

***TRS**

Tecnologia, Redes e Sociedade

e-planning | networks | e-learning | e-government

Relatório Interno TRS 03/2014

Título

Análise de um sistema de *backoffice* de Ensino a Distância para a Universidade Católica de Angola

Autor(es)

Paulo Francisco António, UFP
Luis Borges Gouveia, UFP

Mês, Ano

Junho, 2014

Local de presença Web

<http://tecnologiaredesesociedade.wordpress.com>

Repositório de trabalho científico *trs

<http://bdigital.ufp.pt/handle/10284/3787>

Universidade Fernando Pessoa
Praça 9 de Abril, 349
4249-004 Porto, Portugal

Análise de um sistema de *backoffice* de Ensino a Distância para a Universidade Católica de Angola

Paulo Francisco António, Luis Borges Gouveia

Resumo

O rápido e crescente desenvolvimento de recursos técnicos de comunicação e de informação digital tem possibilitado cada vez mais romper com as barreiras da distância, das dificuldades de acesso à educação e dos problemas de aprendizagem de quem “estuda individualmente, mas não isolado e sozinho” e pretende investir na aquisição de conhecimento especializado.

O presente Relatório tem como objectivo a Análise de Sistema do *BackOffice* da Plataforma de Ensino à Distância para a Universidade Católica de Angola, apresentando a descrição das técnicas e ferramentas utilizadas na implementação deste sistema.

O BackOffice do Sistema de Ensino a Distância visa colmatar as dificuldades e constrangimentos que muitos alunos têm em relação à frequência do Ensino Presencial. Muitos dos alunos têm limitações de tempo e muitos afazeres complementares que impedem a frequência do regime presencial. Deste modo, propõe-se o estabelecimento de uma plataforma que ajudará todas as pessoas que residem fora de Luanda, a terem também acesso a um Ensino com as mesmas exigências de rigor e qualidade, mas com as questões de locomoção ao campus universitário resolvidas por este Sistema, desde que tenham acesso à Internet para utilizar o Sistema *everywhere, anywhere e anytime*.

Palavras-chave: Ensino Superior, *BackOffice*, Análise de Sistemas, Universidade Católica de Angola, Internet, Ensino a Distância.

1. Introdução

Com o passar do tempo, aumenta o número de pessoas com o desejo de frequentar um Curso Superior nas mais variadas áreas do saber. Porém, muitos têm limitações de tempo devido ao trabalho e não só (por exemplo, custos de deslocação, exigências logísticas e

mesmo por opção).

A rápida e constante evolução tecnológica desperta-nos para uma nova realidade com impacte na forma como é organizado o processo de ensino e aprendizagem no contexto do ensino superior: o *e-learning*. O recurso crescente à mediação digital proporciona novas formas de intermediar p ensino e permite modalidades alternativas para a sua exploração, como é o caso de complemento ao ensino presencial ou o EaD – Ensino a Distância, ou ainda uma versão mista que mistura distância e ensino presencial (*blended learning* ou *b-learning*). Estas novas propostas devem ser encaradas de forma activa e galvanizadoras para o acesso à educação e para modernizar o ensino superior.

O avanço registado nas tecnologias de informação e na Informática permite que actualmente seja possível aceder a informação em tempo real, independentemente do local onde o utilizador se encontre (Oliveira, et al, 2003). Este avanço nas tecnologias permite atribuir um considerável nível de colaboração e interactividade através do ambiente virtual entre o Professor-Aluno-Universidade.

A possibilidade de um cidadão aceder à informação em qualquer lugar e a qualquer hora é tornada possível pelo uso de computadores e ou dispositivos móveis (sejam eles telemóveis de terceira geração, *Smartphones*, PDA (*Personal Digital Assistant*), *Pocket PC* ou mesmo computadores portáteis) que estão cada vez mais acessíveis, atravessando assim um processo de massificação, que se deve ao rápido avanço tecnológico verificado nas áreas de hardware, software e comunicações. A preocupação em utilizar recursos tecnologicamente avançados para melhorar as condições de trabalho, de investigação, de ensino e de aprendizagem constitui um permanente desafio de grande parte das instituições de ensino superior.¹

¹ ALVES, Paulo, et al. Estudo Sobre a Utilização das Ferramentas do Ambiente Sakai numa Instituição de Ensino Superior. Bragança: Ipb, [s.d.]. Disponível em: <
https://bibliotecadigital.ipb.pt/bitstream/10198/8763/1/CL2013Tcai_cap11.pdf> Acesso em: 17 Jun. 2014

Em Angola, não regista de até ao momento que uma Universidade tenha implementado o Ensino a Distância, talvez devido a razões que se prendem com a falta de competência técnica, falta de interesse e investimento, ou por falta de arriscar inovar num contexto do ensino superior, ainda muito marcado por abordagens mais tradicionais. Porém, e para inverter o quadro, a Universidade Católica quer ser a pioneira deste Sistema em Angola, razão pela qual se implementa o *BackOffice* do Sistema de Ensino a Distância (EaD).

Uma das formas de permitir que os estudantes tenham acesso aos recursos educacionais à distância, consiste na implementação de Sistema de Gestão de Ensino e Aprendizagem.

É nesta perspectiva que se implementa o *BackOffice* da UCAN para posterior integração com o *FrontOffice*, normalmente associado com o recurso a uma plataforma digital de gestão de aprendizagem e conteúdos e que gere todo os processos de natureza pedagógica e de interacção (LMS – *Learning Management Systems* ou .LCMS – *Learning Management Content Systems*, também muitas vezes referidos como VLE – *Virtual Learning Environments*).

1.1 Objectivo do relatório

O presente Relatório visa descrever a implementação do *BackOffice* do Sistema de Ensino a Distância da UCAN, tendo em conta os procedimentos da área de Análise de Sistemas, contemplando os procedimentos a seguir desde a CONCEPÇÃO à IMPLEMENTAÇÃO do Sistema no Ambiente de Produção.

Para a materialização do objectivo proposto, adoptou-se a implementação da plataforma de gestão de aprendizagem SAKAI² (ambiente colaborativo de aprendizagem) por ser uma

² O Projecto Sakai surgiu em 2004 através de um consórcio entre várias universidades: University of Michigan, Indiana University, MIT, Stanford, uPortal Consortium e *Open Knowledge Initiative* (OKI), com o objectivo de desenvolverem um conjunto de ferramentas colaborativas destinadas a docentes, alunos e investigadores, para o apoio ao ensino e à investigação. É um projecto *Open Source* que visa a partilha de aplicações entre instituições de ensino superior.

plataforma poderosa, com muitas funcionalidades e sem custos de licenciamento – *open source* e ainda permitir a sua programação, desenvolvimento e adaptação – código aberto. Os ambientes colaborativos de aprendizagem surgem como uma forma de dar resposta aos desafios que a educação enfrenta. Esses desafios implicam uma mudança de estratégias, nomeadamente: aumento de capacidade, eficiência e eficácia, melhoria da acessibilidade e ênfase nos recursos e na personalização.³

1.2 O BackOffice

BackOffice é um conceito, utilizado na área da informática para denominar um conjunto de funcionalidades que fazem parte do sistema, mas que não são visíveis por todos os utilizadores do sistema e que contemplam o suporte de gestão de dados e de administração e controle de informação, para realizar a actividade de valor de uma organização (por exemplo, o controle de alunos e suas inscrições, a afectação a aulas e controle de sumários e outros procedimentos realizados de forma administrativa que permitem o contexto de ensino e aprendizagem e as aulas numa universidade).

Assim, o BackOffice, alberga na generalidade, funcionalidades e procedimentos que permitem ao utilizador fazer uma melhor gestão do seu *FrontOffice*, isto é, do que está visível também para os restantes utilizadores, sem a necessidade de recorrer a um especialista na área da informática, sempre que precisa de realizar alguma alteração aos conteúdos da aplicação (Martins, 2013).

³ *Ibidem*

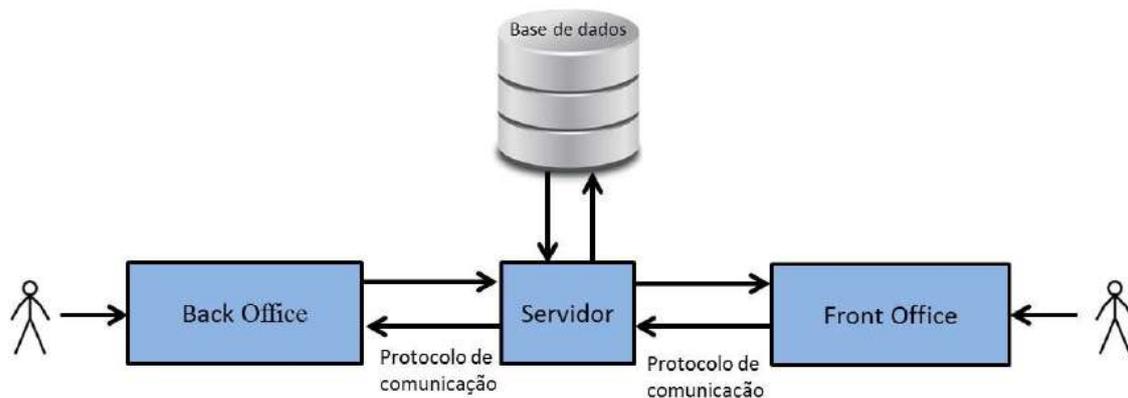


Figura 1 - Estrutura de uma plataforma web com BackOffice⁴

Nos dias de hoje há uma necessidade crescente de que o *BackOffice* seja parte integrante de qualquer aplicação, pois esta componente fornece ao utilizador uma maior liberdade e flexibilidade na configuração das suas aplicações. Em resposta a esta mesma necessidade, as plataformas de *BackOffice* têm sofrido uma rápida evolução, tanto a nível da eficiência operacional, como a nível da automatização de processos e da autonomia do utilizador⁵ para efectuar configurações cada vez mais personalizadas (Tavares, 2012).

1.3 Plataforma Web modular

Uma plataforma web modular é constituída por um conjunto de módulos que funcionam de forma independente, mas que quando combinados entres si aumentam o potencial da plataforma, e facilmente cobrem as necessidades de um alargado grupo de utilizadores. Actualmente, este tipo de plataformas já existe nas mais diversas áreas, graças à sua fácil adaptação e composição (Henderson, 2006). Entre as vantagens deste tipo de abordagem destacam-se as seguintes:

- Reutilização dos módulos em várias plataformas;

⁴ **Fonte:** *BackOffice para plataformas web de Gestão de Conferências Científicas: Desenvolvimento e Avaliação*, p. 4 [Adaptado]

⁵ Estamos a falar nesse caso do utilizador com permissões ou privilégios de administrador, pois só este é que consegue alterar as configurações.

- Independência dos módulos;
- Adaptabilidade da plataforma;
- Desenvolvimento mais rápido da plataforma.

2. Descrição da análise do sistema

2.1 Requisitos

Contexto

Para a definição dos requisitos do *BackOffice* do Sistema de Ensino à Distância analisaram-se os vários sistemas de ensino a distância *open source* (Moodle, Sakai, ELIS) no que concerne as funcionalidades existentes comparando com as necessidades da Instituição em relação ao Processo Ensino-Aprendizagem.

Todo sistema quando é concebido deve cumprir um conjunto de requisitos para que produza os resultados esperados. O correcto levantamento desses requisitos é indispensável para o bom andamento do projecto e produção dos resultados esperados.

Assim sendo, elegeu-se a plataforma SAKAI para a implementação do *BackOffice* em questão

Requisitos Funcionais

Especificam as funções realizadas pelo sistema, isto é, aquilo que a aplicação (sistema) permite fazer.

Os requisitos funcionais que o *Backoffice* do sistema de Ensino a Distância serão organizados em treze (13) módulos para a sua melhor gestão e organização.

#GERIR ÁREA PESSOAL DO ALUNO

RF1. Adicionar Área Pessoal

RF2. Adicionar Editar Área de Informação Pessoal

RF3. Adicionar Área de Informação Básica

RF4. Adicionar Área de Informação de Contacto

RF5. Adicionar Área de Informação Académica

RF6. Cadastrar Aluno

#GERIR UNIDADES CURRICULARES

RF7. Adicionar Unidade Curricular

RF8. Adicionar Área do Conteúdo Programático

RF9. Adicionar Área de Módulos

R10. Adicionar Área de Cacifo Digital

#GERIR CURSOS

RF11. Adicionar Curso

#GERIR TURMAS

RF12. Adicionar Turmas

#GERIR AULA VIRTUAL

RF13. Criar Aula Virtual

RF14. Criar Sala Virtual

RF15. Adicionar Módulo de Video Conferência

#GERIR FÓRUNS

RF16. Adicionar Fórum

RF17. Adicionar Configurações do Fórum

#GERIR UTILIZADORES E ACESSOS

RF18. Adicionar Área de Cadastro de Utilizador.

RF19. Cadastrar Utilizador

RF20. Adicionar Grupo de Utilizadores

RF20.1. Adicionar Permissões e Papéis

RF21. Adicionar Logout

#GERIR RECURSOS

RF22. Adicionar Workspaces

RF23. Adicionar Opções para os Recursos na Área

#GERIR MENSAGENS

RF24. Adicionar Criar Mensagens

RF25. Adicionar Mensagens Recebidas

RF26. Adicionar Mensagens Eliminadas

RF27. Adicionar Mensagens Enviadas

RF28. Adicionar Opções de Configuração de Envio

.

#GERIR CHATS

RF29. Adicionar Área de Chat

#GERIR ANÚNCIOS

RF30. Criar Anúncios

#GERIR AVALIAÇÃO

RF31. Criar Avaliação

#GERIR PROFESSOR

RF32. Adicionar Área do Professor

RF33. Cadastrar Professor

Requisitos não Funcionais

Os requisitos não funcionais estão relacionados com os requisitos de desempenho e outros atributos de qualidade, requisitos lógicos de dados e restrições ao design do sistema.

O bom levantamento dos requisitos não funcionais de desempenho é crucial, especialmente para aplicações Web. Para esse tipo de aplicação, o grande desafio é que a forma aleatória com que os utilizadores acedem o sistema, gera uma grande diferença entre a carga de trabalho média e os picos de uso (levando a considerações de balanceamento, mesmo no desenho das respostas tecnológicas do sistema a conceber).

Assim sendo, os requisitos não funcionais que o sistema deve contemplar são:

- Capacidade: o sistema deve permitir 2000 acessos/minuto (cerca de 34 acessos simultâneos, por segundo);
- Integração: o sistema deve ser desenvolvido em módulos permitindo a integração a fim de constituir uma aplicação completa;
- Persistência: a informação processada pela aplicação deverá ser armazenada numa base de dados, em tecnologia MySQL com suporte a múltiplas transacções;
- Escalabilidade: a aplicação deve adaptar-se ao futuro crescimento da instituição e da demanda do número de utilizadores;

- Interoperabilidade: o sistema deve comunicar de forma transparente (ou o mais próximo disso) com outro sistema (semelhante ou não);
- Disponibilidade: o sistema deve estar disponível 24h por dia/7 dias por semana;
- Segurança: Garantir segurança no acesso web através do protocolo https. Encriptar a senha dos utilizadores. O sistema não deve permitir acessos não autorizados. O sistema deve ainda garantir a não repudição;
- O Sistema será hospedado na infra-estrutura da Universidade com recurso a uma plataforma Linux
- O sistema deverá armazenar logs de todas as transacções dos utilizadores e das alterações feitas pelos administradores, a fim de garantir a fé do sistema e arbitrar disputas entre utilizadores futuros.
- O sistema deve continuar funcionando em casos críticos, mesmo que mais lentamente.
- Metáforas de Interface: A interface do sistema deve ser baseada na plataforma web com código Java, com o intuito de tornar o conteúdo dinâmico e prover a conexão com a base de dados.

2.2 Diagramas Entidade-Relação

A técnica de modelação mais usada para a especificação dos modelos de dados é a proporcionada pela abordagem Entidade-Relação (Modelo E-R). O modelo de dados é representado graficamente através de um diagrama de Entidade-Relação (E-R).

O modelo Entidade-Relação toma uma perspectiva que tem por base que o mundo real é formado por um conjunto de objectos chamados entidades e pelo conjunto de relacionamentos entre esses objectos que proporcionam uma relação bem definida entre essas entidades.

3. Especificações técnicas

Segundo Martins (2013, p. 11) *“o software têm o seu desenvolvimento assente num conjunto de requisitos que podem ser representados por modelos UML. O processo de criação de um modelo consiste na abstracção dos aspetos essenciais do problema abordado e permite a estruturação da aplicação a desenvolver, sem contemplar ainda os detalhes sobre a implementação. Pode-se considerar um modelo UML como o passo intermédio entre o enunciado do problema e o desenvolvimento da solução. Estes modelos são úteis pois têm um grau de legibilidade elevado, assim como uma representação gráfica clara e que facilita uma compreensão do problema a tratar. O UML disponibiliza vários tipos de diagramas para representar um modelo”*. No contexto do presente trabalho, são apresentados os diagramas de casos de uso.

3.1 Diagramas de Casos de Uso

Diagrama de casos de uso, define-se como sendo uma sequência de eventos, que descrevem as interações entre um actor e um sistema desde a identificação do pré-requisito até à concretização da acção pretendida. Cada um dos casos de uso apresentado no diagrama, é uma unidade discreta da interacção entre um utilizador e o sistema que produz um resultado observável.

Descrição dos Casos de Uso

Caso de Uso: (RF1) Adicionar Área Pessoal

Actores: Administrador

Descrição: O Sistema deve permitir o cadastro dos dados do estudante (criação de perfil), ou seja, da informação pessoal, informação básica, informação de contacto

e informação académica. Este é caso de uso é também responsável pela realização da matrícula e de todo processo relacionado a efectividade académica.

Caso de Uso: (RF2) Adicionar Área de Informação Pessoal

Actores: Administrador

Descrição: O sistema deve permitir a inserção da área relacionada com o cadastro da informação pessoal do aluno.

Caso de Uso: (RF3) Adicionar Área de Informação Básica

Actores: Administrador

Descrição: O sistema deve permitir a inserção da área relacionada ao cadastro da informação básica do aluno.

Caso de Uso: (RF4) Adicionar Área de Informação de Contacto

Actores: Administrador

Descrição: O sistema deve permitir a inserção da área relacionada ao cadastro da informação de contacto de aluno.

Caso de Uso: (RF5) Adicionar Área de Informação Académica

Actores: Administrador

Descrição: O sistema deve permitir a inserção da área relacionada ao cadastro da informação académica do aluno.

Caso de Uso: (RF6) Cadastrar Alunos

Actores: Administrador

Descrição: O sistema deve permitir cadastrar todas informações sobre os alunos, como: nome, número de matrícula, data de matrícula, ano.

Caso de Uso: (RF7) Adicionar Unidade Curricular

Actores: Administrador

Descrição: O sistema deve permitir cadastrar todas informações sobre as unidades curriculares como: o nome da unidade curricular, a sigla e o ciclo de estudos em que pertence.

Caso de Uso: (RF8) Adicionar Área do Conteúdo Programático

Actores: Administrador

Descrição: O sistema deve permitir adicionar a área em que será preenchido o conteúdo programático da unidade curricular.

Caso de Uso: (RF9) Adicionar Área de Módulos

Actores: Administrador

Descrição: O sistema deve permitir adicionar a área em que serão criados os módulos da unidade curricular.

Caso de Uso: (RF10) Adicionar Área de Cacifo Digital

Actores: Administrador

Descrição: O sistema deve permitir adicionar a área em que serão feitos os uploads dos trabalhos da unidade curricular.

Caso de Uso: (RF11) Adicionar Curso

Actores: Administrador

Descrição: O sistema deve permitir adicionar todas informações sobre os cursos como: nome, sigla e ciclo de estudo.

Caso de Uso: (RF12) Adicionar Turmas

Actores: Administrador

Descrição: O sistema deve permitir a criação de turmas e todas informações relacionadas a este procedimento como: nome da turma e o curso atribuído.

Caso de Uso: (RF13) Criar Aula Virtual

Actores: Administrador

Descrição: O Sistema deve permitir a criação de aulas (sessões síncronas) contendo informações como: nome da sessão, status, data de início e fim e o criador da sessão.

Caso de Uso: (RF14) Criar Sala Virtual

Actores: Administrador

Descrição: O Sistema deve permitir a criação de uma área para a sala virtual de molde a permitir a criação das aulas virtuais.

Caso de Uso: (RF15) Adicionar Módulo de Video Conferência

Actores: Administrador

Descrição: O Sistema deve permitir a implementação de um módulo (plugin) de video conferência para a realização da sessão síncrona.

Caso de Uso: (RF16) Adicionar Fórum

Actores: Administrador

Descrição: O Sistema deve permitir a implementação ou criação de fóruns de discussão para os mais variados temas relacionados às unidades curriculares.

Caso de Uso: (RF17) Adicionar Configurações do Fórum

Actores: Administrador

Descrição: O Sistema deve permitir a definição das configurações do fórum para um maior controlo.

Caso de Uso: (RF18) Adicionar Área de Cadastro de Utilizador.

Actores: Administrador

Descrição: O Sistema deve permitir adicionar a área de cadastro de utilizador com um formulário para preenchimento de todas informações para o efeito.

Caso de Uso: (RF19) Cadastrar Utilizador

Actores: Administrador

Descrição: O Sistema deve permitir cadastrar todas as informações sobre o cadastro do utilizador como: nome de utilizador, senha, hora de login, hora de logout e grupo

Caso de Uso: (RF20) Adicionar Grupo de Utilizadores

Actores: Administrador

Descrição: O Sistema deve permitir cadastrar todas as informações sobre os grupos ou perfis de utilizadores.

Caso de Uso: (RF20.1) Adicionar Permissões e Papéis

Actores: Administrador

Descrição: O Sistema deve permitir cadastrar permissões e perfis para os grupos.

Caso de Uso: (RF21) Adicionar Logout

Actores: Administrador

Descrição: O Sistema deve permitir adicionar uma área de *logout* da aplicação.

Caso de Uso: (RF22) Adicionar Workspaces

Actores: Administrador

Descrição: O Sistema deve permitir a criação de *workspaces*, ou seja, a criação de recursos (documentos, ficheiros ou links) para cada uma das unidades curriculares com os seguintes parâmetros: título, acesso, criador, tamanho e modificação. Deve ainda permitir a criação de pastas para organizar os documentos/ficheiros em categoria.

Caso de Uso: (RF23). Adicionar Opções para os Recursos na Área

Actores: Administrador

Descrição: O Sistema deve permitir adicionar uma área de opções para os recursos existentes em cada área.

Caso de Uso: (RF24) Adicionar Criar Mensagens

Actores: Administrador

Descrição: O sistema deve permitir adicionar a opção de criar mensagens contendo os campos necessários para a composição de uma mensagem tais como: remetente, destinatário, assunto e a área do corpo da mensagem.

Caso de Uso: (RF25) Adicionar Mensagens Recebidas

Actores: Administrador

Descrição: O sistema deve permitir adicionar a área de mensagens recebidas para organizar todas as mensagens desta categoria.

Caso de Uso: (RF26) Adicionar Mensagens Eliminadas

Actores: Administrador

Descrição: O sistema deve permitir adicionar a área de mensagens eliminadas para organizar todas as mensagens desta categoria.

Caso de Uso: (RF27) Adicionar Mensagens Enviadas

Actores: Administrador

Descrição: O sistema deve permitir adicionar a área de mensagens enviadas para organizar todas as mensagens desta categoria.

Caso de Uso: (RF28) Adicionar Opções de Configuração de Envio

Actores: Administrador

Descrição: O sistema deve permitir adicionar a área de configuração das opções de envio das mensagens como: reencaminhamento automático e endereço de email para reencaminhar.

Caso de Uso: (RF29). Adicionar Área de Chat

Actores: Administrador

Descrição: O sistema deve permitir a criação de área de *chat* para cada uma das unidades curriculares, contendo elementos como: área de texto e opções relacionadas ao envio da mensagem do chat.

Caso de Uso: (RF30) Criar Anúncios

Actores: Administrador

Descrição: O Sistema deve permitir a criação/inserção de anúncios das unidades curriculares contendo informações como: assunto, data, autor, área (unidade curricular), data de início e data de fim.

Caso de Uso: (RF31) Criar Avaliação

Actores: Administrador

Descrição: O Sistema deve permitir a criação de testes ou provas *online* com questões de tipo múltipla escolha, verdadeiro-falso e questões de desenvolvimento tendo o tempo do teste definido com a respectiva correcção automática configurada de molde que o sistema gere a nota final.

Caso de Uso: (RF32) Adicionar Área do Professor

Actores: Administrador

Descrição: O sistema deve permitir a inserção da área relacionada com o cadastro da informação do Professor.

Caso de Uso: (RF33) Cadastrar Professor

Actores: Administrador

Descrição: O sistema deve permitir cadastrar todas informações sobre os professores como: nome, grau académico, categoria e unidade curricular que lecciona.

Diagramas de casos de uso

O Diagrama de Caso de Uso descreve a funcionalidade e os utilizadores de um sistema partindo de uma perspectiva de fora para dentro, isto é, interna ao sistema a desenvolver. Este diagrama é utilizado para mostrar os relacionamentos entre actores que empregam o sistema e os casos de uso utilizados por eles e permitindo de igual forma, expressar e/ou possibilitar o relacionamento entre casos de uso.

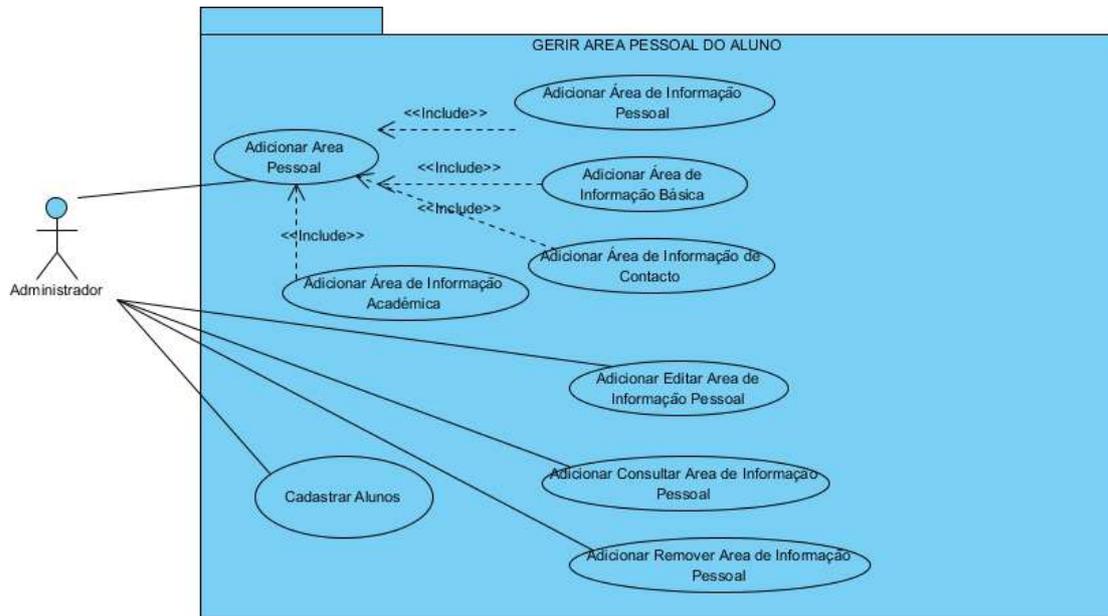


Figura 3 - Diagrama de Casos caso de uso gerir área pessoal do aluno

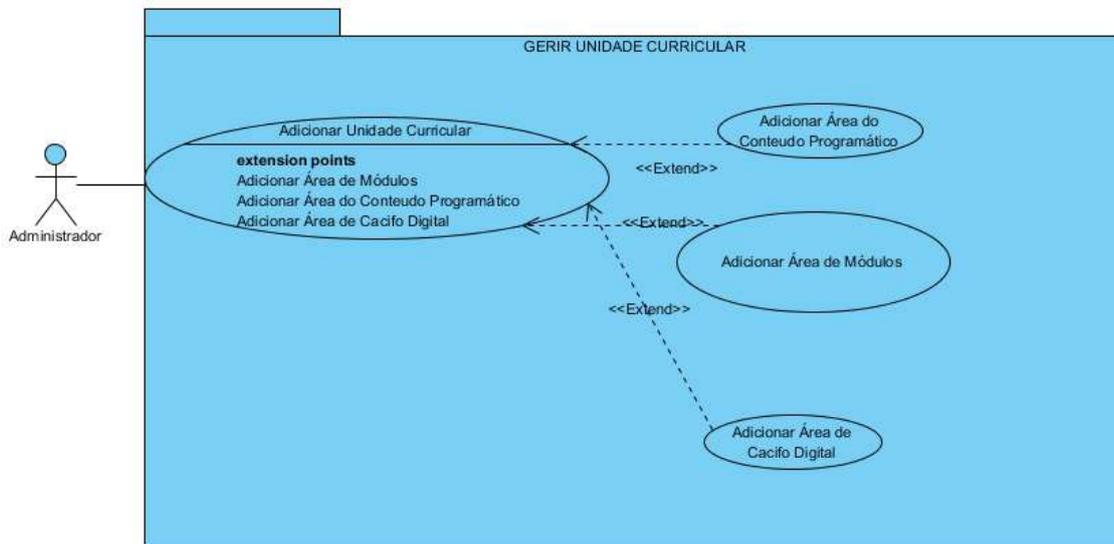


Figura 3 - Diagrama de caso de uso gerir unidade curricular

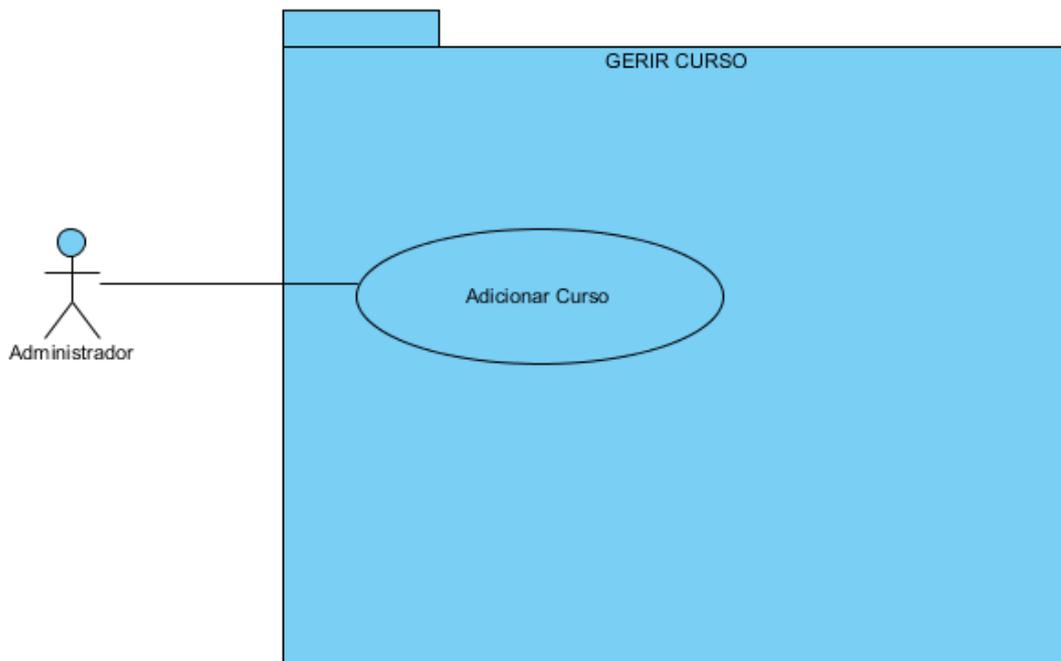


Figura 4 - Diagrama de caso de uso gerir curso

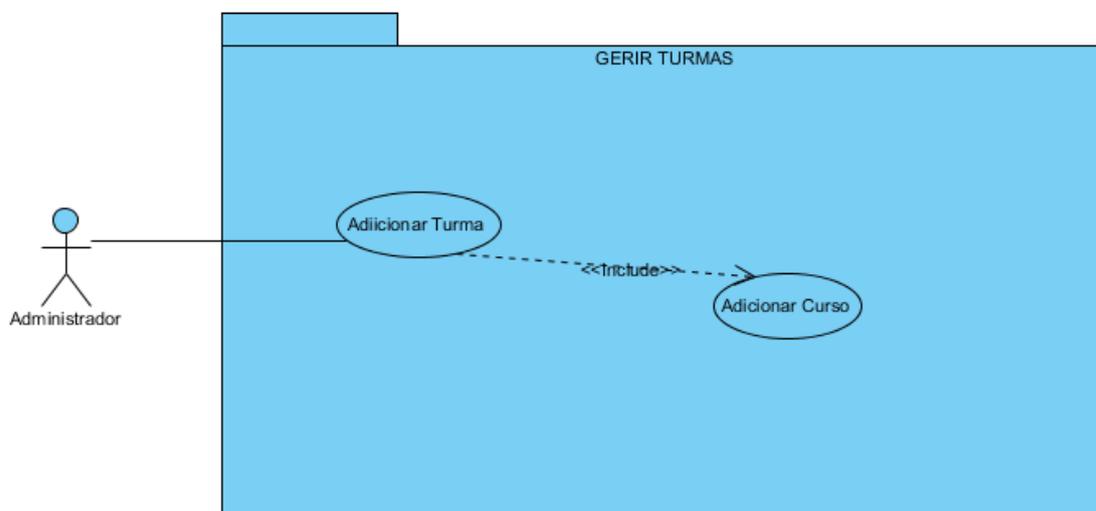


Figura 5 - Diagrama de caso de uso gerir turma

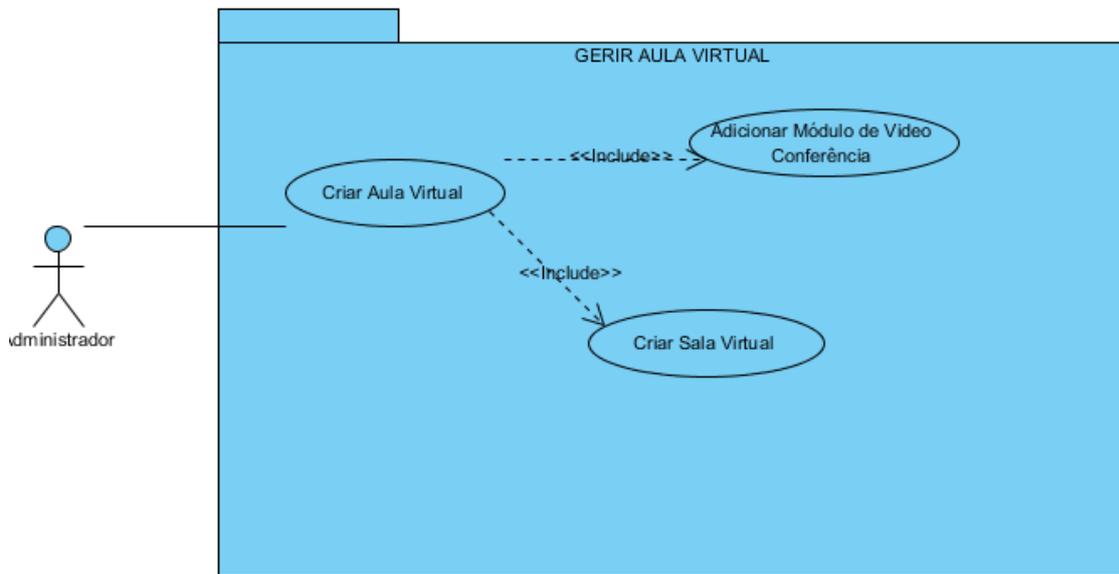


Figura 6 - Diagrama de caso de uso gerir aula virtual

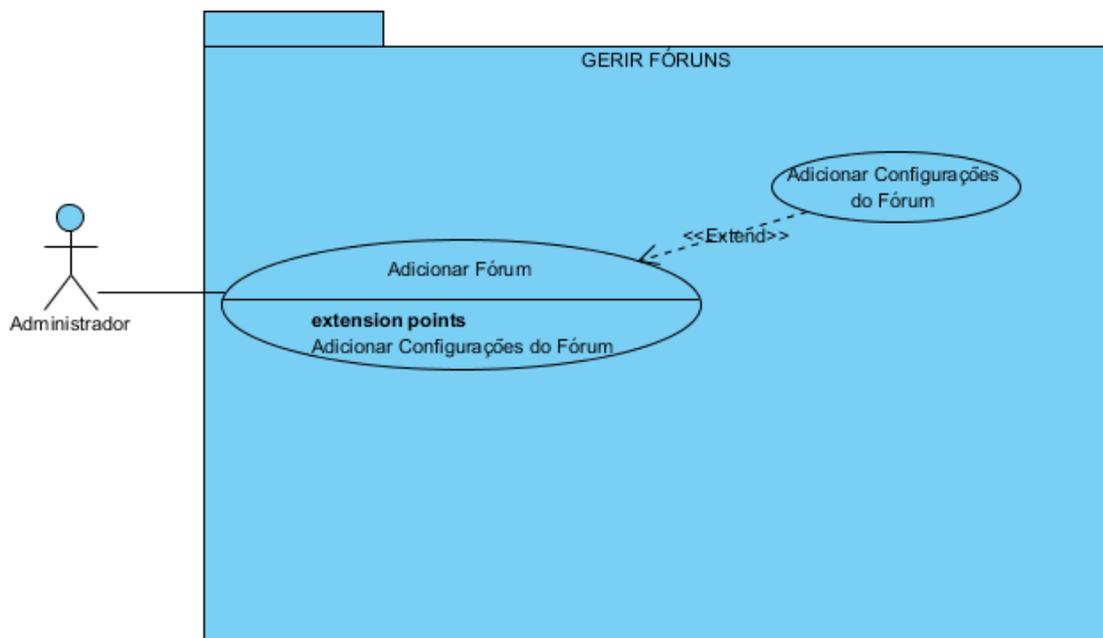


Figura 7 - Diagrama de caso de uso gerir fóruns

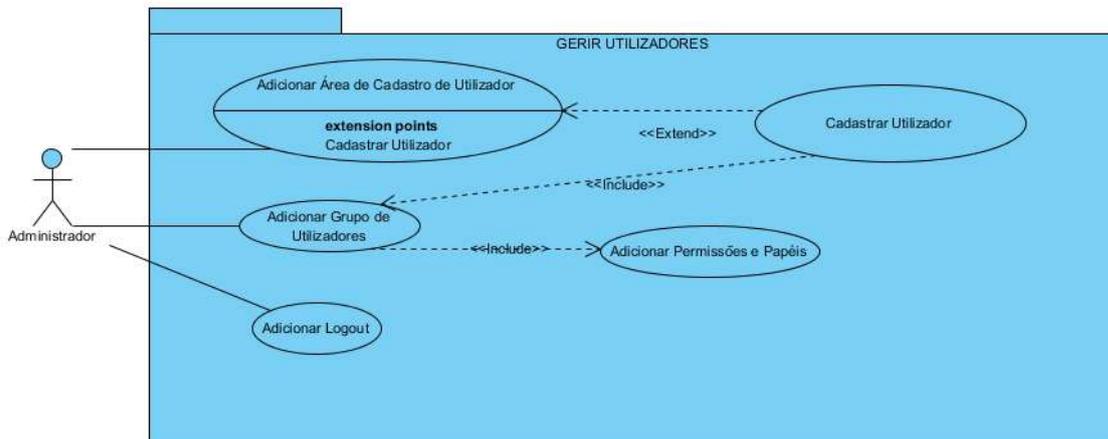


Figura 8 - Diagrama de caso de uso gerir utilizadores

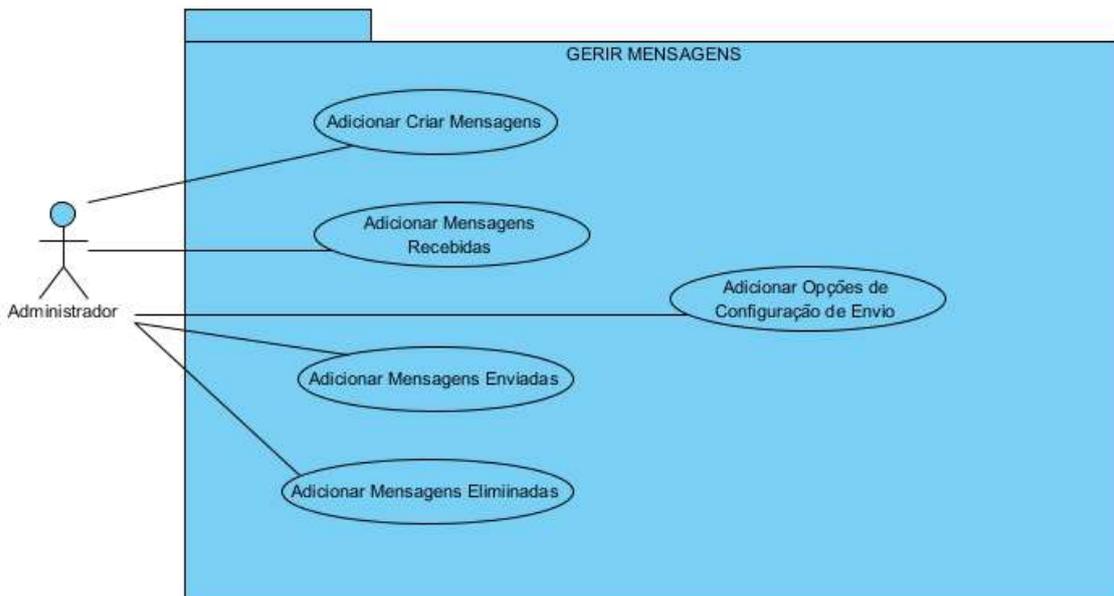


Figura 9 - Diagrama de caso de uso gerir mensagens

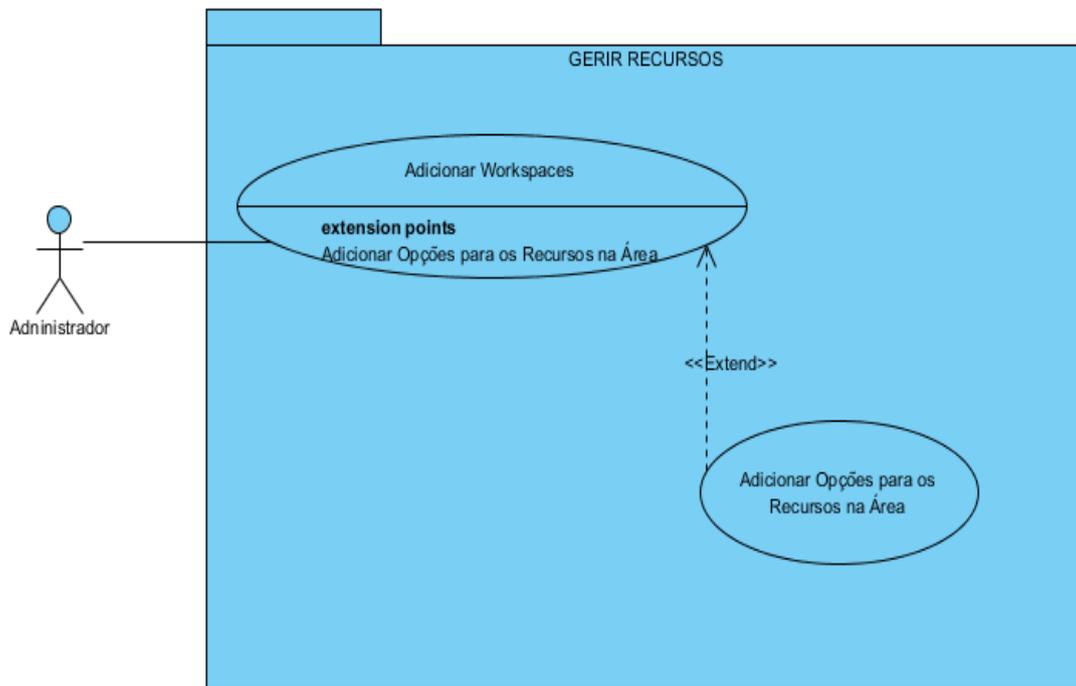


Figura 10 - Diagrama de caso de uso gerir anúncios

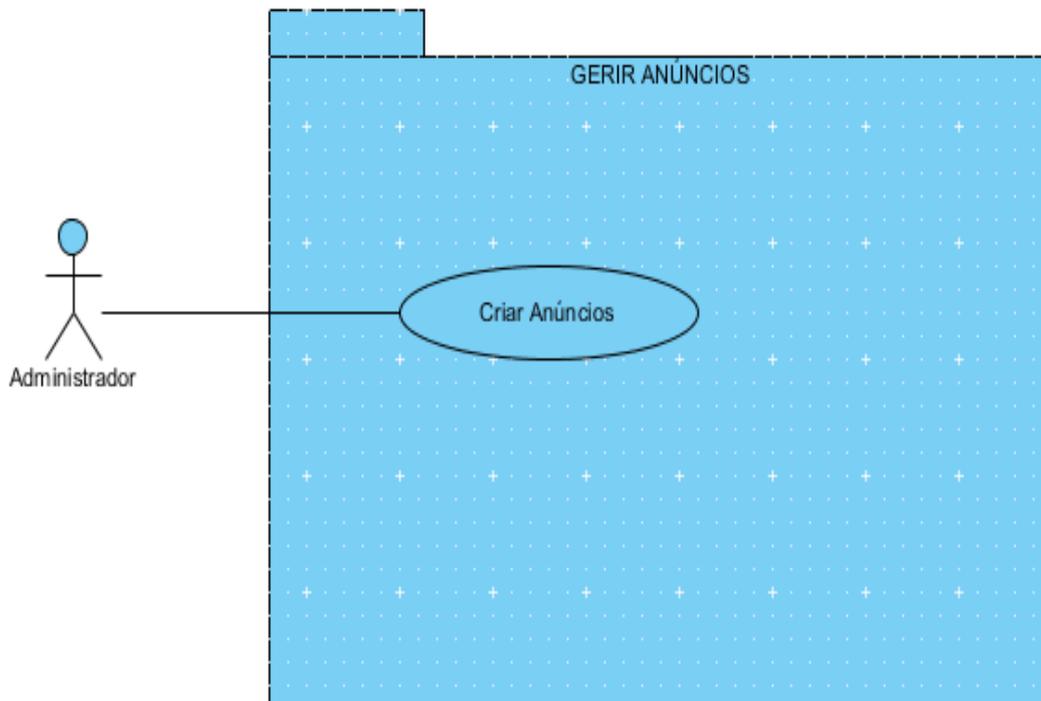


Figura 11 - Diagrama de caso de uso gerir anúncio



Figura 12 - Diagrama de caso de uso gerir professor

Descrição detalhada dos casos de uso

É uma descrição textual de como os eventos fluem ou acontecem para realizar uma determinada operação, tendo em conta todos os actores do caso de uso e a respectiva resposta do sistema.

I. Adicionar Área Pessoal

Actor: Administrador

Tipo: Primário

Pré-condição: *Login* efectuado e área pessoal seleccionada

Fluxo de Eventos para adicionar área pessoal:

1. Utilizador selecciona opção Adicionar Área Pessoal
2. São mostrados campos para preenchimento de informação referente à área pessoal (informação pessoal, informação básica, informação de contacto e informação académica).⁸
3. Utilizador preenche os campos
4. Sistema guarda na base de dados a informação inserida
5. Sistema confirma que a informação foi guardada com sucesso

Pós-condição: Área Pessoal adicionada

II. Adicionar Área de Informação Pessoal

Actor: Administrador

Tipo: Primário

Pré-condição: *Login* efectuado e área pessoal seleccionada

⁸ Este requisito funcional permite o cadastro dos dados do aluno, ou seja, a criação do seu e perfil.

Fluxo de Eventos para adicionar área pessoal:

1. Utilizador selecciona opção Adicionar Área de Informação Pessoal
2. São mostrados campos para criação da área de Informação Pessoal.
3. Utilizador preenche os campos
4. Sistema guarda na base de dados a informação inserida
5. Sistema confirma que a informação foi guardada com sucesso

Pós-condição: Área de Informação Pessoal adicionada

III. Adicionar Unidade Curricular

Actor: Administrador

Tipo: Primário

Pré-condição: Login efectuado e área de unidade curricular seleccionada

Fluxo de Eventos para adicionar área pessoal:

1. Utilizador selecciona opção Adicionar Unidade Curricular
2. São mostrados campos para preenchimento de informação referente à unidade curricular.
3. Utilizador preenche os campos
4. Sistema guarda na base de dados a informação inserida
5. Sistema confirma que a informação foi guardada com sucesso

Pós-condição: Unidade Curricular adicionada

IV. Adicionar Curso

Actor: Administrador

Tipo: Primário

Pré-condição: Login efectuado e área de Curso Seleccionada

Fluxo de Eventos para adicionar curso:

1. Utilizador selecciona opção Adicionar Curso
2. São mostrados campos para preenchimento de informação referente ao curso.
3. Utilizador preenche os campos
4. Sistema guarda na base de dados a informação inserida
5. Sistema confirma que a informação foi guardada com sucesso

Pós-condição: Curso adicionado

V. Criar Aula Virtual

Actor: Administrador

Tipo: Primário

Pré-condição: 1. Login efectuado
2. Criação de Sala Virtual
3. Seleccionar área de Aula Virtual

Fluxo de Eventos para adicionar aula virtual:

1. Utilizador selecciona opção Criar Aula Virtual
2. São mostrados campos para preenchimento de informação referente a criação da aula virtual.
3. Utilizador preenche os campos e cria o ficheiro relacionado a aula virtual

4. Sistema guarda na base de dados a informação inserida
5. Sistema confirma que a informação foi guardada com sucesso

Pós-condição: Aula Virtual adicionada

VI. Adicionar Fórum

Actor: Administrador

Tipo: Primário

Pré-condição: 1. Login efectuado e área de Fórum seleccionada

Fluxo de Eventos para adicionar fórum:

1. Utilizador selecciona opção Adicionar Fórum
2. São mostrados campos para preenchimento de informação referente ao fórum.
3. Utilizador preenche os campos relacionados ao fórum
4. Sistema guarda na base de dados a informação inserida
5. Sistema confirma que a informação foi guardada com sucesso

Pós-condição: Fórum adicionado

VII. Adicionar Área de Cadastro de Utilizador

Actor: Administrador

Tipo: Primário

Pré-condição: 1. Login efectuado

Fluxo de Eventos para adicionar área de cadastro de utilizador:

1. Utilizador selecciona opção Adicionar Área de Cadastro de Utilizador
2. São mostrados campos para preenchimento da informação referente a área de cadastro do utilizador.
3. Utilizador preenche os campos relacionados à área de cadastro
4. Sistema guarda na base de dados a informação inserida
5. Sistema confirma que a informação foi guardada com sucesso

Pós-condição: Área de Cadastro de Utilizador adicionada

VIII. Cadastrar Utilizador

Actor: Administrador

Pré-condição: 1. Login efectuado

2. Área de cadastro de utilizador seleccionada

Fluxo de Eventos para cadastrar utilizador:

1. Utilizador selecciona opção Cadastrar Utilizador
2. São mostrados campos para preenchimento da informação referente ao cadastro do utilizador.
3. Utilizador preenche os campos relacionados ao cadastro
4. Sistema guarda na base de dados a informação inserida
5. Sistema confirma que a informação foi guardada com sucesso

Pós-condição: Cadastro de Utilizador adicionado

IX. Adicionar Grupo de Utilizadores

Actor: Administrador

Pré-condição: 1. Login efectuado
2. Adicionar Permissões e Pápeis

Fluxo de Eventos para adicionar grupo utilizador:

1. Utilizador selecciona opção Adicionar Grupo de Utilizador
2. São mostrados campos para preenchimento da informação referente aos grupos de utilizadores.
3. Utilizador preenche os campos relacionados aos grupos de utilizadores
4. Sistema guarda na base de dados a informação inserida
5. Sistema confirma que a informação foi guardada com sucesso

Pós-condição: Grupo de Utilizador adicionado

X. Adicionar *Workspace*

Actor: Administrador

Tipo: Primário

Pré-condição: 1. Login efectuado

Fluxo de Eventos para adicionar logout:

1. Utilizador Adicionar a Opção Adicionar *Workspace*
2. É mostrado o campo para preenchimento da informação referente a adição do *Workspace* para todas as disciplinas
3. Utilizador preenche os campos relacionados ao *workspace*
4. Sistema guarda na base de dados a informação inserida
5. Sistema confirma que a informação foi guardada com sucesso

Pós-condição: *Workspace* adicionado

XI. Cadastrar Professor

Actor: Administrador

Tipo: Primário

Pré-condição: 1. Login efectuado
2. Área de cadastro de professor seleccionada

Fluxo de Eventos para cadastrar utilizador:

1. Utilizador selecciona opção Cadastrar Professor
2. São mostrados campos para preenchimento da informação referente ao cadastro do professor.
3. Utilizador preenche os campos relacionados ao cadastro do professor
4. Sistema guarda na base de dados a informação inserida
5. Sistema confirma que a informação foi guardada com sucesso

Pós-condição: Cadastro de Professor adicionado

XII. Adicionar Criar Mensagem

Actor: Administrador

Tipo: Primário

Pré-condição: 1. Login efectuado e área de mensagens seleccionada

Fluxo de Eventos para adicionar criar mensagem:

1. Utilizador selecciona opção Adicionar Criar Mensagem
2. São mostrados campos para preenchimento da informação referente a criação de mensagem.
3. Utilizador preenche os campos relacionados a criação de mensagem
4. Sistema guarda na base de dados a informação inserida
5. Sistema confirma que a informação foi guardada com sucesso

Pós-condição: Criar Mensagem adicionado

XIII. Adicionar Mensagens Recebidas

Actor: Administrador

Tipo: Primário

Pré-condição: 1. Login efectuado e área de mensagens seleccionada

Fluxo de Eventos para adicionar mensagens recebidas:

1. Utilizador selecciona opção Adicionar Mensagens Recebidas
2. São mostrados campos para preenchimento da informação referente as mensagens recebidas.
3. Utilizador preenche os campos relacionados as mensagens recebidas
4. Sistema guarda na base de dados a informação inserida
5. Sistema confirma que a informação foi guardada com sucesso

Pós-condição: Mensagens Recebidas adicionada

XIV. Adicionar Mensagens Eliminadas

Actor: Administrador

Tipo: Primário

Pré-condição: 1. Login efectuado e área de mensagens seleccionada

Fluxo de Eventos para adicionar mensagens eliminadas:

1. Utilizador selecciona opção Adicionar Mensagens Eliminadas
2. São mostrados campos para preenchimento da informação referente as mensagens eliminadas.
3. Utilizador preenche os campos relacionados as mensagens eliminadas
4. Sistema guarda na base de dados a informação inserida
5. Sistema confirma que a informação foi guardada com sucesso

Pós-condição: Mensagens Eliminadas adicionadas

XV. Adicionar Mensagens Enviadas

Actor: Administrador

Tipo: Primário

Pré-condição: 1. Login efectuado e área de mensagens seleccionada

Fluxo de Eventos para adicionar mensagens enviadas:

1. Utilizador selecciona opção Adicionar Mensagens Enviadas
2. São mostrados campos para preenchimento da informação referente as mensagens enviadas.

3. Utilizador preenche os campos relacionados as mensagens enviadas
 4. Sistema guarda na base de dados a informação inserida
 5. Sistema confirma que a informação foi guardada com sucesso
- Pós-condição:** Mensagens Enviadas adicionado

XVI. Adicionar Opções de Configuração de Envio

Actor: Administrador

Tipo: Primário

Pré-condição: 1. Login efectuado e área de mensagens seleccionada

Fluxo de Eventos para adicionar opções de configuração de envio:

1. Utilizador selecciona opção Adicionar Opções de Configuração de Envio
 2. São mostrados campos para preenchimento da informação referente as opções de configuração de envio.
 3. Utilizador preenche os campos relacionados as opções de envio
 4. Sistema guarda na base de dados a informação inserida
 5. Sistema confirma que a informação foi guardada com sucesso
- Pós-condição:** Opções de Configuração de Envio adicionada

XVII. Criar Sala Virtual

Actor: Administrador

Tipo: Primário

Pré-condição: 1. Login efectuado

Fluxo de Eventos para adicionar aula virtual:

1. Utilizador selecciona opção Criar Sala Virtual
 2. São mostrados campos para preenchimento de informação referente a criação da sala virtual.
 3. Utilizador preenche os campos relacionados a criação da sala virtual
 4. Sistema guarda na base de dados a informação inserida
 5. Sistema confirma que a informação foi guardada com sucesso
- Pós-condição:** Sala Virtual adicionada

XVIII. Adicionar Turma

Actor: Administrador

Pré-condição: Login efectuado e área de turma seleccionada

Fluxo de Eventos para adicionar turma:

1. Utilizador selecciona opção Adicionar Turma
 2. São mostrados campos para preenchimento de informação referente turma.
 3. Utilizador preenche os campos
 4. Sistema guarda na base de dados a informação inserida
 5. Sistema confirma que a informação foi guardada com sucesso
- Pós-condição:** Turma adicionada

Dicionário de Dados

O dicionário de dados é, uma listagem organizada de todos os elementos de dados do sistema, com definições precisas e rigorosas para que os utilizadores e os analistas de sistemas possam reconhecer todas as entradas, saídas, componentes de depósitos de dados e cálculos intermédios (Rocha, 2008).

ALUNO			
Nome campo	Descrição	Tipo	Chave
#numero_matricula	Número de matrícula do aluno	Numeração Automática	Primária
#id_pessoa	Código da pessoa que é aluno	Inteiro	Estrangeira
data_matricula	Data da matrícula	Data	
ano_curso	Ano de frequência do curso	Inteiro	
#ALUNO_CURSOid_curso	Código do curso	Inteiro	Estrangeira
AVALIACAO_ALUNO			
#id_avaliacao	Código da avaliação	Inteiro	Primária
AVALIACAO_PROFESSOR			
#id_professor	Código do professor	Inteiro	Primária
#id_avaliacao	Código da avaliação	Inteiro	Primária
AVALIACAO			
#id_avaliacao	Código da avaliação	Numeração Automática	Primária

data_avaliacao	Data da avaliação	Data	
duração	Tempo de duração da avaliação	Hora	
#id_uc	Código da unidade curricular	Inteiro	Estrangeira
tipo_avaliacao	Tipo de avaliação. Por exemplo: avaliação contínua, frequência, exame, recurso, exame especial	Texto	
ANO_CURSO			
#numero_matricula	Número de matrícula do aluno	Inteiro	Primária
#id_curso	Código do curso	Inteiro	Primária
UTILIZADOR			
#id_utilizador	Código do utilizador	Numeração	Primária
#id_pessoa	Código da pessoa que é utilizador	Inteiro	Estrangeira
nome_utilizador	Nome da conta do utilizador	Texto	
senha	Palavra passe do utilizador	Texto	
hora_logout	Hora de término da sessão	Hora	
hora_login	Hora de início da sessão	Hora	
#id_grupo	Código do grupo de utilizador	Inteiro	Estrangeira

CURSO			
#ALUNO_CURSOid_curso	Código do aluno	Inteiro	Estrangeira
#CURSO_UNIDADE_CURRICULARid_unidade_curricular	Código da unidade curricular	Inteiro	Estrangeira
#id_curso	Código do curso	Numeração Automática	Primária
nome_curso	Nome ou descrição do curso	Texto	
sigla	Sigla de abreviatura do curso	Texto	
ciclo	Ciclo de estudos do curso	Texto	
ALUNO_AULA			
#id_aluno	Código do aluno	Inteiro	Primária
#id_aula	Código da aula	Inteiro	Primaria
TURMAS			
#id_turma	Código da turma	Numeração Automática	Primária
nome_turma	Nome da turma	Texto	
#id_curso	Código do curso	Inteiro	Estrangeira
PESSOA			
#id_pessoa	Código da pessoa	Numeração Automática	Primária
nome	Nome da pessoa	Texto	

telefone	Telefone da pessoa	Inteiro	
email	Correio electrónico da pessoa	Texto	
sexo	Sexo da pessoa	Texto	
#id_endereco	Código do endereço	Inteiro	Estrangeira
PROFESSOR			
#id_professor	Código do professor	Numeração Automática	Primária
#id_pessoa	Código da pessoa	Inteiro	Estrangeira
grau_academico	Descrição do grau. Por exemplo_ licenciado, mestre ou doutor	Texto	
categoria	Descrição da categoria. Por exemplo: assistente estagiário, assistente, auxiliar, associado, titular	Texto	
#id_unidade_curricular	Código da unidade curricular	Inteiro	Estrangeira
UC_PROFESSOR			
#id_professor	Código do professor	Inteiro	Primária
#id_uc	Código da unidade curricular	Inteiro	Primária
NOTA			
#id_nota	Código da nota	Numeração	Primária

		Automática	
nota	Nota do aluno	Inteiro	
#id_uc	Código da disciplina pertencente a nota	Inteiro	Estrangeira
#id_professor	Códig do professor	Inteiro	Estrangeira
data_lancamento	Data de lançamento da nota	Data	
CURSO_UNIDADE_CURRICULAR			
#id_curso	Código do curso	Inteiro	Primária
#id_unidade_curricular	Código da unidade curricular	Inteiro	Primária
UNIDADE CURRICULAR			
#id_unidade_curricular	Código da unidade curricular	Numeração Automática	Primária
nome_uc	Nome da unidade curricular	Texto	
sigla_uc	Sigla da unidade curricular	Texto	
nome_ciclo	Nome do ciclo pertencente a disciplina	Texto	
CURSO_UNIDADE_CURRICULARid_curso	Código do curso	Inteiro	Estrangeira
AULA VIRTUAL			
#id_aula	Código da aula	Numeração Automática	Primária

data_criacao	Data de criação da aula	Data	
nome_aula	Nome da aula	Texto	
duracao	Duração da aula	Hora	
#id_professor	Código do Professor	Inteiro	Estrangeira
MENSAGEM			
#id_mensagem	Código da mensagem	Inteiro	Primaria
data_envio	Data de envio da mensagem	Data	
conteudo	Conteúdo da mensagem	Texto	
#id_professor	Código do professor	Inteiro	Estrangeira
FORUM			
#id_forum	Código do fórum	Inteiro	Primária
topico	Tópico do fórum	Texto	
data_publicacao	Data de publicação da mensagem	Data	
#id_curso	Código do curso	Inteiro	Estrangeira
primeira_leitura	Controlo das primeiras leituras do forum	Inteiro	
ultima_leitura	Controlo das últimas leituras do fórum	Inteiro	
#id_professor	Código do professor	Inteiro	Estrangeira

#numero_matricula	Número de matricula do aluno	Inteiro	Estrangeira
ENDEREÇO			
#id_endereco	Código do endereço	Numeração Automática	Primária
rua	Nome da rua	Texto	
local	Nome do local	Texto	
codigo_postal	Número Postal	Inteiro	
CHAT			
#id_chat	Código do chat	Inteiro	Primária
remetente	Nome do remetente	Texto	
destinatario	Nome do destinatário	Texto	
conteudo	Conteúdo da mensagem	Texto	
#id_professor	Código do professor	Inteiro	Estrangeira
ANÚNCIOS			
#id_anuncio	Código do anúncio	Numeração Automática	Primária
conteudo	Conteúdo do anúncio	Texto	
data_criacao	Data de criação do anúncio	Data	
#id_professor	Código do professor	Inteiro	Estrangeira
RECURSO			

#id_recurso	Código do Recurso	Numeração Automática	Primária
nome_recurso	Nome do recurso	Númeraço Automática	
data_criacao	Data de criação do recurso	Data	
#id_consulente	Código do consulente	Inteiro	Estrangeira
#id_professor	Código do professor	Inteiro	Estrangeira

Diagrama de Fluxo de Dados (DFD) de alguns dos casos de uso

Um DFD é uma representação em rede de um sistema (independentemente de ser digital ou analógico) e que representa o fluxo de dados a sua transformação. O DFD modela o sistema representando as suas partes componentes, enfatizando os processos realizados (funções) e os dados que fluem entre eles. Um DFD mostra o Fluxo dos Dados e não o fluxo de controlo. Portanto, controlos, como interacções e decisões não são representados neste diagrama.

Apresenta-se o DFD dos casos de uso cadastrar professor e cadastrar curso, a título de exemplo:

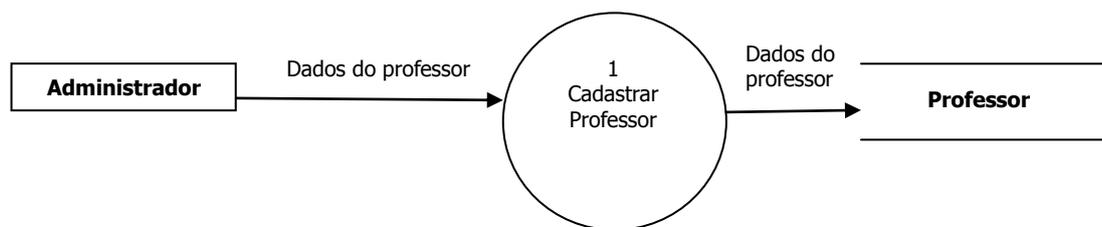


Figura 13 – DFD Administrador cadastrando professor no sistema

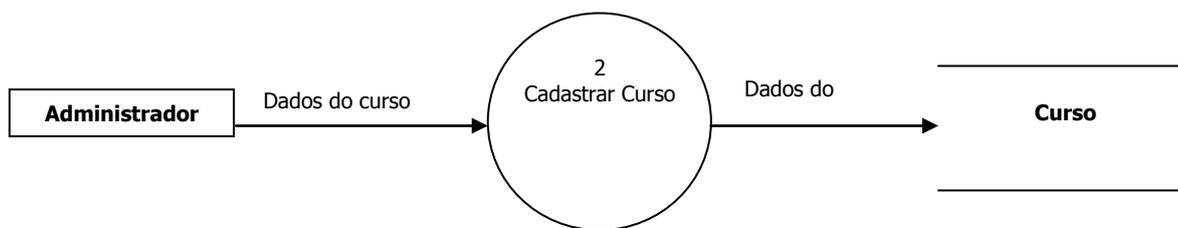


Figura 14 – DFD Administrador cadastrando curso no sistema

4. Arquitectura do sistema

A Arquitectura de um Sistema descreve os dispositivos necessários ao funcionamento do sistema, isto é, define a relação entre os diferentes componentes da aplicação e os mecanismos de gestão para o seu funcionamento. O funcionamento de uma aplicação requer um conjunto de componentes disponíveis tendo em conta a especificidade da aplicação.

Desta forma, a Arquitectura do Sistema fornece os meios para a implementação de um projecto concebido antecipadamente para funcionamento futuro do sistema ou aplicação, de forma estável e segura.

Os servidores necessários à arquitectura do BackOffice do Sistema de Ensino a Distância para Universidade Católica de Angola são: Servidor de Base de Dados, Servidor Web⁹ (Servidor Apache) e Servidor de Aplicações J2EE. É necessário ter em conta que a presença destes três servidores é fundamental para o bom funcionamento da aplicação.

É necessário ter em conta que os três servidores são indissociáveis quando se pretende implementar uma aplicação distribuída.

Servidor de Aplicação J2EE: é o servidor desenvolvido para disponibilizar uma aplicação J2EE. Além de disponibilizar aplicações, os SA também oferecem vários recursos ao

⁹ Pelo seu cariz, o servidor *web* é responsável pelo armazenamento das páginas de uma aplicação ou sistema, compreendendo assim a disponibilização de dados, imagens, arquivos e informações das mais diversas naturezas.

desenvolvedor como pool de conexão, controlo de transacção, segurança, gestão de recursos, etc.

O Servidor de Aplicação¹⁰: contém a lógica da aplicação e situa-se entre o servidor web e o servidor de base de dados. Controla a conexão com a base de dados em função das requisições do browser.

Servidor Web: Servidor responsável pelo armazenamento de páginas do Sistema de Ensino à Distância, que serão requisitados pelos clientes através de *browsers*.

Os servidores Web¹¹ são responsáveis por armazenar e trocar informações com outras máquinas. Por causa disso, pelo menos dois participantes são envolvidos em cada troca de informações: um cliente, que solicita informações, e um servidor, que atende a esses pedidos.

Para o caso da Sistema de Ensino a Distância usando a plataforma de e-learning Sakai, utilizou-se o protocolo https, visando prover maior segurança durante os acessos.

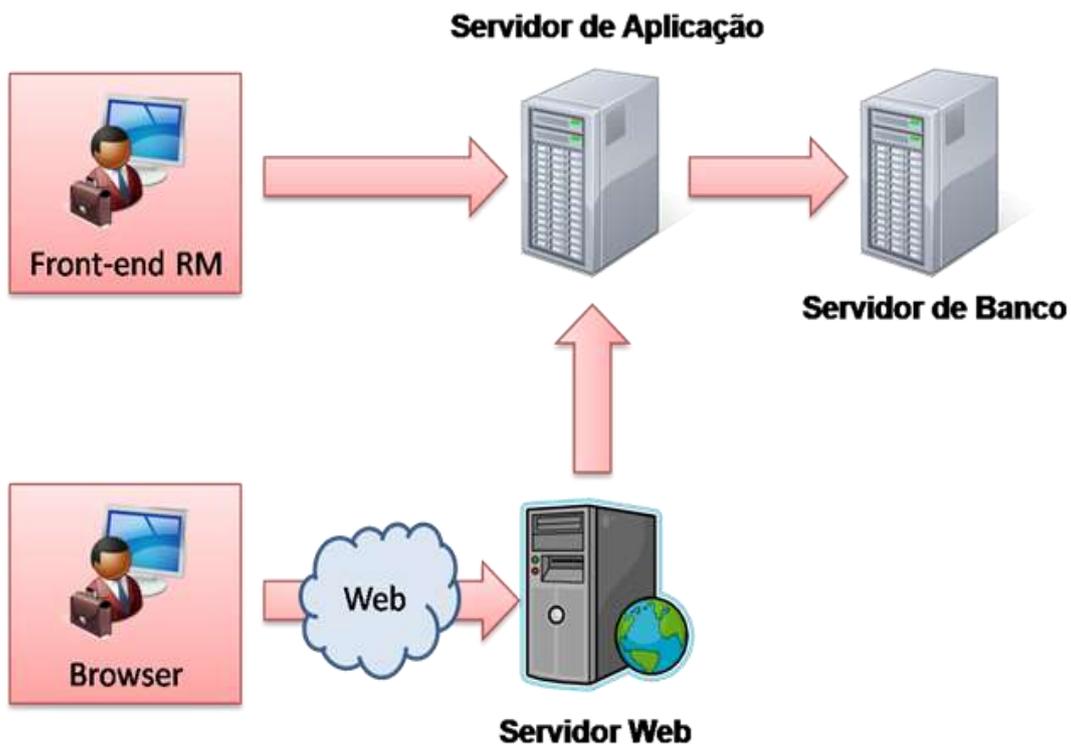
Servidor de Base de Dados: é responsável pelo armazenamento das informações do Sistema de Ensino a Distância e pela geração de páginas dinâmicas resultantes de consultas ou guardar dados preenchidos em formulários. Permite ainda manipular as informações existentes numa base de dados.

Em suma, a implementação deste exige a utilização da arquitectura em 3 camadas: camada de apresentação, camada de aplicação e camada de dados, donde os servidores supra apresentados fazem parte

¹⁰ Para o sistema em questão será utilizado como servidor de aplicação o Tomcat.

¹¹ Os servidores de aplicação também são conhecidos como *software de middleware*.

As figuras abaixo mostram a arquitectura física da aplicação e seus respectivos servidores.



NOTA: É importante ter em conta que esta arquitectura terá um servidor adicional onde serão guardados todos os *backups* do Sistema de *e-learning*, de molde a facilitar o plano de *disaster recovery*.

Figura 15 - Arquitectura de Sistema (Servidores)

4.1 Diagrama de interacção

Os diagramas de interacção são utilizados em UML para modelar os aspectos dinâmicos do sistema em termos dos objectos e suas relações, com base nas mensagens trocadas entre objectos e suas relações. São modelos que descrevem como grupos de objectos colaboram entre si na obtenção de um dado comportamento. Agrupam dois tipos de diagramas: diagramas de sequência e diagramas de colaboração que serão usados para representação da interacção da arquitectura do sistema.

Diagrama de sequência

Os diagramas de sequência servem para representar a ordem sequencial e temporal de vários exemplos, processos e operações do sistema que ocorrem para atingir uma determinada finalidade.

O diagrama de sequência abaixo apresentado mostrará a interação entre os vários servidores na interação do utilizador com o sistema.

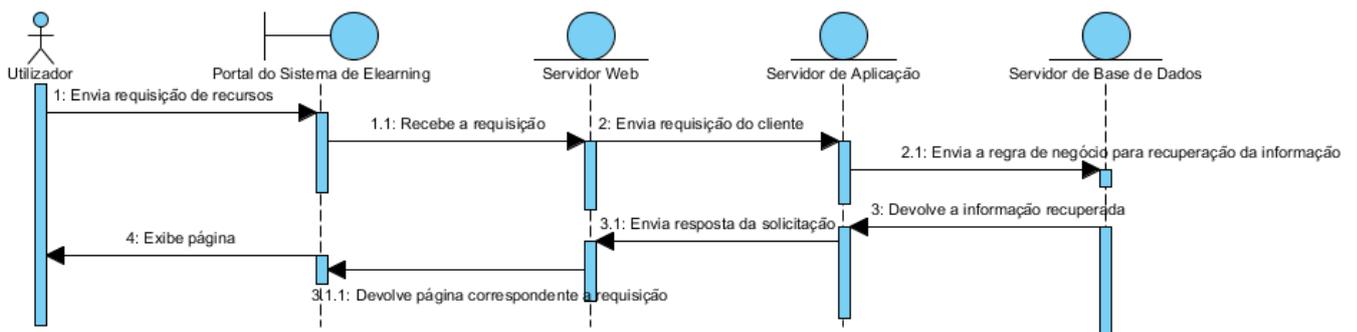


Figura 16 - Diagrama de sequência da arquitectura do sistema

Diagrama de colaboração

Os diagramas de colaboração transmitem a mesma informação que os diagramas de sequência, mas em vez dos tempos em que as mensagens são enviadas dão ênfase à estrutura organizacional dos objectos que enviam e recebem as mensagens.

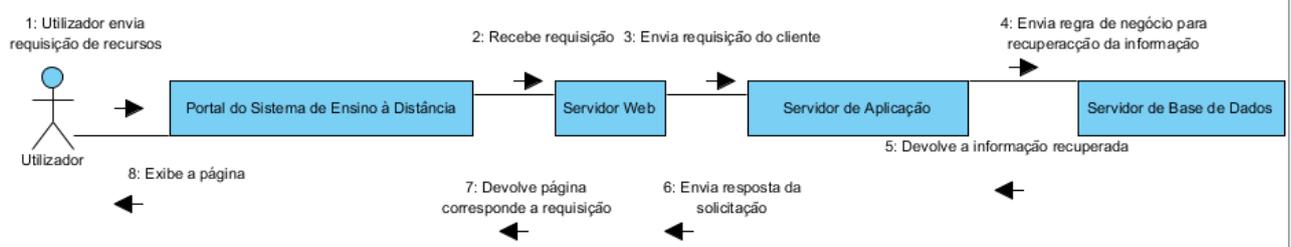


Figura 17 – Diagrama de Colaboração do Sistema

4.2 Diagrama de componentes

Mostra as dependências entre componentes de software, inclusive componentes de código fonte, código binário, e códigos executáveis. Alguns componentes existem no momento da ligação, outros em tempo de execução.

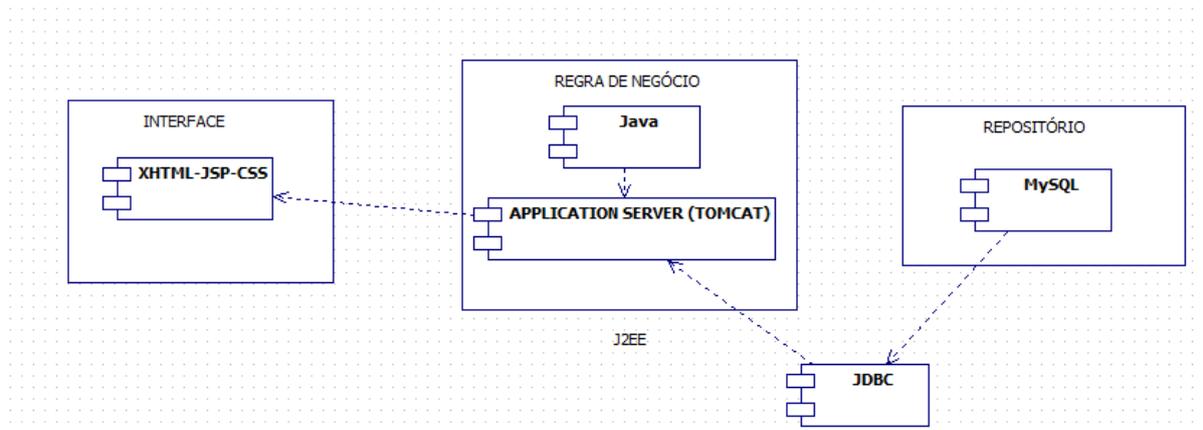


Figura 18 - Diagrama de Componentes do Sistema

4.3 Diagrama de implementação

Mostra os relacionamentos físicos entre os componentes de software e hardware no sistema instalado. Na prática um diagrama de implementação permite demonstrar como o hardware está organizado e como os componentes (software) estarão distribuídos estabelecendo assim a sua relação física.

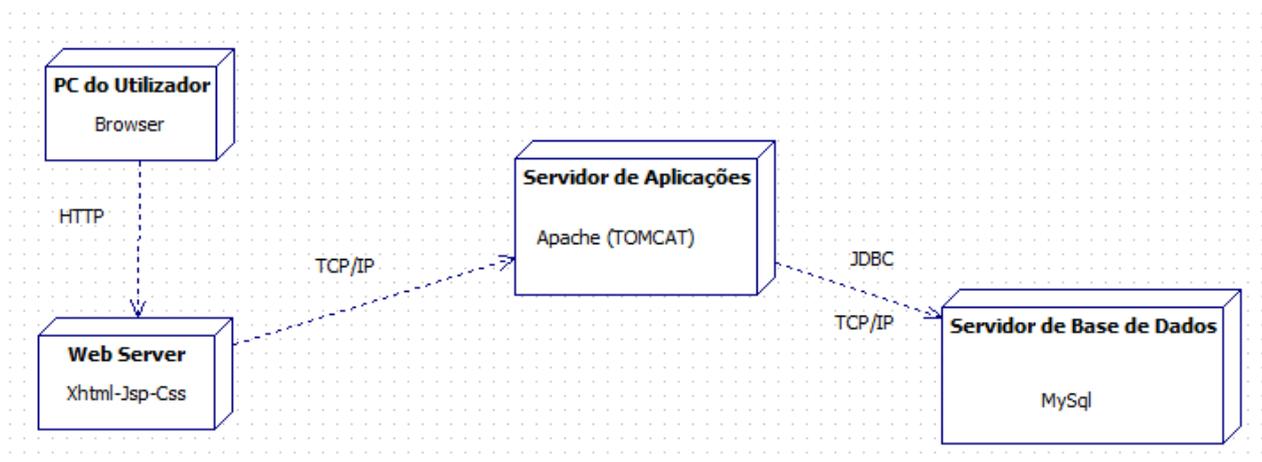


Figura 19 - Diagrama de Implementação do Sistema

4.4 Metodologia e tecnologias usadas

O Sistema foi desenvolvido com base nas metodologias sugeridas pela Engenharia de Software (Pressman, 2006).

Modelagem

Para modelação dos requisitos utilizou-se o padrão UML para sistemas orientados aos objectos.

Software de modelação

Para a modelação do sistema foram utilizados as seguintes ferramentas:

- *Visual Paradigm for UML*: é uma ferramenta CASE voltada para UML (*Unified Modeling Language*), tornando possível trabalhar com os diversos diagramas da UML 2.0, como diagramas de caso de uso, diagramas de classes e diagramas de sequência, entre outros.
- *StarUML*: é uma ferramenta CASE de código aberto (*open source*) sob licença GPL (*General Public License*). Dá suporte à modelação de sistemas utilizando os diagramas da UML 2. Permite a importação/exportação de modelos utilizando o formato XMI.

Levantamento de Requisitos

Para o levantamento de requisitos utilizou-se o processo proposto por Sommerville (2003)¹².

¹² Sommerville propõe 6 fases para o levantamento de requisitos que são: compreensão do domínio, colecta de requisitos, classificação, resolução de conflitos, definição de prioridades e verificação de requisitos.

Codificação

Para a codificação utilizou-se a linguagem de programação JAVA, orientado aos objectos com o ambiente de desenvolvimento *NetBeans*.

Base de Dados

Para a base de dados, foi utilizado o MySQL com o auxílio das ferramentas *MySQL GUI TOOLS*.

Plataforma de *e-learning*

Para implementação do sistema de *e-learning* foi utilizada a plataforma de software livre e opensource SAKAI¹³ por ser uma plataforma escalável, segura, interoperável e extensível. É um sistema de gestão de conteúdos, que permite um ambiente colaborativo de ensino e aprendizagem.

Plataforma para as Aulas Virtuais

Foi utilizado o *BlackBoard Collaborate*¹⁴, plataforma que permite a criação de salas e aulas virtuais. Este componente implica um custo e pode ser substituído por uma ferramenta do SAKAI¹⁵.

4.5 Requisitos técnicos

Para instalar o software Sakai, é necessário satisfazer um conjunto de requisitos técnicos abaixo apresentados, tendo em conta a especificidade de cada servidor e instalação específica (sua escala, ferramentas/serviços adoptados e sistemas de desenvolvimento).

Para mais informações ver: <http://www.devmedia.com.br/engenharia-de-software-2-tecnicas-para-levantamento-de-requisitos/9151>

¹³ <http://sakaiproject.org/>

SERVIDOR DE BASE DE DADOS	
Memória	128 GB (recomenda-se 1 GB por cada 20 utilizadores)
CPU	CPU 2x Xeon E5 2620V2
Cores	12
Disco	10 TB (usando RAID 10)
Sistema Operativo	Linux (preferencialmente <i>Ubuntu Server</i> ou <i>Red Hat Enterprise Linux (RHEL) AS</i>)
Base de Dados	<i>MySQL</i>
SERVIDOR APLICACIONAL	
Memória	16 GB
CPU	CPU 2x Xeon E5 2620V2
Cores	4
Disco	1 TB (usando RAID 5)
Sistema Operativo	Linux (preferencialmente <i>Ubuntu Server</i> ou <i>Red Hat Enterprise Linux (RHEL) AS</i>)
Aplicação	<u>Tomcat 5.5.25+</u>
SERVIDOR WEB	
Memória	32 GB
CPU	CPU 2x Xeon E5 2620V2
Cores	6
Disco	4 TB (usando RAID 10)

¹⁴ <http://www.blackboard.com/platforms/collaborate/products/blackboard-collaborate.aspx>

¹⁵ <https://confluence.sakaiproject.org/display/BBB/Home>

Sistema Operativo	Linux (preferencialmente <i>Ubuntu Server</i> ou <i>Red Hat Enterprise Linux (RHEL) AS</i>)
Aplicação	Apache 2.0.x/2.2.x (with mod_jk connector 1.2.26)
SERVIDOR DE <i>BACKUP</i>	
Memória	64 GB
CPU	CPU 2x Xeon E5 2620V2
Cores	18
Disco	20 TB (usando RAID 10)
Sistema Operativo	Linux (preferencialmente <i>Ubuntu Server</i> ou <i>Red Hat Enterprise Linux (RHEL) AS</i>)
Base de Dados	MySQL
UTILIZADORES	
Software	<i>Browser Web</i> JDK 1.6

5. Comentários Finais e recomendações

A Universidade Católica de Angola, desejando inovar e explorar as tecnologias de informação, especificamente Sistemas de Gestão de Aprendizagem, propôs-se implementar o *BackOffice* de um Sistema de Ensino a Distância que no final estará disponível para a comunidade académica (docentes, alunos) através da implementação do *FrontOffice*.

As plataformas tecnológicas que suportam os ambientes virtuais de aprendizagem têm assumido especial interesse pelos órgãos de decisão das instituições de ensino superior

devido ao papel nuclear que assumem na gestão do processo de ensino, aprendizagem e investigação. Estas plataformas admitem várias designações, tais como: ambientes colaborativos de aprendizagem (CLE), sistemas de gestão da aprendizagem (LMS), sistemas de gestão de cursos (CMS) ou ambientes virtuais de aprendizagem (VLE).

Tendo em conta a redução do custo de implementação que é o principal factor considerado na implementação de um projecto de *e-learning*, decidiu-se utilizar o ambiente colaborativo de aprendizagem *open source* e não um de oferta comercial. Se por um lado as plataformas comerciais têm um tempo de implementação menor, por outro, podem ter um elevado custo de licenciamento e exigem um maior investimento em dinheiro (do ponto de vista de tempo, podem constituir uma vantagem, contando com o tempo de exploração e aprendizagem técnica mais exigente, nas soluções não comerciais).

A plataforma *open source*¹⁶ escolhida para a implementação do BackOffice do Sistema de Ensino à Distância foi o SAKAI, por ser uma plataforma escalável, segura, interoperável e extensível, tendo como vantagem a partilha de aplicações entre instituições de ensino superior que aderiram ao projecto, enriquecendo desse modo a própria plataforma.

A seguir fez-se o levantamento dos requisitos funcionais do sistema usando os princípios de UML de molde a responder às necessidades da instituição.

Por uma questão de boas práticas de desenvolvimento, estruturou-se o sistema em módulos para uma melhor e maior gestão, visando a integração do referido sistema.

¹⁶ As principais vantagens de um LMS *Open Source* são as seguintes: não têm custos de licenciamento, tem maior flexibilidade, normalmente assegurando uma maior continuidade de serviço e podem ser melhoradas continuamente pela comunidade que a suporta.

Implementou-se o sistema com base na Arquitectura em 3 camadas: camada de apresentação, camada de aplicação e a camada de dados, tendo o sistema codificado na linguagem Java.

Para a Arquitectura física configuraram-se servidores de topo de gama com requisitos técnicos de alto desempenho na plataforma linux devido a robustez e a estabilidade, de formas a prover melhor performance tendo em conta a demanda de utilizadores concorrentes.

O *FrontOffice* do sistema é baseado em JSP (*Java Server Pages*) uma vez que o sistema foi codificado em Java. Estão planeados vários testes numa plataforma virtual com o intuito de avaliar o seu funcionamento real. Da exploração inicial realizada com um grupo de vinte alunos da Universidade Católica de Angola, os resultados são satisfatórios. Só depois de realizados mais testes e medida a consistência do sistema no ambiente virtual é que será instalado na infra-estrutura do ambiente de produção da UCAN.

Resultados Obtidos

Relativamente aos objectivos propostos para este projecto, é importante concluir que foram globalmente atingidos, tendo o sistema desenvolvido tido uma boa aceitação dos utilizadores envolvidos, quer alunos quer docentes, em especial dos órgãos directivos da Universidade, demonstrado o potencial do uso de práticas de *e-learning* no Ensino Superior

Recomendações

Tendo em conta os objectivos alcançados, são propostas algumas recomendações para

trabalho futuro:

- Configuração do sistema para ser visualizado também em dispositivos móveis usando a tecnologia responsiva;
- Integração com o Sistema de Informação da UCAN.

Quadro resumo do trabalho

EXPLICITAÇÃO DO TRABALHO EM QUESTÃO

Tendo em conta a massificação do Ensino Superior em Angola e a necessidade de atender todas as franjas desta sociedade, uma vez que grande parte destas, devido a afazeres profissionais, enfrenta dificuldades em frequentar um Curso Superior Presencial.

Assim sendo, e para colmatar este problema, o presente relatório descreve a implementação do *BackOffice* para um Sistema de Ensino a Distância no intuito de colmatar o problema anteriormente apresentado.

PRESSUPOSTOS

Os pressupostos para a criação do *BackOffice* para o Sistema de Ensino a Distância são:

- A democratização do conhecimento, buscando dar acesso ao saber, às pessoas que tenham dificuldades de usufruir do ensino presencial;
- Prover uma aprendizagem aberta, tendo como pressupostos: flexibilidade, de modo a atender as necessidades do estudante, ao invés de lhe impor conceitos que nem sempre são significativos;
- Modularidade, de maneira a estruturar o curso de acordo com as necessidades específicas da comunidade, de forma dinâmica e com recurso a meios virtuais de aprendizagem;
- Permitir o estudo ou acesso ao conteúdo didático 24/7 tendo como cerne acesso

ao conteúdo pedagógico, segundo a máxima: *anytime, anywhere e everywhere*.

O *BackOffice* para o Sistema de Ensino a Distância será desenvolvido e implementado para a Universidade Católica de Angola, visto ser uma instituição com muitos alunos e de prestígio, o que faz com que uma boa percentagem dos candidatos ao Ensino Superior se dirija a esta academia. O sistema proposto está orientado para os cursos do 1º ciclo (licenciatura) e do 2º ciclo (mestrado).

VANTAGENS

A implementação do Sistema de Ensino à Distância na UCAN trará um conjunto de vantagens, das quais se destacam:

- Previsível aumento exponencial do número de alunos porque uma boa percentagem de pessoas que não estudam no regime presencial, terão maior facilidade em se inscrever neste regime;
- Alto retorno de investimento (ROI): retorno rápido do dinheiro investido tendo em conta que muitos mais estudantes inscritos, implicam um maior valor de serviço.

GESTÃO DE PROJECTOS

FASES	ACTIVIDADES
PLANEAMENTO	
Plano de Execução do Projecto	1. Contacto Inicial com o Cliente 2. Levantamento de Recursos do Projecto 3. Cronograma Inicial
	1. Definição do Sistema

Levantamento de Requisitos	<ol style="list-style-type: none"> 2. Identificação dos Requisitos 3. Análise e Classificação dos Requisitos
Casos de Uso	<ol style="list-style-type: none"> 1. Construção do Diagrama de Casos de Uso 2. Descrição em Alto Nível 3. Descrição em Nível Detalhado 4. Priorização e Escalonamento dos Casos de Usos
CONSTRUÇÃO	
Análise	<ol style="list-style-type: none"> 1. Modelo Conceptual de Classes 2. Diagrama de Estados/Actividades
Projecto	<ol style="list-style-type: none"> 1. Diagrama de Interação/Colaboração 2. Diagrama de Classes do Projecto 3. Esquema da Base de Dados 4. Modelo da Arquitectura do Sistema
Implementação	<ol style="list-style-type: none"> 1. Implementação 2. Segurança
Testes	<ol style="list-style-type: none"> 1. Testes
Implantação	<ol style="list-style-type: none"> 1. Plano de Implementação 2. Pacote de Entrega ao Cliente 3. Treino e formação de utilizadores
Avaliação do Cliente/Manutenção	<ol style="list-style-type: none"> 1. Garantia de Qualidade

Referências

António, P. (2010). *Sistema Integrado de Gestão para o BDA: Módulo de Gestão de Património & Gestão de Utilizadores e Acessos*. Luanda: FEUCAN. 198f. Relatório do

Projecto Final do Curso (Licenciatura em Engenharia Informática).

Barrico, C. (2011). *UML – Modelação de Comportamento – Diagrama de Interação*.

Departamento Informática da UBI. Disponível em

<http://www.di.ubi.pt/~cbarrico/Disciplinas/EngenhariaSoftware/Downloads/Cap%207%20-%20UML%20-%20Modelacao%20do%20Comportamento.pdf>. Acesso em: 10 Jun. 2014.

Dennis, A.; Wixom, B.; Roth, R. (2009). *System Analysis and Design*. 5th ed. Usa: John Wiley & Sons.

Dias, M. (2009). *Projecto de criação do Instituto Superior Técnico - Universidade Virtual no*

Second Life. Lisboa: IST. Disponível em

https://fenix.tecnico.ulisboa.pt/downloadFile/395139481638/tese_manuel_matos_dias_final.pdf. Acesso em: 10 Jun. 2014

Gouveia, L. (1999). Digital support for teachers teaching. Current experience on using Internet facilities in virtual university environments. *Educational Media International*. Journal of ICEM, Vol. 36, nº1, March. Routledge, pp 19-31. ISSN 0952-3987.

Gouveia, L. (2002). Is there any room for face-to-face teaching in a digital world? A proposed framework for web usage. *Educational Media International*. Journal of ICEM, Routledge. Vol 38, n 4, pp 299-305. December 2001. ISSN 0952-3987.

Henderson, C. (2006). *Building scalable web site*. O'Reilly Media.

Jacques, A. (n/d). *Requisitos não funcionais*. Disponível em:

<http://www.dsc.ufcg.edu.br/~jacques/cursos/proj/gerenciadesenv/naofuncionais.htm>.

Acesso em: 18 Mar. 2014

Lima, R. (s/d). *Metodologia de Desenvolvimento de Sistemas de Informação baseados em*

OO. São Paulo: Unibratec. Disponível em: <http://www.unibratec.edu.br/tecnologus/wp->

content/uploads/2007/08/n2_lima_rr.pdf. Acesso em: 31 Mai. 2014

Lins, F. (n/d). Requisitos não-funcionais em Web Services. Disponível em:

http://www.cin.ufpe.br/~in1020/arquivos/monografias/20081/monografia_fernando_aires.pdf. Acesso em: 20 Fev. 2014

Martins, P. (2013). *BackOffice para plataformas web de Gestão de Conferências Científicas*. Desenvolvimento e Avaliação. Porto. Disponível em:

http://paginas.fe.up.pt/~tavares/downloads/publications/teses/MSc_Pedro_Martins.pdf.

Acesso em: 24 Mar. 2014.

Morais, É. (2007). *Metodologia e Diretrizes para desenvolvimento de um Ambiente Computacional para Ensino a Distância*. São Paulo: Universidade Estadual Paulista.

Disponível em *http://www.feis.unesp.br/Home/departamentos/engenhariaeletrica/pos-graduacao/208-dissertacao_eric_vasconcelos_de_morais.pdf*. Acesso em: 8 Jun. 2014

Rocha, Á. (2008). *O Essencial da Análise de Sistema*. Porto: UFP, 2008. Disponível em

http://pt.scribd.com/doc/200962145/O-Essencial-da-Analise-de-Sistemas. Acesso em: 10 Jun. 2014

Rodrigues, J. (2007). *mSynapse: Uso de m-Learning no Ensino Superior*. Aveiro:

Universidade de Aveiro. *https://ria.ua.pt/bitstream/10773/1533/1/2008001205.pdf*. Acesso em: 15 Mar. 2014.

Roques, P. (2004). *UML in Practice*. England: John Wiley & Sons.

Tavares, F. (2012). *Desenvolvimento de Aplicações em PHP*. Lisboa: FCA.

University of Mumbai (s/d). *System Analysis and Design*. Disponível em

http://www.mu.ac.in/myweb_test/MCA%20study%20material/M.C.A.%20%28Sem%20-I%29%20Paper%20-%20II%20-%20System%20Ananalysis%20and%20Design.pdf

Acesso em: 6 Jun. 2014.

Silva, Hugo; Nascimento, V. (2007). *Relatório de Laboratório/Projecto 2*. ESOFTE. Porto: Isel, 2007. Disponível em www.dei.isep.ipp.pt. Acesso em: 10 Jun. 2014.

Alves, P. et al. (s/d). *Estudo Sobre a Utilização das Ferramentas do Ambiente Sakai numa Instituição de Ensino Superior*. Bragança: Instituto Politécnico de Bragança. Disponível em: https://bibliotecadigital.ipb.pt/bitstream/10198/8763/1/CL2013Tcai_cap11.pdf. Acesso em: 10 Jun. 2014.

Anexo A – Plano de execução do projecto

SISTEMA DE GESTÃO DE ENSINO-APRENDIZAGEM

A.1 CONTATO INICIAL COM O CLIENTE

Identificação do Cliente

- Nome do Cliente/Sigla: UCAN – Universidade Católica de Angola
- Endereço: Avenida Pedro de Castro Van-Dúnem Loy nº 24, Luanda - Angola
- Fone/Fax: 244 304060
- Dirigentes Actuais e respectivos cargos:
 - Vicente Cacuchi - Reitor
 - Jerónimo Cahinga – Vice-Reitor para Investigação
 - Maria Helena Miguel – Vice-Reitora para Área Académica
- Ramo de actuação: Instituição de Ensino Superior.
- Região de actuação: Norte - Luanda
- Infraestrutura Tecnológica: Possui um “Parque Informático” com mais de 400 PCs distribuídos pela área administrativa, académica e laboratórios.
- Responsável pelo Levantamento: Paulo Francisco António

A.2 IDENTIFICAÇÃO DO SERVIÇO SOLICITADO

Título: *BackOffice* do Sistema de Ensino à Distância da Universidade Católica de Angola – Administração de Aulas à Distância Usando o Ambiente Virtual (Internet)

Conceptualização Geral (Objetivos): Permitir a inscrição dos estudantes com o Curso Médio e Licenciatura concluídos nos Cursos de Licenciatura e Mestrado respectivamente.

Problemas relatados pelo utilizador:

- Inexistência de instituições que leccionam o Ensino à Distância em Angola e baixo número de vagas para o regime presencial.
- Massificar e possibilitar o ensino para as pessoas impedidas de frequentar o ensino presencial

Necessidades do utilizador: Possibilidade de continuar os estudos de Licenciatura ou Mestrado a partir de qualquer lugar e hora usando as Tecnologias de Informação e Comunicação

Informações complementares (Opcional)

Resistência do utilizador: Nenhuma;

Nível do Pessoal: Nível Médio e Superior incompleto/completo.

Condições do local: Óptima.

Directrizes para o serviço: Análise, Desenvolvimento e Implantação.

Responsável pelo levantamento: Paulo Francisco Antonio

A.3 LEVAMENTAMENTO DE RECURSOS HUMANOS DO PROJECTO (SISTEMA)

RECURSOS HUMANOS	PERFIL
01 – Coordenador (Project Leader)	- Nível Superior; - Facilidade de comunicação; - Capacidade de Liderança; - Capacidade de Negociação; - Personalidade carismática; - Organizado.
02 - Designer	- Nível Superior; - Facilidade de comunicação; - Conhecimento em Artes, produção gráfica e animação; - Organizado;

	- Extrovertido.
03 – Profissionais Web Master/ Web Design / Analista /Administrador de Sistemas/ Desenvolvedor (Programador)	- Nível Superior; - Facilidade de comunicação; - Conhecimento de administração - Personalidade carismática; - Organização

A.4 CRONOGRAMA DE IMPLEMENTAÇÃO DO BACKOFFICE DO SISTEMA DE ENSINO À DISTÂNCIA DA UCAN

Nº	ACTIVIDADE	DURAÇÃO
01	Levantamento e Contratação de Recursos Humanos para o Projecto	2 Semanas
02	Levantamento de Requisitos -Identificação dos Requisitos funcionais -Análise e Classificação dos requisitos -Identificação dos Requisitos não funcionais	4 Semanas
03	Casos de Uso -Construção do Diagrama de Casos de Uso; -Descrição em Alto Nível; -Descrição em Nível Detalhado; -Priorização e Escalonamento dos Casos de Uso	4 Semanas
04	Desenvolvimento do Sistema -Criação do Diagrama de Entidade Relacionamento -Criação do Diagrama de Fluxos de Dados -Criação do Dicionários de Dados -Definição e Modelagem da Base de Dados	2 Semanas

	-Criação do Diagrama de Interacção -Criação do Diagrama de Classes	
05	Definição da Arquitectura do Sistema	4 Semanas
06	Instalação e Configuração da Arquitectura do Sistema (Servidores) -Instalação e Configuração do Servidor de Base de Dados -Instalação e Configuração do Servidor Aplicacional -Instalação e Configuração do Servidor <i>Web</i>	2 Semanas
IMPLEMENTAÇÃO DO SISTEMA		
07	Instalação e Configuração do Sistema de Ensino à Distância na Infraestrutura de Produção	4 Semanas
IMPLEMENTAÇÃO DO SISTEMA – IMPLEMENTAÇÃO DOS MÓDULOS		
08	#GERIR ÁREA PESSOAL DO ALUNO	2 Semanas
09	#GERIR UNIDADES CURRICULARES	2 Semanas
10	#GERIR CURSOS	1 Semana
11	#GERIR TURMAS	1 Semana
12	#GERIR AULA VIRTUAL	3 Semanas
13	#GERIR FÓRUNS	2 Semanas
14	#GERIR UTILIZADORES E ACESSOS	4 Semanas
15	#GERIR RECURSOS	3 Semanas
16	#GERIR MENSAGENS	2 Semanas
17	#GERIR CHATS	2 Semanas
18	#GERIR ANÚNCIOS	2 Semanas
19	#GERIR PROFESSOR	2 Semanas

20	Segurança -Criação de uma DMZ	4 Semanas
21	ADAPTAÇÃO DE IDENTIDADE GRÁFICA -Elaboração de proposta de <i>design</i> adaptado à identidade gráfica do cliente para sua aprovação; -Implementação de <i>design</i> aprovado no Sakai. Isto inclui a colocação de imagens identificativas do cliente, logótipos, rodapé, cabeçalho, cores, desenvolvimento de CSS, etc. -Introdução de textos, configuração de menus, etc. no Sakai	3 Semanas
TESTES		
22	Testes de Estabilidade, consistência e robustez	3 Semanas
23		
AVALIAÇÃO DO CLIENTE/MANUTENÇÃO		
24	MANUTENÇÃO EVOLUTIVA -Análise dos novos requisitos -Desenvolvimento e integração de novas funcionalidades no <i>software</i> -Disponibilização do software num servidor de pré-produção, para a análise e validação dos desenvolvimentos pelo cliente -Actualização do <i>software</i> na infraestrutura de produção.	6 Semanas
TEMPO DE EXECUÇÃO		14 Meses