|  |  |
| --- | --- |
|  |  |

**Licenciatura em Engenharia Informática**

Escola Superior de Tecnologia e Gestão

*Instituto Politécnico de Viana do Castelo*

Projeto 2

Desktop + Acesso à BD

Gestão de um ginásio

Alexandre Santos - 24585

Sofia Sousa - 23435

2022/2023

Sumário

[1. Introdução 3](#_Toc135495945)

[2. Modelo Casos de Uso 3](#_Toc135495946)

[3. Modelo Físico (Base de Dados) 4](#_Toc135495947)

[4. Acesso à base de dados 5](#_Toc135495948)

[5. Aplicação Desktop 7](#_Toc135495949)

[6. Conclusão 9](#_Toc135495950)

[7. Bibliografia 10](#_Toc135495951)

# Introdução

Este relatório servirá para demonstrar o ponto de situação atual do projeto. Nesta entrega, vamos incluir o módulo de acesso à base de dados *PostgreSQL*, integrado no módulo da aplicação em *Javafx*. Para modelar a interface gráfica, utilizamos a ferramenta *SceneBuilder* com *Javafx*.

# Modelo Casos de Uso

No modelo de casos de uso temos 4 atores:

* Administrador, que cria utilizadores de todo o tipo e tem acesso a todas as outras funcionalidades de gestão do sistema;
* Instrutor, que pode gerir aulas de grupo e planos de treino;
* Rececionista, que gere clientes, subscrições e pagamentos;
* Cliente, que consegue ver e editar a sua conta, consultar a sua subscrição, aulas em que se inscreveu ou aulas que irão acontecer para se inscrever e por fim os planos de treino que já teve e que tem atualmente.

Uma imagem com diagrama

Descrição gerada automaticamente

# Modelo Físico (Base de Dados)

Uma imagem com texto, captura de ecrã, diagrama

Descrição gerada automaticamenteA base de dados final, usada para armazenar toda a informação e relações entre entidades que são a base de funcionamento da aplicação, pode ser analisada na figura seguinte.

# 4. Acesso à base de dados

Para aceder à base de dados utilizamos as ferramentas *Hibernate* e *JPA*.

Começamos por criar um projeto vazio em *Maven*, onde criamos o ficheiro “*pom.xml*” que guarda informação sobre todas as dependências necessárias para o projeto, no nosso caso, é o *Hibernate* e os drivers para conexão com o *PostgreSQL*. De seguida criamos dentro da pasta “*resources*” o package “*META-INF*”, que deverá conter o ficheiro “*persistence.xml*”. Este ficheiro contém toda a informação relacionada com a conexão à base de dados, incluindo o nome da unidade do *persistence* que usamos para realizar todo o tipo de consultas e atualizações na base de dados.

Uma imagem com texto, captura de ecrã, software, Software de multimédia

Descrição gerada automaticamente

No acesso da base de dados, temos ainda três packages de grande importância.

O primeiro é o “*repository*”, onde são guardadas as entidades da base de dados, que correspondem às tabelas na BD. Em cada ficheiro são definidos os campos das tabelas, as relações e as restrições como chaves primárias, valores gerados automaticamente, etc.

De seguida, temos os ficheiros “*DAO*”, estes ficheiros vão ter os métodos principais para o *CRUD*, e as *queries* de consulta, inserção e atualização de dados.

Uma imagem com captura de ecrã, Tipo de letra, texto, design

Descrição gerada automaticamenteUma imagem com texto, captura de ecrã

Descrição gerada automaticamente Por fim, no *package* “*BLL*”, estão os ficheiros responsáveis pela lógica de negócio principal do projeto.

Uma imagem com texto, captura de ecrã, Tipo de letra

Descrição gerada automaticamente

# 5. Aplicação Desktop

A aplicação desktop, está a ser desenvolvida em Javafx com o SceneBuilder para construir as interfaces. Este projeto tem dois componentes essências:

* No package “*controller”* estão os ficheiros responsáveis porcontrolar cada ecrã. Estão neles os métodos para cada componente do ecrã como botões e tabelas, e permitem criar a lógica de ação para esses mesmos elementos.

Uma imagem com texto, captura de ecrã, Tipo de letra, design

Descrição gerada automaticamente

* No package “*resources”* encontram-se os ficheiros fxml que representam cada ecrã. Nestes ficheiros é onde fazemos o design da interface.

Uma imagem com texto, captura de ecrã, software

Descrição gerada automaticamente

Uma imagem com texto, captura de ecrã, software, Software de multimédia

Descrição gerada automaticamenteAté ao momento, temos desenvolvido o login onde o utilizador deve inserir o seu email e respetiva password para ter acesso ao seu *dashboard*. Depois temos parte do *dashboard* do Instrutor, onde já é possível ver tabelas com dados para aulas de grupo e planos de treino.

Uma imagem com texto, captura de ecrã, software, Ícone de computador

Descrição gerada automaticamente

# 6. Conclusão

Apesar de, o prazo final se estar a aproximar, sentimos que ainda somos capazes de finalizar o projeto Desktop e Web apesar de estarmos bastante atrasados devido às dificuldades que sentimos com o a realização do acesso à base de dados. De todos os componentes do projeto foi o que nos deu mais problemas e onde fomos forçados a investir mais tempo.

No entanto, apesar das dificuldades que surgiram, sentimos que estamos a aprender mais, e a conseguir ter uma melhor mentalidade sobre como deve ser estruturado um projeto antes e durante o seu desenvolvimento.

# 7. Bibliografia

<https://www.jetbrains.com/help/idea/javafx.html>

<https://www.jetbrains.com/help/idea/ddl-data-sources.html#configuring-ddl-generation-settings>

<https://www.javaguides.net/2020/07/three-tier-three-layer-architecture-in-spring-mvc-web-application.html>

<https://www.jetbrains.com/datagrip/features/generation.html>

<https://cezarcruz.com.br/como-exportar-jar-utilizando-intellij/>

<https://www.geeksforgeeks.org/how-to-add-external-jar-file-to-an-intellij-idea-project/>

<https://www.youtube.com/watch?v=QJddHc41xrM>

<https://blog.jetbrains.com/idea/2021/02/creating-a-simple-jpa-application/>

<https://www.jetbrains.com/help/idea/persistence-tool-window.html#generate_entities_and_mappings>

<https://docs.oracle.com/javafx/2/get_started/fxml_tutorial.htm>

<https://www.geeksforgeeks.org/how-to-add-external-jar-file-to-an-intellij-idea-project/>