

# Simulador Epidémico Interativo de Observação Sistémica

Projeto Interdisciplinar

**João Ventura – 2024064**

**José Dias Ferreira - 2024128**

Licenciatura em Engenharia Informática

Sistemas Multimédia 1

Professor Helder Pinto

Porto, 2026

## Resumo

A preencher só na Sprint C.

Máximo 300 palavras.

Um resumo é uma representação abreviada e precisa de um documento, sem acrescento de interpretação ou crítica, escrita de forma impessoal (ISO 214).

O resumo deve incluir as seguintes componentes:

- Parágrafo inicial de introdução do contexto geral do trabalho.
- Resumo dos aspetos mais importantes do trabalho descrito para ajudar o leitor a decidir se deve ou não consultar o restante relatório.
- Parágrafo final com as conclusões do trabalho realizado.

**Palavras-chave:** Mínimo 3 e máximo 6 palavras-chave.

# Índice

<i>Resumo</i> .....	1
<i>Índice</i> .....	2
<i>Índice de Figuras</i> .....	4
<i>Índice de Tabelas</i> .....	5
<i>Lista de Acrónimos e Siglas</i> .....	6
<b>1. Introdução</b> .....	<b>7</b>
1.1. Enquadramento .....	7
1.2. Objetivos.....	8
<b>2. Desenvolvimento</b> .....	<b>9</b>
<b>2.1. Planeamento do Projeto</b> .....	<b>9</b>
2.1.1. Metodologia .....	9
2.1.2. Cronograma .....	9
2.1.3. Recursos .....	9
<b>2.2. Análise de Sistemas</b> .....	<b>10</b>
2.2.1. Levantamento de Requisitos — FURPS+ .....	10
2.2.2. Diagrama de Use Cases .....	10
2.2.3. Descrição Estruturada do UC "Executar Simulação" .....	11
2.2.4. Fluxograma do UC "Executar Simulação" .....	11
2.2.5. Modelo de Domínio (DDD) .....	12
2.2.6. Modelo Relacional de Dados.....	13
2.2.7. Dicionário de Dados .....	14
<b>2.3. Estudo UI/UX</b> .....	<b>15</b>
<b>2.4. Desenvolvimento do Projeto</b> .....	<b>15</b>
<b>2.5. Resultados</b> .....	<b>15</b>
<b>2.6. Subtítulo Exemplo</b> .....	<b>16</b>
<b>3. Conclusão</b> .....	<b>18</b>
3.1. Limitações e Sugestões Futuras.....	18
3.2. Considerações sobre a Inteligência Artificial no Contexto deste Projeto .....	18
<b>Referências</b> .....	<b>19</b>



# **Índice de Figuras**

A atualizar em cada Sprint.

Figura 1 - Diagrama Use Cases .....	11
Figura 2- Fluxograma do UC "Executar Simulação" .....	12
Figura 3- Modelo de Domínio .....	13
Figura 4 - Modelo Relacional de Dados .....	14
Figura 5- Dicionário de Dados .....	15
Figura 6 – Legenda inserida automática a partir do menu: referências   Inserir legenda .....	16

# **Índice de Tabelas**

**No table of figures entries found.**

# **Lista de Acrónimos e Siglas**

A atualizar em cada Sprint.

<b>API</b>	<i>Application Programming Interface</i> , interface utilizada para comunicação entre frontend e backend.
<b>CPU</b>	<i>Central Processing Unit</i> , processador utilizado para execução da simulação (se relevante no FURPS+).
<b>CRUD</b>	<i>Create, Read, Update, Delete</i> , conjunto de operações básicas sobre dados.
<b>DDD</b>	<i>Domain-Driven Design</i> , abordagem de modelação centrada no domínio e agregados.
<b>ERD</b>	<i>Entity-Relationship Diagram</i> , diagrama de modelo relacional de dados.
<b>FK</b>	<i>Foreign Key</i> , chave estrangeira numa tabela.
<b>FURPS+</b>	Modelo de classificação de requisitos: <i>Functionality, Usability, Reliability, Performance, Supportability</i> + requisitos adicionais.
<b>Git</b>	Sistema de controlo de versões distribuído.
<b>GitHub</b>	Plataforma de alojamento de repositórios Git utilizada para colaboração.
<b>I(t)</b>	Número de infetados num instante $t$
<b>IDE</b>	<i>Integrated Development Environment</i> , ambiente de desenvolvimento (VS Code, PyCharm).
<b>ISO</b>	<i>International Organization for Standardization</i>
<b>JSON</b>	<i>JavaScript Object Notation</i> , formato de dados utilizado na API.
<b>PBL</b>	<i>Project Based Learning</i>
<b>PK</b>	<i>Primary Key</i> , chave primária numa tabela.
<b>R(t)</b>	Número de recuperados num instante $t$
<b>Rt</b>	<i>Número de Reprodução Efetivo</i> , métrica que indica quantas pessoas, em média, um infetado contagia num momento específico da simulação.
<b>S(t)</b>	Número de suscetíveis num instante $t$
<b>SIR</b>	<i>Susceptible–Infected–Recovered</i> , modelo epidemiológico clássico utilizado para simulação da propagação de doenças.
<b>UC</b>	<i>Use Case</i> (Caso de Uso).
<b>UI</b>	<i>User Interface</i> , interface com o utilizador.
<b>UML</b>	<i>Unified Modeling Language</i> , linguagem padrão de modelação de sistemas de software.
<b>UX</b>	<i>User Experience</i> , experiência geral do utilizador ao interagir com o sistema.
<b><math>\beta</math> (beta)</b>	Taxa de infeção
<b><math>\gamma</math> (gamma)</b>	Taxa de recuperação
<b><math>\Delta t</math></b>	Step temporal da simulação

# 1. Introdução

O presente relatório descreve o trabalho desenvolvido no âmbito do Projeto Interdisciplinar da Unidade Curricular de Sistemas Multimédia I, integrado no segundo ano da Licenciatura em Engenharia Informática do ISTECP Porto.

O projeto tem como objetivo o desenvolvimento de um **Simulador Epidémico Interativo**, capaz de modelar e visualizar a evolução de uma doença infeciosa com base no modelo epidemiológico SIR (Suscetíveis–Infetados–Recuperados).

Este documento apresenta o enquadramento teórico e tecnológico, os objetivos gerais do projeto, bem como o processo de análise realizado durante a Sprint A.

A sprint incidiu essencialmente na fase de planeamento e modelação, englobando o levantamento de requisitos funcionais e não funcionais, o estudo dos atores e funcionalidades, a definição dos casos de uso e respetivo cenário de sucesso, a construção dos modelos de domínio e da base de dados, bem como a organização inicial do repositório colaborativo para desenvolvimento.

O relatório encontra-se organizado em secções que refletem o progresso do trabalho. A secção inicial apresenta o contexto e os objetivos do projeto, seguindo-se a secção de desenvolvimento, onde é descrita a metodologia adotada, o planeamento, os modelos produzidos e as análises realizadas até ao final da Sprint A.

## 1.1. Enquadramento

O Simulador Epidémico Interativo surge como resposta à necessidade de compreender e visualizar a dinâmica de propagação de doenças infecciosas, numa época em que fenómenos epidémicos e pandémicos têm impacto direto na sociedade, na saúde pública e na tomada de decisões. O projeto enquadra-se no contexto multidisciplinar das unidades curriculares de Sistemas Multimédia I, Programação III, Tecnologias de Internet II, Bases de Dados e Teoria das Probabilidades e Modelos de Simulação.

Enquanto estudantes da Licenciatura em Engenharia Informática, este projeto permite aplicar conhecimentos transversais adquiridos ao longo do curso, integrando conceitos de modelação, desenvolvimento web, programação orientada a objetos, gestão de dados e visualização de informação.

A motivação para o tema advém da relevância atual da modelação epidemiológica e da oportunidade de criar uma plataforma interativa que permita ao utilizador explorar diferentes cenários e observar o seu impacto através de representações visuais claras e acessíveis.

Este enquadramento fornece a base necessária para estruturar o trabalho técnico desenvolvido nas fases seguintes do projeto.

## 1.2. Objetivos

O projeto tem como objetivo geral desenvolver uma aplicação web interativa que permita configurar e simular a evolução de uma epidemia, com base em modelos probabilísticos aplicados ao modelo SIR.

De forma mais detalhada, destacam-se os seguintes objetivos específicos:

- **Implementar uma plataforma web intuitiva**, desenvolvida em React, que permita ao utilizador configurar parâmetros epidemiológicos e visualizar resultados.
- **Desenvolver um back-end robusto**, em Python com Flask, capaz de executar simulações estocásticas com base em parâmetros como população total, taxa de infecção ( $\beta$ ), taxa de recuperação ( $\gamma$ ) e duração da simulação.
- **Criar um modelo de dados eficiente**, que permita guardar parâmetros, resultados e métricas derivadas (como  $R_t$ ), garantindo persistência e histórico.
- **Construir um dashboard**, que apresente a evolução temporal dos estados S, I e R através de gráficos interativos.
- **Garantir modularidade e escalabilidade**, preparando o sistema para futuras extensões e melhorias.
- **Aplicar metodologias de desenvolvimento colaborativo**, com recurso a GitHub e boas práticas de versionamento.

## 2. Desenvolvimento

A secção seguinte descreve o trabalho realizado durante a Sprint A, incluindo a metodologia adotada, o planeamento, o levantamento de requisitos, os modelos produzidos e a análise das principais funcionalidades do sistema.

### 2.1. Planeamento do Projeto

#### 2.1.1. Metodologia

O desenvolvimento do projeto segue uma abordagem inspirada em metodologias ágeis, com organização do trabalho em sprints incrementais. Cada sprint possui objetivos claros, entregáveis definidos e uma revisão final. A Sprint A focou-se exclusivamente na fase conceptual, envolvendo análise, modelação e estruturação inicial do sistema.

Foi também utilizada a metodologia **FURPS+** para o levantamento de requisitos funcionais e não funcionais, garantindo uma visão abrangente do sistema desde o início.

#### 2.1.2. Cronograma

O cronograma fornecido pelo docente serviu de base para o planeamento interno. A Sprint A corresponde à fase inicial do projeto e envolve:

- Levantamento de requisitos (FURPS+)
- Revisão e atualização do diagrama de Use Cases
- Modelação do UC “Executar Simulação”
- Criação dos modelos UML (Domínio, ERD)
- Criação do Dicionário de Dados
- Configuração do repositório GitHub
- Preparação do relatório e apresentação

As sprints seguintes abordarão a implementação do backend, frontend, integração, testes e finalização da documentação.

#### 2.1.3. Recursos

Os principais recursos necessários ao desenvolvimento incluem:

- **Hardware:** computador pessoal com capacidade para executar servidores locais e ferramentas de desenvolvimento.

- **Software:**
  - React.js (frontend)
  - Python + Flask (backend)
  - MySQL (base de dados)
  - Draw.io (modelação UML)
  - GitHub (controlo de versões)
  - VS Code / PyCharm
  - Microsoft Office
- **Equipa:** João Ventura e José Ferreira, responsáveis pela modelação, implementação backend, frontend, análise funcional e gestão de projeto.

## 2.2. Análise de Sistemas

A análise do sistema teve como objetivo compreender o domínio do problema, identificar utilizadores, funcionalidades e requisitos, bem como modelar a estrutura lógica e de dados que dará suporte ao sistema final.

Durante esta fase, foram produzidos os seguintes artefactos:

### 2.2.1. Levantamento de Requisitos — FURPS+

Documento onde se identificam requisitos funcionais, usabilidade, fiabilidade, desempenho, suporte e requisitos adicionais como segurança e histórico.

### 2.2.2. Diagrama de Use Cases

Representa todas as interações possíveis entre os três perfis de utilizador (não registado, registado e administrador) e o sistema. O diagrama identifica funcionalidades como executar simulação, configurar parâmetros, consultar histórico, gerir utilizadores e visualizar estatísticas.

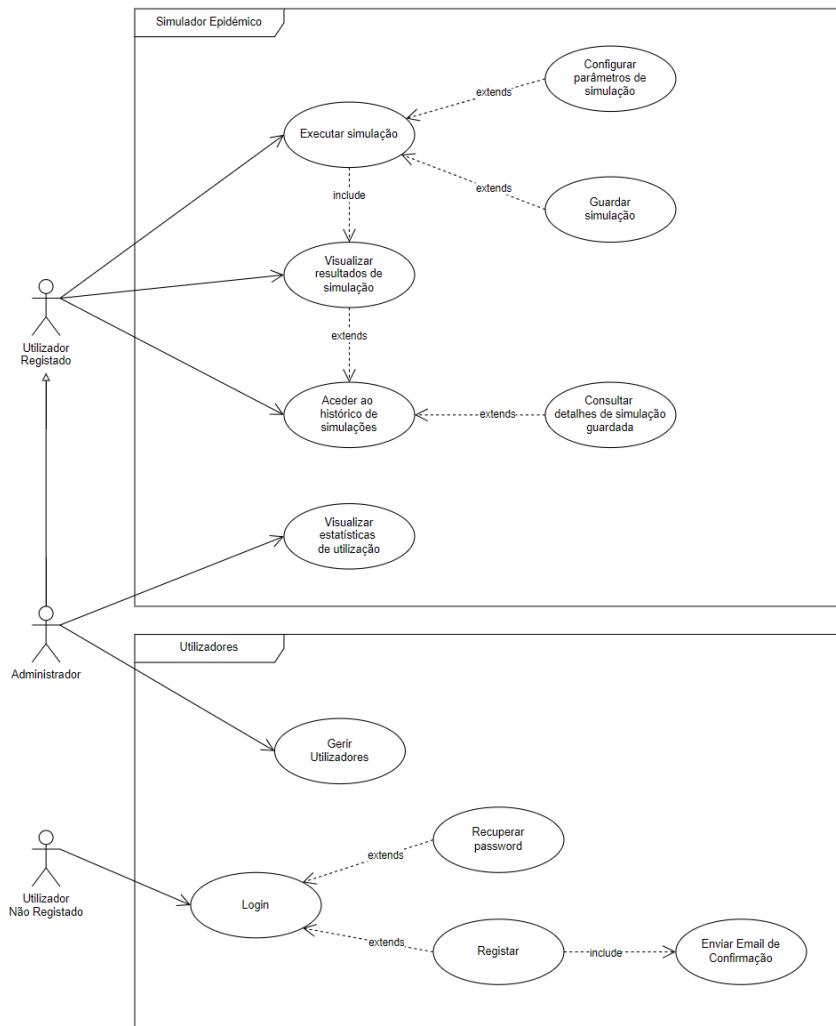


Figura 1 - Diagrama Use Cases

### 2.2.3. Descrição Estruturada do UC "Executar Simulação"

Inclui o cenário de sucesso principal, descrevendo passo a passo o fluxo funcional desde a introdução dos parâmetros até à visualização e armazenamento dos resultados.

### 2.2.4. Fluxograma do UC "Executar Simulação"

Representa o comportamento interno do processo, incluindo validação dos parâmetros, execução do modelo SIR, iteração por steps, cálculo de métricas Rt e armazenamento da informação.

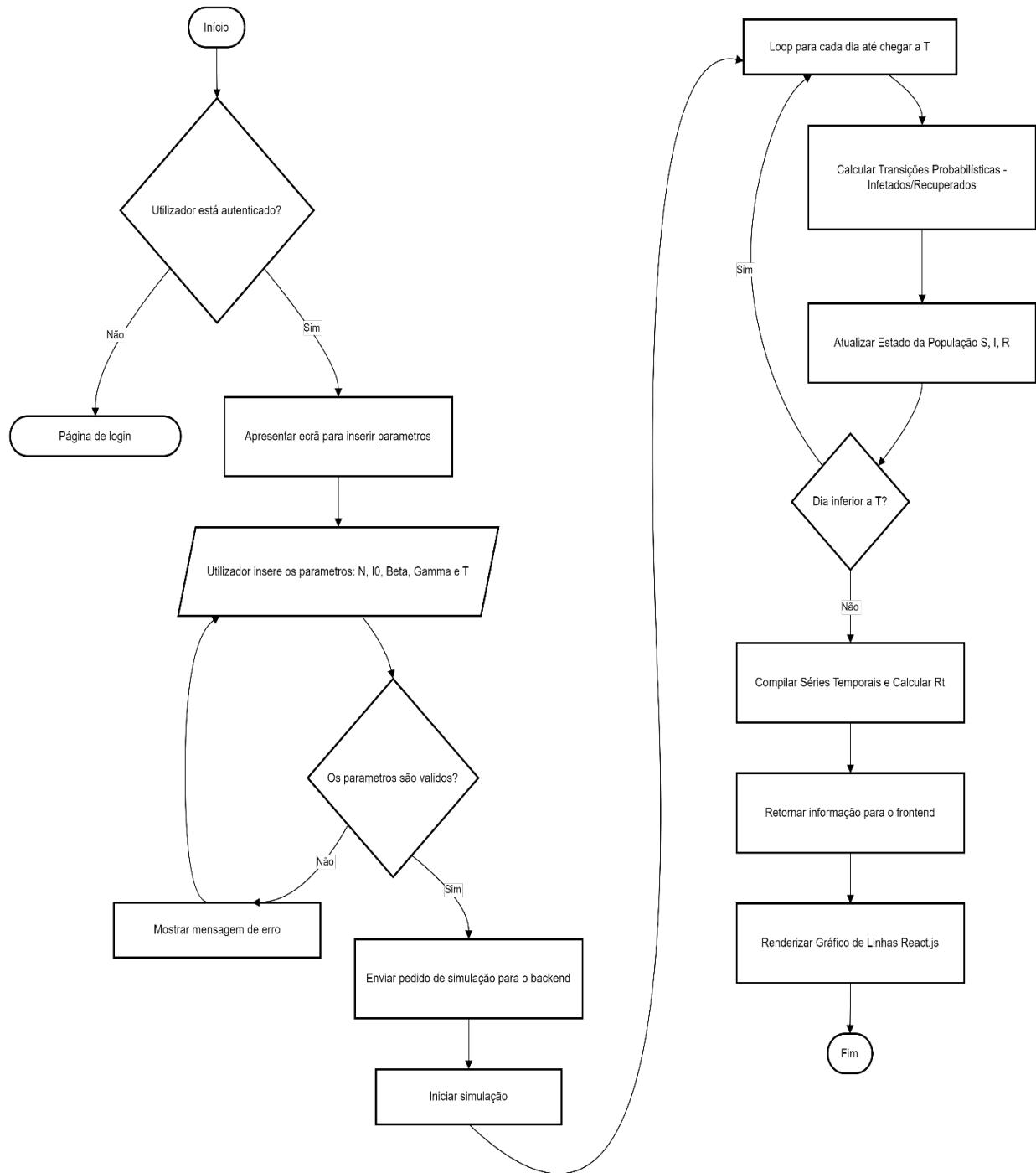


Figura 2- Fluxograma do UC "Executar Simulação"

### 2.2.5. Modelo de Domínio (DDD)

Define as entidades, agregados e relações principais do sistema: User, Simulation, SimulationParameters, SimulationResults, SimulationMetrics e System Stats.

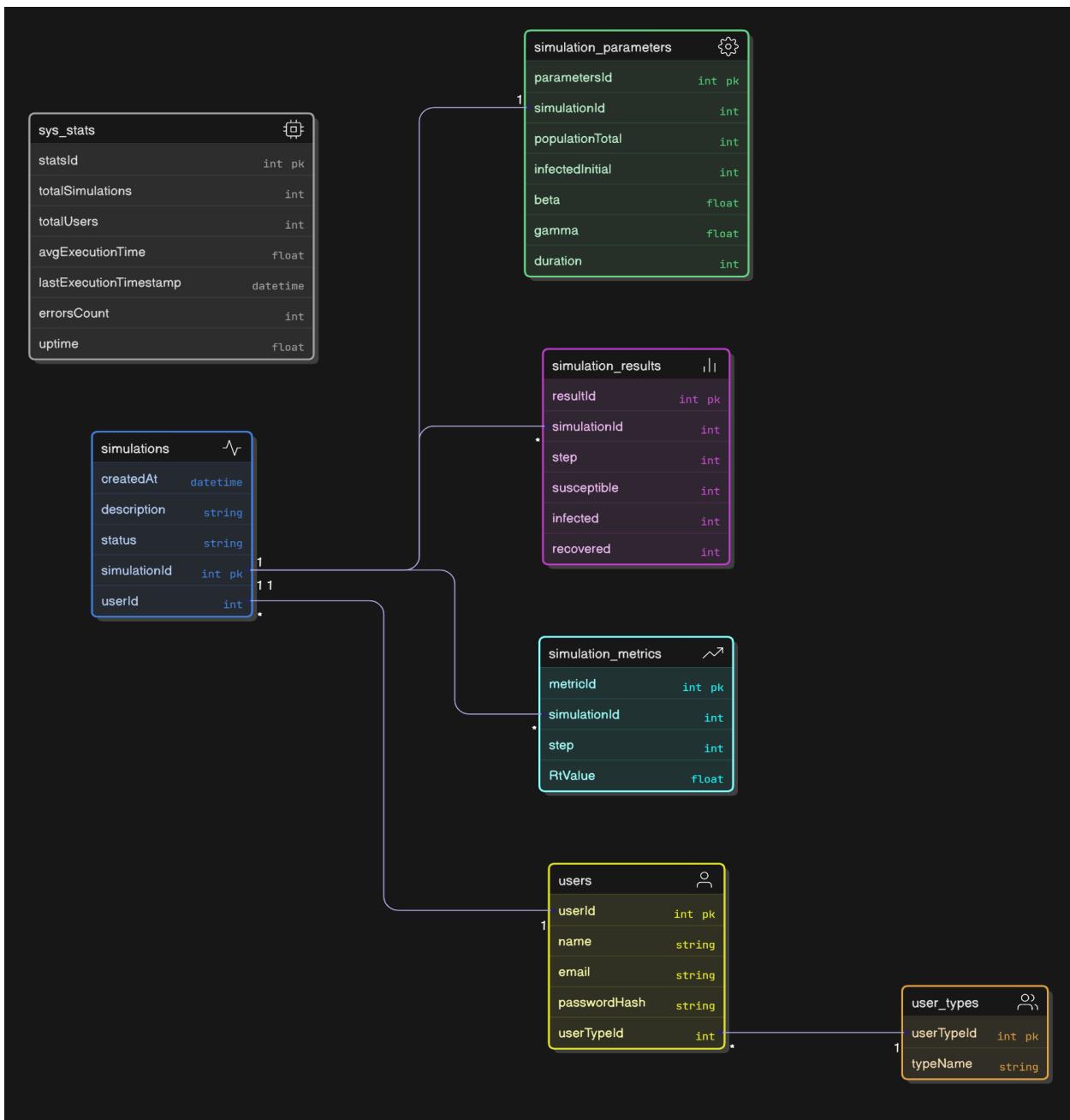


Figura 3- Modelo de Domínio

## 2.2.6. Modelo Relacional de Dados

O modelo foi refinado para incluir novas tabelas e campos essenciais ao modelo epidemiológico, garantindo suporte completo às simulações e à visualização gráfica.

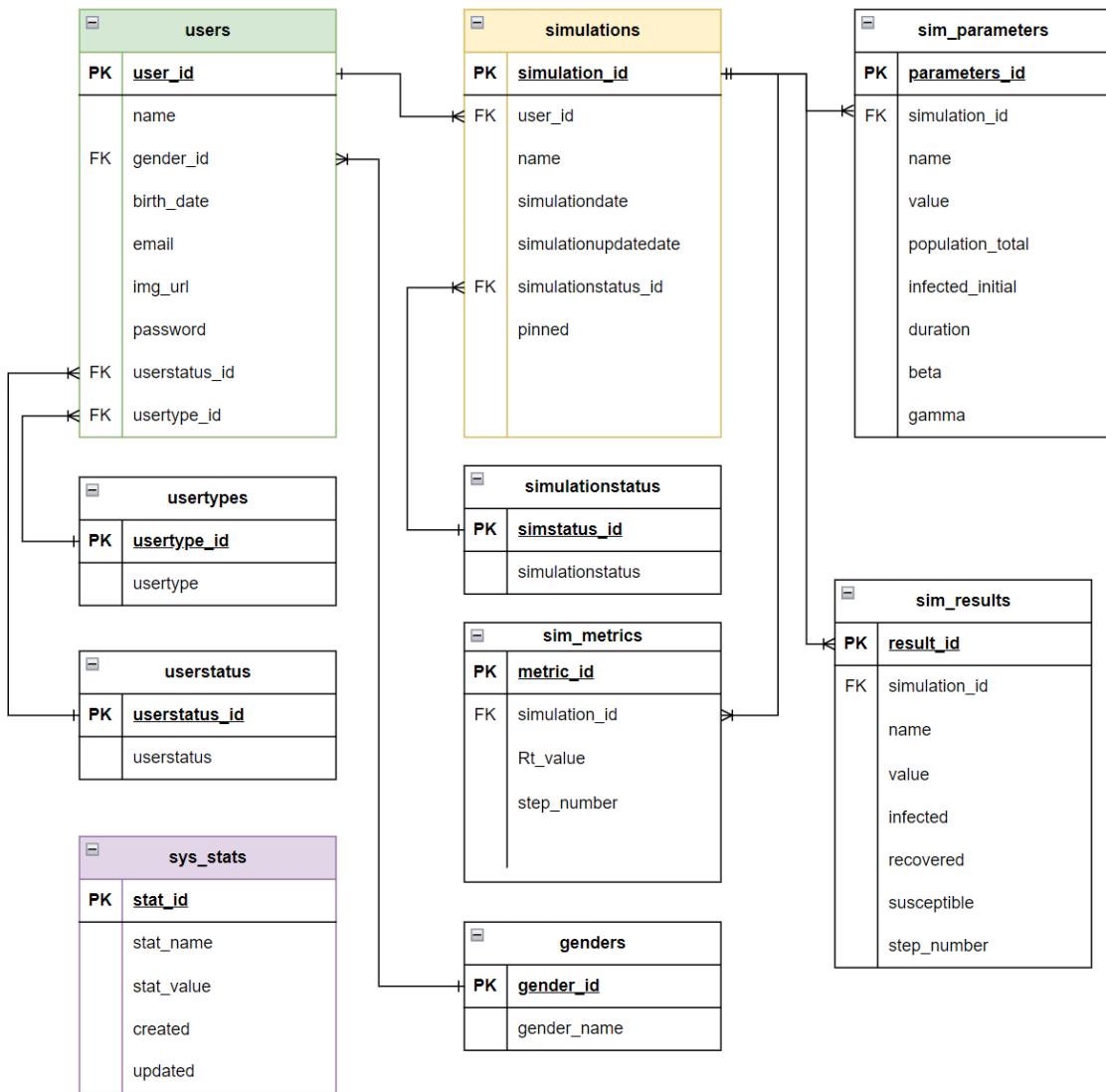


Figura 4 - Modelo Relacional de Dados

### 2.2.7. Dicionário de Dados

Documento contendo definições detalhadas de todas as tabelas, atributos, tipos de dados, restrições e relações da base de dados.

Tabela: User				
Campo	Tipo	PK	FK	Descrição
user_id	INT	PK	—	Identificador do utilizador
name	VARCHAR(100)	—	—	Nome completo
email	VARCHAR(100)	—	—	Email único
gender_id	INT	—	FK	Identificador do género
birth_date	DATETIME	—	—	Data de Nascimento
img_url	VARCHAR(255)	—	—	URL da imagem de perfil
password	VARCHAR(255)	—	—	password (possível hash?)
userstatus_id	INT	—	FK	ID to Status
user_type_id	INT	—	FK	ID do tipo de User

Tabela: UserType				
Campo	Tipo	PK	FK	Descrição
usertype_id	INT	PK	—	Identificador
usertype	VARCHAR(50)	—	—	ex: "registered", "admin"

Tabela: UserStatus				
Campo	Tipo	PK	FK	Descrição
usestatus_id	INT	PK	—	Identificador
usestatus	VARCHAR(50)	—	—	ex: "active", "pending", "suspended"

Tabela: SimulationResults				
Campo	Tipo	PK	FK	Descrição
result_id	INT	PK	—	Identificador
simulation_id	INT	—	FK	Simulação
step_number	INT	—	—	Dia/step
susceptible	INT	—	—	S
infected	INT	—	—	I
recovered	INT	—	—	R

Tabela: Genders				
Campo	Tipo	PK	FK	Descrição
gender_id	INT	PK	—	Identificador
gender_name	VARCHAR(50)	—	—	Nome do Género

Tabela: Simulations				
Campo	Tipo	PK	FK	Descrição
simulation_id	INT	PK	—	Identificador
user_id	INT	—	FK	Criador da simulação
created_at	DATETIME	—	—	Data da criação
description	TEXT	—	—	Descrição
pinned	boolean	—	—	Permite listar e.g. favoritos
status	VARCHAR(20)	—	—	completed, failed, running

Tabela: SimulationStatus				
Campo	Tipo	PK	FK	Descrição
simstatus_id	INT	PK	—	Identificador
simstatus	VARCHAR(50)	—	—	ex: "complete", "paused", "deleted"

Tabela: SimulationMetrics				
Campo	Tipo	PK	FK	Descrição
metric_id	INT	PK	—	Identificador
simulation_id	INT	—	FK	Simulação
step_number	INT	—	—	Dia/step
Rt_value	FLOAT	—	—	Número de reprodução efetivo

Legenda de Cores:			
Simulation-centric		User-centric	
Admin-centric			

Figura 5- Dicionário de Dados

## 2.3. Estudo UI/UX

A preencher na Sprint B.

## 2.4. Desenvolvimento do Projeto

A preencher na Sprint B e Sprint C.

Descrição do Processo de Trabalho: Detalhar o desenvolvimento do projeto, desde a conceção até à execução, incluindo os desafios enfrentados.

Ferramentas Utilizadas: Listar as ferramentas (software/hardware) usadas durante o desenvolvimento.

Soluções Implementadas: Descrever as soluções encontradas para resolver os problemas identificados e como essas soluções foram implementadas.

Documentos importantes produzidos ou utilizados durante o trabalho e que, pela sua dimensão, não seja possível colocar nesta secção, devem constar nos anexos ou nas referências.

## 2.5. Resultados

A preencher só na Sprint C.

Produto Final: Apresentar o produto final desenvolvido, com imagens, gráficos ou capturas de ecrã.

Testes e Validação: Descrever como o projeto foi testado e validado, apresentando os resultados obtidos e as conclusões tiradas a partir dos testes.

## Comentários relativos a formatos a utilizar:

Títulos:

- Usar os estilos: Heading 1; Heading 2; Heading 3

Corpo de texto:

- Usar o estilo: Normal

A secção da capa não deve incluir cabeçalhos nem rodapés.

A secção do resumo e dos índices deve ser paginada com numeração romana, incluindo cabeçalho com o título do projeto e nome do(s) autor(es).

A secção dos capítulos principais deve ser paginada com numeração árabe, incluindo cabeçalho com o título do projeto e nome do(s) autor(es).

Legendas:

- Usar o estilo: Caption

As figuras e tabelas devem ser referidas no corpo do texto e posicionadas corretamente, junto ao parágrafo onde são referidas pela primeira vez. Podem ser inseridas com numeração automática, seguindo as instruções ilustradas na figura 1. A legenda de uma figura deve ser colocada por baixo da figura. A legenda de uma tabela deve ser colocada por cima da tabela a que diz respeito.

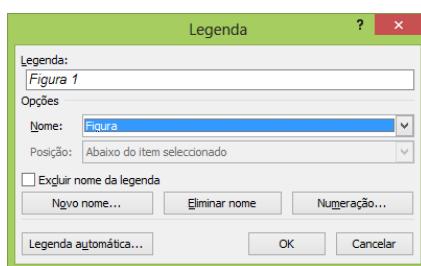


Figura 6 – Legenda inserida automática a partir do menu: referências | Inserir legenda

## **2.6. Subtítulo Exemplo**

Texto adequado.



### **3. Conclusão**

A preencher só na Sprint C.

Sumário do Projeto: Recapitular brevemente o que foi desenvolvido ao longo do projeto.

Objetivos Alcançados: Avaliar se os objetivos inicialmente propostos foram atingidos.

#### **3.1. Limitações e Sugestões Futuras**

A preencher só na Sprint C.

Dificuldades Encontradas: Descrever as principais dificuldades enfrentadas e como foram resolvidas.

Sugestões Futuras: Apresentar sugestões para melhorias ou trabalhos futuros que poderiam ser feitos com base no projeto.

#### **3.2. Considerações sobre a Inteligência Artificial no Contexto deste Projeto**

A preencher só na Sprint C.

Texto adequado.

## **Referências**

Deve continuar a utilizar a numeração das páginas do relatório.

Exemplo de Referências:

Pinto, H. (2024). *Scrum Project Based Learning (PBL) na Educação da Licenciatura em Engenharia Informática*. Pratica – p.porto.