

Mestrado em Engenharia Informática
Estágio
Relatório Final

Apoio ao Desenvolvimento do Projeto NONIO

Sérgio Dias Martins
sdmart@student.dei.uc.pt

Orientador:
Professor Doutor Marco Vieira
Data: 12 de Julho de 2012



FCTUC DEPARTAMENTO
DE ENGENHARIA INFORMÁTICA
FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Resumo

O trabalho no primeiro semestre deste estágio contribuiu para preparar o NONIO a fim de suportar futuramente a leccionação em modo e-learning. Foi elaborado um Documento de Visão e um estudo sobre reproduzir ficheiros multimédia. Com base neste último é agora possível reproduzir conteúdo multimédia diretamente a partir do NONIO. Foi também desenvolvida a funcionalidade Diário, útil para comunicação privada entre cada aluno e os seus docentes no contexto das disciplinas.

O trabalho no segundo semestre consistiu no desenvolvimento da funcionalidade Requerimentos no NONIO. Tratou-se da informatização de um processo já existente e de grande importância para o funcionamento dos Serviços de Gestão Académica da Universidade de Coimbra. Agora este frequente processo pode ser realizado de forma mais acelerada e prática, o que contribui para um aumento de eficiência nos serviços de atendimento aos alunos.

Durante o seu período de estágio, o estagiário esteve integrado na equipa de desenvolvimento do projeto NONIO tendo utilizado o Processo de Desenvolvimento de Software adotado pela equipa, bem como uma vasta gama de tecnologias e ferramentas. O trabalho que realizou capacitou-o para o seu futuro profissional como Engenheiro Informático.

Palavras-Chave

decisão, diário, educação à distância, e-learning, multimédia, NONIO, ofício, parecer, requerimento, responsável

Índice

Resumo.....	2
Palavras-Chave	2
Lista de Figuras	4
Lista de Tabelas	4
Acrónimos.....	5
Agradecimentos.....	6
Capítulo 1 Introdução	7
Capítulo 2 Enquadramento	9
2.1. Processo de Desenvolvimento de Software.....	9
2.2. Arquitetura.....	12
2.3. Tecnologias.....	13
Capítulo 3 Atividades Realizadas.....	15
3.1. E-learning.....	17
3.2. Desenvolvimento da funcionalidade Diário.....	19
3.3. Desenvolvimento da funcionalidade Requerimentos	24
Capítulo 4 Outras Atividades	32
4.1. Reuniões.....	32
4.2. Formações.....	33
4.3. Contato com o cliente.....	34
4.4. Testes funcionais	34
4.5. Pedidos de alteração.....	34
4.6. Manutenção	34
Capítulo 5 Conclusões e Futuro	35
Anexos.....	38
Referências e Bibliografia.....	40

Lista de Figuras

Aqui são listadas todas as figuras utilizadas neste relatório:

Figura 1 – PDS utilizado no projeto NONIO

Figura 2 – Ciclo de vida de um documento no projeto NONIO

Figura 3 – Relação entre as tecnologias e a arquitetura lógica da aplicação NONIO

Figura 4 – Diagrama de Gantt com as atividades realizadas

Figura 5 – Modelo de dados da funcionalidade Diário

Figura 6 – Modelo de dados da funcionalidade Requerimentos

Lista de Tabelas

Aqui são listadas todas as tabelas utilizadas neste relatório:

Tabela 1 – Acrónimos utilizados neste relatório e respetivo significado

Tabela 2 – Estatísticas de utilização da funcionalidade Diário

Tabela 3 – Distribuição da implementação dos ecrãs da funcionalidade Requerimentos

Tabela 4 – Principais reuniões realizadas no âmbito do estágio

Acrónimos

Aqui são listados todos os acrónimos utilizados ao longo deste relatório:

Acrónimo	Significado
AGA	Aplicação de Gestão Académica
CCA	Comissão de Controlo de Alterações
CSS	Cascading Style Sheets
CVS	Concurrent Version System
DAO	Data Access Object
DEA	Documento de Especificações Adicionais
DEI	Departamento de Engenharia Informática
DTO	Data Transfer Object
E-R	Entidade-Relacionamento
FCTUC	Faculdade de Ciências e Tecnologia da UC
GSIIIC	Gestão de Sistemas e Infra-estruturas de Informação e Comunicação da UC
HQL	Hibernate Query Language
HTML	HyperText Markup Language
IDE	Integrated Development Environment
IoC	Inversion of Control
IPN	Instituto Pedro Nunes
JSP	Java Server Pages
JSTL	Java Server Pages Standard Tag Library
MEI	Mestrado em Engenharia Informática
MOODLE	Modular Object-Oriented Dynamic Learning Environment
MVC	Model-View-Controller
NONIO	NOvo sistema académico uNiversitárIO
ORM	Object Relational Mapping
PA	Pedido de Alteração
PCA	Plano de Controlo de Alterações
PDF	Portable Document Format
PDS	Processo de Desenvolvimento de Software
PGQ	Plano de Garantia de Qualidade
POJO	Plain Old Java Objects
SGA	Serviços de Gestão Académica
SQL	Structured Query Language
UC	Universidade de Coimbra
WoC	Web on Campus
XML	eXtensible Markup Language

Tabela 1 – Acrónimos utilizados neste relatório e respetivo significado

Agradecimentos

Em primeiro lugar, agradeço aos meus orientadores de estágio, o Professor Marco Vieira e o Eng.º Pedro Pinto, não só pela oportunidade concedida em estagiar no projeto NONIO e de tudo o que isso significa, mas também pela disponibilidade e prontidão sempre demonstrada nos momentos em que recorri à sua orientação.

Agradeço também a toda a equipa de desenvolvimento do projeto NONIO pelo apoio prestado no esclarecimento de dúvidas e também pelo agradável ambiente de trabalho, fatores que suavizaram bastante a minha integração na equipa desde muito cedo no estágio.

Aos muitos e diversos colegas de curso com quem tive oportunidade de trabalhar ao longo da minha passagem pela UC, com os quais aprendi imenso e ganhei uma das competências essenciais para o futuro: saber trabalhar em equipa. Um agradecimento especial é direcionado para os colegas que no ano letivo anterior me apoiaram numa fase menos boa, e me deram o necessário incentivo para que conseguisse concluir o meu percurso académico na UC.

Por fim, agradeço à minha família mais próxima, nomeadamente os meus pais, irmã e cunhado, por estarem presentes e pelo apoio a diversos níveis.

A todos, muito obrigado!

Capítulo 1

Introdução

Este relatório tem como principal objetivo apresentar o trabalho realizado pelo aluno Sérgio Dias Martins, durante o ano lectivo 2011/2012, no âmbito da unidade curricular de “Dissertação/Estágio” do Mestrado em Engenharia Informática (MEI) da Universidade de Coimbra (UC) [1]. O estágio com o tema “Apoio ao Desenvolvimento do Projeto NONIO” realizou-se no edifício central da Faculdade de Ciências e Tecnologia da Universidade de Coimbra (FCTUC) [2], sob orientação do Professor Doutor Marco Vieira, docente do Departamento de Engenharia Informática (DEI) [3], e do Eng.º Pedro Pinto, gestor do projeto NOvo sistema académico uNIversitário (NONIO).

O projeto NONIO foi iniciado em meados de 2006, tendo como principal objetivo a criação de uma aplicação única para a gestão académica de toda a UC. Nessa altura tornou-se também necessário ter um sistema capaz de suportar a adopção do processo de Bolonha no ensino superior. A sua génese teve lugar na FCTUC onde a gestão académica era realizada por várias aplicações separadas: a Aplicação de Gestão Académica (AGA), o sistema interno de pautas e a InforEstud@nte. Essas aplicações estavam implementadas em diferentes tecnologias e não foram estruturadas para permitir facilmente a adaptação a mudanças de requisitos. A criação de uma nova aplicação foi a opção escolhida na altura. Aproveitou-se também essa oportunidade para informatizar e consolidar processos de gestão académica. Desde meados de 2009 o NONIO começou o processo de expansão para as restantes faculdades da UC, visto ser esse um dos objetivos iniciais. Também começaram a ser desenvolvidas funcionalidades de gestão pedagógica para dar apoio aos docentes e alunos no contexto das disciplinas, substituindo assim a utilização do sistema Web on Campus (WoC) [4] por funcionalidades semelhantes dentro do NONIO. Atualmente em 2012, o NONIO é a aplicação utilizada em toda a UC para a gestão académica dos alunos matriculados nesta instituição, e tem também algumas funcionalidades de gestão pedagógica.

Este estágio foi criado visando dar um passo importante numa área específica: e-learning. O NONIO como sistema de informação que é e com a sua relevância na leccionação em modo presencial dos cursos da UC, apresenta-se com potencial para também suportar a leccionação em modo e-learning. O âmbito deste estágio no primeiro semestre foi contribuir para preparar o NONIO a fim de suportar futuramente a leccionação em modo e-learning. Para isso seria necessário o desenvolvimento de novas funcionalidades de gestão pedagógica, eventuais alterações a funcionalidades já existentes, bem como trabalho na vertente logística para permitir candidaturas e matrículas em cursos e-learning. O estagiário ingressou na equipa de desenvolvimento no início do presente ano letivo, sendo composta por dez elementos a tempo integral. No primeiro semestre era esperado que o estagiário se integrasse na equipa, e tivesse contato com o Processo de Desenvolvimento de Software (PDS) e as tecnologias utilizadas, através do desenvolvimento de uma ou mais funcionalidades com algum nível de complexidade, para que no segundo semestre conseguisse realizar maior volume de trabalho.

De acordo com esse âmbito o estagiário elaborou um Documento de Visão que teve o objectivo de saber que alterações seriam necessárias à aplicação NONIO para que consiga suportar futuramente a educação à distância. Elaborou também um estudo sobre reproduzir ficheiros multimédia, com o objetivo de saber que opções estão atualmente disponíveis para se realizar essa ação, tendo indicado e justificado a opção escolhida. Desenvolveu ainda uma funcionalidade completa denominada Diário, útil para a comunicação privada entre cada aluno e os seus docentes no contexto das disciplinas em que se encontra inscrito.

No segundo semestre o âmbito deste estágio foi alterado devido ao surgimento de uma nova funcionalidade com elevada prioridade para desenvolver: gestão de requerimentos através do NONIO. Um requerimento é basicamente um pedido feito por um estudante, ex-estudante, ou candidato perante os Serviços de Gestão Académica (SGA) com a intenção de obter aprovação para a situação que apresenta. Podem ser de diversos tipos tais como admissões a provas, atribuições de estatutos, inscrições em exames, reapreciações, reclamações, etc. Antes de a decisão ser registada cada requerimento poderá ter ou não pareceres que não são mais do que opiniões. Aquando da solicitação da funcionalidade Requerimentos já existia no NONIO a funcionalidade Pedidos, apenas disponível para utilização na InforGestão onde se podia registar cada requerimento e respetiva decisão. Geralmente existia também um requerimento em papel que circulava pelos intervenientes tornando o processo demorado, pouco prático e pouco eficiente. Com esta nova funcionalidade é possível acelerar o processo, visto deixar de ser necessária a circulação do requerimento físico pelos intervenientes, e torná-lo prático visto que toda a informação fica registada no NONIO e disponível a todos os intervenientes. Além disso, também se consegue melhorar os serviços de atendimentos aos alunos, visto que eles podem consultar os seus próprios requerimentos na aplicação e mais tarde poderão também registá-los, reduzindo as deslocações até aos SGA. Outro aspeto que demonstra claramente a importância desta funcionalidade é o seu volume, todos os dias dezenas de novos requerimentos são registados na aplicação, e com as vantagens desta nova funcionalidade a utilização por certo irá aumentar.

Perante esta nova prioridade o estagiário decidiu deixar de trabalhar no tema do e-learning e dedicar-se a esta nova funcionalidade, estando as razões para esta decisão detalhadas mais à frente no relatório. Por isso, no segundo semestre o estagiário participou em atividades de desenvolvimento da funcionalidade Requerimentos, passando novamente por todas as fases do PDS utilizado no projeto NONIO. Para o segundo semestre era esperado que o estagiário aprofundasse os conhecimentos adquiridos no primeiro de modo a produzir trabalho com um maior volume e complexidade, e dessa forma conseguir ganhar competências na área de desenvolvimento de software. Conforme será demonstrado ao longo do relatório, o estagiário alcançou este objectivo.

O presente relatório foi estruturado de forma a facilitar a sua compreensão. Começa com as secções de resumo e palavras-chave, úteis para se obter rapidamente uma percepção do estágio e do seu âmbito. A introdução é o primeiro e atual capítulo do relatório onde se fornecem algumas informações relacionadas com o projeto NONIO, o estágio e o papel do estagiário. O capítulo de enquadramento faculta informação relacionada com os processos e tecnologias utilizadas no projeto. Nas atividades realizadas é descrito o trabalho realizado pelo estagiário em todo o período de estágio, juntamente com um diagrama de Gantt, sendo dada especial ênfase às duas funcionalidades desenvolvidas pelo estagiário: Diário e Requerimentos. A parte descritiva do trabalho é complementada com algumas reflexões. Em outras atividades destacam-se tarefas não menos importantes realizadas de forma paralela com o desenvolvimento das funcionalidades. Nas conclusões é feito um balanço final sobre o trabalho realizado pelo estagiário, passando pelo cumprimento dos objectivos esperados, quais os benefícios resultantes a nível pessoal, e como decorreu a interação com a equipa. São também destacadas decisões, desafios/dificuldades e lições, concluindo com algumas breves referências ao futuro profissional do estagiário. Além destes capítulos, existem também outras secções de complemento tais como a tabela de acrónimos e as listas de figuras, tabelas, anexos e referências bibliográficas.

Capítulo 2

Enquadramento

Este capítulo visa facultar alguma informação relacionada com os processos e as tecnologias utilizadas no projeto NONIO para facilitar a compreensão do restante do documento. A maioria do conteúdo neste capítulo foi resumida dos documentos de processo utilizados pela equipa, que podem ser encontrados nos anexos 1. Para cada figura ou conteúdo provenientes desses anexos é feita uma referência específica ao anexo do qual provém.

2.1. Processo de Desenvolvimento de Software

O PDS utilizado no projeto NONIO encontra-se pormenorizadamente descrito no anexo 1.1 “Nonio_ProcessoDesenvolvimento_2009-06-11_v1.2r.pdf”. A figura 1, retirada do referido anexo, menciona os documentos de processo, as atividades de gestão, e as várias fases junto com os artefactos específicos de cada uma.



Figura 1 – PDS utilizado no projeto NONIO

Os documentos de processo foram elaborados no início do projeto, e neles são descritos processos a respeitar por toda equipa com o objetivo de facilitar a gestão e assegurar a qualidade dos artefactos produzidos. As atividades de gestão são da responsabilidade do gestor de projeto, e consistem na criação e manutenção dos documentos indicados.

Fases

A fase de especificação é a fase inicial do PDS e nela são criados três artefactos que servem de base para as fases seguintes:

- **Protótipos** – Definição dos ecrãs em páginas HyperText Markup Language (HTML) estáticas, com algum eventual JavaScript e CSS, com os dados a apresentar em cada ecrã, demonstrando também o comportamento funcional e navegação entre ecrãs;
- **Documento de Especificações Adicionais (DEA)** – Documento escrito que complementa o protótipo, e que explica aspectos funcionais que não são se consegue descrever claramente através do protótipo;
- **Modelo de dados** – Diagrama Entidade-Relacionamento (E-R) que representa conceptualmente a base de dados;

Na fase de design outros três artefactos são considerados:

- **Arquitetura e design genérico** – Apenas é atualizado quando alguma situação o justifique (anexo 1.4 “Nonio_ArquitecturaDesignGenerico_2009-01-24_v1.4r.pdf”);
- **Design detalhado da base de dados** – Apenas é atualizado quando alguma situação o justifique;
- **Design detalhado dos módulos** – É normalmente o único documento que é criado ou atualizado nesta fase. O facto de estar a cor laranja demonstra o carácter opcional desta tarefa.

A fase de construção envolve dois artefactos:

- **Código da aplicação** – Escrita do código da funcionalidade e integração na aplicação;
- **Código dos testes regressivos** – Quando se decide realizar testes regressivos é nesta fase que são codificados para serem aplicados na fase de lançamento. Tal como o design detalhado dos módulos, a cor laranja demonstra o carácter opcional desta tarefa.

A fase de lançamento envolve os seguintes artefactos:

- **Relatórios de testes funcionais** – Estes testes são aplicados de forma manual, e um relatório deve ser preenchido com informação detalhada sobre cada erro encontrado, bem como identificação da pessoa que testou e quando;
- **Relatórios de testes regressivos** – Quando considerados, estes testes são aplicados de forma automática, com recurso a uma ferramenta de automatização de testes, sendo obtidos relatórios no formato HTML;
- **Aplicação** – Disponibilização da funcionalidade aos utilizadores finais, desde o ambiente de desenvolvimento para o ambiente de produção.

A fase de manutenção ocorre após a funcionalidade estar disponibilizada, quando os utilizadores finais podem reportar defeitos para correção, ou nota-se que alguns aspetos podem ser aprimorados. Essas pequenas alterações, depois de concretizadas, são disponibilizadas através de atualizações das aplicações em ambiente de produção.

Documentos transversais a todo o PDS

Existem dois documentos transversais a todo o PDS. O primeiro documento é o Plano de Controlo de Alterações (PCA), (anexo 1.2 “Nonio_PlanoControloAlteracoes_2009-06-11_v1.2r.pdf”), onde são especificados os possíveis estados que um documento produzido no âmbito do projeto pode-se encontrar num determinado momento. Especifica também outros detalhes tais como regras para formatação, atribuição de nomes e numeração de versões nos documentos. Esses estados são apresentados na figura 2, retirada do referido anexo.

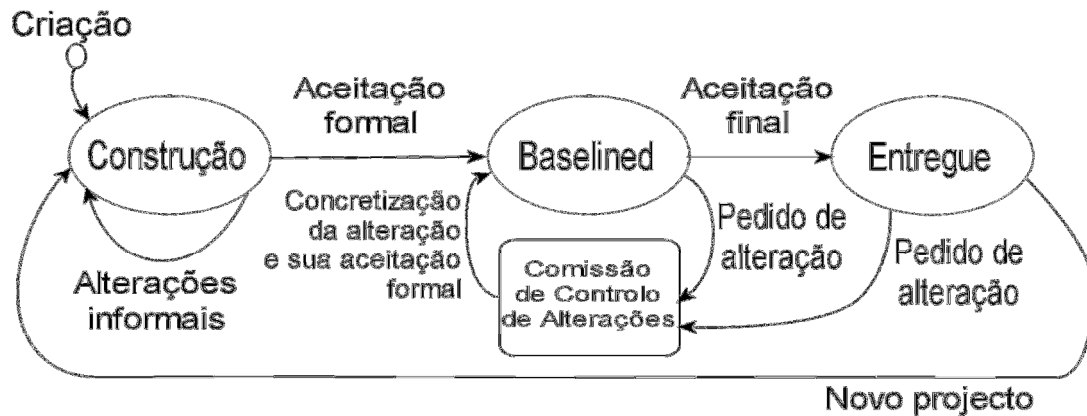


Figura 2 – Ciclo de vida de um documento no projeto NONIO

De seguida é explicado brevemente cada um desses estados:

- **Construção** – Após ter sido criado, o(s) autor(es) procede(m) à escrita e alteração do conteúdo do documento de forma informal;
- **Baselined** – Depois de passar pelo processo de aceitação formal e ter sido aprovado. Nessa altura é também colocado no repositório do projeto. Quaisquer eventuais alterações são indicadas num Pedido de Alteração (PA) para ser avaliado pela Comissão de Controlo de Alterações (CCA), podendo essas alterações serem aplicadas ou descartadas;
- **Entregue** – Depois de passar pelo processo de aceitação final e ter sido aprovado. Os processos de aceitação formal e aceitação final encontram-se descritos no Plano de Garantia de Qualidade.

O segundo documento é o Plano de Garantia de Qualidade (PGQ) (anexo 1.3 “Nonio_PlanoGarantiaQualidade_2009-01-10_v1.5r.pdf”), onde são indicados os processos adoptados pela equipa para assegurar a qualidade dos artefactos produzidos. Os processos podem ser formais (quando é necessário guardar um registo escrito) ou informais. Os processos também podem ser divididos nos que se aplicam a qualquer tipo de artefacto:

- **Inspecções** – Trata-se de um método formal, rigoroso e manual de revisão;
- **Revisões** – Igualmente formal, mas simplificado em relação às inspecções;
- **Reuniões de análise** – Têm como objetivo detetar erros e omissões, e alternativas também podem ser discutidas.

e os que se aplicam apenas a código:

- **Testes unitários** – São aplicados pelo autor do código, ao nível dos métodos e procedimentos que desenvolveu;
- **Testes de integração** – São também aplicados pelo autor do código, com a intenção de averiguar a correta integração das funcionalidades que desenvolveu com as já existentes;
- **Testes funcionais** – Podem ser aplicados pelo autor do código e por outros elementos da equipa, com a intenção de averiguar a correta implementação das funcionalidades desenvolvidas, tendo como base o ponto de vista da interface;
- **Testes regressivos** – São realizados com recurso a uma ferramenta de automatização de testes.

2.2. Arquitetura

O conteúdo desta secção e da próxima provém do anexo 1.4 “Nonio_ArquitecturaDesignGenerico_2009-01-24_v1.4r.pdf”. Existem diferentes tipos de utilizadores que acedem ao sistema NONIO, e para cada um deles existe uma vista específica. A InforEstud@nte é utilizada pelos estudantes, a InforDocente é utilizada pelos docentes, e a InforGestão é utilizada pelos SGA.

Além da vertente física da arquitetura (componentes físicos, distribuição e comunicação) existe também a vertente lógica que é constituída por quatro camadas, que se encontram apresentadas na figura 3, retirada do anexo 1.4.

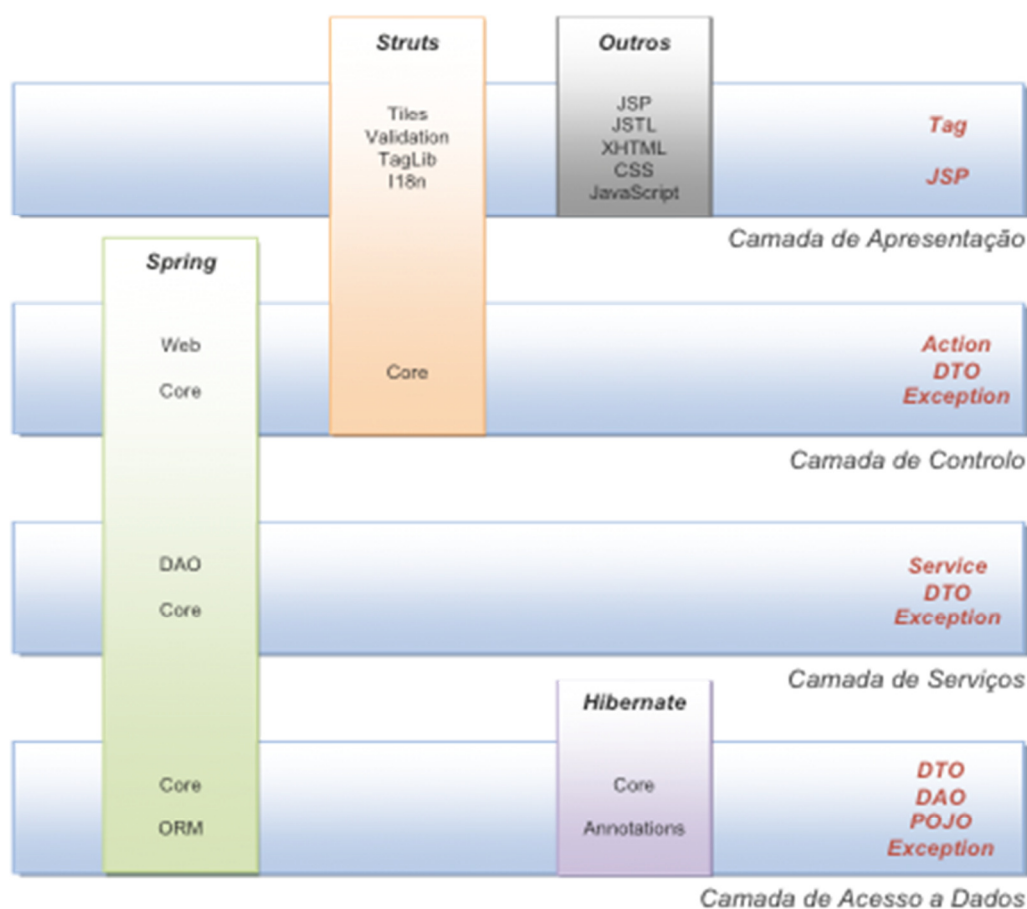


Figura 3 – Relação entre as tecnologias e a arquitetura lógica da aplicação NONIO

A camada de apresentação consiste na interface gráfica onde ocorre uma interação bidirecional entre os utilizadores e a aplicação. A camada de controlo contém ações que tratam os pedidos vindos da camada de apresentação, se necessário enviam informação para as camadas inferiores podendo também receber informação de volta, e redirecionam o utilizador para o ecrã adequado. Na camada de serviços pode ser encontrada a lógica de negócio da aplicação, onde é feita a ligação entre a camada de acesso a dados e as camadas superiores. A camada de acesso a dados contém as classes que permitem abstrair o conteúdo da base de dados.

Existem três formas para realizar a comunicação entre as camadas, sendo cada uma delas adequada para uma situação específica que envolve o conteúdo da base de dados: push-up para obter informação, pull-down para guardar informação e remove para remover informação.

2.3. Tecnologias

A figura 3 também relaciona as quatro camadas da arquitetura lógica com as tecnologias utilizadas no projeto NONIO. A linguagem de programação Java [5] está na base de todo o projeto. No entanto, três frameworks principais são utilizadas:

- **Apache Struts** [6] – Implementa o padrão de desenho Model-View-Controller (MVC) que permite uma separação entre os ecrãs da interface, a lógica de negócio, e o acesso à base de dados. Na aplicação NONIO são utilizados os componentes Controller e View. Além disso, esta framework disponibiliza um conjunto de módulos bastante úteis que são utilizados no NONIO e estão apresentados na figura 3. Além dos ficheiros Java são também utilizados ficheiros eXtensible Markup Language (XML) [7] para a definição de diversas configurações;
- **Spring Framework** [8] – Implementa o padrão de desenho Inversion of Control (IoC) que permite uma separação entre as várias camadas que constituem a aplicação. É utilizada em três das quatro camadas da aplicação, e dos vários módulos que disponibiliza os utilizados no NONIO estão apresentados na figura 3;
- **Hibernate** [9] – É um Object Relational Mapping (ORM) que permite representar a informação de uma base de dados relacional numa hierarquia de objetos. Disponibiliza também o Hibernate Query Language (HQL) que é uma linguagem própria para realizar consultas à base de dados, baseada em Structured Query Language (SQL) mas orientada a objetos. Existem ainda as Annotations que mapeiam os campos das tabelas em atributos nas classes.

A camada de apresentação utiliza um conjunto de tecnologias específicas. Java Server Pages (JSP) [10] são páginas Web dinâmicas com elementos que permitem definir a respetiva estrutura, tendo semelhanças com as páginas HTML. Java Server Pages Standard Tag Library (JSTL) [11] é uma biblioteca para manipular conteúdo nos JSP e contém instruções para realizar condições, iterações, etc. O JavaScript [12] é uma linguagem de scripting para adicionar interatividade e animações em páginas Web. O jQuery [13] é uma biblioteca de funções JavaScript que permite adicionar interatividade e animações mais complexas de forma mais rápida e simples do que com a tradicional utilização de JavaScript. Cascading Style Sheets (CSS) [14] permite definir a apresentação e o estilo do conteúdo presente nos JSP.

A framework Struts disponibiliza vários módulos, alguns deles são utilizados na aplicação. O módulo Tiles permite definir a estrutura das páginas apresentadas aos utilizadores, através da definição de blocos comuns. O Validation disponibiliza mecanismos de validação que podem ser aplicados sobre campos de formulários. O TagLib fornece um conjunto predefinido de taglibs para facilitar a construção das páginas. Na prática o i18n permite a definição de vários ficheiros ApplicationResources para os vários idiomas que se pretende traduzir os conteúdos estáticos das páginas.

Existem também alguns tipos de objetos específicos em cada camada. As Action são as classes que recebem os pedidos do utilizador na camada de apresentação, e consoante o pedido pode ser necessário aceder a uma classe de serviços ou não. As Service são classes com métodos invocados a partir de classes na camada de controlo, que por sua vez invocam métodos nas classes DAO para obtenção de dados, podendo realizar algum processamento sobre eles antes de os enviar à camada de controlo. Ao passo que a classe Data Access Object (DAO) é uma interface tendo apenas a declaração dos métodos, a classe DAOImpl tem a declaração e respetiva implementação de cada método. São invocados a partir das classes da camada de serviços, e por sua vez invocam consultas anotadas nos POJO, devolvendo os resultados obtidos para a camada de serviços. Os Plain Old Java Objects (POJO) [15] representam sob a forma de classes e atributos a informação guardada na base de dados em tabelas e campos. Podem também ter

consultas HQL anotadas neles. Os Data Transfer Object (DTO), HqlItem e Item são classes que permitem agregar e transferir entre camadas a informação resultante de consultas à base de dados. As Exception podem ser as comuns exceções em Java ou outras específicas na aplicação.

Por fim, faltam referir algumas ferramentas também utilizadas no projeto NONIO. O Adobe Dreamweaver [16] é utilizado para se elaborar os ecrãs do protótipo. O Microsoft Visio [17] utiliza-se para elaborar os designs detalhados dos módulos. O Power Designer [18] serve para se definir o modelo de dados da aplicação. O Eclipse [19] é o Ambiente de Desenvolvimento Integrado, ou Integrated Development Environment (IDE) em inglês, utilizado para escrita do código. O Apache Tomcat [20] é o servidor aplicacional. Como motor de base de dados é utilizado o Oracle [21]. Para se aceder e manipular o conteúdo da base de dados de desenvolvimento utiliza-se o SQL Developer [22]. O Concurrent Versions System (CVS) [23] é o sistema de controlo de versões utilizado para gestão da documentação e código produzidos.

Capítulo 3

Atividades Realizadas

Este capítulo apresenta as atividades realizadas pelo estagiário ao longo de todo o período de estágio.

Nos primeiros dias de Setembro e em alguns dias do final de Outubro, o estagiário realizou atividades de ambientação ao projeto, que envolveram a leitura dos documentos de processo, o estudo das tecnologias, a configuração do ambiente de trabalho, etc. Estas tarefas foram importantes para o estagiário visto que não tinha trabalhado no projeto no passado.

Conforme referido na introdução deste relatório, o âmbito do estágio foi diferente em cada um dos semestres. O âmbito no momento da criação da proposta de estágio e durante todo o primeiro semestre foi o e-learning. De acordo com este âmbito o estagiário elaborou em Setembro um Documento de Visão que pode ser encontrado no anexo 2.1 “e-learning@NONIO_DocumentoVisao_2011-09-20.pdf”. Em Outubro elaborou um estudo sobre reproduzir ficheiros multimédia que pode ser encontrado no anexo 2.2 “e-learning@NONIO_ReproduzirFicheirosMultimedia_2011-10-10.pdf”. O conteúdo de ambos os documentos é descrito na secção 3.1 deste relatório.

No restante do primeiro semestre e durante todo o segundo, o estagiário esteve envolvido em atividades de desenvolvimento de funcionalidades seguindo o PDS utilizado no projeto NONIO. De Novembro a Fevereiro desenvolveu a funcionalidade Diário, e de Março a Junho desenvolveu a funcionalidade Requerimentos. Visto que a grande maioria do período temporal deste estágio foi dedicado a estas atividades, elas são detalhadas nas secções 3.2 e 3.3 deste relatório, sendo a explicação do desenvolvimento de cada funcionalidade dividida pelas diversas fases do PDS.

Além da descrição do trabalho realizado, são também apresentadas algumas reflexões acerca desse trabalho, a um nível específico na parte final de cada secção. Essas reflexões pretendem destacar a evolução gradual do estagiário, demonstrando o que aprendeu, o que fez bem ou o que poderia ter feito melhor. Este exercício de reflexão foi importante porque permitiu ao estagiário ter consciência do seu desempenho durante o estágio, e permitiu-lhe identificar o que manter e o que melhorar futuramente.

Embora o estagiário tenha escrito o seu relatório intermédio e final de estágio um pouco ao longo de todo o primeiro e segundo semestre, respectivamente, na fase final de cada um dos semestres teve ainda de proceder à finalização da escrita do respetivo relatório. Essa finalização foi realizada sobretudo em horário pós-laboral e nos fins-de-semana. Após cada entrega também teve de preparar a respetiva apresentação perante os elementos do júri e eventual assistência.

Na seguinte figura é apresentado um diagrama de Gantt onde são apresentadas as atividades mencionadas, e tem o intuito de fornecer uma perspectiva temporal sobre o trabalho realizado.

3.1. E-learning

Nesta secção são descritos os dois documentos elaborados pelo estagiário no âmbito do e-learning.

Documento de Visão

O Documento de Visão foi elaborado com o objetivo de procurar saber que alterações, ao nível das funcionalidades, seriam necessárias à aplicação NONIO para que consiga suportar futuramente a leccionação em modo e-learning. Neste documento são indicadas as funcionalidades existentes atualmente no NONIO que serão úteis num contexto de e-learning, funcionalidades desnecessárias, novas funcionalidades a desenvolver, e ainda eventuais funcionalidades futuras para um período posterior ao deste estágio.

As quatro novas funcionalidades que poderão ser desenvolvidas são as seguintes:

- **Diário** – Meio de comunicação privada entre cada aluno e os seus docentes. Permite ao docente obter feedback individual e eventualmente dar conselhos personalizados, e ao mesmo tempo pode ser um exercício de reflexão e aprendizagem para o aluno;
- **Glossário** – Lista de definições de termos pouco conhecidos elaborada pelos alunos, e que permite a colaboração na construção de uma base comum de conhecimento;
- **Testes** – Sequências de perguntas preparadas pelo docente, podendo cada uma ter um tipo de resposta específico (verdadeiro ou falso, escolha múltipla, resposta livre, etc.);
- **Avaliação por colegas** – Semelhante à funcionalidade Submissão de Trabalhos, mas a nota final de cada trabalho de um estudante é composta não só pela avaliação desse trabalho, mas também pela avaliação que o estudante fez de trabalhos de outros estudantes.

Após esta breve análise foi possível constatar que muito trabalho poderia ser feito para preparar o NONIO a fim de suportar futuramente a leccionação em modo-learning. Por isso, de acordo com a prioridade, o Diário foi a funcionalidade escolhida para desenvolvimento pelo estagiário no primeiro semestre, e será explicada mais à frente na secção 3.2.

Reproduzir ficheiros multimédia

Uma das alterações indicadas no Documento de Visão para a funcionalidade Materiais de Apoio era reproduzir conteúdo multimédia diretamente a partir do NONIO. Com esse objetivo em mente o estagiário realizou um estudo com as opções disponíveis para realizar essa ação. Foram analisadas as opções para reproduzir ficheiros de áudio e de vídeo enviados para o NONIO, bem como reproduzir vídeo existente na Internet e pré-visualizar ficheiros de imagem ou documentos de texto. A opção escolhida para reproduzir ficheiros de áudio e de vídeo foi adoptar uma solução híbrida que utiliza um player HTML5 sempre que possível ou em alternativa um player Flash.

Após a realização deste estudo um elemento da equipa de desenvolvimento colocou em prática as sugestões apresentadas para ficheiros de áudio e de vídeo enviados para o NONIO e agora é possível reproduzir conteúdo multimédia diretamente a partir do NONIO. Além disso, esse elemento escreveu ainda um documento a explicar o seu trabalho e pode ser encontrado no anexo 11.2 “NONIO_BackendEncodingStreaming_27-12-2011.pdf”. Embora este documento não tenha sido elaborado pelo estagiário, foi anexado com o objectivo de comprovar o trabalho realizado por esse elemento da equipa. O estagiário, apesar de não ter participado na concretização, ficou satisfeito por ter sido dada atenção ao estudo que realizou e em saber que atualmente já se pode reproduzir conteúdo multimédia diretamente a partir do NONIO.

Apresentação temporal da informação

Atualmente a informação no NONIO pode ser acedida através de categorias no menu lateral. Trata-se de uma forma simples e organizada para um utilizador aceder rapidamente à informação que pretende, incluindo os conteúdos no contexto das disciplinas.

Aquando da criação desta proposta de estágio, uma tarefa importante que estava prevista no âmbito do e-learning era a criação de uma nova forma de apresentação da informação no NONIO. Essa nova forma deveria responder às necessidades do ensino à distância. A forma como a aplicação Modular Object-Oriented Dynamic Learning Environment (MOODLE) [24] organiza a informação numa página dividida por módulos revelou-se um ponto de partida. Basicamente a ideia era ter uma página em cada disciplina onde se poderiam diretamente adicionar e organizar materiais e atividades, numa sequência temporal, procurando também criar um ambiente dinâmico. Essa nova forma de apresentação da informação seria utilizada nos futuros cursos de e-learning através do NONIO, mas também poderia ser utilizada nas disciplinas em modo presencial. Além disso, a atual forma por categorias continuaria a existir, seguindo o mesmo comportamento encontrado no Moodle.

Com isso em mente, foi agendada uma reunião com a empresa Tangível, uma empresa especialista em usabilidade e design de interação, sediada no Instituto Pedro Nunes (IPN), com a qual o NONIO já trabalhou em ocasiões anteriores. Tal reunião foi realizada no dia 11 de Outubro, na qual o estagiário e o gestor de projeto estiveram presentes. O pretendido foi explicado em detalhe à empresa. Devido à importância e complexidade desta tarefa, podemos classificá-la como a principal de todo o estágio no âmbito do e-learning e que maior impacto teria nos utilizadores do NONIO. No entanto, este projeto não chegou a ser iniciado visto que não foram contratados os serviços da empresa.

Reflexão:

Quando o estagiário integrou a equipa não existia uma lista priorizada de funcionalidades para desenvolvimento que poderiam ser especialmente úteis no contexto de e-learning. O facto da elaboração do Documento de Visão ter sido a primeira tarefa delegada ao estagiário permitiu que dedicasse algum tempo ao estudo das questões de e-learning e da sua aplicação ao NONIO. O estagiário vê como o aspeto mais positivo o facto de ter recorrido aos conselhos do Professor António Mendes, que se revelaram bastante úteis para a elaboração do Documento de Visão.

Quanto ao estudo sobre reproduzir ficheiros multimédia, o estagiário vê como o aspeto mais positivo a abrangência do estudo, porque não se limitou às opções mais óbvias e rápidas, mas levou em conta um leque abrangente de opções.

Além disso, o facto de ambos os documentos terem sido elaborados antes de contato com o PDS foi positivo no sentido de que o estagiário esteve concentrado apenas nestas tarefas. Sem dúvida, estes documentos contribuíram para o estagiário obter um conhecimento mais aprofundado sobre os temas do e-learning e da reprodução de conteúdo multimédia.

Sobre a elaboração destes dois documentos o estagiário tem dificuldades em encontrar aspetos que poderiam ter sido melhores, embora possam existir.

3.2. Desenvolvimento da funcionalidade Diário

Especificação

Com base no Documento de Visão foi decidido avançar com o desenvolvimento da funcionalidade Diário, por apresentar uma complexidade inferior às restantes e por aparentar ser mais útil.

O estagiário começou por estudar a funcionalidade Journal do Moodle (que serve de base), com o intuito de procurar replicar o seu funcionamento. No Journal cada aluno comunica de forma privada com o professor, podendo o professor dar algum feedback. É esperado que seja principalmente o aluno a escrever e as intervenções do docente sejam em situações pontuais. As aplicações desta funcionalidade são diversas: obter feedback generalizado (e privado) sobre um tema, fazer time-tracking do esforço de uma disciplina, etc.

Constatou-se que o Journal é uma funcionalidade bastante simples mas que apresenta uma grande limitação: cada aluno tem um espaço único para escrever conteúdo, se quiser adicionar novo conteúdo no futuro tem de editar o já existente, e se quiser adicionar uma referência temporal em cada comentário tem de adicioná-la manualmente. Uma preocupação constante era procurar desenvolver no NONIO uma funcionalidade que fosse igual ou melhor que a original no Moodle, por isso o estagiário viu nesta limitação uma oportunidade para fazer um trabalho melhor. Aprofundou um pouco mais o seu estudo das funcionalidades disponíveis no Moodle e encontrou a Dialogue, que permite estabelecer um diálogo direto entre professor e aluno. Foi decidido que o Diário deveria resultar da junção dessas duas funcionalidades, aproveitando o melhor de cada uma delas.

Portanto, na funcionalidade Diário no NONIO cada aluno pode comunicar de forma privada com o professor (como no Journal), mas pode adicionar conteúdo em espaços individuais, com referências temporais adicionadas de forma automática, apresentados numa sequência temporal (como na Dialogue). Esta adaptação significa que, no limite, o aluno pode ter um diálogo com os docentes, de forma muito semelhante a uma sessão de chat.

Após este período inicial de levantamento de requisitos que envolveu essencialmente o estudo das funcionalidades existentes no Moodle, o estagiário elaborou o protótipo da funcionalidade Diário através da criação dos ecrãs necessários na vista alunos e na vista docente. Tal como em várias outras funcionalidades já existentes no NONIO, também nesta existem semelhanças entre ecrãs com o mesmo nome na vista alunos e na vista docente, tendo normalmente o docente mais permissões. O estagiário preocupou-se em seguir as convenções já existentes em relação à elaboração de ecrãs no protótipo, contribuindo assim para manter a coerência visual a que os utilizadores do NONIO estão habituados e não sintam dificuldades quando utilizarem esta nova funcionalidade.

Foram especificados no total cinco ecrãs: três na InforDocente e dois na InforEstudante. Ambas as vistas partilham dois ecrãs: um deles apresenta a lista de diários associados à edição, e o outro apresenta os detalhes e mensagens de um diário selecionado. O ecrã específico da vista docente é o que permite fazer a gestão de diários (adicionar um novo, ou alterar um já existente). Esses ecrãs foram acrescentados ao módulo Ensino, onde já se encontravam os ecrãs das outras funcionalidades de leccionação: materiais de apoio, fóruns de discussão e submissão de trabalhos. Esses ecrãs podem ser encontrados nos anexos 3.1 a 3.5.

O estagiário procedeu à escrita do DEA, procurou seguir a forma de escrita habitualmente utilizada e fornecer a informação necessária para os ecrãs que especificou. Colocou o conteúdo produzido dentro do DEA do módulo Ensino, que pode ser encontrado no

anexo 4.1 “NONIO_DocumentoEspecificacoesAdicionais_Ensino.pdf”. Agendou também uma segunda reunião com o Professor António Mendes, desta vez com os objetivos de analisar e validar o protótipo. Essa reunião foi realizada no dia 16 de Novembro e no final concluiu-se que o protótipo demonstrava o funcionamento esperado para o Diário, com base nas funcionalidades base do Moodle.

A reunião de análise foi agendada entre o estagiário e outros dois elementos da equipa de desenvolvimento. Foram aplicadas várias alterações no protótipo, principalmente no ecrã de detalhes da vista docente, que envolveram a organização das mensagens e a criação dos campos de pesquisa e ordenação. Embora tenham aumentado o esforço de implementação, essas alterações irão facilitar imenso a utilização desta funcionalidade por parte do docente. As alterações foram aplicadas diretamente no protótipo e no DEA sem necessidade de se elaborar um registo escrito, visto a reunião de análise ser um processo informal. A reunião de análise foi muito importante, não só porque permitiu aumentar a qualidade da especificação em relação à primeira versão elaborada pelo estagiário, mas também para o estagiário obter um conhecimento mais aprofundado sobre essa fase do PDS junto de elementos da equipa mais experientes.

A reunião de inspeção foi agendada e nela participaram o estagiário, um dos elementos da equipa de desenvolvimento que participou na reunião de análise, e outros dois elementos. Foi elaborado o respectivo documento de inspeção, visto a reunião de inspeção ser um processo formal, onde se listaram as alterações a aplicar. Outro elemento da equipa aplicou as alterações no protótipo e no DEA e aprovou a versão da especificação onde consta a funcionalidade Diário. Por isso, além do estagiário, 5 dos 10 elementos da equipa de desenvolvimento estiveram envolvidos de alguma forma na especificação do Diário.

O modelo de dados desta funcionalidade é relativamente simples. Na figura 5 são apresentadas as cinco tabelas que compõem o modelo de dados físico. De seguida é apresentada uma breve descrição para cada tabela:

- **DIARIOS_GRUPOS_FE** – Esta é a tabela principal onde em cada registo são guardados os atributos de um diário adicionado no contexto de uma disciplina;
- **DIARGRUP_ASSOCIADOS_FE** – A tabela **DIARIOS_GRUPOS_FE** tem uma relação N para N com a tabela **FICHAS_EDICAO**, e é esta a tabela que realiza essa ligação. Em cada registo desta tabela são guardadas as chaves primárias de ambas para indicar que um diário está associado a uma ficha de edição;
- **DIARIOS_INDIVIDUAIS** – Dentro de um diário podem existir vários individuais cada um correspondendo a um aluno específico. Em cada registo é guardado o diário e o aluno a que pertence;
- **DIARIOS_MENSAGENS** – Dentro de cada individual podem existir várias mensagens. Em cada registo são guardados os atributos de uma mensagem específica: o diário individual (aluno) a que pertence, a pessoa que escreveu a mensagem (o próprio aluno ou um docente), a data de criação, eventual data de alteração, e o conteúdo da mensagem;
- **DIARIOS_GRUPOS_CONSULTAS** – É uma tabela que relaciona as tabelas **DIARIOS_GRUPOS_FE** e **PESSOAS**, e guarda a data da última consulta de um diário por uma pessoa. É útil para determinar que mensagens devem ser marcadas como novas ou atualizadas, no momento do acesso ao diário por uma pessoa.

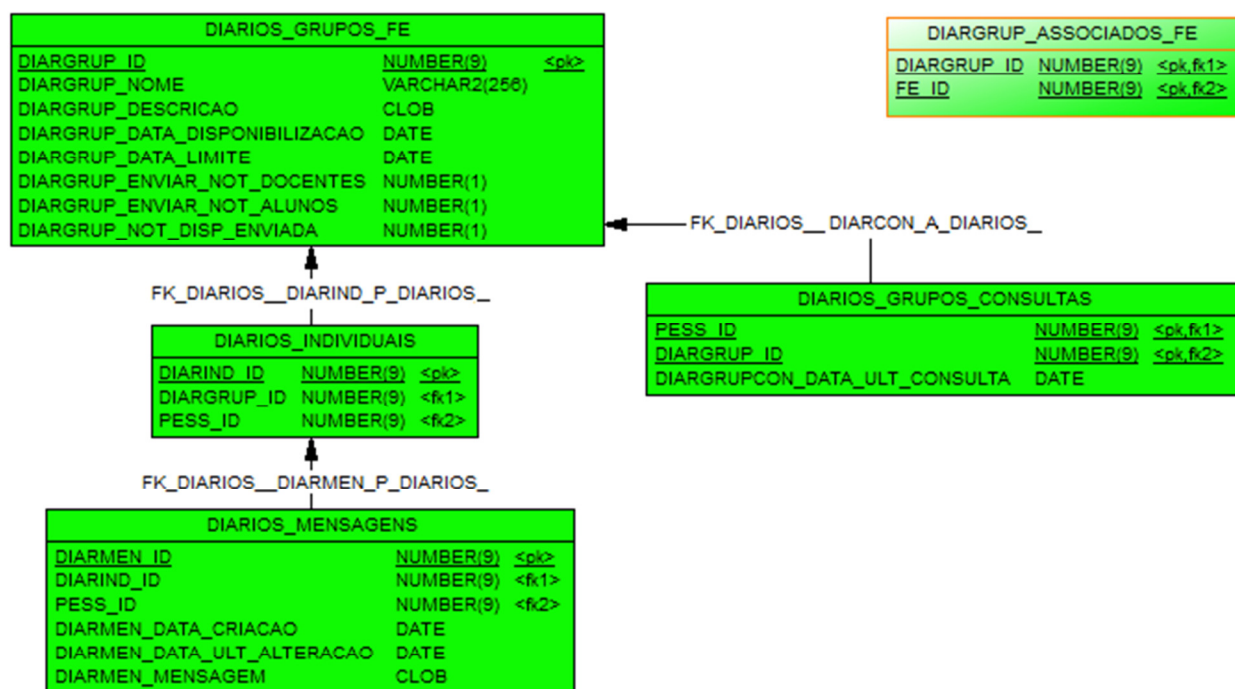


Figura 5 – Modelo de dados da funcionalidade Diário

O anexo 5.1 “Nonio_ModeloDadosConceptual.pdf” contém o modelo de dados conceptual de toda a aplicação NONIO. Devido ao crescimento da aplicação, neste momento já se podem encontrar mais de 600 tabelas.

Design

Esta fase servia para ajudar os elementos mais novos no projeto a perceber o funcionamento das tecnologias. O estagiário obteve esse conhecimento através de sessões de formação e de colocar em prática o que aprendeu. Por isso, não houve necessidade de considerar a fase de design para esta funcionalidade.

Construção

As duas sessões de formação foram conduzidas pelo Eng.º João Rafael. A primeira sessão foi realizada no dia 15 de Novembro e focou as camadas superiores (apresentação e controlo). O estagiário aplicou os conhecimentos adquiridos na funcionalidade que estava a começar a implementar. Após ter realizado uma boa parte da implementação das camadas superiores, solicitou a realização da segunda sessão da formação, que foi realizada no dia 20 de Dezembro e focou as camadas inferiores (serviços e acesso a dados). O estagiário foi implementando partes da funcionalidade com complexidade cada vez maior, à medida que aumentava a sua compreensão das tecnologias. Durante todo este tempo o estagiário foi acompanhado pelo Eng.º João Adriano, para averiguar se nas situações que poderiam ser implementadas de várias formas estava a ser seguida a mais correta.

O estagiário aplicou testes unitários e de integração ao longo da fase de construção. Tal como aconteceu para a fase de design, os testes regressivos também não foram considerados para esta funcionalidade. Foi uma funcionalidade bastante diversificada e permitiu ao estagiário conhecer muitos aspectos da aplicação ao nível da programação. De seguida são listados os

principais para dar uma noção da diversidade e complexidade de aspectos com os quais o estagiário teve contato:

- **Fotografias de utilizadores** – Perceber e utilizar a associação entre cada pessoa e as suas fotos inseridas na aplicação;
- **Auditoria** – Registo de mensagens de log com a descrição das ações realizadas pelos utilizadores na aplicação;
- **Notificações** – Criação de mensagens a serem enviadas por email aos utilizadores do NONIO e também para ficarem disponíveis para consulta na própria aplicação;
- **Procedimentos** – Definição de procedimentos a ser aplicado na base de dados, a fim de tornar a pesquisa mais rápida do que através das consultas anotadas nos POJOS;
- **Job** – Definição de uma tarefa a ser realizada de forma periódica pela aplicação;

Lançamento

A fase de lançamento começou nos primeiros dias de Fevereiro. Nesta fase foram aplicados testes funcionais à funcionalidade, tendo sido elaborado um relatório com os erros encontrados. Esse relatório pode ser encontrado no anexo 8.1 “NONIO_RelatorioTestesFuncionais_Diário_JGomes_09-02-2012.pdf”. Durante a aplicação dos testes, o estagiário esteve ocupado a finalizar a preparação da defesa intermédia. Após a defesa, o estagiário procedeu à correção dos erros encontrados. Embora tenham sido em grande quantidade, o que se justifica pelo desconhecimento, pressão e alguma falta de experiência por parte do estagiário, cerca de metade dos erros eram detalhes de apresentação e tiveram uma correção rápida. Alguns dos erros resultaram na elaboração de um PA, que pode ser encontrado no anexo 6.1 “NONIO_PA_2012-03-06_538.pdf”. Quando todos foram corrigidos, o que aconteceu no dia 1 de Março, a funcionalidade foi disponibilizada aos utilizadores finais através da atualização das aplicações em ambiente de produção. O estagiário elaborou ainda um manual de utilizador para auxiliar os docentes seguindo um template definido pela equipa, e pode ser encontrado no anexo 9.1 “NONIO_ManualUtilizador_Diario.pdf”.

Manutenção

Após a funcionalidade ter sido disponibilizada aos utilizadores finais e até ao momento da entrega deste relatório não houve a necessidade de realizar alterações, possivelmente pela baixa utilização até ao momento. De seguida são apresentadas algumas estatísticas sobre a utilização da funcionalidade Diário pelos utilizadores do NONIO, que foram recolhidas no dia indicado na capa deste relatório.

Estatística	Valor
Total de diários	24
Total de diários individuais	1448
Média de diários individuais por diário	60,3
Total de mensagens	19
Média de mensagens por diário	0,79

Tabela 2 – Estatísticas de utilização da funcionalidade Diário

Como se pode constatar a utilização do Diário foi bastante reduzida no primeiro semestre de utilização. O mesmo também sucedeu após o lançamento de outras funcionalidades no NONIO tais como Fóruns de Discussão e Submissões de Trabalhos. No entanto, tal como aconteceu com essas funcionalidades é esperado que a utilização do Diário também aumente nos próximos semestres e anos lectivos.

Reflexão:

Sobre o desenvolvimento da funcionalidade Diário muitos aspetos correram bem. O estagiário já tinha noções de HTML e JavaScript (que não são ensinados em nenhuma disciplina do curso) que o ajudaram a elaborar os primeiros ecrãs do protótipo, embora tenha tido algumas dificuldades pontuais como é natural. Os sólidos conhecimentos de base de dados também foram de imensa ajuda porque relativamente ao modelo de dados o estagiário não teve dificuldades. Embora o estagiário desconhecesse as frameworks utilizadas no projeto NONIO o facto de serem todas baseadas em Java, e Java ser uma linguagem de programação bem conhecida pelo estagiário, facilitou imenso a compreensão do seu funcionamento e ligação.

Um dos aspetos que poderia ter corrido melhor teve a ver com a estimativa do tempo necessário para a implementação do Diário. Apesar de ser uma funcionalidade simples, foi introduzida alguma complexidade durante a fase de especificação, nomeadamente no ecrã de detalhes. Além disso, foi detetado um número elevado de erros durante a aplicação dos testes funcionais. O estagiário considera que estes foram os dois principais fatores que mais contribuíram para a funcionalidade ser disponibilizada depois do início do segundo semestre.

3.3. Desenvolvimento da funcionalidade Requerimentos

Mudança de âmbito

Conforme referido na introdução deste relatório o âmbito do estágio foi alterado entre o primeiro e segundo semestre. As razões para essa alteração são detalhadas de seguida.

Após o trabalho realizado no primeiro semestre, seria de esperar que o estagiário se envolvesse em desenvolver as restantes funcionalidades que se encontram descritas no Documento de Visão (glossário, testes online, avaliação por colegas). Ou então dedicar-se à nova forma de apresentação da informação. No entanto, como já vem sendo habitual no projeto NONIO e acontece em muitos outros projetos reais, surgem novas prioridades ao longo do tempo. No final do mês de Dezembro foi apresentada ao estagiário uma nova funcionalidade com elevada prioridade para desenvolver: gestão de requerimentos através do NONIO. Foi solicitada ao estagiário a decisão de continuar a trabalhar no tema do e-learning ou dedicar-se a esta nova funcionalidade. Após alguma séria ponderação, o estagiário optou pela segunda opção.

Uma das principais razões que contribuiu para esta escolha foi que não seria possível ter e-learning a funcionar no NONIO no início do próximo ano letivo. No primeiro semestre, o estagiário dedicou-se apenas à vertente de leccionação ou pedagógica do e-learning. Além desta, existe também a vertente académica ou logística, que envolve por exemplo as candidaturas e matrículas dos alunos nos cursos de e-learning. Mesmo que o estagiário conseguisse desenvolver as restantes três funcionalidades descritas no Documento de Visão e ainda a nova forma de apresentação de informação, seria preciso que outro(s) elemento(s) da equipa de desenvolvimento trabalhasse(m) na vertente logística. Acontece que não houve recursos humanos disponíveis para trabalharem também na questão do e-learning, devido as outras prioridades existentes no projeto NONIO. No entanto, o e-learning por certo não será um tema abandonado para sempre, futuros estagiários ou elementos da equipa de desenvolvimento poderão dar continuidade ao trabalho iniciado neste estágio.

O estagiário também estava convicto que esta mudança de âmbito seria benéfica para todos. Para os utilizadores do NONIO, a informatização deste frequente processo permitiria acelerar o tempo de processamento dos requerimentos que podia durar semanas ou até meses, para uma questão de dias. Além disso, devido à grande quantidade de requerimentos em papel entregues na UC (cerca de vinte e cinco mil por ano, de acordo com fontes dos SGA), esta funcionalidade também permitiria melhorar a eficiência dos serviços de atendimento aos alunos, e por conseguinte, da UC.

O estagiário também viu esta funcionalidade com uma mais-valia para si, pois iria permitir ter contacto direto com pessoas da área académica com o potencial de aprender imenso sobre diversos assuntos relacionados com essa área. Dessa forma, poderia participar no desenvolvimento de funcionalidades nas duas principais áreas do projeto NONIO: Diário (área pedagógica) e Requerimentos (área académica). Embora a decisão tenha sido consensual entre os orientadores e o estagiário, convém salientar ainda que esta não foi uma decisão fácil para o estagiário, visto o e-learning ser um tema com o qual se familiarizou e no qual estava bastante empenhado.

Esta mudança de âmbito demonstra bem a natureza dinâmica da UC, o que tem impacto na definição das prioridades do projeto NONIO. Essas prioridades dependem de vários factores que precisam ser levados em conta tais como a importância, o impacto da execução, os recursos disponíveis, etc.

Especificação

O ponto de partida para o desenvolvimento da funcionalidade Requerimentos no NONIO foi o anexo 11.1 “Implementação de Requerimentos no NONIO.pdf”, anexo esse que não foi elaborado pelo estagiário, foi-lhe entregue ainda no primeiro semestre para o ajudar a tomar a decisão de mudança de âmbito do estágio. Na introdução deste relatório é explicado o que é um requerimento, como este processo era feito e como passou a ser, a importância e os benefícios desta funcionalidade.

De acordo com o pretendido, a primeira etapa no desenvolvimento desta funcionalidade envolveu essencialmente a InforGestão e a InforDocente, e concentrou a grande maioria dos requisitos pretendidos. Desta forma, foi possível disponibilizar mais cedo aos utilizadores finais uma parte da funcionalidade, ficando para uma etapa posterior o desenvolvimento dos restantes requisitos.

O funcionamento da funcionalidade é resumidamente descrito de seguida. Através da InforGestão um colaborador dos SGA pode registar requerimentos e realizar várias ações sobre cada um deles, incluindo reencaminhar para outros utilizadores com acesso à InforDocente a fim de registarem parecer ou decisão. Através da InforDocente um utilizador pode aceder aos requerimentos que lhe foram reencaminhados no âmbito de uma responsabilidade e registar o seu parecer ou decisão que lhe foi pedido. Através da InforEstudante o requerente pode consultar os seus requerimentos. Na InforGestão além do separador principal que permite gerir os requerimentos, existe ainda um segundo separador que permite associar pessoas a responsabilidades e para as quais se podem reencaminhar requerimentos, e um terceiro separador que permite gerir ofícios, sendo um ofício um requerimento sem requerente.

Tal como se previa e realmente se veio a verificar após a especificação da funcionalidade estar terminada, o seu funcionamento detalhado veio a ser diferente em vários aspectos em relação ao conteúdo do anexo. Embora a maioria dessas alterações tenha a ver com melhorias ou refinamentos, uma alteração significativa foi a não utilização de estados, que era um conceito central no anexo. Foram antecipadas eventuais dificuldades futuras para gerir esses estados e optou-se por não utilizá-los, embora a sequência lógica de passos na funcionalidade tenha ficado muito semelhante ao que se pretendia inicialmente.

O estagiário começou por elaborar uma primeira versão do protótipo com base no conteúdo do anexo 11.1, tendo sido discutida entre alguns elementos da equipa. Depois foi agendada e realizada uma primeira reunião com Dra. Sílvia Figueiredo (Chefe da Divisão de Graduação e Formação da UC) visto ser a pessoa representativa do cliente nesta funcionalidade. Foi apresentado e discutido o protótipo, foram aplicadas alterações, foi agendada e realizada outra reunião, e assim sucessivamente até se obter uma versão do protótipo suficientemente estável. Ao todo foram realizadas cinco reuniões, sendo que a partir da terceira também esteve presente o Eng.º Jorge China (Chefe do Serviço de Gestão de Sistemas e Infra-estruturas de Informação e Comunicação da UC (GSIIC)), que contribuiu com a sua perspectiva.

Ao todo o estagiário especificou 29 ecrãs, 23 para a InforGestão, 5 para a InforDocente, e 1 para a InforEstudante, que podem ser encontrados do anexo 3.6 ao anexo 3.34. Depois procedeu à escrita do DEA que pode ser encontrado no anexo 4.2 “NONIO_DocumentoEspecificacoesAdicionais_Requerimentos.pdf” e ao contrário da funcionalidade Diário que se encontra dentro do módulo de Ensino, a funcionalidade Requerimentos trata-se de um novo módulo o que implicou a escrita de um novo DEA.

O modelo de dados da funcionalidade Requerimentos aproveitou algumas das tabelas que eram utilizadas para a funcionalidade Pedidos, e manteve-se também a informação já

registada nessas tabelas. Na figura 6 são apresentadas as 18 tabelas que compõem o modelo de dados físico. De seguida é apresentada uma breve descrição para cada uma dessas tabelas:

- **REQUERIMENTOS** – Esta é a tabela principal da funcionalidade, e guarda os detalhes de um requerimento. Relaciona-se com a tabela PESSOAS para se saber quem é o requerente, bem como a pessoa que registou o requerimento. Relaciona-se consigo própria caso seja a base para um requerimento reapreciado. Relaciona-se ainda com várias outras tabelas onde podem ser encontrados vários detalhes de um requerimento;
- **REQ_TIPOS** e **REQ_TIPOS_I18N** – Onde estão registados todos os possíveis tipos para um requerimento;
- **REQ_MODOS_ENTRADA** e **REQ_MODOS_ENTRADA_I18N** – Onde estão registados todos os possíveis modos de entrada para um requerimento;
- **DOC_ASSOCIADOS_REQ** – No caso de o requerimento ter um ficheiro associado, é nesta tabela onde se regista essa associação, bem como a indicação se esse ficheiro deve ser impresso junto com o resumo do requerimento;
- **REQ_PARECERES** – Esta tabela guarda os detalhes de um parecer, bem como os reencaminhamentos de pedido e devolução do parecer.
- **REQPAR_FICHEIROS** – No caso de o parecer ter um ficheiro associado, é nesta tabela onde se regista essa associação;
- **DESPACHOS** – Esta tabela guarda os detalhes de uma decisão, e relaciona-se com a tabela PESSOAS para se saber a pessoa que registou a decisão;
- **DOC_ASSOCIADOS_DESP** – No caso de a decisão ter um ficheiro associado, é nesta tabela onde se regista essa associação;
- **REQ_REENCAMINHAMENTOS** – Esta tabela guarda os detalhes de todos os reencaminhamentos relativos a um requerimento. Relaciona-se com as mesmas tabelas duas vezes para se obter informação sobre o remetente e o destinatário de cada reencaminhamento;
- **REQ_MSG_AUDITORIA** – Esta tabela guarda todas as mensagens de auditoria relativas a um requerimento. Foi criada com o objetivo de tornar mais rápida a obtenção dessas mensagens para um requerimento específico do que seria caso fosse utilizada a tabela onde normalmente são registadas as mensagens de log.
- **REQ_REGISTOS_NOTIFICAÇÃO** – Esta tabela guarda todas as notificações ao requerente relativas a um requerimento;
- **REQ_CONSULTA_PESS** – Esta tabela guarda a data da última consulta e a data da última notificação sobre um requerimento e uma pessoa;
- **REQ_TIPOS_RESPONSABLEIS** e **REQ_TIPOS_RESPONSABLEIS_I18N** – Onde estão registados todos os possíveis tipos de responsabilidades que se podem associar a uma pessoa.
- **REQ_RESPONSABLEIS** – Esta tabela guarda todas as associações entre pessoas e tipos de responsabilidades;
- **REQTIPRESP_DEST_PEDIDOS_PAR** – Esta tabela guarda as associações entre responsabilidades, para se saber na InforDocente que responsabilidades têm permissão para pedir parecer.

Algumas das tabelas têm uma correspondente I18N para permitir a eventual tradução dos conteúdos para outros idiomas, além do português.

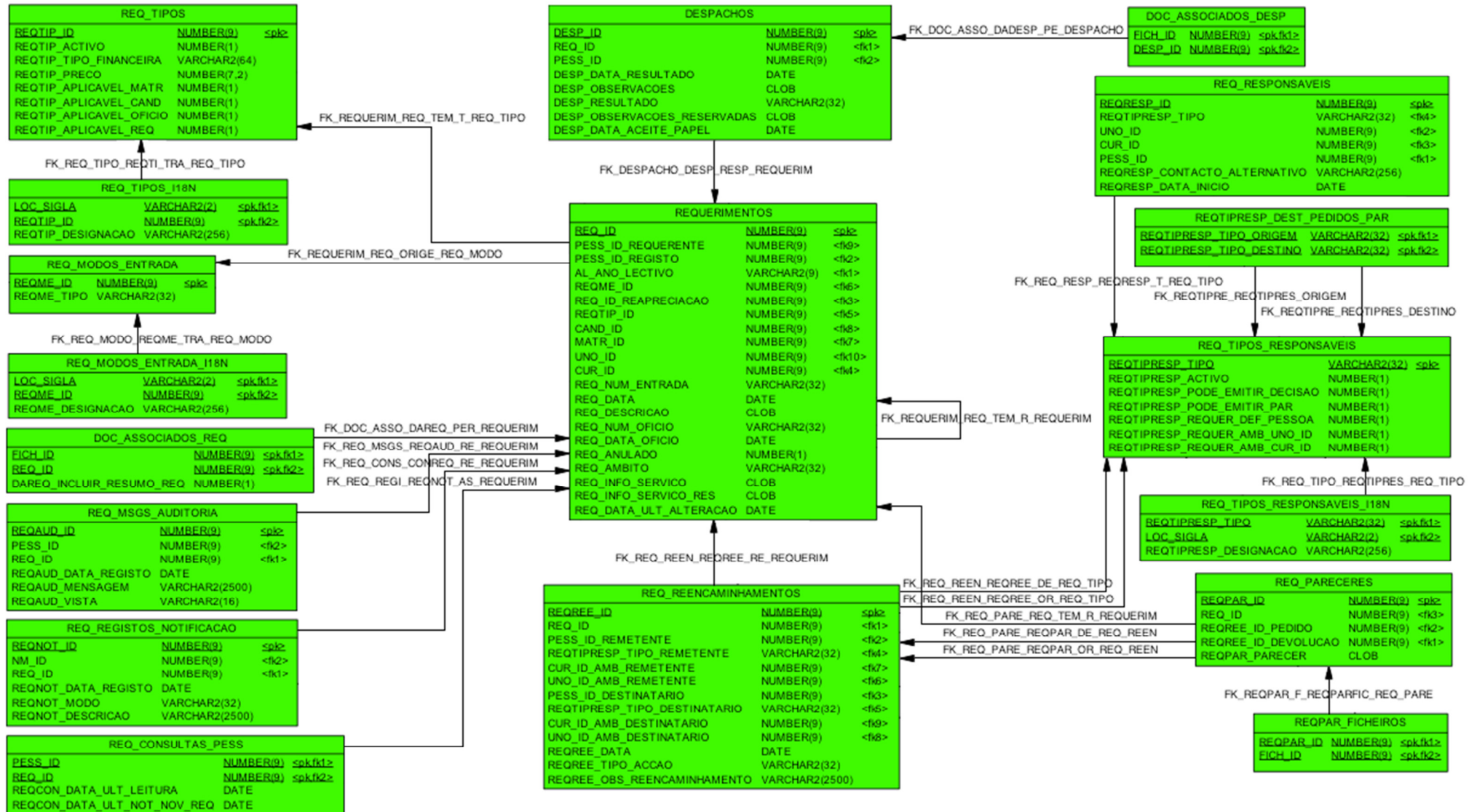


Figura 6 – Modelo de dados da funcionalidade Requerimentos

A reunião de análise foi agendada e realizada, e devido às várias e frequentes revisões pelas quais a especificação passou, esta reunião foi relativamente rápida comparativamente à dimensão e complexidade da funcionalidade.

De salientar que a fase de especificação demorou mais tempo do que o previsto, devido principalmente a dificuldades no levantamento de requisitos. Uma das principais dificuldades foi a questão dos reencaminhamentos. Procurou-se saber exatamente quem poderia reencaminhar para quem, e em que circunstâncias. Após alguma séria discussão foi possível chegar a um pequeno conjunto de regras, que poderia ser facilmente implementado na funcionalidade. Por vezes ao se desenvolver software a fase mais complicada é a de especificação, devido a diversas razões, e foi o caso de Requerimentos, contrariamente ao Diário em que a fase de especificação foi mais fácil para o estagiário do que a fase de construção.

Além disso, também houve a preocupação constante em tornar a funcionalidade o mais intuitiva possível para os seus futuros utilizadores. Essa preocupação pode ser visível na distribuição do conteúdo pelos ecrãs e dentro do mesmo ecrã, no nome e posicionamento dos elementos que constituem cada ecrã. Por vezes é preciso mais trabalho para se chegar a uma melhor solução, e foi o caso desta funcionalidade. Após esse esforço o estagiário acredita que valeu a pena o investimento, e está satisfeito com a especificação da funcionalidade Requerimentos no NONIO.

Devido à sobrecarga de trabalho na equipa nos últimos meses, a especificação da funcionalidade Requerimentos só foi formalmente aprovada durante o mês de Julho. No entanto, o estagiário foi anotando alterações pontuais para serem inseridas na especificação antes da aprovação formal.

Design

Apesar de esta fase do PDS ser opcional o estagiário ainda elaborou em papel um esboço rápido com as ações necessárias e respectiva navegação entre ecrãs, devido à maior dimensão e complexidade desta funcionalidade relativamente ao Diário. No entanto, este esboço não chegou a ser finalizado visto que a partir de determinada altura esta tarefa deixou de trazer benefícios para o estagiário, o que também justifica o carácter opcional desta fase para os elementos mais experientes da equipa de desenvolvimento.

Construção

A primeira tarefa realizada pelo estagiário nesta fase foi o mapeamento das tabelas na base de dados em classes POJO, esta tarefa não chegou a ser realizada pelo estagiário no desenvolvimento do Diário. Das 18 tabelas da funcionalidade, o estagiário mapeou 10. Devido à dimensão e complexidade desta funcionalidade e ao limitado tempo disponível para a sua implementação, foi necessário distribuir o esforço de implementação por outros elementos da equipa de desenvolvimento, além do estagiário. Na seguinte tabela é apresentada a listagem dos ecrãs implementados por cada pessoa:

Sérgio Martins (16 ecrãs)	Eng.º João Rafael (6 ecrãs)	Eng.ª Sandra Alves (8 ecrãs)
REQ 06 ao REQ 12 (7 ecrãs)	REQ 21 ao REQ 23 (3 ecrãs)	REQ 01 ao REQ 05 (5 ecrãs)
REQ 17 ao REQ 20 (4 ecrãs)	W_DC_08	REQ 13
REQ 24	REQ D01 – DC	REQ 15
REQ D03 – DC	REQ D02 – DC	REQ 16
REQ D04 – DC		
REQ D09 – DC		
REQ A01 – AL		

Tabela 3 – Distribuição da implementação dos ecrãs da funcionalidade Requerimentos

Esta divisão do trabalho permitiu que a fase de implementação cumprisse o prazo estipulado: final de Junho. Relativamente à implementação do Diário, a implementação de Requerimentos foi bastante mais fluida e rápida, visto o estagiário ter obtido uma boa preparação no primeiro semestre. Pode-se afirmar que o estagiário foi mais produtivo porque realizou um maior volume de trabalho num menor espaço de tempo. Embora a implementação tenha decorrido ao longo de dois meses, o estagiário implementou a maior parte da funcionalidade no primeiro mês deixando para o segundo algumas partes em que teve mais dificuldade ou envolviam aspetos menos conhecidos pelo estagiário. Tal como no Diário, o estagiário aplicou testes unitários e de integração, mas não aplicou testes regressivos.

Em relação ao primeiro semestre, o estagiário teve contato com vários outros aspectos ao nível da programação, e são aqui listados os principais:

- **Ficheiros** – Perceber e aplicar os mecanismos de submissão de ficheiros no NONIO, visto ser necessário anexar ficheiros a requerimentos, pareceres e decisões;
- **Classes abstractas** – Vários dos ecrãs apresentam semelhanças entre si, por isso foram definidas classes abstractas com código reutilizável. O estagiário teve não só de criar, mas também de saber utilizar classes abstractas criadas por outras pessoas;
- **Sistema de controlo de versões (SVN)** – Visto que a implementação foi distribuída por várias pessoas, o estagiário teve de utilizar numa base diária o sistema de controlo de versões, o que contribuiu para uma maior e melhor compreensão do seu funcionamento;
- **qTip [25]** – Perceber e aplicar a tooltip que também é utilizada na funcionalidade Horário;
- **iReport [26]** – Geração de um ficheiro Portable Document Format (PDF) com o resumo do requerimento a imprimir pelo docente ou pelo colaborador dos SGA;
- **Normas para apresentação dos ecrãs na InforEstudante** – No primeiro semestre foi criado um template com a estrutura para os JSP dos ecrãs da InforEstudante a fim de dar maior coerência visual.
- **Normas para implementação segura** – No primeiro semestre também foram delineadas algumas regras a fim de proteger a aplicação de eventuais ações ilegais por parte dos utilizadores.

Lançamento

A fase de lançamento decorreu desde a última semana de Junho até meados de Julho. O relatório com os erros encontrados pode ser lido no anexo 8.2 “NONIO_RelatorioTestesFuncionais_Requerimentos_SAlves_25-06-2012.pdf”.

Após todos os erros terem sido corrigidos, era ainda necessário agendar e realizar uma formação a fim de preparar os utilizadores para saberem utilizar a funcionalidade. Antes da formação foi realizada uma reunião com o intuito de apresentar a implementação da funcionalidade. Nessa reunião estiveram presentes a Dra. Sílvia Figueiredo, que esteve em todas as reuniões de levantamento de requisitos, a Dra. Susete Araújo (Diretora dos Serviços de Gestão Académica na UC), e ainda a Sra. Mónica Jesus, colaboradora dos SGA. Todos os intervenientes mostraram-se bastante satisfeitos com o trabalho realizado.

Quanto à formação foi necessário dividi-la em dois dias devido ao limitado número de computadores disponíveis na sala e também para permitir um melhor acompanhamento. No total estiveram presentes 27 colaboradores dos SGA, 13 no primeiro dia e 14 no segundo dia. O estagiário elaborou uma pequena apresentação que pode ser lida no anexo 10.1 “NONIO_Formacao_Requerimentos.pdf”. A formação foi essencialmente prática tendo o estagiário demonstrado o funcionamento da funcionalidade através da utilização dos ecrãs implementados, seguido da realização de exercícios pelos utilizadores. O estagiário esteve sempre disponível para o esclarecimento de dúvidas, tendo conseguido dar respostas satisfatórias.

Manutenção

Devido ao facto de a funcionalidade ainda não ter sido disponibilizada no dia da entrega deste relatório não foi possível apresentar estatísticas de utilização. No entanto, é esperado que a utilização venha a ser superior à utilização antes da disponibilização. É esperado que a utilização aumente ainda mais quando for possível aos requerentes registarem os seus próprios requerimentos no NONIO. Poderá até a vir a ser uma das funcionalidades mais utilizadas no NONIO.

Quanto a futuras alterações à funcionalidade, nas sessões de formação surgiram algumas como seria de esperar, mas de salientar que são alterações de baixa complexidade que não afectam a base da funcionalidade o que denota a solidez da especificação.

Desenvolvimentos futuros e considerações

Para a segunda etapa do desenvolvimento desta funcionalidade pretende-se permitir que os utilizadores com acesso à InforEstudante, nomeadamente estudantes, ex-estudantes e candidatos, possam eles próprios registar os seus requerimentos no NONIO. Além disso, também deve-se incluir informação financeira relacionada com cada requerimento, na secção já reservada para esse efeito no ecrã de detalhes. Também na segunda etapa ou eventualmente numa terceira, poderá ser recolhida e apresentada alguma informação estatística, como por exemplo: total de requerimentos processados, média do tempo de processamento de um requerimento, etc. Essa informação estatística será útil para os SGA a fim de poderem ter uma visão generalizada do comportamento, e eventualmente saberem onde é preciso atuarem para melhorar este processo.

Embora a parte relativa aos requerentes poderem registar os seus requerimentos através da InforEstudante tenha sido considerada no planeamento do trabalho para o segundo semestre, não foi desenvolvida pela razão de falta de tempo. Conforme já referido, a fase de especificação foi mais longa do que o esperado o que encurtou o tempo disponível no segundo semestre. No entanto, neste momento a grande maioria da funcionalidade foi desenvolvida e é esperado que essa parte seja desenvolvida em relativamente pouco tempo. A parte estatística, apesar de estar mencionada no anexo 11.1, não se encontrava no planeamento do trabalho para o segundo semestre, seria desenvolvida caso sobrasse tempo após as duas partes anteriores terem sido desenvolvidas.

Além disso, relativamente ao estudo de estado da arte o tempo também foi muito limitado e não chegou a ser realizado. O estagiário reconhece que tal tipo de estudo é importante e tem o benefício de se descobrirem aspetos existentes em outras aplicações ou sistemas e que podem ser aplicados na solução que se está a desenvolver. Um estudo de estado da arte abrangente e completo não foi possível devido ao tempo limitado. No entanto, para um aspeto específico da funcionalidade (por exemplo reencaminhamentos), o estagiário reconhece que apesar do tempo limitado poderia ter sido feito. Seja como for, o tempo que o estagiário tinha disponível utilizou-o para entregar a tempo uma solução totalmente desenvolvida e pronta a ser utilizada. Além disso, tem a vantagem de ter ocupado os seus dois primeiros meses de estágio com a elaboração de dois documentos que se revelaram importantes para o estágio no âmbito do e-learning.

Se o âmbito deste estágio tivesse sido desde o início o desenvolvimento da funcionalidade Requerimentos, por certo teria havido tempo suficiente para a realização de um estudo de estado da arte, bem como ainda o desenvolvimento dos restantes requisitos. Mas tal não foi possível porque só a meio do estágio é que o âmbito foi alterado devido ao surgimento da prioridade para o desenvolvimento de Requerimentos.

Reflexão:

No desenvolvimento da funcionalidade Requerimentos também há muitos aspectos positivos a destacar. Na fase de especificação o estagiário aprofundou os conhecimentos de HTML e JavaScript, não tendo grandes dificuldades para elaborar o protótipo. Também na escrita do DEA verificou-se uma melhoria do estagiário em saber que aspetos deveria especificar, o que resultou em menores correções na reunião de análise. A fase de construção foi bastante mais fácil e fluida devido sobretudo ao conhecimento das tecnologias, o que também se refletiu na fase de lançamento pelo menor número de erros encontrados e facilidade de correção. A colaboração com os outros elementos da equipa na implementação da funcionalidade também decorreu de forma suave, devido principalmente à adequada distribuição de tarefas e à boa comunicação.

Quanto à formação o estagiário considera que correu bastante bem devido a diversos factores. As pessoas que estiveram presentes são utilizadores regulares do NONIO e a funcionalidade Requerimentos segue inúmeras convenções existentes em outras funcionalidades do NONIO, o que tornou a compreensão do funcionamento desta funcionalidade mais fácil e rápida. Além disso, o estagiário teve o cuidado de dividir a formação em tópicos com dimensão aceitável e de impor um ritmo moderado para que todos pudessem acompanhar facilmente. Para o estagiário a formação foi mais uma oportunidade para ganhar competências em falar perante um público, nas quais já teve imensas dificuldades no passado.

O que poderia ter corrido melhor nesta funcionalidade foi a estimativa do tempo para a fase de especificação, conforme já mencionado neste relatório. Além disso, o facto de as formações terem sido realizadas na mesma semana que a entrega deste relatório dificultou ambas as tarefas, mas o estagiário procurou e acredita que conseguiu conciliar bem a realização simultânea destas tarefas.

Capítulo 4

Outras Atividades

Neste capítulo destacam-se tarefas não menos importantes realizadas de forma paralela com o desenvolvimento das funcionalidades. Apesar de também serem referidas ao longo do capítulo anterior, são apresentadas aqui agrupadas por tipo.

4.1. Reuniões

O estagiário esteve envolvido em diversas reuniões ao longo do estágio. Na seguinte tabela são mencionadas as principais:

Dia (Duração)	Objetivo	Participantes (além do estagiário)
9 de Setembro (1h15min)	Perceber o background do estágio no âmbito do e-learning e decidir a próxima tarefa: Documento de Visão	Eng.º Pedro Pinto
19 de Setembro (1h10min)	Saber que alterações poderiam ser aplicadas e novas funcionalidades poderiam ser desenvolvidas no NONIO para suportar e-learning	Prof. António Mendes
29 de Setembro (30min)	Analisar o Documento de Visão e decidir a próxima tarefa: reproduzir ficheiros multimédia	Eng.º Pedro Pinto
11 de Outubro (1h)	Explicar o pretendido para a nova forma de apresentação da informação no NONIO	Eng.º Pedro Pinto, 2 elementos da Tangível
15 de Novembro (1h20min)	Primeira formação aos estagiários: camadas de apresentação e controlo	Eng.º João Rafael, Estagiário João Amado
17 de Novembro (45min)	Apresentação da primeira versão do protótipo da funcionalidade Diário	Prof. António Mendes
22,23,25,28,29 de Novembro (35h)	Reunião de análise da funcionalidade Diário	Eng.º Pedro Vieira, Eng.º João Adriano
12,13 de Dezembro (13h20min)	Reunião de inspeção da funcionalidade Diário	Eng.º Pedro Vieira, Eng.º Cristóvão Pires, Eng.º João Rafael
20 de Dezembro (50min)	Segunda formação aos estagiários: camadas de serviços e acesso a dados	Eng.º João Rafael, Estagiário João Amado
28 de Dezembro (45min)	Ponto de situação e apresentação da funcionalidade de gestão de requerimentos	Prof. Marco Vieira, Eng.º Pedro Pinto
16 de Janeiro (15min)	Planeamento do trabalho a realizar no segundo semestre	Eng.º Pedro Pinto
16 de Março (1h)	Apresentação e discussão da primeira versão do protótipo da funcionalidade Requerimentos	Eng.º Pedro Pinto, Dra. Sílvia Figueiredo
26 de Março (2h)	Apresentação e discussão da segunda versão do protótipo da funcionalidade Requerimentos	Eng.º Pedro Pinto, Eng.º Ricardo Bica, Dra. Sílvia Figueiredo

30 de Março (1h30min)	Apresentação e discussão da terceira versão do protótipo da funcionalidade Requerimentos	Eng.º Pedro Pinto, Eng.º Ricardo Bica, Dra. Sílvia Figueiredo, Eng.º Jorge China
10 de Abril (2h15min)	Apresentação e discussão da quarta versão do protótipo da funcionalidade Requerimentos	Eng.º Pedro Pinto, Eng.º Ricardo Bica, Dra. Sílvia Figueiredo, Eng.º Jorge China
15 de Abril (2h)	Apresentação e discussão da quinta versão do protótipo da funcionalidade Requerimentos	Eng.º Pedro Pinto, Eng.º Ricardo Bica, Dra. Sílvia Figueiredo, Eng.º Jorge China
23,24,26,27,30 de Abril (38h25min)	Reunião de análise da funcionalidade Requerimentos	Eng.º Ricardo Bica, Eng ^a Sandra Alves
6 de Julho (2h35min)	Apresentação da implementação da funcionalidade Requerimentos	Eng ^a Sandra Alves, Dra. Sílvia Figueiredo, Dra. Susete Araújo, Sra. Mónica Jesus
10 de Julho (2h10min)	Primeira formação da funcionalidade Requerimentos	Eng ^a Sandra Alves, Dra. Sílvia Figueiredo, 13 colaboradores SGA
11 de Julho (1h55min)	Segunda formação da funcionalidade Requerimentos	Eng ^a Sandra Alves, Dra. Sílvia Figueiredo, 14 colaboradores SGA

Tabela 4 – Principais reuniões realizadas no âmbito do estágio

Como se pode constatar facilmente pela tabela este estágio foi rico em reuniões, e contribuíram para preparar melhor o estagiário para o seu futuro profissional, designadamente nas competências de comunicação e relação interpessoal.

Além das reuniões mencionadas na tabela, são também realizadas reuniões semanais da equipa habitualmente cada segunda-feira de tarde. Em todo o período de estágio o estagiário não esteve presente em apenas duas dessas reuniões.

4.2. Formações

Das reuniões mencionadas na tabela, um tipo específico são as formações. As duas primeiras foram dirigidas aos estagiários e decorreram no início da fase de construção do Diário. Foram úteis para conhecer o funcionamento e a ligação entre as várias camadas da arquitetura lógica da aplicação.

As outras duas formações foram organizadas e conduzidas pelo estagiário, decorreram na fase final do estágio, e tiveram como objetivo preparar os utilizadores para saberem como utilizar a funcionalidade Requerimentos. O estagiário considera estas duas formações como dos melhores momentos do estágio visto que teve a oportunidade de apresentar o seu trabalho diretamente aos futuros utilizadores e pelo facto de terem corrido bastante bem.

4.3. Contato com o cliente

Outro tipo de reuniões foram as que envolveram contato com o cliente. Este aspeto foi diferente se compararmos os semestres. No primeiro semestre, aquando do desenvolvimento da funcionalidade Diário, foi reduzido porque se limitou a duas breves reuniões com o Professor. António José Mendes. No segundo semestre, aquando do desenvolvimento da funcionalidade Requerimentos, foi bastante maior porque envolveu cinco reuniões com várias pessoas.

4.4. Testes funcionais

As duas funcionalidades desenvolvidas pelo estagiário foram alvo de aplicação de testes funcionais de acordo com o estipulado no PDS. Os relatórios com os erros detetados nesses testes podem ser encontrados nos anexos 8.1 e 8.2. Salienta-se a gradual melhoria de qualidade na fase de implementação pelo estagiário, que é visível pelo conteúdo destes relatórios.

O estagiário não chegou a aplicar testes funcionais em funcionalidades desenvolvidas por outros elementos da equipa de desenvolvimento. No entanto, obteve o conhecimento necessário para saber como elaborar um relatório de testes funcionais, bem como que aspetos devem ser testados e em que condições.

4.5. Pedidos de alteração

Esta tarefa não chegou a ser realizada pelo estagiário no primeiro semestre, mas no segundo elaborou dois e podem ser encontrados nos anexos 6.1 e 6.2. Apesar de não terem sido muitos, foram os suficientes para o estagiário saber como elaborar devidamente um pedido de alteração.

4.6. Manutenção

O estagiário não chegou a realizar quaisquer atividades de manutenção das funcionalidades por ele desenvolvidas. No caso do Diário a razão mais provável é a baixa utilização da funcionalidade, no caso de Requerimentos tem a ver com o fato de ter sido disponibilizada muito recentemente. De qualquer modo, o estagiário esforçou-se em desenvolver as suas funcionalidades de modo a reduzir ao mínimo a possibilidade de futuras intervenções ao código por si produzido.

Capítulo 5

Conclusões e Futuro

Neste capítulo são apresentadas algumas conclusões resultantes do trabalho realizado pelo estagiário durante o seu inteiro período de estágio.

Os objetivos esperados para o estagiário no primeiro semestre foram atingidos. O estagiário integrou-se muito bem na equipa de desenvolvimento desde o primeiro dia e desenvolveu uma funcionalidade completa. O Diário revelou-se uma funcionalidade adequada para se familiarizar com o PDS, no geral teve uma complexidade não muito elevada, mas superior ao habitual em estágios de anos anteriores para o primeiro semestre. De salientar o ecrã de detalhes na InforDocente que é dos mais complexos existentes na aplicação. Além disso, o primeiro semestre envolveu também algum trabalho diferente do habitual em estágios anteriores, através da elaboração do Documento de Visão e do estudo sobre reproduzir ficheiros multimédia. Esses documentos enriquecem e acrescentam valor ao trabalho realizado.

Os objetivos esperados para o estagiário no segundo semestre também foram atingidos. O estagiário aprofundou os conhecimentos que havia adquirido no primeiro semestre, conseguiu produzir trabalho com um maior volume e complexidade, e dessa forma conseguiu ganhar mais competências na área de desenvolvimento de software. Requerimentos foi uma funcionalidade com dimensão e complexidade bem superiores ao Diário e o sucesso no seu desenvolvimento deveu-se sobretudo à boa preparação que recebeu no primeiro semestre. Embora o estagiário apenas tenha desenvolvido uma funcionalidade no segundo semestre, normalmente os estagiários desenvolvem várias mas de menor dimensão, este facto não foi de alguma forma limitativo, pelo contrário foi vantajoso a diversos níveis.

A nível pessoal foram vários os benefícios resultantes de estagiar no projeto NONIO. O estagiário trabalhou com inúmeras tecnologias e ferramentas que ou desconhecia ou conhecia muito pouco, por isso este estágio resultou numa grande aprendizagem a nível técnico. Realizou um leque diversificado de tarefas, representativo do trabalho realizado diariamente no NONIO, que também é representativo do trabalho realizado numa profissão na área de engenharia de software, o que sem dúvida o preparou para o seu futuro profissional. O estagiário também adquiriu ao longo do tempo cada vez maior autonomia, e antes de solicitar apoio sempre procurou primeiro encontrar soluções autonomamente para as dificuldades com que se deparava. Ao mesmo tempo procurou demonstrar sentido de responsabilidade por realizar o seu trabalho de forma séria e com qualidade, tendo consciência da importância do seu trabalho na reputação de qualidade que a equipa se tem esforçado em alcançar e manter.

Em relação à equipa e ao ambiente de trabalho no qual esteve inserido, o estagiário só tem pontos positivos a destacar. Sempre que solicitou apoio, situação cada vez menos frequente, a equipa demonstrou disponibilidade e prontidão no esclarecimento de dúvidas, e conseguiu dar respostas satisfatórias. Devido à experiência que o estagiário adquiriu na elaboração de trabalhos de grupo para disciplinas do seu curso, tendo por várias vezes a responsabilidade de gerir, juntamente com o facto de praticamente toda a equipa ter obtido formação académica no DEI facilitou bastante a comunicação e o trabalho em equipa. Sobre o ambiente de trabalho, apesar da grande responsabilidade que a equipa carrega, foi encontrado um equilíbrio neste aspecto o que permite usufruir de um ambiente de trabalho nem pesado nem descontraído demais, o que tem impacto positivo na produtividade. Além disso, o estagiário sempre teve liberdade para se dedicar ao seu estágio, não tendo sido sobrecarregado com tarefas pouco ou nada relacionadas com o seu papel na equipa.

Quanto a decisões foram diversas as ocorridas neste estágio, nas quais o estagiário esteve envolvido. Apesar de estarem mencionadas ao longo do relatório são aqui listadas de forma resumida:

- **HTML5** – Resultou do estudo sobre reproduzir ficheiros multimédia, onde foi decidido utilizar um player HTML5 sempre que possível ou em alternativa utilizar um player Flash;
- **Diário = Journal + Dialogue** – Após a descoberta de uma limitação na funcionalidade Journal do Moodle, foi decidido que o Diário deveria resultar da junção das funcionalidades Journal e Dialogue do Moodle, aproveitando o melhor de cada uma;
- **Requerimentos** – Resultou da solicitação feita ao estagiário no final do mês de Dezembro, onde decidiu deixar de trabalhar no tema do e-learning e dedicar-se a esta nova funcionalidade.

Embora o estagiário tenha adquirido nos últimos anos uma sólida formação em Engenharia Informática, mesmo assim era esperado encontrarem-se alguns desafios e dificuldades ao se integrar um novo projeto. De seguida são listados os principais e como foram superados:

- **Documento de Visão** – A elaboração deste documento tratou-se de um desafio porque o estagiário não é um especialista na área de e-learning, e não conhecia o NONIO a fundo especialmente a InforDocente. Este desafio foi superado com a ajuda dos conselhos do Professor António Mendes nas duas reuniões que o estagiário realizou e de um estudo das funcionalidades existentes na InforDocente;
- **Implementação do Diário** – O estagiário teve dificuldades quando começou a implementar a sua primeira funcionalidade no NONIO, devido sobretudo ao desconhecimento das tecnologias. Apesar de o formador ter feito o seu melhor, esta dificuldade foi superada pelo estagiário através de muito esforço e perseverança ao colocar em prática a formação recebida;
- **Especificação de Requerimentos** – Ao contrário do Diário em que a fase de especificação foi relativamente mais fácil, especificar Requerimentos foi mais complicado devido a dificuldades encontradas nas reuniões com o cliente. Esta dificuldade foi superada através de melhor comunicação, mais direcionada para o levantamento de requisitos, tendo em conta que o cliente pode ter e geralmente tem uma formação académica numa área diferente da Informática.

Do tempo que o estagiário esteve envolvido no projeto NONIO e do trabalho que realizou, foi capaz de aprender várias lições. A principal lição foi saber o que é trabalhar num projeto real já com alguns anos de existência, com uma dimensão e complexidade apreciáveis, e com um total de utilizadores na ordem das dezenas de milhar. Estas características, diferentes de qualquer trabalho académico realizado pelo estagiário, exigiram dele um cuidado redobrado porque o trabalho que realizou no NONIO tem impacto numa grande quantidade de utilizadores. O estagiário acredita que esteve à altura destas exigências, tendo procurado demonstrar isso diariamente ao longo do estágio.

Quanto ao futuro profissional do estagiário ainda não se encontra definido. Na eventualidade de continuar na equipa de desenvolvimento do NONIO estará apto para continuar a realizar o seu trabalho sem grandes dificuldades devido à boa preparação que teve durante o período de estágio. Ou caso inicie a sua atividade profissional em outro lugar, a maior dificuldade por certo será a adaptação a esse novo local de trabalho, não será o desenvolvimento de software, nem o trabalho em equipa, nem qualquer outro aspeto relativo à sua formação académica. Seja qual for o futuro, o estagiário está bastante satisfeito por ter escolhido o NONIO como local de estágio para concluir o seu percurso académico na UC.

Por tudo isto, o estagiário faz um balanço bastante positivo do trabalho que realizou, está convencido de que cumpriu com o que se pretendia dele enquanto estagiário, e está motivado para iniciar a sua atividade profissional como Engenheiro Informático, após um breve período de merecido descanso.

Anexos

Aqui são listados e enumerados todos os anexos a este relatório:

1. Documentos de processo

- 1.1. NONIO_ProcessoDesenvolvimento_2009-06-11_v1.2r.pdf
- 1.2. NONIO_PlanoControloAlteracoes_2009-06-11_v1.2r.pdf
- 1.3. NONIO_PlanoGarantiaQualidade_2009-01-10_v1.5r.pdf
- 1.4. NONIO_ArquitecturaDesignGenerico_2009-01-24_v1.4r.pdf

2. Estudos realizados no âmbito do e-learning (da autoria do estagiário)

- 2.1. e-learning@NONIO_DocumentoVisao_2011-09-20.pdf
- 2.2. e-learning@NONIO_ReproduzirFicheirosMultimedia_2011-10-10.pdf

3. Protótipos (da autoria do estagiário)

- 3.1. docentes_diarioLista.html
- 3.2. docentes_diarioAdicionarEditar.html
- 3.3. docentes_diarioDetalhes.html
- 3.4. alunos_diarioLista.html
- 3.5. alunos_diarioDetalhes.html
- 3.6. academicos_reqResponsaveisGerir.html
- 3.7. academicos_reqResponsavelSeleccionar.html
- 3.8. academicos_reqResponsavelAdicionar.html
- 3.9. academicos_reqResponsavelAlterarEmail.html
- 3.10. academicos_reqResponsavelRemover.html
- 3.11. academicos_reqIndex.html
- 3.12. academicos_reqAdicionarSeleccionarRequerente.html
- 3.13. academicos_reqAdicionarAlterarInfo.html
- 3.14. academicos_reqDetalhes.html
- 3.15. academicos_reqHistoricoReencaminhamentos.html
- 3.16. academicos_reqAuditoria.html
- 3.17. academicos_reqAdicionarAlterarInfoComplementar.html
- 3.18. academicos_reqParecerPedir.html
- 3.19. academicos_reqParecerPedirConfirmar.html
- 3.20. academicos_reqParecerPedirAnular.html
- 3.21. academicos_reqDecisaoRegistarAlterar.html
- 3.22. academicos_reqDecisaoRegistarAlterarConfirmar.html
- 3.23. academicos_reqDecisaoAceitarPapel.html
- 3.24. academicos_reqDecisaoNotificarRequerente.html
- 3.25. academicos_reqResumo.pdf
- 3.26. academicos_reqOficios.html
- 3.27. academicos_reqOficiosAdicionarAlterar.html
- 3.28. academicos_reqOficiosDetalhes.html
- 3.29. docentes_reqLista.html
- 3.30. docentes_reqHistoricoRequerimentos.html
- 3.31. docentes_reqDetalhes.html
- 3.32. docentes_reqParecerRegistarAlterar.html
- 3.33. docentes_reqDevolver.html
- 3.34. alunos_reqLista.html

- 4. Documento de Especificações Adicionais (com participação do estagiário)**
 - 4.1. NONIO_DocumentoEspecificacoesAdicionais_Ensino.pdf
 - 4.2. NONIO_DocumentoEspecificacoesAdicionais_Requerimentos.pdf
- 5. Modelo de dados**
 - 5.1. NONIO_ModeloDadosConceptual.pdf
- 6. Pedidos de alteração (da autoria do estagiário)**
 - 6.1. NONIO_PA_2012-03-06_538.pdf
 - 6.2. NONIO_PA_2012-04-03_549.pdf
- 7. Scripts para aplicação na base de dados (da autoria do estagiário)**
 - 7.1. NONIO_PatchBD_2012-03-01_SDMart_ProcPesquisaAlunosDiario.sql
 - 7.2. NONIO_PatchBD_2012-03-05_SDMart_PermissõesNonioWS.sql
 - 7.3. NONIO_PatchBD_2012-04-12_SDMart_RequerimentosTarefas.sql
 - 7.4. NONIO_PatchBD_2012-05-07_SDMart_Requerimentos_NONIOACADV4.sql
 - 7.5. NONIO_PatchBD_2012-05-24_SDMart_ProcPesqReqAcademicos.sql
 - 7.6. NONIO_PatchBD_2012-05-25_SDMart_ProcPesqRequerente.sql
 - 7.7. NONIO_PatchBD_2012-05-29_SDMart_LogsRequerimentos.sql
 - 7.8. NONIO_PatchBD_2012-06-04_SDMart_ProcPesqReqAlunos.sql
 - 7.9. NONIO_PatchBD_2012-06-05_SDMart_PermissoesReqConsultaPess.sql
 - 7.10. NONIO_PatchBD_2012-06-12_SDMart_DestinatariosPedirParecerInfordocente.sql
 - 7.11. NONIO_PatchBD_2012-06-27_SDMart_ModosEntradaI18N.sql
 - 7.12. NONIO_PatchBD_2012-06-29_SDMart_AlteracaoTabelaRequerimentos.sql
 - 7.13. NONIO_PatchBD_2012-07-02_SDMart_RequerimentosPreparacaoUtilizacao.sql
- 8. Relatórios de testes funcionais**
 - 8.1. NONIO_RelatorioTestesFuncionais_Diário_JGomes_09-02-2012.pdf
 - 8.2. NONIO_RelatorioTestesFuncionais_Requerimentos_SAlves_25-06-2012.pdf
- 9. Manuais de utilizador (da autoria do estagiário)**
 - 9.1. NONIO_ManualUtilizador_Diario.pdf
- 10. Formações (da autoria do estagiário)**
 - 10.1. NONIO_Formacao_Requerimentos.pdf
- 11. Outros anexos**
 - 11.1. Implementação de Requerimentos no NONIO.pdf
 - 11.2. NONIO_BackendEncodingStreaming_27-12-2011.pdf

Referências e Bibliografia

Aqui são listadas todas as referências e bibliografia utilizadas para a elaboração deste relatório. Todas as páginas Web mencionadas foram acedidas e estavam disponíveis no dia indicado na capa deste relatório.

- [1] **UC** – <http://www.uc.pt/>
- [2] **FCTUC** – <http://www.uc.pt/fctuc>
- [3] **DEI** – <http://www.uc.pt/fctuc/dei/>
- [4] **WoC** – <http://www.uc.pt/woc>
- [5] **Java** – <http://www.oracle.com/technetwork/java/>
- [6] **Apache Struts** – Chuck Cavaness, Programming Jakarta Struts, O'Reilly, 2004
- [7] **XML** – <http://www.w3.org/XML/>
- [8] **Spring Framework** – Rob Harrop, Jan. Machacek, Pro Spring, Apress, 2005
- [9] **Hibernate** – <http://www.hibernate.org/>
- [10] **JSP** – <http://www.oracle.com/technetwork/java/javase/jsp/>
- [11] **JSTL** – <http://www.oracle.com/technetwork/java/jstl-137486.html>
- [12] **JavaScript** – David Flanagan, JavaScript: The Definitive Guide, O'Reilly, 2006
- [13] **jQuery** – <http://jquery.com/>
- [14] **CSS** – <http://www.w3.org/Style/CSS/>
- [15] **POJO** – http://en.wikipedia.org/wiki/Plain_Old_Java_Object
- [16] **Adobe Dreamweaver** – <http://www.adobe.com/products/dreamweaver.html>
- [17] **Microsoft Visio** – <http://office.microsoft.com/en-us/visio>
- [18] **Power Designer** – <http://www.sybase.com/products/modelingdevelopment/powerdesigner>
- [19] **Eclipse** – <http://www.eclipse.org/>
- [20] **Apache Tomcat** – <http://tomcat.apache.org/>
- [21] **Oracle** – <http://www.oracle.com/us/products/database/index.htm>
- [22] **SQL Developer** – <http://www.oracle.com/technetwork/developer-tools/sql-developer/>

- [23] **CVS** – <http://savannah.nongnu.org/projects/cvs>
- [24] **MOODLE** - <http://moodle.org/>
- [25] **qTip** – <http://craigsworks.com/projects/qtip/>
- [26] **iReport** – <http://jasperforge.org/projects/ireport>