



**ESCOLA
SUPERIOR
DE TECNOLOGIA
IPCA**

Licenciatura Engenharia em Desenvolvimento de Jogos

RELATÓRIO DO PROJETO MYZOO

**João Ribeiro - 27926
Martim Moreira - 22354
Gonçalo Veloso - 22348
Hugo Batista - 27921**

Resumo

Para o primeiro trabalho prático, pretende-se que sejam desenvolvidas soluções em C# para problemas reais de complexidade moderada, para isso, o nosso grupo denominado “My zOo”, escolheu fazer um sistema que permita a gestão de tarefas de um jardim zoológico.

Este documento serve como Relatório ao 1º trabalho prático da disciplina de “Programação Orientada a Objetos (POO) – EDJD”.

Declaração de Integridade

Afirmo por minha honra que não recebi qualquer apoio não autorizado na realização deste trabalho prático. Afirmo igualmente que não copiei qualquer material de livro, artigo, documento web ou de qualquer outra fonte exceto onde a origem estiver expressamente citada.

Martim Moreira - 22354

Gonçalo Fernandes - 22348

João Ribeiro - 27926

Hugo Batista - 27921

Índice

1. Introdução	5
1.1 Objetivos	5
1.2 Estrutura do Documento	6
2. Instigadores do Projeto	7
2.1 Propósito	7
2.2 Partes Interessadas	7
2.2.1 Gestores	7
2.2.2 Clientes	7
3. Metodologia de Trabalho	8
3.1 Comunicação e Coordenação	8
3.2 Ambientes de desenvolvimento	8
4. Classes	9
5. Requisitos	10
5.1 Requisitos Funcionais	10
5.2 Requisitos Não Funcionais	10
6. Sistema de gestão	11
7. Base de dados	11
8. Conclusão	12
9. Anexos	13

Índice de Figuras

Figura 1 - Classe Animal	9
Figura 2 - Classe Food	9
Figura 3 - Classe TypeFood	9
Figura 4 - Classe Event	10
Figura 5 - Classe AnimalEvent	10
Figura 6 - Classe Location	10
Figura 7 - Classe Cell	10
Figura 8 - Employee	11
Figura 9 - Classe Job	11

1. Introdução

Neste projeto é proposto o desenvolvimento de um sistema que permita a gestão das tarefas de um jardim zoológico.

Ao desenvolver funcionalidades específicas desde o monitoramento da saúde dos animais até a gestão dos horários de alimentação e abertura para o público, o sistema procura proporcionar uma gestão capaz de satisfazer todos os aspetos envolvidos no funcionamento do zoológico.

Dessa forma, este projeto almeja não apenas simplificar as tarefas diárias, mas também garantir um ambiente saudável para os animais, funcionários e visitantes, promovendo assim uma excelente gestão de um jardim zoológico.

Este projeto necessita de atenção especial devido a vidas de animais estarem em risco, uma falha do sistema pode custar a vida de uma pessoa ou de um animal, daí o Grupo “My zOo” ter tomado atenção ao ambiente em questão e garantir que existem meios emergenciais dentro do sistema.

1.1 Objetivos

O principal objetivo será implementar um sistema capaz de realizar de satisfazer as necessidades dos gestores do zoo. Para isso tivemos como meta neste projeto realizar:

- **Sistema que gere o inventário de animais.**
- **Sistema que gere a alimentação dos animais.**
- **Sistema para auxiliar a limpeza dos habitats / jaulas.**
- **Sistema para gestão de eventos.**
- **Sistema que gere os funcionários do zoo.**
- **Sistemas de alerta para acidentes garantem o cumprimento de prazos.**

1.2 Estrutura do Documento

Este documento está dividido em sete capítulos, sendo este o primeiro introdutório. Os restantes capítulos são:

1. **Instigadores do projeto** – Descrição do problema e dos intervenientes;
2. **Metodologia de Trabalho** – Funcionamento do grupo no âmbito do projeto;
3. **Classes** – Descrição das classes do projeto;
4. **Requisitos** – Levantamento de Requisitos;
5. **Sistema de Gestão do Zoo** – Demonstração do produto final;
6. **Base de dados** – Explicação sobre a base de dados utilizada;
7. **Conclusão** - Análise de resultados;

2. Instigadores do Projeto

Este capítulo tem como objetivo descrever a análise feita para o desenvolvimento do sistema e as suas partes interessadas.

2.1 Propósito

O propósito do projeto é desenvolver um sistema para gerir todos os aspetos de um zoo, desde a saúde e alimentação dos animais, à venda de bilhetes e gestão dos funcionários.

2.2 Partes Interessadas

Nesta fase foram identificados os Gestores de um zoo, e os Clientes como partes interessadas no desenvolvimento do sistema de gestão.

2.2.1 Gestores

Os gestores de zoo irão beneficiar de um sistema lógico, rápido e intuitivo para gerir o zoo.

2.2.2 Clientes

Os clientes poderão beneficiar de uma maneira mais fácil de compra de bilhetes e inscrição em eventos.

3. Metodologia de Trabalho

Para o âmbito do desenvolvimento deste projeto, o grupo “My zOo” utilizou diversas técnicas e ambientes de desenvolvimento para auxiliar o trabalho em equipa.

3.1 Comunicação e Coordenação

Como vias de comunicação foram utilizadas as seguintes ferramentas:

- **Discord** - (Comunicação de Grupo)

Para coordenação de grupo utilizamos:

- **Trello** – (Atribuição de tarefas)
- **Github** – (Repositório de dados)

Martim Moreira - 22354 - **Patoinks no Github**

Gonçalo Fernandes - 22348 - **GONCAVF no Github**

João Ribeiro - 27926 - **Jo6o43 no Github**

Hugo Batista - 27921 - **oliotsuke no Github**

3.2 Ambientes de desenvolvimento

Os principais ambientes de desenvolvimento utilizados neste projeto foram:

- **Visual Studio 2022** – (Desenvolvimento do sistema de gestão)
- **Firebase** - (Base de dados online)

Em **Visual Studio 2022** utilizamos a linguagem de programação “**C#**”, junto da arquitetura **MVC** e utilizando a convenção de nomenclatura **PascalCase** nas funções e **camelCase** para as variáveis.

Na base de dados **Firebase** utilizamos um modelo de base de dados relacional.

4. Classes

Iremos então proceder ao levantamento das classes necessárias para o projeto, tendo em atenção às necessidades de um jardim zoológico.

Classe Animal:

Animal	
PK	<u>Id: int</u>
	Name: str
	DateOfBirth: date
	Family: str
	TypeOfFood: TypeFood
	NeedOfMedicalCheckup: bool
	Gender: str
	IsDeceased: bool
	Cell: Cell

Figura 1 - Classe Animal

```
namespace MyzOo.Models
{
    15 references
    internal class Animal
    {
        2 references
        public int Id { get; set; }
        2 references
        public string Name { get; set; }
        2 references
        public DateTime Birthday { get; set; }
        2 references
        public bool Checkup { get; set; }
        2 references
        public bool IsDeceased { get; set; }

        5 references
        internal enum Gender
        {
            Male,
            Female,
        }

        2 references
        public Gender AnimalGender { get; set; }

        // Reference to Cell and Food
        2 references
        public Cell AnimalCell { get; set; }
        2 references
        public Food AnimalFood { get; set; }

        public static List<Animal> AnimalList = new List<Animal>();

        // Save animal data
        1 reference
        public void SetData(int Id, string Name, DateTime Date, bool Checkup, bool IsDecease, Gender gender, Cell cell, Food food)
        {
            AnimalCrud animal = new AnimalCrud();
            animal.SetData(Id, Name, Date, Checkup, IsDecease, gender, cell, food);
        }
    }
}
```

Classe Food:

Food					
PK	<u>Id: int</u>				
	Description: str				
	Type: TypeFood				
	Stock: int				

Figura 2 - Classe Food

```
namespace MyzOo.Models
{
    20 references
    internal class Food
    {
        3 references
        public int Id { get; set; }
        2 references
        public string Description { get; set; }
        2 references
        public TypeFood TypeFood { get; set; }
        2 references
        public int Stock { get; set; }

        public static List<Food> FoodList = new List<Food>();

        // Save animal data
        0 references
        public void SetData(int Id, string Description, TypeFood typeFood, int Stock)
        {
            FoodCrud food = new FoodCrud();
            food.SetData(Id, Description, typeFood, Stock);
        }
    }
}
```

Classe TypeFood:

TypeFood					
PK	<u>Id</u>				
	Type: str				

Figura 3 - Classe TypeFood

Classe Event:

Event	
PK	<u>Id: int</u>
	Name: str
	Description: str
	DateBegin: date
	HourBegin: time(Minutes)
	Duration: int
	NumAttendees: int
	ListAnimalsAttending: list(Animal)
	Location: Location

Figura 4 - Classe Event

Classe AnimalEvent:

AnimalEvent	
PK	<u>Id</u>
	Animal: Animal
	Event: Event

Figura 5 - Classe AnimalEvent

Classe Location:

Location	
PK	<u>Id</u>
	Location: str

Figura 6 - Classe Location

Classe Cell:

Cell			
PK	<u>Id</u>		
	Location: Location		
	Animals: list(Animals)		
	NeedOfCleaning: bool		

Figura 7 - Classe Cell

Classe Employee:

Employee			
PK	<u>Id: int</u>		
	Name: str		
	Birthday: date		
	Password: str		
	Adminlevel: int		
	Job: Job		

Classe Job:

Job			
PK	<u>Id: int</u>		
	Role: str		
	Salary: int		

5. Requisitos

Neste capítulo, serão apresentados os requisitos levantados do sistema de gestão do zoo.

5.1 Requisitos Funcionais

RF 1 – O Sistema deve permitir o acesso aos gestores por meio de credenciais.

RF 2 – O Sistema deve manter um catálogo completo de todos os animais no zoológico e as suas informações detalhadas.

RF 3 – O Sistema deve permitir criar e gerir um calendário para a alimentação dos animais.

RF 4 – O Sistema deve incluir um módulo de bilheteira, onde é possível controlar o acesso dos visitantes ao Zoo.

RF 5 – O Sistema deve incluir alertas e notificações para acesso imediato a informações críticas.

RF 6 – O Sistema deve permitir o registo dos habitats/jaulas dos animais.

RF 6.1 – O Sistema deve permitir controle da manutenção dos habitats/jaulas dos animais.

RF 7 – O Sistema deve permitir a gestão dos funcionários e guardar os seus detalhes.

RF 8 – O Sistema deve permitir gestão da localização dos eventos e dos habitats/jaulas do zoo.

RF 9 – O Sistema deve permitir que um gestor possa registar outros gestores para dar acesso ao sistema.

RF 10 – O Sistema deve possuir uma base de dados online.

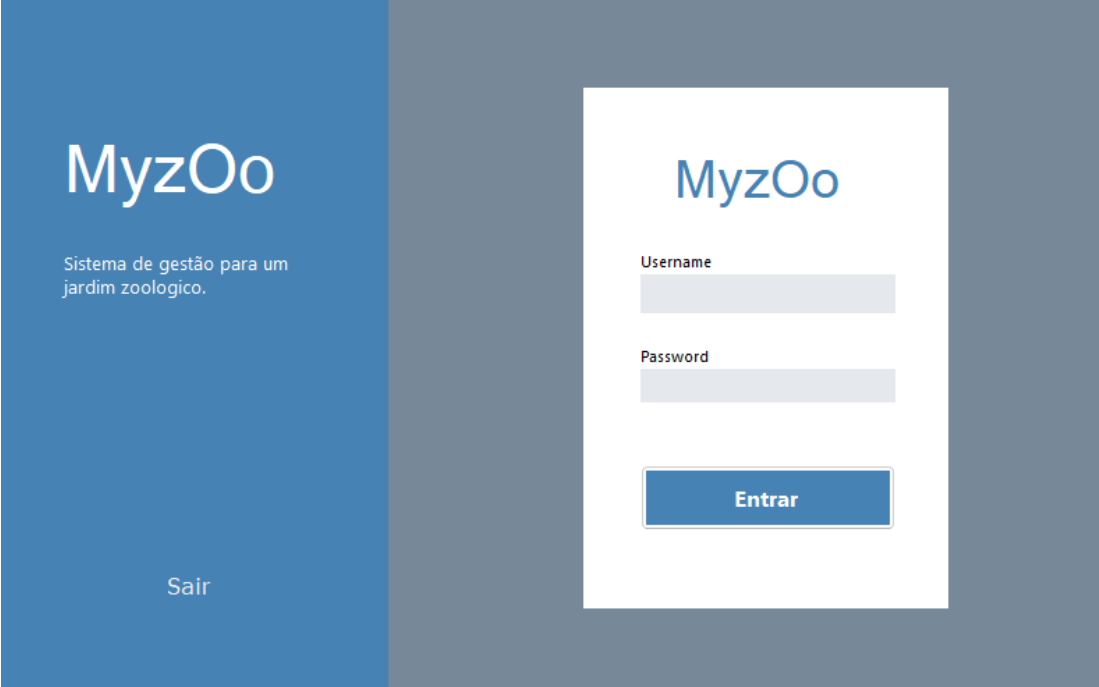
5.2 Requisitos Não Funcionais

RNF 1 – O Sistema deve ser capaz de se manter online 24h por dia.

RNF 2 – O Sistema deve possuir uma interface lógica e rápida.

6. Sistema de gestão

Módulo 1 - Login



The image shows the login screen for the MyzOo system. It features a blue sidebar on the left with the MyzOo logo and the text 'Sistema de gestão para um jardim zoológico.' Below this is a 'Sair' button. The main area on the right is grey and contains a white login box. Inside the box, the MyzOo logo is at the top, followed by 'Username' and 'Password' labels with input fields. At the bottom of the box is a blue 'Entrar' button.

Para aceder à aplicação é necessário um login para maior segurança, o registo é realizado mais tarde por algum gestor do software.

My zOo

Registrar Celas

Lista Celas

Registrar Comida

Lista Comida

Registrar Evento

Lista Eventos

Registrar Funcionário

Lista Funcionário

Registrar Animal

Lista Animal

Este é o menu principal da aplicação que possui todas as ferramentas para a gestão do Zoo.

Registrar funcionario

Nome do Funcionário :

Data de Aniversário :

2 de janeiro de 2024

Função :

Palavra Passe :

Nível de Administração :

Finalizar

Menu para realizar o registo de um novo funcionário.

Lista de Funcionários - X

ola
martim
joão

Menu de listagem de todos os funcionários, ao clicar em qualquer um pode editá-lo.

Editar Funcionário - X

Nome do Funcionário :

Data de Aniversário :

Função :

Nível de Administração :

Nova password:

Menu de edição de funcionário.

Adicionar Animal x

Nome do Animal :

Data do Animal :

Cela do Animal :

Comida do Animal :

Genero :

☐ Doente

Registrar

Menu para realizar o registo de um novo animal.

Lista de Animais - x

Martim
Pato Francês
Égua Descalça
Garrafa de Água

Menu de listagem de todos os animais, ao clicar em qualquer um pode editá-lo.

Inspecionar Animal

Nome do Animal :

Pato Francês

Data do Animal :

2 de janeiro de 2024

Cela do Animal :

15

Comida do Animal :

Teste

Genero :

Male

☒ Doente

Atualizar

Apagar

Menu de edição de animal.

Adicionar Cella

Número da Cella :

84

Descrição da Cella:

É uma cela, e pode caber animais dentro.

☐ Limpeza

Registrar

Menu para realizar o registo de uma nova cela.

Lista de Celas - X	
1	
15	
24	
83	

Menu de listagem de todas as celas, ao clicar em qualquer um pode editá-lo.

Inspeccionar Cella X	
Número da Cella :	Descrição da Cella:
15	Cavalos
<input type="checkbox"/> Limpeza	
Atualizar	Apagar

Menu de edição das celas.

Adicionar Comida

Comida :
Came

Stock:
0

Descrição da Comida:
Água

Registrar

Menu para realizar o registo de uma nova comida.

Lista de Comidas

Pizza
Teste

Menu de listagem de todas as celas, ao clicar em qualquer um pode editá-lo.

Inspecionar Comida x

Comida :
 Peixe ▼

Stock:
 0

Descrição da Comida:
 Pizza

Atualizar

Apagar

Menu de edição das comidas.

Adicionar Evento x

Descrição do Evento :

Data do Evento :
 2 de janeiro de 2024

Hora do Evento :
 23:40:24

Limite de Pessoas para o Evento :

Duração Prevista para o Evento :

Localização :

Registrar

☐ Martim
☐ Pato Francês
☐ Égua Descalça
☐ Garrafa de Água

Menu para realizar o registo de um novo evento.

Lista de Eventos

- X

Evento Bom

Menu de listagem de todos os eventos, ao clicar em qualquer um pode editá-lo.

Inspeccionar Animal

X

Nome do Animal :

Data do Animal :

2 de janeiro de 2024

Cela do Animal :

Comida do Animal :

Genero :

☐ Doente

Atualizar

Apagar

Menu de edição dos eventos.

7. Base de dados

Para a nossa base de dados online utilizamos a “Firebase” junto do modelo de base dados relacionais, porém, apesar da “Firebase” ter uma estrutura de base de dados não-relacional, nós imitamos uma relacional, devido à familiarização e por ser mais lógico e intuitivo.

A estrutura da mesma está representada no modelo de classes, modelo que não pode ser colocado neste documento devido ao seu tamanho, pode ser encontrado nos anexos.

8. Conclusão

O Projeto em questão colocou o grupo numa situação desconhecida, devido a ser a primeira vez que trabalhamos em um projeto juntos.

O Resultado da análise que fizemos acreditamos ser um resultado bastante positivo, pois tomamos em atenção todos os detalhes, sabendo do risco que é gerir um zoo.

9. Anexos

Repositório do GitHub:

- <https://github.com/DigitalGameDevTeam/MyzOo>

Diagrama de classes

- https://drive.google.com/file/d/1LqW4VAA02RrhKz6C55LDRdhGA8ZX3cUV/view?usp=drive_link