

FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

MESTRADO INTEGRADO EM ENGENHARIA ELECTROTÉCNICA E DE
COMPUTADORES

Sistema de Combate à Violência
REDES DE COMPUTADORES
PL2

• U C •



04/2021

Contents

Índice	I	
1	Introdução	1
2	Objetivos do trabalho	2
3	Gestão de Equipa	3
3.1	Tarefa 1	3
3.2	Tarefa 2	5
3.3	Tarefa 3	7
4	Gestão de cliente	8
4.1	Mock-ups	8
4.1.1	Login - Sign Up	8
4.1.2	Aplicação para o Profissional de Saúde (APS)	9
4.1.3	Aplicação de Gestor de Sistema(AGS)	10
4.1.4	Aplicação para o Agente de Segurança (AAS)	11
4.2	Diagrama de Funcionamento	14
5	Gestão de Software	16
5.1	Softwares	16
5.1.1	Trello	16
5.1.2	Github	17
5.1.3	Python3 e SQL	17
5.1.4	Postgresql	18
5.1.5	Onda	18
5.1.6	Discord	19
5.1.7	Overleaf	20
6	Gestão de Riscos e Testes	21
6.1	Análise de pontos vulneráveis	21
6.2	Análise de testes	23
7	Gestão de Qualidade	28
7.1	Estado da arte	28
7.1.1	Ankira	28
7.1.2	Callisto	29
7.1.3	Conclusão	30
7.2	Qualidade Geral do Projeto	31
7.2.1	Plano geral da qualidade	31

7.2.2	Controlo e melhoria da qualidade:	31
8	Equipa de desenvolvimento	32
8.1	Estrutura	32
8.1.1	Aplicação Central	32
8.1.2	Aplicação Cliente e Redireccionamento	33
8.1.3	Código de desenvolvimento de software	33

1 Introdução

Como proposto pelo docente da cadeira de Redes de Computadores, MiEEC, iremos desenvolver uma aplicação com recurso a sockets, cuja interface com utilizador irá ser feita com recurso à linguagem de programação Python. Com a realização deste projeto esperamos colocar em prática e consolidar os conhecimentos adquiridos em Redes de Computadores, nomeadamente a utilização de sockets.

Este breve relatório descreve qual o objetivo do programa e as especificações de interface com o utilizador. Nele explicitamos também o plano de distribuição de tarefas aos autores deste projeto e ainda outras informações relevantes ao desenvolvimento da aplicação.

2 Objetivos do trabalho

O objetivo do trabalho é implementar um sistema de registo, de alerta, de prevenção e de combate à violência contra profissionais no sector da saúde. Para tal, iremos desenvolver um conjunto de aplicações de suporte que recorrem a sockets TCP e UDP.

APS - Aplicação para o Profissional de Saúde

Esta aplicação permite a qualquer profissional de saúde, depois de devidamente autenticado na aplicação, por um sistema de Registo/Login, registar na aplicação qualquer ocorrência de violência.

Estas ocorrências seguem o seguinte formato: *Data;Hora;Local;Tipo de Agressão;Nome*

AGS - Aplicação de Gestor de Sistema

Esta aplicação possibilita que um gestor, depois de devidamente autenticado pelo sistema de Login, tendo um username e uma password destinada apenas ao próprio, possa consultar, validar e apagar qualquer registo no sistema de novos profissionais de saúde e/ou de agentes de segurança.

AAS - Aplicação para o Agente de Segurança

Esta aplicação torna possível aos agentes de segurança consultar todas as ocorrências de violência reportadas pelos utilizadores, tem ainda a possibilidade de aplicar filtros de consulta das ocorrências, nomeadamente por data, por local e por pessoa.

AC - Aplicação Central

Como indicado, esta é a aplicação com a qual todas as anteriores irão interagir através de sockets. Esta irá fazer o acesso à base de dados, onde irão estar guardados todos os regtos e ocorrências inseridos pelos utilizadores.

3 Gestão de Equipa

Carlos Gaspar 2018288869 - Responsável pela gestão de equipa

3.1 Tarefa 1

Nesta etapa definimos qual o papel de cada elemento a desempenhar ao longo do primeiro percurso deste projeto, definindo assim cada tarefa e deadlines associados a cada pessoa.

Nesta fase do projeto deverão ser apenas implementadas as seguintes funcionalidades:

- Comunicação por sockets entre as aplicações constituintes do sistema.
- Gestão de credenciais para que apenas as pessoas autorizadas possam utilizar as respetivas aplicações.

Garantindo assim que as 4 aplicações possam interagir, apresentando os primeiros menus de interface com o utilizador e suportando as credenciais por estes inseridas.

Este projeto está dividido em 3 etapas diferentes. Nesta primeira etapa, o grupo é composto pelos seguintes elementos:

ELEMENTO	NÚMERO
Carlos Miguel Mendes Gaspar	2018288869
Mara Rafaela Oliveira Teixeira	2017259686
José Luís Dias	2018278568
Diogo Seabra Mota Henriques de Gouveia	2018279473
Diogo Emanuel Ribeiro Cruz	2018306709
Débora Sofia Valério Taborda de Seiça	2018278107
Ágata Palma	2009020534
Mafalda Amaro	2018281636

Figure 3.1: Tabela de elementos para Tarefa 1

Distribuindo os cargos pelos membros da equipa, originou a seguinte colocação:

Gestor de equipa: Carlos Miguel Mendes Gaspar

Gestor de cliente: Débora Sofia Valério Taborda de Seiça

Gestor de software: Diogo Emanuel Ribeiro Cruz

Gestor de riscos e testes: Mafalda Amaro

Gestor de qualidade: José Luís Dias

Equipa de desenvolvimento:

Ágata Palma

Diogo Gouveia

Mara Teixeira

As tarefas concretizadas durante esta etapa, e os seus tempos de realização, podem ser consultadas no seguinte diagrama:

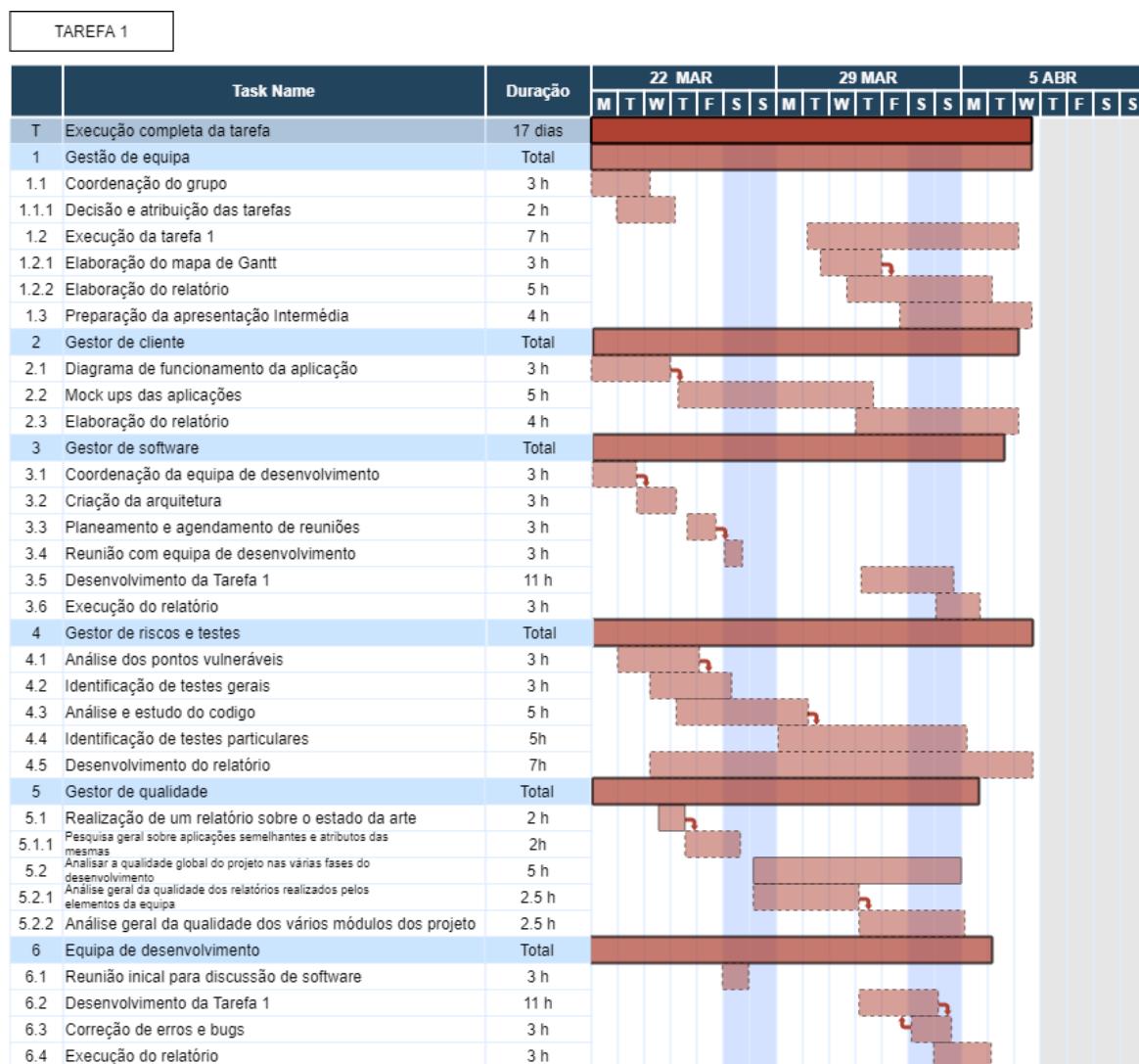


Figure 3.2: Diagrama de Gantt - Tarefa 1

3.2 Tarefa 2

Numa segunda etapa do projeto, a equipa é dividida em 2 equipas com igual número de elementos (4 a cada), que serão respetivamente:

EQUIPA 1

ELEMENTO	NÚMERO
Carlos Miguel Mendes Gaspar	2018288869
Mara Rafaela Oliveira Teixeira	2017259686
Diogo Emanuel Ribeiro Cruz	2018306709
Ágata Palma	2009020534

EQUIPA 2

ELEMENTO	NÚMERO
José Luís Dias	2018278568
Diogo Seabra Mota Henriques de Gouveia	2018279473
Débora Sofia Valério Taborda de Seiça	2018278107
Mafalda Amaro	2018281636

Figure 3.3: Tabelas de elementos da Equipa 1 e Equipa 2 para Tarefa 2

Nesta etapa, procedeu-se à seguinte atribuição de cargos, com base nas funções exercidas durante a *Tarefa 1*:

Equipa 1	Equipa 2
Gestor de equipa e do software	
Carlos Gaspar	Mafalda Amaro
Gestor de cliente, qualidade, riscos e testes	
Mara Teixeira	José Luís
Equipa de programadores	
Ágata Palma	Débora Seiça
Diogo Cruz	Diogo Gouveia

Figure 3.4: Tabela de cargos atribuídos para Tarefa 2

Durante esta fase do projeto, as tarefas vão estar distribuídos da seguinte forma:

Gestores de equipa e software:

- Coordenar o grupo de trabalho e as tarefas referentes aos seus elementos.
- Controlar, e/ou ajustar o planeamento geral do grupo e dos seus integrantes para a *Tarefa 2* e *Tarefa 3*.
- Responsáveis pela apresentação do trabalho feito ao longo da *Tarefa 2*.

Gestor de cliente, qualidade, riscos e testes:

- Interface com o cliente
- Coordenação dos relatórios
- Garantir uma qualidade global dos vários módulos do projeto
- Responsável pela gestão de testes e pelo plano de riscos e mitigação.

Equipa de programadores:

- Desenvolver o software necessário para esta etapa

Com base no primeiro diagrama, é possível fazer uma previsão do tempo e das funções que esta tarefas vão exigir, com isto, apresentamos uma antevista do diagrama de Gantt, que se referem à próxima etapa deste projeto:

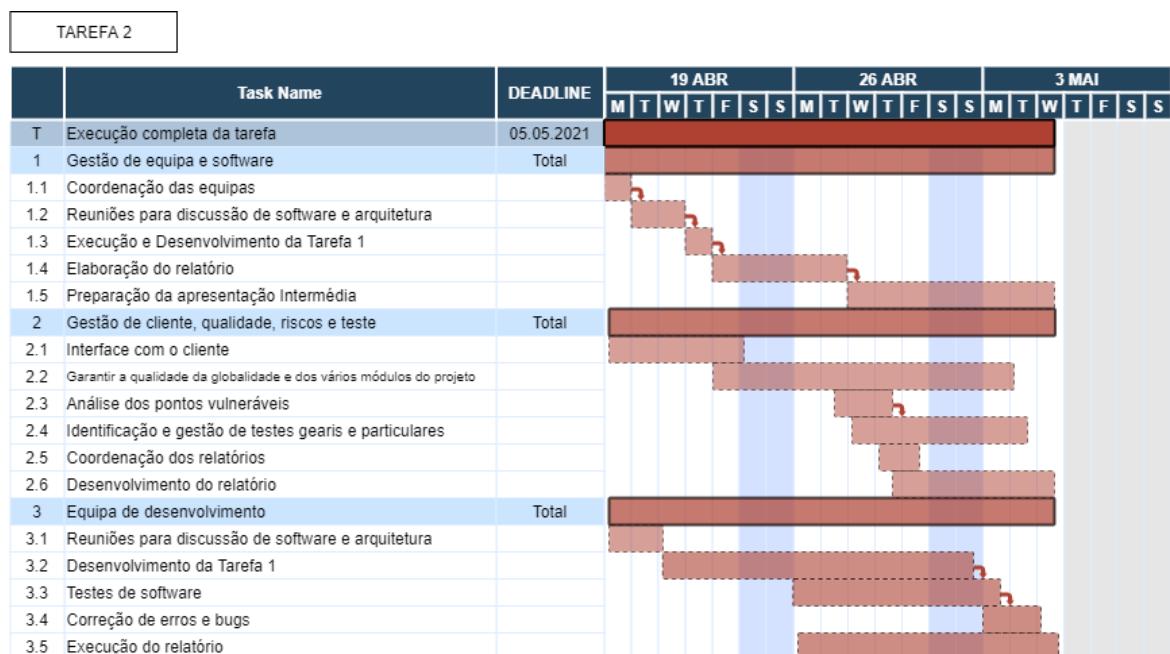


Figure 3.5: Diagrama de Gantt - Tarefa 2

3.3 Tarefa 3

Nesta última fase do projeto, *Tarefa 3*, as equipas separar-se-ão em grupos de trabalho constituídos por 2 elementos, que serão os seguintes:

- Carlos Gaspar
- Ágata Palma
- Mara Teixeira
- Diogo Cruz
- Débora Seiça
- José Luís
- Mafalda Amaro
- Diogo Gouveia

Onde nesta fase o projeto não terá cargos objetivos, uma vez que ambos estarão a projetar e desenvolver esta terceira etapa em conjunto. A distribuição de trabalho para esta etapa pode ser descrita no seguinte diagrama de Gantt:

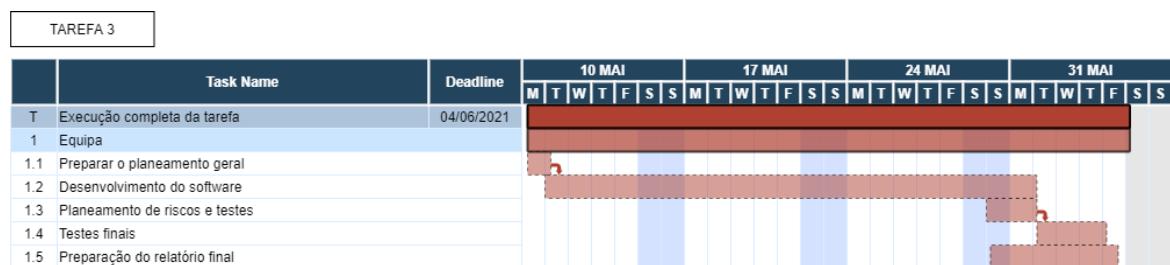


Figure 3.6: Diagrama de Gantt - Tarefa 3

4 Gestão de cliente

Débora Seiça - Responsável pela gestão de cliente

Esta fase do projeto iniciou-se com a implementação de um diagrama de funcionamento, de forma a prever a estrutura das aplicações. De seguida, criaram-se mock-ups, com o intuito de simular o design gráfico de cada aplicação.

4.1 Mock-ups

4.1.1 Login - Sign Up

As três aplicações começam por pedir as credenciais ao utilizador, podendo este fazer “Sign up”, se ainda não tiver conta registada, ou, se já tiver, “Login”.

The image displays two separate wireframe mock-ups of user interface forms, each enclosed in a light gray rectangular frame. Both forms have a pink header bar at the top. The top form is labeled "Sign up" and contains two input fields: "E-mail:" and "Password:", followed by a red "Submit" button. The bottom form is labeled "Login" and also contains "E-mail:" and "Password:" input fields, along with a "Confirm" button and a smaller "Sign up" link below it.

4.1.2 Aplicação para o Profissional de Saúde (APS)

Após aceder à aplicação com as credenciais, o utilizador é deparado com um menu principal:



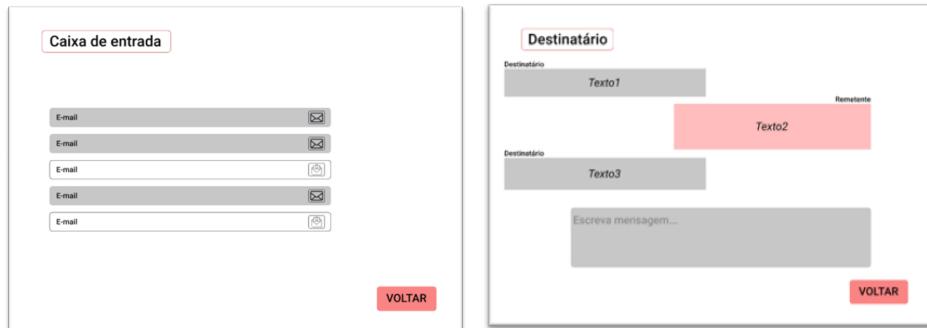
1. Ao selecionar “CONTA”, o utilizador é direcionado para uma página onde pode editar a sua conta, incluindo a password:



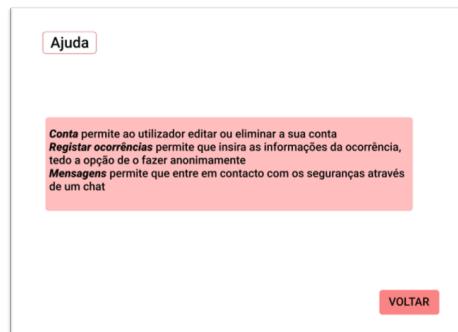
2. Ao selecionar “REGISTRAR OCORRÊNCIAS”, é direcionado para uma página onde pode inserir as informações da ocorrência:



3. Ao selecionar “MENSAGENS”, é direcionado para uma caixa de entrada, onde pode selecionar as mensagens que quer ver e, depois de selecionadas, abre um chat:



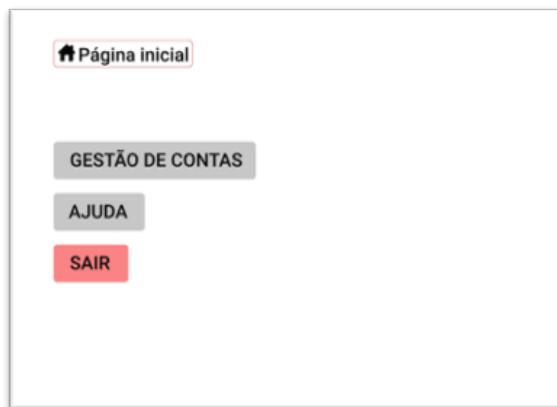
4. Ao selecionar “AJUDA”, é direcionado para uma página com as informações da aplicação:



5. Na página inicial também se encontra um botão de emergência que permite enviar uma mensagem de socorro aos seguranças.

4.1.3 Aplicação de Gestor de Sistema(AGS)

Após aceder à aplicação com as credenciais, o utilizador é deparado com um menu principal:



1. Ao selecionar “GESTÃO DE CONTAS”, o administrador é direcionado para uma página onde consegue ver a informação das contas dos utilizadores, conseguindo eliminá-las ou aprová-las, com as que faltam aprovar em destaque, podendo filtrá-las por e-mail, pelas “por aprovar”, ou ver todas:

The image consists of two side-by-side screenshots of a web application. The left screenshot, titled 'Gestão de contas', displays a list of user accounts. One account, 'E-mail' with ID 'x', is highlighted in red. The right screenshot, titled 'Conta', shows a detailed view of this account with the ID 'x'. It includes a button labeled 'Aprovar' (Approve) and another labeled 'Eliminar' (Delete).

2. Ao selecionar “AJUDA”, é direcionado para uma página com as informações da aplicação:

The image shows a single screenshot of a help page. The title is 'Ajuda'. The content area contains a pink box with the text: 'Esta aplicação permite ao administrador gerir as contas dos utilizadores, conseguindo ver a sua informação, podendo eliminá-las ou aprová-las.' Below this is a 'VOLTAR' (Return) button.

4.1.4 Aplicação para o Agente de Segurança (AAS)

Após aceder à aplicação com as credenciais, o utilizador é deparado com um menu principal:

The image shows a main menu for the AAS application. The top bar has a home icon and the text 'Página inicial'. Below it is a vertical list of menu items: 'CONTA', 'CONSULTAR OCORRÊNCIAS', 'MENSAGENS', 'AJUDA', and 'SAIR'. The 'SAIR' button is highlighted with a red background.

1. Ao selecionar “CONTA”, o utilizador é direcionado para uma página onde pode editar a sua conta, incluindo a password:

Conta

Informações de conta

E-mail: x
Password

Editar Eliminar

VOLTAR

2. Ao selecionar “CONSULTAR OCORRÊNCIAS”, é direcionado para uma página onde pode encontrar todos os registos de ocorrências feitos pelos profissionais de saúde, conseguindo filtrá-los por nome, local e data, e ver os seus detalhes. Nessa página encontram-se também, em destaque, os registos de emergência:

Ocorrências

ALERTAS

REGISTOS

VOLTAR

Registo de ocorrências

Para registrar a ocorrência de violência necessitamos das seguintes informações:

Name: _____ Androíme:

Data: _____ Hora: _____

Local: _____ Tipo de Agressão: _____

VOLTAR

3. Ao selecionar “MENSAGENS”, é direcionado para um menu:

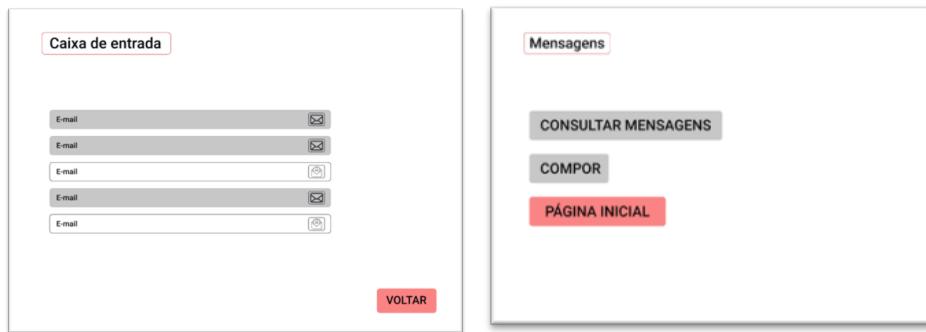
Mensagens

CONSULTAR MENSAGENS

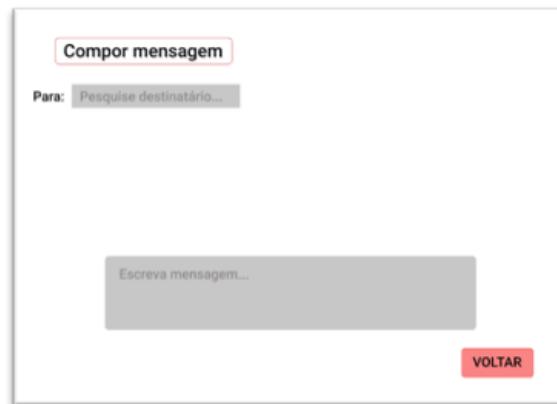
COMPOR

PÁGINA INICIAL

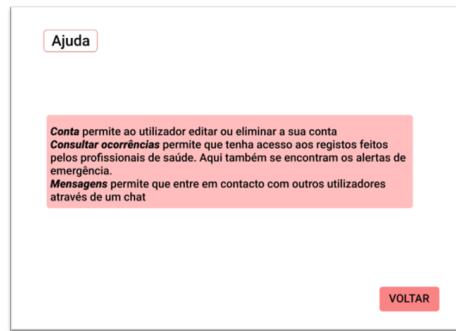
- a. Ao selecionar “CONSULTAR MENSAGENS”, é direcionado para uma caixa de entrada, onde pode selecionar as mensagens que quer ver e, depois de selecionadas, abre um chat:



- b. Ao selecionar “COMPOR”, é direcionado para uma página onde pode escrever uma mensagem para um destinatário à escolha:



4. Ao selecionar “AJUDA”, é direcionado para uma página com as informações da aplicação:



4.2 Diagrama de Funcionamento

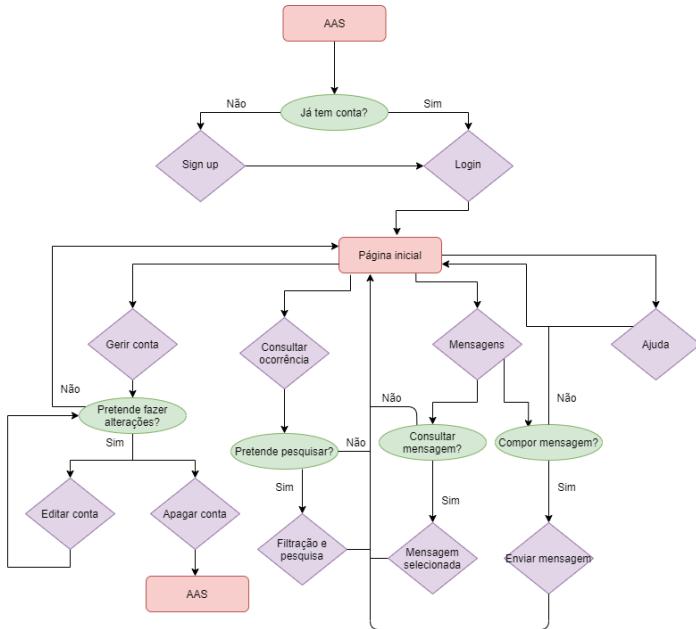


Figure 4.1: Aplicação para o Agente de Segurança

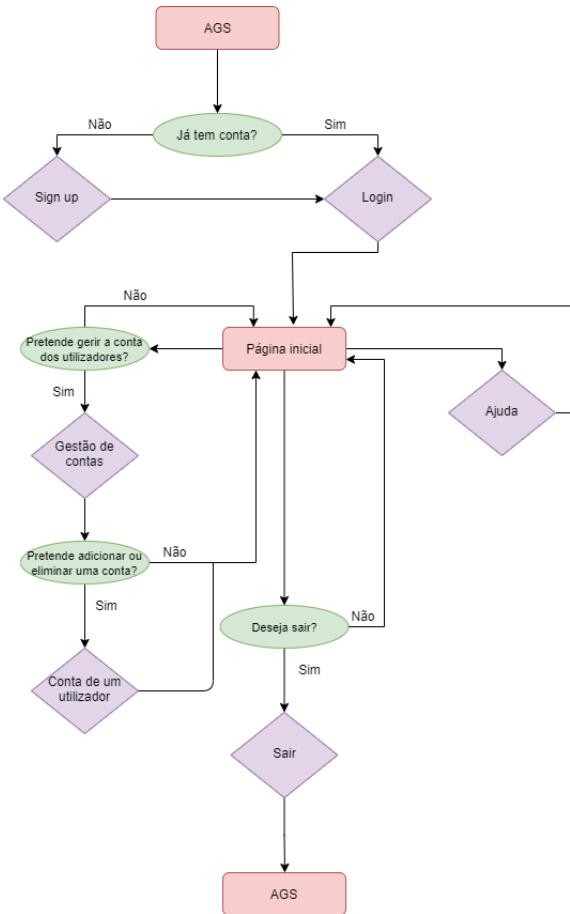


Figure 4.2: Aplicação de Gestor do Sistema

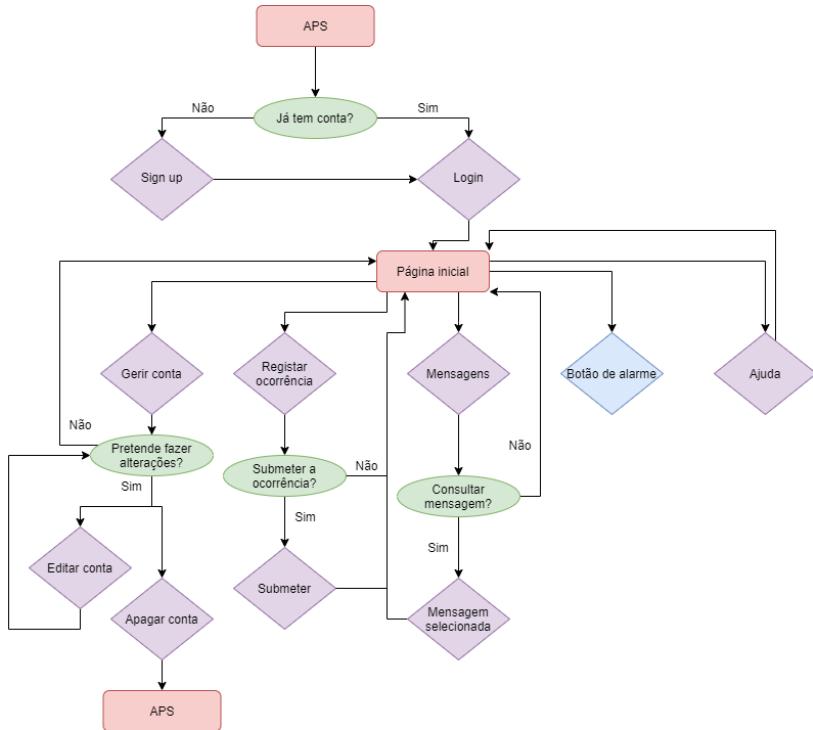


Figure 4.3: Aplicação para o Profissional de Saúde

5 Gestão de Software

Diogo Cruz - Responsável pela gestão de software

O gestor de software ficou responsável por definir as arquitecturas de software e criar os seus templates iniciais a utilizar pela equipa, coordenar a equipa de desenvolvimento e fornecer suporte no desenvolvimento do código.

5.1 Softwares

Organização: Trello

Repositório de versões: Github

Programação: Python3 e SQL

Base de dados: Postgresql

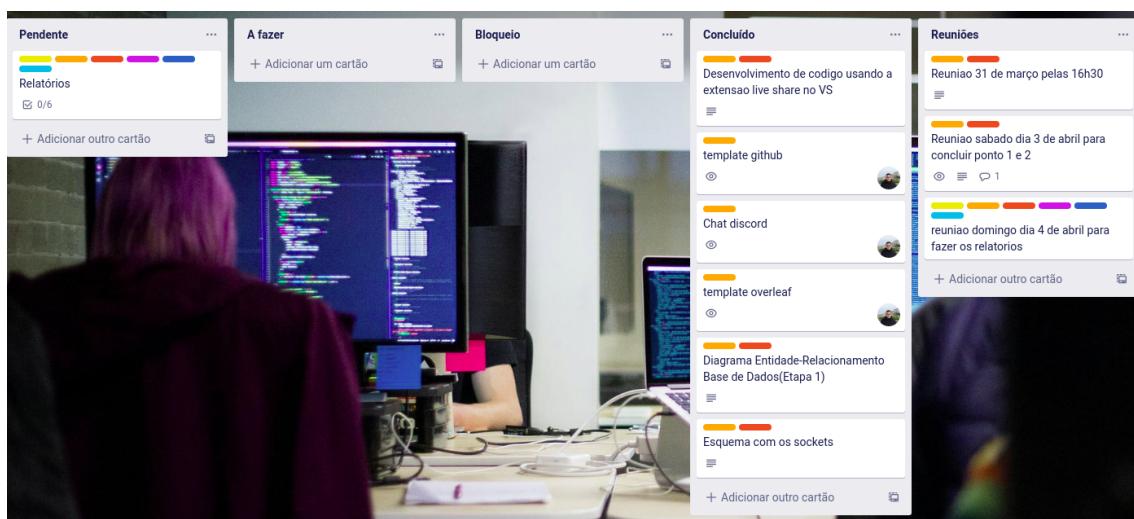
Diagrama ER: onda.dei.uc.pt

Chat e outras infos: Discord

Relatórios: Overleaf

5.1.1 Trello

Utilizamos esta ferramenta para lembrete de reuniões e divisão de tarefas. Segue-se imagem ilustrativa:



Os cartões estão separados pelo estado da tarefa, excepto o ultimo que serve como lembrete das reuniões, que vão desde o pendente até ao concluído. O bloqueio acontece quando a tarefa depende também de entidades exteriores. Relativamente às etiquetas com várias cores, estas correspondem a cada pelouro da equipa. Por exemplo, o laranja é para o pelouro do desenvolvimento, o vermelho para o gestor de software, etc.

5.1.2 Github

O Github foi utilizado para gestão e repositório de versões do código, em que toda a equipa tem acesso. Segue-se imagem ilustrativa:

The screenshot shows a GitHub repository page for 'DiogoCruz40 / Project_RC'. The repository is private. The main interface includes navigation tabs for Code, Issues, Pull requests, Actions, Projects, Wiki, Security, Insights, and Settings. Below these are buttons for Go to file, Add file, and Code dropdown. The code section displays a list of 17 commits from 'DiogoCruz40' made 19 hours ago, with details like file changes and commit messages. The README.md section contains the project's name and a brief description of its purpose and tools used.

```

DiogoCruz40 / Project_RC Private

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags Go to file Add file Code

DiogoCruz40 Todo a funcionar 639da2f 19 hours ago 17 commits
__pycache__ Tudo a funcionar 19 hours ago
.gitignore Update .gitignore 4 days ago
1_etapa.json adicionado diagrama ER,esquema e detalhes ao readm 4 days ago
1_etapa.png adicionado diagrama ER,esquema e detalhes ao readm 4 days ago
Client.py Tudo a funcionar 19 hours ago
Etapa-1.docx etapa 1 doc gestor de equipa 4 days ago
Health_Professional.py Tudo a funcionar 19 hours ago
Proj_RC_2020-2021.pdf adicionado diagrama ER,esquema e detalhes ao readm 4 days ago
README.md adicionado breaks 4 days ago
Security_Officer.py Tudo a funcionar 19 hours ago
Socket_Esquema.png adicionado diagrama ER,esquema e detalhes ao readm 4 days ago
System_Manager.py Tudo a funcionar 19 hours ago
bd.txt Tudo a funcionar 19 hours ago
server.py Tudo a funcionar 19 hours ago

README.md
project_rc

Projeto RC
Softwares:
Repositório de versões - GitHub
Chat e outras Infos - Discord
Relatórios - Overleaf
Organização - Trello
Programação - Python e SQL
Base de dados - pgadmin4(Postgresql)
Diagrama ER - onda.dei.uc.pt

```

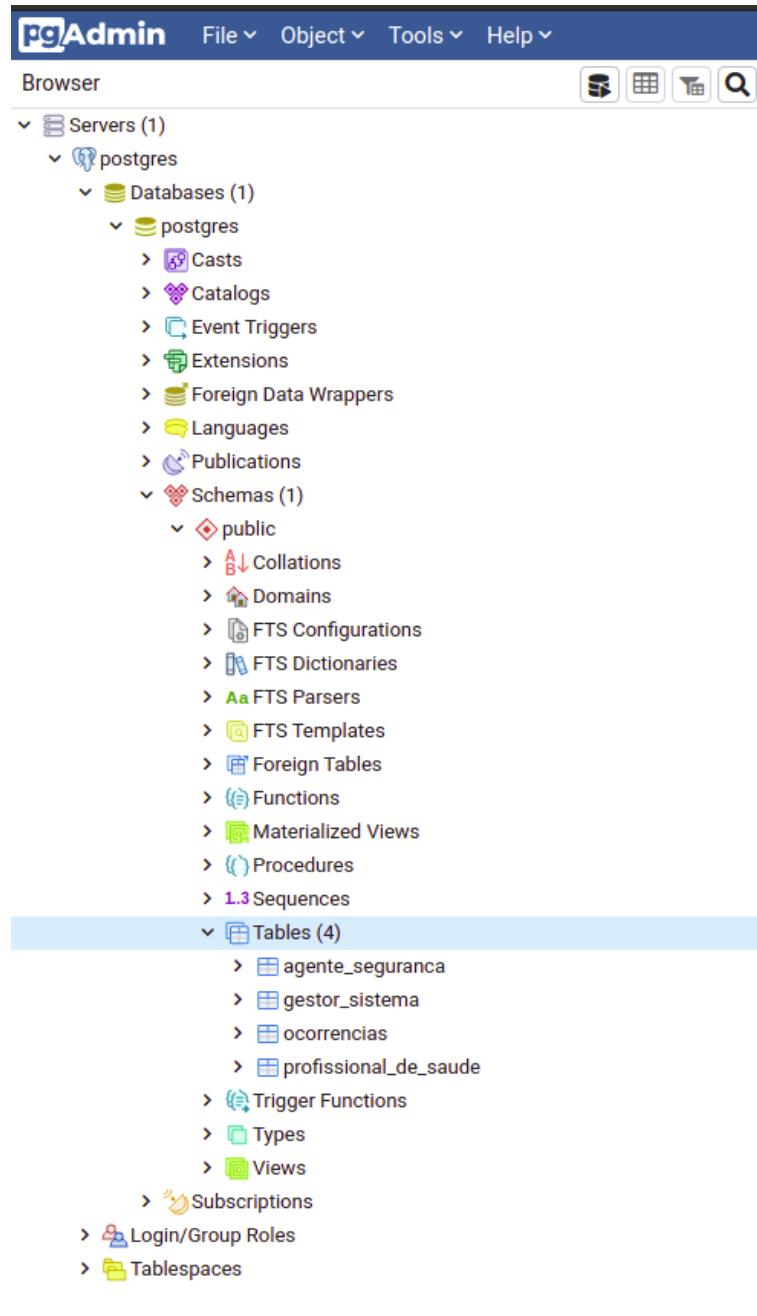
Isto é uma ferramenta essencial para qualquer programador, visto que reduz o risco de corrupção do código. Uma grande vantagem desta plataforma é a possibilidade de trabalhar paralelamente em diferentes "features" da aplicação com recurso à funcionalidade de branching.

5.1.3 Python3 e SQL

Estas linguagens permitem criar uma aplicação dinâmica, segura e eficiente. Sendo que actualmente são muito utilizadas e requisitadas no mercado. Foi utilizado python no desenvolvimento da interface e estabelecimento de sockets. Para a comunicação com a base de dados e gestão da mesma, foi utilizado SQL.

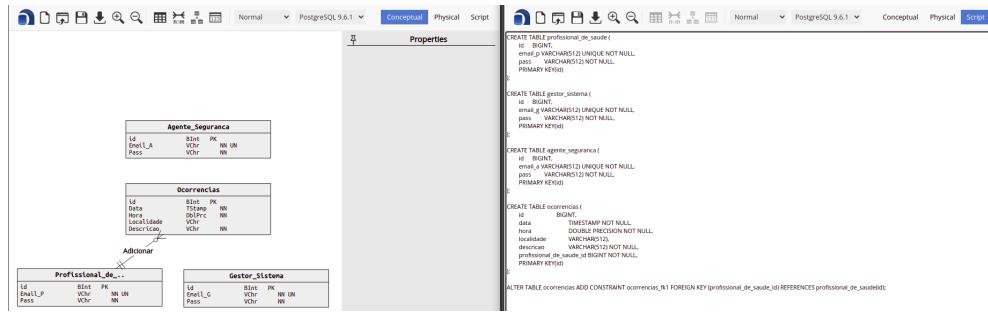
5.1.4 Postgresql

Utilizámos a ferramenta pgAdmin4 para gerir a base de dados (open-source) PostgreSQL. A nossa base de dados encontra-se alojada nesta plataforma. Segue-se imagem ilustrativa:



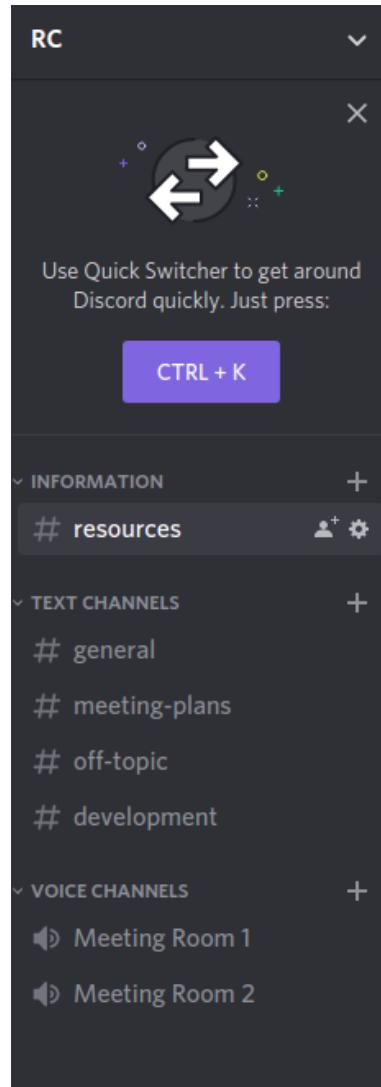
5.1.5 Onda

Utilizamos esta plataforma para criarmos o diagrama entidade relacionamento da nossa base de dados, onde isto nos permite gerar um ficheiro script automaticamente em linguagem sql para colocar no pgadmin4. Segue-se imagem ilustrativa:



5.1.6 Discord

Utilizamos o Discord para comunicar entre nós, marcar reuniões, partilhar informações, entre outros. Segue-se imagem ilustrativa:



5.1.7 Overleaf

Para escrita dos relatórios utilizamos o overleaf, que usa a linguagem de programação L^AT_EX, onde podemos escrever todos em simultâneo no mesmo ficheiro. Foi organizado por capítulos, do 1 ao 8, em que cada capítulo corresponde a um determinado pelouro. É tudo compilado dentro de um ficheiro principal e possui histórico de versões, permitindo assim que nenhum trabalho seja perdido. Por fim exporta-se como PDF. Segue-se imagem ilustrativa:

The screenshot shows the Overleaf project interface. At the top, there are navigation buttons (Menu, F, A, J, Revisar, Compartilhar, Submit, Historico, Bate-papo) and a download button (Download do projeto nessa versão). Below the header, there's a breadcrumb trail: Navegando projeto como 4th Apr 2021, 8:47 pm. The main area displays the code of the main.tex file, which includes imports for geometry, a4paper, amsmath, graphicx, tabular, booktabs, hyperref, titlesec, and color packages. It also defines document class, page style, and various sections (1.1 to 1.8). On the right side, there's a sidebar titled 'Projeto RC' showing a list of files: 1.tex, 2.tex, 3.tex, 4.tex, 5.tex, 6.tex, 7.tex, 8.tex, and pictures. Below this, the 'Historico' section lists recent edits: Projeto RC/Chapters/5.t (Edited at 8:47 pm), Projeto RC/Chapters/7.t (Edited at 8:41 pm), Projeto RC/Chapters/5.t (Edited at 8:41 pm), and Projeto RC/pictures/ (Renamed at 8:39 pm). At the bottom of the sidebar, it says 'Criado Projeto RC/pictures/Captura de ecrã de 2021-04-04 20-39-02.png - Projeto RC/picture s/Discur.png'.

```
1 \documentclass[12pt]{report}
2 \usepackage{a4paper, top=25mm, bottom=25mm, left=25mm, right=25mm, bindingoffset=<0mm>}[geometry]
3 \usepackage{amsmath}
4 \usepackage{graphicx}
5 \usepackage{tabular}
6 \usepackage{booktabs}
7 \usepackage{hyperref}
8 \usepackage{titlesec, color}
9 \usepackage{listings}
10 \usepackage{tikz}
11 \usepackage{pgfplots}[tikz]
12 \usepackage{float}
13 \usepackage{etoolbox}[ifthen]
14 \usepackage{chaperone}[chaperone]
15 \usepackage{etoolbox}[etoolbox]
16 \usepackage{titlesec}[titlesec]
17 \usepackage{titlesec}[titlesec]
18 \usepackage{secnumdepth}[secnumdepth]
19 \usepackage{tocify}[tocify]
20 \usepackage{tocify}[control the TOC formatting]
21 \usepackage{etoolbox}[etoolbox]
22 \setlength{\parindent}{1em}
23 \usepackage{fancyhdr}[fancyhdr]
24 \usepackage{fancyhf}[fancyhf]
25 \usepackage{fancyhf}[fancyhf]
26 \fancyhf{} % Clear all header and footer fields
27 \renewcommand{\headrulewidth}{0pt}
28 \renewcommand{\headrulewidth}{0pt}
29 \usepackage{subcaption}[subcaption]
30 \usepackage{caption}[caption]
31 \usepackage{raggedright}[raggedright]
32 \usepackage{raggedright}[raggedright]
33 \usepackage{raggedright}[raggedright]
34 \usepackage{raggedright}[raggedright]
35 \usepackage{pdf, pdf, for}
36 \usepackage{toc, section, standard, chapter}
```

6 Gestão de Riscos e Testes

Mafalda Amaro - Responsável pela gestão de riscos e testes

Nesta secção, o desenvolvimento do projeto é analisado de forma a identificar os seus pontos vulneráveis e refletir sobre estes, de forma a reduzir a probabilidade da sua ocorrência. Para além de antecipar os pontos vulneráveis, esta secção engloba a análise de testes mais adequados a realizar ao software desenvolvido.

6.1 Análise de pontos vulneráveis

A análise dos pontos mais vulneráveis corresponde aos riscos a que o projeto está sujeito. É um conjunto de acontecimentos que desequilibram o funcionamento e desenvolvimento de um projeto. Associados ao projeto a ser desenvolvido, e adaptável às três etapas de desenvolvimento, são identificados os seguintes pontos:

1. Incumprimento da data de entrega do projeto;
2. Ao longo do desenvolvimento do projeto encontrar uma falha no planeamento que altere a estrutura inicialmente prevista para o projeto;
3. Um ou múltiplos elementos integrantes do grupo desistirem da realização do projeto;
4. Perda de informação desenvolvida, ao eliminar indevidamente informação importante;
5. Tarefas dependentes em cadeia que dão origem a atrasos sucessivos;
6. Avarias nas plataformas de desenvolvimento dos elementos da equipa;
7. Mudanças/reestruturações devidas a riscos que não tenham sido anteriormente identificados.

Ao identificar a ocorrência de um risco é essencial recolher informação sobre a sua natureza à medida a que esta se torna disponível. É necessário reanalizar o risco e priorizar em função da sua relevância. Os riscos identificados têm prioridades distintas ao longo do desenvolvimento do projeto. É então, indispensável, ao longo do desenvolvimento do projeto, analisar as prioridades dos riscos e reavaliar as suas consequências.

O risco 1, o incumprimento do prazo de entrega definido, é adaptável ás três etapas do projeto e ás suas respetivas datas de entrega. Este tipo de riscos pode ter origem numa conjugação de causas. Como tal, é necessário avaliar detalhadamente as causas e reajustar recursos de forma a reduzir os atrasos gerados. É então, um risco latente que deve ser evitado, monitorizando os restantes riscos.

O risco 2 surge ao longo da evolução do projeto e está associado a erros no planeamento do seu desenvolvimento. Pode surgir devido a de múltiplos acontecimentos, sendo identificados como mais prováveis, erros no planeamento da estrutura do software e/ou erros na estrutura do planeamento das etapas do projeto. Para evitar e reduzir a probabilidade de ocorrem erros associados ao risco 2 é indispensável um planeamento exaustivo. Este planeamento deverá considerar múltiplas abordagens, simulações do projeto e identificar possíveis incompatibilidades.

O risco 3 engloba o risco de, ao longo de todo o desenvolvimento, um ou vários elementos pertencentes à equipa do projeto renunciarem do seu cargo. Apesar das causas associadas à ocorrência do risco 3 serem maioritariamente exteriores, é essencial desenvolver um plano estratégico para colocar em prática, na sua eventualidade. Como estratégia para reduzir os efeitos do risco em questão, deve ser elaborado um plano de realocação de trabalho. Deve(m) ser identificado(s) o(s) elemento(s) da equipa mais adequando para substituir as funções do elemento que abandona o seu cargo, de modo a encontrar um novo equilíbrio na distribuição de tarefas.

O risco 4, a perda de informação desenvolvida, aumenta com o numero de indivíduos envolvidos na manipulação de documentos importantes. Para evitar a perda de informação, todos os membros da equipa, devem saber como manipular corretamente as plataformas utilizadas. E toda a informação deve ser salvaguardada em múltiplas plataformas, incluindo os computadores individuais dos membros da equipa.

O risco 5 está associado a atrasos no desenvolvimento. No planeamento das diferentes tarefas a desenvolver, é inevitável a existência de dependências entre tarefas. Ao ocorrer um atraso numa das tarefas a ser desenvolvida, terá como consequência o atraso nas tarefas dependentes. Para quebrar a cadeia de atrasos deverá, mais uma vez, ser implementado um plano de redistribuição de funções para reduzir o seu impacto.

O risco 6, a avaria de plataformas utilizadas no desenvolvimento do projeto, envolve fatores fora do controlo dos membros da equipa. No entanto, devem ser planeados mecanismos de reação no caso da sua ocorrência. Estes devem envolver, mais uma vez, realocação de funções entre os membros da equipa de forma a que o impacto seja reduzido.

O risco 7 envolve riscos não previsto no planeamento. Na eventualidade de surgir um risco nesta categoria este deverá ser estudado e avaliado de modo a obter um plano de reação. O plano deverá ter como objetivo final reestabelecer o bom funcionamento do projeto.

6.2 Análise de testes

A análise de testes tem como propósito verificar se o programa desenvolvido realiza as operações requeridas e identificar defeitos, para que estes sejam corrigidos. É importante realçar que os testes efetuados têm o objetivo de identificar possíveis erros, e não demonstrar a ausência destes. Para tal os seguintes testes devem ser desenvolvidos:

1. Analisar o código fonte;
2. Testar todas as funcionalidades pretendidas;
3. Desenvolver inputs que podem originar problemas no programa;
4. Analisar o design.

Antes de começar a avaliar o programa através dos testes definidos, durante o desenvolvimento do software, duas questões devem ficar verificadas. Se o programa a ser desenvolvido coincide com o estabelecido no enunciado do projeto. E se o produto está a ser desenvolvido corretamente e eficientemente. De seguida os restantes testes devem ser realizados.

O teste 1, que engloba a análise do código fonte, deve ser efetuado anteriormente a todos os restantes testes. É essencial ter conhecimento de todo o código desenvolvido e de todo o seu funcionamento.

O teste 2 tem como objetivo testar as funcionalidades do programa e verificar se estas estão todas operacionais. É imprescindível realizar, no mínimo, um teste redirecionado para cada uma das funcionalidades.

O teste 3 pretende, através de combinações de diferentes inputs, não esperados numa utilização correta do programa, tornar o sistema mais robusto.

O teste 4 é uma avaliação do design verificando o seu bom funcionamento e compatibilidade com o requerido.

De seguida é apresentado uma adaptação dos testes a realizar a cada uma das etapas:

Etapa 1

A etapa 1 requer as funcionalidades F1 e F2, que são, respetivamente, a comunicação por sockets entre as aplicações do sistema e a gestão de credenciais para a utilização das aplicações.

Todos os testes mencionados (1-4) devem ser realizados. Os testes 1 e 4 não têm especificações distintas para as diferentes funcionalidades. Aplicando diretamente às funcionalidades F1 e F2 deve ser realizado o seguinte:

Tarefa F1

Para o teste 2 a comunicação entre as aplicações tem de ser verificada. Considerando que a Aplicação para o Profissional de Saúde (APS), a Aplicação de Gestor do Sistema (AGS) e a Aplicação para o Agente de Segurança (AAS) comunicam apenas com a Aplicação Central (AC), para verificar a conexão entre a AC e as restantes aplicações, ao executar o código do cliente, cada uma das 3 aplicações deve ser testada. Para tal ao iniciar o código do cliente e ao selecionar qualquer uma das três aplicações cliente, APS, AGS e AAS, a informação de conexão da aplicação correspondente deve ser visualizada no terminal onde o código do server é executado.

Por exemplo, ao selecionar a APS, no terminal onde o código cliente é executado, deverá gerar, no terminal onde o código server é executado, a mensagem de “[NEW CONNECTION] - APS”. Um processo semelhante deve ser efetuado para a AGS e para a AAS.

O teste 3 não tem especificações para a tarefa F1.

Tarefa F2

Para testar a funcionalidade F2, as credenciais de utilização de cada uma das aplicações deve ser testada. Para tal, ao executar o código dos clientes, e selecionando a aplicação a utilizar, o registo e o login de cada uma das aplicações deve testada. Para um funcionamento correto apenas as credenciais registadas e autorizadas devem ser aceites pelo programa.

Por exemplo, ao efetuar o registo, numa das aplicações, este só deve ser bem-sucedido se o email não tiver sido anteriormente registado na base de dados. O login só deverá ser bem-sucedido se as credenciais estiverem registadas na base de dados.

Para o teste 3, mais testes que não devem ser autenticados devem ser verificados. Por exemplo, trocar algumas letras das autenticações corretas e verificar a recusa dessas modificações quando realizar o login nas aplicações. Apenas devem ser autorizados registos com email válido.

Etapa 2

A etapa 2 requer as funcionalidades F3, F4, F5, F6 e F7, que são, respetivamente, a gestão dos seus regtos por parte do profissional de saúde e agente de segurança, gestão das contas registadas nas aplicações por parte do administrador do sistema, o registo de crimes por parte do profissional de saúde e a consulta dos crimes registados, com ou sem filtros, por parte dos agentes de segurança. Todos os testes mencionados (1-4) devem ser realizados. Os testes 1 e 4 não têm especificações distintas para as diferentes funcionalidades. Aplicando diretamente às funcionalidades da etapa 2 deve ser realizado o seguinte:

Tarefa F3 e F4

Para o teste 2 deve ser criado um perfil para um profissional de saúde. E verificar que foi criado como o suposto. De seguida alterar-se esse perfil e verificar a mudança. Posteriormente eliminar-se o perfil e, mais uma vez, verificar as modificações.

Para o teste 3, com um email já utilizado num registo, tentar criar uma nova conta, que deverá produzir uma mensagem de erro. O mesmo deve acontecer se o utilizador tentar eliminar um registo não existente.

Tarefa F5

Para testar as funcções da AGS, deve conseguir-se validar ou eliminar o registo de um profissional de saúde e dos agentes de segurança. Para tal, a APS deve ser iniciada e efetuar um registo de uma nova conta. Com a AGS deverá ser possível visualizar o pedido de validação de registo, efetuado pelo profissional de saúde. Na AGS os pedidos de novos registos devem aparecer como notificação, que é possível aceitar ou recusar. Se o gestor aceitar o registo, deve ser possível realizar o login com as credenciais aceites. Se o gestor eliminar o registo, as credenciais não devem ser aceites ao efetuar o login. O mesmo procedimento deve ser repetido para a AAS.

Esta funcionalidade não requer inputs não standards.

Tarefa F6

Para testar as funcionalidades de F6, ao entrar na APS, depois da autenticação, ao registar na plataforma uma ocorrência de um crime, este deve ser registado pela AC na base de dados.

Para testar a resistência a inputs os diferentes campos necessitam de testes distintos. O campo da data deverá aceitar apenas datas válidas, por exemplo, deverá ser recusado o registo com uma data localizada no futuro. Também deve recusar datas com o campo do mês fora dos valores entre 1 a 12 ou com o dia superior a 31 e inferior a 1. O campo da hora deverá apenas ser validado com valores entre de 0 a 23 na hora e de 0 a 59 nos minutos. O email inserido apenas deverá ser aceite se fizer parte dos registes validados pela AGS.

Tarefa F7

Para testar as funcionalidades de F7, ao efetuar o registo na AAS, depois do processo de autenticação, todos os registos da base de dados devem ser visualizados. Ao aplicar filtros apenas os dados do respetivo filtro devem ser visualizados.

Esta funcionalidade não requer inputs não standards.

Etapa 3

A etapa 3 requer as funcionalidades F8, F9, F10, F11 e F12, que são, respetivamente, a possibilidade de o profissional de saúde efetuar um registo anónimo, a funcionalidade do botão de alarme, o sistema de apoio às funcionalidades das aplicações, o chat e todas as funcionalidades adicionais.

Todos os testes mencionados (1-4) devem ser realizados. Os testes 1 e 4 não têm especificações distintas para as diferentes funcionalidades. Aplicando diretamente às funcionalidades da etapa 3 deve ser realizado o seguinte:

Tarefa F8

Para testar o funcionamento de um registo anónimo deve ser efetuado uma simulação de um registo com o campo do email em anónimo, através da APS. De seguida consultar a informação do registo e verificar a ausência de uma associação entre o registo do crime e o profissional de saúde. Para tal, ao consultar o crime na AAS, o campo do email, do crime registado, deverá aparecer como o email anónimo.

Esta funcionalidade não requer inputs não standards.

Tarefa F9

Para verificar o funcionamento do botão de alarme na APS deve ser simulada uma situação de teste. Para tal, ao aceder à APS deve ser acionado o protocolo do botão de alarme, acionando o mesmo. De seguida, com a AAS devidamente iniciada, uma notificação consequente ao acionar o “botão de alarme” deve ser apresentada no ecrã.

Tarefa F10

Com o objetivo de testar o sistema de apoio o teste seguinte deve ser aplicado a todas as aplicações envolvidas. Depois de realizada a autenticação, ao introduzir o comando “help”, uma mensagem de ajuda, com a explicação das respetivas aplicações, deverá aparecer no ecrã.

Para testar múltiplos inputs, palavras semelhantes a “help”, devem ser introduzidas. Como “jelp”, “helç” ou “hekp”. Todas estas devem gerar mensagens de erro.

Tarefa F11

Para testar o chat entre a vitima e o agente de segurança, ambas as aplicações (APS e AAS) devem ser devidamente iniciadas, tal como a Aplicação Central. E ao enviar mensagens de uma das aplicações a outra deverá receber. A mensagem deve ser recebida sem erros de transmissão de caracteres.

Tarefa F12

Todas as funcionalidades dentro da categoria F12 devem ser testadas e apresentar comportamentos previstos. A tolerância a inputs deverá ser testada, em semelhança das tarefas anteriores.

7 Gestão de Qualidade

José Dias - Responsável pela gestão de qualidade

7.1 Estado da arte

Como proposto pelo professor, foi efetuada uma pesquisa sobre o estado da arte em relação a sistemas de registo, alerta e prevenção de ocorrências. Estes sistemas, não sendo idênticos àquele que vai ser desenvolvido, apresentam recursos e ferramentas interessantes e com potencial para serem implementadas e/ou melhoradas.

7.1.1 Ankira

A **Ankira** é uma plataforma portuguesa de gestão de ocorrências em lares de idosos. Esta plataforma utiliza um design simples e utilitário, com uma tipologia básica para uniformizar a linguagem. O registo de ocorrências é efectuado por um profissional já registado. Este pode inserir a data e a hora da ocorrência assim como o tipo, observações e associar outras ocorrências do mesmo paciente, caso existam. O registo é efetuado como mostra a figura.

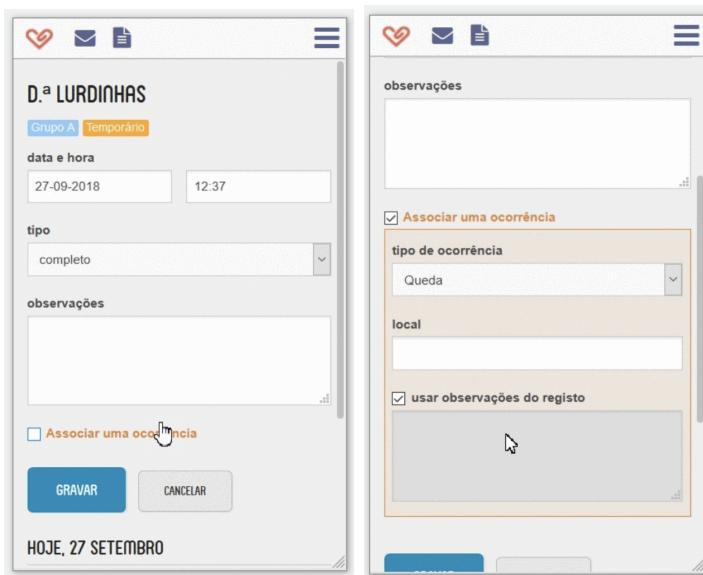


Figure 7.1: Registo de ocorrências

No processo de registo, cada paciente é atribuído a um profissional responsável dependendo do seu caso, por exemplo, um paciente que sofreu uma queda é atribuído a um enfermeiro. Esta plataforma conta também com um sistema de alarmes automáticos, para o acompanhamento dos pacientes. Nesta plataforma está presente um sistema

de segurança e confidencialidade, tal que o caso de cada paciente é só exposto a cada profissional indicado.

Por fim, podemos contar com uma avaliação contínua dos registo efetuados, realizando relatórios sobre o número de casos ao longo do tempo, as tendências, situações anómalas e até onde atuar para obter melhorias. A figura demonstra o exemplo de um gráfico dos registo de quedas no último ano.

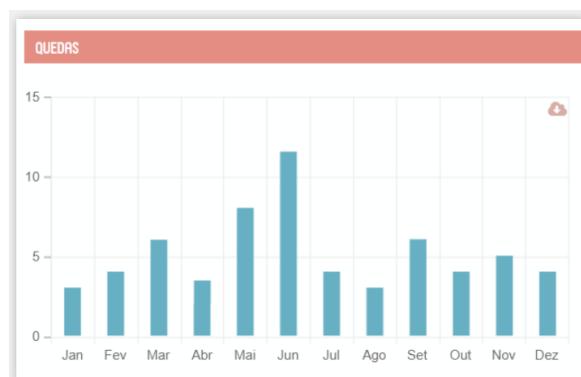


Figure 7.2: Gráfico do número de quedas no último ano

7.1.2 Callisto

Callisto é uma plataforma de registo e prevenção de ocorrências de violência sexual norte americana. Esta plataforma utiliza um design minimalista e simples para apelar a jovens. É possível reportar e descrever o acontecimento tanto por pessoas individuais ou por comunidades.



Figure 7.3: Quem pode registar ocorrências.

Nesta aplicação é também usado um **Matching System**, cujo objetivo é detetar quando a mesma pessoa é reportada por várias vítimas.

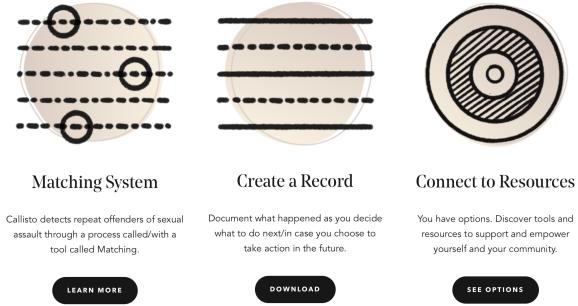


Figure 7.4: Registo e ”Matching”.

Existem também várias ferramentas que incentivam a ajuda aos outros e o tipo de ações que a vítima deve tomar, dependendo do caso reportado.



Figure 7.5: Restantes recursos do sistema.

7.1.3 Conclusão

No fim da apresentação destas duas plataformas de registo e alerta de ocorrências, denoto algumas funcionalidades interessantes e possíveis de implementar. Entre elas, o **Matching System** da aplicação **Callisto**, pois conta com a possibilidade de registrar quando o mesmo agressor é registado por várias pessoas, no nosso caso profissionais, possibilitando um tratamento mais adequado.

No lado da **Ankira**, o sistema de alerta é um recurso ótimo, pois a sua implementação tornará possível um agendamento por parte da aplicação, ajudando significativamente o trabalho dos profissionais.

7.2 Qualidade Geral do Projeto

Nesta secção irá ser gerida e planeada a qualidade geral do projeto desenvolvido. Para tal, daqui para a frente, vão ser definidas as técnicas, requisitos, padrões, procedimentos e práticas que apontarão para garantir, controlar e melhorar a qualidade geral do projeto.

7.2.1 Plano geral da qualidade

- **Padrões, procedimentos e práticas:** Durante todo o desenvolvimento do projeto irá proceder-se a uma revisão e gestão da qualidade. No fim de cada etapa, o trabalho desenvolvido por cada membro da equipa (desenvolvedores e gestores) deverá ser submetido para ser revisto e testado pelos restantes membros. Todo o trabalho deverá ser desenvolvido com relativa antecedência, de modo a presevar estas práticas e a qualidade do projeto.
- **Objetivos:** Em cada etapa os objetivos projetados serão ligeiramente diferentes dos objetivos da etapa anterior, progredindo para a finalização do desenvolvimento do sistema indicado. No final de cada etapa é necessário obter todo o trabalho aprovado nas fases de teste e revisão. O objetivo será obter todas as aplicações e funcionalidades, previstas para cada etapa, concluídas e a funcionar como planeado.
- **Papéis e Responsabilidades:** Em cada etapa o gestor responsável por cada campo deverá submeter o seu relatório para revisão. Esta será efetuada pelos restantes membros da equipa. Os desenvolvedores deverão, à semelhança dos gestores, submeter o código para ser testado. O gestor de qualidade terá a responsabilidade de garantir que todo o trabalho desenvolvido é revisto, testado e otimizado.

7.2.2 Controlo e melhoria da qualidade:

Para controlar a qualidade, todos os procedimentos definidos anteriormente deverão ser cumpridos. Em todas as etapas deve-se verificar se o projeto final será cumprido, ou seja, se o desenvolvimento realizado até à data levará à finalização do projeto, na data prevista, e com as funcionalidades anunciadas pela equipa. Se tal não acontecer, os planos definidos deverão ser revistos e, em último caso, reestruturados de modo a melhorar e otimizar o desenvolvimento.

Todos os resultados das atividades realizadas deverão ser registados e monitorizados, de modo a mitigar erros futuros assim como erros recorrentes. Todas as situações anómalas deverão ser tratadas com a maior brevidade possível, pois estas podem levar a um acumular de situações não benéfico para o desenvolvimento do projeto. Novos padrões e procedimentos poderão, e deverão, ser adicionados ao longo do desenvolvimento, abrindo a possibilidade de lidar com uma panóplia de situações que ao início não foram discutidas ou planeadas.

8 Equipa de desenvolvimento

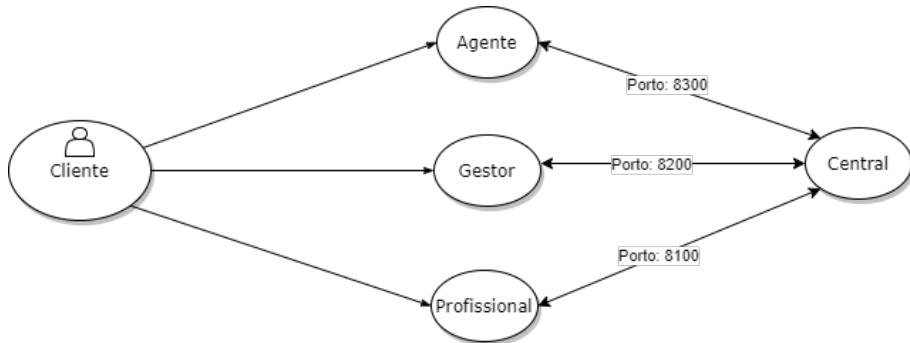
Ágata Palma; Diogo Gouveia; Mara Teixeira - Responsáveis pelo desenvolvimento

8.1 Estrutura

Relativamente ao desenvolvimento do programa, utilizámos **Python 3** como linguagem de programação, juntamente com uma base de dados em **PostgreSQL**.

Nesta etapa desenvolveram-se duas aplicações principais, a **Aplicação Central** (AC) que funciona como Servidor e a **Aplicação Cliente**, esta subdividiu-se em três aplicações específicas relativamente aos três intervenientes: a **Aplicação para o Profissional de Saúde** (APS), a **Aplicação para o Gestor de Sistema** (AGS) e a **Aplicação para o Agente de Segurança** (AAS).

Para a interligação e comunicação entre as aplicações, nomeadamente entre as aplicações APS, AGS e AAS com a Aplicação Central foram estabelecidos **Sockets TCP** essenciais para o correto funcionamento do projeto. Na elaboração do código recorreu-se à função **fork()** e a **threads()**.



8.1.1 Aplicação Central

O Servidor depois de ser executado cria três processos em que cada um estabelece uma ligação socket do tipo TCP com um porto diferente para cada tipo de utilizador e fica num estado de espera até uma conexão surgir. Assim que haja uma conexão, o Servidor cria uma nova thread dentro do processo respetivo ao tipo de utilizador. De seguida, o Servidor recebe as credenciais introduzidas pelo utilizador no registo e no login, fazendo a autenticação com recurso à base de dados. De modo a garantir o máximo de segurança, apenas a Aplicação Central pode aceder à base de dados, sendo estas ligações abertas apenas quando necessário, fechando-as logo após a sua utilização.

8.1.2 Aplicação Cliente e Redireccionamento

Para uso futuro e para efeitos de registo de atividade na aplicação, foi criada uma tabela na base de dados, possibilitando o armazenamento das ocorrências quando efectuadas por qualquer cliente registado como profissional de saúde

Esta aplicação servirá para redireccionar o ”cliente” para a sua respectiva sub-aplicação - Profissional de Saúde, Gestor de Sistema ou Agente de Segurança. Quando o cliente efectua o registo, os seus dados (e-mail e password(encriptada)) ficam armazenados na base de dados, sendo deste modo simples a recuperação destes registo para efeitos de login e registos de actividade na aplicação. É ainda garantido em todos os momentos de inserção de dados, que estes são válidos e inexistentes na base de dados(nomeadamente os e-mails).

De modo a permitir uma filtragem de dados mais rápida, os registos estão separados na base de dados por tipo de cliente, sendo esta divisão feita no momento do registo. Em cada uma destas sub-aplicações, foi implementado o processo de registo e login com encriptação da password, sendo esta informação passada através de sockets TCP para a aplicação central (servidor). De modo a garantir o máximo de segurança, nenhuma destas aplicações - Cliente e sub-aplicações - podem aceder à base de dados, sendo todas as credenciais enviadas pelo servidor para o respectivo cliente.

8.1.3 Código de desenvolvimento de software

<https://we.tl/t-ok2mDoDnzb>