent	112           112           114           115           117           118           119           121           123           126           127           130           132           134           138           128
Ann	nex A: Additional Use Case Models A.1 Core Competence Management	112           112           114           115           117           118           119           121           123           126           127           130           132           134           138           138
Ann	nex A: Additional Use Case Models         A.1 Core Competence Management         A.1.1 Competence Derivation         A.1.2 Competence Derivation: Derivation Graph         A.1.3 Wishlist         A.1.4 Basic Management         A.1.5 Hierarchy         A.1.6 Competence Profile         A.1.7 Competence Grouping         A.1.8 Import/Export         A.2 Training         A.2.1 Gap Analysis         A.2.2 Evaluation         A.2.3 Lookup         A.3 Coaching         A.4.1 Sniffing         A.4.2 Scouting	112         112         114         115         117         118         119         121         123         126         127         126         127         130         132         134         138         144
Ann	nex A: Additional Use Case Models         A.1 Core Competence Management         A.1.1 Competence Derivation         A.1.2 Competence Derivation: Derivation Graph         A.1.3 Wishlist         A.1.4 Basic Management         A.1.5 Hierarchy         A.1.6 Competence Profile         A.1.7 Competence Grouping         A.1.8 Import/Export.         A.2 Training         A.2.1 Gap Analysis         A.2.2 Evaluation         A.2.3 Lookup         A.3 Coaching         A.4 Talent         A.4.2 Scouting         A.5 Dashboard	112         112         114         115         117         118         119         121         123         126         127         126         127         130         132         134         138         144
Ann	nex A: Additional Use Case Models         A.1 Core Competence Management         A.1.1 Competence Derivation         A.1.2 Competence Derivation: Derivation Graph         A.1.3 Wishlist         A.1.4 Basic Management         A.1.5 Hierarchy         A.1.6 Competence Profile         A.1.7 Competence Grouping         A.1.8 Import/Export         A.2 Training         A.2.1 Gap Analysis         A.2.2 Evaluation         A.3 Coaching         A.4 Talent         A.4.2 Scouting         A.5 Dashboard         A.6 Gadgets	112         112         114         115         117         118         119         121         123         126         127         126         127         130         132         134         138         138         140         144         144
Ann	nex A: Additional Use Case Models         A.1 Core Competence Management         A.1.1 Competence Derivation         A.1.2 Competence Derivation: Derivation Graph         A.1.3 Wishlist         A.1.4 Basic Management         A.1.5 Hierarchy         A.1.6 Competence Profile         A.1.7 Competence Grouping         A.1.8 Import/Export         A.2 Training         A.2.1 Gap Analysis         A.2.2 Evaluation         A.2.3 Lookup         A.4 Talent         A.4.1 Sniffing         A.4.2 Scouting         A.5 Dashboard         A.6 Gadgets         A.7 Social Network	112         112         114         115         117         118         119         121         123         126         127         126         127         130         132         134         138         140         144         145         147
Ann	nex A: Additional Use Case ModelsA.1 Core Competence ManagementA.1.1 Competence DerivationA.1.2 Competence Derivation: Derivation GraphA.1.3 WishlistA.1.4 Basic ManagementA.1.5 HierarchyA.1.6 Competence ProfileA.1.7 Competence GroupingA.1.8 Import/ExportA.2 TrainingA.2.1 Gap AnalysisA.2.2 EvaluationA.3 CoachingA.4 TalentA.4.1 SniffingA.4.2 ScoutingA.5 DashboardA.6 GadgetsA.7 Social NetworkA.8 Security/Privacy	$\begin{array}{c} 112 \\ 112 \\ 114 \\ 115 \\ 117 \\ 118 \\ 119 \\ 121 \\ 123 \\ 126 \\ 127 \\ 127 \\ 127 \\ 127 \\ 130 \\ 132 \\ 134 \\ 138 \\ 138 \\ 138 \\ 140 \\ 144 \\ 144 \\ 144 \\ 145 \\ 147 \end{array}$
Ann	nex A: Additional Use Case Models         A.1 Core Competence Management         A.1.1 Competence Derivation         A.1.2 Competence Derivation: Derivation Graph         A.1.3 Wishlist         A.1.4 Basic Management         A.1.5 Hierarchy         A.1.6 Competence Profile         A.1.7 Competence Grouping         A.1.8 Import/Export         A.2 Training         A.2.1 Gap Analysis         A.2.2 Evaluation         A.3 Coaching         A.4 Talent         A.4.1 Sniffing         A.4.2 Scouting         A.5 Dashboard         A.6 Gadgets         A.7 Social Network         A.8 Security/Privacy	112         112         114         115         117         118         119         121         123         126         127         126         127         130         132         134         138         138         140         144         145         147         150

# **List of Figures**

Figure 1: Process diagram for strategic working with competences [1]8
Figure 2: Key issues in the strategic perspective of competence management (adapted from [1])
Figure 3: Macro-phases in Competence Management11
Figure 4: Example of a competence gap analysis (adapted from [115])
Figure 5: Example of an employee performance review on Taleo [116]20
Figure 6: Critical Manufacturing logo [74]24
Figure 7: Location of Critical Manufacturing within the Critical Group [74]24
Figure 8: Example of an Excel sheet used for internal CM at CMF26
Figure 9: Detail of Figure 8, where a sample set of competences can be identified26
Figure 10: WISE in action, showing a list of CMF's workers27
Figure 11: CoreCompetence in action, showing a list of a worker's self-evaluation parameters 28
Figure 12: Detail of Figure 11, where there is a list of competences to be evaluated, which the user can grade and add personal comments to
Figure 13: <i>Learning Center</i> , an internal CMF application focused primarily on competence development through e-learning
Figure 14: Outline of iKnow's use cases
Figure 15: System actors' overview42
Figure 16: Core Competence Management use cases
Figure 17: Example of a competence derivation
Figure 18: A competence hierarchy example
Figure 19: <i>Training</i> use cases
Figure 20: Coaching use cases
Figure 21: <i>Talent</i> use cases
Figure 22: Dashboard use cases
Figure 23: Gadgets use cases
Figure 24: Alerts use cases
Figure 25: Social network use cases
Figure 26: Security/Privacy use cases

Figure 27: MVC concept, where direct and indirect associations are represented by a solid a dashed lines, respectively [80]	nd a 67
Figure 28: Explanation of DI's main principles [85]	68
Figure 29: Spring .NET framework logo [92]	75
Figure 30: jQuery logo [94]	76
Figure 31: Telerik Extensions for ASP .NET MVC logo [95]	76
Figure 32 – PostgreSQL logo [97]	77
Figure 33: Overview of the WCF framework [101]	78
Figure 34: iKnow's physical architecture	80
Figure 35: iKnow's vertical logical architecture	83
Figure 36: Vernhout's strategic perspective on Competence Management vs. the deployed prototype features	86
Figure 37: The CM process described by Draganidis et al. vs. the deployed prototype feature	es 87
Figure 38: Organization of the <i>iKnow</i> project, namely the <i>iKnow.Database</i> class library	88
Figure 39: Classes corresponding to the system's data model in <i>iKnow.Database</i>	88
Figure 40: <i>iKnow.Database</i> resources file	89
Figure 41: Interface classes deployed in <i>iKnow.Contract</i>	90
Figure 42: <i>iKnow.GUI</i> overview	90
Figure 43: User interface for competence management, where a list of existing competence presented	s is 91
Figure 44: User interface for system users' account management	92
Figure 45: List of competences within a given group	93
Figure 46: Competence group list	93
Figure 47: Visualization of competences as hierarchies	94
Figure 48: Overview of developed scripts using the jQuery library	95
Figure 49: Intuitive description of the rendering of a table in a view using jQuery	95
Figure 50: Dialog for competence evaluation	96

# **List of Tables**

Table 1: Competence definitions by author [24]	12
Table 2: List of CMSs reviewed [24]	16
Table 3: CM features included in the analysed LMSs [24]	17
Table 4: Common features of the analysed CMSs [24]	18
Table 5: iKnow's user stories for core competence management	34
Table 6: iKnow's user stories regarding training lookup and evaluation	35
Table 7: iKnow's user stories related to coaching	37
Table 8: iKnow's user stories for dashboard features	37
Table 9: iKnow's user stories regarding alerts	37
Table 10: iKnow's user stories regarding talent sniffing and scouting	38
Table 11: iKnow's user stories concerning social network characteristics	
Table 12: iKnow's user stories regarding security/privacy	
Table 13: List of system modules	40
Table 14: List of system actors	42
Table 15: Auto-evaluate a competence use case	44
Table 16: Add/remove competence use case	47
Table 17: Add/remove preferable competence use case	49
Table 18: Perform gap analysis on myself use case	52
Table 19: Search for workers for a project use case	56
Table 20: Generate job interviews from project competences use case	57
Table 21: Add gadget to dashboard use case	59
Table 22: Configure alerts use case	60
Table 23: Accept/decline/ignore friend request use case	62
Table 24: Delegate own role on another user use case	63
Table 25: iKnow's supplementary requirements list	64
Table 26: Clarification of the Maturity parameter	70
Table 27: Details of the Support Community parameter	70
Table 28: Explanation of the (Expected) Learning Curve parameter	70

Table 29: Final comparison matrix of web-based MVC frameworks	72
Table 30: Final DBMS comparison matrix	74
Table 31: Implemented use cases	85
Table 32: Unit tests performed on the Core Competence Management deployed use cases	97
Table 33: Unit tests performed on the Security/Privacy deployed use cases	99

# Abbreviations

ADO	ActiveX Data Objects
AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
API	Application Programming Interface
C#	C-sharp
СМ	Competence Management
CMF	Critical Manufacturing
COM	Component Object Model
CRUD	Create, Read, Update and Delete
DBMS	Database Management System
DI	Dependency Injection
DOM	Document Object Model
DOM	Document Object Model
ERP	Enterprise Resource Planning
GUI	Graphical User Interface
HTML	HyperText Markup Language
IP	Internet Protocol
IT	Information Technology
JSON	JavaScript Object Notation
KM	Knowledge Management
LDAP	Lightweight Directory Access Protocol
LMS	Learning Management System
MVC	Model-View-Controller
REST	Representational State Transfer
SGPS	Sociedade Gestora de Participações Sociais
SOAP	Simple Object Access Protocol
ТСР	Transmission Control Protocol
URI	Uniform Resource Locator
URL	Uniform Resource Identifier
WCF	Windows Communication Foundation
WPF	Windows Presentation Foundation
WSDL	Web Service Definition Language
WSE	Web Services Enhancements
XAML	eXtensible Application Markup Language
XHTML	eXtensible HyperText Markup Language
XML	eXtensible Markup Language
XPath	XML Path Language
XSD	XML Schema

### **Chapter 1**

## Introduction

A competent organization has the ability to structurally and systematically coordinate and commit resources for creating and distributing value to customers and stakeholders.

Arjan Vemhout [1]

Competence Management (CM) is used today for a broad set of purposes: company and project staffing, competence detection and development for single persons, whole companies or virtual organizations [2]. For a corporation, the alignment of human resources to business goals dictates its strategic success [3]. Considering that each individual has a set of competences from various domains, an effective matching of competences with project needs and, subsequently, corporate goals, becomes crucial.

#### 1.1 Report Scope

This document supports the author's completion of the Master degree in Informatics and Computer Engineering, held by the Faculty of Engineering at the University of Porto, Portugal. The research was carried out at Critical Manufacturing S.A. headquarters at Tecmaia (Maia, Portugal), from October 26<sup>th</sup>, 2009 to March 1<sup>st</sup>, 2010.

The writer was enrolled in a project called iKnow – Improved Knowledge Management, oriented to the development and prototype deployment of a Competence Management System for the holding company. This report branches on the work developed throughout that assignment, answering the following key questions:

- What is the project about?
- What is its innovative stamp?
- Which were the project results?
- To what extent did it contribute to the scientific community?

#### Introduction

#### **1.2** Motivation

Undoubtly, part of an organization's knowledge resides in the people who form it. The employees' knowledge value depends on their potential to contribute to the achievement of an organizational competitive advantage [4]. Recent research suggests that human capital attributes (including training, experience and skills) – and in particular the executives' human capital – have a clear impact on organizational results [5], [6], [7], [8], [9], [10]. Indeed, it is the management of these assets that sustains the creation of a coherent and consistent competitive advantage.

Competence Management (CM) has become a leading practice in human resources management [11] and is often applied in organizations to guide selection, assessment, development and performance appraisal [12]. CM can be described as an integrated set of human resources activities aimed at optimizing the development and the use of employee competences. Hence, individual and, subsequently, organizational effectiveness are expected to increase [13].

Critical Manufacturing presents difficulties at the CM level, such as data dispersion and a lack of modern competence appraisal and reporting tools. Indeed, it craves for a flexible and coherent tool for competence mapping, evaluation, development and even monitoring transversal to all business areas – a Competence Management System.

#### 1.3 Goals

The main objective of this project is to identify the requirements for the design of an effective Competence Management System (CMS) and to implement a working prototype. In particular, the goals are as follows:

- Study of Competence Management (CM) concepts and key issues;
- State-of-the-art review of CMSs and main features;
- Proposal of a CMS suitable for a given organization (Critical Manufacturing), applying the acknowledged CM concepts to this corporate instance. However, this system ought to be generic enough as to successfully adapt other organization.;
- *Implementation of a working CMS prototype.*

#### **1.4 Document Overview**

This report approaches Competence Management (CM) in a top-down fashion: it begins with a broad perspective of this area and its key issues, then restricting this research field to a given corporation. The document is divided into eight chapters as follows.

The initial section introduces the project, detailing its motivation and goals. Additionally, the document's scope and structure overview are stated. Apart from the conclusion, each subsequent chapter is closed by a summary subdivision, which gathers the main ideas of that section.

Chapter 2 analyses CM, focusing on its impact on the organization, its history, strategic perspective as well as key issues such as competence modelling and profiling.

The third part delivers a state-of-the-art review on Competence Management Systems and key challenges mentioned in the previous chapter.

#### Introduction

Section 4 details the project, mentioning its motivation and goals from a perspective that's closely linked to the target company: Critical Manufacturing.

Chapter 5 performs a requirements analysis, presenting the most relevant use cases.

The sixth part depicts the solution specification, namely its architecture, main design decisions and technology platform, justifying the choices for the latter.

Part 7 details the implemented prototype and acceptance tests.

Chapter 8 is dedicated to concluding remarks and future work considerations.

A glossary is also contemplated, where all relevant vocabulary used in this manuscript is clarified. Annex A provides additional use cases for the reader, while Annex B illustrates the performed technology survey. Finally, Annex C supplies code samples from the deployed prototype.

### Chapter 2

## **Competence Management**

Sustained competitive advantage is no longer rooted in physical assets and financial capital, but in effective channelling of intellectual capital. [14]

We are not only in a new millennium, but also in a new era: the knowledge era [15]. As the contemporary business scenario becomes increasingly mutable and heterogeneous, the corporation success is ever more rooted on its capacity to anticipate the changes and to respond to new business challenges. In particular, organizations strive to obtain competitive advantage through knowledge [16].

Businesses have long recognized the importance of managing their resources, tangible or intangible. Indeed, as stated by Prahalad and Hamel [3], the ability to develop and leverage the value of intangible assets comprises a core competence for organizations, particularly those providing financial and professional services. In such knowledge-intensive corporations, processing knowledge is central to business success. Particularly, focusing on competences allows the organization to identify the key knowledge that an employee or the organization itself should possess in order to achieve the desired targets.

Competence Management - a Knowledge Management strategy - is key to aligning human capital assets with business strategy in order to create value for organizations and drive their performance forward in a competitive talent market. Inevitably, corporate success is driven by people – human capital – through their creativity, knowledge and their relationships with customers, co-workers and professional networks. Hence, the challenge for organizations is to align their human capital assets with the strategic priorities identified in their business strategy to create value for their organization [17].

The present chapter is dedicated to the study of Competence Management and its impact on the organization<sup>1</sup>.

<sup>&</sup>lt;sup>1</sup> The terms competency and competence are often used interchangeably in the literature. For the purpose of this document, please consider them as synonyms from this point on.

# 2.1 Competences in the Organization: Dawn of the Knowledge Era

A critical aspect of business management is the successful creation of processes which drive the development of a continuous flow of innovation, to give a basis for competitive advantage. To reach this goal, the establishment of a Knowledge Management (KM) strategy may be considered the best way to channel the organization's efforts to this end [18].

In 1984, Wernerfelt proposed a resource-based view of the corporation, rooted in the heterogeneity of organizations. In particular, resources could be interpreted as strengths or weaknesses of the corporation, including tangible and intangible assets such as in-house technology knowledge, personnel skills or trade contacts, to mention a few [19]. Most importantly, those assets ought to be rare, valuable, difficult to imitate and non-substitutable. In this sense, knowledge was seen as a strategic asset with the potential to be a source of competitive advantage for an organization [15].

During the 1990s, the importance of unique and inimitable assets such as tacit knowledge and competencies [20] came to be seen as the core competencies for firms that wished to compete in the new 'knowledge based economy' [21]. As the key strategic resource for organizations, knowledge needed to be efficiently managed to gain competitive advantage among a company's peers.

KM focuses on techniques to manage a common base of organizational knowledge that allows heterogeneous organizational groups, functions and communities to coordinate their efforts and share knowledge across time, function, discipline and task specific boundaries [22]. With the crescent interest in KM and the recognition of the value of tacit knowledge that took place at the same time, there was a renewed interest in trying to develop Knowledge Management Systems in general and Competence Management Systems (CMSs) in particular [23]. Hence, Competence Management (CM) is an important research object of the more general area of KM [24].

Developing versatile employees and leveraging their competencies in order to cope with different corporate needs is a matter of pivotal importance [25]: employees' competences in the form of technical and cognitive capabilities directly affect the company's knowledge creation abilities. Competence is a way to put into practice some knowledge, know-how, and also attitudes within a specific context. Therefore, CM has the prime objective to well define and continuously maintain a set of competences according to the goals of the corporation [26]. In particular, it provides a roadmap that helps organizations develop their talent pool [27].

In the 21<sup>st</sup> century workplace, the most important corporate resource over the next 20 years will be talent: smart, sophisticated business people who are technologically literate, globally astute and operationally agile. Yet as the demand for high performing talent grows, the supply of qualified talent doesn't build up as quickly, creating an intensely competitive environment for the remaining available talent. How do organizations shift from traditional approaches to talent shortages – such as hiring their way out – to developing the talent they need today [27]?

#### 2.2 Why Manage Competences?

Lately, competences have proven to be useful as an abstraction of work-relevant human behaviour in a variety of contexts and across different actor groups (individual, organizational and market). Within an organization, competences allow a more effective resource allocation (e.g. team staffing), Knowledge Management and informal learning support, as well as human resource development in general [28]. Indeed, competences make the individual skills and their relationship to corporate goals transparent. If, as stated earlier in this chapter, competences are crucial to organizational competitive advantage among peers, then they should be carefully managed towards (near) optimal corporate performance.

#### 2.2.1 Impact on the Organization Structure

Broadly speaking, Competence Management (CM) concerns the way in which competences in a corporation, of a group or individuals of the corporation are organized and controlled [26]. Considering competence as a way to put in practice some knowledge, know-how and also attitudes in a particular context [26], its management has the prime goal to well define and continuously maintain the set of competences according to the overall organization's goals.

CM has long been associated to the management of people's qualifications and their annual assessment. Initially, these qualifications were closely linked to predefined jobs, tasks and roles within (relatively) stable job-based organizations. In such an environment, the standardization of organization practices (and processes) was the main approach to maintain or improve corporate performances [26]. However, this strong linkage between competences and standard organizational jobs gradually decoupled over time as inter-organizational competitiveness grew and businesses became swiftly mutable. Thus, corporations were forced to deal with less structured, less stable and less standardized processes [26].

Such adaptation of the organization's behaviour and structure originated CM approaches based on two major (complementary) concepts/axis: *individual competence* and *core competence*. The first one focuses on the increased importance of the individual employees' competences for companies in the achievement of corporate goals. On the other hand, the latter has its foundations on the idea that companies rely on very specific and (ideally) unique competences – core competences – which directly sustain the company's competitiveness.

Given this gradual organizational structure shift, CM caused significant impact on the overall corporation. In particular, as pointed out by Harzallah et al. [26], nowadays CM is part of all three company control levels, as follows:

- *Strategic Level*, because it ensures that the competences required for the fulfilment of corporate goals are correctly identified. For instance, it may provide recruiting and training staff policies or even support (re)organization decisions by checking required and available competences;

- *Tactical Level*, as managers must guarantee the achievement of the tasks for which they are responsible by reorganizing the work, reallocating the personnel, recruiting new members, or deciding on the persons to be trained;

- *Operational Level*, since it can be used for day-by-day personnel reallocation - for instance, to face some unexpected situations (e.g. missing staff).

#### 2.2.2 Corporate Benefits

As pointed out by Draganidis and Mentzas [24], organizations can highly benefit from Competence Management approaches, namely because:

- Competence Management Systems allow the identification of the skills, knowledge, behaviours and capabilities needed to meet current and future personnel selection needs in alignment with the differentiations in strategies and organizational priorities;
- Corporations can focus the individual and group development plans to eliminate the gap between the competences requested and those available according to a project, job role or enterprise strategy.

#### 2.2.3 Application Areas

Concerning employee management, Competence Management (CM) is crucial in the following activities [24]:

- *Workforce planning*: competences are used to evaluate the current and future competence needs at macro (organization wide) and micro (individual employees) levels. A competence gap analysis can reveal the chasm between the competences of a that a target object (employee, groups or the organization) has and those it should have, contributing to the workforce development plans;
- *Recruitment management*: competences are used to compare the capabilities of the candidate against the requirements of the offered position; once the best candidate is identified, competence gaps form the basis for a new-hire learning plan;
- *Learning management*: competence gap analysis can identify the needed competences, which can then be linked to the equivalent learning objects;
- *Performance management*: worker performance is evaluated against job competence requirements as well as objectives;
- *Career development*: competences are used to create the personal development plans for the employee. The latter can review the needed competences of all the positions and, through comparison with the possessed competences identify potential positions and develop career plans;
- *Succession planning*: organizations assess potential replacements for key positions based on competence requirements.

In sum, by setting ground for a better management of knowledge capital, CM allows companies to evolve quickly in a dynamic and highly competitive environment [29].

#### 2.3 Competence Management: Strategic Perspective

According to Vernhout [1], working with competences at a strategic level is a cyclic process, in which eight significant phases are distinguished, as pictured in Figure 1 and explained as follows.



Figure 1: Process diagram for strategic working with competences [1]

Firstly, it is imperative that management determines the position of the organization on the social-economic scenario, by clearly defining its mission, goals and contribution. This is the preliminary phase called *Define the role of the organization within the social-economic environment*.

Once the mission has been determined, both qualitative and quantitative organization goals must be formulated. Realizing them is of pivotal importance in order to survive in the constant changing business environment. While performing an environment and internal analyses, the role of the organization (strategic position) is refined and the most relevant opportunities and threats are identified. Additionally, the internal analysis allows the identification of the available assets and the organizational capacity to exploit them effectively and efficiently (that is, its strategic capabilities). This strategic development phase is named *Determine the need for competences of the organization*.

When the goals of the organization are clear, the needed competences to realize those goals should be determined. Such action takes place at the level of organizational units (logistics, purchasing, sales, finance and communication, to mention a few). Thus, the focus is on the competences linked to the organizational units of the corporation. This phase is called *Determine the need for competences of the organization*. Accordingly, *competence needs* must

be determined, that is, it should be assessed if internal and/or external recruitment of competences is necessary for achieving the corporate goals.

The third phase – *Inventory the present competences of the organization* – answers the previous question, by determining existing and required competences by organizational unit. In particular, such inventory is performed at two levels: organizational unit and employee-wise. With this approach, the corporation is able to identify competence needs with different granularity, thus allowing a more effective approach when targeting lacked competences. Additionally, employee competences are graded in order to determine the degree in which employees apply a certain competence.

When employee (and organizational unit specific) competences have been identified, the following step is to *Determine job competency profiles and individual competency profile*. A job competency profile describes the most relevant competences for a job as well as the required level of competences. On the other hand, an individual competence profile refers the competences an employee currently possesses. When both profiles are clear, the next step is to determine to what extent competence discrepancy exists, that is, what is the difference between the present and the needed competences at the employee level.

Dependent of the first three phases, the fourth (*Analyse the 'fit' between the present and needed competences*) takes place both at organizational unit and employee level. Such 'fit' is obtained through the comparison of job competency profiles to employee's individual competency profiles, by organizational unit. Such analysis (also called competence gap analysis) allows the identification of competence needs as well as of new business opportunities not previously discovered. In short, phase 4 is about the management of the *competence stock*.

Given the competence 'fit' determined in the previous phase, *Competence-based training and development* focuses on the development of employee's competences as to increase the 'fit' between present and needed competences in the various organizational units. Hence, competence needs are identified and employees are directed to learning processes for improvement.

Competence-based training and development of an employee must explicitly lead to effective work behaviour. Prior to a performance appraisal (phase 7), it should be determined to what extent an employee has shown effective work behaviour. Hence, the phase *Determine effective work behaviour based on the formulated objectives* concerns the assessment how employees performed their tasks in order to realise corporate goals (i.e., to what extent).

Based on the results of the previous phase, the stage *Performance appraisal and competence-based rewarding* concerns the assessment of employee behaviour and results relative to global corporate goals, as opposed to the organization unit analysis previously performed. Hence, the management of future performance, promotions and employee career development can be improved. Based on this performance appraisal results, organizations may reward employees for the applied competences towards corporate goals fulfilment.

The determination of to what extent the organization goals are realised (phase 8) sets ground for the formulation of new organization goals and the process described above is repeated. Hence, this *strategic working with competences* process has a cyclic nature, with the following key issues, as pictured in Figure 2:

- Organization goals
- Competence management
- Competence development
- Performance appraisal



Figure 2: Key issues in the strategic perspective of competence management (adapted from [1])

Similarly, Draganidis et al. [24] present CM as an aggregation of four macro-phases directed to the continuous enhancement and development of individual and organizational competences. As shown in Figure 3, they are the following:

- *Competence mapping*: aims to provide the organization with an overview of all the necessary competences in order to fulfil its targets, which are defined by the organizational business plan, the projects requirements, the group needs and the job role requirements. The required proficiency level for each job profile is defined in this phase as well;
- *Competence diagnosis*: meaning an instance of the current situation of the competences and equivalent proficiency level that each individual employee possesses. A *competence gap analysis* is also essential in this phase in order to define the breach between the number and level of competences that the employees possess compared to the number and level of the competences required by the organization, according to their job role;
- *Competence development*: deals with the scheduling of activities to increase the number and proficiency level of competences that the employees should have, according to the previous two phases and the skill gap analysis;
- *Competence monitoring*: the last phase, which is a continuous examination of the results achieved by the competence development stage.



Figure 3: Macro-phases in Competence Management

#### 2.4 Key Challenges

As previously discussed, Competence Management (CM) is a process of multiple stages, each with its inherent complexity. This section narrows this problem's domain to two key issues, closely linked to competence mapping: *competence modelling* and *competence profiling*. Pictured hereafter, these issues are not isolated; they extend to other CM phases. For example, competence profiling may be used for 'fit' analysis (6th stage in Figure 2)/competence diagnosis (first macro-phase in Figure 3).

#### 2.4.1 Competence Modelling

According to Draganidis and Mentzas [24], over the past few years many companies and governmental organizations have addressed issues such as organizational transformation, performance improvement, employee development, succession planning, etc by the implementation of competence modelling programs, i.e., through strategic initiatives aimed at aligning employee performance with job requirements. A competence model is a list of competences which are derived from observing satisfactory or exceptional employee performance for a specific occupation [24].

Indeed, a competence model can provide identification of the competences employees need to develop in order to improve performance in their current job or to prepare for other jobs via promotions or transfer. It can also be useful in skill gap analysis where the comparison between available and needed competences of individuals or organizations can motivate the construction of individual development plans as to eliminate the gap.

As mentioned before, a competence is an organization's core asset, thus viable as a way to gain competitive advantage. To be managed, a competence must be standardized as a model.

CM is concerned with the improved usage of human skills and knowledge [30]. However, the term competence is used in different ways in the management, organizational and information research literature [30]. In this work, there is the challenge of modelling individual, group and organizational competencies in such a way that they could form part of a corporate knowledge base.

Competences are the building blocks of competence models. For example, a model for an area sales manager might include competences such as sales planning, team working, market competition, industry trends and strategic thinking. However, there is not a consensual definition of competence. In this sense, the difficulty is resident on the definition of the competence concept, reliant on the target organization. Table 1 illustrates this issue, by presenting several definitions of competence gathered by Draganidis and Mentzas [24].

Table 1: Competence definitions by author [24]

Author	Definition
Marrelli [31]	Competencies are measurable human capabilities that are required for effective work performance demands.
Dubois [32]	Competencies are those characteristics (knowledge, skills, mindsets, thought patterns, and the like) that, when used either singularly or in various combinations, result in successful performance.
HR-XML [33] (www.hr-xml.org)	A specific, identifiable, definable, and measurable knowledge, skill, ability and/or other deployment-related characteristic (e.g. attitude, be- haviour, physical ability) which a human resource may possess and which is necessary for, or material to, the performance of an activity within a specific business context.
Boyatzis [34]	Boyatzis described competencies as underlying characteristics of an individual, which are causally (change in one variable cause change in another) related to effective job performance.
Selby et al. [35]	Selby described it as an ability expressed in terms of behaviour.
UK National Vocational Council for Vocational Qualification [36]	The National Vocational Council for Vocational qualification described competency as performance standards, the ability to perform in work roles or jobs to the standard required in employment.
Treasury Board of Canada Secretariat [37]	"Competencies" are the knowledge, skills, abilities and behaviours that an employee applies in performing his/her work and that are the key employee-related levers for achieving results that are relevant to the organization's business strategies.
Perrenaud [38]	A capacity to mobilize diverse cognitive resources to meet a certain type of situation.
LeBoterf [39]	LeBoterf says that competencies are not themselves resources in the sense of knowing how to act, knowing how to do, or attitudes, but they mobilize, integrate and orchestrate such resources. This mobilization is only pertinent in one situation, and each situation is unique, although it could be approached as an analogy to other situations that are already known.
Jackson and Schuler [40]	Competencies are defined as "the skills, knowledge, abilities and other characteristics that someone needs to perform a job effectively".
Intagliata et al. [41]	Most fundamentally, competencies provide organizations with a way to define in behavioural terms what their leaders need to do to produce the results the organization desires and do so in a way that is consistent with and builds its culture. They should provide the 'North Star' by which leaders at all levels navigate in order to create synergy and produce more significant and consistent results.
PeopleSoft	A set of measurable and observable knowledge, skills and behaviours that contribute to success in a job/position.
Gartner group	A competency is a set of characteristics, including skills, knowledge and attributes, that causes or forecasts performance.

#### 2.4.2 Competence Profiling

Generally speaking, a competence profile is a set of competences obtained from the observation of the behaviours, skills, knowledge and attitudes required to excel in a task. Hence, such outlines are mainly associated to employees. Moreover, they may be linked to projects, jobs or even stand as *ad hoc* profiles to express competence requirements/needs. However, since competences cannot be directly measured, sensed or observed [28], the question is: *how should organizations build competence profiles*? Likewise, *how should they use them*?

Regarding the second issue, competence profiles can be used as benchmarks for competence gap analysis. Performing a competence gap analysis allows the organization to assess breaches in performance, thus enabling the targeting of areas for training and developing. In essence, it denounces organizational competence needs through the comparison of (current vs. required) competence profiles. Figure 4 shows an example of a competence gap analysis, where gaps  $Gap_0$  and  $Gap_1$  are identified. As noted by Carbone et al. [42], the difference between the two tends to increase when the organization doesn't invest in the employees' competence development. In particular, the greater the range, the greater is the urgency to direct corporate competence development efforts to that target.



Figure 4: Example of a competence gap analysis (adapted from [115])

As previously stated, competence gap analysis facilitates the identification of breaches between competence profiles. Then, intuitively, if such profiles have different meanings, then so will have the identified gaps. Moreover, the more refined the competence profiles (being compared), the greater will be the information granularity about the gap. Thus, the targeting of competence development efforts can be more accurate.

#### 2.5 Summary

In the 21<sup>st</sup> century knowledge-intensive business scenario, a company's intellectual capital is crucial to corporate success. Hence, Competence Management (CM) approaches provide a strategic management of such resources, setting the basis for the creation of sustained competitive advantage.

This chapter focused on the review of CM main concepts and key challenges: competence modelling and profiling. Indeed, the construction of a competence model is no easy task, as there is no standard definition of the underlying concept - it is strongly dependent on the context in which it is used. Moreover, competence profiles are of pivotal importance, as they stand as viable benchmarks for competence gap analysis.

Having studied the theme of CM, the next step is to perform a bibliographic review on IT supported CM, learning how the analysed challenges are approached.

### Chapter 3

## **Competence Management Systems**

As stated by Draganidis and Mentzas [24], Competence Management (CM) is an important research object in the more general area of Knowledge Management (KM). It has a crucial contribution at organizational and personal levels, as it identifies the key knowledge that the corporation or its employee should possess in order to achieve their goals.

Competence and skills management have been tightly linked to organizational efforts towards the creation of solid grounds for workforce empowerment. In this sense, competitive advantage is promoted, and innovation and effectiveness are expected to increase [43]. Also relevant is that CM is directly related to corporate efforts to leverage internal knowledge and initiate consistent KM initiatives [44]. It is a research field which has recently begun attracting approaches to leverage personal development [45], knowledge sharing [46], corporate e-learning efforts [47] as well as applications of semantic technologies such as ontology management [48].

As part of KM, CM plays a crucial role to sustained organizational competitive advantage. Hence, enterprises deploy Competence Management Systems (CMSs) to support CM activities.

The goal of the present chapter is to review the most common and widespread CMSs and present the research directions of this field.

#### 3.1 Competence Management Systems: Review

Given the contextualization presented in the previous chapter, an assessment of existing Competence Management System (CMS) approaches is of pivotal importance at this stage.

With the aim to systematically analyse the main features of Competence Management (CM) as treated by existing systems, Draganidis and Mentzas [24] conducted a review of 22 CMSs capable of operating as standalone applications, as well as of 18 Learning Management Systems (LMS) which incorporate a CM module. Such information can be found in Table 2 and Table 3. That review and analysis reveals common features among the analysed targets, which are depicted in Table 4.

#### Table 2: List of CMSs reviewed [24]

CMS	Company	Website	
Skills gap management system	CAPSA, Consulting incorporated	www.capsaconsulting.com	
SkillBase	Claymore Inc.	www.skillbase.net	
On track for training and ontrack online	DKSystems, Inc.	www.dksystems.com	
Occupational skills analysis system (OSAS)	Educational Data Systems	www.edsincorporated.com	
KSA online	Gyrus Systems	www.gyrus.com	
HRScope	HR Technologies	www.hrscope.com	
TL4	Humeng	www.humeng.ca	
RemoteHR	Illion	www.remotehr.com	
Talent management system	KnowledgePool Americas	usa.knowledgepool.com	
Learner	LifeTime Media	www.ltmedia.com	
Meridian KSI knowledge centre	Meridian Knowledge Solutions	www.meridianksi.com	
SkillQuest	People Sciences	www.peoplesciences.com	
PeoplePower	People3	www.people3.com	
ClickEM	Peopleclick	www.peopleclick.com	
HR pulse	Pilat NAI	www.pilat-nai.com	
PlanView software	PlanView	www.planview.com	
Training and performance management (TAP)	Prismatics		
PROTECH	Proactive Technologies		
CompAssess	Right/Career Directions	www.careerdir.com	
SkillView enterprise 5.0	SkillView Technologies	www.skillview.com	
IntelliSkill and ITG databases	- Skill view Technologies		
SkillSolutions	SynTact Solutions	www.syntactsolutions.com	

LMS vendor	Product	CM features included
Click2learn, Inc.	Aspen Learning Management Server 1.1	Determine gaps in the individual's skills. It also tracks an employee's progress in skills acquisition.
Docent, Inc.	Docent LMS	Measures competency at a company level. Skill search capabilities. It tracks gaps be- tween student competency levels and job profile specifications, produces personalized learning plans to fill individual skill gaps and matches employees to job requirements us- ing skills-mapping capabilities.
Element K, LLC	KnowledgeHub (Learning management system)	None
GENESYS Software Systems, Inc.	PeopleComeFirst – hosted version, client requirements 1.0	Skills gap analysis and personalized learning paths.
	PeopleComeFirst – licensed, in-house version 1.0	
	PeopleComeFirst Healthcare Learning Advantage 1.0	
Hyperwave Informa- tion Management, Inc.	eLearning Suite 1.3	None
IBM	Lotus learning management system	None
KnowledgePlanet.com, Inc.	HCM Suite 6.0 KP Learning (No CMS Features)	It performs a skill gap analysis based on which a performance review or personalised learning path can take place. Through its learning content management system, it leverages the reuse of learning objects. Fi- nally, through its application interfaces and content development kit it provides organiza- tions and enterprises with access to skills data and competency models.
Learnframe, Inc.	Learning Center 1.5 Nebo 1.2	None
	Pinnacle Learning Manager 4.3	
Longview Solutions, Inc.	LRAL 3	None

#### Table 3: CM features included in the analysed LMSs [24]

Vendor of LMS	Product	CM features included
Open Text Corporation	Livelink for Learning Management 1.2	None
Oracle Corp.	Oracle iLMS	It integrates with HR systems in order to update employees' skills inventory and skills gap analysis.
OutStart, Inc.	Evolution 4.0	None
Pathlore Software Corp.	Pathlore LMS 5	Skills management
Pedagogue Solutions	PedagogueTesting	None
Plateau Systems, Ltd	Plateau 4 Learning Management System 4.2	Skills gap analysis and customization of curriculum according to it.
Saba Software, Inc. Saba Enterprise Learning Suite 3.5		
SAP GmbH	SAP Learning Solution	
THINQ Learning Solutions, Inc.	Learning Management System 5.x	A web based system which performs skills gap analysis.

#### Table 4: Common features of the analysed CMSs [24]

Feature	Description
Competency categories	A group that homogeneous or similar competencies belong to (e.g. people management competencies).
Competency definitions	Statement (s) that explains the basic concept of a competency.
Proficiency scale	Most of the CMS use a 0-4 or 0-5 scale, in order to grade the proficiency level that an employee possesses on a particular competency.
Behaviours associated with specific competencies	Behaviour indicators an individual should demonstrate if the specified competency is possessed.
Job descriptions	Job descriptions and definitions of employee roles.
Assessment tool	To identify and rate competencies.
Competency tracking by position, department and company	Most CMSs include a competencies' viewer which depicts the status of competencies possessed by an individual, a department or a company.
Link of skills gap analysis with a personalised learning path	The competencies that an employee should work on are linked with specific learning objects in order to facilitate their acquisi- tion.

*Microsoft* implemented a CMS by defining more than 300 competences in four categories (foundation, unique, global, and universal skills) [49]. Basic, working, leadership, and expert are the terms for different competence levels. In this approach, staff members are rated by themselves and by their superiors within an iterative process. Consequently, the company expects a better matching of employees to jobs and work teams [50].

Before introducing a CMS at *Ericsson*, they had individual solutions in each country, sometimes paper-based or relied on databases or spreadsheets. They extended the existing SAP R/3 Personal Management Module and implemented a competence catalogue. For the grades the corporation adopted a scale from 0 (not assessed) to 9 (excellent). The values are evaluated by using the period of time a person has used the competence [51].

Zelewsky et al. [52] describe a CMS designed to support the product development process in engineering networks made of different organizations. To support the understanding of competences and their development between organizations, an ontology is used. They match tasks that require certain competences with those of humans.

Dom and Pichlmair [53] have developed a prototype of a university CMS. The system is based on an ontology defining competences, evidence types and jobs in information systems and computer science domain. The system distinguishes competences in knowledge and an experience aspect. To decide on a competence, evidences such as passed examinations, books read by the person, trainings, assessments, project work, e-learning courses and more are stored. Each competence can be evaluated on three levels: beginner, advanced and expert and on each of these levels a finer scale exists with real values between 0 and 1. The competence of the individual is computed by the system when required and all evidences until that moment are used for the computation (historic perspective). A gap analysis algorithm determines differences between actual and desired goal profiles [54]. The computed gap is then used to recommend further courses to a student.

Taleo [55], a self-proclaimed talent management framework, sees employee competences as talent, thus supporting the entire employee lifecycle, talent management and development as well as career evolution. Figure 5 pictures an example of an employee performance review (evaluation) using this tool; please take notice of the presence of competence evaluations along with the employee's personal data.

O 🔆 Business I	Edition	TBETY   Help   Support   About   Logout	Application View1 Admo
Employees Revi	Users Administration Reports		
s Review Templates	Search Goals Goals Search		REVIE
4	Performance Review 2008 Annual Review * Each To Englished		
Employees for keywords:	Edit Delete Printable New Employee View Import Employee Values	Reset Submitted Rag	
Gol	👽 Review Data		
ed Employee Search	Review Code: Annual08	Review Manager: Nanney, Kevin	
lly Viewed Herris	Due Date: 12/31/06	Job Title:	
atham, Lara	Status: New	Manager Rating: ★★★★☆	
lanney, Kevin	👽 Employee Data		
and the second second	Employee Code: EMP000012	Department: HQ	
Surger Construction	First name: Lara	Manager Name: Nanney, Kevin	
1123]	Last name: Latham	Start date: 1/4/08	
lanney, Kevin	S Competencies		
Wyland, Bradley	Attendance & Punctuality		*****
Spaces. Eric	***	Nanager Comments	
loop. Jennifer	★★★★★ G Begins working on time		
Charles Marchine	★★★★★ @ Keeps absences within guidelines		
STREE TOWN	****		
Cooloo, Alex	★★★★★☆ @ Arrives at meetings and appointments on time		
Vewton, Murray	Communications		*****
Collina, Kevin	****	Manager Comments	
	★★★★★ @ Expresses ideas and thoughts in written form		
LE 30 DAY TRIAL	★★★★☆ @ Exhibits good listening and comprehension		
days remaining	★★★★★☆ Keeps others adequately informed		
period expires on: 3/31/09	★★★★☆ Selects and uses appropriate communication methods		
abscribe Today!	Leadership		*****
	★★★★☆ © Exhibits confidence in self and others	Manager Comments	
	★★★★☆ ◎ Inspires respect and trust		
	★★★★☆ @ Reacts well under pressure		
	★★★★☆ Shows courage to take action		
	★★★★☆ Motivates others to perform well		
	Employee Goals		
	Goals Summary		
		controller and shared as service of the service function in	

Figure 5: Example of an employee performance review on Taleo [116]

As noted by Draganidis and Mentzas [24], the majority of CMSs have been or plan to be enhanced with the use of web services in their development environments. As technology matures, the use of web services and XML is gradually allowing components of workforce management systems to read competence data from the Human Resources Management System "master" repository, updating the master with new entries or revisions based on completion of business processes.

Similarly, another area which could prove to be increasingly interesting for further research is the area of employee portals with self-service technology. Most organizations (mainly financial and telecommunications companies), have adopted or intend to deploy such portals [24]. In this sense, an employee self-service portal is integrated with the enterprise portal, thus facilitating tracking of specific competences which are easily self-reported (e.g. courses, knowledge acquired outside the organization scope). Such approach would address the problem of employees' lack of time and commitment to update the content of a CMS.

Moreover, ontology capabilities leading to the semantic web are becoming a core future technology [24]. In particular, ontology-based competence models can present a tight integration of capabilities and information in a highly contextualized user interface. Simultaneously, different services and components may be loosely coupled through a dynamic architecture whose coherence is ensured via a common semantic model in a rich competence ontology [24].

Enhanced by IT support, CM is a practice of increasing importance in private and public organizations. It provides them the means to attract and develop talented human capital, identify the right person for a job position, performing succession planning and training analysis, among other core human resources functions.

#### 3.2 Key Challenges and Literature Review

Concerning the key challenges presented in the previous chapter, this section is dedicated to the literature review on existing state-of-the-art approaches to those issues.

#### 3.2.1 Competence Modelling

There are multiple approaches to competence modelling.

The IMS Reusable Definition of Competencies or Educational Objective [56] and the later IEEE Reusable Competency Definition [57] focus on reusable competency definitions. The primary idea is to build central repositories which define competencies for certain communities. Hence, these definitions can be referenced by external data structures, encouraging interoperability and reusability. However, IEEE RCD lacks information on context and proficiency level and doesn't allow relationships or recursive dependencies among competencies.

On the other hand, HR-XML focuses on the modelling of a wide range of information related to human resource tasks (such as contact data or aspects of the curriculum vitae) [58]. The work performed in HR-XML Measurable Competencies [59] tries to define profiles in order to use such competency definitions. It specifies data sets like job requirement profiles (describing the competencies required for an employee) or personal competency profiles (stating the current competencies held by a person). Such profiles are composed of evidences (required or acquired), referring to competency definitions. However, the proposed model fails to clearly separate required from acquired profiles. As a consequence, an acquired competency could have mandatory and optional elements, according to the model. Moreover, it is unclear why a competency is composed of several evidences: since a competency is a reusable object, evidences should rather represent a requirement or demonstrate the acquirement of a competency [58]. Hence, the evidences should refer or contain competency definitions and not vice versa.

The Simple Reusable Competency Map [60] tries to model relationships between competencies. A map can contain information about dependencies/equivalencies among competencies, including the composition of complex competencies from simple ones. In this approach (SRCM), relationships are modelled using a directed acyclic graph. However, the semantics of the proposed model is confusing [58]. In particular, relationships among different nodes may have different meanings: composition, equivalence or order dependency. This leads to confusion when modelling tasks and when creating algorithms to use such information. Furthermore, combination and weighting of competencies is not clearly defined and external references to the maps (e.g. from profiles) must point to the root (and not to any node), therefore requiring the traversal of the graph until the appropriate node is found [58].

More oriented to competence gap analysis, De Coi et al. [58] propose a model for competence representation based on three dimensions: skill, proficiency level and context. In OntoProPer [61], profiles are described by flat vectors containing weighted skills (ranging from 0 to 3), which are expressed as labels. The system itself focuses primarily on profile matching and introduces an automated way of building and maintaining profiles based on ontologies. Colucci et al. [62] describes an ontology-based semantic matching between skills demand and supply. Lefebvre et al. [63] define a competence ontology for domain knowledge dissemination and retrieval. In this approach, a competence is related to capabilities, skills and expertise (measured by levels growing from 1 to 5).

#### 3.2.2 Competence Profiling

On the organizational level, the most obvious problem is getting the competency profile. One fundamental issue is that competencies cannot be measured, sensed or observed directly [28]. What we can observe is performance [64] in various forms: assessment of learning outcomes or performance in everyday job activities, to mention a few.

In general, two approaches can be observed [65]

- *Self-assessment*, where employees themselves are asked to provide their competencies, possibly mediated by their superior;
- *External assessment*, done by superiors or through formal assessment procedures.

White the latter is very expensive, cumbersome and only observable in limited areas [58], the first often fails due to missing motivation. This lack of incentive can be traced back to no immediate benefit for the employees. For instance, systems are hardly embedded into everyday work activities and have not proven their usefulness there [58]. As Becerra-Fernandez reports [66], in such scenario employees might downplay or exaggerate their competencies. Ultimately, an organization's competence catalogue may lack pivotal information due to the long update intervals.

Several studies address this problem by automatically extracting profile information from data the user generated in his/her daily work, such as publications [67] or documents [68]. Ley et al. [69] propose a competence performance approach that derives competencies from executed tasks. In this approach, a task competency matrix is created together with domain experts [58]. This matrix relates a set of tasks (e.g. required for a position) to a set of competencies needed to fulfil these tasks successfully. Based on this model, the system can infer a user's competency from his/her successful performance of a task in his/her daily work.

Another approach to employee profiles is based on community expert finding: (enterprise) social networking (e.g. LinkedIn [70] or Xing [71]). These platforms are based on the self-promotion paradigm: people can represent themselves with a profile and indicate their connections to other users. Moreover, the principle of social tagging and bookmarking is transferred to people. For instance, Xing allow the organization of contacts with tags. Within IBM's Fringe Contacts [58], each employee can describe their colleagues by tagging them with key words on their expertise and interests. This leverages network effects for setting up some sort of profile for the individual, tagging people with competences across the community. However, the resulting profiles lack legitimating and commitment by the organization [58], as the competence vocabulary is not directly supported.

#### 3.3 Summary

The purpose of a Competence Management System (CMS) is to control in a logical and integrated way activities within the company or organization that will assure and further develop competent performance at work. The aim is to ensure that individuals are clear about the performance that is expected of them, that they have received appropriate training, development and assessment, and that they maintain or improve their skills over time [72].

In this section a global review of CMSs was performed, pointing out their main features and comparing them. Similarly, a link was made to the previous chapter by reviewing the work done relative to the key challenges identified on Chapter 2: competence modelling and profiling.

Given this preliminary approach to CM and CMS, we proceed to describe this project's contribution – the application of CM principles and concepts to a corporate instance and the implementation of a CMS prototype.

### **Chapter 4**

## iKnow: a General Purpose CMS

At Critical (Manufacturing) we know our greatest asset are our employees and we understand it is the know-how, inspiration, energy and passion of our people that drives us forward. [73]

So far, this document has focused on the project's contextualization. It has addressed the issue of Competence Management (CM) in the corporate environment and the use of information technologies for this purpose – Competence Management Systems.

This section focuses on the instantiation of this problem (CM) to a given organization: Critical Manufacturing. In particular, the following topics are covered: a project's description, the presentation of the company, motivation and specific goals.

#### 4.1 Project Proposal

In a dynamic environment in which competition is strong, companies need to evolve quickly and to base their management on performance key factors, among which human resources play a major role [29]. Taking as premise that its employees dictate decisively the company's success [74], Critical Manufacturing craves for an efficient management of its internal know-how. This project realizes that need by proposing the development of a Competence Management System.

More than a trivial employees' skills data warehouse, this assignment envisions a Competence Management transversal to the diverse aspects of human resources management, such as training, people-to-project allocation, recruitment and internal competence evaluation processes, to mention a few.

#### 4.2 Client Portrayal: Critical Manufacturing

At Critical Manufacturing our main competitive advantage is to be able to deliver high-quality solutions in a competitive business model, addressing our customer needs and ensuring their final satisfaction. [74]

Created in 2009, Critical Manufacturing (CMF) - Figure 6 - was based on a joint business plan between the Critical Group and individual founders with large experience in semiconductor business [74]. It is devoted to the delivery of *avant-garde* solutions for the high-tech manufacturing market, aiming at becoming the prime solution provider for that niche.



Figure 6: Critical Manufacturing logo [74]

CMF is part of the Critical Group (Figure 7), a private set of companies founded in 1998 to provide solutions for mission and business critical information systems. It is based on an experienced services team coming from the Critical Group and established a product team of individuals with more than 10 years experience in high-tech manufacturing.

Critical SGPS (holding company for the Critical Group) integrated the formerly Critical Software's semiconductor manufacturing solution business unit into Critical Manufacturing, where it formed the services business division. With an annual turnover of 3 Million euros and approximately 50 employees, this unit has competed in several manufacturing areas, ranging from integration solutions to data analysis and business intelligence areas [74].



Figure 7: Location of Critical Manufacturing within the Critical Group [74]
The individual founders have over 10 years of experience in the area of solutions for semiconductor, and along with a group of top analysts and developers with approximately 30 persons, formerly part of a team that created solutions for semiconductor production facilities over the past 6 years, formed the product business unit of Critical Manufacturing, which creates solutions for manufacturing systems for high-tech industries, with strong focus on semiconductors, solar cells, flat panel and other electronic industries [74].

These two business units (services and product) are the core of CMF. The company comprises over 80 people at its various sites, with headquarters and main technical centre in Maia, Portugal and additional facilities in Regensburg (Germany) and Suzhou (China).

CMF's continuous success is built on a solid track record of on-time and on-budget projects for a growing list of satisfied customers [74]. Much patent in its slogan, CMF takes its know-how and delivers an ambitious yet sustained attitude:

Unleash the power of manufacturing<sup>2</sup>

# 4.3 Motivation

This project aims to improve Knowledge Management at Critical Manufacturing (CMF), namely using a Competence Management (CM) strategy.

For its dimension, CMF possesses a large set of workers and associated projects (planned, in progress or closed). If you consider that each individual has a set of skills, hence competences, the problem of managing these competences becomes exponentially complex as workforce load increases.

Moreover, the company lacks a centralized repository of its workers' competences transversal to all business relevant activities, such as project management, competence development (training) and even gap analysis. Instead, all relevant information is found throughout CMF's internal systems, forcing a manager to stumble upon these numerous tools, gather all relevant information and then place it in a central and unique infrastructure, for instance an Excel spreadsheet (Figure 8 and Figure 9).

<sup>&</sup>lt;sup>2</sup> Critical Manufacturing's slogan (source: [74])

iKnow: a General Purpose CMS



Figure 8: Example of an Excel sheet used for internal CM at CMF

Knowledge Index	PM	MT	DBA	Data Model	Converters	STDF Loader	EXF Loader	Rules Engine	Loader Oper & Ana	Lot Summary Export	Wſ	Install Scripts	WP Scan	Admin GUI	TAS
Skills	PM, Soft Skills	TTM, Soft Skills	Oracle Tuning	Orade, SQL	Unix, C++	Unix, C++	Unix, C++	PL/SQL, SQL	PL/SQL, SQL	++ 5	C++		Unix, C++	Java, JSP	

Figure 9: Detail of Figure 8, where a sample set of competences can be identified.

Critical (in particular CMF) possesses a wide range of tools for internal management related to competences, such as *WISE*, *CoreCompetence* or *Learning Center*. However, none of these is specifically oriented to CM, lacking the set of complementary activities studied in section 2.3 (Competence Management: Strategic Perspective). In particular, the following is verified:

- *WISE* (Figure 10) is Critical's ERP, which, among other features, allows for employee and project management. However, it lacks competence mapping and evaluation.

- *CoreCompetence* (Figure 11 and Figure 12), more closely linked to workers' competences, focuses solely on competence evaluation through performance appraisal and training programmes surveys.

- Learning Center (Figure 13), an e-learning platform, is oriented to competence acquisition, providing tutorial material. However, it lacks competence management and evaluation features.

Wise - Gestão	× 🎽 Info About Our Tools 🛛 🗐 🚾 Wise	e - Colaboradores 🛛 🛛 🕹	Critical Learning Cen	ter × 0		- Ø
C fi tr		elizioniter-				▲ ► D-
	LOI					
				Prenerences	Here Inspect Accord Leased In-	
		CONCERNS	QUALITY RE	SOURCES EVENTS	PT IN	
	Internets Incertains Com	in concinity			Auto / press	
	Worker List			Login / N ► <u>Adxans</u>	ame :	
	[First/Prev] 1 2 2 4 5 6 [Next/Las	3)			110 items found, displaying 1 to 20	
	Name	Location Comp	aný Area 🔅	Contacts	E-mail 0 Photo	1
	Addition of the Association of t	Porto (TecMaia) CMF	CMF - PBU	ALTERNAL COMMENSION		
	Averagine Consist Press (Patient)	Porto (TecMaia) CMF	CMF - SBU-ENG	Alfreda	<b>F</b>	
	multilitation Phonese Uterane.	Porto (TecMaia) CMF	CMF - SBU-ENG	ADATAS.		
	micciael Mension discriming.	Porto (TecMaia) CMF	CMF - SBU-ENG	Allerani		
	and the Personal Part	Other Location CMP	Shared Services	Alfren : Re: Research : Street Balance		
	piece (Personi Perint) Prime.	Other Location CMP	Shared Services	Allerten. Res. Transmitter Res. Transmitter		
	Applie: Messal Crister (Apple)	Porto (TecMaia) CMF	CMF - PBU-ENG	Alfren		
	ann - Marsan - Italianan - Spithian	Porto (TecMaia) CMF	CMF - SBU-ENG	ADDORES: Mic		
	son Charle President Charlester.	Porto (TecMaia) CMF	CMF - PBU-ENG	Alteriani Rec		
	101-12100-1799-0-0-10-1799-	Porto (TecMaia) CMF	CMF - ADMIN	Alteriate Rec.		

Figure 10: WISE in action, showing a list of CMF's workers



Figure 11: CoreCompetence in action, showing a list of a worker's self-evaluation parameters

Competence	Grade	Personal Comments
Non-Technical Competences		
R Training Academic Quality 0	N/A A B C D E F G H I I I I I I I I I F- Good 	
Competence Description: Did the ac execute well the activities defined.	ademic training have provided the competencies required to	
In what extend the quality of acades training contributed to a gr performance during the internship?	mic N/AABCDEFGH ood IIIIIII + F-Good	
Setail Design and Coding 0	N/A A B C D E F G H I I I I I I I I I I B 	

Figure 12: Detail of Figure 11, where there is a list of competences to be evaluated, which the user can grade and add personal comments to

	× \ Info About Our Tools × \ 🐱 Wis	e - Colaboradores X [[[ Critical Learning Center X		
← → C fi ☆	https://cbrapps.critical.pt/moodie/index.php			<b>∂</b> ► <b>□</b> - ⊁-
	MAIN MENU MAIN MENU Site News MESOURCES Nice and free DVD players Training Platforms	COURSE CATEGORIES e-Library Internal Workshops critical Assessment Center Search courses: @	CLIC aims to be the best resource for training materials, provide training courses and allow interactive activities between teachers and students.	

Figure 13: *Learning Center*, an internal CMF application focused primarily on competence development through e-learning

Moreover, a common topic among these platforms is that they lack a competence gap analysis tool. Hence, an efficient Competence Management System is needed to bridge this flaw. From a broad perspective, this issue comprises four key streams of concern:

- *Competence Profiling*: a set of competences should be associated to employees, projects, jobs or even *ad hoc* profiles;

- *Competence Evaluation*: means should be provided to quantitative and qualitatively assess employee's competences, whether associated to their performances at a given task, project or even training programme;

- *Competence Training*: competences ought to be developed through time, thus empowering the organization's workforce, whether by the acquisition of new competences or by extending acquired ones;

- *Competence Gap Analysis*: transversal to other three issues, a 'fit' should be determined between the current competence values and the aimed ones. The latter can be measured against the following:

- Other competence profiles (other employees, projects, jobs or *ad hoc*);

- Global aimed corporate competences;
- Competences expected after a training programme.

Such feature would allow the search for the most adequate workers for a given profile, ranking by gap analysis.

## 4.4 Goals

The ground objective of this project is to study the impact of Competence Management (CM) in an organization. In particular, the requirements and design of an effective Competence Management System (CMS) should be identified and a working prototype ought to be implemented. Thus, the following key features are suggested:

- Storage, edition, report and search of a vast set of data needed of the resources management of the Engineering Area in the Service unit;
- Default rich and detailed reporting set including statistical analyses to support the decision making process;
- Web interface for data access;
- Extensibility;
- Interoperability with resident applications.

The system must include a Web interface for data access and interact with other existing support decision systems.

The envisioned CMS shall be used by Critical Manufacturing (CMF) workers. That includes interns, engineers and managers (project or area supervisors). It is indended to support the decision making process for an estimate of 50-249 employees. If CMF has prospects of extending this system to other Critical groups, then an exponential increment of users (and data overload) must be considered.

The product targets users that are somehow interested in CM let it be core tasks such as the rating of competences or skill evaluation or even more complex work like gap analysis of a set of workers' competences.

# 4.5 Summary

Through its methodological approach to an organization's internal knowledge dynamics, a Competence Management System (CMS) provides the means for an enhanced management of the corporate intellectual capital and, ultimately, leverages its competitive advantage among its peers.

Although it possesses a set of applications related to competences (*WISE*, *CoreCompetence*, *Learning Center*), Critical Manufacturing lacks a systematic approach to its internal Competence Management. Consequently, such task is accomplished through cumbersome processes, such as manual data mining and *ad hoc* Excel style sheets analysis. Hence, a CMS is of pivotal importance for this organization.

#### iKnow: a General Purpose CMS

If a company's main asset is its workforce, how can it be productive if its employees' knowledge is not correctly used? Thus, skills can be overlooked and success might be compromised.

Hence, as the title of this dissertation states, it is intended the study, design and implement a working prototype of a *Competence Management System towards Improved Corporate Success*.

# Chapter 5

# **Requirements Analysis**

The hardest part of building a software system is deciding precisely what to build. No other part of the conceptual work is as difficult as establishing the detailed technical requirements, including all of the interfaces to people, to machines, and to other software systems. No other part of the work so cripples the resulting system if done wrong. No other part is more difficult to rectify later.

- Frederick Brooks [75]

This section is dedicated to the envisioned system's requirements analysis.

Loosely speaking, iKnow intends to improve Knowledge Management in Critical Manufacturing (CMF), namely regarding the Competence Management strategy. More than a simple skills' storehouse, the envisioned system shall provide the tools for a (near) optimal management of CMF workforce's skills: core competence management, employee's coaching, talent management and historical analysis of acquired skills, to mention a few.

The user scenario consists of a computer connected to the Internet, independent of the operating system, where accessing the application is done via any Web browser.

## 5.1 Users Characteristics

The envisioned system shall be used by CMF's workers. That includes interns, engineers and managers (project or area supervisors). It is indented to support the decision making process for an estimate of 50-249 employees. If CMF has prospects of extending this system to other Critical groups, then an exponential increment of users (and data overload) must be considered.

The product targets users that are somehow interested in Competence Management let it be core tasks such as competence evaluation or more complex work like gap analysis on competence profiles.

Of pivotal importance is CMF's hierarchy of user roles. Strongly related to competence evaluation/performance appraisals, this corporation deploys the concepts of *Facilitator* and *Coach*. Usually a project/team manager, a *Facilitator* is directly responsible for one or more worker's periodic competence appraisal and must be his/her supervisor. On the other hand, a *Coach* may not be a worker's supervisor and doesn't directly appraise competences. He/she considers each worker's individual profile and, accordingly, guides the worker towards (near) optimal performance, advising him/her on competence development directions.

# 5.2 Restrictions

Given the envisioned system, the following constraints must be considered:

- To perform any competence management related activity, a user must be successfully authenticated in the system;
- A worker cannot be coach, facilitator or manager of another hierarchically superior worker.

# 5.3 Assumptions and Dependencies

For its web-based nature, iKnow shall be independent of the user's operating system. Thus, an Internet or Intranet connection is required (depending on the system's location and access constraints imposed by CMF). Equally mandatory is a Web server to host the application.

# 5.4 External Interface Requirements

As to define the interfaces to other systems and external entities within the project scope, the following external interface requirements were identified:

- *User interfaces*: an intuitive, simple and usable user interface is required due to the expected frequency of its use. It should also be competence oriented;
- *Hardware interfaces*: the system must be hosted in a Web server with a connection to a dataset. On the client side, an Internet connection device is required;
- *Software interfaces*: iKnow's usage relies on a web browser. A database management system is also required to perform all data storage-related operations;
- *Communication interfaces*: the system shall be accessed and used online, based on a standard HTTP protocol. Furthermore, the system is expected to communicate with other resident applications at CMF. In this sense, it must provide a proper software interface as to be integrated with those systems and also access them to retrieve relevant data.

# 5.5 Functional Requirements

### 5.5.1 User Stories

Used in agile software development methodologies, a user story is a faster means of handling customer requirements without formalized documentation overhead and with increased adaptability to mutable real world scenarios.

Tables Table 5-Table 12 present the user stories for the aimed Competence Management System (CMS), grouped by feature type.

Core Competence Management					
Identifier	As a/an	I want to	so that		
US-CCM-1	manager	create, view, update and delete/disable competences	I can manage the competences associated to the workers		
US-CCM-2	manager	categorize competences	I can relate a group of tasks with common grounds		
US-CCM-3	manager	create competence derivation graphs	competence profiling overhead can be minimized		
US-CCM-4	manager	rate a competence according to a worker's experience and level of expertise (e.g.1 to 5)	workers may be distinguished by their skills' experience and train- ing		
US-CCM-5	manager	associate a competence to a worker	team's competence's are updated		
	manager	Obtain a set of competences that are de- duced from others I currently possess	I don't have to rate every single		
US-CCM-6		Note: The deduced competence should have a special notation to distinguish it from the regular ones, such as a "pending approval" tag.	competence – reduce competence editing overhead.		
US-CCM-7	worker	define a set of "desired competences" I wish to acquire	my competencies can evolve and expand towards improved per- formance and career progression		
US-CCM-8	worker	associate and/or remove a set of compe- tences to/from my profile	I can have my own competence profile updated		
US-CCM-9	worker	rate my profile's competences	I can have a detailed profile by competence expertise and experi- ence		

#### Table 5: iKnow's user stories for core competence management

Training					
Identifier	Gap Analysis				
	As a/an	I want to	so that		
US-TR-GA-1	manager or facilitator	perform a gap analysis on a set of users (singleton or not) from my hierarchical level down	I'm able to learn my team's training needs		
US-TR-GA-2	manager or worker	associate a time frame to a gap analysis.	I can restrict the gap analy- sis to a period of interest		
US-TR-GA-3	manager or worker	compare gap analyses from distinct time frames	I can perceive the evolution of the analysed competences over time		
US-TR-GA-4	manager or worker	obtain training recommendations based on a gap analysis, where competence levels of a user are analytically examined and compared to its ideal values.	more suitable training pro- grammes can be made		
		For example, a user may be enrolled in a project where he/she is expected to have a level 5 skill but only possess level 2. This gap analysis should also recommend features to be acquired by the analysed user and priori- tise them.			
US-TR-GA-5	manager, coach or facilitator	perform gap analysis on a set of workers by competence or on a set of compe- tences by worker	I may analyse which compe- tence training is lacking for a set of workers		
		Note: I can only analyse workers that are not hierarchically superior to me.			
US-TR-GA-6	manager	produce an overall analysis of the com- pany's competences, presented by text and/or graphics	I can see the "big picture" of corporate needs, rather than focus on a restricted set of users		
US-TR-GA-7	worker	perform gap analysis on myself	I can perform an auto-evaluation of my own skills		

# Table 6: iKnow's user stories regarding training lookup and evaluation

	Evaluation				
	As a/an	I want to	so that		
US-TR-EV-1	manager , coach or worker	qualitatively assess (rate) training pro- gramme: was it useful? Difficulties and drawbacks?	more adequate training pro- grammes can be chosen in the future		
		Note: A manager / coach can only assess a training programme of his/her subordi- nates, while a worker is restricted to a training programme he's participated in.			
US-TR-EV-2	manager, facilitator, coach or worker	<ul> <li>qualitatively assess a training program through gap analysis of a worker, consider- ing:</li> <li>evolution of competences</li> <li>projects he/she's participated in and associated competences that were ap- proached in his/her trainings</li> </ul>	more adequate training pro- grammes can be chosen in the future		
		Note: A manager can only assess a train- ing programme of his/her subordinates, while a worker is restricted to a training programme he's participated in.			
US-TR-EV-3	manager, coach or worker	gain a historic perspective of a worker's trainings and competences evolution	I can analyse the efficiency and worthiness of training programmes		
		Note: A manager / coach can only assess a training programme of his/her subordi- nates, while a worker is restricted to a training programme he's participated in.			
	Lookup				
	As a/an	I want to	so that		
US-TR-LU-1	manager, facilitator, coach or worker	search for adequate/advised training pro- grammes for a set of workers (singleton or plural) considering each particular profile: competences, projects he/she's enrolled in and desired competences vs. available courses <i>Note: A worker can only perform this</i>	workers may improve their competences		
		analysis on him/herself. A facilitator or coach may perform this analysis if and only if he is responsible for that target set of users. In all cases an analysis cannot be performed on someone hierarchically superior			

# Table 7: iKnow's user stories related to coaching

Coaching					
Identifier	As a/an	I want to	so that		
US-COA-1	worker	become coach of another worker if and only he/she's not hierarchically superior to me	I may help my colleague to improve his competences		
US-COA-2	coach	recommend that the worker I'm coaching acquires a set of competences	I may help my colleague to improve his competences		
US-COA-4	coached worker	ask for training recommendations from my coach	I may get a broader perspec- tive of my training needs		

#### Table 8: iKnow's user stories for dashboard features

Dashboard				
Identifier	As a/an	I want to	so that	
US-DSB-1	worker	be presented with a customizable dashboard upon login	my personal area only shows data that is relevant to me	
US-DSB-2	worker	be able to configure my dashboard	my personal area only shows data that is relevant to me	

# Table 9: iKnow's user stories regarding alerts

Gadgets					
Identifier	Alerts				
	As a/an	I want to	so that		
US-GD-AL-1	worker	have an alert gadget, similar to a feed reader (feed of alerts)	I can have a non overloaded notification tools (as oppo- site to e-mail, for example)		
US-GD-AL-2	worker	ignore, accept or delete alerts	I may customize my alerts		
US-GD-AL-3	manager, facilita- tor or coach	upon login, be presented with relevant notifi- cations such as updated competences from workers I am related to (if I am a facilitator, coach or manager)	relevant information is pre- sented		
US-GD-AL-4	worker	see upcoming training courses and projects of interest (I'm enrolled in or eager to participate)	relevant information is pre- sented		

		Talent	
Identifier	Sniffing		_
	As a/an	I want to	so that
US-TA-SF-1	manager	<ul> <li>search for adequate workers for a given project, namely regarding:</li> <li>their competences vs. project's required (and preferential) competences</li> <li>workers training and associated deadlines</li> </ul>	I may allocate the best suit- able workers to a project
US-TA-SF-2	manager	be able to sort the list of adequate workers. By de- fault I wish to obtain a ranked list, but I should be able to customize the sorting	I can rapidly access relevant information
US-TA-SF-3	manager	search for workers with a set of competences and sort it	I can rapidly check who knows what and at his/her level of expertise without depending on a project for the search
	Scouting		
	As a/an	I want to	so that
US-TA-SC-1	manager	obtain job interviews associated to a project, busi- ness area or even a customized set of competences ( <i>ad hoc</i> profile)	I have a standard profile for competences
US-TA-SC-2	manager	manually create job interviews associated to a pro- ject, business area or even a customized set of com- petences ( <i>ad hoc</i> profile)	I have a standard profile for competences
US-TA-SC-3	manager	suggest changes in existing job interviews that must be approved by the corresponding examiners	job interviews are more adequate to the target profile
US-TA-SC-4	examiner	be able to configure the desired job interview prior to its creation	generated job interviews are closer to the aimed profile
US-TA-SC-5	examiner	be capable of editing a generated job interview questions	created job interviews are closer to the aimed profile
US-TA-SC-6	examiner	associate a time frame to a job interview	job interviews can be tem- porarily located
US-TA-SC-7	examiner	play/resume and pause a job interview	I can control the interview's evolution
US-TA-SC-8	examiner	create, edit and delete examinee data that is not related to the generated job interview	I may have a coherent and straightforward job inter- view
US-TA-SC-9	examiner	upon the completion of a job interview, obtain a gap analysis report, where it mirrors the difference be- tween the target profile's required skills and the examinee's evaluated competences	I may understand "how fitting" the exami- nee/candidate is for the target profile
US-TA-SC-10	examiner or examinee	create, edit and delete job interview responses	the examinee can perform the job interview
US-TA-SC-11	examiner or examinee	obtain a set of questions that are only visible to me	the examinee and examiner may edit private data

# Table 10: iKnow's user stories regarding talent sniffing and scouting

Social Network					
Identifier	As a/an	I want to	so that		
US-SN-1	worker	be inserted into a social network of worker's com- petences	I can be part of a compe- tence network		
US-SN-2	worker	manage my social network friends	I can expand my compe- tence social network		
US-SN-3	worker	communicate with my social network colleagues	I can exchange messages with my peers		
US-SN-4	worker	manage social network groups	I can join groups of interest within the social network		

### Table 11: iKnow's user stories concerning social network characteristics

### Table 12: iKnow's user stories regarding security/privacy

Security/Privacy			
Identifier	As a/an	I want to	so that
US-SEC-1	administrator	manage users: create, edit, delete and privileges	users can access the system with different levels of in- formation access
US-SEC-2	user	be able to log off the system	I may utilize the system
US-SEC-3	user	obtain a temporary examinee login	I may carry out a job inter- view
US-SEC-4	visitor	see generic information about the system	I can know about this com- petence management system
US-SEC-5	visitor	register	I can gain access to the system

## 5.5.2 Use Case Models

This section describes the main use cases for iKnow. Please note the priorities that are assigned to the use cases, with the subsequent significance [76]:

- *Essential*: if not implemented, the application will not run. These use cases are mandatory;
- *Important*: if not implemented the application will run but in a unsatisfactory manner;
- *Desirable*: if not implemented, the system operation is not compromised. These types of use cases may be postponed for future developments.

The use cases are presented per module, detailing properties such as its identifier, name, brief description, involved actors, use case priority, main and alternative scenarios, pre and post-conditions as well as assumptions. Use cases not here described are in Annex A.

### A. Overview

As with user stories, eight modules of features can be identified, to be analysed in the upcoming sections: *Core Competence Management, Training, Coaching, Talent, Dashboard, Gadgets, Social Network* and *Security/Privacy.* Table 13 describes the mentioned modules, while Figure 14 portrays the corresponding use cases.

Identifier Name			Description	
MD-CCM Core Competence Management		nagement	Operations on competences and associated values	
	Sub modules			
	MD-CCM-CE	Competence	evaluation	Tools to qualitatively and quantitatively assess a worker's skills
	MD-CCM-CD	Competence	derivation	Features to relate competences
		Sub module	!	
		MD-CCM-CI	D-DG	Derivation graph Means to logically relate competences
	MD-CCM-WL	Wishlist		Record for worker's personal competence training aspirations
	MD-CCM-BM	Basic manag	ement	Elementary management operations on competences
	MD-CCM-HR	Hierarchy		Tools to relate competences according to their importance
	MD-CCM-CP	Competence	profile	Features to relate competences to workers
	MD-CCM-CG	Competence	grouping	Means to group related competences without restrictively considering their hierarchical position
	MD-CCM-IE	Import/expo	rt	Means to increase the system competences' portability
MD-TRN	Training	Features regarding workers' instruction		
	Sub modules			
	MD-TRN-GA	Gap analysis	Tools to r tences and	measure the breach between workers' ideal/recommended compe- d their real values, thus identifying training needs
	MD-TRN-EV	Evaluation	Features t	to understand the worthiness and efficiency of competence trainings
	MD-TRN-LU	Lookup	Search en	ngine for a successful match between competences and trainings
MD-COA	Coaching	g Collabor	ation betwe	een workers towards improved levels of competence expertise
MD-TAL	Talent	Tools to	discover we	orkers with a specific competence profile
	Sub modu	les		
	MD-TAL-SI	F Sniff	ing Featu	rres to find adequate workers within Critical employee scope
	MD-TAL-SO	C Scou	ting Tools	s to recruit new talents
MD-DSB	Dashboa	rd Desktop	like enviro	onment where all relevant information is presented to the user
MD-GAD	Gadgets	Set of to	ols for an ir	mproved work experience within the CMS
	Sub modu	le		
	MD-GAD-A	AL Alert	s Noti	ification widget
MD-SNT	Social No	etwork En	vironment to	to support a competence network
MD-SEC	Security	Privacy Fea	tures for us	ser and privileges management



Figure 14: Outline of iKnow's use cases

### **B.** Actors List

According to the user characteristics and feature modules previously identified, the following actors are identifiable: *Visitor*, *Administrator*, *Worker*, *Facilitator*, *Coach*, *Coached Worker*, *Examiner* and *Examinee*. Table 14 and Figure 15 describe the system actors.

Table 14: List	of system actors
----------------	------------------

Identifier	Name	Description
ACT-VS	Visitor	User who hasn't logged in the system and can only access publicly available information
ACT-AD	Administrator	User responsible for user management and maximum privileges
ACT-WK	Worker	User who works for the Critical group, namely Critical Manufacturing
ACT-FA	Facilitator	User who is responsible for a set of workers, evaluating their competences.
ACT-MG	Manager	User who coordinates a set of workers
ACT-CO	Coach	User who helps a group of workers to achieve ideal competence profiles, accompanying his/her coached workers throughout relevant career mile- stones. This actor periodically evaluates his coached workers, not exclu- sively related to their competences: their difficulties, satisfaction level and evolution, to mention a few. This information is then reported to the respon- sible Facilitator, who evaluates his/her workers.
ACT-CW	Coached Worker	User that's being coached
ACT-EX	Examiner	Worker that carries out a job interview
ACT-Exe	Examinee	User (may be worker or not) who's being interviewed



Figure 15: System actors' overview

### C. Core Competence Management

Focused on core management activities, this module incorporates the following modules, as pictured in Figure 16: *Competence evaluation*, *Competence derivation*, *Wishlist*, *Basic management*, *Hierarchy*, *Competence profile*, *Competence grouping* and *Import/export*.



Figure 16: Core Competence Management use cases

As the reader may recall, in section 2.3 CM was described as a strategic process of multiple phases. In particular, a significant phase was competence evaluation through performance appraisal. The *Competence Evaluation* module translates that stage. Two types of evaluation are approached: self-evaluation (Table 15 describes an example) and facilitation. In the first, a *Worker* evaluates his/her own competences (from his/her competence profile) using a rating scale defined by the organization. An analogous use case is carried out by a *Facilitator*, where he/she appraises he/she's responsible for. A *Coach*, on the other hand, takes competence evaluation as a means to establish competence risks and suggests actions to the analysed worker. A competence risk is assessed when a worker's evaluation reveals severe flaws. In sum, this module contemplates the following use cases:

- A Worker may auto-evaluate a competence and rate a competence;
- A Facilitator can evaluate a worker's competence;
- A Coach may assess competence risks and suggest actions associated to a competence level of expertise.

Identifier	UC-CCM-CE -2	
Name	Auto-evaluate a competence	
Description	Actor rates his/her own competences.	
Actor(s)	Worker	
Priority	Essential	
Scenarios		
Main	The actor accesses his/her own profile, clicks on the evaluation menu and sees a list of compe- tences to evaluate.	

#### Table 15: Auto-evaluate a competence use case



	Interface Prototype					
	i C	Home My Profile (action1). My Competences		Logout My gadgets		
		υ				
	Г	Name	Evaluation	Level		
		Competence A	5	?	PA -	
		Competence B	2	2	80	
		Competence C	1	?	80	
		Competence D	4	2	80	
		Competence E	1	?	80	Alerts
	L			•		Competency
		Rated competences				Other info
	5	Name	valuation	vel	—————————————————————————————————————	Two Labels, and a yup
		Competence A 5	Ve	ry good	80	✓ Alert read (>)
		Competence B 2	Lo	w	PA	• A Bullet 📃
		Competence C 1	Ve	ry Low	PA	
		Competence D 4	Go	ood	PA	Conf CONFIGURE
		Competence E 1	Ve	ry Low	8AL	Hide OFF
	L		I			
Alt	ernative	The actor cancels the	operation.			
Pre	e-conditions	Actor must be logged	l in.			
Pos	st-conditions	Actor can view the complete competence evaluation.				

Considering competence profile management, every time a user gained a new competence that should be manually introduced by his/her superior. However, if you consider an exponential growth of both competences and supervised workers, then the complexity and data management overload for the supervisor becomes cumbersome. Hence, the *Competence derivation* module intends to diminish that burden. In essence, a *Manager* can construct a directed competence derivation of its predecessor(s). For example, consider the competences 'C Programming Language' and 'Imperative Programming'. If your worker knows the first, then (supposedly) he will know the latter. Thus, an edge can be added to the competence derivation graph between these two competences, as pictured in Figure 17. Table 16 details the use case for a competence addition/removal to/from a derivation graph.



Figure 17: Example of a competence derivation

Similar to *Competence derivation*, the *Hierarchy* module also relates competences to each other. The main difference in this case is that the association is through a particular type of graph: a tree. While in a competence derivation graph a competence can have any number of

predecessors, in a hierarchy there is only one (the parent node). The goal of such approach is to be able to group competences in a structured and hierarchical way. The main practical impact on users is that, when associating a given competence to a worker's profile, the sub tree of the competence (node) is also imported. Once again, there is an effort to minimize all data manipulation efforts by the user. As an example, consider the competences '.NET', 'C#', 'Windows Presentation Foundation' and 'XAML'. The last three technologies are part of the .NET framework. Moreover, XAML is an extensible markup language used in Windows Presentation Foundation which, in its turn, is also part of .NET. Hence, these competences could be organized as a tree (Figure 18).



Figure 18: A competence hierarchy example

#### Table 16: *Add/remove competence* use case

Identifier	UC-CCM-CD-DG-1	
Name	Add/remove competence	
Description	Actor adds/removes a competence from the derivation graph.	
Actor(s)	Manager	
Priority	Desirable	
Scenarios		
Main	Actor accesses the competence menu and chooses to edit a competence derivation graph. There, he/she may add/remove competences to the graph.	

#### **Sequence Diagram**

The sequence diagram for the 'add competence' scenario is presented. For competence removal the behaviour is analogous.



Interface Prototype				
The interface	prototype for the 'add competer iKnow Home My Profile (action1) My Competence Derivation Gra Choose competence graph V Details Competence graph's properties Name Competence A Competence B Competence C Competence C C Competence C C Competence C C Competence C C Competence C C Competence C C Competence C C C C C C C C C C C C C C	nce' scenario is presented. aphs Picture with the selected graph's representation Here specify competence predecessor and successors	Logout My gadgets My gadgets Alerts ⓒ Competency ⓒ Other info Two Labels, and a yup ✓ Alert read (>) ● A Bullet E Conf CONFIGURE Hide CONFIGURE Hide	
Alternative	The actor is viewing a competa competence. In the former, a menu to choose the system's option is displayed that can be from the graph.	tence derivation graph. In this ca the actor clicks an option 'add co s competence. In the later, next t e clicked. Upon actor confirmati	ase, he/she may add or remove ompetence' and is redirected to to each competence a 'delete' on, the competence is removed	

Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view the derivation graph, in textual or graphical form.
Assumptions	Actor cannot add a competence already present in the graph nor remove a competence that's being edited by another user. Additionally, he/she cannot remove a competence that does not exist in the graph.

There can be cases when there is an interest to associate competences but without an inherent structure, such as a graph. If you are a *Facilitator*, you are in charge of evaluating workers. However, workers have different competence profiles which, probably, will not be the same as your own. Hence, it makes sense that a *Facilitator* only appraises a given set of competences, according to his/her knowledge domain. Such set is provided by the *Competence grouping* module. Indeed, it provides the means to associate competences as a generic group, independent of their hierarchy or derivation graph. The practical impact of such approach is you are able to define that a *Facilitator* appraises competences that are in a given group. This sets way for a more coherent process appraisal, since you can control which competences a *Facilitator* is allowed to appraise, thus avoiding that he/she evaluates competences out of his/her knowledge scope.

In an organization, each worker possesses a set of competences. Accordingly, the *Competence profile* module is responsible for the construction of such profiles, thus linking competences to workers.

Although every employee possesses a particular competence profile, he/she may wish to acquire new competences or developed existing ones. In order to express his/her knowledge aspirations, the *Wishlist* module allows each user to create a ranked list of competences that the worker wishes to develop. Such feature can be useful when searching for training programmes

or projects, acting as a constraint to such decision making problems. Table 17 details a use case for the edition of a competence wishlist.

Considering basic operations on competences (CRUD), *Basic management* contemplates allows for the creation, edition or visualization of organizational competences. Similarly, the *Import/export* module allows the portability of competence appraisals and hierarchies. For example, a competence hierarchy can be imported from an Excel sheet.

Identifier	UC-CCM-WL-1		
Name	Add/remove preferable competence		
Description	<u>Add preferable competence</u> : Actor adds a competence to a wishlist. This means that the user is eager to acquire that competence.		
	<u>Remove preferable competence</u> : The actor accesses the wishlist menu and sees competences in that list. Then he/she clicks on the 'remove competence' option and a list of competences appears. He/she chooses a set of competences to remove and submits.		
Actor(s)	Worker		
Priority	Desirable		
Scenarios			
Main	<u>Add preferable competence</u> : The actor accesses the wishlist menu and sees competences in that list. Then he/she clicks on the 'add competence' option and a list of competences appears. He/she chooses a set of competences to add and submits.		
	<u>Remove preferable competence</u> : The actor accesses the wishlist menu and sees competences in that list. Then he/she clicks on the 'remove competence' option and a list of competences appears. He/she chooses a set of competences to remove and submits.		

Table 17: Add/remove preferable competence use case

#### Sequence Diagram

It is presented the sequence diagram for the 'add preferable competence' scenario. For competence removal the behaviour is analogous.



Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view his/her wishlist.
Assumptions	Actor cannot add competences he already possesses not interfere with a wishlist he/she doesn't own.

## D. Training

In section 2.3 (strategic perspective of CM), it was stressed the importance of competence development; that is, a corporate effort in the improvement of the workers competence profiles, accomplished by the development of existing competences or by acquiring new ones. The *Training* module follows that premise, as it translates a corporate effort towards the capitalization of corporate knowledge. Figure 19 provides an overview of this module's use cases, with the following sub modules: *Gap analysis, Evaluation* and *Lookup*.



Figure 19: Training use cases

As noted in chapter 2, gap analysis is a crucial feature of a CMS, as it allows the detection of corporate competence breaches. Accordingly, the *Gap analysis* module provides the tools for a more accurate targeting of competence development efforts. Assuming that gap analysis is performed between competence profiles, then different profile comparisons will enhance different competence needs. For instance, a gap analysis between a project competence profile and a worker will assess if he/she possesses the competence levels of expertise required by the project. Similarly, a gap analysis between a worker's profile and a group of competences can be used to determine that worker's expertise in a specific competence set. Moreover, gap analysis can be performed among a set of workers for ranking purposes.

Competence development needs mustn't be exclusively known by an employee's supervisor. It is also important that the worker is aware of his/her own competence development needs, which is also possible with this module. Such use case is detailed in Table 18.

If a gap analysis aims to emphasize competence development needs, then, when performed on workers, it should be able to recommend training programmes that shall minimize the gap. Furthermore, the *Gap Analysis* module allows the storage of gap analysis results, thus providing the means for an historic perspective of competence development needs evolution.

Identifier	UC-TRN-GA-6		
Tuchtiner			
Name	Perform gap analysis on myself		
Description	Actor performs a gap analysis on his own competence profile.		
Actor(s)	Worker		
Priority	Essential		
Scenarios			
Main	The actor accesses the gap analysis menu and chooses 'self-diagnosis'. Then, he/she configures the gap analysis parameters and submits. The system presents the results in graphical and textual form to the user.		



Table 18: Perform gap analysis on myself use case

	Interface Prototype
jK Ho	me       My Profile       (action1>         me       My Profile       (action1>         Time Frame       Image: Compare current expertise       Image: Compare current expertise         Ault time analysis       Image: Compare current expertise       Image: Compare current expertise         Graup Name       Image: Compare current expertise       Image: Compare current expertise         Sy competence and ided       Image: Compare current expertise       Image: Compare current expertise         Graup Name       Image: Compare current expertise       Image: Compare current expertise       Image: Compare current expertise         Graup Name       Image: Compare current expertise       Image: Compare current expertise       Image: Compare current expertise         Graup Name       Image: Compare current expertise       Image: Compare current expertise       Image: Compare current expertise       Image: Compare current expertise         Graup Name       Image: Compare current expertise       Image: Compare current expertise       Image: Compare current expertise       Image: Compare current expertise         Graup Name       Image: Compare current expertise       Image: Compare current expertise       Image: Compare current expertise       Image: Compare current expertise         Graup Hartence name       Image: Compare current expertise       Image: Compare current expertise       Image: Compare current expertise
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view analysis results by text and/or graphics.

Evaluation shouldn't be confined to the worker's competence profiles. Considering training programmes, the system should provide the means to assess the effectiveness of a given programme. The *Evaluation* module follows this premise, allowing workers to appraise training programmes they've been enrolled in. Thus, a manager can identify which courses are more effective and profitable. Since competence development is a gradual process, this module also contemplates the presentation of a worker's competence expertise evolution and related training programmes.

A user may be interested in searching for available training programmes to develop his/her own competence or of his/her subordinates. In this sense, the *Lookup* module allows a user to seek adequate training programmes, whether by comparing them against projects profiles, a worker's own competence profile or even his/her own wishlist.

## E. Coaching

Still related to competence development, a worker might benefit from another user's tutoring. This tutor, called *Coach*, analyses a worker's competence profile and reports it to the *Manager*. Hence, the latter is able to construct a competence development plan more adequate to that worker's needs. Figure 20 shows the main use cases for this module.



Figure 20: Coaching use cases

## F. Talent

Towards the enrichment of corporate knowledge capital, the *Talent* module provides the tools for a user to search for the most suitable candidate for a given set of competence needs. The user can seek these resources in two ways: by looking inside CMF's internal competence pool (*Sniffing* sub module) or by conducting job interviews (*Scouting*) sub module. In the latter, the user acts as an *Examiner*, while the analysed individual is an *Examinee*. Talent *Sniffing* can be performed by a *Manager*, who searches for worker according to a competence profile (*ad hoc* or from a project). On the other hand, in the case of *Scouting*, an *Examiner* may generate job interviews from competence profiles or choose to do so manually and then execute them with an *Examinee*. Figure 21 portraits the use cases for this module. Taking a project competence profile as a basis, Table 19 describes the use case for the search of workers, while

Table 20 details the use case for the generation of a job interview.



Figure 21: *Talent* use cases

# Table 19: Search for workers for a project use case

Identifier	UC-TAL-SN-2		
Name	Search for workers for a project		
Description	Actor searches for workers to allocate to a project.		
Actor(s)	Manager		
Priority	Essential		
Scenarios			
Main	Actor accesses the 'talent' menu and chooses to search for workers. In the search engine configuration, he/she specifies that the search should be targeted for a project. Then, he/she chooses the target pro- ject, fills the search field data and submits.		
Sequence Diagram			



-

Interface Prototype							
iKnow						Logout	Î
Home My Pro	file <action1></action1>	]					
Talent Sn	iffing rence profile ————					My gadgets	
Associate to p	roject						11
Define li	ist of competences		$\frown$				11
	port competence protile		60)				11
Winner							
$\bigcirc$	<name> <other data=""></other></name>						
Name		Competence Level	#Rank	More Info	A		
Ranking Giacomo	Guilizzoni		1	±			11
Top 10 Guido J	ack Guilizzoni		2	±		erts	
Options Mariah	Maclachlan		3	±		Competency	
Marco	Botton\Tuttofare		4	<b>±</b>		Other info	
Valerie	Liberty		5	±	Ти	o Labels, and a yup	
		measures the pro	vimity to the	desired profile	🔽	Alert read (>)	
		chiedsdres me prov			•	A Bullet 📃	
					C0	nf CONFIGURE	
					ні	de OFF	
					III C	)	¥
ernative	The actor cancels t	he operation.					
-conditions	Actor must be logg	ged in.					
t-conditions	Actor views ranked	l list of workers.					

# Table 20: Generate job interviews from project competences use case

Identifier	UC-TAL-SC-1
Name	View job interview
Description	Actor views the job interview questions and properties.
Actor(s)	Manager, Examiner
Priority	Important
Scenarios	
Main	Actor accesses the 'talent scouting' menu and chooses to view the existing job interviews. The system presents a list of the job interviews that the actor has access, he/she selects a list item and clicks on the option 'view details'. Then, the job interview outline and proper- ties are shown.



### G. Dashboard

Towards a Rich Internet Application, the system ought to provide a highly configurable desktop-like user interface, where the user can manage multiple gadgets. Hence, the *Dashboard* module contemplates use cases related to the configuration of the interface's backbone – the dashboard. Figure 22 illustrates the main use cases for this module, while Table 21 details the use case of a worker who adds/removes a gadget from the dashboard.



Figure 22: *Dashboard* use cases

Table 21: Add g	gadget to	dashboard	use case
-----------------	-----------	-----------	----------

Identifier	UC-DSB-2		
Name	Add/remove gadget to/from dashboard		
Description	Actor adds a new gadget to the dashboard.		
Actor(s)	Worker		
Priority	Essential		
Scenarios			
Main	Actor accesses the configuration menu of the dashboard and views the currently present gadgets in the dashboard. If the user selects a gadget from the list and clicks 'remove', that gadget will no longer appear in the dashboard. On the other hand, the user can add a new gadget by clicking on the 'add gadget' option and selecting the gadget to add.		
Alternative	The actor cancels the operation.		
Pre-conditions	Actor must be logged in.		
Post-conditions	Actor views updated dashboard.		

## H. Gadgets

The worker's dashboard is populated by gadgets. In this sense, this module contains the sub modules for each type of provided feature. For instance, a gadget might be responsible for giving warning messages to the user. Such is the case of the *Alerts* module. The reader may find irrelevant to specify only one sub module. However, that option was made as the project contemplates the addition of more gadgets in future developments. Figure 24 details the use cases for the *Alerts* module, which is pictured in Figure 23 as part of the *Gadgets* module.



Figure 23: Gadgets use cases



Figure 24: Alerts use cases

Table 22:	Configure	alerts	use case
-----------	-----------	--------	----------

Identifier	UC-GAD-AL-1		
Name	Configure alerts		
Description	Actor configures gadget alerts		
Actor(s)	Worker		
Priority	Essential		
Scenarios			
Main	Actor clicks on the 'configuration' menu of the alert gadget and the main setup options are displayed.		
Alternative	The actor cancels the operation.		
Pre-conditions	Actor must be logged in.		
#### **Requirements Analysis**

#### I. Social Network

Although Competence Management is primarily a process linked to supervision activities, it would be interesting if the organization's workers were aware of their colleagues' skills. Hence, the *Social Network* module allows a worker to become part of a competence network with a social component, similar to LinkedIn [70], for example. The main use cases of this module are pictured in Figure 25, where a *Worker* may build a friends network, exchange messages with other workers and edit his/her public competence profile. Table 23 describes an example use case for this module.



Figure 25: Social network use cases

#### **Requirements Analysis**

Identifier	UC-SNT-2				
Name	Accept/decline/ignore friend request				
Description	Actor accepts/declines/ignores an invite to join another worker's friend network.				
Actor(s)	Worker				
Priority	Essential				
Scenarios					
Main	Actor is viewing a list of his/her pending friend requests. Then, he/she selects a request and clicks on the option 'accept', 'decline' or 'ignore'.				
Alternative	The actor cancels the operation.				
Pre-conditions	Actor must be logged in.				

#### Table 23: Accept/decline/ignore friend request use case

#### J. Security/Privacy

Regarding user accounts and permissions, the *Security/Privacy* module considers all corresponding activities, such as user login/logout, account registration, and account CRUD (Create, Read, Update and Delete) operations, reserved for the system *Administration*. The main use cases of this module are shown in Figure 26.



Figure 26: Security/Privacy use cases

#### **Requirements Analysis**

A relevant use case is the delegation of roles. A *Manager, Examiner, Facilitator* or *Coach* may replicate his permissions to another user. For a *Manager*, this can be useful for time management purposes and to capitalize his/her subordinates competences. An *Examiner* might delegate his/her role so that another user carries out his/her job interview. A *Facilitator* can pass on his/her role to another user when, for example, he/she has a competence appraisal effort overload. An analogous situation might happen with a Coach. Such use case is detailed in Table 24.

Identifier	UC-SEC-3				
Name	Delegate own role on another user				
Description	Actor replicates to another actor his/her own privileges.				
Actor(s)	Manager, Examiner, Facilitator, Coach				
Priority	Essential				
	Scenarios				
Main	The user goes to the 'security' menu and chooses the 'delegate role' option. Then, a list of subordinate workers is shown. The actor selects a worker and clicks on the option 'delegate role'.				
Alternative	The actor cancels the operation.				
Pre-conditions	Actor must be logged in.				

#### Table 24: Delegate own role on another user use case

## 5.6 Supplementary Requirements

Aside from the requirements specified so far, non-functional system requirements must be considered. These are not directly concerned with the specific functions delivered by the system. Instead, they specify or constrain its emergent properties [76]. Table 25 details such requirements, prioritizing them.

Identifier	Name	Priority
SR-1	Portability	Essential
SR-2	Usability	Essential
SR-3	Efficiency	Essential
SR-4	Reliability	Essential
SR-5	Security	Essential
SR-6	Robustness	Essential
SR-7	Extensibility	Essential
SR-8	Maintenance	Important
SR-9	Compatibility	Important
SR-10	Internationalization	Important

Table 25: iKnow's supplementary requirements list

## 5.7 Summary

Critical to the success of a development project [77], this chapter focused on an extensive definition of the envisioned system's requirements. Hence, user characteristics, restrictions, assumptions and dependencies as well as external interface requirements were specified. Moreover, the main functional requirements – including user stories and use case models – were presented. Finally, the supplementary requirements were prioritized.

Eight main modules were identified, covering all key phases of Competence Management, from competence mapping to monitoring: *Core competence management, Training, Coaching, Talent, Dashboard, Gadgets, Social Network* and *Security/Privacy.* 

With the requirements analysis complete, the next step is to design the system, specifying its architecture and adopted technologies.

## **Chapter 6**

# **Solution Specification**

We have observed two traits common to virtually all of the successful object-oriented systems we have encountered, and noticeably absent from the ones that we count as failures: the existence of a strong architectural vision and the application of a well-managed iterative and incremental development cycle.

Grady Booch [78]

This chapter contemplates the solution specification, thus covering system architecture, technology platform, data model and main design decisions.

### 6.1 Architectural Requirements

Prior to a detailed definition of the system high level design, its architecturally significant requirements ought to be described.

Regarding *organizational requirements*, the system needs to have a clear *separation of concerns*. In particular, the following responsibilities should be detached [79]:

- *Presentation*, encompassing processes and components exclusively related to the visual display needs of the application, namely the user interfaces;
- *Business Logic*, concerning processes and components solely related to the application domain, namely the definition of the object model, system control workflow and business processes. The application's core business concerns should be encapsulated, exclusive of how data and behaviour is exposed, or how data is specifically obtained;
- Data Access, regarding processes and components utterly related to the access to external system data. This includes components which interface with a local data store or remote service. Providing an abstraction layer to the details specific to data access, this responsibility includes tasks such as the establishment of database and service connections, maintaining knowledge about database schemas or stored procedures,

knowledge about service protocols and the marshalling of data between service entities and business entities;

- *Data*, encompassing processes and components absolutely related to data storage and manipulation.

By grouping processing concerns based on their role within the application, a number of benefits are expected which improve the overall system manageability, such as ease of maintenance through consistent architecture and isolation of process, increased insulation from change impact, increased adaptability to change and increased potential for reuse.

Concerning *functional requirements*, iKnow should be able to interact with external Critical systems, providing accessibility to functional building blocks over standard Internet protocols, independent of platforms and programming languages. Hence, interoperability should be promoted.

## 6.2 Main Design Decisions

At the beginning of the design engineering phase, Critical Manufacturing (CMF) set out design decisions, namely concerning architectural patterns: *Model-View-Controller* and *Dependency Injection*. This section briefly clarifies these patterns and their benefits to the project.

#### 6.2.1 Model-View-Controller Pattern

The purpose of many computer systems is to retrieve data from a repository and display it for the user. After he/she changes data, the system submits the updates to the data store. Because the key flow of information is between the data store and the user interface, you might be inclined to tie these two pieces together to reduce the amount of coding and to improve application performance [80]. However, this seemingly natural approach has major drawbacks, such as a domino effect delay and the coupling of different domains. For instance, the user interface tends to change much more frequently than the data storage system, so joining them would make the latter dependant on the user interface needs. The second issue can be found when data and user interface pieces are coupled: business applications tend to incorporate logic that goes far beyond data transmission, so you can't really join these two action domains. Thus, there is a need to modularize the user interface functionality as to easily modify the individual parts.

*Model-View-Controller* (MVC) is an architectural pattern that separates the modelling of the domain, presentation and the actions based on user input. The *Model* manages the behaviour and data of the application domain, responding to requests for information data about its state (usually from the view) and responds to instructions to change state (usually from the *Controller*). On the other hand, the *View* renders the model, thus managing information display, while the *Controller* interprets mouse and keyboard inputs from the user, informing the *Model* and/or the *View* to change as appropriate. Figure 27 illustrates the MVC concept.



Figure 27: MVC concept, where direct and indirect associations are represented by a solid and a dashed lines, respectively [80]

The key benefit of this pattern is that both the *View* and the *Controller* depend on the *Model*, though the inverse is not verified. This separation of responsibilities allows the *Model* to be built and tested independent of the visual presentation. In essence, this pattern is a fundamental design standard for the separation of user interface from business logics – ideal to meet the previously mentioned *separation of concerns* architectural requirement. In the target system, a rich internet application is envisioned with a user interface that is highly volatile - it changes very quickly, according to the different users' needs. For example, a gap analysis shall be presented differently according to each target worker.

#### 6.2.2 Dependency Injection Pattern

Today there is a greater focus than ever on reusing existing components and wiring together disparate components to form a cohesive architecture. But this wiring can quickly become a daunting task because as application size and complexity increase, so do dependencies. One way to mitigate the proliferation of dependencies is by using *Dependency Injection* (DI), which allows you to inject objects into a class, rather than relying on the class to create the object itself [81].

One of the most important concepts in computer programming refers to the process of supplying an external dependency to a software component [82], thus avoiding a chain of dependencies in object graphs. DI is a design pattern that offers a unique and concise solution to the problem of constructing object graphs, with a strong emphasis on unit testing. Indeed, all your code is freed from constructing dependencies [83].

In DI, there are three elements: a dependent, its dependencies and an injector (also known as provider or container). The dependent is a consumer that needs to accomplish a task in a computer program. In order to do so, it needs the help of various services (called dependencies) that execute certain sub-tasks [84]. The provider is the component that is able to compose the dependent and its dependencies so that they are ready to be used, while also managing these objects' life-cycles. By using DI, the life-cycle of a service is handled by a dependency provider which is in normal case a web container rather than the consumer. The consumer would thus only need a reference to an implementation of the service that it needed in order to accomplish the necessary task. Figure 28 explains how DI works.



Figure 28: Explanation of DI's main principles [85]

The main benefit of DI is dependency decoupling, thus meeting the *separation of concerns* architectural requirement, also providing a more reusable and testable code [86].

## 6.3 Technology Platform

Technology wise, the following questions arise:

- How to deploy the solution?
- Which tools should be used and why?

Hence, this difficulty needs to be broken down by levels of responsibility as to correctly assign the technologies to be deployed.

Regarding architectural *functional requirements*, CMF chose to deploy Web Services, preferably using REST (Representational State Transfer), which is described in this section. Such decision affected the survey on web-based *Model-View-Controller* frameworks, which should ideally support Web Services.

#### 6.3.1 Preliminary Survey

Broadly speaking, there are two streams of concern:

- Web-based application: *How should it be deployed? Which framework should be used?*
- Database management: Which Database Management System is more adequate?

In this sense, a study was carried out to compare viable technologies for the project's deployment. These two streams were assumed as the review basis. As an attempt to narrow down the subset of viable technologies, the following assumptions were taken:

- Development tools must be (preferably) Open Source;
- Web-based framework implementations must follow the *Model-View-Controller* architectural pattern;
- Web services must to be supported.

#### A. Web-based MVC Frameworks

The set of analysed frameworks is strongly based on [87]. The subsequent parameters were taken into account, by programming language:

- Model-View-Controller pattern: is it present in the framework's core or as a component?
- Maturity;
- Latest (stable) release;
- Programming language version;
- (Expected) learning curve;
- Open source/free IDE (Integrated Development Environment);
- Web services support;
- Scalability;
- Extensibility;
- Support community;
- If the framework is or not cross-platform;
- Development status (active/inactive).

Tables 26-28 clarify how parameters *Maturity*, *Support Community* and *(Expected) Learning Curve* were measured.

### Table 26: Clarification of the Maturity parameter

Parameter name	Maturity		
<b>Description</b> Measures the technologies level of maturity. It depends directly on the last stable release number.			
Levels			
Low	Last stable release between 0.0 and 1.0 (inclusive)		
Medium	Last stable release between 1.0 and 2.5 (inclusive)		
High	Last stable release between 2.5 and 3.5 (inclusive)		
Very High	Last stable release higher than 3.5		

## Table 27: Details of the Support Community parameter

Parameter name	Support Community			
Description	<ul> <li>Dynamic knowledge base where a user can collaborate, create and quickly access the latest technical support content. This is measured according to: <ul> <li>Framework homepage support (wiki, forums, tutorials, demos)</li> <li>Google results when framework name is used as search keyword</li> </ul> </li> </ul>			
Levels				
Reduced	Very little documentation. Inexistent or poorly active forums.			
Medium	Active forums but scarce results in Google search.			
Large	Strong homepage support and extensive Google results.			
Very Large	Large support community and books dedicated to the framework analysis.			

### Table 28: Explanation of the (Expected) Learning Curve parameter

Parameter name	(Expected) Learning Curve			
<b>Description</b> Related to the developer (Gabriela Soares, author of this Dissertation) skills framework's programming language.				
	Levels			
Low	The developer has never worked with that language before.			
Medium	The developer has worked with that language but only at a basic level.			
High	The developer knows the language and has done more than one development pro- ject using it.			

A comparative analysis was performed regarding the following technologies: ActionScript, ASP, C++, ColdFusion, Flex, Groovy, Java, JavaScript, ASP .NET, Perl, PHP, Python and Ruby. A detailed study for each of these languages can be found in Annex B.

In this section, the final comparative matrix is presented in Table 29<sup>3</sup>, where the following restrictions apply:

- Support community must be above medium (inclusive);
- Development status must be active;
- Must have web services support.

Moreover, the final list was ranked using Excel's lexicographic sorting, according to the subsequent priorities:

P1 (most important): *Extensibility* 

P2: Scalability

P3: Support Community

P4: Web Services Support

P5 (least important): (Expected) Learning Curve

<sup>&</sup>lt;sup>3</sup> In the comparative matrix presented, the reader will find references to RESTful web services in the column 'Web Services'. A RESTful web service is a simple web service implemented using HTTP and the principles of REST (Representational State Transfer), which means that each unique URL is a representation of some object. This technology is explained in the 'Final Decisions' subsection.

### Table 29: Final comparison matrix of web-based MVC frameworks

MVC Web-Based Frameworks												
#Rank	Name	Language	MVC	Maturity	(Expected) Learning Curve	Support Community	Development Status	Scalability	Extensibility	Web Services	Cross- Platform?	Notes
1	Catalyst	Perl	Core	Very high	Medium	Very large	Active	High	High	Support for REST- ful services	Yes	"the most useful and easy to use MVC Framework"
2	CakePHP	PHP	Core	Medium	Low	Large	Active	High	High	Support for REST- ful services	Yes	
3	PureMVC	C#, Coldfusion, Java, JavaScript, PHP, Ruby, Python, Ac- tionScript	Core	Medium	Medium	Large	Active	High	High	Not explicit in the framework. Web Services through its programming lan- guage features	Yes	
4	CodeIgniter	PHP	Core	Medium	Low	Large	Active	High	High	Yes	Yes	
5	Apache Tapestry	Java	Core	Very high	Low	Large	Active	High	High		Yes	Object-oriented
6	Django	Python	Core	Medium	Low	Medium	Active	High	High	Support for REST- ful services	Yes	
7	TurboGears	Python	Core	High	Low	Medium	Active	High	High	Yes	Yes	
8	JSF (JavaServer Faces)	Java	Core	High	Low	Medium	Active	Medium	High	Supported by Java	Yes	AJAX-enabled
9	Ruby on Rails	Ruby	Core	High	Medium	Very large	Active		High	Support for REST- ful services	Yes	
10	FLOW3	PHP	Core	Low	Low	Large	Active			Support for REST- ful services	Yes	Recommended for enterprise applica- tions
11	Zend	PHP	Component	Medium	Low	Large	Active			Support for REST- ful services	Yes	Component frame- work
12	Symfony	PHP	Core	Medium	Low	Medium	Active			SOAP web services	Yes	Component frame- work

#### **B.** Database Management Tools

Analogous to the previous section, this subdivision focuses on the review of adequate Database Management Systems (DBMSs) for iKnow. In particular, the following parameters are considered:

- Database type (filesystem-based, document-oriented, hierarchical, network, object-oriented or relational);
- Open source tools;
- Maturity (according to Table 26);
- Associated programming languages;
- Relevant features, namely: enterprise directory compatibility, transactions and referential integrity;
- Latest stable release;
- Maximum database size.

The analysis was performed on a wide set of DBMSs by database type and strongly based on [88], [89] and [90]. Using Excel's lexicographic sorting, Table 30 presents the final comparative matrix, ranked by Maturity levels and restricted to document-oriented, relational and object-oriented database types. Not all previously mentioned parameters are included in the provided matrix: should the reader demand further analysis detail. He/she may see Annex B.

	Database Management Systems									
1st #Rank	2nd #Rank	Name	Туре	Maturity	Cross-Platform?	Notes				
	1.1	Mnesia	-	High		Distributed, soft real-time DBMS.				
	1.2	MongoDB		Medium	Yes	High-performance, open source, schema-free. Supports dynamic queries, C++, Java, JavaScript, Perl, PHP, Python and Ruby supported (drivers).				
1	1.3	Cassandra	Document-	Low		Initially developed by Facebook for storing very large amounts of data				
	1.4	HyperTable	oriented	Low		Inspired on Google's BigTable; Initially designed for large-scale data-intensive tasks.				
	1.5	CouchDB		Low		Designed of local replication and to scale vertically along a wide range of devices.				
2	2.1	Apache Derby		Very high	Yes					
	2.2	MySQL		Very high	Yes					
	2.3	OpenLink Virtuoso	- Relational	Very high	Yes					
	2.4	PostgreSQL		Very high	Yes	Is considered object-relational DBMS				
	2.5	PostgreSQL Plus Stan- dard Server		Very high	Yes					
	2.6	SQLite		High	Yes					
	2.7	Firebird	-	Medium	Yes					
_	2.8	HSQLDB	-			Medium	Yes			
	3.1	OpenLink Virtuoso	Object- oriented	Very high						
3	3.2	Databeans		Medium	Fully object oriented persistence framework for java. Distributed, transactional, with GC, persistent Collections, XML export/import, security					
	3.3	Eloquera						Medium	SQL for objects and arrays of any complexity, client/server, web oriented	

## Table 30: Final DBMS comparison matrix

### 6.3.2 Final Decisions

This section presents the adopted technologies. According to the preliminary study results, Catalyst or CakePHP would be the best suitable solutions as Web-based MVC frameworks. However, the project team chose to use ASP .NET MVC (please refer to Annex B): since the majority of Critical Manufacturing's engineers had experience in ASP .NET, the company would benefit from a bidirectional knowledge transfer. That is, the engineering team would contribute to the development of the author's competences in ASP .NET and the latter, in turn, would transmit to the team her competences in the application of MVC in ASP .NET at the completion of this project. Regarding DBMSs, PostgreSQL was the chosen technology for being a widespread open source object-relational DBMS and due to the project's team wide experience with that tool.

This section describes not only the mentioned chosen technologies, but also complementary tools for the project's deployment.

#### A. ASP .NET MVC

ASP.NET MVC is a free and fully supported Microsoft framework for building web applications that use a model-view-controller pattern. Like ASP.NET Web Forms, ASP.NET MVC is built on the ASP.NET framework, providing the following key benefits [91]:

- Complete control over HTML markup;
- Rich AJAX integration;
- Intuitive website URLs;
- Clear *separation of concerns* which results in web applications that are easier to maintain and extend over time;
- Testability, including support for test-driven development.

Based on ASP.NET, this technology allows software developers to build a Web application as a composition of three roles: *Model*, *View* and *Controller*. This framework will be the backbone of the system's web rich application, deploying the *Model-View-Controller* pattern.

#### B. Spring .NET

Spring .NET (Figure 29) is an open source framework that provides comprehensive infrastructural support for developing enterprise .NET applications. It allows you to remove incidental complexity when using the base class libraries, supporting test driven development practices.



Figure 29: Spring .NET framework logo [92]

Spring.NET is created, supported and sustained by SpringSource [92]. Its design is based on the Java version of the Spring Framework, which has shown real-world benefits and is used in thousands of enterprise applications worldwide. At the core of Spring .NET is the concept of *Dependency Injection* (DI), studied in the previous section of this chapter. This technology shall be used throughout the system to deploy the DI design pattern.

#### C. jQuery

jQuery (Figure 30) is a free, open source lightweight cross-browser JavaScript library that emphasizes the interaction between JavaScript and HTML. Its syntax is designed to make it easier to navigate a document, select DOM elements, create animations, handle events, and develop Ajax applications [93]. jQuery also provides capabilities for developers to create plugins on top of the JavaScript library. Hence, developers are able to create abstractions for low-level interaction and animation, advanced effects and high-level, theme-able widgets, contributing to the creation of powerful and dynamic web pages [94]. This technology shall be used in the GUI deployment, relative *Presentation* concerns.



Figure 30: jQuery logo [94]

#### D. Telerik Extensions for ASP .NET MVC

Telerik Extensions for ASP.NET MVC (Figure 31) extend the ASP.NET MVC platform by delivering reusable user interface controls built specifically for MVC. A server-based framework that integrates with client-side modules built on the popular JavaScript library, jQuery delivers productivity by enhancing controls for MVC that do not violate the patterns, practices, and guidance of the platform. The Telerik Extensions restore developer productivity to ASP.NET MVC and make it possible to build clean rendering web applications without writing all code by hand [95]. This technology shall be applied in user interface design, concerning *Presentation* responsibilities.



Figure 31: Telerik Extensions for ASP .NET MVC logo [95]

#### E. PostgreSQL and Npgsql

PostgreSQL (Figure 32) is a powerful, cross-platform, open source object-relational Database Management System (DBMS) with more than 15 years of active development and a proven architecture that has earned it a strong reputation for reliability, data integrity, and correctness [96].

With rich standards compliance, this enterprise class DBMS boasts sophisticated features such as Multi-Version Concurrency Control, point in time recovery, tablespaces, asynchronous replication, nested transactions, online/hot backups, a sophisticated query planner/optimizer, and write ahead logging for fault tolerance [97]. It supports international character sets, multibyte character encodings, Unicode, and it is locale-aware for sorting, case-sensitivity, and formatting. It is highly scalable both in the sheer quantity of data it can manage and in the number of concurrent users it can accommodate [97]. This technology shall be used concerning *Data* responsibilities.



Figure 32 – PostgreSQL logo [97]

Npgsql is a .NET data provider for PostgreSQL. It allows any program developed for .NET framework to access database server [98]. Concerning *Data Access* responsibilities, the system will use this technology to connect the ASP .NET MVC application to the data repository in PostgreSQL.

#### F. RESTful Web Services

Representational State Transfer (REST) has gained widespread acceptance across the Web as a simpler alternative to SOAP - and Web Services Description Language (WSDL) - based Web services. Key evidence of this shift in interface design is the adoption of REST by mainstream Web 2.0 service providers—including Yahoo, Google, and Facebook—who have deprecated or passed on SOAP and WSDL-based interfaces in favour of an easier-to-use, resource-oriented model to expose their services [99].

REST defines a set of architectural principles by which you can design web services that focus on a system's resources, including how resource states are addressed and transferred over HTTP by a wide range of clients written in different languages. In its purest form, a concrete implementation of a REST web service (called RESTful web service) follows four basic design principles [99]:

- Explicit use HTTP methods;
- Statelessness;
- Exposure of directory structure-like URIs.
- Transfer of XML, JavaScript Object Notation (JSON), or both.

REST is different from SOAP. SOAP is a complex protocol for interoperating, offering many layers of security and versioning. Usually, SOAP runs over HTTP, but in Windows

Communication Foundation (WCF), it can run over TCP or other connections too. SOAP bypasses a lot of HTTP semantics and implements them again as its own. REST, on the other hand, is a lightweight, simpler protocol that follows the HTTP way of working and all of its semantics: a header value of GET to retrieve something, PUT or POST to save something, and DELETE to delete a resource. SOAP is more versatile, but very complex and slow. REST is a lot faster, and offers a lot less, and is limited to HTTP [100].

Using WCF, RESTful web services shall be deployed in iKnow, thus promoting interoperability between external applications and the envisioned system and meeting the *functional architectural requirements* previously identified.

#### G. Windows Communication Foundation (WCF)

The Windows Communication Foundation (Figure 33) is an application programming interface in the .NET framework for building connected, service-oriented applications.

WCF is designed in accordance with service oriented architecture principles to support distributed computing where services are consumed. Clients can consume multiple services and services can be consumed by multiple clients [101]. It provides a unified framework for building secure and reliable transacted Web services. WCF combines and extends the capabilities of distributed systems, Microsoft .NET Remoting, web services, and Web Services Enhancements (WSE), to develop and deliver unified secured systems. Hence, this framework builds loosely-coupled applications on a service oriented architecture that interoperates more securely and reliably across platforms.

WCF simplifies the development effort to make service oriented applications by combining the technologies together, leading to higher development productivity. Additionally, it reduces the complexity of applications by unifying Enterprise Services, Messaging, .NET Remoting, Web Services, and WSE. WCF builds applications with an attributed programming model, leading to higher developer productivity. It supports a broad range of web service standards like XML, XSD, SOAP, XPath, WSDL, and advanced standards and specifications like WS-Addressing, WS-Policy, WS-Security, WS-Trust, WS-Secure Conversation, WS-Reliable Messaging, WS-Atomic Transaction, WS-Coordination, and WS-Policy [98].

This framework is useful for the envisioned system, thus providing a resource oriented architecture, required for the deployment of RESTful web services [102].



Figure 33: Overview of the WCF framework [101]

## 6.4 High Level Design

Large systems are usually decomposed into sub-systems that provide some related set of services. The initial design process of identifying these sub-systems and establishing a framework for sub-system control and communication is called architectural design [76].

Referring Bass et al. [103], there are three clear advantages of explicitly designing and documenting software architecture:

- *Stakeholder communication*: as a high-level representation of the system, it may be used as an asset for discussion by a wide range of stakeholders;
- *System analysis*: explicitly documenting a system's architecture enforces architectural design decisions that have a profound effect on whether it can meet critical requirements such as performance, reliability and maintainability [76];
- *Large-scale reuse*: as a compact description of a system's organization and internal components interaction, its architecture if often the same for others with similar requirements. Thus, major software reuse can be supported [76].

Hence, this section is dedicated to the presentation of the envisioned system architecture, accomplished by presenting the logic and physical architectures.

### 6.4.1 System Components

The following main components are identifiable:

- Application server, where the web-based application resides;
- Database management system, where storage data is kept.

### 6.4.2 Physical Architecture

A system's physical architecture defines the location of the system's components and required configurations [76]. In the case of a web-based application, the physical architecture follows a client-server model, where at least one server receives from and sends data to the clients. This section presents an overview of iKnow's high-level physical structure.

As pictured in Figure 34, there is a web server where the web application and the web services reside, with a local data repository. There is a variable set of clients who access the application via Internet and, in its turn, the web server communicates with outside repositories to fetch relevant data as needed using RESTful web services.



Figure 34: iKnow's physical architecture

iKnow is composed by three main elements:

- Client, composed by a web browser that, via HyperText Transfer Protocol (HTTP) accesses the application server. Using this request/response standard (typical in client-server computing), the client is an application (web browser) on the computer held by an end-user. Accordingly, the server is an application running on the computer hosting the web site Application Server. The client submits HTTP requests and the responding server stores or creates resources (such as HTML files and images);
- Application Server, which contains the components for the web server and the database. In particular, there is a Web Server component, responsible for answering the HTTP requests, thus allowing web access to the system. This component is implemented in ASP .NET with the *Model-View-Controller* architectural pattern. In this sense, Figure 34 presents the components 'Web Server with ASP .NET MVC Support' and 'PostgreSQL Database'. In its turn, the PostgreSQL database stores relevant system data in the application server;
- Nodes external to iKnow, but within Critical domain. These comprise applications and data repositories that belong to the Critical organization, named as the following:
  - Critical Internal Application Repository;
  - Critical Internal Data Repository.

Moreover, the application, resident in the application server, is accessible in a dual fashion:

- Client, through the end user's web browser;
- By external applications, via RESTful web services (promoting interoperability).

### 6.4.3 Logical Architecture

A software system's logical architecture defines a collection of components and their relationships compliant with all system's requirements [104]. Unlike the physical architecture, the logical architecture is concerned with the conceptualization of the system's structure and processes. Thus, it is relevant to specify the system's logical components.

A system's logical architecture defines the processes (activities and functions) that are required to provide the required user services. From such perspective, a system can be viewed from two complementary angles:

- *Horizontally*: by implementation layers and relatively independent of the system's features;
- *Vertically*: by subsystems in a hierarchical fashion, where each subsystem matches a group of features and spans through all implementation layers.

#### A. Horizontal Partitioning

Please recall the distinct responsibilities (*Presentation, Business Logic, Data Access* and *Data*) discussed in the initial section of this chapter (6.1), related to organizational architectural requirements (*separation of concerns*). Taking that approach, the following application layers can be listed, with the previously described responsibilities:

- User Interface (corresponding to Presentation);
- Business Logic;
- Data Access;
- Data.

With this multitier architecture, the presentation, application processing and data management are logically separate processes. This topology allows the independent upgrade or replacement of any of the tiers, as requirements or technology change [105]. For example, a change of operating system in the *User Interface* tier would only affect the related code.

The User Interface layer allows the user access to the system's features. The information given to this layer shall be sent/received by the subjacent layer (*Business Logic*). The lower layers are responsible for the validation and processing of requests, while *User Interface* exclusively deals with data delivery to the user, namely concerning an accessible and usable presentation of contents. ASP .NET MVC's views, jQuery and Telerik Extensions for MVC are used in this layer.

On the other hand, the *Business Logic* layer is responsible for the system's functionalities. It gives meaning to the graphical interface components since it allows the corresponding actions by performing detailed processing. The models from ASP .NET MVC support the business

logic at this tier, whereas the controllers act as contracts between the views and the models and handle user actions [106].

Concerning the *Data Access* layer, it operates as a link between the *Business Logic* and *Data* layers. It possesses all the features required for repository information access.

Finally, the *Data* layer embodies the physical site of all vital system data. All data management is handled by the *Data Access* layer, where information is stored and is retrieved from. This tier keeps data neutral and independent from application servers or business logic, which improves scalability and performance [105].

#### **B.** Vertical Partitioning

This subsection describes a set of subsystems responsible for features that are transversal to the system's implementation layers, depicted in the previous subdivision: *User Interface, Business Logic, Data Access* and *Data*. This architecture identifies the system components (and their dependencies) that provide the software services needed to meet the business goals of the project deployment.

Please note the presence of web services in the proposed architecture - they are a software system designed to support interoperable machine-to-machine interaction over a network [107].

As depicted in Figure 35, there are eight subsystems corresponding to the system modules identified in the Requirements Analysis (5.5.2) chapter:

- Core Competence Management;
- Training;
- Dashboard;
- Gadgets;
- Social Network;
- Coaching;
- Talent;
- Security/Privacy.



Figure 35: iKnow's vertical logical architecture

All eight subsystems are present in the first two layers: *User Interface* and *Business Logic*. In the later, we have a *Services Interface* component, which is responsible for providing an abstract boundary for the services exposure. Hence, it defines the operations provided by the services, the messages required to interact with each operation and the patterns by which these messages interact [107]. It is described by a service contract, which specifies the service behaviour and the messages required as the basis for interaction. Additionally, it contains service adapters, which implement the service contract and expose its functionality to endpoints and implements the translation between data outside and within the service [108].

In its turn, the *Data Access* layer is responsible for accessing data stored in persistent storage. Hence, *Data Access Logic* is used for data retrieval from the database and to save data back to the database. Additionally, it also contains any business logic needed to achieve data-related operations. On the other hand, *Service Agents* isolate the idiosyncrasies of calling diverse services from the application and can provide additional services such as basic mapping between the format of the data exposed by the service and the format required by the application [108].

The *Data* tier embodies database server(s) for information storage and retrieval. Additionally, there is a *Services* component, which may provide CRUD and custom system operations.

## 6.5 Data Model

To support all the specification work carried out so far, a data model was developed. Due to the extension of this work (approximately 40 tables), the whole data model won't be described in detail. Hence, the following tables are relevant:

- *Competence*, used to store competence properties. In particular, this table is also used to build competence hierarchies, by having a *Parent\_ID* attribute. This table approaches the *Competence Modelling* issue (mentioned in section 2.4.1), based on HR-XML's competency definition [33]:

(Competence is) a specific, identifiable, definable, and measurable knowledge, skill, ability and/or other deployment-related characteristic (e.g. attitude, behaviour, physical ability) which a human resource may possess and which is necessary for, or material to, the performance of an activity within a specific business context.

This definition is extended by including the competence in a context, hierarchy and into groups.

- *Group\_CPermissions*, used to store information about competence groups. Groups are associated to users by *Group\_User* and to competences by *Competence\_Group*;
- *Job\_Interview*, used to store properties for job interviews. In particular, questions are associated to a job interview using *Job\_Interview\_Question*. Similarly, job interviews can be associated to projects, through *Project\_Job\_Interview*;
- *Project\_Competence, Training\_Competence* and *User\_Competence* are used to deploy competence profiles (approached in section 2.4.2), by associating competences to projects, training programmes of system users;
- *Training\_Evaluation* and *Competence\_Evaluation* support the association of evaluation processes to trainings and workers competence profiles, defined using *Evaluation*.

## 6.6 Summary

Sequential to the requirements analysis, this section focused on the solution specification. In particular, the technology platform was presented along with a high level design proposal and the data model, including both physical and logical architecture.

With a clear *separation of* concerns (*Presentation*, *Business Logic*, *Data Access* and *Data*), the envisioned system promotes application interoperability following the guidelines of a Service-Oriented Architecture (SOA).

At this stage, a prototype deployment and validation is the next step.

## **Chapter 7**

# **Prototype Deployment**

As a proof-of-concept for the study presented so far, a prototype was implemented. This section is devoted to the presentation of such results.

The development was carried out in a Windows XP running machine, using Microsoft Visual Studio Team System 2008 as Integrated Development Environment.

## 7.1 Approached Features

Due to the considerable extension and complexity of the envisioned CMS, the prototype deployment was oriented towards core features to assemble a solid base for future developments. Hence, the use cases implemented correspond to those included in the *Core Competence Management* and *Security/Privacy* modules. Table 31 details the deployed use cases, by module.

Module	Sub module	Use Case(s)
		Auto-evaluate a competence
	Competence evaluation	Evaluate worker's competence
		Rate a competence
Core competence	Competence derivation	Edit derived competence
management	Competence derivation	View derived competence
		Create/delete competence
	Basic management	Enable/disable competence
		View competence

Table 31: Implemented use cases

		Create/delete competence hierarchy
	Hierarchy	Edit competence hierarchy
		View competence hierarchy
	Competence profile	Add/remove competence
	Competence prome	View worker's competence profile
		Add/remove competence
	Competence grouping	Create/delete competence group/subgroup
		Edit competence group/subgroup
		Login
		Logout
Security/Privacy		Register account
		Manage user accounts

From a strategic perspective, the implementation of the *Core competence management* module covers key phases in this process. Considering Vernhout's approach [1], the *Competence management* phase is covered through the *Basic management* and *Competence profile* modules, while *Performance appraisal* is approached by the *Competence Evaluation* module. The approached features are highlighted in Figure 36.



Figure 36: Vernhout's strategic perspective on Competence Management vs. the deployed prototype features

Similarly, a correspondence can be found to the four macro-phases presented by Draganidis et al. [24]. Indeed, *Basic management* and *Competence profile* refer to the *Competence mapping* stage, while *Competence evaluation* is part of the *Competence monitoring* phase. The approached features are highlighted in Figure 37.



Figure 37: The CM process described by Draganidis et al. vs. the deployed prototype features

## 7.2 Implementation Details

This section is dedicated to the discussion of deployment minutiae. Hence, all relevant code is present in Annex C. The project adopts a modular organization, with a high *separation of concerns*. Hence, the project is structured as follows:

- *iKnow.Auth*, a class library concerning system authentication features;
- *iKnow.Database*, a class library for database management operations;
- *iKnow.Contract*, a class library with interfaces to *iKnow.Database*;
- *iKnow.GUI*, an ASP .NET MVC project for the construction of the system's GUI.

Concerning the *Security/Privacy* module, authentication plays a key role. Since the system is expected to become part of CMF's application's portfolio, user authentication data ought to be unique and global to all systems. Hence, *iKnow.Auth* deploys authentication against CMF's LDAP servers. In the prototype, that task is accomplished via Active Directory, using a DirectoryEntry per authentication attempt.

*iKnow.Database* (Figure 38) handles database access and manipulation. It provides services for core data operations (SELECT, UPDATE, DELETE), using a resources file for template queries storage.





Moreover, *iKnow.Database* portraits the system's data model, where each table is translated to its correspondent class (Figure 39).



Figure 39: Classes corresponding to the system's data model in *iKnow.Database* 

Each class follows a structure analogous to the subsequent example, where each table attribute is declared and a constructor and get methods are deployed.

Spring .NET provides a generic factory called IDbProvider for creating ADO .NET API artifacts used to connect to data providers. In particular, the class DbProviderFactory creates IDbProvider instances given a provider name. Hence, the Npgsql .NET data provider is used to connect to the PostgreSQL database.

The queries specified in the resources file (ResourceDB.resx, in Figure 40) are invoked in database manipulation methods, using the String.Format method to insert query parameters. In particular, we use the abstraction for data access via ADO .NET provided by the Spring .NET framework. Based on a *Dependency Injection* design, Spring's AdoTemplate provides a QueryWithRowCallback method used to execute the query, by calling an implementation of IRowCallback (which, in its turn, hands a IDataReader to process the current row) for each row in the result set. For each mapped class, an IRowCallback internal class is deployed.

Name	A Value	Comment	-
addChildCompetence	update {0}.competence set(hierarchy_jd, parent_jd, last_updated)=({3}, {2}, now()) where competence_id={1}		
addCompetenceToGroup	insert into {0}.competence_group (competence_id, group_id) values ({1}, {2})		Ī
associateCompetenceToUser	insert into {0}.user_competence(competence_id, user_id) values ({1}, {2})		
associateCompToHierarchy	update {0}.competence set (hierarchy_id, last_updated)=({2}, now()) where competence_id={1}		
associateUserToRole	insert into {0}.user_role (user_id, role_id) values ({1},{2})		
closeEvaluation	update {0}.evaluation set (status_id)=(0) where (evaluation_id={1})		
createRootCompetence	update {0}.hierarchy set (root_id, last_updated)=({1}, now()) where hierarchy_id={2}		
deleteCompetence	delete from {0}.competence where competence_id = {1}		
deleteCompGroup	delete from {0}.group_cpermissions where group_cpermissions_id = {1}		
deleteRoleFromUser	delete from {0}.user_role where (role_id={1} and user_id={2})		
evaluateCompetence	update {0}.competence_evaluation set (manual_value)=({2}) where (competence_evaluation_id={1})		
getAllCompetences	select * from {0}.competence order by name_c		
getAllDomains	select * from {0}.domain order by domain_id asc		
getAllUsers	select * from {0}.user_iknow		
getChildrenComp	select * from {0}.competence where parent_id={1}		
getChildrenCompEvals	select * from {0}.competence_evaluation where (evaluation_id={2} and competence_id in (select competence_id from {0}.competence where parent_id={1}))		
getChildrenNotUser	<pre>select * from {0}.competence where (parent_id={1} and competence_id not in (select competence_id from {0}.user_competence where user_id={2}))</pre>		
getChildrenUser	select * from {0}.competence where (parent_jd = {1} and competence_jd in (select competence_jd from {0}.user_competence where (user_jd={1})))		
getCompetenceByGroup	select a.* from {0}.competence as a, {0}.competence_group as b where (a.competence_id=b.competence_id and b.group id = $\{1\}$ )		
getCompetenceByHierarchy	select * from {0}.competence where hierarchy_id={1}		
getCompetenceByID	select * from {0}.competence where competence_id ={1}		

Figure 40: *iKnow.Database* resources file

In order to encapsulate database manipulation methods, *iKnow.Contract* deploys interfaces to the services and structures defined in *iKnow.Database* (Figure 41) – that is, it implements a contract to the database services.



Figure 41: Interface classes deployed in *iKnow.Contract* 

Closely related to user interfaces, *iKnow.GUI* (Figure 42) deals with presentation and input handling issues.

Solution Explorer	•	ц	х
Solution 'Know' (4 projects)			
🐨 📴 iKnow. Auth			
🖅 📴 iKnow.Contract			
💼 🛛 📴 iKnow.Database			
🖮 🎑 iKnow.GUI			
🖬 🗉 Properties			
🖮 🔤 References			
📷 App_Data			
違 🚞 Content			
💷 🗁 🛅 Controllers			
🚞 Models			
🖅 📄 Scripts			
😟 🖷 🛄 Views			
🔤 🚵 coaching-central-nav.sitemap			
competence-central-nav.sitemap			
🖅 🛅 Default.aspx			
🖅 🔬 Global.asax			
av-config.sitemap			
🚵 profiling-nav.sitemap			
🔤 🔛 Web.config			

Figure 42: iKnow.GUI overview

Following the MVC pattern, the views are used to display data as ASP .NET pages, while controllers receive input and initiate responses by making calls on model objects. The model is not implemented in *iKnow.GUI* as it is already deployed in *iKnow.Database*.

For presentation purposes, the user menus are implemented using sitemaps and Telerik MVC Extensions. Figure 43 shows a view for competence management, where the user can view, edit and delete competences from the system. There are two navigational menus: the top one showing the main sections (*Administration, Competence Central, Evaluation, Profiling* and *Reporting*) and the lateral one detailing the subsections (*Core Management, Groups* and *Hierarchies*).

	a Toola Help								
	. Tools Deb						2.10		
C X O	and the second s						¥.	Google	
	- 1727								
ompetences			_						
								Welcome	[ <u>Log 0</u>
lnow - Im	proved	Knowledg	je M	lanag	emen	it			
ministration 👻 Com	petence Central Ev	aluation Profiling Re	porting						
Manage Comp	etences								
manage comp									
Core Management							Creat	e new competence	
Groups	Name 🗢	Description	Weight	Is enabled?	Is derivate?	Date Creation	Last Update	Actions	
Hierarchies	.NET	.NET framework	1	true	false	01-01-2010	11-02-2010	PEdit ODelete	
	ASP	Web	3	true	false	07-02-2010	07-02-2010	CEdit ODelete	
	C#	.NET Programming	2	true	false	07-02-2010	26-02-2010	CEdit ODelete	
	Coaching		3	true	false	02-01-2010	27-02-2010	Bedit ODelete	
	Decision Making		3	true	false	02-01-2010	27-02-2010	Edit ODelete	
	Leadership		3	true	false	02-01-2010	27-02-2010	CEdit ODelete	
	Problem Solving		3	true	false	02-01-2010	27-02-2010	Bedit ODelete	
	Project Managem	Manage CMF projects	1	true	false	08-02-2010	08-02-2010	CEdit ODelete	
	Soft Skills	General soft skill compete	6	true	false	07-02-2010	07-02-2010	CEdit ODelete	
	WPF	WPFoundation	4	false	false	07-02-2010	07-02-2010	Bedit ODelete	
		ie ve f	age 1			of 2 🎫 🖬		View 1 - 10 of 11	

Figure 43: User interface for competence management<sup>4</sup>, where a list of existing competences is presented

For a dynamic data presentation, jQuery is used to display data in a table layout. In particular, this approach is quite useful when showing associated data, such as hierarchies, groups or even functional associations (defined by the organization).

For example, a system user may have multiple roles (*Manager, Worker*, and *Facilitator*, to name a few). When managing user accounts, an *Administrator* views the list of all system users. In particular, he/she may desire to manage the roles associated to a specific user. In this scenario, it is crucial to have a listing that is associated to the selected user. That can be accomplished by a sub table, as shown in Figure 44.

<sup>&</sup>lt;sup>4</sup> In this figure's context, 'competence management' mustn't be considered the organization-wide strategy defined in Chapter 2. Instead, it should be interpreted as CRUD operations on competence data.

a second	igtory <u>B</u> ookmarks	Iools Help					
C	× 🔬 [		and the second	100%):			☆ • Scoople
User Ac	counts						
		Concept Name					History admini Line A
							welcome admini ( Log Ci
Knc	ow - Im	proved Ki	nowled	ae Mar	nademe	ent	
				3	3		
dministr	ation 👻 Comp	etence Central Evaluatio	n Profiling I	Reporting			
unninga	durin - comp		in the second	reporting			
Man	age User A	ccounts					
							Create new user
	Name	Email	Date Creation	Status	Coach	Facilitator	Actions
+	admin	admin@cmf.com	18-01-2010	Enabled	and a second second		Add Role ØEdit ODelete
-	ag-soares	ag-soares@cmf.com	15-01-2010	Enabled		a-santos	Add Role Sedit ODelete
. 1	Role 🛳	Actions					
	Manager	Bemove					
	Worker	Bemove					
	Facilitator	ØRemove.					
				Enabled	admin	csilva	
+	a-santos	a-santos@cmf.com	09-02-2010	FURDIDA	aurini		Add Role Fdit Delete
+	a-santos csilva	a-santos@cmf.com csilva@cmf.com	09-02-2010	Disabled	preis	ag-soares	Add Role ØEdit ODelete
+ + + +	a-santos csilva preis	a-santos@cmf.com csilva@cmf.com preis@cmf.com	09-02-2010 09-02-2010 01-01-2010	Disabled	preis	ag-soares ag-soares	Add Role CEdit Opelete
+ + + +	a-santos csilva preis f-man	a-santos@cmf.com csilva@cmf.com preis@cmf.com f-man@cmf.com	09-02-2010 09-02-2010 01-01-2010 09-02-2010	Disabled Disabled Enabled	preis	ag-soares ag-soares	Add Role & Edit Delete
+ + + + +	a-santos csilva preis f-man	a-santos@cmf.com csilva@cmf.com preis@cmf.com f-man@cmf.com	09-02-2010 09-02-2010 01-01-2010 09-02-2010	Disabled Disabled Enabled	preis	ag-soares ag-soares	Add Role Feit. Opelete Add Role Feit. Opelete Add Role Feit. Opelete Add Role Fit. Opelete
+ + + + +	a-santos csilva preis f-man	a-santos@cmf.com csilva@cmf.com preis@cmf.com f-man@cmf.com	09-02-2010 09-02-2010 01-01-2010 09-02-2010 Page 1	Disabled Disabled Enabled	preis	ag-soares ag-soares	Add Role Of Edit. Delete.
+ + + + + + + + + + + + + + + + + + + +	a-santos csilva preis f-man	a-santos@cmf.com csilva@cmf.com preis@cmf.com f-man@cmf.com	09-02-2010 09-02-2010 01-01-2010 09-02-2010	Disabled Disabled Enabled	preis	ag-soares ag-soares	Add Role Cfdit ODelete Add Role Cfdit ODelete Add Role Cfdit ODelete Add Role Cfdit ODelete View 1 - 6 of 6
+ + + + + + + + + + + + + + + + + + + +	a-santos csilva preis f-man	a-santos@cmf.com csilva@cmf.com preis@cmf.com f-man@cmf.com	09-02-2010 09-02-2010 01-01-2010 09-02-2010 14 <4 Page 1	Disabled Disabled Enabled	preis	ag-soares ag-soares (1 => =>	Add Role. ØEdit ODelete. Add Role. ØEdit ODelete. Add Role. ØEdit ODelete. Add Role. ØEdit ODelete. View 1 - 6 of 6

Figure 44: User interface for system users' account management

Similarly, the mentioned approach is used when managing competence groups. Within a given group, a user might want to visualize its competences. Once again, a sub table with a further level of detail is necessary. Such situation is pictured in Figure 46 and Figure 45

ge Compe it View I	tence Groups - A History Bookmarks	lozilla Firefo Tools Help				Ŀ
C	XA	1 total libration	t Williamstern Gruntler	and a second	🗘 • 💐• Goode	
	~ w (	1		age on a	M Roope	
_						_
nage Comp	etence Groups					
					Welcome admin! [ Lo	g Off ]
iKno	w - Im	nrove	d Knowler	dae Manademe	nt	
ITAIL	vv - IIII	prove		age manageme		
	1.1.					
Administr	ration - Compe	tence Central	Evaluation Profiling	Reporting		
Man	ane Comne	tence Gr	une			
man	age compe	tence on	Jups			
Co	re Management				Create new group	
Gra	nuns		Name 🔺	Description	Actions	
01	Jups	+	Group A	Competence group A		
File	erarchies	+	Group E	Competence group E		
		+	Group B	Competence group B		
		+	Group C	Competence group C	Cedit ODelete	
		+	Group D	Competence group D	CEdit ODelete	
			Page	1 of	1	
			- S			
		_				



Know - Ir						Utalaama	100.06
		VO	Knowlod	ao Managon	ont	Weicome	Log or
	npro	vec	I Knowled	ge managen	lent		
Administration - Co	mpetence Ce	ntral	Evaluation Profiling P	Reporting			
		10000					
Manage Com	netence	Gro	uns				
manage com	iperence	9100	iha				
Core Managemen	nt					Create new gro	up
Groups			Name 🗢	Description		Actions	
Hierarchies	-		Group A	Competence group A	6	Edit ODelete Add	
		ID	Name 🗢	Description	Weight	Actions	
		4	WPF	WPFoundation	4	ODelete_	
		5	XAML	MS XML	5	S Delete	
	+		Group E	Competence group E	6		-
	+		Group B	Competence group B		Edit ODelete	-
			Group C	Competence group C	6	Edit ODelete	-
	+						
	+		Group D	Competence group D	6	Edit Delete Add	
	+		Group D	Competence group D	of 1	Edit View 1 - 5 of	5

Figure 45: List of competences within a given group

A typical example for associated data dynamic display needs is competence hierarchies. Indeed, the dynamic presentation provided by jQuery allows global perspective on the competence tree, where it is possible to browse throughout the whole tree, with variable complexity levels (Figure 47).



Figure 47: Visualization of competences as hierarchies

To build these dynamic tables, the jQuery library was used in several JavaScript files (Figure 48), which were referenced by the corresponding ASP .NET pages. That is, for the visualization of competence hierarchies, the hierarchyManagement.js file was referenced in.



Figure 48: Overview of developed scripts using the jQuery library

**HierarchyManagement.aspx**, for example. In its turn, the script was responsible for building the table (alternatively called grid) dynamically. To populate the table with data, the script used an **url** parameter that pointed to an action in the controller. This action was responsible for delivering the data to the grid. In its turn, the controller action returned JSON data to the grid, which was then presented in the view. An intuitive graphical approach to this process is shown in Figure 49.



Figure 49: Intuitive description of the rendering of a table in a view using jQuery

jQuery is also useful for the display of modal dialogs. Such approach is adequate when it is intended that the user focuses on a single area within the current GUI's context (alert messages). In particular, this scenario can be applied to competence evaluation. When viewing a set of competence evaluation processes, a user may wish to open one of them. Opening this process implies the appraisal of a set of competences, which can be presented using a dialog (Figure 50).



Figure 50: Dialog for competence evaluation
# 7.3 Testing

As to verify and validate the implemented prototype, unit tests were performed with successful results. An overview of the performed tests in each implemented module is detailed for *Core competence management* in Table 32 and *Security/Privacy* in Table 33.

Table 32: Unit tests	performed on th	e Core Con	npetence Manag	gement deploye	d use cases
			1 0	, , ,	

Sub module	Use Case(s)	Sample Tests
Competence evaluation	Auto-evaluate a competence	<ul> <li>Check if evaluated competence belongs to the worker's pro- file;</li> <li>Check if current competence evaluation process is not closed.</li> </ul>
	Evaluate worker's competence	<ul> <li>Check if the user is a facilitator of the worker being evaluated;</li> <li>Check if current competence evaluation process is not closed;</li> <li>Check if the user has permissions to evaluate the competence (that is, if he/she is associated to a competence group that contains that competence).</li> </ul>
	Rate a competence	- Check if rating level is integer.
Competence derivation	Edit derived com- petence	<ul> <li>Check if user has permissions to change competence properties;</li> <li>Check if competence name already exists;</li> <li>Check if competence name is in the correct format.</li> </ul>
	View derived com- petence	- Check if user has permissions to view that derived compe- tence.
Basic management	Create/delete competence Enable/disable competence	<ul> <li>Check if user has permissions to manage that competence;</li> <li>Check if competence name already exists (when creating a new competence);</li> <li>Check if competence to be deleted is not associated to other users;</li> <li>Check if competence to be deleted is an intermediate node of a competence derivation graph;</li> <li>Check if competence to be deleted is a leaf node of a competence hierarchy tree.</li> <li>Check if user has permissions to manage that competence.</li> </ul>
	View competence	- Check if user has permissions to view that competence.

Hierarchy	Create/delete com- petence hierarchy	- Check if user has permissions to manage competences in that hierarchy;
		- Check if competence hierarchy already exists (when creating a new one);
		- Check if any competence from the hierarchy to be deleted is associated to some user.
	Edit competence hierarchy	- Check if user has permissions to manage the competences in that hierarchy;
		- Check if added/removed competences do not belong to other hierarchies;
		- Check if tree has one and only one root competence.
	View competence hierarchy	- Check if user has permissions to view competences in that hierarchy.
	Add/remove com- petence	- Check if user has permissions to manage the target profile;
~		- Check if added competence is not already in the target profile;
Competence profile		- Check if competence to be removed belongs to the profile.
	View worker's competence profile	- Check if user has permissions to view that worker's profile, that is, if he/she is his/her supervisor or coach;
Competence grouping	Add/remove com- petence	- Check if competence to be added is not already in the group;
		- Check if competence to be removed is associated to the group.
	Create/delete com- petence group/subgroup	- Check if name of group(subgroup to be created does not exist;
		- Check if competence group/subgroup to be removed exists;
		- Check if competence group/subgroup is not associated to other users;
		- Check if user has permissions to manage the group/subgroup.
	Edit competence group/subgroup	- Check if competence group/subgroup name to be updated does not exist;
		- Check if user has permissions to edit the group/subgroup.

### Prototype Deployment

#### Prototype Deployment

Use Case(s)	Sample Tests
	- Check if username and/or login domain exist;
Login	- Check if password is correct for the specified user name.
Logout	- Check if user is logged in the system.
Register account	- Check if username specified does not exist.
Manage user accounts	- Check if user has permissions to manage user accounts (that is, he/she must be administrator);
	- Check if username to be updated does not exist.

Table 33: Unit tests performed on the Security/Privacy deployed use cases

#### 7.4 Summary

So far, it has been studied the impact of competence management in organizations. In particular, the requirements for a Competence Management System for Critical Manufacturing were analysed and its design was specified. As a proof-of-concept for all the research carried out so far, a prototype was implemented and focused on the *Core competence management* and *Security/Privacy* modules.

Using Visual Studio as Integrated Development Environment, the architecture and technologies presented in Chapter 6 were deployed.

As to validate the developed prototype, unit tests were carried out for each module's use cases, with successful results.

The deployed use case models match key phases in the competence management strategy (competence mapping and appraisal), setting a solid ground for future developments.

Conclusions and Future Work

## **Chapter 8**

# **Conclusions and Future Work**

Learning is at the heart of a company's ability to adapt to a rapidly changing environment. It is the key to being able both to identify opportunities that others might not see and to exploit those opportunities rapidly and fully. This means that in order to generate extraordinary value for shareholders, a company has to learn better than its competitors and apply that knowledge throughout its businesses faster and more widely than they do. The way we see it, anyone in the organization who is not directly accountable for making a profit should be involved in creating and distributing knowledge that the company can use to make a profit.

Sir John Browne, CEO of British Petroleum [109]

Whilst financial wealth has a direct impact on today's corporate profitability, the role of intellectual capital is to build future earning capabilities [110]. Indeed, it is the capitalization of an organization's workforce knowledge – in particular, competences – that decisively contributes for a long-term sustained competitive advantage. Such strategic perspective on knowledge-intensive enterprises is the core of Competence Management (CM).

Competence Management Systems (CMSs) deploy this approach, providing multiple corporate benefits, such as the following [111]:

- Support of a systematic strategic competence development;
- Identification of competence gaps for each business unit, allowing the targeting of competence development;
- Establishment of a sound basis for the construction of individual competence development plans closely aligned with corporate goals;
- Support for talent management, i.e., global search for the identification of employee with high levels of expertise;

#### Conclusions and Future Work

- Detailed overview of organizational competences through competence profiles and statistics for employees, positions and organizational units;
- Increased flexibility, speed and accuracy, by identifying who knows what on a global basis and improving effective utilization of the overall human resource potential;
- Support for project teams staffing with specific competence demands;
- Increased awareness and focus on personal competence development pushing management harder for support and development;
- Exposure of previously hidden/unknown competences for employees, giving them possibilities for new and interesting assignments.

Taking a corporate instance with a lack of a solid CM strategy (Critical Manufacturing), this project approached the mapping of these benefits to the organization through the requirements analysis, design and prototype deployment of a CMS. Hence, corporate performance among its competition peers and, ultimately, corporate success are expected to increase.

From a global perspective, this document answered the initially proposed questions, as follows:

#### What is the project about?

This project is about CM and its impact on organization performance. Focused on corporate knowledge capital, it is intended to understand how CM can aid to improve corporate success. However, as with most enterprise applications, bear in mind that a CMS only provides the basis for improved organization performance (among its competitors) – it is not a guarantee. Corporate success is not a direct and absolute consequence of a CMS deployment: the organization must make an effort to adapt itself to this strategy and its workforce must embrace it if the organization wishes to obtain a non-negative Return On Investment.

#### - What is its innovative stamp?

The innovative stamp of this project is the study and IT-supported application of CM concepts in a company oriented to manufacturing systems which lacks a solid competence management strategy. Moreover, the whole software development cycle is contemplated, ranging from requirements analysis to prototype deployment, validation and acceptance. In spite of taking CMF as the corporate instance for CM strategy deployment, iKnow's design is quite generic, thus adaptable to other corporate contexts.

- Which were the project results?

Aside from the research effort on state-of-the-art CMSs, the iKnow project materialized in a fully documented software development cycle (requirements analysis, design engineering and prototype deployment and acceptance) of a CMS for a corporate instance: Critical Manufacturing.

#### - To what extent did it contribute to the scientific community?

The main contribution was the study of the impact of CM strategy in corporate environment. In particular, all the key phases of such strategy (competence mapping, diagnosis, development and monitoring) were translated into use cases, grouped by feature modules and a proof-of-concept prototype was deployed and validated.

#### 8.1 Goal Satisfaction

All initially proposed goals were successfully met, as follows:

- *Study of CM concepts and key issues*, through the analysis of the motivation for CM in organizations, corporate benefits, application areas and key challengers. Additionally, two complementary strategic outlooks on CM (Vernhout [1] and by Draganidis et al. [24]) were reported, covering all key phases ranging from competence mapping to evaluation processes;
- *State-of-the-art review of CMSs and main features*, reporting a study of current CMS tools as well as a literature review on the key challenges identified in the previous topic (competence modelling and profiling);
- Proposal of a CMS suitable for a given organization (Critical Manufacturing), applying the acknowledged CM concepts to this corporate instance, accomplished through a complete cycle of software development. As previously stated, although a CM strategy has been instanced for this organization, iKnow's design is comprehensive and generic enough to be marketed for external corporations;
- *Implementation of a working CMS prototype*, as a proof-of-concept for the underlying research work, where significant core features where deployed, thus setting the basis for future developments.

Additionally, all software development phases were fully documented, alongside with weekly project progress reports.

### 8.2 Future Work

The work developed so far stands a solid basis for future developments. In particular, the subsequent enhancements are suggested:

- *Deployment of RESTful Web Services* towards the interoperability of the implemented prototype with internal Critical applications
- *Implementation of the remaining use cases modules*, as to fully profit from a CMS's potential benefits
- Use of portlets for the Social Network module deployment, thus reducing programming effort overload and taking advantage of already existing and fully-featured common APIs, such as OpenSocial from Google [112].
- *Competence concept formalization*, as proposed by the several works analysed in the state-of-the-art review, namely using ontology-based models of competences (semantic web) [24]
- Application of distributed artificial intelligence techniques; focused on the complexity of resource management in the civil construction sector, Oliveira et al [113] suggest the implementation of a computer system based on distributed artificial intelligence techniques, enabling a decentralized management of different resources with negotiation processes. Moreover, it comprises the existence of an agent hierarchy, mentioning roles such as *Facilitator* or *Workers*, quite similar to iKnow's. Should

#### Conclusions and Future Work

Critical Manufacturing (CMF) decide to expand iKnow to its additional facilities in Germany and/or China, it would be exciting to take advantage of a distributed artificial intelligent system for its adaptability, cost efficiency and autonomy to manage competences. In this case, an adaptation of the system proposed by Oliveira et al. could be performed, where resources were competences and agents represented CMF's workers.

- Development of preference-based project and training programmes assignment; in their work, Peters and Zelewski [114] present a goal programming model for workplace assignment, which takes into account both employee competences and preferences and workplace competence requirements and attributes. It would be interesting to apply this approach using workers' wishlist as preferences and training programmes or projects as workplaces to be assigned.

# References

- 1. Vernhout, A.: Competence-based Strategic Management. (2007)
- Dorn, J., Pichlmair, M., Tellioğlu, H.: Supporting Competence Management in Software Projects. In : 14th International Conference on Consurrent Enterprising, Lisboa, Portugal, p.8 (2008)
- 3. Prahalad, C., Hamel, G.: The Core Competence of the Corporation. (2003)
- 4. Ordonez de Pablos, P., Lytras, D.: Competencies and human resource management: implications for organizational competitive advantage. Journal of Knowledge Management 12(6), 48-56 (2008)
- 5. Barney, J.: Firm resources and sustained competitive advantage. Journal of Management 17(1), 99-120 (1991)
- 6. Finkelstein, S., Hambrick, D.: Strategic Leadership. West Publishing Co (1996)
- Huselid, M.: The impact of human resource management practices on turnover, productivity, and corporate financial performance. Academy of Management Journal 38, 635-72 (1995)
- 8. Pennings, J., Lee, K., Van Witteloostuijn, A.: Human capital, social capital and form dissolution. Academy of Management Journal 41, 425-40 (1998)
- 9. Pfeffer, J.: The human equation: Building profits by putting people first. Harvard Business School Press, 64–98 (1998)
- 10. Wright, P., Smart, D., McMahan, G.: Matches between human resources and strategy among NCAA basketball teams. Academy of Management Journal 38, 1052-74 (1995)
- Sparrow, P., Bognanno, M.: Competency requirement forecasting: issues for international selection and assessment. International Journal of Selection and Assessment 1(1), 50-8 (1993)
- 12. Holmes, L.: HRM and the irrestible rise of the discourse of competence. Personnel Review 24(4), 34-49 (1995)
- 13. Athey, T., Orth, M.: Emerging competency methods in the future. Human Resource Management 38(3), 215-26 (1999)
- 14. Seubert, E., Balaji, Y., Makhija, M.: The Knowledge Imperative. CIO Special Advertising Supplement (2001)
- 15. Halawi, L., Aronson, J., McCarthy, R.: Resource-Based View of Knowledge Management for Competitive Advantage. The Electronic Journal of Knowledge Management 3(2), 75-86 (2005)
- 16. Hofer-Alfeis, J.: Effective Integration of Knowledge Management into the Business Starts with a Top-down Knowledge Strategy. Journal of Universal Computer Sciente 9(7), 719-728 (2003)
- 17. Durgin, T.: Implementing a Successful Competency Model., Human Capital Institute

(August 2006)

- Pengxiang Jia, Q.: Knowledge Management, Collaboration and Innovation. In : 2008 International Seminar on Future Information Technology and Management Engineering (2008)
- Wernerfelt, B.: A resource-based view of the firm. Strategic Management Journal 5(2), 171-180 (1984)
- Prahalad, C. K., Hamel, G.: The Core Competence of the Corporation. Harvard Business Review 68(3), 79-93 (1990)
- 21. Hessami, A., Moore, M.: Competence Matters more than Knowledge. The Electronic Journal of Knowledge Management 5(4), 387-398 (2007)
- 22. D' Adderio, L.: Configuring Software, Reconfiguring Memories: The Influence of Integrated Systems on the Reproduction of Knowledge and Routines. Industrial and Corporate Change 12(2), 321-350 (2003)
- 23. Abdullah, M., Kimble, C., Bernest, I., Paige, R.: Knowledge-Based Systems A Re-Evaluation. Journal of Knowledge Management 10(3), 127-142 (2006)
- 24. Draganidis, F., Mentzas, G.: Competency based management: a review of systems and approaches. (2006)
- 25. Michellone, G., Zollo, G.: Competencies management in knowledge-based firms. Journal of Manufacturing Technology and Management 1(1), 20-41 (2000)
- Harzallah, M., Berio, G., Vernadar, F.: Analysis and Modeling of Individual Competencies: Toward Better Management of Human Resources. In : IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans 36. (2006)
- 27. Durgin, T.: Using Competency Management to Drive Organizational Performance., Human Capital Institute (June 2006)
- 28. Braun, S., Kunzmann, C., Schmidt, A.: People Tagging & Ontology Maturing: Towards Collaborative Competence Management.
- 29. Kaplan, R., Norton, D.: Le Tableau de Bord Prospectif Pilotage stratégique: les 4 axes du succès., Paris (1998)
- Vasconcelos, J., Kimble, C., Miranda, H., Henriques, V.: A Knowledge-Engine Architecture for a Competence Management Information System. In : Proceedings of 14th UKAIS Conference, Oxford (1999)
- 31. Marelli, A.: An introduction to competency analysis and modeling. Performance Improvement 37, 8-17 (1998)
- 32. Dubois, D.: Preface. In : The Competency Casebook: Twelve Studies in Competency-Based Performance Improvement. HRD Press (1998)
- 33. HR-XML Consortium: HR-XML. (Accessed 2010) Available at: http://www.hr-xml.org
- Boyatzis, R.: The Competent Manager: A Model for Effective Performance. Wiley, New York, NY (1982)
- 35. Selby, S., Wright, L.: Investigation of the application of frontline management training in the community services and health industry: a progress report. In : Victorian Community Services in Health Industry Training Board Conference, Melbourne (2000)
- 36. The UK National (and Scottish) Vocational Qualification System: State of the art or in a state?, 242 (1997)
- Treasury Board of Canada Secretariat: Framework for competency-based management in the public service of Canada. Report of joint initiative between the Treasury Board of Canada Secretariat and the Public Service Commission (1999)
- 38. Perrenaud, P.: 10 Novas competencias para enseñar. ArtMed Editora (2000)

- 39. LeBoterf, G.: Evaluer les compétences, quels jugements? Quels critères? Quelles instances? La Compétence au Travail 135(2), 143-51 (1998)
- 40. Jackson, S., Sculer, R.: Managing Human Resources through Strategic Partnerships 8th edn. South-Western (2003)
- 41. Intagliata, J., Ulrich, D., Smallwood, N.: Leveraging leadership competencies to produce leadership brand: creating distinctiveness by focusing on strategy and results. Human Resource Planning 23(3), 12-23 (2000)
- 42. Carbone, P., Brandão, P., Leite, J., Vilhena, R.: Gestão por competência e gestão do conhecimento 2nd edn. FGV, Rio de Janeiro, Brazil (2006)
- 43. Houtzagers, G.: Empowerment, using skills and competence management. Participation & Empowerment: An International Journal 7(2), 27-32 (1999)
- Hellstrom, T., P. and Malnquist, U.: Knowledge and competency management in ericson: decentrilization and organizational fit. Journal of Knowledge Management 4(2), 99-110 (2000)
- 45. Beck, S.: Skill and competence management as a base of an integrated personnel development (IPD) a pilot project in the Putzmeister, Inc/Germany. Journal of Universal Computer Science 9(12), 1381-7 (2003)
- 46. Won, M., Pipek, V.: Sharing knowledge on knowledge the exact peripheral expertise awareness system. Journal of Universal Computer Science 9(12), 1388-97 (2003)
- Hockermeyer, C., Conlan, O., Wade, V., Albert, D.: Applying competence prerequisite structures for elearning and skill management. Journal of Universal Computer Science 9(12), 1428-36 (2003)
- Colucci, S., Di Noia, T., Sciascio, E., Domini, E., Mongiello, M., Mottola, M.: A formal approach to ontology-based semantic match of skills descriptions. Journal of Universal Computer Science 9(12), 1437-54 (2003)
- Dorn, J., Pichlmair, M., Tellioğlu, H.: Supporting Competence Management in Software Projects. In : 14th International Conference on Consurrent Enterprising, Lisboa, Portugal, p.8 (2008)
- 50. Davenport, T., Prusak, L.: Working Knowledge: How Organizations Manage what they Know., Massachusetts, Boston (1998)
- 51. Hustad, Munkvold: IT-Supported Competence Management: A Case Study at Ericsson. Journal of Information Systems Management (2005)
- 52. al., Z.: Ontologiebasierte Kompetenzmanagementsysteme: Grundlagen, Konzepte, Anwendungen. (2005)
- 53. Dorn, J., Pichlmair, L.: A Competence Management System for Universities. In : European Conference on Information Systems, St. Gallen, pp.759-770 (2007)
- 54. Pichlmair, M.: Universitares Kompetenzmanagement, Ph. D. Thesis., Vienna University of Technology, Computer Science Faculty (2008)
- 55. Talent Management. In: Taleo. (Accessed January 2010) Available at: http://www.taleo.com
- 56. IMS RDCEO: IMS Reusable Definition of Competency or Educational Objective (RDCEO). In: IMS GLC: Competencies Specification. Available at: http://www.imsglobal.org/competencies/
- 57. IEEE In: IEEE 1484.20.1/Draft 4 Draft Standard for Reusable Competency Definitions. Available at: <u>http://www.ieeeltsc.org:8080/Plone/working-group/competency-data-standards-working-group-20/IEEE\_1484.20.1.D4.pdf</u>
- 58. De Coi, J., Herder, E., Koesling, A., Lofi, C.: A Model for Competence Gap Analysis. INSTICC Press (2007)

- 59. HR-XML Measurable Competencies. In: hr-xml. (Accessed 2010) Available at: http://www.hr-xml.org
- 60. Simple Reusable Competency Map proposal (SRCM). In: Ostyn Consulting Resources. Available at: <u>http://www.ostyn.com/resources.htm#competency</u>
- 61. Sure, Y., Maedche, A., Staab, S.: Leveraging corporate skill knowledge from proper to ontoproper. In : 3rd International Conference on Practical Aspects of Knowledge Management, Basel, Switzerland (2000)
- 62. Colucci, S., Noia, S., Sciascio, E., Domini, F., Mongiello, M., Mottola, M.: A formal approach to ontology-based semantic match of skills descriptions. (2003)
- 63. Lefebvre, B., Gauthier, G., Tadié, S., Duc, T., Achaba, H.: Competence ontology for domain knowledge dissemination and retrieval., 845-859 (2005)
- 64. Lau, T., Sure, Y.: Introducing Ontology-based Skills Management at a large Insurance Company. In : Proceedings of the Modellierung. (2002) 123-134
- 65. Biesalski, E., Abecker, A.: Human Resource Management with Ontologies. In : Professional Knowledge Management, 3rd Biennial Conference, Kaiserslautern, Germany, pp.499-507 (2005)
- 66. Becerra-Fernandez, I.: Searching for experts on the Web: A review of contemporary expertise locator systems. ACM Transactions on Internet Technologies, 333-355 (2006)
- 67. Crowder, R., Hughes, G., Hall, W.: Approaches to locating expertise using corporate knowledge. Journal of Intelligent Systems in Accounting, Finance & Management 11(4), 185–200 (2002)
- 68. Reichling, T., Veith, M., Wulf, V.: Expert Recommender: Designing for a Network Organization. Journal of Collaborative Computing 14(4/5), 431-465 (2007)
- 69. Ley, T., Lindstaedt, S., Albert, D.: Competency Management Using the Competence Performance Approach: Modelling, Assessment, Validation and Use. Competencies in Organizational E-Learning, 83–119 (2006)
- 70. LinkedIn. (Accessed 2010) Available at: http://www.linkedin.com
- 71. Xing. (Accessed 2010) Available at: <u>http://www.xing.co</u>
- 72. In: Competence Management Systems. Available at: <u>http://www.rail-reg.gov.uk/server/show/ConWebDoc.9915</u>
- 73. Critical: Critical Careers. (Accessed 2010) Available at: <u>http://tbe.taleo.net/NA8/ats/careers/searchResults.jsp?org=CRITICALSOFTWARE&cws</u> <u>=1</u>
- 74. Critical Manufacturing S.A.: Critical Manufacturing. (Accessed 2009) Available at: http://www.criticalmanufacturing.com/
- 75. Brooks, F.: The Mythical Man-Month: Essays on Software Engineering 2nd edn. Addison-Wesley Professional (1995)
- 76. Sommerville: Software Engineering 8th edn. Addison-Wesley (2006)
- 77. Abran, A., Moore, J.: Guide to the Software Engineering Body of Knowledge. (2004)
- 78. Booch, G.: Object Oriented Design with Applications. The Benjamin/Cummings Publishing Company, Inc., Redwood City (1991)
- 79. Greer, D.: The Art of Separation of Concerns. (Accessed 2010) Available at: http://www.aspiringcraftsman.com/2008/01/art-of-separation-of-concerns/
- 80. Model-View-Controller. In: msdn Microsoft patterns & practives. (Accessed 2009) Available at: <u>http://msdn.microsoft.com/en-us/library/ms978748.aspx</u>
- 81. Caprio, G.: Design Patterns Dependency Injection. (Accessed 2010) Available at: http://msdn.microsoft.com/en-us/magazine/cc163739.aspx

- 82. What is Dependency Injection? (Accessed 2010) Available at: http://abhisays.com/java/what-is-dependency-injection.html
- Prasanna, D.: Dependency Injection Design Patterns using Spring and Guice. Manning (2009)
- 84. Wikipedia: Dependency Injection. (Accessed 2010) Available at: http://en.wikipedia.org/wiki/Dependency\_injection
- 85. Fossmo, P.: Dependency Injection explained. (Accessed 2010) Available at: http://blog.fossmo.net/post/Dependency-Injection-explained.aspx
- 86. Jenkov, J.: Dependency Injection Benefits. (Accessed 2010) Available at: http://tutorials.jenkov.com/dependency-injection/dependency-injection-benefits.html
- 87. Implementations of MVC as Web-Based Frameworks. In: Wikipedia. (Accessed November 10, 2009) Available at: <u>http://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller#Implement</u> <u>ations\_of\_MVC\_as\_web-based\_frameworks</u>
- List of Relational Database Management Systems. In: Wikiepdia. (Accessed November 13, 2009) Available at: http://en.wikipedia.org/wiki/List of relational database management systems
- List of Object Database Management Systems. In: Wikipedia. (Accessed November 11, 2009) Available at: http://en.wikipedia.org/wiki/List of object database management systems
- 90. Document-Oriented Database. In: Wikipedia. (Accessed November 10, 2009) Available at: <u>http://en.wikipedia.org/wiki/Document-oriented\_database#Implementations</u>
- 91. Microsoft: What is ASP.NET MVC? (Accessed 2010) Available at: http://www.asp.net/mVC/whatisaspmvc/
- 92. Spring.NET: Spring.NET Overview. (Accessed 2010) Available at: http://www.springframework.net/overview.html
- 93. jQuery. (Accessed 2010) Available at: <u>http://jquery.com/</u>
- 94. jQuery: The Write Less, Do More, JavaScript Library. (Accessed 2010) Available at: http://jquery.com/
- 95. telerik In: Telerik Extensions for ASP.NET MVC. (Accessed 2010) Available at: http://www.telerik.com/products/aspnet-mvc.aspx
- 96. Wikipedia: PostgreSQL. (Accessed 2010) Available at: http://en.wikipedia.org/wiki/PostgreSQL
- 97. PostgreSQL: The world's most advanced open source database. (Accessed 2010) Available at: <u>http://www.postgresql.org/</u>
- 98. Npgsql: Npgsql -.Net Data Provider for Postgresql. (Accessed 2010) Available at: http://npgsql.projects.postgresql.org/
- 99. Rodriguez, A.: RESTful Web Services. (Accessed 2010) Available at: http://www.ibm.com/developerworks/webservices/library/ws-restful/
- 100. Evenepoel, K.: Windows Communication Foundation and RESTful Web Services Primer. (Accessed 2010) Available at: <u>http://www.codeproject.com/KB/WCF/REST\_Primer.aspx</u>
- 101. Nair, D.: A Windows Communication Foundation (WCF) Overview. (Accessed 2010) Available at: <u>http://www.codeproject.com/KB/WCF/WCFOverview.aspx?msg=3242458</u>
- 102. Wikipedia: Resource oriented architecture. (Accessed 2010) Available at: http://en.wikipedia.org/wiki/Resource\_oriented\_architecture
- 103. Len Bass, P.: Software Architecture in Practice 2nd edn. Addison-Wesley, Boston (2003)

- 104. Kang, S., Choi, Y.: Designing Logical Architectures of Software Systems. In : Sixth International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing and First ACIS International Workshop on Self-Assembling Wireless Networks (SNPD/SAWN'05) (2005)
- 105. Multitier Architecture. In: Wikipedia. (Accessed October 8, 2009) Available at: http://en.wikipedia.org/wiki/Multitier\_architecture
- 106. Normén, F.: Controllers in the MVC pattern is not a replacement for Business logic layer. (Accessed 2010) Available at: <u>http://weblogs.asp.net/fredriknormen/archive/2008/12/01/controllers-in-the-mvc-pattern-is-not-a-replacement-for-business-logic-layer.aspx</u>
- 107. Web Services Architecture. In: W3C. (Accessed February 10, 2004) Available at: http://www.w3.org/TR/ws-arch/
- 108. Microsoft: Architecture. In: MSDN Microsoft Patterns and Practices. (Accessed 2010) Available at: <u>http://msdn.microsoft.com/en-us/library/cc304695.aspx</u>
- 109. The knowledge factory. (Accessed 2010) Available at: http://www.nickmilton.com/2010/02/knowledge-factory.html
- Baladi, P.: Knowledge and Competence Management: Ericsson Business Consulting. Business Strategy Review 10(4), 20-28 (1999)
- 111. Hustad, E., Munkvold, B., Møll, B.: Using IT for Strategic Competence Management: Potential Benefits and Challenges. IS Management 22(2), 78-88 (2005)
- 112. Google: OpenSocial. (Accessed 2010) Available at: http://code.google.com/apis/opensocial/
- Oliveira, E., Fonseca, J., Steiger-Garção, A.: MACIV: a DAI based Resource Management System. Applied Artificial Intelligence 11(6), 525-550 (September 1997)
- Peters, L., Stephan, Z.: Competence and Preference-based Workplace Assignment. In : Integrating Human Aspects in Production Management 160/2005. Springer Boston (2005) 31-43
- 115. Gestão de competências e gestão de desempenho: tecnologias distintas ou instrumentos de um mesmo constructo? Revista de Administração de Empresas 41(1), 8-15 (2001)
- 116. Taleo. (Accessed 2010) Available at: http://www.mentalmodels.com/pages/portfolio/taleo/PerfModule-review-c.png

# Glossary

.NET Enterprise Services	Feature included in the Microsoft .NET Framework where you can take advantage of COM+ features for .NET objects. Additionally, .NET Enterprise Services includes some new features for .NET components that are not available to COM components. Provided through the System.EnterpriseServices namespace in the .NET Framework, Enterprise Services provides managed APIs to COM+.
.NET Framework	Software framework that can be installed on computers running Microsoft Windows operating systems. It includes a large library of coded solutions to common programming problems and a virtual machine that manages the execution of programs written specifically for the framework.
.NET Messaging	Included in the .NET Framework, it provides managed APIs to Microsoft Message Queuing, enabling developers to build asynchronous reliable distributed applications.
.NET Remoting	Microsoft API for interprocess communication released in 2002 with the 1.0 version of .NET Framework. It allows an application to make an object (termed remotable object) available across remoting boundaries, which includes different application domains, processes or even different computers connected by a network.
Active Directory	Technology created by Microsoft that provides a variety of network services, including LDAP-like directory services, DNS-based naming and central storage location for application data, to name a few.
ADO .NET	Set of computer software components that can be used by programmers to access data and data services. It is a part of the base class library that is included with the Microsoft .NET framework. It is commonly used by programmers to access and modify data stored in relational database systems, though it can also be used to access data in non-relational sources.
Architectural Design Pattern	An architectural design pattern is a software pattern that offers well- established solutions to architectural problem in software engineering, thus giving a description of the elements and relation type together with a set of utilization constraints. Fundamentally, it expresses a structural organization schema for a software system on a large scale.

COM+	An extension of Component Object Model (COM), Microsoft's strategic building block approach for developing application programs. It is both an object-oriented programming architecture and a set of operating system services. It adds to COM a new set of system services for application components while they are running, such as notifying them of significant events or ensuring they are authorized to run.
DOM	Cross-platform and language-independent convention for representing and interacting with objects in HTML, XHTML and XML documents
Gap analysis	Analytic review of competence values to compare the actual profile to a potential or aimed one.
LDAP	An application protocol for querying and modifying data using directory services running over TCP/IP.
Return On Investment	Very popular for its versatility and simplicity, it is a performance measure used to evaluate the efficiency of an investment or to compare the efficiency of a number of different investments. To calculate Return On Investment, the benefit (return) of an investment is divided by the cost of the investment; the result is expressed as a percentage or a ratio.
Service-Oriented Architecture	Flexible set of design principles used during the phases of systems development and integration. A deployed Service-Oriented based architecture will provide a loosely-integrated suite of services that can be used within multiple business domains.
Stakeholders	Entity within or outside the organization which has an interest/gain upon the successful completion of the project.
Web Services	Business logic components which provide functionality via the Internet using standard protocols such as HTTP. Web services communicate via either SOAP or REST messages.
Web Services Enhancements	Add-on to the Microsoft .NET Framework which includes a set of classes that implement additional WS-* web service specifications chiefly in areas such as security, reliable messaging, and sending attachments.
Windows Communication Foundation	An application programming interface in the .NET Framework for building connected, service-oriented applications.
Windows Presentation Foundation	A graphical subsystem for rendering user interfaces in Windows-based applications, initially released as part of .NET Framework 3.0.
WS-Addressing	Specification of transport-neutral mechanisms that allow web services to communicate addressing information (also known as Web Services Addressing).
WS-Security	Flexible and feature-rich extension to SOAP to apply security to Web services (also known as Web Services Security).

# Annex A : Additional Use Case Models

In chapter 5.5.2 (Use Case Models) the main use case models for iKnow were presented. This section presents the remaining use cases, grouped by (sub) modules.

### A.1 Core Competence Management

#### A.1.1 Competence evaluation



Identifier	UC-CCM-CE -1	
Name	Assess competence risk	
Description	Actor evaluates the risk associated to that competence value.	
Actor(s)	Coach	
Priority	Important	
	Scenarios	
Main	The actor accesses a worker's profile (his/her trainee) and associates a risk level to a compe- tence.	
Alternative	The actor cancels the operation.	
<b>Pre-conditions</b>	Actor must be logged in.	
Post-conditions	Actor can view the complete competence evaluation.	
Assumptions	Actor cannot assess competence risk of a worker that is her/his supervisor.	
Identifier	UC-CCM-CE -3	
Name	Evaluate a worker's competence	
Description	Actor evaluates the risk associated to that competence value.	
Actor(s)	Coach, Facilitator	
Priority	Essential	
	Scenarios	
Main	The actor accesses a list of workers he/she's responsible for and clicks on the evaluation menu. Then, the evaluation form of the selected worker is shown and evaluation can be performed.	
Alternative	The actor cancels the operation.	
Pre-conditions	Actor must be logged in.	
Post-conditions	Actor can view the complete competence evaluation.	
Assumptions	Actor cannot evaluate a worker that is her/his supervisor. Facilitator and Coach's evaluations mustn't overlap.	
Identifier	UC-CCM-CE -4	
Name	Rate a competence	
Description	Actor assigns a level of expertise to a competence.	
Actor(s)	Worker, Facilitator, Coach	
Priority	Essential	
Scenarios		
Main	The actor attributes a numerical/qualitative level to a competence and submits.	
Alternative	The actor cancels the operation.	
Pre-conditions	Actor must be logged in. An evaluation process must have been started.	
Post-conditions	Actor can view the competences evaluation.	
Assumptions	Actor cannot rate a competence of his/her supervisor. Worker can only rate his/her own competences. Facilitator and Coach can only rate competences of workers they are responsible for.	

## A.1.1 Competence Derivation



Identifier	UC-CCM-CD-1	
Name	View derived competence	
Description	Actor consults a list of competences that were automatically deduced from others he/she currently possesses.	
Actor(s)	Worker	
Priority	Desirable	
Scenarios		
Main	Actor accesses his/her list of competences and sees derived competences tagged.	
Alternative	The actor cancels the operation.	
Pre-conditions	Actor must be logged in.	
Post-conditions	Actor can view a list of the derived competences.	
Assumptions	Actor cannot view the derived competences of other workers.	

Identifier	UC-CCM-CD-2	
Name	Edit derived competence	
Description	Actor edits the value of derived competences in his profile.	
Actor(s)	Worker	
Priority	Desirable	
Scenarios		
Main	Actor accesses his/her competences and clicks to edit the derived competence. Then, he/she rates the competence and submits.	
Alternative	The actor cancels the operation.	
Pre-conditions	Actor must be logged in.	
Post-conditions	Actor can view a list of the derived competences.	
Assumptions	Actor cannot edit the derived competences of other workers.	

# A.1.2 Competence Derivation: Derivation Graph



Identifier	UC-CCM-CD-DG-2	
Name	Create/delete competence derivation graph	
Description	Actor creates/deletes a competence derivation graph in/from the system. In the former case, the actor specifies the new competence derivation graph properties. For both the action must be submitted to affect the system.	
Actor(s)	Manager	
Priority	Desirable	
	Scenarios	
Main	Actor accesses the competence menu and chooses to create a new competence derivation graph or to remove an existing one. In the former case, the user is redirected to a page where he/she must define the new graph properties and submit the provided data. In the later, the actor selects an existing graph and permanently deletes it from the system.	
Alternative	The actor is viewing a list of existing competence graphs and clicks on the options 'create new' or 'delete'.	
<b>Pre-conditions</b>	Actor must be logged in.	
Post-conditions	Actor can view the derivation graph, in textual or graphical form.	
Assumptions	Actor cannot create a graph identical to an existing one. Similarly, he/she mustn't delete a competence graph that's being edited.	
Identifier	UC-CCM-CD-DG-3	
Name	Edit competence derivation graph	
Description	Actor edits an existing competence derivation graph.	
Actor(s)	Manager	
Priority	Desirable	
Scenarios		
Main	Actor accesses the competence menu and chooses to edit an existing competence derivation graph, upon which he/she is redirected to a page where the graph's properties are exposed and can be updated.	
Alternative	The actor is viewing a competence derivation graph and chooses the option 'edit'.	
Pre-conditions	Actor must be logged in.	
Post-conditions	Actor can view the competence derivation graph properties, in textual or graphical form.	
Assumptions	Actor cannot edit a competence derivation graph that does not exist in the system.	

Identifier	UC-CCM-CD-DG-4	
Name	View competence derivation graph	
Description	Actor views a competence derivation graph and its properties.	
Actor(s)	Manager	
Priority	Desirable	
Scenarios		
Main	Actor accesses the competence menu and chooses to consult a competence derivation graph.	
Alternative	The actor is viewing a list of the competence derivation graphs in the system. Then, he/she selects an item from the list and click on the option 'view details', where the graph's properties are displayed.	
Pre-conditions	Actor must be logged in.	
Post-conditions	Actor can view the competence derivation graph in textual or graphical form.	
Assumptions	Actor cannot view a competence derivation graph that does not exist.	

## A.1.3 Wishlist



Identifier	UC-CCM-WL-2	
Name	View wishlist	
Description	Actor removes a competence from his/her wishlist	
Actor(s)	Worker	
Priority	Desirable	
Scenarios		
Main	The actor accesses the wishlist menu and sees competences in that list. Then he/she clicks on the 'remove competence' option and a list of competences appears. He/she chooses a set of competences to remove and submits.	
Alternative	The actor cancels the operation.	
Pre-conditions	Actor must be logged in.	
Post-conditions	Actor can view his/her wishlist.	
Assumptions	Actor cannot interfere with a wishlist he/she doesn't own.	

# A.1.4 Basic Management



Identifier	UC-CCM-BM-1
Name	Create/delete competence
Description	Actor creates a new/permanently removes a competence in/from the system.
Actor(s)	Manager
Priority	Essential
	Scenarios
Main	In the former, the actor goes to the competence menu and chooses 'create new competence'. Then, a form appears with the main competence properties to edit. The actor fills in the information and submits. In the later, the user goes to the competence menu and chooses to view the list of compe- tences. He/she then selects a set of competences and clicks the 'delete' option.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view competence list.
Identifier	UC-CCM-BM-2
Name	Enable/disable competence
Description	Actor enables/disables a competence in the system without removing it.
Actor(s)	Manager
Priority	Important
	Scenarios
Main	The actor accesses the competence menu and chooses to view a list of the system compe- tences. Aside each competence there is a 'enable/disable' option. The user clicks in that option and enable/disables the competence in the system.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.

Post-conditions	Actor can view competence list.
Identifier	UC-CCM-BM-3
Name	Edit competence
Description	Actor updates competence properties.
Actor(s)	Manager
Priority	Essential
	Scenarios
Main	The actor accesses the competence menu and chooses to view a list of the system compe- tences. Aside each competence there is an 'edit' option. The user clicks in that option and updates the competence properties. Finally, he/she submits the entered values.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view competence details.
Identifier	UC-CCM-BM-4
Name	View competence
Description	Actor consults competence properties.
Actor(s)	Manager
Priority	Essential
Scenarios	
Main	The actor accesses the competence menu and chooses to view a list of the system compe- tences. Aside each competence there is a 'see details' option. The user clicks in that option and views the selected competence's properties.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view competence details.

# A.1.5 Hierarchy



Identifier	UC-CCM-HR-1
Name	Create/delete competence hierarchy
Description	In the former, the actor creates a competence hierarchy, where a competence may have <i>n</i> subcompetences and so on. In the later (delete competence hierarchy), the actor permanently removes a competence hierarchy from the system.
Actor(s)	Manager
Priority	Important
	Scenarios
Main	To create a competence hierarchy, the actor goes to the competence menu and chooses to create a new competence hierarchy. Then, a form appears to define the new hierarchy's main properties, namely its root competence. Finally, he/she submits the data entered. To delete a competence hierarchy, the actor view a list of the existing competence hierarchies, He/she then selects a competence hierarchy and click the option 'delete'; upon user confirmation, the competence hierarchy is permanently removed from the system.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Upon creation of a new competence hierarchy, the actor can view the new competence hier- archy. Upon removal of an existing competence hierarchy, the actor can view the list of remaining competence hierarchies in the system.
Assumptions	The user cannot remove a competence hierarchy that does not exist or one he/she has no permissions over. Additionally, he/she cannot delete a competence hierarchy that's being edited by another user.
Identifier	UC-CCM-HR-2
Name	Edit competence hierarchy
Description	Actor edits an existing competence hierarchy.
Actor(s)	Manager
Priority	Important
	Scenarios
Main	The actor goes to the competence menu and chooses to view the competence hierarchies, upon which a hierarchy list is presented. He/she chooses selects a competence hierarchy and clicks on the 'edit' competence. The competence hierarchy details are presented and the user changes the properties at will. The actor then submits the changes and the competence hier- archy is updated in the system.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view the competence hierarchy.
Assumptions	Actor cannot edit a competence hierarchy he/she has no permissions over.

Identifier	UC-CCM-HR-3
Name	View competence hierarchy
Description	Actor visualizes the complete competence hierarchy, in graphical or textual form.
Actor(s)	Manager
Priority	Important
	Scenarios
Main	The actor accesses the competence menu and chooses to view the competence hierarchies. Then, a list of the existing hierarchies is presented. The user selects one item from the list and clicks on the option 'view details', upon which the chosen hierarchy's properties are shown to the user.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	The competence hierarchy properties are displayed.
Assumptions	The actor can only view a competence hierarchy he/she has permissions over.

# A.1.6 Competence Profile



Identifier	UC-CCM-CP-1
Name	Add/remove competence to worker profile
Description	Actor adds/removes a skill to/from a worker's information details.
Actor(s)	Manager
Priority	Essential
	Scenarios
Main	<u>Add competence</u> : Actor accesses a worker's profile and chooses the option 'add compe- tence'. Then, he/she is redirected to a form where he/she can choose the system competences to add. <u>Remove competence</u> : Actor is viewing the worker's competences. Then, he/she selects a competence a clicks on the 'delete' option'. Upon user confirmation, the competence is permanently removed from the worker's profile.
Alternative	The actor cancels the operation.
<b>Pre-conditions</b>	Actor must be logged in.
Post-conditions	Actor can view the worker's competence profile.
Assumptions	Actor cannot add a competence that already exists in the target profile nor intervene in a profile of his/her supervisor. Similarly, he/she cannot delete a competence he/she that doesn't exist in the target worker profile.
Identifier	UC-CCM-CP-2
Name	View worker's competence profile
Description	Actor view the competences associated to a worker.
Actor(s)	Manager
Priority	Essential
	Scenarios
Main	The actor views a list of workers and selects the option 'see competence profile'. Then, a list of the selected worker's competences is displayed. Each competence has an option 'see details', which allows the actor to see that skill's properties.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view the worker's competence profile.
Assumptions	Actor cannot add a competence that already exists in the target profile nor intervene in a profile of his/her supervisor.

## A.1.7 Competence Grouping



Identifier	UC-CCM-CG-1
Name	Add/remove competence
Description	Actor adds/removes a competence to/from a group.
Actor(s)	Manager
Priority	Essential
	Scenarios
Main	The actor accesses a competence group's properties and chooses the option 'add' or 're- move' competence. In the former, he/she is redirected to a page where he/she can choose the competence to add and submit. In the later, he/she views a list of the group's competences, selects a competence and clicks on the 'delete' option. Upon user confirmation, the compe- tence is permanently removed from the competence group.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor may view competence group.
Assumptions	Actor cannot add a competence that already belongs to another group unless it is in a sub- group.

Identifier	UC-CCM-CG-2
Name	Associate/remove worker with/from competence group
Description	Actor associates or removes a worker to/from a competence group, meaning that he possesses skills in that group.
Actor(s)	Manager
Priority	Essential
	Scenarios
Main	The actor accesses the competence menu and chooses to view competence groups. Then, two possible scenarios can be considered:
	Associate worker to competence group: actor accesses a worker's profiles and clicks the option 'associate to group', where he/she chooses a competence group and submits. Then, the worker is associated to the competences in the selected group.
	<u>Remove worker from competence group</u> : actor accesses a worker's profiles and clicks the option 'associate to group', where he/she chooses a competence group and submits. Then, the worker is associated to the competences in the selected group.
Alternative	The actor consults competence group properties, checks the list of associated workers and edits that list, adding or removing workers to/from that list.
Pre-conditions	Actor must be logged in.
Post-conditions	Worker profile is updated and actor can view competence group.
Assumptions	Actor cannot associate a worker that is already in that group. Similarly, he/she cannot re- move a worker that doesn't belong to that group.
Idontifion	LIC COM CC 2
Nama	Create/delete.competence.group/subgroup
Description	A stor groups a new competence group, subgroup
A stor(s)	Actor creates a new competence group, specifying its properties.
Actor(s)	Econtial
rnorny	Essential
Main	Actor concerns the competence many choosing to view competence groups/subgroups
wiain	Then, two scenarios are possible:
	<u>Create competence group/subgroup</u> : Actor chooses the option 'create group', upon which he/she is redirected to a page where he/she must specify the new group's properties. In the case of a subgroup creation, the 'parent' group must be chosen. <u>Delete competence group/subgroup</u> : Actor views a list of the competence groups or sub- groups of a competence group. Then, he/she selects a group/subgroup and clicks on the option 'delete'. Upon user confirmation, the group/subgroup is permanently removed from the system.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Upon competence group/subgroup creation, the actor can view new competence group prop- erties. In the case of competence group/subgroup removal, the actor views a list of the re- maining competence groups/subgroups.
Assumptions	Actor cannot create a competence group/subgroup equal to an existing one nor remove a competence group/subgroup that is being edited by another user.

Identifier	UC-CCM-CG-4
Name	Edit competence group/subgroup
Description	Actor edits competence group/group properties.
Actor(s)	Manager
Priority	Essential
	Scenarios
Main	Actor accesses the competence menu, chooses a competence group/subgroup and clicks the 'edit' option. Then, the selected group/subgroup properties are shown, which the user can change. Upon submission, the properties are updated in the system.
Alternative	The actor cancels the operation.
<b>Pre-conditions</b>	Actor must be logged in.
Post-conditions	Actor can view competence group/subgroup properties.
Identifier	UC-CCM-CG-5
Name	Give/remove facilitator/coach permissions to evaluate competences in a group
Description	Actor gives/prohibits a facilitator or coach privileges to evaluate workers who have compe- tences in that group/subgroup.
Actor(s)	Manager
Priority	Essential
	Scenarios
Main	Actor accesses the group/subgroup properties and checks the workers associated to that group.
Alternative	The actor cancels the operation.
<b>Pre-conditions</b>	Actor must be logged in.
Post-conditions	Actor can view group properties, including coaches/facilitators allowed in that group. Then, he/she uses the options 'add' or 'remove' to respectively give/remove users permissions to evaluate competences in that group.
Assumptions	Actor cannot give permissions to facilitators/coaches already in that group.
Identifier	UC-CCM-CG-6
Name	View competence group/subgroup
Description	Actor inspects a competence group/subgroup's properties.
Actor(s)	Manager
Priority	Essential
	Scenarios
Main	Actor selects a competence group/subgroup and clicks the option 'see details'. Then, the user is redirected to a page where the selected group/subgroup's properties are displayed.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view group properties, including competences in that group.
Assumptions	Actor cannot remove a competence that is not in that group.

# A.1.8 Import/Export



Identifier	UC-CCM-IE-1
Name	Import/export competence hierarchy
Description	Actor imports/exports a competence hierarchy to/from the system in a specified format.
Actor(s)	Manager
Priority	Essential
	Scenarios
Main	Actor accesses the competence menu and views a list of the existing competence hierarchies. Then, two possible scenarios prevail:
	<u>Import competence hierarchy</u> : Actor clicks on the option 'import hierarchy'. Then, he/she must specify a path to a target file where the competence hierarchy is defined and submit.
	Export competence hierarchy: Actor selects a hierarchy and clicks on the option 'export'. Then, he/she chooses the format to which the hierarchy should be exported and submits.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view the selected competence hierarchy details.

Identifier	UC-CCM-IE-2
Name	Import/export competence evaluation
Description	Actor imports/exports a worker's competence evaluation to/from the system in a specified format.
Actor(s)	Manager
Priority	Essential
	Scenarios
Main	Actor accesses the competence menu and views a list of the existing competence evalua- tions. Then, two possible scenarios prevail:
	<u>Import competence evaluation</u> : Actor clicks on the option 'import evaluation'. Then, he/she must specify a path to a target file where the competence evaluation is defined and submit.
	Export competence evaluation: Actor selects a worker's evaluation and clicks on the option 'export'. Then, he/she chooses the format to which it should be exported and submits.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view the selected evaluation details.

# A.2 Training

# A.2.1 Gap Analysis



Identifier	UC-TRN-GA-1
Name	Compare gap analyses
Description	Actor compares result of two or more gap analyses.
Actor(s)	Worker
Priority	Important
	Scenarios
Main	The actor selects two or more gap analyses. Then, he selects the option 'compare results', configuring its parameters.
Alternative	The actor cancels the operation.
<b>Pre-conditions</b>	Actor must be logged in.
Post-conditions	Actor can view comparison results by text and/or graphics.
Assumptions	Worker can only compare gap analyses he/she has permissions over.
Identifier	UC-TRN-GA-2
Name	Obtain training recommendations from gap analysis
Description	Gain an ordered list of training courses according to the actor's competence profile.
Actor(s)	Worker, Manager, Coach, Facilitator
Priority	Important
	Scenarios
Main	The actor performs a gap analysis, obtaining its results. Then, he/she clicks on the option 'get training recommendations', upon which the system presents a ranked list of trainings recommended for the analysed competence profile, sorted by descending average competence gap.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view an ordered list of courses and access their details.
T1 /*@	
Identifier	UC-1RN-GA-5
Name Demoisticat	Actor actor a new patient of the provincial to between a worker of group) and a worker
Description	Actor gets a perspective of the proximity between a wanted prome and a worker.
Actor(s)	Examiner
Priority	Essential
Main	The actor calents a majorit or group and a worker. Then, he/she clicks on the action 'gap
Main	analysis', upon which the system produces a gap analysis report, both in textual and graphi- cal form.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view comparison results by text and/or graphics.
Assumptions	Actor cannot intervene in a supervisor's profile.

Identifier	UC-TRN-GA-4
Name	Perform gap analysis between worker's and coach's competences
Description	Actor gets a perspective on the level of expertise of a worker and coach concerning a set of skills as to match an adequate coach to a worker.
Actor(s)	Manager
Priority	Essential
	Scenarios
Main	The actor selects a worker and clicks the option 'gap analysis against coach' and the system retrieves a gap analysis considering these two competence profiles.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view comparison results by text and/or graphics.
Assumptions	Actor cannot intervene in a supervisor's profile.
T 1 / 01	
Identifier	UC-TRN-GA-5
Name	Perform gap analysis on a set of workers
Description	Actor gets a perspective of the competence levels of a group of workers.
Actor(s)	Manager, Coach, Facilitator
Priority	Essential
Scenarios	
	Scenarios
Main	Scenarios           From a list of workers, the actor selects <i>n</i> clients and clicks the option 'gap analysis'. The system then performs a gap analysis and retrieves the result, ranking the analysed workers.
Main Alternative	Scenarios           From a list of workers, the actor selects <i>n</i> clients and clicks the option 'gap analysis'. The system then performs a gap analysis and retrieves the result, ranking the analysed workers.           The actor cancels the operation.
Main Alternative Pre-conditions	Scenarios         From a list of workers, the actor selects n clients and clicks the option 'gap analysis'. The system then performs a gap analysis and retrieves the result, ranking the analysed workers.         The actor cancels the operation.         Actor must be logged in.
Main Alternative Pre-conditions Post-conditions	ScenariosFrom a list of workers, the actor selects n clients and clicks the option 'gap analysis'. The system then performs a gap analysis and retrieves the result, ranking the analysed workers.The actor cancels the operation.Actor must be logged in.Actor can see results as a ranked list and access textual/graphical details.
Main Alternative Pre-conditions Post-conditions Assumptions	ScenariosFrom a list of workers, the actor selects n clients and clicks the option 'gap analysis'. The system then performs a gap analysis and retrieves the result, ranking the analysed workers.The actor cancels the operation.Actor must be logged in.Actor can see results as a ranked list and access textual/graphical details.Actor cannot intervene in a supervisor's profile.
Main Alternative Pre-conditions Post-conditions Assumptions	Scenarios         From a list of workers, the actor selects n clients and clicks the option 'gap analysis'. The system then performs a gap analysis and retrieves the result, ranking the analysed workers.         The actor cancels the operation.         Actor must be logged in.         Actor can see results as a ranked list and access textual/graphical details.         Actor cannot intervene in a supervisor's profile.
Main Alternative Pre-conditions Post-conditions Assumptions Identifier	Scenarios         From a list of workers, the actor selects n clients and clicks the option 'gap analysis'. The system then performs a gap analysis and retrieves the result, ranking the analysed workers.         The actor cancels the operation.         Actor must be logged in.         Actor can see results as a ranked list and access textual/graphical details.         Actor cannot intervene in a supervisor's profile.         UC-TRN-GA-7
Main Alternative Pre-conditions Post-conditions Assumptions Identifier Name	Scenarios         From a list of workers, the actor selects n clients and clicks the option 'gap analysis'. The system then performs a gap analysis and retrieves the result, ranking the analysed workers.         The actor cancels the operation.         Actor must be logged in.         Actor can see results as a ranked list and access textual/graphical details.         Actor cannot intervene in a supervisor's profile.         UC-TRN-GA-7         Save gap analysis results
Main Alternative Pre-conditions Post-conditions Assumptions Identifier Name Description	Scenarios         From a list of workers, the actor selects n clients and clicks the option 'gap analysis'. The system then performs a gap analysis and retrieves the result, ranking the analysed workers.         The actor cancels the operation.         Actor must be logged in.         Actor can see results as a ranked list and access textual/graphical details.         Actor cannot intervene in a supervisor's profile.         UC-TRN-GA-7         Save gap analysis results         Actor stores results of gap analysis in the system or downloads them.
Main Alternative Pre-conditions Post-conditions Assumptions Identifier Name Description Actor(s)	Scenarios         From a list of workers, the actor selects n clients and clicks the option 'gap analysis'. The system then performs a gap analysis and retrieves the result, ranking the analysed workers.         The actor cancels the operation.         Actor must be logged in.         Actor can see results as a ranked list and access textual/graphical details.         Actor cannot intervene in a supervisor's profile.         UC-TRN-GA-7         Save gap analysis results         Actor stores results of gap analysis in the system or downloads them.         Worker
Main Main Alternative Pre-conditions Post-conditions Assumptions Identifier Name Description Actor(s) Priority	Scenarios         From a list of workers, the actor selects n clients and clicks the option 'gap analysis'. The system then performs a gap analysis and retrieves the result, ranking the analysed workers.         The actor cancels the operation.         Actor must be logged in.         Actor can see results as a ranked list and access textual/graphical details.         Actor cannot intervene in a supervisor's profile.         UC-TRN-GA-7         Save gap analysis results         Actor stores results of gap analysis in the system or downloads them.         Worker         Desirable
Main Alternative Pre-conditions Post-conditions Assumptions Identifier Name Description Actor(s) Priority	Scenarios From a list of workers, the actor selects <i>n</i> clients and clicks the option 'gap analysis'. The system then performs a gap analysis and retrieves the result, ranking the analysed workers. The actor cancels the operation. Actor must be logged in. Actor can see results as a ranked list and access textual/graphical details. Actor cannot intervene in a supervisor's profile. UC-TRN-GA-7 Save gap analysis results Actor stores results of gap analysis in the system or downloads them. Worker Desirable Scenarios
Main Alternative Pre-conditions Post-conditions Assumptions Identifier Name Description Actor(s) Priority Main	Scenarios         From a list of workers, the actor selects n clients and clicks the option 'gap analysis'. The system then performs a gap analysis and retrieves the result, ranking the analysed workers.         The actor cancels the operation.         Actor must be logged in.         Actor can see results as a ranked list and access textual/graphical details.         Actor cannot intervene in a supervisor's profile.         UC-TRN-GA-7         Save gap analysis results         Actor stores results of gap analysis in the system or downloads them.         Worker         Desirable         The actor performs a gap analysis and clicks the option 'save'. Then, the system stores the gap analysis in the system so the user can access it later.
Main Alternative Pre-conditions Post-conditions Assumptions Identifier Name Description Actor(s) Priority Main Alternative	Scenarios         From a list of workers, the actor selects n clients and clicks the option 'gap analysis'. The system then performs a gap analysis and retrieves the result, ranking the analysed workers.         The actor cancels the operation.         Actor must be logged in.         Actor can see results as a ranked list and access textual/graphical details.         Actor cannot intervene in a supervisor's profile.         UC-TRN-GA-7         Save gap analysis results         Actor stores results of gap analysis in the system or downloads them.         Worker         Desirable         Stenarios         The actor performs a gap analysis and clicks the option 'save'. Then, the system stores the gap analysis in the system so the user can access it later.         The actor cancels the operation.
Main Alternative Pre-conditions Post-conditions Assumptions Identifier Name Description Actor(s) Priority Main Alternative Pre-conditions	Scenarios         From a list of workers, the actor selects n clients and clicks the option 'gap analysis'. The system then performs a gap analysis and retrieves the result, ranking the analysed workers.         The actor cancels the operation.         Actor must be logged in.         Actor can see results as a ranked list and access textual/graphical details.         Actor cannot intervene in a supervisor's profile.         UC-TRN-GA-7         Save gap analysis results         Actor stores results of gap analysis in the system or downloads them.         Worker         Desirable         The actor performs a gap analysis and clicks the option 'save'. Then, the system stores the gap analysis in the system so the user can access it later.         The actor cancels the operation.         Actor must be logged in.

### A.2.2 Evaluation



Identifier	UC-TRN-EV-1
Name	Assess training programme
1 vanie	
Description	Actor qualitatively evaluates a training programme he/she's participated in.
Actor(s)	Worker
Priority	Essential
Scenarios	
Main	Actor accesses the menu 'evaluation' and chooses to evaluate trainings. Then, a list of the trainings he/she's participated appears, where he/she can select one to evaluate. Upon rating and commenting on the programme, the user submits and the information is stored in the system.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Assumptions	Actor must have been a participant of the evaluated programme.

Identifier	UC-TRN-EV-2
Name	Get historic perspective of my own trainings and competences evolution
Description	Actor gets a perspective of his/her own skills evolution.
Actor(s)	Worker
Priority	Essential
Scenarios	
Main	Actor accesses the 'evaluation' menu and chooses to get an historical view. Then, the system presents the actor's skills evolutions both in graphical and textual form, according to a time frame defined by the user.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view results in a graphical and/or textual form.
Identifier	UC-TRN-EV-3
Name	Get historic perspective of worker's trainings and competences evolution
Description	Actor gets a perspective of a worker's skills evolution
Actor(s)	Manager, Coach, Facilitator
Priority	Essential
Scenarios	
Main	Actor accesses the 'evaluation' menu and chooses to get an historical view. Then, the system presents a list of authorized workers to analyse. The actor selects a worker and clicks on the 'historic evolution' option. Then, the selected worker's skills evolution is presented both in graphical and textual form, according to a time frame defined by the user.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view results in a graphical and/or textual form.
Assumptions	Actor cannot view results of workers he/she hasn't permissions over.

## A.2.3 Lookup



Identifier	UC-TRN-LK-1
Name	(search training programmes) By desirable competences
Description	Actor searches for adequate training programmes for a worker according to his/her compe- tences wishlist.
Actor(s)	Manager, Coach, Facilitator
Priority	Essential
Scenarios	
Main	Actor accesses the training menu and chooses to search for available trainings. Then, while configuring the search engine, he/she chooses to filter by desirable competences and submits.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view search results.
Assumptions	Actor cannot act upon workers he/she has no permissions over.
Identifier	UC-TRN-LK-2
-----------------------	--
Name	(search training programmes) By projects
Description	Actor searches for adequate training programmes for a worker according to projects he/she's enrolled in.
Actor(s)	Manager, Coach, Facilitator
Priority	Essential
	Scenarios
Main	Actor accesses the training menu and chooses to search for available trainings. Then, while configuring the search engine, he/she chooses to filter by projects and submits.
Alternative	The actor cancels the operation.
<b>Pre-conditions</b>	Actor must be logged in.
Post-conditions	Actor can view search results.
Assumptions	Actor cannot act upon workers he/she has no permissions over.
T1 (*0*	
Identifier	UC-TRN-LK-3
Name	Search training programmes for a worker
Description	Actor searches for available training programmes.
Actor(s)	Manage, Coach, Facilitator
Priority	Essential
	Scenarios
Main	Actor is viewing a worker's profile. Then, he/she goes to the 'training' menu and chooses to search for available programmes. After that, a ranked list of trainings is presented to the user.
Alternative	The actor cancels the operation.
<b>Pre-conditions</b>	Actor must be logged in.
Post-conditions	Actor can view search results.
Assumptions	Actor cannot act upon workers he/she has no permissions over.
Identifier	
Nama	Search training programmes for my profile
Description	Actor searched for training programmes suitable for his/har own profile
A stor(s)	Worker
Actor(s)	Escential
	Sconorios
Main	The actor is viewing his/her own profile, accesses the 'training' many and chooses to search
	for training programmes. Then, the system presents a ranked list of training programmes according to his/her own competence profile.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view results.

Identifier	UC-TRN-LK-5
Name	Tag/untag desired training programmes
Description	Actor selects training programmes he/she would like to attend.
Actor(s)	Worker
Priority	Important
Scenarios	
Main	The actor views a list of training programmes and tags/untags programmes.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view list of desired training programmes.

## A.3 Coaching

Identifier	UC-COA-1
Name	Send request for coaching
Description	Actor asks the system for a coach.
Actor(s)	Worker
Priority	Important
	Scenarios
Main	Actor accesses the 'coaching' menu and chooses to ask for a coach. Upon submission, the system sends a request for coaching to that actor.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor views the state of the request (pending).
Identifier	UC-COA-2
Name	Send application for coach
Description	Actor requests the system to become a coach.
Actor(s)	Worker
Priority	Important
Scenarios	
Main	Actor accesses the 'coaching' menu and chooses to apply for coach. Upon submission, the system sends a request for coaching.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor views the state of the request (pending).

Identifier	UC-COA-3
Name	Approve/decline coach application
Description	Actor accepts/refuses to turn worker into coach.
Actor(s)	Manager
Priority	Important
	Scenarios
Main	The user accesses the 'coaching' menu and chooses to view pending coaching requests. Then, he/she selects a coaching request from the list presented by the system and clicks on the option 'approve/decline request'.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Identifier	UC-COA-4
Name	Associate/disassociate coach to worker
Description	Actor assigns a coach to a worker.
Actor(s)	Manager
Priority	Important
Scenarios	
Main	The actor accesses the 'coaching' menu and chooses to view current coaches. Then, the system presents a list of the coaches and associated trainees. Then, two scenarios are viable: <u>Associate coach to worker</u> : A list is presented of the pending coaching requests and coaches' availability. Then, the user may select a worker and a coach, selecting the option 'assign coach' and submitting. <u>Disassociate coach to worker</u> : On the list of coaches and associated trainees, the actor may select a coach and one of his/her associated trainees, clicking on the option 'break link' and submitting.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Coach and worker are notified of association or disassociation.
Assumptions	Worker must be hierarchically inferior to actor.
Identifier	UC-COA-5
Name	Send coaching request to coach
Description	Manager asks coach to accept a new trainee.
Actor(s)	Manager
Priority	Important
	Scenarios
Main	Actor accesses the 'coaching' menu and chooses to view current available coaches. Then, he/she selects a coach from the list presented by the system and clicks on the option 'send coaching request', associating that request to a trainee.
Alternative	The actor cancels the operation.
<b>Pre-conditions</b>	Actor must be logged in.

Post-conditions	Actor views the state of the request (pending).
Identifier	UC-COA-6
Name	Point competence needs
Description	Actor enumerates competences gaps he/she finds relevant.
Actor(s)	Trainee
Priority	Important
	Scenarios
Main	Actor accesses the 'coaching' menu and chooses to report on competence needs. Then, he/she fills out a form where he/she can fill out a description with his/her opinion. Addition- ally, he/she may select competences from a list and, per competence selected, tag as 'needed competence' and rate that need.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
T1 /···	NG 004 7
Identifier	
Name	Add/remove competence to/from 'needs list'
Description	Actor adds a competence that he/she's believes to be important to acquire.
Actor(s)	Trainee
Priority	Important
	Scenarios
Main	Actor goes to 'coaching' menu and chooses to view the 'needs list'. Then, a list of the needed competences is presented to the user. The user can then choose to add a new competence to that list or remove an existing one by clicking the options 'add' or 'remove', respectively.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view 'needs list'.
T1 (10)	
Identifier	
Name	(Periodically) appraise trainee
Description	Actor regularly evaluates trainee performance, needs and satisfaction levels.
Actor(s)	Coach
Priority	Essential
	Scenarios
Main	Actor goes to the 'coaching' menu and chooses to appraise trainees. Then, the system pre- sents a list of the coach's trainees. The actor selects a trainee and click on the option 'ap- praise'. Then, he/she fills out a form with the evaluation data, considering the trainee's per- formance, skill needs and satisfaction levels.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view his evaluation details.

Identifier	UC-COA-9
Name	Assess risks
Description	Actor describes risks associated to trainee's competence levels,.
Actor(s)	Coach
Priority	Essential
	Scenarios
Main	The actor is executing a trainee's appraisal and fills out a field regarding the risks associated to the trainee's skills and self-evaluation.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view his evaluation details.
Identifier	UC-COA-10
Name	Report competence needs
Description	Actor describes what competences he/she thinks his/her trainee should acquire or improve.
Actor(s)	Coach
Priority	Essential
	Scenarios
Main	The actor is executing a trainee's appraisal and fills out a field regarding the competence needs of that trainee. This information is directed to the manager and facilitator.
Alternative	The actor cancels the operation.
<b>Pre-conditions</b>	Actor must be logged in.
Post-conditions	Actor can view his evaluation details.
<b>T</b> 1 (10)	
Identifier	
Name	Accept/decline coaching request
Description	Actor accepts/declines request to be coach of another user.
Actor(s)	Coach
Priority	Important
	Scenarios
Main	Actor goes to 'coaching' menu and chooses to view pending coaching requests. The system then presents a list of the pending requests and associated trainees and target competences. The user selects a list item relative to a request and clicks on the 'accept' or 'decline' option. Upon submission, the manager is notified of the coach's decision.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor can view list of current trainees.

### A.4 Talent

# A.4.1 Sniffing



Identifier	UC-TAL-SN-1	
Name	Search for workers with a set of competences	
Description	Actor searches for workers with an <i>ad hoc</i> set of competences.	
Actor(s)	Manager	
Priority	Essential	
	Scenarios	
Main	Actor accesses the 'talent' menu and chooses to search for workers. In the search engine configuration, he/she specifies that the search should be filtered by set of competences and creates or loads an existing set of competences. The actor then submits the data and the system retrieves the target data.	
Alternative	The actor cancels the operation.	
Pre-conditions	Actor must be logged in.	
Post-conditions	Actor views ranked list of workers.	
Identifier	UC-TAL-SN-3	
Name	Filter by project competences	
Description	Actor searches for workers concerning a project needed competences.	
Actor(s)	Manager	
Priority	Essential	
	Scenarios	
Main	Actor accesses the 'talent' menu and chooses to search for workers. In the search engine configuration, he/she specifies that the search should be targeted for a project. Then, he/she chooses the target project and chooses to filter by project competences. Upon submission, the search results are presented to the user.	

Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor views ranked list of workers.
Identifier	UC-TAL-SN-4
Name	Filter by training and associated deadlines
Description	Actor searches for workers concerning a project needed competences vs. Related training programmes.
Actor(s)	Manager
Priority	Important
Scenarios	
Main	Actor accesses the 'talent' menu and chooses to search for workers. In the search engine configuration, he/she specifies that the search should be targeted for a project. Then, he/she chooses the target project and chooses to filter by trainings. Upon submission, the search results are presented to the user.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor views ranked list of workers.

### A.4.2 Scouting



Identifier	UC-TAL-SC-2
Name	Generate job interviews from competence set
Description	Actor obtains a job interview from a custom set of competences, which can be one of the following: <i>Ad hoc</i> set of competences Project competences Competence group
Actor(s)	Manager, Examiner
Priority	Important
	Scenarios
Main	Actor goes to the 'talent' menu and chooses to generate a job interview. Then, he/she fills a form with the target interview parameters, namely the base set of competences. There, the user may define an <i>ad hoc</i> set of competences or load them from a project or competence group. Upon submission, the generated job interview is shown.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor views generated interview.
I.J	
Nama	VC-TAL-SC-5
Description	Actor creates a job interview entering questions and associating them with required compa
Description	tences.
Actor(s)	Manager, Examiner
Priority	Essential
Scenarios	
Main	Actor goes to the 'talent' menu and chooses to manually create a job interview. Then, he/she fills a form with the target interview parameters, namely the questions and associated competences. Upon submission, the created job interview is shown.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor views generated interview.
I.J	
Nama	Sava ish interview
Description	Actor stores job interview in the system or downloads it
Actor(s)	Manager Examiner
Priority	Important
Troncy	Scenarios
Main	Actor is viewing or editing a job interview and clicks on the 'save interview' option
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.

Identifier	UC-TAL-SC-5
Name	Edit job interview
Description	Actor edits job interview details (questions, competences,)
Actor(s)	Manager, Examiner
Priority	Essential
	Scenarios
Main	Actor is viewing a list of job interviews and clicks on the 'edit' option. Then, the selected job interview's properties and questions are shown, which the user may change. Upon sub-mission, the job interview is updated in the system.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor views updated interview.
Identifier	UC-TAL-SC-6
Name	Delete job interview
Description	Actor permanently deletes a job interview from the system.
Actor(s)	Manager
Priority	Essential
	Scenarios
Main	Actor is viewing a list of job interviews and clicks on the 'delete' option. Upon user confir- mation the job interview is permanently removed from the system.
Alternative	The actor cancels the operation.
<b>Pre-conditions</b>	Actor must be logged in.
Post-conditions	Actor views list of existing job interviews.
<b>T</b> 1. (10)	
Identifier	UC-TAL-SC-7
Name	Obtain gap analysis comparing target profile with examinee's profile
Description	Actor gets a perspective of the gap between the examinee and the requited competences' levels of expertise.
Actor(s)	Manager
Priority	Essential
	Scenarios
Main	Actor is viewing a list of completed/executed job interviews and clicks on the 'diagnosis' option. Upon submission, the system performs a gap analysis between the target profile and the examinee's profile and shows its results, both in textual and graphical form. Additionally, the system presents an estimate of the examinee's 'adequacy' to the target competence profile.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor views results as text and/or graphics.

Identifier	UC-TAL-SC-8
Name	Edit examiner data
Description	Actor edits his/her interview fields.
Actor(s)	Examiner
Priority	Essential
	Scenarios
Main	A job interview has been started by the actor and he/she edits his/her interview fields. Upon submission, the information is stored in the system.
Alternative	The actor cancels the operation.
Pre-conditions	Actor must be logged in.
Post-conditions	Actor views updated fields.
Identifier	UC-TAL-SC-9
Name	Edit examinee data
Description	Actor edits his/her interview fields.
Actor(s)	Examiner
Priority	Essential
Scenarios	
	Scenarios
Main	Scenarios           A job interview has been started by the assigned Examiner. The actor edits his/her interview fields and, upon submission, the information is stored in the system.
Main Alternative	Scenarios         A job interview has been started by the assigned Examiner. The actor edits his/her interview fields and, upon submission, the information is stored in the system.         The actor cancels the operation.
Main Alternative Pre-conditions	ScenariosA job interview has been started by the assigned Examiner. The actor edits his/her interview fields and, upon submission, the information is stored in the system.The actor cancels the operation.Actor must be logged in.
Main Alternative Pre-conditions Post-conditions	ScenariosA job interview has been started by the assigned Examiner. The actor edits his/her interview fields and, upon submission, the information is stored in the system.The actor cancels the operation.Actor must be logged in.Actor views updated fields.
Main Alternative Pre-conditions Post-conditions	Scenarios         A job interview has been started by the assigned Examiner. The actor edits his/her interview fields and, upon submission, the information is stored in the system.         The actor cancels the operation.         Actor must be logged in.         Actor views updated fields.
Main Alternative Pre-conditions Post-conditions Identifier	Scenarios         A job interview has been started by the assigned Examiner. The actor edits his/her interview fields and, upon submission, the information is stored in the system.         The actor cancels the operation.         Actor must be logged in.         Actor views updated fields.         UC-TAL-SC-10
Main Alternative Pre-conditions Post-conditions Identifier Name	ScenariosA job interview has been started by the assigned Examiner. The actor edits his/her interview fields and, upon submission, the information is stored in the system.The actor cancels the operation.Actor must be logged in.Actor views updated fields.UC-TAL-SC-10Start/end job interview
Main Alternative Pre-conditions Post-conditions Identifier Name Description	ScenariosA job interview has been started by the assigned Examiner. The actor edits his/her interview fields and, upon submission, the information is stored in the system.The actor cancels the operation.Actor must be logged in.Actor views updated fields.UC-TAL-SC-10Start/end job interviewActor initiates/terminates a job interview process
Main Main Alternative Pre-conditions Post-conditions Identifier Name Description Actor(s)	ScenariosA job interview has been started by the assigned Examiner. The actor edits his/her interview fields and, upon submission, the information is stored in the system.The actor cancels the operation.Actor must be logged in.Actor views updated fields.UC-TAL-SC-10Start/end job interviewActor initiates/terminates a job interview processExaminer
Main Main Alternative Pre-conditions Post-conditions Identifier Name Description Actor(s) Priority	ScenariosA job interview has been started by the assigned Examiner. The actor edits his/her interview fields and, upon submission, the information is stored in the system.The actor cancels the operation.Actor must be logged in.Actor views updated fields.UC-TAL-SC-10Start/end job interviewActor initiates/terminates a job interview processExaminerDesirable
Main Alternative Pre-conditions Post-conditions Identifier Name Description Actor(s) Priority	ScenariosA job interview has been started by the assigned Examiner. The actor edits his/her interview fields and, upon submission, the information is stored in the system.The actor cancels the operation.Actor must be logged in.Actor views updated fields.UC-TAL-SC-10Start/end job interviewActor initiates/terminates a job interview processExaminerDesirableScenarios
Main Main Alternative Pre-conditions Post-conditions Identifier Name Description Actor(s) Priority Main	ScenariosA job interview has been started by the assigned Examiner. The actor edits his/her interview fields and, upon submission, the information is stored in the system.The actor cancels the operation.Actor must be logged in.Actor views updated fields.UC-TAL-SC-10Start/end job interviewActor initiates/terminates a job interview processExaminerDesirableThe actor selects a job interview an clicks on the option 'start' or 'end'.
Main Main Alternative Pre-conditions Post-conditions Identifier Identifier Actor(s) Priority Main Alternative	ScenariosA job interview has been started by the assigned Examiner. The actor edits his/her interview fields and, upon submission, the information is stored in the system.The actor cancels the operation.Actor must be logged in.Actor views updated fields.UC-TAL-SC-10Start/end job interviewActor initiates/terminates a job interview processExaminerDesirableThe actor selects a job interview an clicks on the option 'start' or 'end'.The actor cancels the operation.
Main Main Alternative Pre-conditions Post-conditions Identifier Name Description Actor(s) Priority Main Alternative Pre-conditions	Scenarios         A job interview has been started by the assigned Examiner. The actor edits his/her interview fields and, upon submission, the information is stored in the system.         The actor cancels the operation.         Actor must be logged in.         Actor views updated fields.         UC-TAL-SC-10         Start/end job interview         Actor initiates/terminates a job interview process         Examiner         Desirable         The actor selects a job interview an clicks on the option 'start' or 'end'.         The actor cancels the operation.         Actor must be logged in.

### A.5 Dashboard

Identifier	UC-DSB-1
Name	Configure dashboard
Description	Actor customizes dashboard organization.
Actor(s)	Worker
Priority	Essential
Scenarios	
Main	Actor accesses the top menu and chooses 'configuration'. Then, the dashboard configuration options are presented to the user.
Alternative	The actor cancels the operation.
<b>Pre-conditions</b>	Actor must be logged in.
Post-conditions	Actor views updated dashboard.

### A.6 Gadgets

#### A.6.1 Alerts

Identifier	UC-GAD-AL-2										
Name	Enable/disable alert type										
Description	Actor enables the notification of a given alert type.										
Actor(s)	Worker										
Priority	Essential										
Scenarios											
Main	Actor has entered the gadget configuration panel, where he/she can see a list of the enabled and disabled alert types. An alert type can be enabled by checking the respective item and disabled by deselecting it. Upon submission, the gadget updates the alert types displayed.										
Alternative	The actor cancels the operation.										
Pre-conditions	Actor must be logged in.										
Identifier	UC-GAD-AL-3										
Name	Ignore/accept alert										
Description	Actor accepts/rejects gadget notification on dashboard.										
Actor(s)	Worker										
Priority	Essential										
	Scenarios										
Main	While the actor is logged in the system, the alert gadget shows alerts, which he/she may accept or ignore, by clicking its respective option.										
Alternative	The actor cancels the operation.										
Pre-conditions	Actor must be logged in.										
Post-conditions	Actor views updated dashboard.										

#### A.7 Social Network

Identifier	UC-SNT-1									
Name	Send friend request									
Description	Actor asks a user to become his/her friend.									
Actor(s)	Worker									
Priority	Essential									
	Scenarios									
Main	Actor is viewing another worker's public profile and clicks on the option 'send friend request'. Then, a friend request is sent to the target worker.									
Alternative	The actor cancels the operation.									
Pre-conditions	Actor must be logged in.									
Identifier	UC-SNT-3									
Name	Edit profile information									
Description	Actor updates information on his public profile.									
Actor(s)	Worker									
Priority	Essential									
Scenarios										
Main	Actor views his/her own profile information and clicks on the option 'edit'. Then, his/her profile properties are show, which the user can edit. Upon submission, the data is stored in the system and the worker profile is updated.									
Alternative	The actor cancels the operation.									
Pre-conditions	Actor must be logged in.									
Post-conditions	Actor views updated profile.									
Identifier	UC-SNT-4									
Name	Manage messages									
Description	Actor manages the messages exchanged between him/her and the other social network work- ers.									
Actor(s)	Important									
Priority	Essential									
	Scenarios									
Main	Actor accesses the message menu and is capable of viewing, creating and deleting messages from the system.									
Alternative	The actor cancels the operation.									
Pre-conditions	Actor must be logged in.									

Identifier	UC-SNT-5												
Name	Send message												
Description	Actor sends message to another worker.												
Actor(s)	Worker												
Priority	Important												
	Scenarios												
Main	Actor composes a new message, selects the target worker and clicks on the 'send' option.												
Alternative	The actor cancels the operation.												
Pre-conditions	Actor must be logged in.												
Identifier	UC-SNT-6												
Name	Manage friends group												
Description	Actor creates, removes or views group of friends.												
Actor(s)	Worker												
Priority	riority Important												
Scenarios													
Main	Actor chooses the 'view friends groups' option in the top menu and a list of his/her groups is presented. Then, he may choose to create a new group or remove an existing one.												
Alternative	The actor cancels the operation.												
Pre-conditions	Actor must be logged in.												
Post-conditions	Actor views updated dashboard.												
Identifier	UC-SN1=7												
Name	Accept/decline group member proposal												
Description	Actor allows/forbids a user from entering his/her group.												
Actor(s)	Worker												
Priority	Desirable												
	Scenarios												
Main	Actor chooses is viewing a friends group properties and chooses to view pending group member proposals. Then, from the list of requests presented, he/she chooses list items and clicks on the options 'accept' or 'decline'.												
Alternative	The actor cancels the operation.												
<b>Pre-conditions</b>	Actor must be logged in.												

Identifier	UC-SNT-8										
Name	Sign up for group										
Description	Actor requests to group owner to join in.										
Actor(s)	Worker										
Priority	Desirable										
Scenarios											
Main	Actor is viewing a friends group's properties he/she doesn't belong to and clicks on the option 'sign up for group'. Then, a request to join the group is sent to the group owner.										
Alternative	The actor cancels the operation.										
Pre-conditions	Actor must be logged in.										
Post-conditions	Actor views request state (pending).										
Identifier	UC-SNT-9										
Name	View friend's information										
Description	Actor accesses friend public profile information.										
Actor(s)	Worker										
Priority	Essential										
	Scenarios										
Main	Actor is viewing a list of the workers in the social network, and selects a user. Then, by clicking on the option 'see profile', the selected worker's public profile is show to the actor.										
Alternative	The actor cancels the operation.										
Pre-conditions	Actor must be logged in.										

# A.8 Security/Privacy

Identifier	UC-SEC-1									
Name	Manage user accounts									
Description	Actor creates, edits or deletes/disables a user account.									
Actor(s)	Administrator									
Priority	Essential									
Scenarios										
Main	Actor accesses the 'coaching' menu and chooses the 'manage user accounts' options, upon which a list of the system's users is presented. Then, the actor may create, edit or de-									
	lete/disable a user account.									
Alternative	Interview of the system subject of									

Identifier	UC-SEC-2										
Name	Turn worker into coach or disable coach permissions										
Description	Actor changes a worker privileges to coach.										
Actor(s)	Manager										
Priority	Essential										
	Scenarios										
Main	The user accesses the 'coaching' menu and chooses to view current coaches. A list of the system coaches is presented as well as a list of the enabled coaches. Then, two scenarios are possible:										
	<u>'urn worker into coach</u> : Actor selects a worker from the list of pending coaching applica- ons, selects a pending application and clicks on the 'approve' option.										
	Disable coach permissions: Actor views the list of current system coaches, selects a coach and clicks on the option 'disable'.										
Alternative	The actor cancels the operation.										
Pre-conditions	Actor must be logged in.										
Identifier	UC-SEC-4										
Name	gister account										
Description	Actor inserts data for a new account creation.										
Actor(s)	Visitor										
Priority	Essential										
	Scenarios										
Main	Actor views the system information that is publicly available, clicks on the 'register account' option. Then, he/she fills the new account information and submits.										
Alternative	The actor cancels the operation.										
Identifier	UC-SEC-5										
Name	Login										
Description	Actor submits his/her credentials in the system.										
Actor(s)	Visitor										
Priority	Essential										
	Scenarios										
Main	Actor views the system information that is publicly available, clicks on the 'login' option. Then, he/she inserts his/her credentials and submits.										
Alternative	The actor cancels the operation.										

Identifier	UC-SEC-6									
Name	Logout									
Description	Actor exists the system,									
Actor(s) (Authenticated) user										
Priority Essential										
Scenarios										
Main	Actor clicks on the 'logout' option in the top menu.									
Alternative	The actor cancels the operation.									
Pre-conditions Actor must be logged in.										
Post-conditions	Actor views the publicly available system information.									

# **Annex B** : Comparative Technology **Analysis Details**

In section 0., the results of a preliminary survey regarding the Web-based MVC frameworks and database management systems. This annex presents the intermediate comparative matrixes.

Language	Framework Name	Homepage	Has MVC in its core or extensions?	Maturity	Latest (Stable) Release	Language Version	(Expected) Learning Curve	Open Source/Free IDE?	Supports Web Services?	Scalability	Extensibility	Support Community	Cross-platform?	Development Status
ActionScript		http://www.actionscript.org/		High	3.0 (July 2006)		Medium	Yes (Netbeans with FlexBeans plugin)				Large	Yes	Active
	FlashMVC	http://www.flashmvc.com/	Core	Low	2.01				No	Medium		Reduced	Yes	Active
	PureMVC	http://purenwc.org/	Core	Medium	2.0.4 (August 2008)	3.0		Yes	No	High	High	Large	Yes	Active
ASP		http://www.actionscript.org/		High	3.0 (no further are planned)		Medium	Yes				Large	Yes	Active
	ASP Xtreme Evolution (AXE)	http://zend.lojcomm.com.br/goodie s/asp-xtreme-evolution/	Core	Low		3.0		Yes	No			Reduced	Yes	Active
C++		http://www.cplusplus.com/		Very High			Low	Yes	Yes, with gSOAP			Large	Yes	Active
	cppcms	http://cppcms.sourceforge.net/wi kipp/en/page/main	Core	Low	0.0.4 (February 2009)		Low		No			Reduced	No	Active
	Wt - Web toolkit	http://www.webtoolkit.eu/wt	Embedded in each individual component	Low	2.99.5 (September 2009)		Low		No			Reduced	Yes	Active
ColdFusion		http://www.adobe.com/products/ coldfusion/		High	9.0		High	Yes (CFEclipse)				Large	Windows, Linux, UNIX, Macintosh	Active
	Coldbox	http://www.coldboxframework.co m/	Core	Medium	2.6.4						High	Medium	Windows, Mac OS X, UNIX, Linux	
	FW/1	http://fw1.riaforge.org/	Core	Low	0.7.1	8.0.1 / 9.0					Medium	Reduced		Active
	Mach-II	http://www.mach-ii.com/	Core	Medium	1.6.1 (March 2009)				No	High	High (filter and plugin	Medium	Yes	Active
	Model-Glue	http://www.model-glue.com/	Core	Medium					No			Reduced		Active
	PureMVC	http://puremvc.org/	Core	Medium	2.0.4 (August 2008)	3.0		Yes	No	High	High	Large		Active
	Switchboard	http://switchboard.riaforge.org/	Core	Low	0.9.7					Medium		Reduced	No (Windows and Mac OS X)	Active

Language	Framework Name	Homepage	Has MVC in its core or extensions?	Maturity	Latest (Stable) Release	Language Version	(Expected) Learning Curve	Open Source/Free IDE?	Supports Web Services?	Scalability	Extensibility	Support Community	Cross-platform?	Development Status
ColdFusion		http://www.adobe.com/products/col dfusion/		High	9.0		High	Yes (CFEclipse)				Large	Windows, Linux, UNIX, Macintosh	Active
	Coldbox	http://www.coldboxframework.com/	Core	Medium	2.6.4						High	Medium	Windows, Mac OS X, UNIX, Linux	
	FW/1	http://fw1.riaforge.org/	Core	Low	0.7.1	8.0.1 / 9.0					Medium	Reduced		Active
	Mach-II	http://www.mach-ii.com/	Core	Medium	1.6.1 (March 2009)				No	High	High (filter and plugin architecture)	Medium	Yes	Active
	Model-Glue	http://www.model-glue.com/	Core	Medium					No			Reduced		Active
	PureMVC	http://puremvc.org/	Core	Medium	2.0.4 (August 2008)	3.0		Yes	No	High	High	Large		Active
	Switchboard	http://switchboard.riaforge.org/	Core	Low	0.9.7					Medium		Reduced	No (Windows and Mac OS X)	Active
Groovy		http://groovy.codehaus.org/		Medium	1.6		High	Yes (e.g. Netbeans )	Yes (Groovy SOAP)			Medium	Yes	Active
	Grails	http://grails.org/	Core	Medium	1.1.1		High		Yes (Xfire plugin)	Medium	High	Large		Active
	Griffon	http://griffon.codehaus.org/	Core	Low	0.2		High					Reduced		Active

Language	Framework Name	Homepage	Has MVC in its core or extensions?	Maturity	Latest (Stable) Release	Language Version	(Expected) Learning Curve	Open Source/Free IDE?	Supports Web Services?	Scalability	Extensibility	Support Community	Cross-platform?	Development Status
Java		http://java.sun.com/		Very high	Java Standard Edition 6		Low	Yes	Yes			Very large	Yes	Active
	Apache Coccon	http://www.araneaframework.org/	Core	Medium	1.2.2 (June 2009)							Reduced	Yes	Active
	Apache Sling	http://sling.apache.org/site/index.html	Core	High	5.0 (May 2009)							Medium	Yes	Active
	Apache Struts	http://struts.apache.org/	Core	High	2.1.8 (September							Large	Yes	Active
	Apache Tapestry	http://tapestry.apache.org/	Core	Very high	5.1.0.5 (May 2009)					High	High	Large	Yes	Active
	Apache Wicket	http://wicket.apache.org/	Core	Medium	1.4.3 (October 2009)							Large	Yes	Active
	Apple WebObjects	http://developer.apple.com/tools/webobj ects/	Core	Very high	5.4.3 (September							Large	No (Mac OS X only)	Active
	Aranea	http://cocoon.apache.org/	Core	High	2.2.0					High	High	Medium	Yes	Active
	Induction	http://www.inductionframework.org/	Core	Medium	1.3.0b (November							Reduced	Yes	Active
	JSF (JavaServer Faces)	http://java.sun.com/javaee/javaserverfac es/	Core	High	2.0					Medium	High	Large	Yes	Active
	PureMVC	http://puremvc.org/	Core	Medium	2.0.4 (August 2008)	3.0		Yes	No	High	High	Large		Active
	Sofia	http://www.salmonllc.com/Jsp/salmonwe b/SalmonPage.jsp?page_id=32	Core	High	2.3							Large	Yes	Active
	Spring MVC	http://www.springsource.org/	Core	High	2.5.6 (October 2008)							Medium	Yes	Active
	Stripes	http://www.stripesframework.org/	Core	Medium	1.5.2 (October 2009)							Reduced	Yes	Active
	WebWork	http://www.opensymphony.com/webwor k/	Core	Medium	2.2.6 (July 2007)							Medium	Yes	Active
JavaScript				Medium	1.8.1 (1009)		Low	Yes	Yes			Very large	Yes	Active
	JavascriptMVC	http://www.javascriptmvc.com/	Core	Medium	2.0							Reduced	Yes	Active
	PureMVC	http://puremvc.org/	Core	Medium	2.0.4 (August 2008)	3.0				High	High	Large	Yes	Active
	SproutCore	http://www.sproutcore.com/	Core	Low	1.0					High	High	Medium	Yes	Active
	Wavemaker	http://www.wavemaker.com/	Core	Very high	5.0.0 (April 2009)					Reduced		Medium	Windows, Mac OS X, Linux	Active

Language	Framework Name	Homepage	Has MVC in its core or extensions?	Maturity	Latest (Stable) Release	Language Version	(Expected) Learning Curve	Open Source/Free IDE?	Supports Web Services?	Scalability	Extensibility	Support Community	Cross-platform?	Development Status
.NET		http://msdn.microsoft.com/en- gb/netframework/default.aspx		Very High	3.5 SP1 (August 2008)		Low	Yes	Yes			Verylarge	No (Windows only)	Active
	ASP .NET MVC	http://www.asp.net/mVC/	Core	Medium	2.0 (November 2009)					Medium	High	Large		Active
	Monorail	http://www.castleproject.org/monorail/i ndex.html	Core	Low	1.0 RC (September							Reduced		Active
	Pure MVC	http://puremvc.org/	Core	Medium	2.0.4 (August 2008)	3.0				High	High	Large	Yes	Active
	Spring Framework	http://www.springsource.org/	Core	High	2.5.6 (October 2008)							Medium	Yes	Active
	Vici MVC (former	http://viciproject.com/wiki/projects/mvc /home	Core	Medium	2.0.1	2.0 or higher						Medium		Active
Perl		http://www.perl.org/		Very high	5.10.1 (August 2009)		Medium	Yes	Yes			Verylarge	Yes	Active
	Catalyst	http://www.catalystframework.org/	Core	Very high	5.800007 (June 2009)				Support for RESTful services	High	High	Very large (books	Yes	Active
	CGI::Applicatio n	http://cgi-app.org/	Core	Medium	1.04 (February 2009)						High	Reduced	Yes	Active
	Gantry	http://www.usegantry.org/	Core									Reduced	Yes	Active
	Maypole	http://maypole.perl.org/	Core	Medium	2.10							Reduced	Yes	Active
	Solstice	http://solstice.eplt.washington.edu/	Core	Very high	6.2				Support for RESTful services			Reduced	Yes	Active

Language	Framework Name	Homepage	Has MVC in its core or extensions?	Maturity	Latest (Stable) Release	Language Version	(Expected) Learning Curve	Open Source/Free IDE?	Supports Web Services?	Scalability	Extensibility	Support Community	Cross-platform?	Development Status
РНР		http://php.net/index.php		High	September 2009	5.3.0	Low	Yes	Yes			Large	Yes	Active
	Agavi	http://www.agavi.org/	Core	Medium	1.0.2 RC2 (May 2009)	PHP 5.2.0 or higher			Yes	High	High	Medium	Yes	Active
	Akelos PHP	http://www.akelos.org/	Core	Low	0.9 (February 2009)	PHP4 or PHP5					High	Medium	Yes	Active
	atomik	http://www.atomikframework.com/	Extensions (internal	Medium	2.1 (February 2009)	5 and higher				Reduced	High	Medium	Yes	Active
	CakePHP	http://cakephp.org/	Core	Medium	1.2.5 (September				RESTful Web Services	High	High	Large	Yes	Active
	Codelgniter	http://codeigniter.com/	Core	Medium	1.7.2 (September	5.3.0			Yes	High	High	Large	Yes	Active
	EuropaPHP	http://europaphp.org/	Core	Medium	1.0.2	5.x						Reduced	Yes	Active
	FLOW3	http://flow3.typo3.org/	Core	Low	1.0.0 alpha	5.3.0			Support for RESTful Web			Large	Yes	Active
	FUSE	http://www.phpfuse.net/	Core	Medium	1.2 (August 2008)	5 or higher						Medium	Yes	Active
	PHP on TRAX	http://www.phpontrax.com/	Core	Low	0.16.0 (September							Medium	Yes	Active
	PRADO	http://www.xisc.com/	Core	High	3.1.6 (July 2009)	5.x						Medium	Yes	Active
	Samstyle PHP	http://code.google.com/p/samstyle-php- framework	Core	Medium	1.2.11							Reduced	Yes	Active
	Symfony	http://www.symfony-project.org/	Core	Medium	1.2.9 (September 2009)	5.x			SOAP web services			Medium	Yes	Active
	Vork	http://www.vork.us/	Core	Medium	1.14	5.x						Medium	Yes	Active
	Yii PHP	http://www.yiiframework.com/	Component	Medium	1.0.10	5.x			Yes	High	High	Reduced	Yes	Active
	Zend	http://framework.zend.com/	Component	Medium	1.9	5.x			Supports RESTful web		High	Large	Yes	Active
	ZNF	http://www.zeronotice.org/	Core	Low	0.7.10	5.x				High	High	Reduced	Yes	Active
	Zoop	http://zoopframework.com/	Core	Medium	1.5.0	4.x or 5.x						Medium	Yes	Active

Language	Framework Name	Homepage	Has MVC in its core or extensions?	Maturity	Latest (Stable) Release	Language Version	(Expected) Learning Curve	Open Source/Free IDE?	Supports Web Services?	Scalability	Extensibility	Support Community	Cross-platform?	Development Status
Python		http://www.python.org/		High	3.1 (July 2009)		Low	Yes	Yes			Very large	Yes	Active
	Django	http://www.djangoproject.com/	Core	Medium	1.1.1 (October 2009)				Supports RESTful web	High	High	Medium	Yes	Active
	PureMVC	http://puremvc.org/	Core	Medium	2.0.4 (August 2008)	3.0				High	High	Large	Yes	Active
	TurboGears	http://www.turbogears.org/	Core	High	2.0.3 (August 2009)				Yes	High	High	Medium	Yes	Active
	web2py	http://www.web2py.com/	Core	Medium	1.2.3.4 (November					High	Hihj	Medium	Yes	Active
Ruby		http://www.ruby-lang.org/en/		Medium	1.9.1		Medium	Yes	Yes			Very large	Yes	Active
	Merb (to be merged with	http://www.merbivore.com/	Core	Medium	1.0.12				Support for RESTful web			Medium	Yes	Active
	Monkeybars	http://monkeybars.rubyforge.org/	Core, though not purely MVC	Medium	1.0.1							Medium	Yes	Active
	PureMVC for Ruby	http://puremvc.org/	Core	Medium	2.0.4 (August 2008)	3.0				High	High	Large	Yes	Active
	Ramaze	http://ramaze.net/	Core	Medium	1.8							Large	Yes	Active
	Ruby on Rails	http://rubyonrails.org/	Core	High	2.3.4 (September 2009)				Support for RESTful web services		High	Very large (books published)	Yes	Active

# **Annex C** : **Prototype Code Samples**

In section 7.2, several prototype deployment details are discussed. Hence, this section presents the relevant code samples.

iKnow.Auth/UserAuth.cs

```
DirectoryEntry entry = new DirectoryEntry("LDAP://" + ldap_server_name, user-
name, pass);
```

iKnow.Database/Structures/CompetenceItem.cs

```
namespace iKnow.Database.Structures
{
    public class CompetenceItem : ICompetenceItem
    {
        private readonly int _competence_id;
        private readonly string _name_c;
private readonly string _description;
        public CompetenceItem(int competence id,
                                  string name_c,
                                  string description,
                                  int weight,
                                  DateTime date_creation,
                                  DateTime last_update,
                                  bool is derivate,
                                  int rating_scale_id,
                                  bool is enabled,
                                  int hierarchy id,
                                  int parent id)
         {
             competence id = competence id;
             _name_c = name_c;
             _description = description;
             ...
         }
        public CompetenceItem() { }
        public int competence_id { get { return _competence_id; } }
        public string name_c { get { return _name_c; } }
public string description { get { return _description; } }
    }
```

```
private readonly IDbProvider dbProvider;
```

```
_dbProvider = DbProviderFactory.GetDbProvider("Npgsql-2.0");
```

iKnow.Database/Services/DatabaseService.cs

iKnow.Database/Structures/CompetenceItemCallback.cs

```
namespace iKnow.Database.Structures
{
    internal class CompetenceItemCallback : IRowCallback
    {
        private IList<ICompetenceItem> _result = new List<ICompetenceItem>();
        public IList<ICompetenceItem> Result
            get { return _result; }
        }
        # region implementation of IRowCallback
        public void ProcessRow(IDataReader reader)
          int competence id = reader.IsDBNull(0) ? 0 : reader.GetInt32(0);
          string _name_c = reader.IsDBNull(1) ? null : reader.GetString(1);
          string _description = reader.IsDBNull(2) ? null :
reader.GetString(\overline{2});
          ICompetenceItem item = new CompetenceItem( competence id,
                                                         _name c,
                                                         _description,
                                                        _weight,
                                                        _date_creation,
_last_update,
                                                         _is_derivate,
                                                         _rating_scale id,
                                                         _is_enabled,
                                                         _hierarchy id,
                                                         _parent_id);
            if ( result == null)
            {
                _result = new List<ICompetenceItem>();
            result.Add(item);
        # endregion
    }
```

```
mamespace iKnow.Contract.Structures
{
    public interface ICompetenceItem
    {
        int competence_id { get; }
        string name_c { get; }
        string description { get; }
        ...
    }
```

#### competence-central-nav.sitemap

CompetenceController.cs

```
[Authorize]
...
[PopulateSiteMap(SiteMapName = "competence-central-nav", ViewDataKey = "compe-
tence-central-nav")]
public ActionResult CompetenceCentral()
{
    localSettings();
    return View();
}
```

#### CompetenceCentral.aspx

```
<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">
       <h2>Competence Central</h2>
      <asp:Table BorderWidth="0" runat="server">
            <asp:TableRow>
                <asp:TableCell Width="250" VerticalAlign="Top" Border-
Style="None">
                    <div>
                        <%= Html.Telerik().PanelBar()
                           .Name("PanelBar")
                           .BindTo("competence-central-nav")
                        응>
                    </div>
                </asp:TableCell>
            </asp:TableRow>
       </asp:Table>
    </asp:Content>
```

```
"""
<asp:Content ID="Content1" ContentPlaceHolderID="TitleContent" runat="server">
Manage Competence Hierarchies
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">
</asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">
</asp:Content="server">
</asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">
</asp:Content="server">
</asp:Content="server">
</asp:Content="server">
</asp:Content="server">
</asp:Content="server">
</asp:Content ID="Content8" ContentPlaceHolderID="CustomScripts"
</asp:Content ID="Content8" ContentPlaceHolderID="CustomScripts"
</asp:Content int="server">
</asp:Content int="server"
</asp:Content int="server">
</asp:Content int="server"
</asp:Content int="server"
</asp:Content int="server"
</asp:Content int="server"
</asp:Content int="server"
</asp:Content int="server"
</asp:Conten
```

hierarchyManagement.js

```
$(document).ready(function() {
    $('#jqgGrid').jqGrid({
        //url from wich data should be requested
       url: '/Competence/GetHierarchyGridData/',
       //type of data
       datatype: 'json',
        //url access method type
       mtype: 'POST',
        //columns names
       colNames: ['ID', 'Name', 'Description', 'Date Creation', 'Last Up-
date', 'Actions'],
       //columns model
        colModel: [
                   { name: 'hierarchy id', index: 'hierarchy id', align: 'cen-
ter', width:'50', key:true },
                            { name: 'name h', index: 'name h', align: 'left',
width: '100' },
                            { name: 'description', index: 'description',
align: 'center' },
                            { name: 'date creation', index: 'date creation',
align: 'center', width:'90'
                            },
                            { name: 'last updated', index: 'last updated',
align: 'center', width:'80'
                            },
                            { name: 'actions', index: 'actions', align: 'cen-
ter', width:'210', formatter: actionsFormatter }
                            ],
        //pager for grid
       pager: $('#jqgpGrid'),
        //number of rows per page
        rowNum: 10,
        //initial sorting column
       sortname: 'name h',
        //initial sorting direction
        sortorder: 'asc',
        //grid width
       width: 'auto',
        //grid height
       height: 'auto'
   });
});
```

```
public ActionResult GetHierarchyGridData(string sidx, string sord, int page,
int rows)
        {
            // test data
            IList<IHierarchyItem> testList = service.getHierarchies();
            // Getting total records count from repository
            int totalRecords = testList.Count();
            // Preparing anonymous variable with json data
            var resultData = new
            {
                // total pages count
                total = (int)Math.Ceiling((float)totalRecords / (float)rows),
                page = page, // page number
                records = totalRecords, // total records count
                rows = (from dt in testList
                        select new
                        // ['ID', 'Name', 'Description', 'Date Creation',
'Last Update', 'Actions'],
                        {
                            id = dt.hierarchy_id, // row id
                            cell = new object[] { // table of cells values
                                     dt.hierarchy_id,
                                     dt.name h,
                                     dt.description,
                                     dt.date creation.ToShortDateString(),
                                     dt.last updated.ToShortDateString(),
                                     dt.hierarchy_id
                                   }
                        }
                    ).ToArray()
            };
            // returning json data
            return Json(resultData);
            }
```

iKnow.Database/Services/DatabaseService.cs

```
public IList<IHierarchyItem> getHierarchies()
{
    IDbParameters parameters = _adoTemplate.CreateDbParameters();
    string procedure = string.Format(ResourceDB.getHierarchies, _db_schema);
    Structures.HierarchyItemCallback callback = new Struc-
tures.HierarchyItemCallback();
    _adoTemplate.QueryWithRowCallback(System.Data.CommandType.Text, procedure,
callback, parameters);
    return callback.Result;
}
```