

Mestrado em Engenharia Informática

Estágio

Relatório final

Preview

Jorge Miguel Batista Figueira

figueira@student.dei.uc.pt

Orientadores:

Prof. Dr. Maria José Marcelino

Eng. José Campos, Tangível

Data: 12 de Julho de 2012



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

Tabela de Revisões

Versão	Autor	Descrição da versão	Aprovadores	Data
0.1	Jorge Figueira	Criação do documento e sua estrutura		10-09-2011
0.2	Jorge Figueira	Inserção do estado da arte		20-10-2011
0.3	Jorge Figueira	Criação do capítulo 3		18-12-2011
1.0	Jorge Figueira	Reestruturação do documento		8-01-2012
1.1	Jorge Figueira	Correção de algumas gralhas		20-01-2012
1.2	Jorge Figueira	Reestruturação de alguns capítulos e complementação de alguns parágrafos.		6-07-2012

Resumo

Uma das metodologias mais usadas na área da usabilidade são os *expert reviews* (também conhecidos por *usability reviews* ou *usability evaluations*), onde especialistas de usabilidade avaliam interfaces com base no seu *expertise*. Os problemas encontrados são tipicamente apelidados de *findings* e são normalmente compostos por uma descrição do problema e recomendações para o corrigir ou mitigar.

A falta de ferramentas Web que sirvam de apoio a este tipo específico de projetos, motivou o aparecimento deste projeto, cujo principal objetivo é desenvolver uma aplicação que permita aos especialistas de usabilidade anotar e detalhar *findings* encontrados durante *expert reviews*, torná-los disponíveis ao cliente e refiná-los com base no seu feedback. Pretende-se também que através desta aplicação se estabeleça uma ponte de comunicação entre profissionais de usabilidade e os seus clientes.

Palavras-Chave

“Preview” “usabilidade” “findings” “expert reviews” “feedback”

Agradecimentos

Este projeto, que culminou na criação deste documento final, não seria possível sem a colaboração de algumas pessoas que ajudaram e motivaram no desenvolvimento deste estágio. Em primeiro lugar gostaria de agradecer à equipa da Tangível pelo seu apoio em todas as fases do projeto, de realçar o excelente ambiente de trabalho que se vive na empresa.

De realçar também o papel da minha orientadora, Prof. Dr. Maria José Marcelino, a quem agradeço pela disponibilidade demonstrada desde o início.

Por fim, agradeço à minha família, namorada e amigos que estiveram sempre presentes e disponíveis nas alturas mais complicadas, que me incentivaram e apoiaram incondicionalmente ao longo de todo este percurso.

A todos, um muito obrigado.

Índice

Capítulo 1 Introdução.....	1
1.1 Enquadramento	2
1.2 Objetivos	3
1.3 Estrutura do documento	3
Capítulo 2 Estado da Arte.....	5
2.1 Gravity.....	5
2.2 Basecamp.....	7
2.3 Lighthouse.....	8
2.4 GoPlan.....	9
2.5 Teambox	11
2.6 Wunderkit	12
2.7 Redmine.....	14
2.8 Preview (alpha)	15
2.9 Comparação entre as aplicações analisadas	18
Capítulo 3 Metodologia do projeto	19
3.1 Equipa do projeto	19
3.2 Processo de desenvolvimento	19
3.3 Riscos	19
3.3.1 Curva de aprendizagem de novas tecnologias.....	20
3.3.2 Requisitos pouco específicos	20
3.3.3 Defeitos na arquitetura.....	20
Capítulo 4 Planeamento do projeto	21
4.1 Plano de estágio	21
4.1.1 Primeiro Semestre	21
4.3.2 Segundo Semestre	24
Capítulo 5 Projeto Preview	28

5.1 Caracterização da aplicação	28
5.2 Requisitos.....	28
5.2.1 Levantamento de requisitos	28
5.2.2 Atores do projeto	29
5.2.3 Descrição geral do sistema.....	29
5.2.4 Prioridades dos requisitos.....	30
5.3 Arquitetura	31
5.3.1 Arquitetura Técnica	31
5.3.2 Arquitetura Funcional	31
5.3.3 Arquitetura Tecnológica.....	32
5.3.4 Arquitetura do <i>site</i>	36
5.4 Funcionalidades	39
5.4.1 Filtros interativos nas tabelas	39
5.4.2 Formulários sem <i>refresh</i>	40
5.4.3 Salvar automaticamente.....	40
5.4.4 Gestão de imagens <i>findings</i>	40
5.4.5 Função de Encriptação.....	41
Capítulo 6 Performance e validação.....	43
6.1 Performance Base de dados.....	43
6.2 Validação.....	43
6.2.1 Testes funcionais	44
6.2.2 Testes de compatibilidade de browsers	44
Capítulo 7 Conclusões	45
7.1 Trabalho futuro	45
Capítulo 8 Lista de Citações.....	47
Capítulo 9 Bibliografia	48
Anexos.....	49

Índice de Figuras

Figura 1 - Página inicial da aplicação Gravity	5
Figura 2 - Visão geral de um determinado projeto	6
Figura 3 - Página inicial de um projeto (visão geral)	7
Figura 4 - Lista de tarefas a executar no Basecamp	7
Figura 5 - Página inicial de uma empresa	9
Figura 6 - Calendário de eventos da empresa	10
Figura 7 - Página inicial de uma empresa	11
Figura 8 - Calendário de eventos da empresa	11
Figura 9 - Página inicial de uma empresa	12
Figura 10 - Calendário de eventos da empresa	13
Figura 11 - Página inicial da aplicação com uma visão geral do projeto	14
Figura 12 - Lista de atividades do projeto	15
Figura 13- Página inicial da aplicação Preview (alpha)	15
Figura 14 - Página inicial de um projeto	16
Figura 15 - Lista de findings de um projeto da aplicação Preview	17
Figura 16 - Previsão inicial das atividades para o primeiro semestre	22
Figura 17 - Diagrama de Gantt final das atividades do primeiro semestre	22
Figura 18 - Previsão inicial das atividades para segundo semestre	25
Figura 19- Diagrama de Gantt final das atividades do segundo semestre	26
Figura 20 - Arquitetura inicial da aplicação	31
Figura 21 - Representação do MVC	32
Figura 22 - Arquitetura tecnológica do sistema	33
Figura 23 - Comparação entre PHP, Rails e Java segundo escalabilidade, tempo de desenvolvimento, ferramentas de desenvolvimento e manutenção. [2, p. 7]	34
Figura 24 - Diagrama de fluxo da <i>framework</i> CodeIgniter [3]	35
Figura 25 - Mapa das páginas da aplicação	37
Figura 27 - Lista de findings de um projeto	38

Figura 28 - Página de gestão de utilizadores da empresa.....	38
Figura 29 - Página de um <i>finding</i>	39
Figura 29 - Estrutura das imagens na aplicação	41
Figura 30 - Arquitetura final da aplicação Preview.....	46

Índice de Tabelas

Tabela 1- Comparação entre aplicações analisadas	18
--	----

Lista de Acrónimos

Acrónimos	Significados
AJAX	Asynchronous Javascript and XML
CSS	Cascading Style Sheets
CSV	Comma-separated values
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure
JSON	JavaScript Object Notification
MD5	Message Digest 5
OS	Operating System
PHP	PHP: Hypertext Preprocessor
SHA	Secure Hash Algorithm
SQL	Structured Query Language

Capítulo 1

Introdução

A Tangível é uma empresa sediada no Instituto Pedro Nunes, em Coimbra, especialista em usabilidade e *design* de interação. Tem como missão ajudar outras instituições e empresas a desenvolver produtos, aplicações e *websites* com um suporte efetivo à produtividade e à satisfação pessoal dos utilizadores como interfaces intuitivas e de rápida aprendizagem.

Na elaboração de projetos com clientes, as empresas de usabilidade produzem documentação específica da área de usabilidade, como por exemplo, relatórios de *expert reviews*, propostas de *visual designs*, *personas*, relatórios de testes de usabilidade, entre outras. Relatórios que posteriormente necessitam de ser entregues aos seus clientes para que estes procedam à melhoria das suas aplicações.

No mercado, a existência de ferramentas de apoio a especialistas de usabilidade ainda é muito escassa. A maior parte destas ferramentas são bastante genéricas, de forma a atingir o maior número de clientes possível. Estas são principalmente focadas para a construção/gestão de tarefas de projetos e não para a gestão dos conteúdos dessas tarefas, permitindo assim uma fácil visualização da evolução desses conteúdos.

Sendo assim, com o objetivo de corrigir essa falha no mercado, em 2010 a Tangível criou um produto, o Preview (versão alfa), que apesar de ser uma versão limitada do ponto de vista de funcionalidades e de segurança, já representa uma mais-valia no apoio à produção de relatórios de problemas encontrados, assim como na divulgação desses relatórios aos seus clientes.

No entanto, a empresa sentiu necessidade de melhorar a aplicação através da adição de novas funcionalidades, melhoria da usabilidade nas funcionalidades existentes e melhoria na segurança da mesma.

A existência de ferramentas deste tipo mais completas é muito importante, por um lado para facilitar o desenvolvimento de relatórios de usabilidade, por outro para garantir a segurança das informações (às vezes delicadas) dos clientes alvos de análise.

Assim, neste estágio é analisada e implementada uma aplicação Web denominada Preview 2.0, que vem tentar colmatar as lacunas existentes na versão anterior da aplicação.

Além disto, pretende-se que a aplicação ofereça aos clientes a possibilidade de terem um papel mais ativo, para que consigam acompanhar todo o desenvolvimento dos documentos de usabilidade inseridos no sistema, podendo comentar os vários documentos assim que estes vão sendo inseridos na aplicação. Por outro lado, pretende-se também possibilitar o controlo por parte dos clientes dos relatórios de usabilidade já executados, não confinando a aplicação apenas aos especialistas de usabilidade.

1.1 Enquadramento

Os relatórios de usabilidade são documentos produzidos para ajudar os clientes a detetarem problemas nas suas aplicações, classificá-los segundo um determinado grau de severidade e produzir recomendações para solucionar os problemas detetados.

Estes documentos produzidos podem ser de um ou mais tipos, consoante o necessário e o acordado com o cliente:

- **Relatórios de *expert review*:** são inventários de problemas, que por sua vez se denominam por *findings*, identificados por especialistas de usabilidade. Estes problemas de interação ou *User Experience*, são habitualmente classificados por severidade, possuem uma descrição e uma ou várias recomendações para resolução do problema. Geralmente para um rápido entendimento do problema e sua resolução, os relatórios são acompanhados por imagens, tanto a nível da descrição dos problemas como possíveis recomendações para os solucionar. Por vezes possuem também alguma justificação teórica tanto na descrição do problema como nas recomendações.
- **Elementos de *visual design*:** para a construção da interface gráfica são necessários vários elementos visuais, nomeadamente *mockups* que são protótipos (esboços rápidos) para futuramente implementar, *screens*, que são ecrãs que contêm as várias funcionalidades que irão formar a aplicação. E por fim *styleguides*, que contêm as informações mais específicas relativamente aos ecrãs, tais como a cor dos elementos existentes e suas dimensões.
- ***User profiling*:** este tipo de relatório de usabilidade define os perfis de utilizador que utilizam uma determinada aplicação ou que utilizarão, caso se trate de uma análise de requisitos para um projeto a desenvolver.
- **Testes de usabilidade:** são definidas tarefas a realizar pelos utilizadores, sendo estas constituídas por atividades a executar consoante o seu perfil. Ao executar essas tarefas são guardadas as observações e métricas de usabilidade que poderão consistir em texto, imagens ou vídeos, recolhidos por especialistas, durante a realização das tarefas.

Em suma, catalogam-se os problemas para facilitar a sua correção pela ordem do mais urgente ao menos urgente, por forma a que exista uma ou várias sugestões de como poderá ser corrigido cada problema.

1.2 Objetivos

Neste estágio pretende-se a construção de uma aplicação Web que sirva de apoio ao desenvolvimento de *expert reviews* pela Tangível e por outras empresas de usabilidade.

A aplicação apenas permitirá o acesso a pessoas que se autenticarem, nomeadamente funcionários de empresas de usabilidade e seus clientes, para que estes possam acompanhar todo o trabalho desenvolvido pelos especialistas de usabilidade.

Esta aplicação deverá permitir criar e gerir projetos, dentro desses projetos deverá ser possível gerir utilizadores (adicionar, remover e alterar os seus privilégios) e *findings*.

Deverá ainda possuir ferramentas de apoio ao técnico de usabilidade e também funcionalidades específicas para os clientes, nomeadamente saber, através de uma check-list de *findings* já corrigidos, se os problemas já foram resolvidos e ter uma perceção dos que ainda estão por resolver pelas empresas clientes.

1.3 Estrutura do documento

Este relatório está estruturado da seguinte forma:

Capítulo 1: Introdução – Este capítulo enquadra o leitor dos temas que irão ser discutidos neste documento. Nele é descrito o âmbito do projeto e os seus objetivos.

Capítulo 2: Estado da Arte – Neste capítulo é efetuada uma análise a várias aplicações semelhantes à que se pretende desenvolver de gestão do desenvolvimento de projetos e é feita uma comparação, tendo em conta o objetivo do estágio, e suas características.

Capítulo 3: Metodologia do projeto – No capítulo 3 é feita uma descrição da metodologia utilizada, a equipa e os riscos do projeto.

Capítulo 4: Planeamento do projeto – No capítulo 4 pode-se averiguar como foi gerido o projeto, qual foi a calendarização das atividades do primeiro e segundo semestre.

Capítulo 5: Projeto Preview – Este capítulo possibilita uma caracterização da aplicação segundo as funcionalidades, requisitos e o desenho da arquitetura. São ainda exibidos alguns *screens* da aplicação.

Capítulo 6: Performance e validação – São apresentadas algumas otimizações assim como testes de performance e validação da aplicação.

Capítulo 7: Conclusões – São apresentadas as principais conclusões relativamente ao projeto de estágio, assim como o planeamento para trabalho futuro.

Capítulo 8: Listas de citações – Neste capítulo são indicadas as referências que surgiram ao longo do relatório.

Capítulo 9: Bibliografia – Por fim, no nono capítulo, apresentam-se uma lista de todos os documentos que foram consultados no decorrer do estágio.

Anexos – Esta secção contém documentos adicionais produzidos no âmbito deste projeto.

Capítulo 2

Estado da Arte

Não existem ferramentas na área de usabilidade que permitam realizar relatórios de *expert review*, criação de testes de usabilidade e construção de *personas*, por exemplo. No entanto, existem ferramentas que contêm funcionalidades genéricas que podem ser interessantes considerar para o desenvolvimento deste projeto, como é o caso de gestão de pessoas, de projetos, de *bug fixing*, ou mesmo de controlo de versões.

As ferramentas escolhidas para o estudo são aplicações *web-based* e assim sendo deixa de ser necessário instalar a aplicação num computador local, bastando apenas um computador com acesso à Internet. O Web 2.0 propõe o uso de diversas tecnologias nomeadamente AJAX e Comet entre outras, de forma a enriquecer a usabilidade do utilizador, com interfaces mais rápidas e fáceis de usar.

De seguida são descritas algumas aplicações com as características que consideramos mais relevantes para servirem de ponto de referência ao desenvolvimento do Preview.

2.1 Gravity

A aplicação *Web Gravity* é uma ferramenta para gestão de projetos. É muito flexível e com um *design* bastante simples. Para manter os utilizadores informados, utiliza um sistema de notificação por *e-mail* e para o fácil manuseamento e interatividade das ferramentas utiliza a tecnologia *drag-and-drop*. Apresentam-se, de seguida, figuras de algumas funcionalidades da aplicação. A figura 1 representa o ecrã inicial deste projeto.



Figura 1 - Página inicial da aplicação Gravity

Quando o utilizador entra na aplicação é-lhe apresentado as listas de projetos em que se encontra inscrito. Nessa página pode visualizar as atividades e atualizações mais recentes de todos os projetos em que se integra.

A figura 2 reproduz a visão geral de um determinado projeto. É possível ver um texto introdutório (que facilmente pode ser alterado por um utilizador com permissões) e as atividades mais recentes desse projeto.



Figura 2 - Visão geral de um determinado projeto

Estas funcionalidades permitem que um novo utilizador tenha uma curva de aprendizagem menor em relação a ferramentas de gestão de projetos *online*, possibilitando a rápida aprendizagem dos conteúdos do projeto e mantendo o utilizador atualizado relativamente às alterações que o projeto sofreu.

O Gravity revelou-se bastante completo para o acompanhamento no desenvolvimento de projetos de *software*, uma vez que possui as seguintes características:

- Definição de papéis para os utilizadores e controlo de acessos;
- Geração de calendários;
- Notificações por *e-mail*;
- Registo de tempo e atividades;
- Possibilidade de exportar projetos para ficheiros CSV;
- Marcadores para organizar o conteúdo do projeto.

Apesar do seu elevado número de funcionalidades esta ferramenta torna-se desajustada para as necessidades de um projeto de usabilidade, isto é, oferece muitas opções que acabam por não se revelar úteis para uma equipa de especialistas de usabilidade. Um dos pontos fracos é a inexistência de uma funcionalidade que permita adicionar imagens para melhor perceção dos problemas.

2.2 Basecamp

A aplicação Basecamp foi desenvolvida pela equipa 37 Signals e assemelha-se muito ao Gravity no que diz respeito à gestão de projetos de forma simples e rápida. É uma ferramenta bastante madura, estando disponível desde 1999 e possui um grande mercado. Mais de 5 milhões de pessoas usaram esta ferramenta, sendo importante a sua análise para perceber os motivos do seu sucesso, que levaram os utilizadores a optarem por esta em detrimento de outras ferramentas disponíveis.

De seguida mostram-se algumas funcionalidades disponíveis. A figura seguinte representa o ecrã de entrada de um projeto.

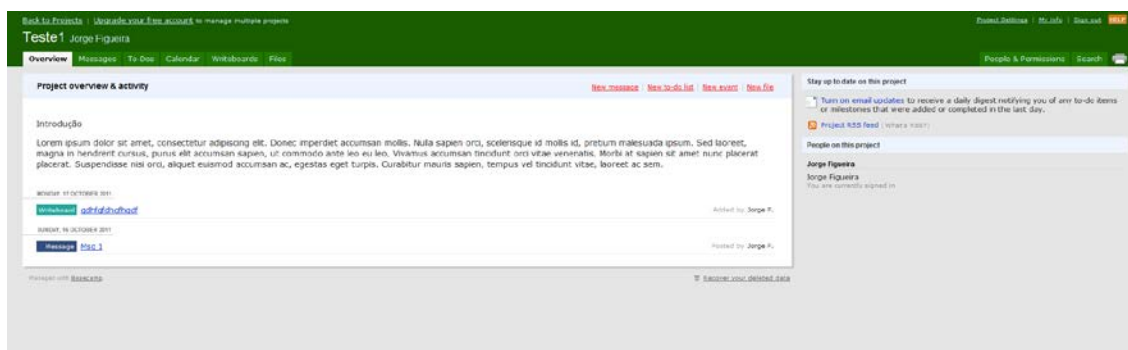


Figura 3 - Página inicial de um projeto (visão geral)

Como podemos observar, esta ferramenta apresenta no ecrã inicial do projeto uma pequena nota introdutória e um registo com as últimas atividades realizadas. No entanto não possui o registo das modificações efetuadas.

Este sistema permite definir metas, visualizar as metas que se aproximam e as que já ocorreram. Na figura 4 apresenta-se uma página onde se encontra uma lista de metas definidas para um projeto, sendo esta bastante útil para a organização por parte dos colaboradores.

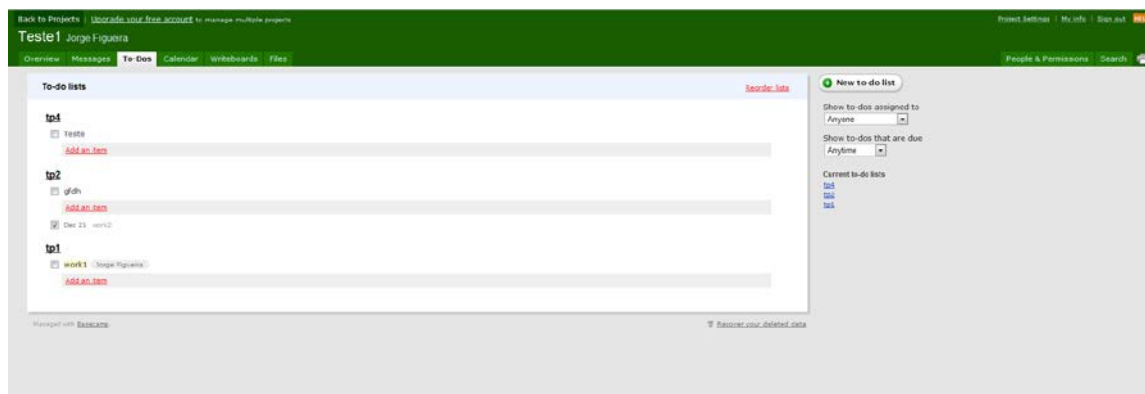


Figura 4 - Lista de tarefas a executar no Basecamp

Destacam-se ainda algumas funcionalidades no Basecamp:

- Multi-projeto: permite gerir vários projetos em simultâneo;
- Perfis de acesso: gestão fácil de permissões de acesso aos diferentes projetos;
- Gestão de objetivos/metast: oferece uma visão simplificada das fases de determinado projeto;
- Gestão de atividades/tarefas: permite criar uma lista de tarefas de um projeto;
- Gestão de ficheiros: submeter e descarregar ficheiros com outros utilizadores/membros projeto ou terceiros e gerir as suas versões;
- Controlo e gestão do tempo: Permite o controlo das horas gastas nas atividades, sendo possível saber quem é o responsável pelo atraso na execução de uma tarefa;
- *Brainstorming*: disponibiliza áreas de trabalho para potenciar a criatividade das equipas;
- Writeboards: permite a elaboração conjunta de artigos com controlo de versões;
- Mensagens instantâneas: sistema de *chat* interno para facilitar a comunicação entre as partes envolvidas;
- Suporte/formação: suporte *online* aos utilizadores.

Recentemente, o Basecamp foi atualizado contendo um requisito muito solicitado pelos utilizadores. Esta nova funcionalidade permite recuperar a informação do *site* no caso de, por acidente, o utilizador ter perdido a ligação à Internet ou atualizado a página sem intenção. Através desta ferramenta, o utilizador não precisa de voltar a introduzir os dados, necessitando apenas de voltar a abrir a página.

2.3 Lighthouse

Lighthouse é uma ferramenta desenvolvida pela ENTP e é bastante idêntica às anteriormente analisadas. Esta ferramenta possibilita reportar defeitos e problemas encontrados em projetos. Apesar da semelhança com as outras aplicações Web analisadas esta é a menos completa, contendo menos funcionalidades e possibilitando apenas a criação de *tickets*, *milestones* e mensagens.

Relativamente à criação de texto, nomeadamente *tickets* e mensagens é necessário bastante conhecimento a priori, isto é, a aplicação é pouco intuitiva, necessitando de introduzir *tags* especiais no texto para inserir imagens, colocar o texto a negrito, etc. Assim sendo, apesar de adicionar bastante personalização ao texto é pouco prática para o utilizador.

Destacam-se as seguintes funcionalidades nesta aplicação:

- Boa organização dos conteúdos com um sistema de *tags* à semelhança do Gravity;
- Pesquisas avançadas, permitindo filtrar por bastantes parâmetros;
- Visualização do progresso das metas;
- Funcionalidades que podem ser executadas através do sistema de *e-mail* não necessitando de estar conectado ao programa.

2.4 GoPlan

O GoPlan é um produto desenvolvido pela empresa portuguesa Webbreakstuff sediada em Coimbra [1]. É uma aplicação *Web* bastante interativa e intuitiva, sendo a interface da aplicação muito bem estruturada e permitindo uma rápida aprendizagem da aplicação.

Possui um ecrã de resumo personalizável (como se mostra na figura 5) permitindo ao utilizador obter o resumo das atividades recentes e, possibilitando ainda, a adição do logotipo da empresa.

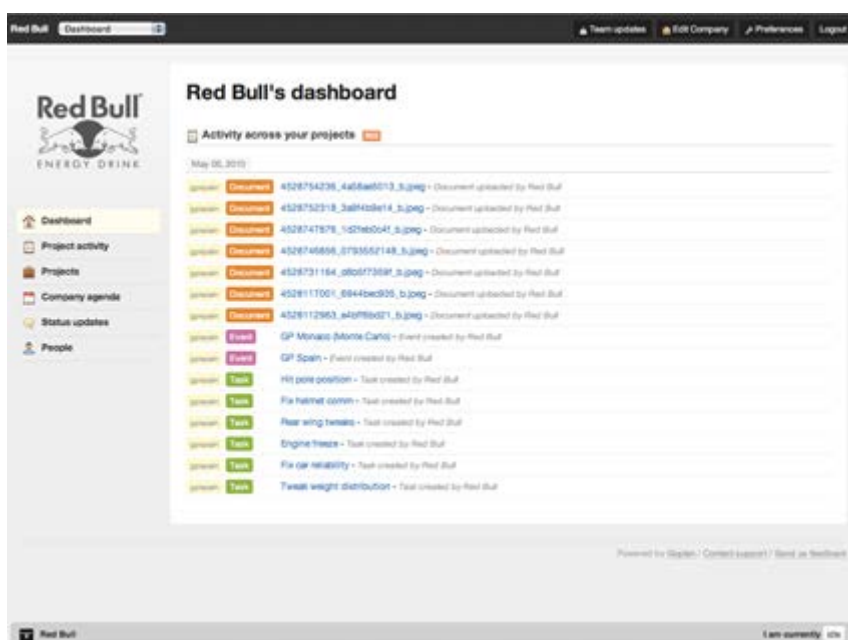


Figura 5 - Página inicial de uma empresa

Muito semelhante às outras aplicações aqui analisadas, esta página resume todos os projetos que a empresa possui. Para além disso é possível gerir facilmente os colaboradores, consultando as atividades que cada um está a realizar, assim como o calendário global da empresa, que abrange todos os projetos.

Assim, é possível aceder à agenda da empresa sendo disponibilizadas as datas de todos os eventos realizados pelos membros da empresa.

Para além da consulta dos eventos através do calendário é possível sincronizar com as aplicações iCal da Apple ou o Google Calendar.

Existe ainda um módulo que permite a discussão dos vários intervenientes do projeto, uma espécie de fórum em que se pode criar temas de discussão e dialogar acerca dos mesmos.

As tarefas são atividades que podem ser associadas a metas no desenvolvimento do projeto. Estas poderão ser atribuídas a um utilizador, que por sua vez pode definir uma data para a sua realização, adicionar uma *tag* ou definir uma categoria para uma melhor organização.

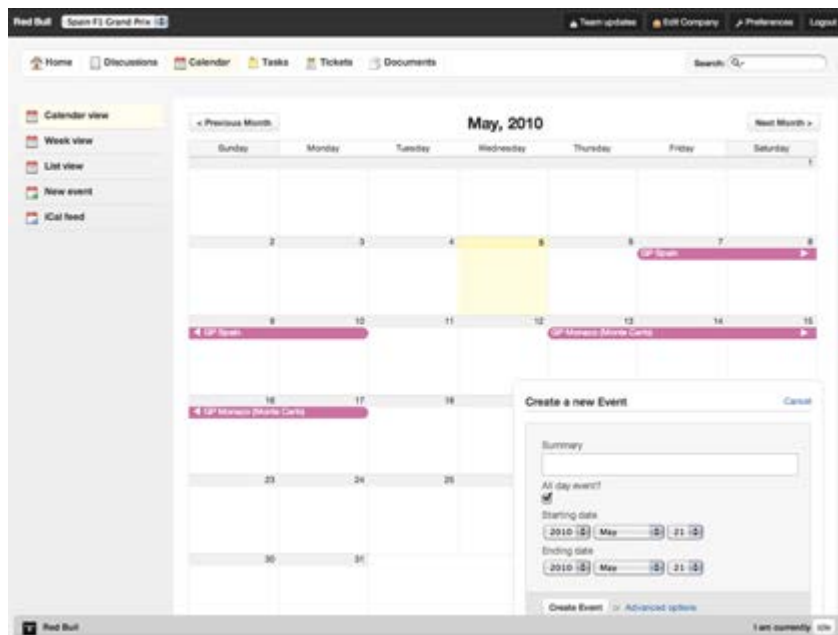


Figura 6 - Calendário de eventos da empresa

De salientar alguns fatores interessantes da aplicação:

- Existência de um módulo de discussão que permite debater ideias entre os vários intervenientes do projeto;
- Calendário prático;
- Monitorização do estado do utilizador em tempo real, sendo possível saber o que o utilizador se encontra a desenvolver naquele momento, assim como o que já fez até aquela data;
- Interface bastante intuitiva e prática, sendo fácil a navegação e utilização pelos vários módulos existentes;
- Controlo de versões de ficheiros.

2.5 Teambox

A Teambox é uma aplicação Web bastante interativa e intuitiva. Possui uma interface muito bem estruturada, permitindo a rápida aprendizagem da aplicação.

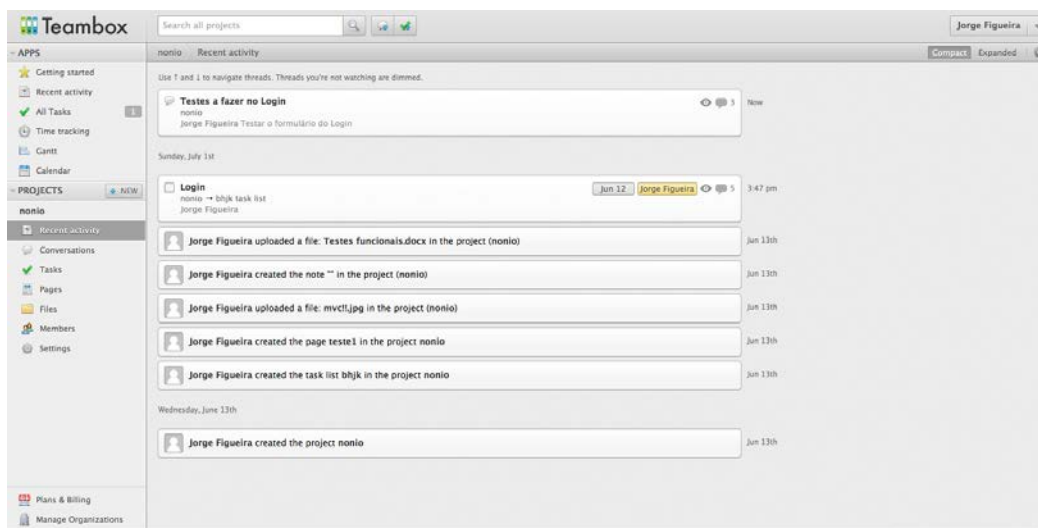


Figura 7 - Página inicial de uma empresa

Para além disso é possível gerir facilmente os colaboradores, consultando as atividades que cada colaborador está a realizar, assim como o calendário global da empresa, que abrange todos os projetos.

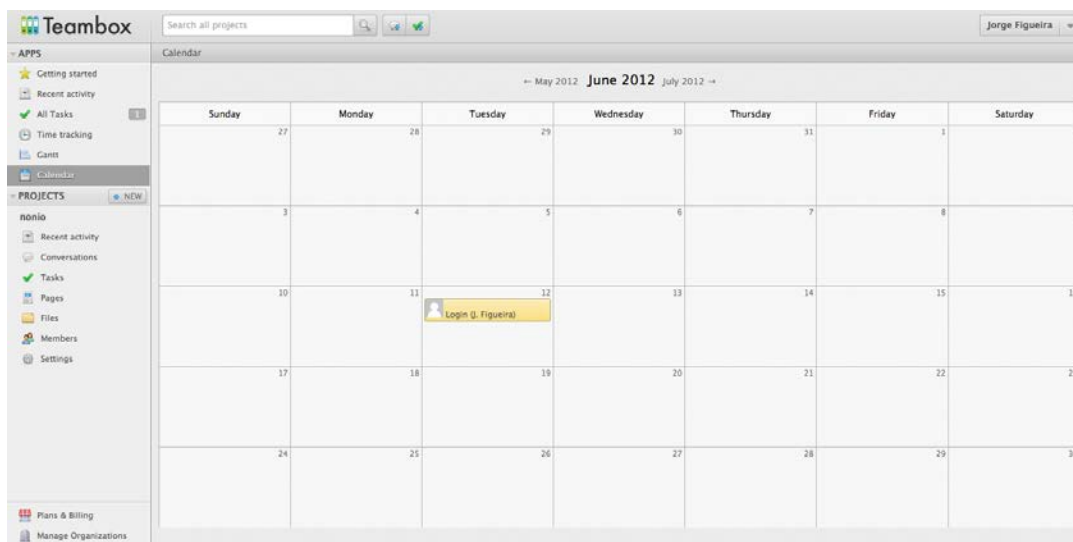


Figura 8 - Calendário de eventos da empresa

É possível aceder à agenda da empresa sendo disponibilizadas as datas de todos os eventos realizados pelos membros da empresa.

Possui ainda integração com várias aplicações, nomeadamente Dropbox e box para sincronismo de ficheiros e Github e Pivotal Tracker para integração com aplicações de gestão de tarefas. Para além disso, também possibilita a integração com várias aplicações da Google nomeadamente Gmail, GoogleCal e +RSS.

Existe ainda um módulo, que permite a discussão entre os vários intervenientes de um projeto, uma espécie de fórum onde se pode criar temas de discussão e dialogar acerca dos mesmos.

As tarefas são atividades que podem ser associadas a metas no desenvolvimento do projeto. Estas podem ser atribuídas a um utilizador, que pode definir uma data para a sua realização, adicionar uma *tag* ou definir uma categoria para uma melhor organização.

Não é possível alterar os documentos introduzidos na aplicação, estes apenas podem ser inseridos e eliminados. Para além disso, é possível utilizar o Dropbox e o GoogleDocs, facilitando um pouco a partilha de documentos entre os utilizadores da aplicação.

De salientar alguns características interessantes da aplicação:

- Existência de um módulo de discussão que permite debater ideias entre os vários intervenientes no projeto;
- Interface bastante intuitiva e prática, sendo fácil a navegação e utilização pelos vários módulos existentes;
- Fácil integração com várias plataformas existentes no apoio a empresas.

2.6 Wunderkit

O Wunderkit é uma aplicação Web bastante interativa e intuitiva, a sua interface faz esquecer que se trata de uma aplicação Web, sendo bastante rápida a sua aprendizagem.

Para além disso, é possível utilizar a aplicação em diferentes plataformas tais como: *browser*, aplicação no computador e *iPhone*.

Possui um ecrã de resumo personalizável, figura 9, permitindo ao utilizador ter um sumário das atividades recentes.

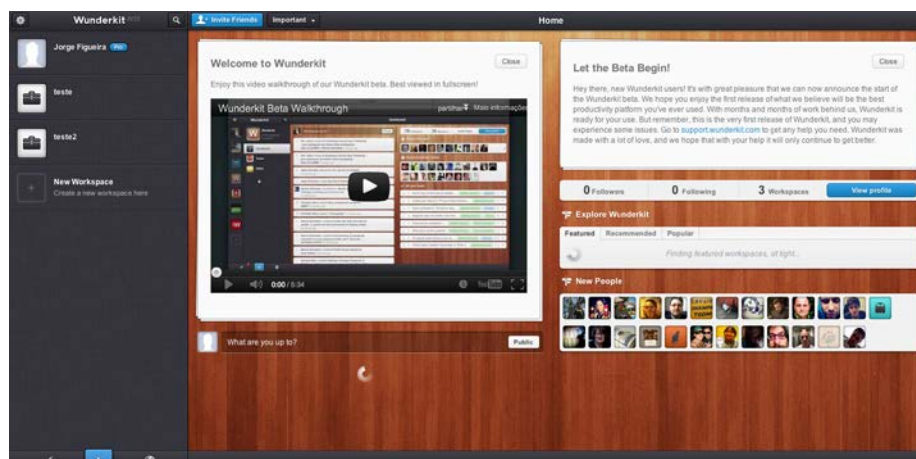


Figura 9 - Página inicial de uma empresa

Muito semelhante às aplicações analisadas anteriormente, esta página resume todos os projetos que a empresa possui. Para além disso é possível gerir facilmente os colaboradores, consultando as atividades que cada colaborador está a realizar, assim como o calendário global da empresa, que abrange todos os projetos.

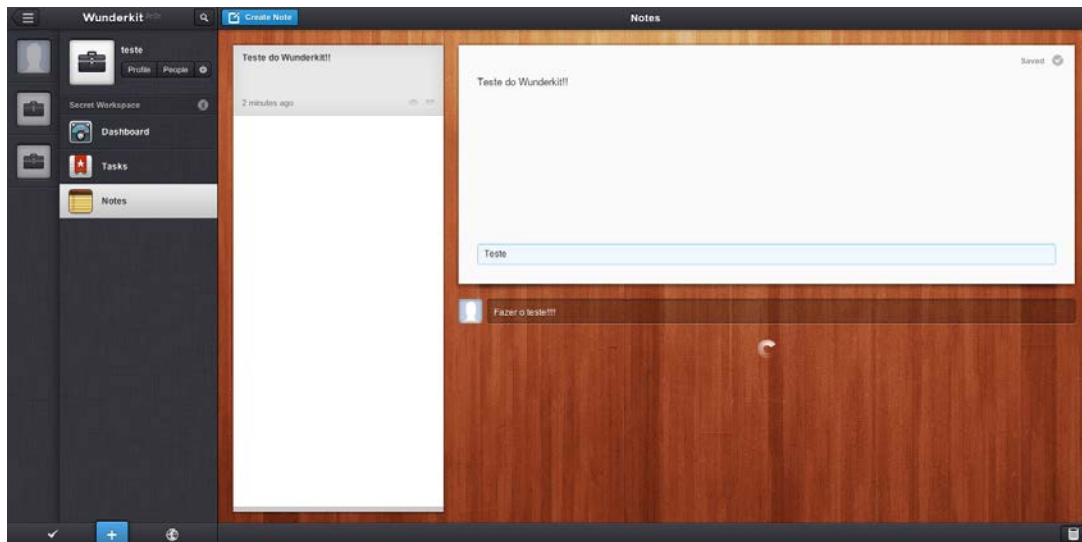


Figura 10 - Calendário de eventos da empresa

É possível aceder à agenda da empresa sendo disponibilizadas as datas de todos os eventos realizados pelos membros da empresa.

Existe ainda um módulo, que permite a criação de notas bastante personalizáveis relativamente ao projeto.

De salientar alguns fatores interessantes da aplicação:

- Existência de um módulo que permite debater ideias entre os vários intervenientes no projeto;
- Interface bastante intuitiva e prática, sendo fácil a navegação e utilização pelos vários módulos existentes.

2.7 Redmine

A aplicação Web Redmine desenvolvida em *Ruby on Rails*, destaca-se das anteriores, em primeiro lugar por ser *open-source*, possuindo uma licença “GNU General Public License V2”(GPL). Em segundo lugar, esta aplicação é bastante flexível, possuindo bastantes módulos para gerir uma maior área de projetos. A figura 11 é um exemplo de uma página inicial de um projeto e apresenta uma visão geral do mesmo, o tempo total gasto no decorrer do projeto, os seus membros e as tarefas executadas e a executar.

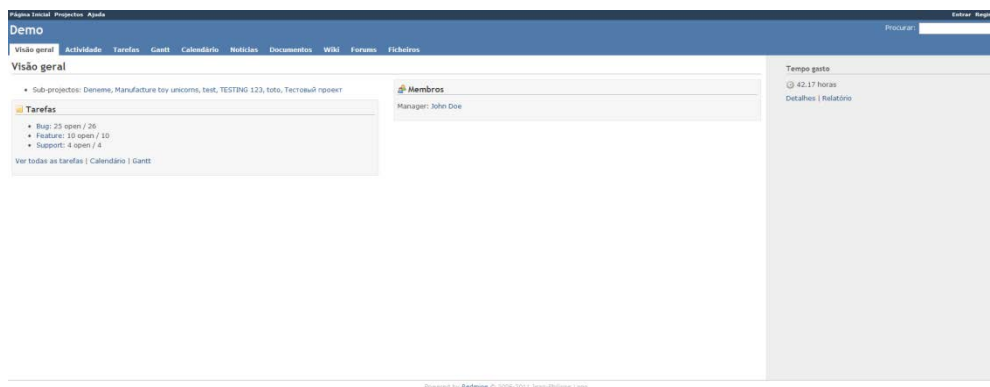
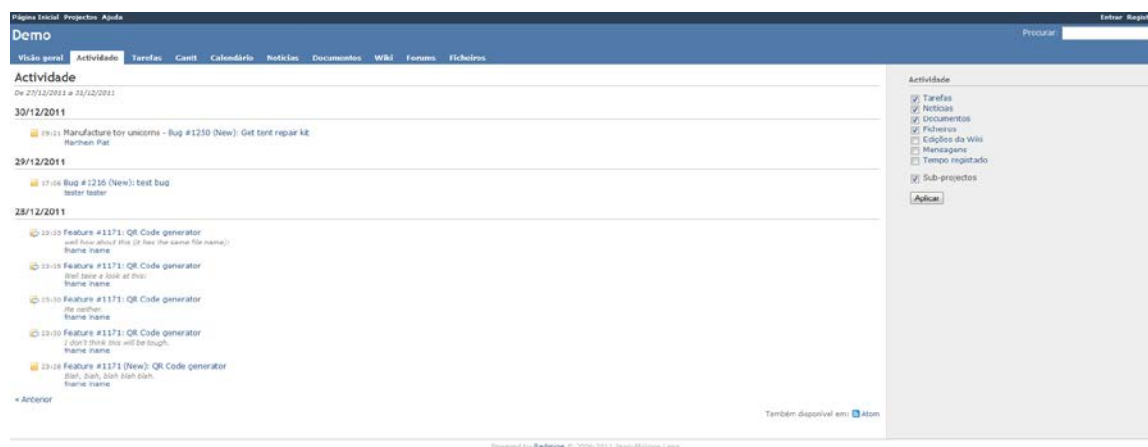


Figura 11 - Página inicial da aplicação com uma visão geral do projeto

Com esta página inicial o utilizador rapidamente verifica as últimas mudanças realizadas na aplicação, assim como, navega nos restantes módulos desta. Na figura 12 é possível visualizar as atividades recentes existentes no projeto. Para facilitar a pesquisa de uma, é permitido filtrar as atividades desejadas.



Com esta página o utilizador monitoriza o histórico de todas as atividades realizadas até à data, mantendo os utilizadores ao corrente de todas as alterações efetuadas no projeto, assim como, do tempo despendido em cada uma.

2.8 Preview (alpha)

O Preview (alpha) é um produto desenvolvido pela Tangível que, de certa forma, tenta colmatar as lacunas das aplicações acima analisadas, focalizando-se exclusivamente em projetos de usabilidade.

Este possui um ecrã inicial, figura 13, com uma lista dos projetos ainda em desenvolvimento e

Figura 12 - Lista de atividades do projeto

os já arquivados.

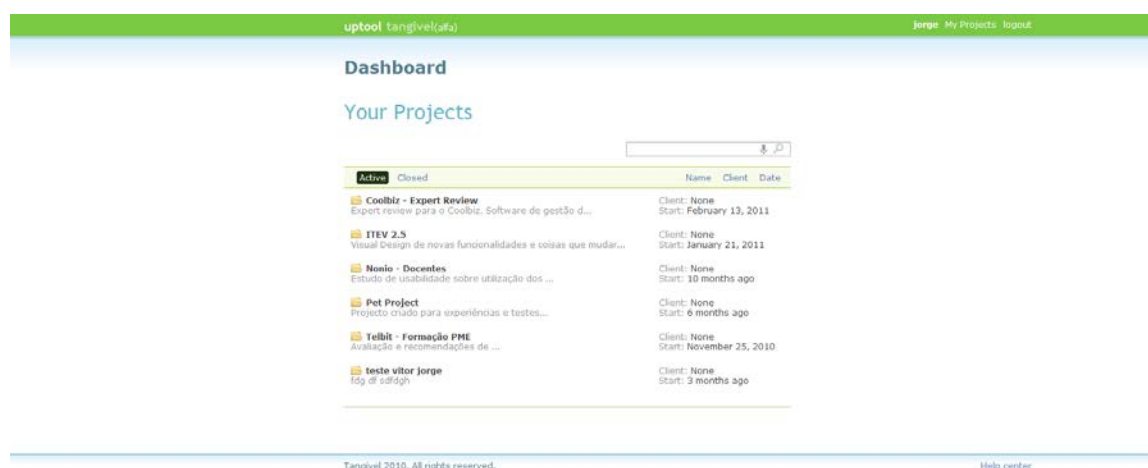


Figura 13- Página inicial da aplicação Preview (alpha)

Quando é selecionado um projeto, o utilizador é direcionado para a página inicial desse documento. Nessa página é fornecida alguma informação sobre esse projeto, nomeadamente as atividades recentes e os membros que dele fazem parte.

The screenshot displays the 'Nonio - Docentes' project page. At the top, there's a green header with 'uptool tangivel@ta' and 'Jorge My Projects Logout'. Below this, a blue navigation bar contains 'Project', 'Expert Review', 'User Tests', 'Details', and 'Users'. The main content area is divided into several sections:

- Activity:** A list of recent activities with timestamps. Examples include:
 - Friday, 06 Jan 2012: 'finding edited: Página de selecção da turma quando só existe uma turma', 'finding edited: Form de acesso às disciplinas de anos anteriores', 'finding edited: Página de escolha da disciplina com muita informação', 'finding edited: Labels desnecessários', 'finding edited: Mensagem de confirmação pouco distinta', 'finding edited: Feedback sobre o período de entrega', 'finding edited: Janela de escolha da edição'.
 - Thursday, 05 Jan 2012: 'finding edited: Ajuda apenas para pautas e folhas de presença'.
 - Saturday, 03 Sep 2011: 'finding created: Divisão dos dados profissionais por duas páginas', 'finding created: Divisão dos dados pessoais por duas páginas'.
- Progress:** A section indicating 'There aren't tasks assigned to this project yet. Define one.' and a 'Description' of the project: 'Estudo de usabilidade sobre utilização dos docentes da plataforma Nonio.' It also lists 'People' involved, categorized by role (Tangivel, FCTUC).
- Milestones:** A section stating 'There aren't milestones created for this project. Go [here](#) to create one.'

At the bottom, there's a footer with 'Tangivel 2010. All rights reserved.' and a 'Help contact' link.

Figura 14 - Página inicial de um projeto

Selecionado o módulo dos *expert reviews* é apresentada uma lista de *findings* contendo todos os problemas encontrados na aplicação. Cada *finding* poderá ter mais do que uma versão.

Ao selecionarmos um *finding* facilmente verificamos que a página fica dividida em 3 partes. A primeira parte contém uma breve explicação, a segunda compreende uma recomendação para o problema e a terceira parte inclui comentários para esse *finding*. Na edição desses *findings* concluímos que só é possível adicionar uma imagem para a explicação do problema e outra para a recomendação, limitando assim o trabalho do especialista de usabilidade.

Findings

All findings
110 findings

Name Last modified Type Severity Search findings

Name	Last modified	Type	Severity
38 Header muito grande		GUI	Low
97 Mensagens de erro são muito complexas		GUI	Medium
44 Sistema de filtragem do calendário pouco óbvio		GUI	Low
98 Alinhamentos ao centro nas mensagens		GUI	Medium
143 Links não parecem ser links		GUI	High
144 Acesso aos detalhes da informação nas listas		GUI	Medium
145 Nomes dos alunos em maiúsculas e sem acentos		GUI	Medium
100 Sem opção de recuperar a palavra-chave		Structural	Critical
27 Possível adicionar observações apenas na pauta impressa		GUI	Low
30 Necessidade de imprimir, assinar e ir entregar as pautas		Structural	High
73 Difícil saber que existem funcionalidades novas		Structural	High
29 Sem indicação de que a sessão expirou		GUI	Medium
31 Perda de contexto ao expirar a sessão		Structural	Critical
146 Legendas quando não existem dados		GUI	Low
147 Checkboxes à direita dos labels		GUI	Low
35 Falta de feedback ao alterar uma nota		GUI	Medium
46 Página de alterar nota confusa		Structural	High
37 Perca de contexto ao alterar uma nota		GUI	Low
148 Tamanho e tipo de letra		GUI	High
28 Impossibilidade de indicar que o aluno faltou à prova oral		Structural	Medium
172 Lista de uma turma sem opção de Exportar		GUI	Medium
173 Indicação de que a inscrição aguarda aprovação		GUI	Low

New finding

All findings

- Ajuda
- Alunos
- Aulas
- Aulas - Alterar
- Aulas - Delegar
- Aulas - Sumários
- Disciplina
- Feedback
- Fóruns de Discussão
- Geral
- Header
- Horário de Atendimento
- Listas
- Login
- Materiais de Apoio
- Menu
- Notificações
- Pautas
- Perfil
- Serviço Docente
- Submissão de Trabalhos
- Uncategorized
- User Experience

Figura 15 - Lista de findings de um projeto da aplicação Preview

Este produto apresentava graves problemas de segurança, sendo possível visualizar outros projetos a partir do acesso a um projeto, apenas alterando o número no URL da página. Não sendo necessário grande conhecimento por parte dos utilizadores para tirar proveito desta lacuna de segurança, alguns clientes solicitaram a correção desta falha.

Outra falha encontrada na aplicação é a não utilização do protocolo de segurança Secure Sockets Layer(SSL). Este protocolo é bastante comum quando se pretende garantir segurança na navegação de páginas (HTTP) e é essencial para proteção dos conteúdos das empresas clientes.

2.9 Comparação entre as aplicações analisadas

Após a análise individual de aplicações concorrentes ou que mais se assemelham à aplicação pretendida foi elaborada uma tabela com as funcionalidades essenciais para o desenvolvimento da aplicação pretendida. A tabela 1 representa a comparação entre essas funcionalidades e as aplicações analisadas.

	Gravity	Basecamp	Lighthouse	GoPlan	Teambox	Wunderkit	Redmine	Preview(alfa)
SSL	Sim	Sim	Sim	Sim	Sim	Sim	Não	Não
Licença	Proprietário	Proprietário	Proprietário	Proprietário	Proprietário	Proprietário	Open Source	Proprietário
Notificações por e-mail	Sim	Sim	Sim	Sim	Não	Não	Sim	Não
Autenticação	Sim	Sim	Sim	Sim	Sim	Sim	Sim	Sim
Controlo acessos	Sim	Sim	Sim	Sim	Sim	Sim	Sim	Não
Gestão recursos humanos	Sim	Sim	Sim	Sim	Sim	Sim	Não	Não
Gestão de tarefas	Sim	Sim	Sim	Sim	Sim	Sim	Sim	Não
Gestão issues / findings	Sim	Não	Não	Sim	Não	Não	Sim	Sim
Guardar conteúdos automaticamente	Não	Sim	Não	Não	Não	Não	Não	Não
Pré-visualização dos conteúdos inseridos	Sim	Sim	Não	Não	Não	Não	Não	Não

Tabela 1- Comparação entre aplicações analisadas

A tabela 1 traduz claramente o que foi referido anteriormente e indica que estas aplicações, exceto o Preview, são aplicações genéricas, não sendo adequadas para a gestão, acompanhamento e armazenamento de *expert reviews*. Este tipo de projetos envolve a criação de *findings*, muitas das vezes necessitam de várias imagens para ajudar na perceção do problema e apresentar exemplos de possíveis resoluções. Esta estrutura muito própria é o calcanhar de Aquiles destas aplicações. O Preview tenta tirar partido desta falha e adiciona essa funcionalidade à aplicação. Todavia esta funcionalidade é muito limitada, sendo apenas possível adicionar uma imagem na criação do problema e outra na resolução deste.

Outra lacuna existente é a não distinção de clientes e especialistas de usabilidade. Na aplicação ambos desenvolvem papéis diferentes, sendo necessário para os especialistas colocar relatórios de usabilidade e para os clientes saber quais desses relatórios já se encontram concluídos. Isto dá à aplicação, não só o papel de um gestor de problemas, mas algo mais útil e profícuo no desenrolar do projeto.

Sendo assim, esta nova aplicação que se pretende desenvolver será viável, trazendo funcionalidades essenciais e sinergias ainda não exploradas com o objetivo de ajudar os profissionais de usabilidade no seu processo de trabalho e na comunicação com o(s) cliente(s).

Capítulo 3

Metodologia do projeto

Esta secção do relatório descreve a metodologia utilizada no processo de desenvolvimento do estágio.

3.1 Equipa do projeto

A equipa envolvida neste projeto é constituída por vários elementos da Tangível, nomeadamente o engenheiro José Campos, gestor do projeto, o engenheiro Vítor Carvalhinho, especialista de usabilidade e pelo aluno estagiário Jorge Figueira.

O papel do estagiário será executar todos os processos de desenvolvimento de *software* necessários para a criação da ferramenta Web Preview. A primeira fase deste estágio, correspondente ao 1º semestre do ano letivo, consistiu na realização do estado da arte, análise de requisitos de *software* e arquitetura de *software*. Numa segunda fase, correspondente ao 2º semestre, procedeu-se à implementação de diversos módulos da aplicação e por fim executaram-se os testes para deteção de possíveis falhas.

3.2 Processo de desenvolvimento

Para a realização deste estágio foi utilizada uma metodologia ágil. Esta pretende produzir resultados mais rapidamente e serviu para que, à medida que fossem construídos os módulos da aplicação estes fossem colocados *online* posteriormente facilitando, desta forma, o feedback da Tangível e produzindo versões de *software* cada vez mais próximas do objetivo proposto.

3.3 Riscos

Como em qualquer projeto de Engenharia de *software*, também este estágio esteve sujeito a alguns riscos. Estes foram tomados em consideração e foram estudados de forma a minimizar o seu impacto no desenvolvimento do projeto.

Esta secção tem como objetivo enumerar os riscos mais importantes e as medidas tomadas para os eliminar e/ou mitigar.

3.3.1 Curva de aprendizagem de novas tecnologias

O tempo de aprendizagem de novas tecnologias é difícil de estimar, pois varia de pessoa para pessoa consoante as suas características intrínsecas e cognitivas. Este fator deve ser considerado como um risco, visto que a duração do estágio é limitada e não se pode cometer grandes deslizes. Assim, no primeiro semestre foi realizado um estudo das principais tecnologias a utilizar, facilitando a adaptação do aluno no segundo semestre na parte da implementação.

O facto de algumas tecnologias já terem sido usadas na empresa noutros projetos, facilitou a aprendizagem das mesmas.

3.3.2 Requisitos pouco específicos

No início do projeto definiram-se os requisitos de forma pouco pormenorizada. No sentido de colmatar esta falha foram efetuadas várias reuniões para encontrar os melhores requisitos, tendo sido estes definidos pela equipa do projeto. Para além disso, o estado da arte contribuiu para uma melhor definição dos requisitos necessários.

3.3.3 Defeitos na arquitetura

Planear a arquitetura traduz-se na parte mais importante do projeto. Um dos riscos é desenhar uma arquitetura com defeitos. Esta poderá não apresentar dificuldades na fase de desenvolvimento, mas poderá ser um grande obstáculo para o suporte de novas funcionalidades.

A construção da arquitetura teve em consideração os riscos descritos anteriormente, para que além de suportar os requisitos inicialmente previstos possuía também espaço para evolução da aplicação.

Capítulo 4

Planeamento do projeto

Esta secção do relatório descreve a equipa do projeto, o processo e a duração das metas necessárias para a conclusão deste estágio.

4.1 Plano de estágio

4.1.1 Primeiro Semestre

Em seguida apresentam-se dois diagramas de Gantt relativos ao primeiro semestre. A figura 16 representa a previsão inicial das atividades para o esse semestre e a figura 17 contém o Gantt real para o mesmo período de tempo. As oscilações existentes atribuem-se a dois fatores: primeiro pela necessidade de mais tempo para a realização das tarefas e segundo pela sobreposição de horas de trabalho com cadeiras do primeiro semestre.

Preview

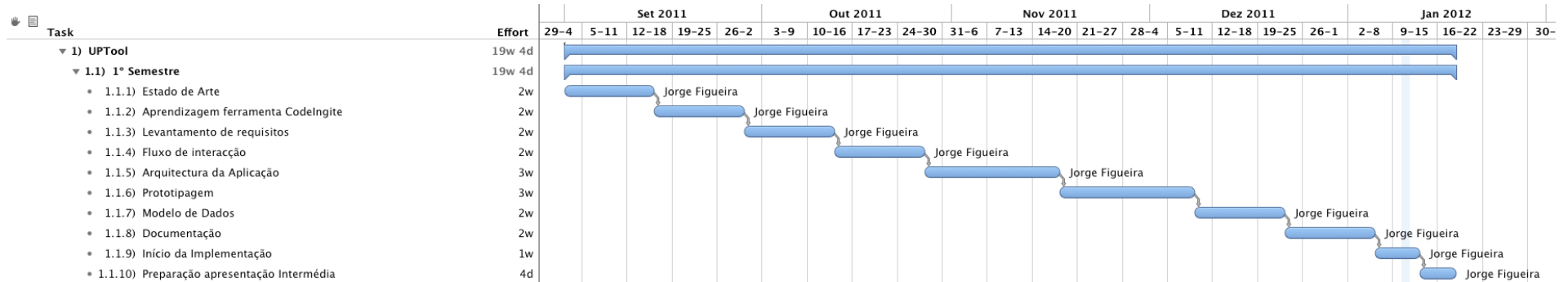


Figura 16 - Previsão inicial das atividades para o primeiro semestre

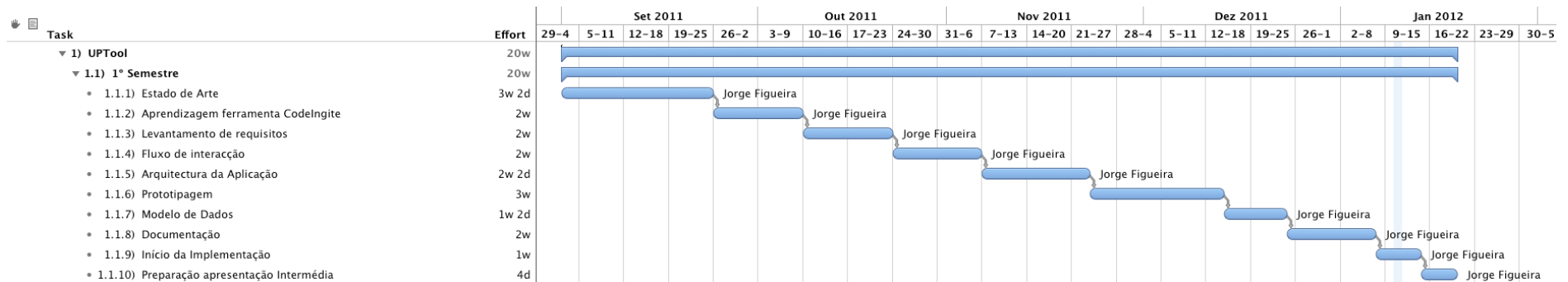


Figura 17 - Diagrama de Gantt final das atividades do primeiro semestre

Estes diagramas correspondem ao primeiro semestre, desde o dia 1 de Setembro de 2011 a 20 Janeiro de 2012 em regime de *part-time*. Neste semestre era essencial a análise das aplicações concorrentes, a análise de requisitos, a prototipagem, a arquitetura da aplicação e o modelo de dados.

Estado de Arte

No início do estágio foram analisadas aplicações concorrentes existentes no mercado. Esta análise tinha como objetivo principal recolher ideias das funcionalidades existentes e também encontrar as suas falhas, de forma a construir uma aplicação mais completa e útil para os especialistas de usabilidade.

Aprendizagem da ferramenta CodeIgniter

Foi estudada o CodeIgniter para uma maior familiarização desta *framework* conhecendo o seu fluxograma e a sua metodologia. Para além disso, foram analisadas formas de introduzir segurança essencial para proteger os dados dos projetos, nomeadamente utilização de SSL e proteção contra SQL *injection*. De referir ainda que, nesta tarefa foram também analisados *plugins* (jQuery, DataTables) que permitem uma maior interatividade com a aplicação.

Levantamento de requisitos

Nesta atividade foi feito o levantamento dos requisitos essenciais que a aplicação deveria suportar de forma a satisfazer as necessidades dos diversos intervenientes do projeto.

Fluxo de interação

Nesta atividade foi elaborado o fluxo de interação do CodeIgniter.

Arquitetura da aplicação

Foi desenhada a arquitetura da aplicação que melhor se adapta à criação do projeto Preview. Foram tomadas todas as diligências, tanto a nível tecnológico como funcional para a execução do projeto.

Prototipagem

Para melhor avaliação das necessidades existentes na aplicação foi elaborado um conjunto de protótipos. Estes serviram para detetar e resolver problemas antes de se passar à fase de desenvolvimento. Para além disso, foi possível receber *feedback* com algumas ideias alternativas, para melhor elaboração da aplicação.

Modelo de dados

Para o desenvolvimento da aplicação é essencial a construção de uma base de dados que represente e guarde todos os conteúdos existentes neste projeto. Assim sendo, foi importante elaborar no primeiro semestre um esboço do modelo planeado que nos serviu para uma futura implementação do projeto.

4.3.2 Segundo Semestre

A figura 18 representa o diagrama de Gantt planeado para o segundo semestre.

Preview

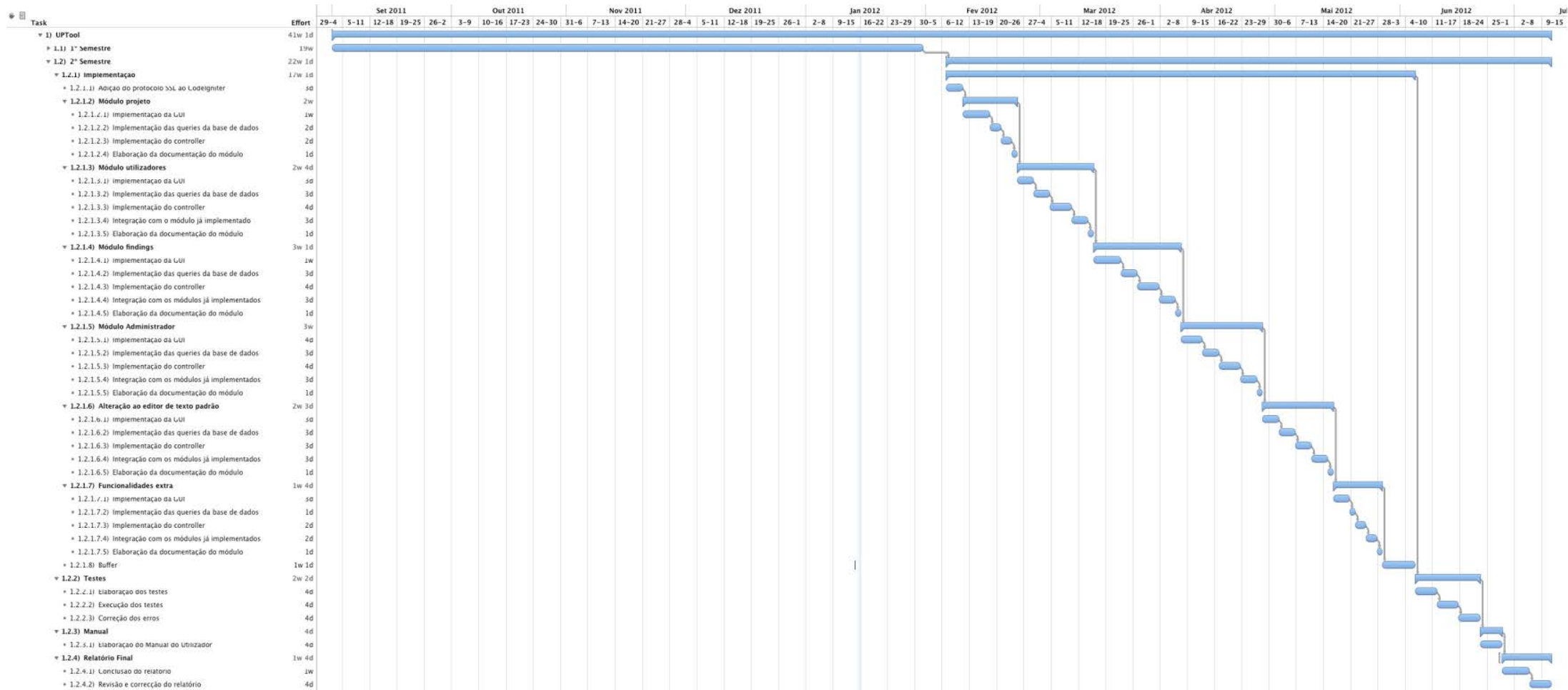


Figura 18 - Previsão inicial das atividades para segundo semestre

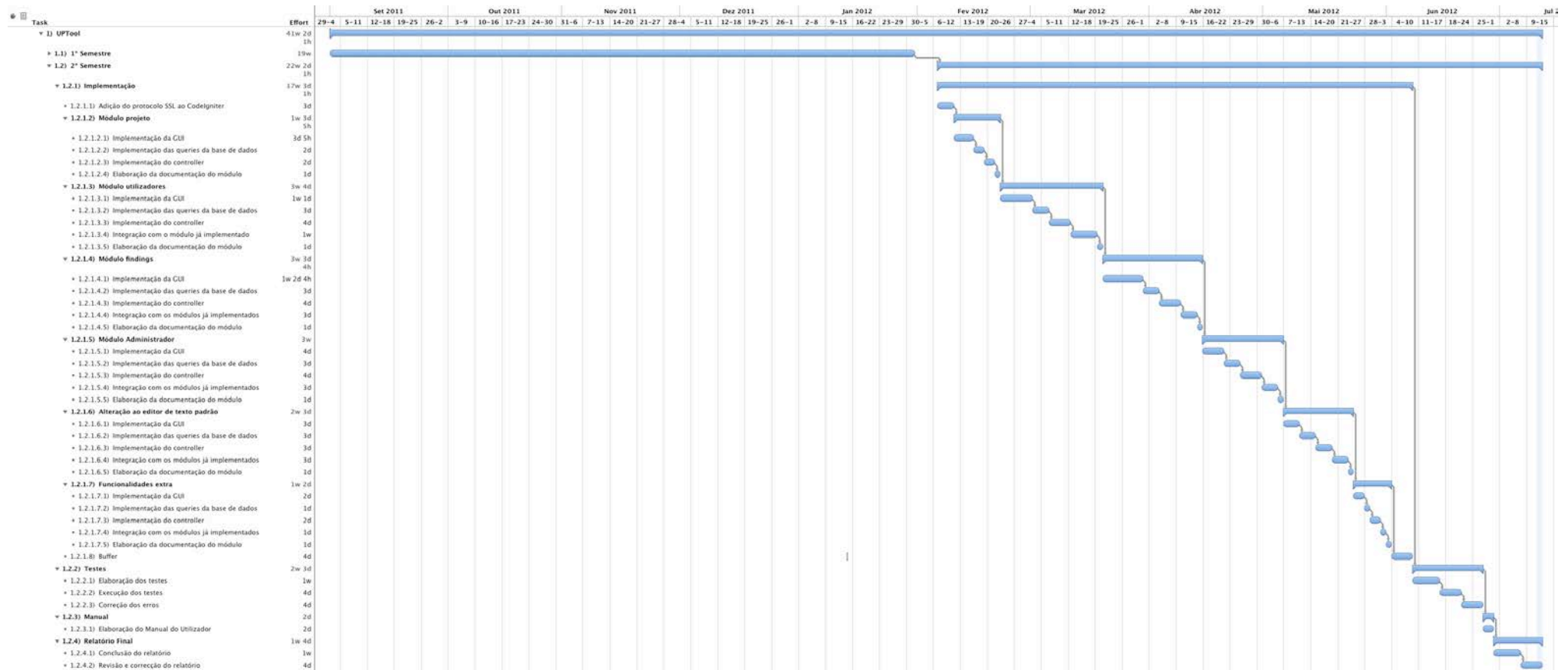


Figura 19- Diagrama de Gantt final das atividades do segundo semestre

Este diagrama corresponde a todo o segundo semestre desde o dia 8 de Fevereiro de 2012 a 8 Julho de 2012 em regime de *full-time*. Neste semestre foi feita a implementação de todos os módulos essenciais ao funcionamento da aplicação e respetiva documentação. Para além disso foram testes de forma a encontrar e corrigir erros existentes e por fim foi elaborado o documento final de estágio.

Segue uma breve explicação das tarefas que foram executadas no segundo semestre.

Implementação

Nesta tarefa foram implementados os diversos módulos delineados para o projeto e respetiva documentação.

Testes

Foram realizados testes funcionais e de performance para garantir o correto funcionamento da aplicação.

Manual do utilizador

Foi também elaborado o manual do utilizador com as funcionalidades mais importantes da aplicação.

Documentação final

Por fim foi elaborada a documentação final do projeto (Relatório Final de Estágio).

Capítulo 5

Projeto Preview

Neste capítulo descrevem-se as funcionalidades e arquitetura que definem a aplicação Preview.

5.1 Caracterização da aplicação

Tal como foi referido anteriormente, este projeto visa a construção de uma aplicação de apoio ao desenvolvimento de avaliações de usabilidade (*expert reviews*). A aplicação será *web-based* com o objetivo de ser mais interoperável.

Esta ferramenta pretende ser imprescindível em muitos dos projetos da Tangível, permitindo a criação e gestão de *findings* por parte dos especialistas de usabilidade nos seus projetos. Será possível acompanhar toda a evolução dos conteúdos do projeto através de um sistema de controlo de versões. E, não menos importante, a ferramenta servirá de ponte de comunicação entre os especialistas de usabilidade e os seus clientes (que normalmente estão fisicamente distantes).

Por último, é pretendido que a ferramenta seja útil não só para os especialistas de usabilidade mas também para os seus clientes, podendo servir de *check-list* para uma fácil perceção dos problemas já corrigidos.

5.2 Requisitos

Na primeira fase do estágio foi realizado um levantamento de requisitos de forma a definir quais as funcionalidades que a aplicação deveria suportar.

5.2.1 Levantamento de requisitos

No início do levantamento de requisitos existiram três fatores que tiveram maior influência na sua elaboração, sendo eles:

- Troca ativa de ideias ao longo de várias reuniões com o orientador do projeto;
- Estado da arte, que possibilitou a adição de algumas funcionalidades interessantes em aplicações analisadas;
- Reunião com um cliente da empresa (que utilizou a versão anterior da aplicação) para recolher ideias na perspetiva do cliente.

5.2.2 Atores do projeto

Nesta fase foram analisados os diversos intervenientes no projeto e dividiram-se os utilizadores por categorias para uma melhor distribuição das suas permissões. Sendo assim, os intervenientes neste projeto são os seguintes:

Administrador geral: O administrador geral terá total poder sobre a aplicação, podendo gerir todos os projetos de todas as empresas, incluindo os seus utilizadores.

Administrador da empresa de usabilidade: O administrador da empresa de usabilidade será a pessoa com maior permissão a seguir ao administrador geral. Este poderá adicionar empresas clientes, técnicos de usabilidade e selecionar gestores de projetos de usabilidade para um determinado projeto.

Gestor de projeto da empresa de usabilidade: Os gestores de projetos das empresas de usabilidade terão permissões acrescidas nos projetos de usabilidade em que se encontrem com este estatuto. Poderão adicionar utilizadores e alterar definições de projeto.

Especialistas de usabilidade: Poderão adicionar *findings* ao projeto, consultar *findings* de outros projetos e discuti-los com os clientes.

Administrador da empresa cliente: O administrador da empresa cliente será a pessoa com maior permissão do lado da empresa cliente. Este poderá adicionar utilizadores da empresa cliente, alterar o estatuto de um utilizador num determinado projeto para gestor de projetos da empresa cliente e alterar o nome e logotipo da empresa.

Gestor de projeto da empresa cliente: Os gestores de projetos das empresas clientes terão apenas permissões acrescidas nos projetos de usabilidade em que tenham este estatuto. Poderão adicionar técnicos da empresa cliente.

Técnico da empresa cliente: Poderão comentar o material introduzido nos projetos a que pertencem e colocar *findings* como corrigidos na lista de *findings*.

5.2.3 Descrição geral do sistema

De seguida apresenta-se um resumo de algumas das funcionalidades gerais essenciais que serviram de guias ao desenvolvimento dos projetos de usabilidade:

Gestão de projetos, utilizadores e empresas (usabilidade e clientes): à semelhança de outro tipo de aplicações, também nesta existirá um módulo de administração, onde se pode gerir os vários recursos do sistema, nomeadamente, empresas, utilizadores e projetos. Este módulo permitirá adicionar e editar utilizadores (da empresa de usabilidade e da empresa cliente), adicionar e remover empresas e criar e editar projetos. Será ainda possível alterar alguns parâmetros relacionados com a aplicação mas relativos à empresa como por exemplo nome da empresa, logotipo, etc.

Gestão de relatórios de *expert review*: os utilizadores poderão adicionar e editar relatórios de usabilidade referentes a cada projeto. Será ainda possível exportar estes relatórios segundo um *template* definido pela empresa de usabilidade.

Comentários: para todos os elementos envolvidos num projeto será possível comentar o trabalho realizado, ao longo do tempo em que o projeto se está a desenvolver. Esta funcionalidade é responsável pela criação de uma ponte de comunicação entre os clientes e os técnicos de usabilidade. Sendo assim, poderão ser reajustadas as recomendações indicadas pelos especialistas de usabilidade de acordo com as necessidades e limitações do cliente.

Controlo de versões: esta funcionalidade permite guardar um histórico das várias versões de um recurso, sendo possível navegar pelas várias versões e visualizar o progresso feito em cada uma. Apenas os técnicos de usabilidade poderão criar e visualizar um esboço para discussão, que mais tarde poderá ser ou não disponibilizado para os clientes da aplicação. Desta forma tanto clientes como técnicos de usabilidade poderão visualizar a qualquer momento do projeto a sua evolução, para facilitar a compreensão das decisões tomadas.

Para um maior detalhe dos requisitos analisados, o anexo A contém uma lista detalhada de cada um destes requisitos.

5.2.4 Prioridades dos requisitos

Para estabelecer a prioridade dos requisitos, foram adotadas as denominações “essencial”, “importante” e “desejável”.

Essencial é o requisito sem o qual o sistema não entra em funcionamento. Requisitos essenciais são requisitos imprescindíveis que têm que ser implementados impreterivelmente.

Importante é o requisito sem o qual o sistema entra em funcionamento, mas de forma não satisfatória. Requisitos importantes devem ser implementados, mas, se não forem, o sistema poderá, mesmo assim, ser implantado e usado.

Desejável é o requisito que não compromete as funcionalidades básicas do sistema, isto é, o sistema funciona de forma satisfatória sem ele. Requisitos desejáveis podem ser deixados para versões posteriores do sistema, caso não haja tempo útil para implementá-los na versão que está a ser especificada.

5.3 Arquitetura

Nesta secção é apresentada a arquitetura implementada para o sistema Preview. Esta é uma aplicação Web com proteções para eventuais falhas de segurança.

5.3.1 Arquitetura Técnica

O projeto desenvolvido será utilizado apenas pela Tangível e seus clientes, não necessitando de *Load Balancers* e replicadores de conteúdos da base de dados, estando esta arquitetura representada na figura 20.

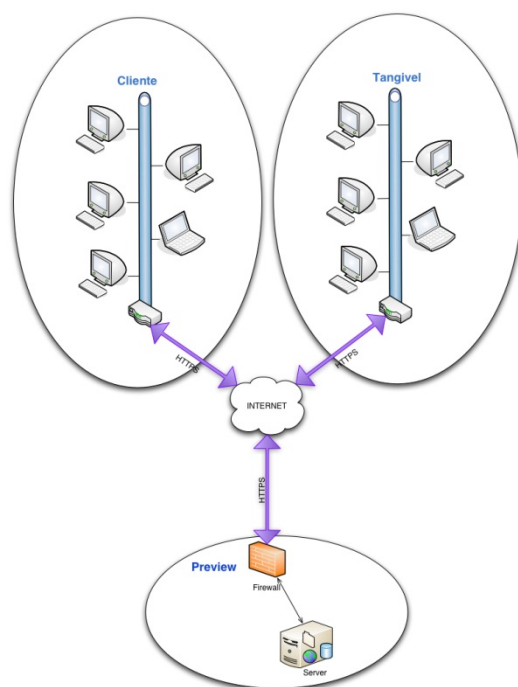


Figura 20 - Arquitetura inicial da aplicação

Para garantir a segurança dos conteúdos introduzidos na aplicação, as ligações à Internet foram efetuadas através de HTTPS (Hypertext Transfer Protocol Secure).

5.3.2 Arquitetura Funcional

O sistema comunica entre as diversas camadas a partir do modelo *Model-View-Controller* (MVC). Este *design pattern* é bastante comum nas aplicações Web, sendo o seu principal objetivo separar os dados da apresentação. Desta forma, a apresentação dos conteúdos da

aplicação é independente da manipulação dos dados e vice-versa. Assim sendo, podem ser alterados separadamente sem alterações nos restantes modelos.

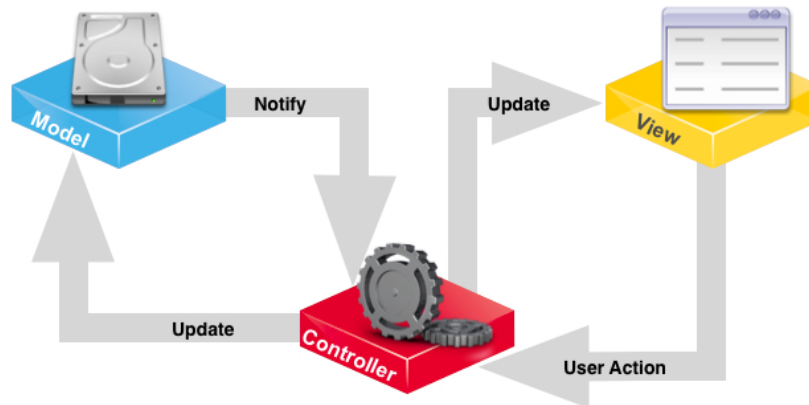


Figura 21 - Representação do MVC

O MVC é constituído por:

- **Model** - Representa a informação e é responsável pela comunicação com a base de dados.
- **View** - Apresenta a informação para que o utilizador consiga compreender e interagir com o conteúdo.
- **Controller** - Responsável pela receção de dados a partir do *view* e resposta a partir do *model*.

Para uma melhor perceção das classes existentes foi redigido o anexo B. Este contém as classes principais da aplicação.

5.3.3 Arquitetura Tecnológica

As tecnologias utilizadas no projeto foram escolhidas de forma a suportar melhor a arquitetura planeada. Foram escolhidas apenas tecnologias livres para diminuir ao máximo os custos da aplicação. Sendo assim todas as tecnologias usadas na criação desta aplicação possuem licenças livres.

Para melhor perceção das camadas existentes na aplicação e suas tecnologias, a figura 22 representa a arquitetura tecnológica da aplicação.

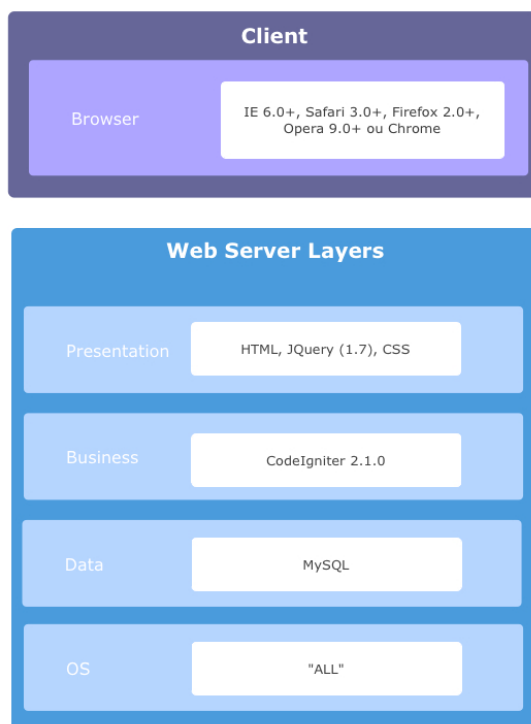


Figura 22 - Arquitetura tecnológica do sistema

Em seguida são dadas a conhecer as razões que levaram à escolha das tecnologias para cada camada da aplicação.

5.3.3.1 Modelo de dados - MySQL

Relativamente ao modelo de dados optou-se pelo MySQL em detrimento do Oracle, MonetDB, PostgreSQL, entre outros. É o motor de base de dados *open-source* mais popular devido ao seu alto desempenho, à alta confiabilidade e à facilidade de utilização. É uma base de dados bastante utilizada em organizações em grande crescimento como o caso da Google, Adobe, Facebook, entre outras. [1]

O MySQL é bastante interoperável podendo ser executado em mais de 20 plataformas incluindo as mais conhecidas, nomeadamente Linux, Windows, Mac OS X e Solaris. Este aspeto é muito importante pois possibilita a instalação na maior parte dos servidores existentes na Internet. Visto que a maior parte dos servidores na Internet são máquinas com o sistema operativo Linux, é relevante funcionar nesse suporte.

Por outro lado possui ferramentas de apoio, nomeadamente o MySQL Workbench que permite não só a modelação e *design* da base de dados mas também a manutenção da própria base de dados.

De realçar também que a *framework* CodeIgniter possui um vasto suporte para este sistema de base de dados, sendo uma mais-valia para o projeto.

5.3.3.2 Camada de negócio – CodeIgniter

5.3.3.2.1 Linguagem

Das linguagens existentes no desenvolvimento de aplicações Web, apenas foram analisadas o Java, Ruby On Rails e PHP por serem as mais usadas na Internet.

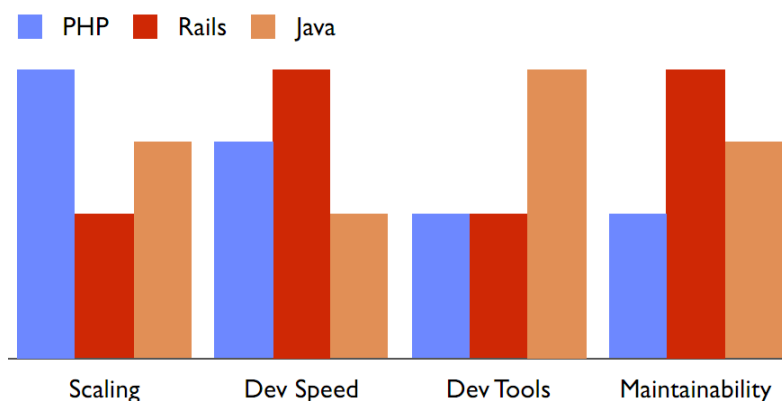


Figura 23 - Comparação entre PHP, Rails e Java segundo escalabilidade, tempo de desenvolvimento, ferramentas de desenvolvimento e manutenção. [2, p. 7]

Segundo uma comparação efetuada por Tim Bray, figura 23, diretor de Tecnologias Web da Sun Microsystems, relativamente a aspetos considerados essenciais no desenvolvimento de aplicações Web (escalabilidade, tempo de desenvolvimento, ferramentas de desenvolvimento e manutenção), o PHP revela-se uma boa escolha, principalmente em aspetos considerados essenciais ao desenvolvimento deste estágio que são a escalabilidade e o rápido desenvolvimento da aplicação.

5.3.3.2.2 Frameworks PHP

Para colmatar as pequenas falhas da linguagem PHP foi estudada a possibilidade de se usar uma *framework* PHP para diminuir o tempo de desenvolvimento da aplicação e aumentar o número de bibliotecas disponíveis ao serviço do *developer* (lacunas existentes na comparação de Tim Bray representada na figura 23).

Das inúmeras *frameworks* PHP existentes foram escolhidas a Zend, Yii, Kohana e CodeIgniter por serem bastante conhecidas e conceituadas por *developers* na Internet [2] e por o estagiário já possuir algum conhecimento *a priori* acerca das mesmas.

Não existe uma *framework* PHP ideal, ambas têm prós e contras. Todas elas têm bom suporte para diversas bases de dados e possuem bons tutoriais.

No entanto, Kohana possui uma pequena desvantagem, em algumas *minor releases* (versão 3.1 para a 3.2 por exemplo) o código precisa de ser refeito, sendo necessário a sua adaptação. O Yii

revelou-se uma excelente ferramenta, mas possui poucos tutoriais, por ser uma ferramenta relativamente nova. Por sua vez, o Zend disponibiliza uma documentação bastante pormenorizada mas complexa, o que se traduz numa curva de aprendizagem maior em relação às restantes *frameworks*.

Por outro lado, CodeIgniter apresenta uma documentação e uma comunidade excelente, muito acima das outras *frameworks*, o que diminui bastante a curva de aprendizagem e permite esclarecer qualquer dúvida ou problema existente através de um fórum bastante ativo. Foram estes fatores que levaram à escolha desta *framework* face às outras.

Para além disso a empresa já tinha utilizado esta tecnologia noutros projetos, tendo sido esta sugerida para o desenvolvimento da aplicação.

5.3.3.2.3 Framework CodeIgniter

Esta *framework* usa o *design pattern* MVC previsto para a realização do projeto. A figura 24 apresenta o diagrama de fluxo para melhor compreensão desta aplicação:

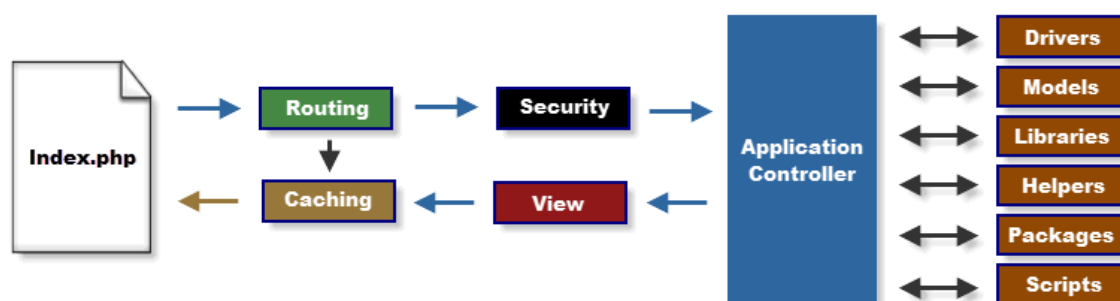


Figura 24 - Diagrama de fluxo da *framework* CodeIgniter [3]

1. O index.php serve como controlador frontal, sendo o responsável por inicializar a base de recursos para executar o CodeIgniter.
2. O Router examina o pedido HTTP ou HTTPS para determinar o que deve fazer com esse pedido.
3. Se o pedido se encontrar em cache é enviado diretamente ao *browser* cliente sem necessitar de entrar no *Controller*.
4. Antes do pedido HTTP ou HTTPS ser processado pelo *Controller*, este é filtrado pelo *Security*.
5. O *Controller* carrega e interage com todos os modelos necessários para processar um determinado pedido.
6. O *View* faz *rendering* à resposta do pedido efetuado pelo *browser* cliente. Em seguida é guardado em cache para ser utilizado em pedidos subsequentes e enviado por fim ao cliente.

5.3.3.3 Camada de apresentação

Nesta secção são apresentadas algumas ferramentas interessantes utilizadas na aplicação para melhorar a experiência do utilizador na aplicação.

jQuery

O jQuery é uma biblioteca JavaScript que permite uma maior simplicidade no desenvolvimento de páginas Web.

Esta ferramenta foi utilizada para produzir uma maior interação com o interface Web.

DataTables

O plugin do jQuery DataTables serviu para adicionar uma maior interação com as tabelas HTML.

Permite a personalização de filtros, pesquisas, ordenamento de colunas entre outras funcionalidades.

AJAX

Esta tecnologia está presente na maior parte do *site* pois é muito importante no conceito de Web 2.0, tornando as páginas mais interativas e atualizando os seus conteúdos sem necessidade de sair da página.

JSON

O JSON é muito usado para transmissão de dados sob a forma de estruturas pela rede. Esta deriva do JavaScript e serve para representação de objetos (estruturas de dados e *arrays* associativos).

Este formato tem duas funções:

- Guardar a informação para seguidamente transmiti-la de forma dinâmica para as tabelas das páginas;
- Guardar a estrutura de dados de sucesso ou erro dos formulários que posteriormente é enviada por AJAX.

5.3.4 Arquitetura do *site*

Para uma melhor compreensão da estrutura da aplicação Web foi desenvolvido um diagrama que mostra a hierarquia das páginas Web da aplicação. A figura 25 representa essa mesma estrutura e os utilizadores que têm permissões para acederem a essas páginas.

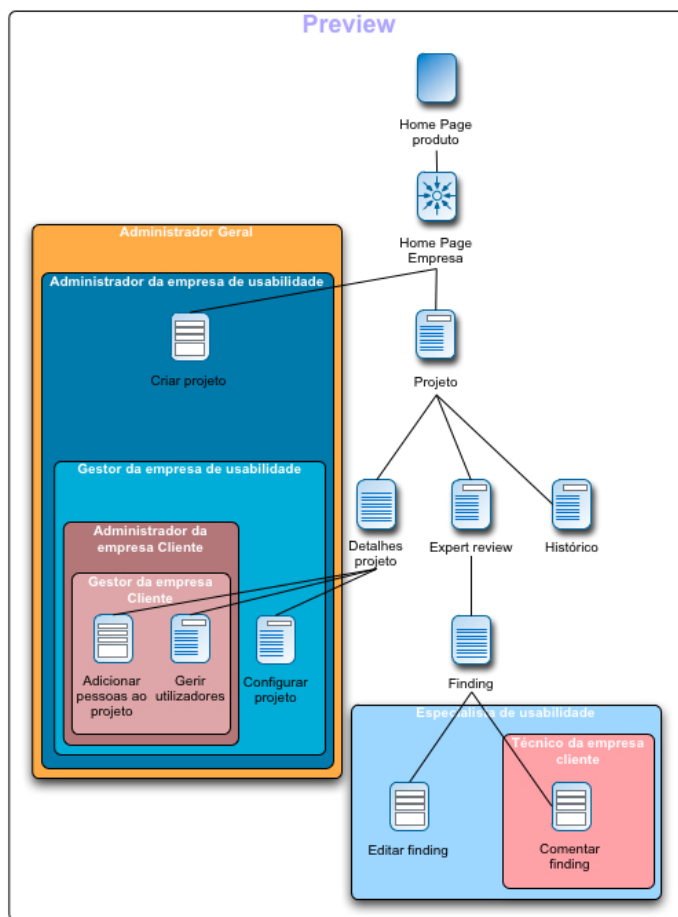


Figura 25 - Mapa das páginas da aplicação

A figura 25 contém todas as páginas necessárias para o funcionamento da aplicação e os utilizadores restritos de cada uma delas. Para um maior detalhe dos conteúdos e ligações existentes em cada página, o anexo C possui *mockups* que permitem uma melhor perceção da constituição inicialmente prevista das várias páginas da aplicação Web.

Seguem ainda alguns *screens* da aplicação pertencentes a várias páginas da aplicação:

- Lista de findings

Project: teste

Add New Finding

Expert reviews

Last versions All versions Search:

Id	Name	Version	Data	Severity
1	Header.multiparagrafo	0.2	2012-06-15 16:02:34	2
2	Header.multiparagrafo	0.1	2012-06-15 11:28:06	2
3	thead	0.1	2012-06-15 11:29:04	2

Showing 1 to 3 of 3 entries (filtered from 4 total entries)

Categories

Search:

Name

Loading...

Showing 0 to 0 of 0 entries

Figura 26 - Lista de findings de um projeto

A figura 26 representa a lista de *findings* existentes num projeto e os filtros criados para uma melhor organizar os conteúdos.

- Gestão de utilizadores

Company: Tangível Manage company users Logout

Manage users

Add a user

Tangível

Administrator	Jorge Figueira
---------------	----------------

TERMS OF USE LEGAL PRIVACY POLICY EUROPA - ENGLISH FAQ SUPPORT CONTACTS

© 2012 Tangível group. All rights reserved.

Figura 27 - Página de gestão de utilizadores da empresa

- **Página de uma versão de um *finding***

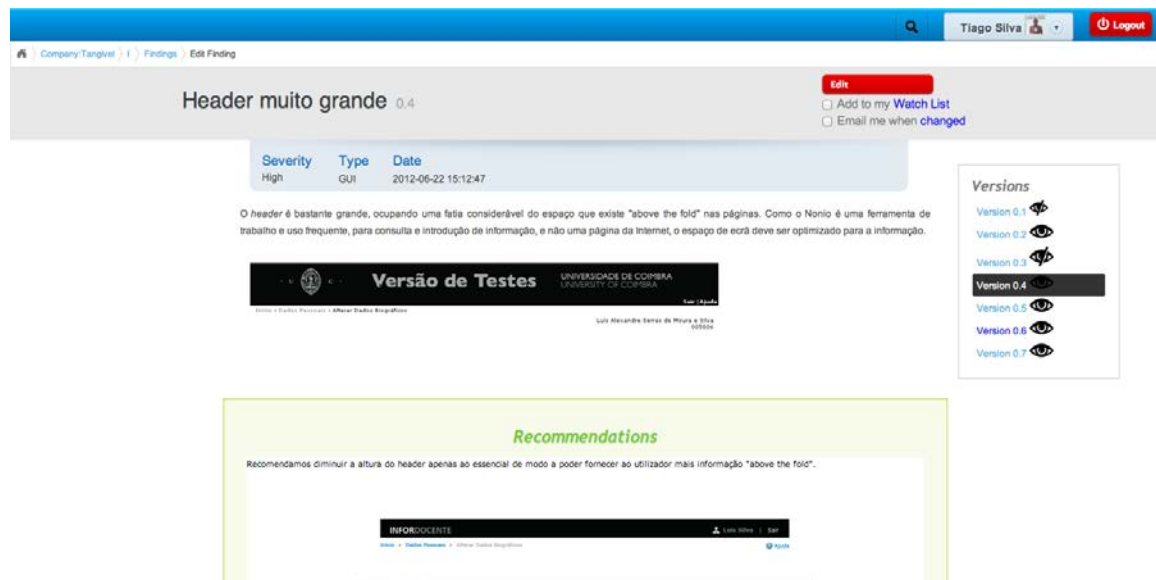


Figura 28 - Página de um *finding*

Para uma melhor percepção da aplicação o anexo D contém o manual do utilizador que possui as páginas da aplicação com maior detalhe.

5.4 Funcionalidades

Nesta secção são apresentadas algumas funcionalidades interessantes introduzidas na aplicação para melhorar a experiência do utilizador.

5.4.1 Filtros interativos nas tabelas

Esta funcionalidade foi feita a partir do *JavaScript* e tem como objetivo filtrar a informação de uma forma mais intuitiva. Caso o utilizador se encontre na página inicial da empresa, poderá seleccionar vários projetos (linhas da tabela “My Projects”) sendo a informação da tabela (“Recent activities”) automaticamente filtrada contendo apenas a informação relativa aos projetos seleccionados.

O mesmo acontece quando o utilizador se encontra na página de *findings*, estes poderão pertencer ou não a categorias (tabela “Categories” na página de *findings*) sendo possível seleccionar categorias existentes e automaticamente filtrar os *findings* que pertençam a essas categorias.

5.4.2 Formulários sem *refresh*

Para tentar usar ao máximo o conceito Web 2.0 foram feitos formulários em AJAX. Esta tecnologia possibilita a sua criação sem a necessidade de carregar novamente a página com os conteúdos introduzidos.

Foram efetuadas validações nos campos dos formulários de duas formas:

- Por *JavaScript* no *browser*, não necessitando recorrer ao servidor para verificações simples (como por exemplo, verificar se o campo apenas contém texto ou números ou se passou o limite máximo de caracteres);
- Por verificação dos dados submetidos no servidor através da tecnologia AJAX, não necessitando assim de voltar a carregar a página caso seja necessário corrigir algum campo do formulário.

5.4.3 Salvar automaticamente

Esta funcionalidade surgiu após a percepção de que à medida que se aumenta o nível de personalização dos conteúdos dos *findings* na aplicação surge um risco, cada vez maior, de perda dos conteúdos, criando algum incómodo aos especialistas de usabilidade na introdução de conteúdos na aplicação.

Sendo assim, à medida que são introduzidos conteúdos nos *findings*, essa informação é guardada automaticamente na base de dados do *browser* que o utilizador está a trabalhar no momento.

Esta informação guardada automaticamente é eliminada de duas formas:

- O utilizador submeta a informação na aplicação e seja bem-sucedido.
- A informação expire (validade de 3 dias).

5.4.4 Gestão de imagens *findings*

Para uma fácil personalização dos *findings*, foi criada uma estrutura para abranger as necessidades existentes relativamente à gestão de imagens nos *findings*. Para isso foram criadas hierarquias de pastas desde o projeto a *findings* e versões, para melhor organização dos conteúdos introduzidos. Caso um utilizador pretenda introduzir uma ou várias imagens num *finding* poderá em primeiro lugar introduzir as imagens numa pasta temporária existente e em seguida, caso o utilizador pretenda submeter os conteúdos, as imagens são copiadas para a pasta da versão do *finding* e eliminadas da pasta temporária.

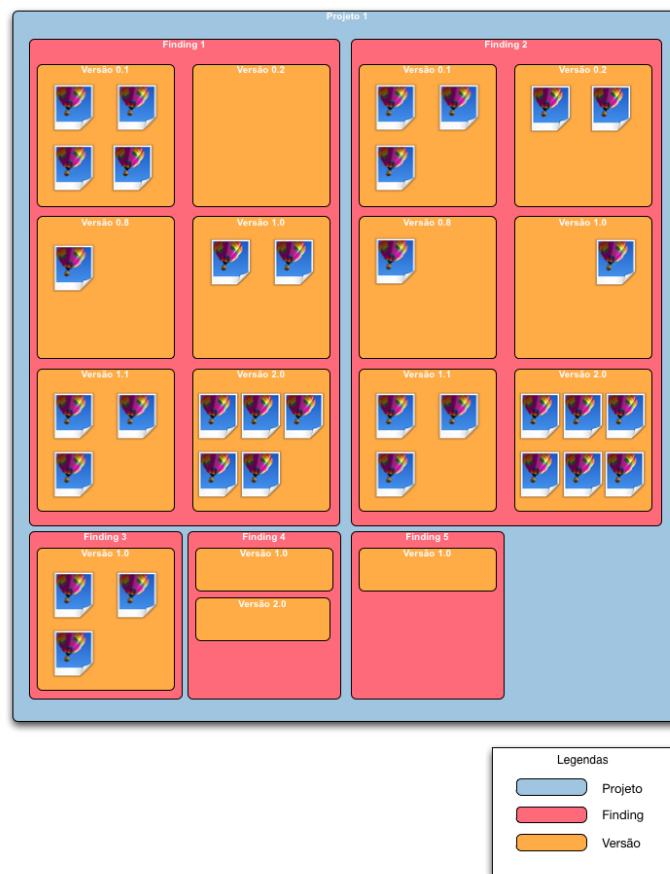


Figura 29 - Estrutura das imagens na aplicação

De salientar ainda, que caso um utilizador se encontre numa versão de um *finding* e pretenda criar outra versão com uma imagem utilizada numa versão anterior esta não será copiada para a nova versão do *finding*. Este facto é bastante importante para minimizar a replicação de imagens introduzidas.

5.4.5 Função de Encriptação

Na fase de implementação do sistema de *logins*, as palavras-chave armazenadas na base de dados foram encriptadas. Esta encriptação possibilita em caso de exposição da base de dados a uma mais-valia na ocultação das palavras-chave. Inicialmente foi pensou-se em usar SHA-1, pois MD5 já sem encontra quebrado, mas após uma pesquisa na Internet, destetou-se que ambos MD5 como SHA-1 já se encontram quebrados.

Foi então escolhida a utilização da família SHA-2, uma vez que até à data ainda não foi quebrada. Desta família são integradas as funções SHA-224, SHA-256, SHA-384, SHA-512, cujo nome de cada representa o número de bits que produz. Sendo assim, optou-se pela função de *hash* de 256 bits, denominada SHA-256, por ainda não ter sido quebrada e por ocupar menos

espaço na base de dados. De referir ainda, que esta função necessita de 64 caracteres hexadecimais daí a utilização de VARCHAR(64) para o seu armazenamento.

Sendo assim, mesmo em caso de comprometimento da base de dados este mecanismo mantém as palavras-chave encriptadas garantido um maior grau de confiança aos utilizadores da aplicação.

Capítulo 6

Performance e validação

A aplicação possuirá bastante informação que necessita de ser apresentada num intervalo de tempo minimamente aceitável. Foram, então, consideradas questões acerca de performance no projeto.

6.1 Performance Base de dados

A partir do comando “Explain” da base de dados foram encontradas possíveis otimizações nas *queries* existentes na aplicação.

Foram feitas análises de conjuntos de índices a introduzir com o objetivo de melhorar a velocidade de pesquisa da base de dados.

Em seguida é apresentado um exemplo de um índice adicionado para melhorar a performance da base de dados:

- Na tabela “History” foi adicionado o índice conjunto de **Project_id** e **date**. Com esta adição obtivemos uma melhoria na ordem dos 28% na *querie* – “ SELECT History.date, History.User_username, History.type, History.action FROM History WHERE History.Project_id = 1 ORDER BY History.date;”
- Na tabela “Finding” foi adicionado o índice conjunto de **Project_id** e **visibility**. Com esta adição obtivemos uma melhoria na ordem dos 16% na *querie* – “ SELECT Finding.id, Finding.name, Finding.version, Finding.date, Finding.severity, Finding.type, Label_has_Finding.Label_id FROM Finding LEFT JOIN Label_has_Finding ON Finding.id = Label_has_Finding.Finding_id WHERE Finding.Project_id = 1 AND Finding.visible LIKE 'Visible';”

No anexo E, Modelo de dados, no sector das otimizações é possível visualizar com mais detalhe os testes e otimizações efetuadas às tabelas.

6.2 Validação

Para a validação da aplicação foram efetuados um conjunto de testes para garantir o correto funcionamento da aplicação.

6.2.1 Testes funcionais

Nesta fase foram planeados testes funcionais para a validação do *software*, tendo como principal objetivo garantir a qualidade do produto final.

Foi então produzido um guião tendo em conta as funcionalidades implementadas com o intuito de garantir a qualidade do produto final. Para execução dos testes foi escolhido um utilizador que não esteve envolvido na fase de desenvolvimento.

Para um maior detalhe dos testes efetuados foi elaborado o anexo F, denominado Testes funcionais.

6.2.2 Testes de compatibilidade de browsers

Um fator muito importante tido em conta na escolha dos *plugins* e tecnologias a utilizar na aplicação foi a compatibilidade.

A funcionalidade de salvar automaticamente na base de dados local do *browser* é a que possui menos suporte em relação às outras, tendo compatibilidade com: Chrome4+, Firefox2+, Internet Explorer6+, Opera10.50+ e Safari4+.

Mesmo assim, tem um grande suporte de *browsers* o que torna a aplicação bastante interoperável. Assim, o cliente não necessitará de possuir um *browser* específico para interagir com a aplicação.

Capítulo 7

Conclusões

Este projeto teve como objetivo a construção de uma aplicação que auxiliasse os projetos de usabilidade, encurtando a distância entre os seus *developers* e clientes. Dando a oportunidade dos clientes acompanharem todas as fases do projeto.

A aplicação visava colmatar lacunas que se verificavam nas ferramentas na área de usabilidade. Permitiu o desenvolvimento de uma ferramenta nova, específica para o desenvolvimento de projetos de usabilidade contendo funcionalidades consideradas essenciais para a gestão de projetos *online*.

Este estágio foi uma oportunidade de pôr em prática todos os conhecimentos adquiridos no curso de Engenharia Informática e contactar com a realidade do mundo profissional.

A nível pessoal, permitiu ganhar conhecimentos na área da usabilidade que até à data, eram mínimos. Além disso, foi uma oportunidade de expandir conhecimentos no mundo Web nomeadamente JavaScript e CSS nunca explorados no curso.

7.1 Trabalho futuro

Apesar de se ter construído a aplicação que se pretendia, ainda existe espaço para melhorias e adição de funcionalidades. Estas melhorias terão sempre o objetivo de facilitar a introdução e realização de relatórios dos especialistas de usabilidade e melhorar a comunicação entre os especialistas e os seus clientes.

Numa fase futura disponibilizar-se-á a aplicação a mais empresas de usabilidade e seus clientes. Nessa altura será necessária maior escalabilidade por parte da aplicação Web e torna-se essencial a utilização de *Load Balancers* para distribuir os pedidos e os replicadores de informação dos conteúdos da base de dados. A arquitetura deste cenário pode ser vista na figura 30.

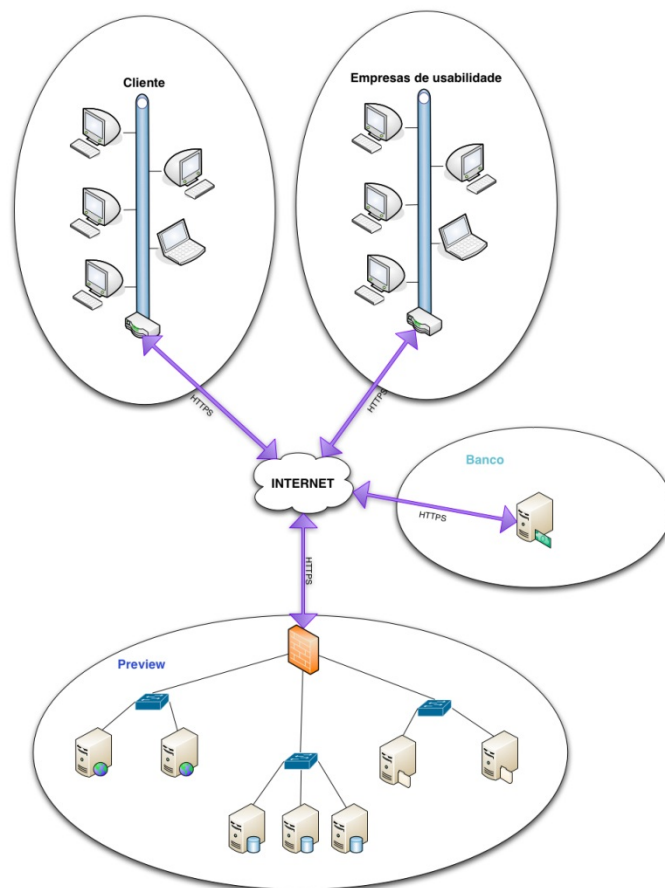


Figura 30 - Arquitetura final da aplicação Preview

Esta aplicação servirá então de ponte de comunicação entre as diversas empresas de usabilidade e os seus clientes.

Capítulo 8

Lista de Citações

- [1] Goplan, “Contact details,” [Online]. Available: <http://goplanapp.com/home/contact>. [Acedido em 23 Janeiro 2012].
- [2] MySQL, “Why MySQL?,” [Online]. Available: <http://www.mysql.com/why-mysql/>. [Acedido em 2 Janeiro 2012].
- [3] “Top 10 PHP frameworks,” [Online]. Available: <http://www.phpframeworks.com/top-10-php-frameworks/>. [Acedido em 2 Janeiro 2012].
- [4] T. Bray, “Issues in Web,” [Online]. Available: <http://www.tbray.org/talks/php.de.pdf>. [Acedido em 10 Janeiro 2012].
- [5] EllisLab, Inc., “Application Flow Chart,” [Online]. Available: http://codeigniter.com/user_guide/overview/appflow.html. [Acedido em 12 Janeiro 2012].

Capítulo 9

Bibliografia

Code Igniter User Guide, http://codeigniter.com/user_guide/, consultado a 20-09-2011.

GoPlan, <http://goplanapp.com/>, consultado a 20-09-2011.

Preview, <http://preview.tangivel.com/>, consultado a 20-09-2011.

Redmine, <http://www.redmine.org/>, consultado a 20-09-2011.

Basecamp, <http://basecamphq.com>, consultado a 21-09-2011.

Tanenbaum, Andrew S., Redes de Computadores, 4ª Edição, Editora Campus, 2003.

37signals, <http://highrisehq.com/>, consultado a 21-09-2011

37signals campfire, <http://campfirenow.com/>, consultado a 21-09-2011

37signals, <http://37signals.com/about>, consultado a 5-10-2011

Basics Codeigniter, <http://net.tutsplus.com/tutorials/php/codeigniter-basics/>, consultado a 6-10-2011

DataTables, <http://www.datatables.net>, consultado a 28-11-2011

MySQL, <http://www.mysql.com>, consultado a 6-12-2011

jQuery, <http://www.jquery.com/>, consultado a 8-12-2011

CodeIgniter from Scratch, <http://net.tutsplus.com/sessions/codeigniter-from-scratch/>, consultado a 20-12-2011

What Is Web 2.0, <http://oreilly.com/web2/archive/what-is-web-20.html>, consultado a 20-01-2012

jQuery Foundation and the jQuery UI Team, <http://jqueryui.com/demos/datepicker/>, consultado a 28-01-2012

TeamBox, <http://www.teambox.com/>, consultado a 28-02-2012.

JSON, <http://json.org/>, consultado a 16-03-2012

XML, <http://www.w3.org/XML/>, consultado a 16-03-2012

Hash algorithm, <http://www.kellermansoftware.com/t-articlestrongesthash.aspx>, consultado a 2-05-2012

Anexos

- [A] Análise de Requisitos**
- [B] Diagrama de classes**
- [C] Mockups**
- [D] Manual do utilizador**
- [E] Modelo de Dados**
- [F] Testes funcionais**