

FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

MESTRADO INTEGRADO EM ENGENHARIA ELECTROTÉCNICA E DE
COMPUTADORES

Sistema de Combate à Violência

REDES DE COMPUTADORES

Diogo Cruz - 2018306709

e

Mara Teixeira - 2017259686



04/2021

Contents

| | |
|--|-----------|
| Sistema de Combate à Violência | I |
| 1 Introdução | 1 |
| 2 Objetivos do trabalho | 2 |
| 3 Gestão de equipa e do software | 3 |
| 3.0.1 Tarefa 3 | 3 |
| 3.1 Gestão de Software | 3 |
| 3.1.1 Softwares | 3 |
| 3.1.2 Descrição do Projeto | 5 |
| 4 Gestão de cliente, qualidade, riscos e testes | 7 |
| 4.1 Gestão de cliente | 7 |
| 4.1.1 Mock-ups | 7 |
| 4.1.2 Diagramas de Funcionamento | 11 |
| 4.2 Gestão de qualidade | 13 |
| 4.2.1 Estado da arte | 13 |
| 4.2.2 Qualidade Geral do Projeto | 15 |
| 4.3 Gestão de riscos e testes | 17 |
| 4.3.1 Análise de pontos vulneráveis | 17 |
| 4.3.2 Análise de testes | 18 |
| 5 Equipa de desenvolvimento | 21 |
| 5.1 Etapa 1 | 21 |
| 5.1.1 Aplicações e funcionamento do programa | 21 |
| 5.2 Etapa 2 | 22 |
| 5.2.1 Tarefa F3 e F4 | 22 |
| 5.2.2 Tarefa F5 | 22 |
| 5.2.3 Tarefa F6 e F8 | 22 |
| 5.2.4 Tarefa F7 | 23 |
| 5.3 Etapa 3 | 23 |
| 5.3.1 Tarefa F9 | 23 |
| 5.3.2 Tarefa F10 | 24 |
| 5.3.3 Tarefa F11 | 24 |
| 5.3.4 Tarefa F12 | 24 |

1 Introdução

Na sequência da realização das Tarefas 1 e 2 foi realizada a última tarefa, a Tarefa 3, que deu o corpo final ao projeto final do Sistema de Combate à Violência proposto na cadeira Redes de Computadores. Ao longo do processo e desenvolvimento deste projeto o grau de exigência aumentou e permitiu-nos colocar em prática e consolidar conhecimentos adquiridos no decorrer do semestre. Neste relatório iremos descrever o projeto na íntegra, incidindo especialmente nas funcionalidades requeridas na terceira etapa deste projeto.

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 registos e ocorrências inseridos pelos utilizadores.

3 Gestão de equipa e do software

3.0.1 Tarefa 3

Como definido nas tarefas anteriores, a equipa para o desenvolvimento desta etapa é composta por 2 elementos.

- Diogo Cruz - 2018306709 e Mara Teixeira - 2017259686

Nesta fase do projeto não há cargos objetivos, uma vez que ambos projetamos e desenvolvemos esta terceira etapa em conjunto. A distribuição de trabalho para esta etapa pode ser descrita no seguinte diagrama de Gantt:

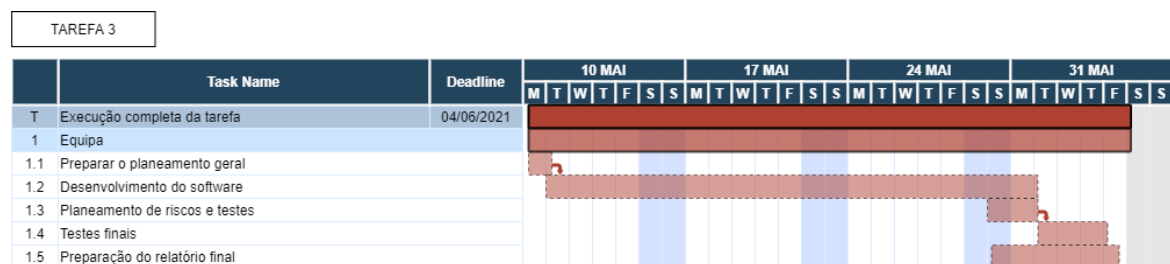


Figure 3.1: Diagrama de Gantt - Tarefa 3

3.1 Gestão de Software

Na qualidade de gestor de software, optámos por recorrer às plataformas já utilizadas para o desenvolvimento das Tarefas 1 e 2, uma vez que já estaríamos familiarizados com os ambientes de trabalho. Estas foram usadas para suplementar o trabalho da equipa, tanto para distribuição de tarefas como para organização da mesma.

3.1.1 Softwares

Repositório de versões: Github

Programação: Python3 e SQL

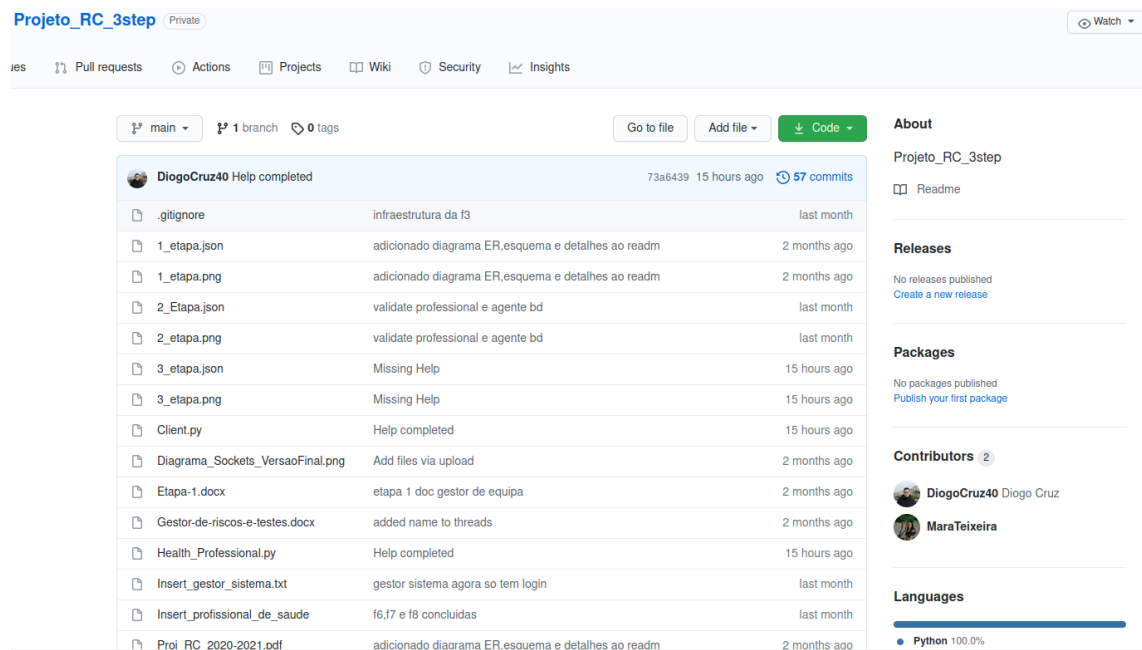
Base de dados: Postgresql

Diagrama ER: onda.dei.uc.pt

Relatórios: Overleaf

Github

O GitHub foi utilizado para gestão e repositório de versões de código, pois esta reduz o risco de corrupção de código tendo ainda a possibilidade de trabalhar paralelamente em diferentes "features" da aplicação com recurso à funcionalidade de branching.



Python3 e SQL

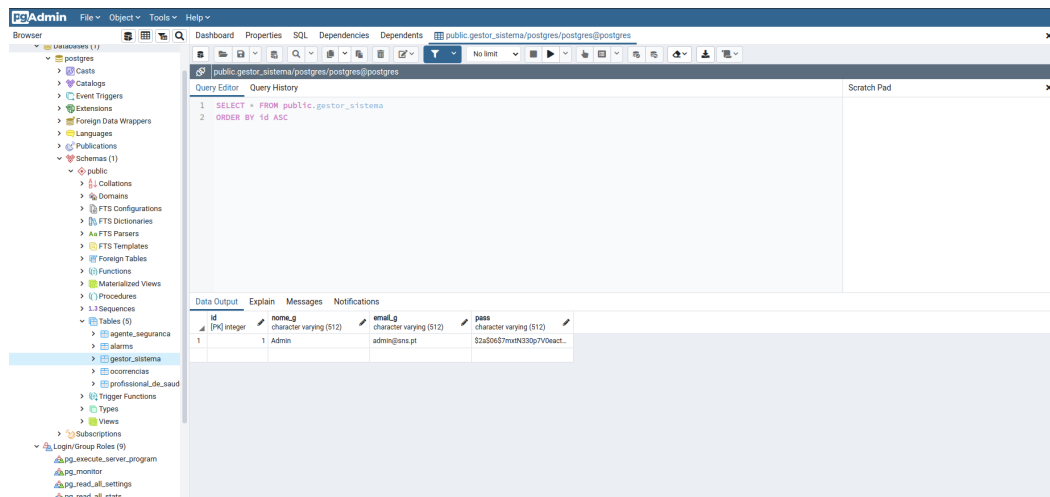
Para o desenvolvimento da interface e estabelecimento de sockets foi utilizado o python, e, para a comunicação com a base de dados e gestão da mesma, foi utilizada a linguagem SQL. Estas linguagens permitem criar uma aplicação dinâmica, segura e eficiente.

Onda

De forma a poder fazer alterações na arquitetura da nossa base de dados, para o desenvolvimento das funcionalidades pedidas na segunda etapa deste projeto utilizamos a ferramenta Onda, uma vez que esta nos permite gerar um script em código SQL para colocar no pgadmin.

Postgresql

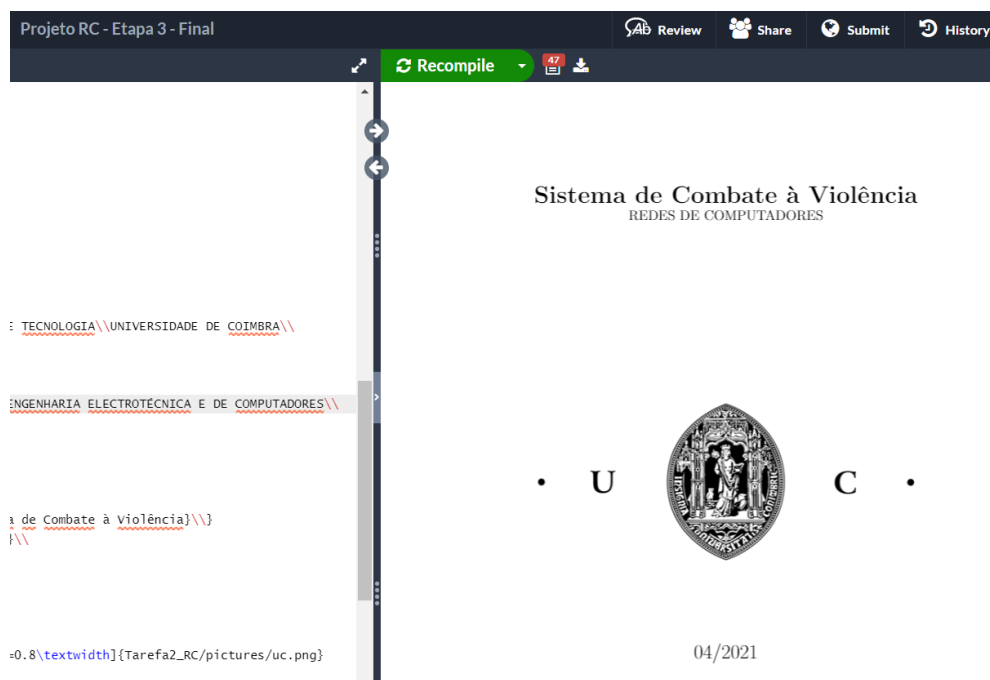
Utilizámos a ferramenta pgAdmin4 para gerir a base de dados (open-source) PostgreSQL. A nossa base de dados encontra-se alocada nesta plataforma.



Overleaf

Para escrever e simplificar a junção dos relatórios decidimos utilizar o Overleaf, que usa a linguagem de programação Latex, pois permite escrita simultânea, e possui ainda a ferramenta de histórico de versões, permitindo assim que nenhum trabalho seja perdido.

Segue-se uma imagem ilustrativa:



3.1.2 Descrição do Projeto

Com a implementação das funcionalidades restantes do projeto, ou seja, as funcionalidades 9, 10 e 12 (sendo que a funcionalidade 8 foi realizada na etapa anterior por interesse do grupo) este programa permite agora que:

Todos os utilizadores desta aplicação tenham um sistema de credenciais de acesso à plataforma guardadas numa base de dados.

Qualquer profissional de saúde, depois de devidamente autenticado na aplicação do profissional de saúde, por um sistema de Registo/Login, possam agora gerir o seu registo, isto é, fazer alterações às suas credenciais e apagar o seu registo. A aplicação permite também registar qualquer ocorrência de violência. Estas ocorrências têm a seguinte estrutura:

Data; Hora; Local; Tipo de Agressão; Nome

Estas ocorrência, quando introduzida na base de dados, podem estar devidamente identificada pelo profissional de saúde que a reportou, ou, se por escolha do mesmo, registadas de forma anónima.

Do mesmo modo, qualquer agente de segurança, depois de seguir o mesmo processo de Registo/Login, pode também gerir as suas credenciais de registo. O mesmo pode ainda consultar todas as ocorrências de violência reportadas pelos utilizadores, tendo ainda a possibilidade de aplicar filtros na pesquisa das ocorrências, nomeadamente data, local e pessoa.

Agora é também possível a um profissional de saúde apertar um "botão de alarme" que sinaliza um crime em tempo real, sendo este alarme imediatamente redirecionado para todos os agentes de segurança que, rapidamente, conseguem ir em auxílio da vítima.

Para além disto, nas duas aplicações relativas ao profissional de saúde e ao agente de segurança, em todos os menus existe uma opção "help" que explica detalhadamente cada opção do respetivo menu e o seu significado, facilitando assim a compreensão da aplicação pelo usuário.

A Aplicação de Gestor de Sistema possibilita a um gestor, depois de devidamente autenticada pelo sistema de Login, tendo um username e uma password destinada apenas ao próprio, consultar e validar qualquer novo registo por parte dos profissionais de saúde e dos agentes de segurança.

4 Gestão de cliente, qualidade, riscos e testes

4.1 Gestão de cliente

Neste ponto, tal como nas etapas anteriores, foram criados mockups e diagramas de funcionamento com o intuito de simular o design gráfico de cada aplicação tornando assim a sua leitura e uso mais clara para os utilizadores.

4.1.1 Mock-ups

Login - Sign Up

Neste projeto existem três utilizadores distintos: o **Profissional de Saúde**, o **Agente de Segurança** e o **Gestor de Sistema**.

Qualquer que seja o utilizador a iniciar a aplicação, este é dirigido para a página inicial com cinco opções, das quais três dizem respeito ao tipo de utilizador, a quarta opção destina-se a um sistema de apoio relativa a este menu em específico e a última opção é para sair do sistema.

9:41

9:41

SELECT

1. Health Professional
2. Security Officer
3. System Manager
4. Help
5. Exit

Select: ____

Assim que o utilizador avance para a opção referente à sua profissão é encaminhado para o menu particular, sendo que no caso do Profissional de Saúde e do Agente de

Segurança o menu é igual, já o Gestor de Sistema não terá a opção para criar conta, uma vez que as suas credenciais estão predefinidas no sistema.



Aplicação para o Profissional de Saúde (APS)

Após a inserção correta das credenciais pelo Profissional de saúde, este é direcionado para a aplicação específica em que são apresentadas seis possibilidades: criar uma ocorrência, alteração do perfil, apagar a conta, botão de alarme, sistema de apoio ("help") ou sair.

Dependendo da escolha inserida pelo Profissional de Saúde, este é encaminhado para diferentes menus. Caso a seleção seja "alterar perfil" o utilizador tem a hipótese de mudar o email, a password ou o nome da sua conta. Se o utilizador pretender registar uma ocorrência este terá que preencher diversos campos, tais como: a data, hora, local e descrição; por fim, quando todos os parâmetros estiverem preenchidos poderá submeter a ocorrência. Se, por alguma razão, a sua intenção não for o registo de uma ocorrência poderá voltar para página anterior. No caso da escolha da opção "Alarme!" o utilizador é reencaminhado para uma página que lhe diz que o alarme foi acionado.

Hello
username,

1. Create occurrence
2. Change Profile
3. Erase Account
4. ALARM!
5. Help
6. Exit

Select: ____

Hello username,

-> Registo de ocorrências

1. Change email
2. Change password
3. Change name
4. Help
5. Exit

Select: ____

1. Registo da data
2. Registo da hora
3. Registo da localidade
4. Descrição da ocorrência
5. Submeter ocorrência
6. Help
7. Exit

Select: ____

Hello
username, YOU
PUSHED THE
ALARM
BUTTON

Press any key to
continue...

Aplicação de Gestor de Sistema(AGS)

Em relação ao Gestor de Sistema, este tem a responsabilidade de validar as contas de novos utilizadores e a opção de apagar registos existentes. Para isso, será questionado

qual o email respetivo à conta a validar ou apagar.

9:41

9:41

Hello Admin,

Hello Admin,

1. Validate an account

2. Delete an account

3. Exit

The email of the account
you want to validate:

mail@mail.mail

Select: _____

Account Validated!

Aplicação para o Agente de Segurança (AAS)

Na aplicação do Agente de Segurança, depois da introdução dos seus dados, as hipóteses de escolha são: consultar ocorrências, alterar o perfil, apagar a conta, consultar alarmes, o sistema de apoio ou sair.

No que diz respeito à alteração do perfil, é apresentado ao Agente de Segurança o mesmo que se sucede na aplicação do Profissional de Saúde anteriormente explicado. Na eventualidade da escolha de consulta de ocorrências é apresentada uma tabela composta por várias características dos eventos: "id" da ocorrência, data, hora, localidade, descrição, "id" do utilizador e nome do utilizador, sendo que todos os dados foram recolhidos no registo de ocorrências por parte do Profissional de Saúde em questão. Posto isto, o Agente ainda tem a opção de pesquisa por filtros, nomeadamente, pesquisa por data, por localidade, pelo nome do Profissional de Saúde ou mesmo por uma palavra nas descrições. No caso da consulta de alarmes, o Agente de Segurança poderá consultar todos os alarmes recentes, sendo que aparece informação sobre o estado do crime, permitindo ao agente ignorar os alarmes já correspondidos e agir em relação aos alarmes por responder, ao selecionar esta opção também terá informações sobre o profissional de saúde, bem como a sua localização.

Hello
username,

1. Consult occurrence
2. Change Profile
3. Erase Account
4. Consult Alarms
5. Help
6. Exit

Select: ____

Hello username,
-> Consulta de ocorrências

| Id Ocorrência | Data | Hora | Localidade | Descrição | Id Utilizador | Nome |
|---------------|----------|-------|------------|-----------|---------------|-------|
| 1 | 4/4/2021 | 12:30 | Columbo | ... | 14 | Ana |
| 2 | 3/4/2021 | 19:15 | Porto | ... | 23 | João |
| 3 | 4/4/2021 | 15:45 | Lisboa | ... | 17 | Maria |
| ... | ... | ... | ... | ... | ... | ... |

1. Pesquisar por palavra na descrição
2. Pesquisar por data
3. Pesquisar por localidade
4. Pesquisar por profissional de saúde
5. Help
6. Exit

Select: ____

Alarms

Alarme 1
(Profissional de saúde X) → Correspondido

Alarme 2
(Profissional de saúde Y) → Não Correspondido

Alarme 3
(Profissional de saúde Z) → Correspondido

4.1.2 Diagramas de Funcionamento

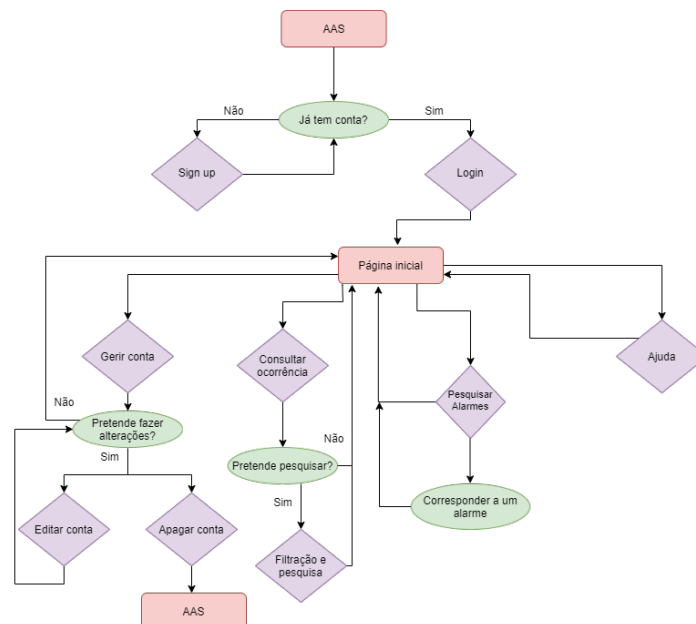


Figure 4.1: Aplicação para o Agente de Segurança

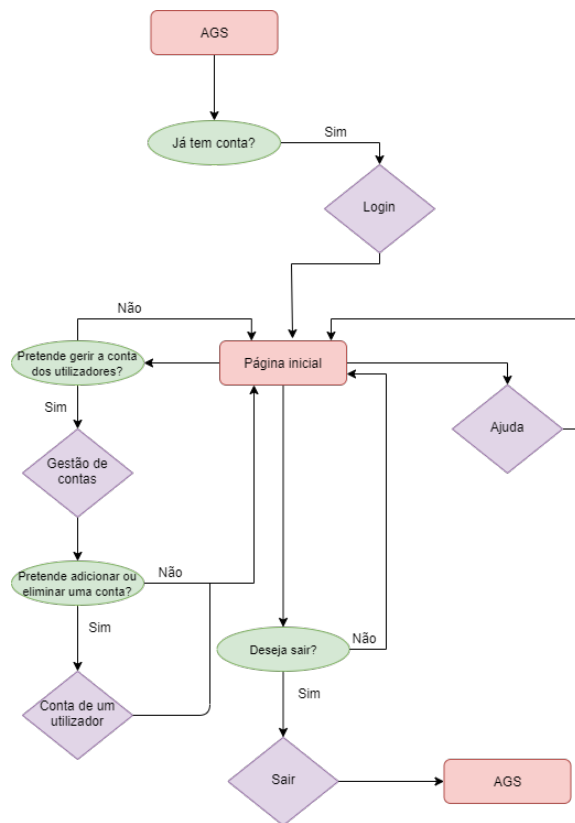


Figure 4.2: Aplicação de Gestor do Sistema

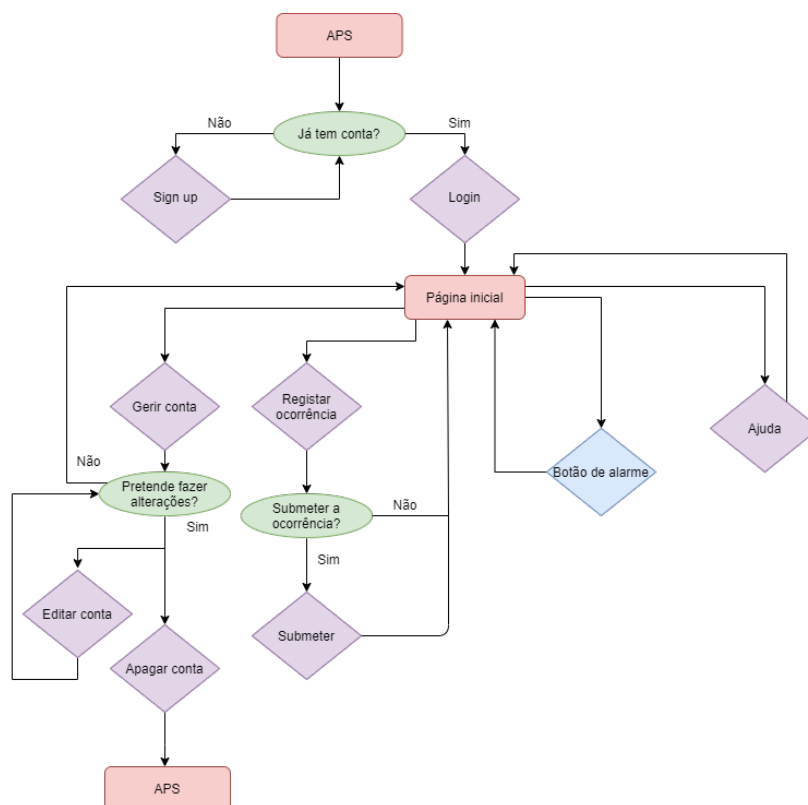


Figure 4.3: Aplicação para o Profissional de Saúde

4.2 Gestão de qualidade

4.2.1 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.

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.

The image displays two side-by-side screenshots of the Ankira mobile application interface. The left screenshot shows the main form for recording an incident. At the top, it identifies the patient as 'D.ª LURDINHAS' with buttons for 'Grupo A' and 'Temporário'. Below this, there are input fields for 'data e hora' (date and time), with '27-09-2018' and '12:37' entered. A 'tipo' (type) dropdown menu is set to 'completo'. There is a large text area for 'observações' (observations). Below the text area is a checkbox labeled 'Associar uma ocorrência' (Associate an occurrence) which is currently unchecked. At the bottom are 'GRAVAR' (Save) and 'CANCELAR' (Cancel) buttons, and a date indicator 'HOJE, 27 SETEMBRO'. The right screenshot shows a detailed view of the 'observações' section. It includes a checkbox 'Associar uma ocorrência' which is checked. Below it is a 'tipo de ocorrência' (type of occurrence) dropdown menu set to 'Queda' (Fall). There is a 'local' (location) input field. Another checkbox 'usar observações do registo' (use observations from the record) is checked. A large text area for additional notes is visible at the bottom of this section.

Figure 4.4: 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 registos 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 registros de quedas no último ano.

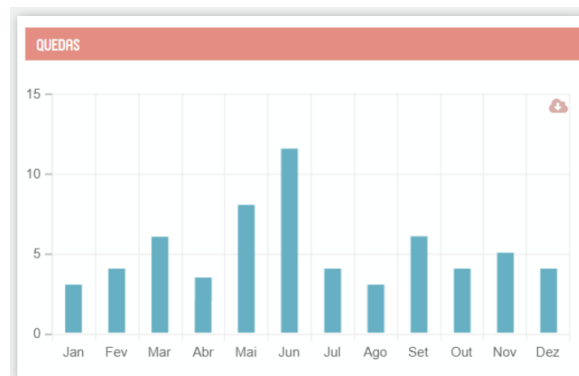


Figure 4.5: Gráfico do número de quedas no último ano

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 4.6: Quem pode registrar ocorrências.

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

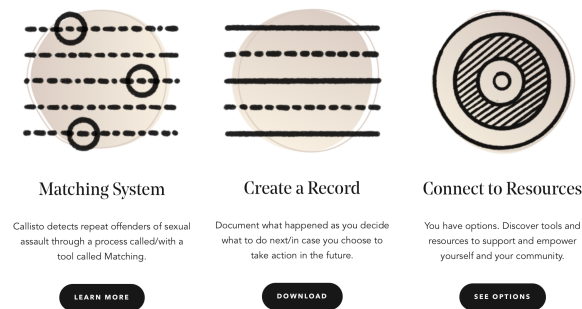


Figure 4.7: Registo e "Matching".

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

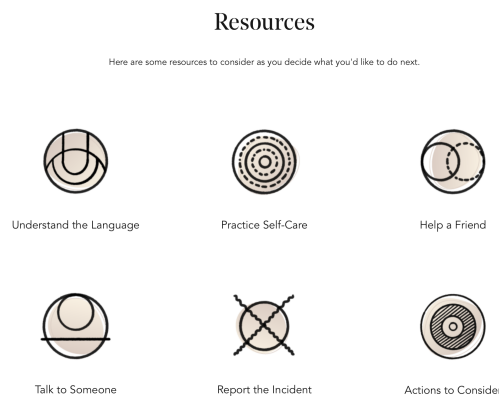


Figure 4.8: Restantes recursos do sistema.

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 registar 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.

4.2.2 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, procedimen-

tos e práticas que apontarão para garantir, controlar e melhorar a qualidade geral do projeto.

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 preservar 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.

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.

4.3 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.

4.3.1 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 reanalisar 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 às três etapas do projeto e às 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 ocorrerem 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.

4.3.2 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 restrições e proteções de 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:

Etapas

1. Etapas 1 e 2

Apesar das duas primeiras etapas já se encontrarem concluídas e testadas, é importante que tudo seja novamente revisto e que sejam feitos os diversos testes, para verificar se os requisitos de cada funcionalidade estão a operar corretamente, uma vez que foi implementada a transição para a interface gráfica, e sendo assim teremos de verificar se não houve corrupção ou perda de código. Para isso, devem ser executadas as funções correspondentes à Etapa 1: funcionalidades 1 e 2; e as funções realizadas na Etapa 2: funcionalidades 3, 4, 5, 6, 7 e 8.

2. Etapa 3

Concluindo o projeto e as três etapas e após o desenvolvimento de todo o software é necessário realizar os testes mencionados anteriormente (1-4).

Esta etapa, inicialmente, englobava três funcionalidades (funcionalidades 8, 9 e 10), no entanto como a funcionalidade 8, referente ao anonimato, já foi implementada e avaliada, restam as funcionalidades 9 e 10. Portanto, nesta etapa,

foram implementados sistemas de apoio para todos os menus nas aplicações do Profissional de Saúde e do Agente de Segurança e um sistema de alarme para casos de emergência.

Funcionalidade 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, não sendo relevante o menu em que o agente de segurança se encontra dentro da aplicação. Para esta funcionalidade é importante ter em conta a velocidade com que os agentes de segurança recebem o aviso, sendo que para isso é necessário otimizar e implementar um código funcional para que o processo seja em tempo real. Cada alarme acionado deverá conter informações sobre o profissional de saúde em perigo para facilitar o trabalho dos agentes e a sua rápida ação. Como vários agentes de segurança vão receber o alarme é também necessário criar um registo de alarmes em que cada agente de segurança deve avisar a que alarme está a corresponder, para que não haja vários agentes preocupados com a mesma situação e para evitar que existam alarmes que não sejam correspondidos.

Funcionalidade F10

Com o objetivo de testar o sistema de apoio em todas as aplicações envolvidas e em todos os menus existentes deve-se selecionar o número correspondente à opção “help” e verificar se funciona corretamente.

Funcionalidade F12

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

5 Equipa de desenvolvimento

5.1 Etapa 1

O trabalho desenvolvido na etapa passada deixou este projecto com ponto de partida no desenvolvimento de funcionalidades nas aplicações previamente criadas. Todo o programa foi desenvolvido em **Python** com recurso a comandos **SQL** para aceder à base de dados e de modo a permitir uma comunicação simultânea entre vários utilizadores com o servidor foi utilizada a função **fork()** e **threads()**.

5.1.1 Aplicações e funcionamento do programa

É importante esclarecer os desenvolvimentos passados de modo a compreender de que forma foram feitas as funcionalidades agora implementadas. Como tal, apresentamos um breve resumo dos processos utilizados na etapa anterior.

Foram desenvolvidas cinco aplicações, destacando-se duas principais, nomeadamente, a **Aplicação Central** e a **Aplicação Cliente**. A Aplicação Cliente subdivide-se ainda nas restantes três aplicações, cada uma para um determinado utilizador (**profissional de saúde**, **gestor de segurança** ou **gestor de sistema**). De modo a permitir a comunicação entre estas aplicações com a Aplicação Central, foram estabelecidos **Sockets TCP** para permitir um funcionamento óptimo do programa.

Foram concluídas na primeira etapa as funcionalidades que dizem respeito à comunicação por sockets entre as aplicações e a gestão de credenciais para apenas possibilitar acesso a pessoas inscritas.

A Aplicação Central funciona como o cérebro da operação, como um Servidor e, como tal, cria três processos distintos para estabelecer uma ligação socket do tipo TCP com os diferentes portos que estão atribuídos a cada um dos tipos de utilizador, ficando sempre em modo "listen" até que haja alguma conexão. Após ser feita uma conexão, é criada uma nova thread dentro do respectivo processo para permitir que outras conexões sejam feitas.

A Aplicação Cliente foi criada para permitir o correcto redireccionamento do "cliente" para a sua respectiva aplicação. Após este passo, os clientes têm a possibilidade de efectuar o seu registo, introduzindo os seus dados (email e password (que é encriptada logo após inserção)). Estes dados serão então enviados para o servidor que irá por sua vez, enviar os dados para a base de dados. Desta forma, é facilitada a recuperação dos dados dos clientes para a confirmação de dados de acesso.

5.2 Etapa 2

5.2.1 Tarefa F3 e F4

No que concerne às tarefas f3 e f4, foram desenvolvidas funções que permitem o login e o registo ,com a devida permissão do gestor do sistema,em que tanto o profissional de saúde como o agente de segurança pode alterar o seu email,password e nome. É lhes permitido apagar as suas contas,onde existe a particularidade das ocorrências registadas pelo profissional de saúde, serem eliminadas quando o seu registo é apagado.

Todas estas funcionalidades estão devidamente protegidas por robustez do código em que não são permitidas respostas inválidas e opções inválidas.Para evitar qualquer constrangimento de dados,foi aplicado na criação,alteração e eliminação da conta opções no caso de engano de alguns dados e também opções para confirmação da criação,alteração ou eliminação de alguns dados da conta.

5.2.2 Tarefa F5

Relativamente ao gestor de sistema,nesta funcionalidade decidiu-se que só faria sentido haver uma opção para fazer o login do gestor de sistema,onde o registo deste é feito directamente na base de dados,com a devida encriptação.A encriptação do gestor de sistema é diferente das outras funcionalidades,visto que a injeccão é feita directamente na base de dados.

Quando o gestor de sistema pretende validar ou apagar uma conta,aparece-lhe uma tabela com os profissionais de saúde e outra com os agentes de segurança. Este terá de digitar o email respectivo da conta que pretende efectuar alterações e seguidamente confirmar a sua validação ou eliminação.Se eliminar a conta do profissional de saúde,as suas ocorrências também serão apagadas.Este não pode apagar a conta Anonymous pois esta é utilizada para fazer as ocorrências pelo profissional de saúde em anónimo.Esta é colocada directamente também na base dados e já validada.

Esta funcionalidade também possui robustez no código para não permitir respostas e opções inválidas.Também possui funcionalidades para evitar o constrangimento no código,onde pode sempre sair digitando "0" sem dar um email tanto na validação de uma conta como na eliminação.

5.2.3 Tarefa F6 e F8

Foram desenvolvidas algumas funções que permitem o que o profissional de saúde **registre as ocorrências na aplicação**. Inicialmente, o utilizador escolhe a opção de fazer um registo de ocorrência, avançando para um sub menu que lhe pede para preencher os campos necessários para este efeito, nomeadamente, a data, a hora, a localidade e uma descrição da ocorrência. Todas as respostas do utilizador são enviadas para o servidor mas, no entanto, só é feito o registo desses dados na base de dados quando o profissional de saúde seleccione a opção de "submeter a ocorrência". Quando

esta opção é accionada, o programa do servidor faz a validação dos dados já inseridos e, se todos os campos estiverem preenchidos e forem válidos, é enviado para o servidor um "sinal" para que avance e guarde estes dados na base de dados.

Toda a troca de informação que aqui ocorre é feita através de sockets TCP, como já referido, e com o objectivo de garantir o bom funcionamento do programa, o código foi estruturado de modo a que haja uma sincronização óptima entre as aplicações.

O código desenvolvido está preparado para inserção de dados incorrectos por parte do utilizador. Atributos como datas, hora, nomes, entre outros, só aceitam valores válidos, havendo medidas implementadas para informar o utilizador do erro cometido.

Apesar de não ser pedido nesta etapa, foi já **possibilitado o anonimato ao profissional de saúde** no momento da submissão de ocorrências. Após o accionamento da opção de submissão da ocorrência, o utilizador pode optar por submeter como anónimo ou não, sendo registado como "Anonymous" na base de dados se assim o preferir. Isto foi possível criando um utilizador "Anonymous" a quem é atribuída a ocorrência. Se preferir manter os seus dados relacionados com a ocorrência, o servidor envia o Id do profissional de saúde e não o do anónimo.

5.2.4 Tarefa F7

Após os registos de ocorrências, **foi desenvolvida uma funcionalidade que permite ao Agente de Segurança ver estes registos**. Para tal, foram criadas funções nas aplicações do agente de segurança e na aplicação central que permitem que a informação seja "levantada" da base de dados para o servidor e posteriormente para a aplicação do agente de segurança, garantindo uma maior segurança dos dados.

Para aceder aos registos de ocorrências, o agente de segurança selecciona essa opção no menu e é-lhe apresentada uma tabela com as ocorrências existentes em sistema. Os dados foram enviados por sockets e recebidos no agente de segurança que formata a informação recebida para uma tabela para facilitar a leitura dos dados. Foi ainda acrescentada a **funcionalidade de filtragem de dados**, sendo possível filtrar por data de ocorrência, descrição (qualquer palavra, substring, existente na descrição), localidade (pesquisa igualmente por substring) e Id de profissional de saúde.

5.3 Etapa 3

5.3.1 Tarefa F9

Houve uma alteração dos forks para threads, para facilitar a partilha de informação por variáveis globais. Criou-se dois novos portos destinados aos alarmes dos Profissionais de Saúde e dos Agentes de Segurança, cada um com conexões por sockets TCP diferentes.

Quando algum Profissional de Saúde selecciona a opção do alarme é, imediatamente,

emitido para todos os Agentes de Segurança, com sessão iniciada, um alerta, independentemente do menu em que este esteja. Entretanto, caso algum agente inicie sessão e um alarme tenha sido emitido momentos antes, este receberá o mesmo alerta.

Foi adicionado à base de dados uma nova tabela para registar todos os alarmes notificados, onde o Agente de Segurança a pode consultar e responder a qualquer pedido de socorro inserindo o "id" respetivo.

Assim que um Agente de Segurança responde a um alarme, o Profissional de Saúde recebe a informação de que algum agente vai no seu auxílio.

Posteriormente, após um Agente de Segurança ter respondido a um pedido de socorro, outro qualquer Agente não poderá corresponder ao mesmo, sendo que ao consultar todos os pedidos de ajuda, os agentes podem verificar quais se encontram ativos ou já atendidos.

Por fim, quando todos os alarmes forem respondidos, a flag, que se apresenta em todos os menus, que indica que existem alarmes ativos fica a off.

5.3.2 Tarefa F10

Foi adicionado em todos os menus uma opção para consultar informações sobre o menu em que o utilizar se encontra. Desta forma, as aplicações tornam-se mais fáceis de manusear e cada usuário consegue, deste modo, entender a função de cada opção.

5.3.3 Tarefa F11

Não implementada.

5.3.4 Tarefa F12

Considerou-se como extras:

- Em relação às passwords, aquando da inserção nas aplicações, estas são substituídas por asteriscos e no que diz respeito à base de dados, estas ficam encriptadas;
- Acesso à localização do Profissional de Saúde, por parte do Agente de Segurança, num pedido de socorro;
- O facto de que em qualquer menu em que o Agente de Segurança se encontre, este recebe notificação sobre os alertas acionados. Bem como, em qualquer menu, o Profissional de Saúde é informado que o seu pedido de auxílio está em andamento;
- Implementação de um submenu na aplicação dos Agentes de Segurança, em que está agrupada a informação sobre todos os alarmes emitidos pelos Profissionais de Saúde, havendo desta forma um manuseamento sobre esses alarmes;

- Implementação de um filtro adicional na pesquisa de ocorrências (pesquisa por palavra na descrição);
- Utilização de Threads para uma execução eficiente do software e para partilha de informações entre as várias aplicações;
- Muita robustez no código para proteção contra inserção de inputs incorretos e para evitar constrangimentos no software;
- Criação de uma base de dados que armazena e permite a manipulação de dados facilmente.