Icon

Description automatically generated with medium confidence

Unidade Curricular

Programação em Ambiente Web

Milestone 1

Docentes:

Fábio André Souto da Silva

Marco Filipe Vieira Gomes

Alunos:

Rui Alexandre Borba Vitorino (8190479)

Gilberto Jorge Da Mota Gomes (8210227)

Licenciatura em Segurança Informática de Redes de Computadores

Índice

Contents

[Introdução 5](#_Toc102637936)

[Enquadramento 5](#_Toc102637937)

[Funcionalidades Propostas 5](#_Toc102637938)

[Funcionalidades Implementadas 6](#_Toc102637939)

[Registo de Funcionários 6](#_Toc102637940)

[Gestão de clientes 7](#_Toc102637941)

[Histórico de operações 8](#_Toc102637942)

[Gestão de livros 8](#_Toc102637943)

[Conclusões 8](#_Toc102637944)

[Referências 8](#_Toc102637945)

Índice Figuras

# Introdução

# Enquadramento

No âmbito do desenvolvimento do trabalho prático da unidade curricular de Programação em Ambiente Web, onde o objetivo passa pela criação de uma aplicação web que irá simular uma livraria online.

Esta aplicação irá ter duas componentes, nomeadamente, *front-end* e *back-end*, tendo sido a componente do *back-end* (e respetivo *backoffice,* ou seja, componente *front-end* apenas disponível para questões de administração e gestão de objetos de negócio), desenvolvida no milestone1.

# Estrutura

De forma a seguir as boas práticas de desenvolvimento para aplicações que tenham como base a *framework express*, bem como o *template engine* utilizado, EJS, a seguinte estrutura foi concebida.

* app.js
* models/
* routes/
* controllers/
* public/
  + imagens/
  + javascripts/
  + stylesheets/
* views/
  + backoffice/
  + cliente/
  + index/
  + error/

Existem ainda outras diretorias, tais como:

* bin/
* certs/
* node\_modules/

## Explicação da estrutura

Existe uma particularidade da estrutura por nós utilizada que merece esclarecimento, nomeadamente, a existência de replicação de código nos diversos controladores (controllers backoffice, admin e employee), bem como nas vistas.

O processo de estruturação ocorreu durante a divisão de tarefas, ou seja, certo elemento do grupo iria desenvolver o backend/backoffice admin e o outro iria desenvolver o backend/backoffice employee, não existindo assim independência de determinada funcionalidade, comportamento e/ou implementação para o funcionamento geral da aplicação, de certa forma, sendo *quase* possível a comparação a Interfaces Java.

Obviamente esta decisão não é a ideal em termos de produção, ou na otimização e reutilização de código, mas após a avaliação da situação, a restruturação implicaria uma grande quantidade de trabalho, e requeria também alterações no workflow de desenvolvimento, levando-nos a manter a estrutura presente.

A restante estrutura segue os padrões de desenvolvimento da *framework* em uso, fazendo uso das funcionalidades disponíveis pela mesma, e pelo *template engine* EJS, tornando algumas páginas estáticas em páginas dinâmicas, consoante a informação recebida e/ou o utilizador autenticado.

# Funcionalidades Propostas

## Milestone1

Foram implementadas todas as funcionalidades propostas, existindo pequenos detalhes e comportamentos que posteriormente, no milestone2, irão sofrer alterações, devido aí, já estar presente a componente de negócio. No entanto estas funcionalidades “incompletas” já têm a lógica e respetiva implementação base desenvolvida, restando apenas “interligar” os componentes.

Sendo o grupo pertencente à Licenciatura de Segurança, Redes e Sistemas Informáticos, outras funcionalidades relativas à segurança da aplicação foram também implementadas.

### Registo de Funcionários

Através do portal de *backoffice* e após a autenticação do administrador ou funcionário, o conteúdo e consequente disponibilidade de funcionalidades são apresentadas dinamicamente.

#### Administrador

Sendo o administrador uma conta com total permissão, é permitido a criação, visualização, atualização e pesquisa de funcionários.

### Funcionário

O funcionário, após a sua autenticação, irá apenas ter acesso ao seu perfil, e capacidades limitadas de atualização das suas informações, não sendo permitida a criação, atualização, visualização e pesquisa de outros funcionários.

### Gestão de clientes

* + Criar nos dois, mas no admin ele pode gerir pontos
  + E possível visualizar e editar no admin
  + employee so pode ver e atualizar menos os pontos
  + Automatica atribuição de 10 pontos na criacao

### Gestão de livros

* + Gestao de livros **usados** a aguardar milestone2, implementação de portal de cliente, embora já esteja pensada a arquitetura de backend/backoffice
  + Criar, atualizar, remover em ambos

### Histórico de operações

* + Ta feito

## Outras implementações

Imagens, paginação, hashing passwords, tokens and cookies, search por certos campos, validação de username, email, isbn; validação de sessão/login/permissão; logout

https

## Milestone2

## Funcionalidades Implementadas

### Registo de Funcionários

### Gestão de clientes

### Histórico de operações

### Gestão de livros

# Conclusões

# Referências