

Universidade Fernando Pessoa

Curso de engenharia informática



Documento de design arquitetural/detalhado

Disciplina: Engenharia de Software

Realizado por: Luís Aguiar nº 36452
David Capela nº 37045
Diogo Costa nº 34138

Document status sheet			
Document title: Projeto ESOF Architectural/Design Document			
Issue	Revision	Date	Reason for change
1	0	Novembro 2019	Rascunho

Document Amendment sheet

Issue	Page	Comment

Introdução

Este documento descreve o projeto de Arquitetura e o projeto Detalhado da aplicação. A aplicação deve permitir a um aluno marcar um atendimento com um explicador. A aplicação deve cobrir as necessidades do utilizador “aluno”, “explicador” e “gestor”

Objetivos

Este documento segue as recomendações ESA-PSS-05-02 (Requisitos Utilizador) e ESA-PSS-05-03 (Requisitos Software), para descrever os requisitos de software e utilizador do projeto desenvolvido. A aplicação necessita de um serviço de dados, cuja existência se assume e cujos requisitos serão também descritos.

Âmbito da aplicação

A aplicação é desenvolvida no âmbito da parte Prática da cadeira de Engenharia de Software. E propõe permitir a um aluno de uma determinada universidade marcar um atendimento com um explicador/professor disponível.

Referências

ESA PSS-05-02, Guide to the user requirements definition phase, Issue 1 Revision 1, Março 1995.

ESA PSS-05-03, Guide to the software requirements definition phase, Issue 1 Revision 1, Março 1995.

Descrição geral

A aplicação deve permitir a um utilizador do tipo aluno escolher segundo uma lista de explicadores (retornados conforme um conjunto de filtros que o aluno utilize), marcar um atendimento com este num horário em que este tenha disponibilidade, isto implica, ter esse dia e essa hora na sua disponibilidade e não ter um outro atendimento no mesmo momento.

Análise do produto

Este tipo de aplicações já existe, porém o seu objetivo principal é fornecer aos docentes uma avaliação para a disciplina.

Perfil dos utilizadores

A aplicação está desenhada para 3 perfis distintos de utilizadores:

Aluno: pode marcar atendimentos, procurar por explicadores e consultar disponibilidade de um explicadores

Explicador: pode ver as suas marcações com os alunos e pode atualizar as suas informações (disponibilidade, cadeiras...)

Gestor: tem a função de gerir a aplicação e pode adicionar faculdade, cadeiras, cursos e gerir as permissões dos outros utilizadores

A aplicação deve permitir que cada um destes perfis possa usá-la de forma fácil, simples e agradável

Qualidade dos resultados

A informação disponibilizada pela aplicação deve ser exata (como a disponibilidade do utilizador, suas cadeiras...), válida e apresentada de forma agradável e fácil de entender e usar. O utilizador deve ser informado quando algum destes fatores pode comprometer a qualidade da informação.

Restrições gerais

Para além do critério do utilizador ter de fazer parte de uma universidade para utilizar a aplicação, alguns dos critérios de qualidade poderão depender de outros fatores: periodicidade de utilização de dados recentes do serviço de dados, existência de conectividade, tempo de bateria, e disponibilidade de utilização de largura de banda. Supõe-se que o serviço de dados tem uma disponibilidade próxima de 100%, e que a largura de banda disponível é suficiente para transferir os dados necessários.

Arquitetura da aplicação

a aplicação segue a seguinte arquitetura:

1º recebe o pedido

2º verifica a viabilidade do pedido

3º faz a query à base de dados

4º retorna a informação do pedido ao utilizador

Descrição dos módulos

Recebe o pedido – o utilizador insere um json com as informações e essas informações são recebidas pelo controller

Verifica a viabilidade do pedido – o controller após receber o json, converte-o e envia ao service para este verificar a sua viabilidade

Faz a query à base de dados – o service depois de verificar a viabilidade do pedido, faz uma query à base de dados tal como foi solicitado pelo pedido/ endpoint

Retorna a informação ao utilizador – o service retorna a informação ao controller e ele envia esta ao utilizador sob forma de json

Normas e boas práticas

Documentação

Toda a documentação, tanto o DIP, como o USR e o STP estão na pasta em conjunto com o código fonte do projeto

Nomes de variáveis, funções e métodos

Todas as variáveis, funções e métodos do projeto têm nomes relevantes e percetíveis, facilitando desta forma a leitura a todo e qualquer pessoa que pretenda ler o código fonte, as funções mais relevantes do projeto estão devidamente comentadas, explicando o que estas fazer, o que são os seus parâmetros e o que retornados

Comentários

Todo o projeto está bem documentado, tornando assim a leitura do código muito mais fácil a qualquer programador que a pretenda alterar/ melhorar