



UNIVERSIDADE  
**LUSÓFONA**

# Aplicação de controlo de assiduidade

## Trabalho Final de Curso

Relatório final

Aluno: Daniel Martins

Orientador: Professor Rodrigo Correia

Entidade Externa: Associação de Reformados, Pensionistas e Idosos de São Julião do Tojal

Trabalho Final de Curso | LEI | 2025-07-13

[www.ulusofona.pt](http://www.ulusofona.pt)

## **Direitos de cópia**

(Aplicação de controlo de assiduidade), Copyright de Daniel Martins, Universidade Lusófona.

A Escola de Comunicação, Arquitectura, Artes e Tecnologias da Informação (ECATI) e a Universidade Lusófona (UL) têm o direito, perpétuo e sem limites geográficos, de arquivar e publicar esta dissertação através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, e de a divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objectivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

Este documento foi gerado com o processador (pdf/Xe/Lua) $\text{\LaTeX}$  e o modelo ULThesis (v1.0.0) [Mat24].

# Resumo

Os sistemas de controlo da atividade laboral, onde se inclui o de gestão de assiduidade, são cruciais para médias e grandes empresas, dada a complexidade de manter o registo do cumprimento do horário de trabalho de cada trabalhador. Apesar da relevância que este sistema tem, quase a totalidade das soluções que existem para dar resposta ao problema foram desenvolvidas há bastantes anos atrás, tendo ficado estagnadas nas tecnologias e abordagens de desenvolvimento da altura.

Este trabalho é o ponto de partida para a criação de uma aplicação de gestão de assiduidade, com base em paradigmas de programação, *User Interface/User Experience* (UI/UX) e *hardware* modernos, focando-se na construção do terminal de registo pontométrico, que irá substituir o atualmente instalado na organização parceira deste trabalho.

# Abstract

The labor management systems, where attendance management is included, are crucial for medium and large businesses given the complexity of keeping track of the work schedule of each employee. Despite the importance of this system, almost all the existing solutions for this problem were developed many years ago, getting stagnated in development technologies and approaches of that time.

This project is the starting point of an attendance management application, using modern development paradigms, *User Interface/User Experience* (UI/UX) and hardware, focusing on building the physical attendance terminal that will replace the ones currently deployed at the partner organization.

# Índice

<b>Resumo</b>	<b>2</b>
<b>Abstract</b>	<b>3</b>
<b>Índice</b>	<b>4</b>
<b>Lista de Figuras</b>	<b>6</b>
<b>Lista de Tabelas</b>	<b>7</b>
<b>1 Introdução</b>	<b>8</b>
1.1 Enquadramento . . . . .	8
1.2 Motivação e identificação do problema . . . . .	8
1.3 Objetivos . . . . .	9
<b>2 Pertinência e Viabilidade</b>	<b>10</b>
2.1 Pertinência . . . . .	10
2.2 Viabilidade . . . . .	10
2.3 Análise Comparativa com Soluções Existentes . . . . .	10
2.3.1 Soluções existentes . . . . .	10
2.3.2 Análise de <i>benchmarking</i> . . . . .	11
2.4 Proposta de inovação e mais-valias . . . . .	11
2.5 Identificação de oportunidade de negócio . . . . .	11
<b>3 Especificação e Modelação</b>	<b>12</b>
3.1 Análise de requisitos . . . . .	12
3.1.1 Enumeração de requisitos . . . . .	12
3.1.2 Descrição detalhada dos requisitos principais . . . . .	12
3.1.3 User stories . . . . .	12
3.2 Modelação . . . . .	14
3.3 Protótipos . . . . .	14
3.3.1 <i>Hardware</i> do Terminal . . . . .	14
3.3.2 Interface do Terminal . . . . .	15
3.3.3 Interface da Administração web . . . . .	15
<b>4 Solução Desenvolvida</b>	<b>16</b>
4.1 Apresentação . . . . .	16
4.2 Arquitetura . . . . .	16
4.3 Tecnologias e Ferramentas Utilizadas . . . . .	17
4.3.1 Escolha de <i>hardware</i> . . . . .	17
4.3.2 Escolha de <i>software</i> . . . . .	17
4.4 Ambientes de Teste e de Produção . . . . .	17
4.5 Abrangência . . . . .	18
4.6 Componentes . . . . .	18
4.7 Interfaces . . . . .	18
4.8 Cenários melhorados . . . . .	18
4.8.1 Permuta de turno sem formalização . . . . .	18

4.8.2 Processamento de tempo trabalhado . . . . .	19
<b>5 Testes e validação</b>	<b>20</b>
<b>6 Método e Planeamento</b>	<b>21</b>
6.1 Planeamento inicial . . . . .	21
6.2 Análise crítica ao planeamento . . . . .	21
<b>7 Resultados</b>	<b>24</b>
7.1 Resultados dos testes . . . . .	24
7.1.1 Preparação dos ambientes de testes . . . . .	24
7.1.2 Ambiente de testes fechado . . . . .	24
7.1.3 Ambiente de testes aberto . . . . .	25
7.1.4 Ambiente de testes alternativo . . . . .	25
7.2 Cumprimento dos requisitos . . . . .	26
<b>8 Conclusão</b>	<b>28</b>
8.1 Conclusão . . . . .	28
8.2 Trabalhos futuros . . . . .	28
<b>Bibliografia</b>	<b>29</b>
<b>Glossário</b>	<b>30</b>
<b>A Análise do sistema Anviz CrossChex</b>	<b>32</b>
<b>B Exploração de <i>hardware</i> para o terminal de picagem</b>	<b>34</b>
B.1 Terminal de picagem com recurso a microprocessador . . . . .	34
B.2 Terminal de picagem com recurso a microcontrolador . . . . .	37
<b>C Orçamento para terminal de picagem</b>	<b>39</b>
<b>D Inquérito realizado</b>	<b>41</b>
<b>E Resultados do inquérito realizado</b>	<b>46</b>

# **Lista de Figuras**

3.1	Diagrama da base de dados	14
3.2	Versão inicial do protótipo do terminal de picagem	14
3.3	Versão atual do protótipo do terminal de picagem	15
4.1	Arquitetura para a solução escolhida	16
4.2	Mapa aplicacional do terminal	18
6.1	Planeamento inicial	21
6.2	Planeamento para o 3º trimestre (revisto)	21
6.3	Execução do calendário no 3º trimestre	22
A.1	Primeiro ecrã mostrado ao abrir a aplicação CrossChex	32
A.2	Ecrã <i>User -&gt; User Scheduling</i> mostrado na aplicação CrossChex	33
A.3	Ecrã <i>Shifts</i> mostrado na aplicação CrossChex	33
B.1	Funcionamento do sensor de impressão digital óptico (esquerda) e capacutivo (direita)	35
B.2	Sensor de impressão digital óptico AS608	36
B.3	Coluna USB	36

# **Lista de Tabelas**

2.1	Tabela comparativa das características dos terminais analisados . . . . .	11
3.1	<i>User Stories</i> . . . . .	13
7.1	<i>User Stories</i> . . . . .	27
B.1	Especificações dos SBCs Raspberry Pi . . . . .	35
B.2	Especificações dos sensores de impressão digital óticos . . . . .	36
B.3	Especificações dos microcontroladores selecionados . . . . .	37

# **1 - Introdução**

## **1.1 Enquadramento**

A Associação de Reformados, Pensionistas e Idosos de São Julião do Tojal, instituição onde dou suporte informático de forma voluntária desde 2021, dá resposta social através de Apoio Domiciliário, Centro de Dia, Atividades de Tempos Livres (ATL) e, desde 2019, Estrutura Residencial para Pessoas Idosas (ERPI). Perante o aumento do número de trabalhadores necessários para a instituição funcionar, resultante da abertura da ERPI, foi instalado um novo sistema de controlo de assiduidade, que além de suportar mais funcionários registados, também permite o tratamento automático dos registo de início e fim de trabalho (registro pontométrico).

## **1.2 Motivação e identificação do problema**

Até à abertura da ERPI, a tarefa de apurar o vencimento mensal do trabalhador seguia a seguinte metodologia:

1. Perto do final do mês, na aplicação de gestão dos terminais de picagem, eram exportados para *Portable Document Format* (PDF) todos os registo existentes para o mês corrente e impressos em papel;
2. Recorrendo ao modelo criado em folha de cálculo, onde consta o nome e número de funcionário, era inserido manualmente o horário de trabalho para cada dia do mês, conforme a escala publicada pela chefia da resposta social que integra, e as picagens do trabalhador. Em caso de anomalia, eram verificados os pedidos no âmbito do trabalho (permute de turno, folga, férias) ou justificações entregues aos Recursos Humanos (RH) em papel, sendo que se existissem, eram referenciados como observação na folha de cálculo.
3. Concluindo o preenchimento da folha de cálculo, eram obtidas as horas de trabalho esperadas e cumpridas, para determinar se tem de existir penalização no vencimento pelo saldo de horas não cumpridas, caso o número de horas em défice fosse superior a 1 dia de trabalho.
4. Em seguida, as informações de horas obtidas eram transcritas para uma folha de registo de majorações e deduções (em papel), isto é, horas extra, faltas e subsídios de turno, referente a cada trabalhador, para poder constar em arquivo.
5. Por fim, era feita a transcrição das majorações e deduções, isto é, faltas (dias de falta, baixas médicas ou baixas de parto), remunerações (subsídios ou trabalho suplementar) e descontos (quotas para sindicatos ou penhoras de vencimento), referente a cada funcionário, para inserir na aplicação de processamento de salários e, posteriormente, constar no arquivo em papel.

Após a instalação do novo sistema de controlo de assiduidade, a metodologia usada continuou a ser a mesma descrita nos pontos anteriores, devido à quantidade de informação necessária de inserir à data de instalação do novo sistema (todas as escalas, faltas e férias para cada trabalhador) e à interface gráfica pouco intuitiva, que torna difícil introduzir e trabalhar essas informações. Existindo uma aplicação adquirida para resolver

o problema, parcial ou totalmente, que se encontra subaproveitada, suscitou o interesse da minha parte para criar algo que fosse ao encontro do pretendido.

### **1.3 Objetivos**

Com este projeto, focando no contexto da instituição parceira, os objetivos passam por:

- Criação de um sistema de gestão de assiduidade que seja funcional e, de facto, aproveitado ao máximo para reduzir o tempo despendido mensalmente com tarefas de rotina, associadas a este tema;
- Desmaterialização de processos, passando a informação a ser inteiramente tratada de forma digital;
- Combate ao elevado número de irregularidades, lançando avisos das mesmas muito mais próximo da altura em que aconteceram;
- Redução da possibilidade de fraude, por meio de estruturas de dados que previnem a existência de "informações soltas" e através do rastreamento de todas as ações realizadas na aplicação.

## 2 - Pertinência e Viabilidade

### 2.1 Pertinência

A pertinência deste projeto surge no seguimento do exposto no capítulo 1.2, a expansão das instalações da ARPI S. J. Tojal, com o consequente aumento considerável de funcionários. Como a construção tem conclusão prevista para o 3º semestre de 2025 e é esperado que se encontre totalmente em utilização no início do 4º trimestre de 2025, o período de desenvolvimento enquadra-se com estes prazos, exigidos pela instituição.

### 2.2 Viabilidade

Para suporte à validação, foi realizado um inquérito às funcionárias dos RH e às chefias da instituição, com o objetivo de aferir a satisfação da solução atualmente instalada e do processo interno relativamente a toda a componente de contabilização de horas de trabalho. Este aconteceu de forma informal, com a maioria das pessoas insatisfeitas, alegando falhas no cálculo de horas, lentidão no processamento de pedidos corretivos e demora desde que uma anomalia relativa ao horário de trabalho acontece, até que é detetada.

Também foi realizado um inquérito formal, com resposta de 42 dos 86 trabalhadores na instituição, onde a maioria considera que o sistema atual funciona bem, no entanto, 50% dos inquiridos revela que no último ano teve anomalias com o seu registo de picagem, principalmente devido a picagens em falta. Houve resposta favorável a todas as funcionalidades propostas, existindo outras 2 sugestões pertinentes como *Qualidade de Vida* (QdV): destacar visualmente quando o turno ultrapassa as 8h, com o excedente trabalhado; indicar há quantos dias foi realizada a última validação de picagens por parte do RH. O resumo das respostas pode ser consultado no apêndice E.

### 2.3 Análise Comparativa com Soluções Existentes

#### 2.3.1 Soluções existentes

Realizada pesquisa de equipamentos de registo pontométrico no mercado, selecionaram-se o Anviz W1 Pro<sup>1</sup>, ZKTeco ZK-UA150<sup>2</sup> e Hysoon HY-C280<sup>3</sup> por serem opções bastante comuns e de funcionalidade semelhante. No caso da Anviz, existe uma aplicação de controlo centralizado, que se encontra analisada em detalhe no apêndice A.

---

<sup>1</sup>Página do produto: <https://www.anviz.com/product/w1pro-time-attendance.html>

<sup>2</sup>Página do produto: <https://zkteco.eu/products/time-attendance/biometric/ua150>

<sup>3</sup>Página do produto: <https://hysoon.es/index.php/product/hy-c280a/>

### 2.3.2 Análise de *benchmarking*

Equipamento	Anviz W1 Pro	ZKTeco ZK-UA150	Hysoon HY-C280	Este projeto
Autenticação por impressão digital	Sim	Sim	Sim	Sim
Autenticação por cartão RFID	Sim	Sim	Sim	Não
Autenticação por PIN	Sim	Sim	Sim	Sim
Capacidade de impressões digitais	3000	500	2000	300
Tamanho do ecrã	2.8"	2.8"	2.8"	5.0"
Fonte de alimentação	5V DC	5V DC 0.8A	5V DC 1A	5V DC 5A
Feedback visual	Sim	Sim	Sim	Sim
Feedback sonoro	Voz	Voz	Voz	Beeps
Aviso de anomalias	Não	Não	Não	Sim
Gestão web	Sim	Não	Não	Sim

Table 2.1: Tabela comparativa das características dos terminais analisados

Numa visão geral, os equipamentos existentes no mercado são bastante idênticos. O contraste notório entre este projeto e as restantes opções identificadas são o aviso de anomalias, gestão web (quando o equipamento opera em modo independente) e a especificação da fonte de alimentação, apesar de não ser esperado que o consumo do terminal desenvolvido exceda 50% da potência disponível do transformador.

## 2.4 Proposta de inovação e mais-valias

Este projeto propõe inovar o tradicional terminal de registo pontométrico, que além de registar uma picagem, permitir verificar a existência de anomalias durante o dia atual e mostrar de imediato informações ao utilizador, com o objetivo de reduzir e normalizar irregularidades. Além disto, o facto de ser concebido de raiz com uma base de *hardware* moderna, tirando partido dos ganhos geracionais de desempenho, eficiência e recursos disponíveis ao longo dos últimos anos, permitindo o desenvolvimento sem compromissos e a posterior adoção de tecnologias emergentes. Por fim, também procura focar-se na acessibilidade, ao fornecer *feedback* visual e sonoro simples de entender.

Este projeto, por ser desenvolvido especificamente para a ARPI, tem as mais-valias de estar ajustado à realidade da instituição parceira e não se trata de uma solução fechada, podendo ser alterada e melhorada ao longo do tempo.

## 2.5 Identificação de oportunidade de negócio

Quanto à continuidade pós Trabalho Final de Curso, considerando que durante o TFC apenas será desenvolvido o terminal de picagem, a aplicação complementar sobre as tarefas de gestão de assiduidade é uma das possibilidades de expansão futura, pelo que se garante que não existe esgotamento com o presente trabalho.

# 3 - Especificação e Modelação

## 3.1 Análise de requisitos

### 3.1.1 Enumeração de requisitos

### 3.1.2 Descrição detalhada dos requisitos principais

Tendo por base o capítulo 1.2, identificação do problema, determinam-se 3 épicos para este trabalho:

Épico 1 - Enquanto presidente, quero um terminal pontométrico físico, para que possa ser instalado na entrada do edifício para os funcionários; (A existência do terminal de picagem propriamente dito);

Épico 2 - Enquanto Recursos Humanos, queremos que os funcionários registem a sua presença, para que possamos controlar a sua assiduidade. (As funcionalidades para funcionários);

Épico 3 - Enquanto Recursos Humanos, queremos uma página de gestão web, para que possamos realizar tarefas administrativas remotamente. (As funcionalidades para os Recursos Humanos).

### 3.1.3 User stories

Feature	User story	Tipo
E1F1	Enquanto presidente, quero que o terminal esteja conectado à internet, para que possa ser gerido remotamente	Obrigatório
E1F2	Enquanto presidente, quero que o terminal tenha alimentação de <i>backup</i> , para que se mantenha operacional durante falhas da rede de energia	Obrigatório
E1F3	Enquanto presidente, quero que seja possível autenticar com impressão digital, para que os funcionários possam registar a sua presença	Obrigatório
E1F4	Enquanto presidente, quero que seja possível autenticar com código PIN, para que os funcionários com impressão digital difícil de ler tenham uma forma alternativa de autenticação	Obrigatório
E2F1	Enquanto Recursos Humanos, queremos registrar a impressão digital dos funcionários no sistema, para que se possam autenticar	Obrigatório
E2F2	Enquanto Recursos Humanos, queremos registrar uma impressão digital secundária, para que os funcionários tenham um dedo alternativo para autenticação	Opcional
E2F3	Enquanto Recursos Humanos, queremos definir um código PIN para os funcionários, para que os funcionários com impressão digital difícil de ler tenham uma forma alternativa de autenticação	Obrigatório

E2F4	Enquanto utilizador, quero autenticar-me com impressão digital, para que possa registar a hora em que compareci para início e fim de turno	Obrigatório
E2F5	Enquanto utilizador, quero autenticar-me com código PIN, para que possa registar a hora em que compareci para início e fim de turno de forma alternativa	Obrigatório
E2F6	Enquanto utilizador, quero ouvir uma sequência de beeps, para que possa rapidamente perceber o resultado da minha autenticação	Obrigatório
E2F7	Enquanto utilizador, quero ver informações no ecrã, para que possa ter mais detalhes do resultado da minha autenticação	Obrigatório
E2F8	Enquanto utilizador, quero ter informação de anomalias com picagens anteriores, para que possa verificar junto dos Recursos Humanos	Obrigatório
E2F9	Enquanto administrador de sistemas, quero que as credenciais sejam encriptadas, para que os dados se mantenham seguros em caso de acesso indevido	Opcional
E2F10	Enquanto programador, quero <i>logs</i> da aplicação, para que possa entender e replicar o que aconteceu quando um <i>bug</i> é reportado.	Opcional
E3F1	Enquanto Recursos Humanos, queremos registar funcionários, para que estes possam usar o sistema	Obrigatório
E3F2	Enquanto Recursos Humanos, queremos carregar escalas de trabalho, para que a aplicação possa saber quando é esperado que o funcionário faça o seu registo de assiduidade	Obrigatório
E3F3	Enquanto Recursos Humanos, queremos descarregar um relatório de horas de cada funcionário para um dado mês, para que possamos processá-lo e pagar salários	Obrigatório
E3F4	Enquanto Recursos Humanos, queremos registar códigos de trabalho, para que seja possível identificar o tipo de presença ou ausência esperada do trabalhador	Obrigatório
E3F5	Enquanto Recursos Humanos, queremos registar cargos, para que seja possível representar a nossa estrutura institucional na aplicação	Obrigatório

Table 3.1: *User Stories*

O controlo de desenvolvimento das *User Stories* listadas está a ser feito através de um quadro na plataforma Trello, disponível em <https://trello.com/b/I5FsFTis/tfc-2024-25-aluno-2108>

## 3.2 Modelação

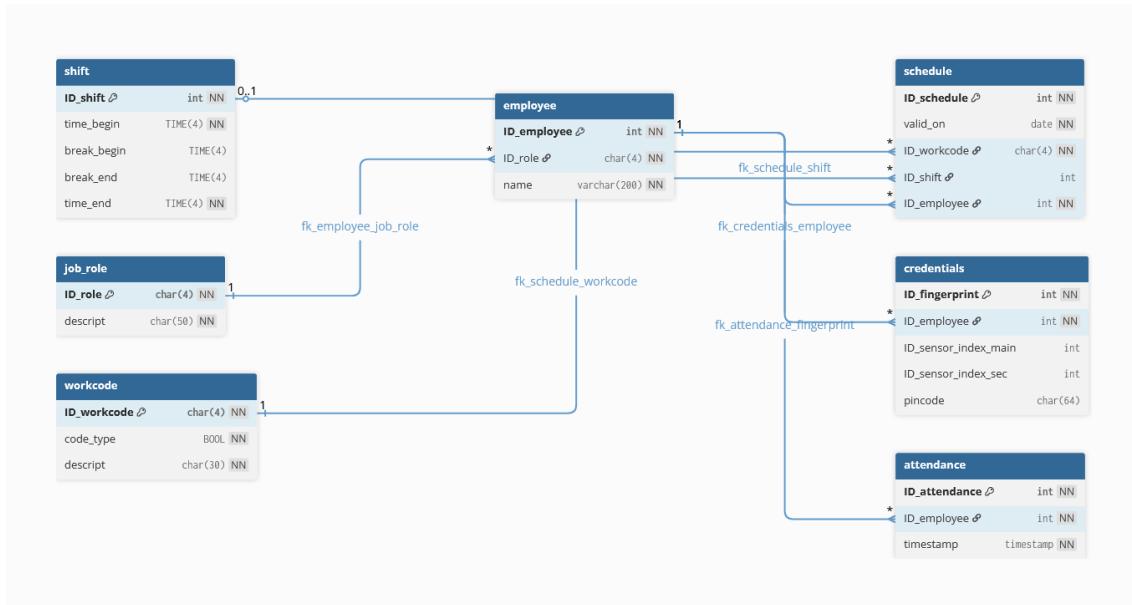


Figure 3.1: Diagrama da base de dados

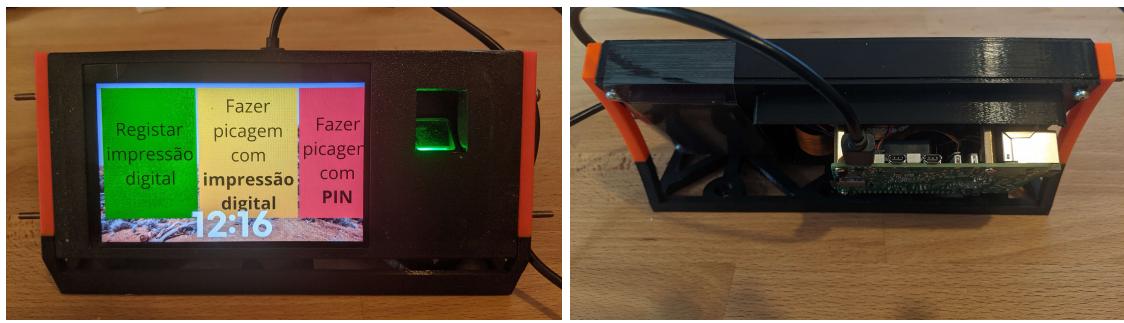
## 3.3 Protótipos

### 3.3.1 Hardware do Terminal

O protótipo do terminal físico foi construído com o objetivo de ser simples e discreto. Foram necessárias várias iterações até à versão atual, onde o Raspberry Pi está orientado de modo diferente e existe um suporte de mesa.



Figure 3.2: Versão inicial do protótipo do terminal de picagem



(a) Frente do terminal

(b) Trás do terminal

Figure 3.3: Versão atual do protótipo do terminal de picagem

### 3.3.2 Interface do Terminal

No terminal, a interface foca-se na acessibilidade, com uso de cores de elevado contraste e texto com tamanho de letra grande. Para não tornar o ambiente demasiado aborrecido, foi usada uma imagem de fundo de alta qualidade em vez de uma cor sólida (à exceção do ecrã de resultado).

### 3.3.3 Interface da Administração web

Para a aplicação web de administração do terminal não foram concebidos protótipos de interface. No entanto a ideia principal é a de uma página com menu de navegação lateral e o conteúdo de cada opção mostrado em tabela.

## 4 - Solução Desenvolvida

### 4.1 Apresentação

A solução desenvolvida contempla o terminal de registo pontométrico e uma aplicação web para gestão do terminal. No terminal é possível realizar uma picagem com impressão digital ou pin e obter *feedback* visual e sonoro, com informação de anomalias para o dia atual. Na aplicação web, é possível realizar operações *Create, Read, Update and Delete* (CRUD) relativamente aos funcionários e às suas picagens, cargos, turnos e códigos de trabalho. Também é possível importar uma escala de trabalho e exportar uma folha de registo de horas.

Para chegar à versão desenvolvida, foi inicialmente criada uma Prova de Conceito (PdC) para validar a capacidade de execução do projeto, de modo a cumprir os requisitos identificados. O vídeo de análise da PdC está disponível em <https://www.youtube.com/watch?v=uFecjalULP4>

Todo o código-fonte desenvolvido pode ser consultado no repositório Git [https://github.com/DEISI-ULHT-TFC-2024-25/TFC-Aluno2108-Aplicacao\\_controlo\\_assiduidade\\_c\\_terminal\\_de\\_picagem](https://github.com/DEISI-ULHT-TFC-2024-25/TFC-Aluno2108-Aplicacao_controlo_assiduidade_c_terminal_de_picagem);

O funcionamento da solução pode ser comprovado pelo vídeo de demonstração [https://www.youtube.com/watch?v=7esj\\_gfMD4o](https://www.youtube.com/watch?v=7esj_gfMD4o).

### 4.2 Arquitetura

A solução desenvolvida está segmentada em 3 componentes, todas alojadas no terminal de picagem.

A primeira componente é o *frontend do terminal*, para dar funcionalidade ao terminal propriamente dito. Esta componente tem o *frontend* e *backend* desenvolvidos de forma independente, estando interligados por uma bridge. A segunda componente é a BD, para guardar os dados da aplicação de forma estruturada. A terceira componente é a aplicação web, para permitir a realização de tarefas de gestão do terminal e dos dados armazenados. Esta componente recorre a uma *framework* para comodidade de desenvolvimento.

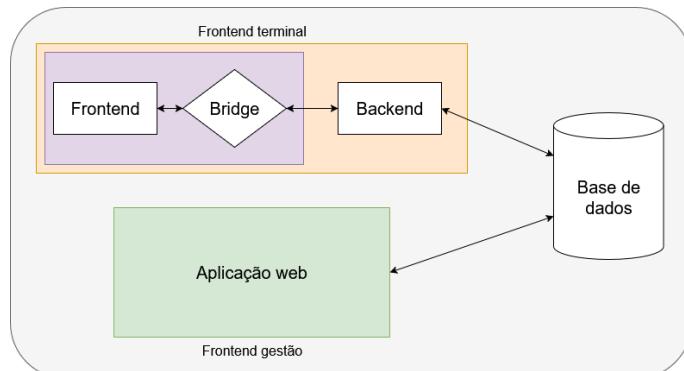


Figure 4.1: Arquitetura para a solução escolhida

## 4.3 Tecnologias e Ferramentas Utilizadas

### 4.3.1 Escolha de *hardware*

Ao nível de *hardware* foi feita uma análise extensiva das opções possíveis, com a análise das vantagens e desvantagens de cada uma, presente no apêndice B,

Fazendo o balanço das opções, a escolhida foi o desenvolvimento do terminal de picagem com recurso a microprocessador. A decisão tem em consideração o facto de a instituição não ter colocado constrangimentos relativamente ao custo estimado do terminal e, que por já ter sido a base usada no desenvolvimento da prova de conceito, sem impasses a considerar, confere segurança, fator esperado para a continuação do projeto.

Esta opção representou um custo de 166,81€, estando o orçamento presente no apêndice C

### 4.3.2 Escolha de *software*

Considerando a escolha de *hardware* feita no ponto B, a arquitetura da solução será executada sobre o sistema operativo RaspberryOS, baseado em Debian 12. Existirão 2 componentes:

*Frontend* do terminal propriamente dito, onde estará toda a funcionalidade disponível no terminal (registo de nova impressão digital e picagem via impressão digital ou PIN. Como a biblioteca usada para o sensor de impressão digital apenas existe para as linguagens C e Python, a escolhida para todos os componentes do terminal será Python. Para o desenvolvimento de interface gráfica em Python para Linux existem opções robustas como Qt6<sup>1</sup> ou GTK4 (via PyGObject<sup>2</sup>), excessivas para o objetivo em causa. Em alternativa, será usado HTML, CSS e Javascript interligado com o *backend* pelo package Eel<sup>3</sup>, para Python.

*Frontend* de gestão, onde serão expandidas as funcionalidades desenvolvidas na prova de conceito, usando a mesma *framework web*, Django.

Para suportar estas componentes existirá uma base de dados relacional alojada localmente no Sistema de Gestão de Bases de Dados (SGBD) MariaDB.

## 4.4 Ambientes de Teste e de Produção

Uma vez que todo o projeto está a ser executado no Raspberry Pi existente no terminal, os ambientes de teste e produção têm as mesmas características:

- SoC Broadcomm BCM2712
- 2GB de memória DDR4X
- Cartão microSD 32GB
- Fonte de alimentação 5V 5A
- Sensor de impressão digital AS308
- Ecrã tátil 5"
- Coluna de som 2x2W USB

<sup>1</sup>Página do Qt6: <https://www.qt.io/product/qt6>

<sup>2</sup>Página do projeto: <https://pygobject.gnome.org/>

<sup>3</sup>Repositório do package Eel: <https://github.com/python-eel/Eel>

## 4.5 Abrangência

Para o desenvolvimento deste projeto foram aplicados conhecimentos adquiridos nas seguintes Unidade Curricular (UC):

- Arquiteturas Avançadas de Computadores, pelo uso da plataforma Raspberry Pi com a intenção de controlar sensores;
- Bases de dados, para o armazenamento estruturado dos dados existentes e criados pela solução;
- Programação Web, pelo desenvolvimento da componente de gestão do terminal (web);
- Interação Humano-Máquina, pelas boas práticas de criação de *User Interface/User Experience* tendo em atenção a acessibilidade;

Adicionalmente, foram aplicados conhecimentos de modelação e impressão 3D adquiridos de forma autónoma ao longo dos últimos anos.

## 4.6 Componentes

### 4.7 Interfaces

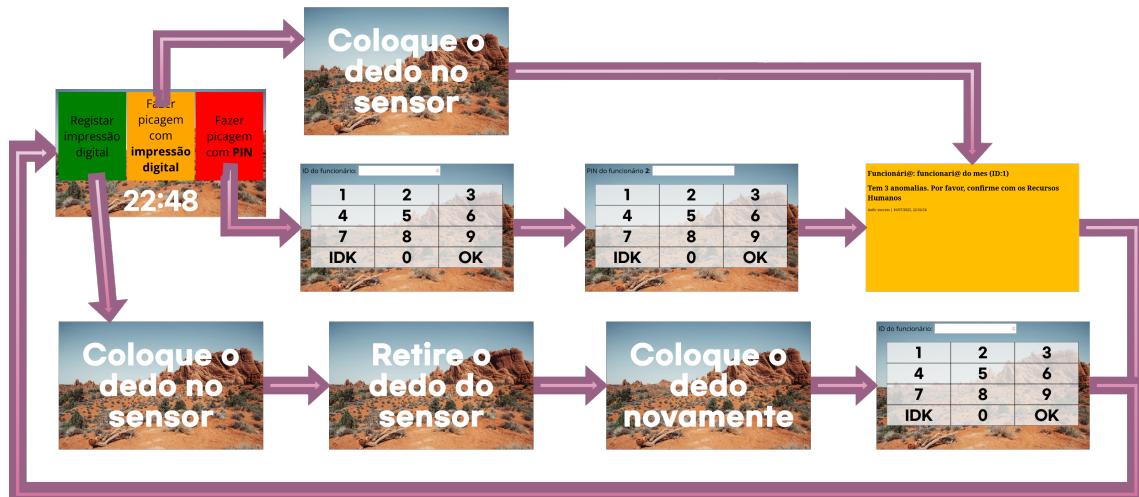


Figure 4.2: Mapa aplicacional do terminal

## 4.8 Cenários melhorados

A solução a desenvolver terá de implementar melhorias relativamente às mostradas no ponto 2.3.2, nomeadamente uma forma de conseguir dar informação ao funcionário relativamente a anomalias quanto às suas picagens, de forma imediata. Passam a ser descritas as abordagens que serão adotadas para estes cenários.

### 4.8.1 Permuta de turno sem formalização

De acordo com a escala, um funcionário está a trabalhar no turno da tarde (16:00 às 22:00) e faz permuta de turno com um colega para trabalhar antes no turno da manhã

(8:00 às 14:00) do mesmo dia, obtendo permissão verbal da chefia. No entanto, nunca chegou a formalizar o pedido de permuta de turno nos RH.

Atualmente, esta anomalia apenas é detetada no fim do mês (altura em que é feito o processamento de horas) pela hora das picagens não corresponder ao turno da escala e por não existir um pedido autorizado de alteração de turno. Apenas neste momento é desencadeado o processo de normalização da regularização junto da chefia e do trabalhador.

Com este projeto, esta anomalia passa a ser detetada no momento em que o funcionário faz a picagem, devido à verificação da escala nesse mesmo momento. O alerta é dado através do equipamento onde é feita a picagem, com uma curta indicação sonora, acrescida de informação para o trabalhador corrigir a anomalia com os RH.

#### **4.8.2 Processamento de tempo trabalhado**

Atualmente para realizar o processamento de horas de trabalho de um dado mês é necessário primeiro agregar todos os dados das diferentes fontes de informação manualmente. Inclusive, alguns destes dados apenas estão disponíveis em papel. Após este passo passa a ser possível construir os relatórios de horas para cada trabalhador, com recurso a uma folha de cálculo. É uma tarefa que dado o número de trabalhadores demora vários dias a concluir.

Com este projeto, visto que passa a ser a fonte principal de informação, o tempo para a construção dos relatórios de horas é drasticamente reduzido, no melhor cenário para até 1 dia, ficando apenas a faltar tratar os casos onde existem picagens em falta e os documentos existentes em papel.

## **5 - Testes e validação**

Para garantir o funcionamento do terminal de registo pontométrico conforme os requisitos estipulados, foram planeadas 2 fases de testes:

A primeira fase, reservada a um grupo mais pequeno de utilizadores, com o objetivo de verificar preliminarmente em contexto real se existem erros que não foram identificados durante o desenvolvimento. Planeou-se para decorrer durante 3 dias, 13 a 15 de maio, de modo a simular o cenário da primeira vez em que o sistema começa ser a usado e compreender se com registo de picagens de dias anteriores o comportamento continua a estar dentro do esperado. Foram convidados funcionários escalados para os dias em que estes testes decorreram, especificamente 1 funcionário por cada tipo de escala (fixa, rotativa, livre) para verificar a deteção de picagem face à escala. Também foi pretendido testar o caso onde a escala de trabalho termina no dia seguinte, por exemplo, 00:00 às 08:00.

A segunda fase, aberta a todos os funcionários da instituição, com o objetivo de simular o funcionamento em ambiente de produção. Aqui o foco foi a capacidade do terminal funcionar de forma persistente, 24 horas por dia, 7 dias por semana, e resiliente no caso de surgirem falhas. Planeou-se para decorrer durante 4 semanas, 19 de maio a 15 de junho.

# 6 - Método e Planeamento

## 6.1 Planeamento inicial

Conforme proposto no 1º relatório, o planeamento seguiu 3 fases, divididas com base nas datas de entrega definidas pela Universidade. O mesmo foi revisto no 2º relatório devido à inclusão de um ambiente de testes fechado, para garantir a boa execução do ambiente de testes aberto, e à necessidade de abranger funcionalidades ainda por desenvolver, mas necessárias para a realização dos ambientes de testes.

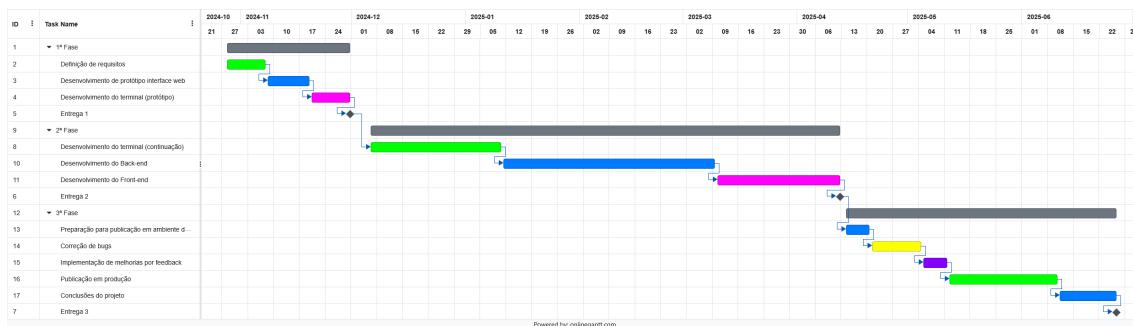


Figure 6.1: Planeamento inicial

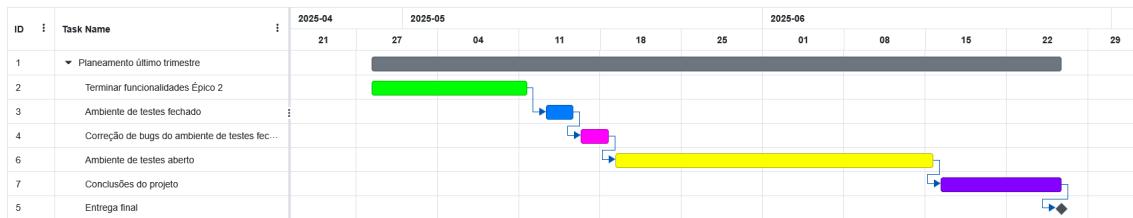


Figure 6.2: Planeamento para o 3º trimestre (revisto)

## 6.2 Análise crítica ao planeamento

Colocados os planeamentos na secção anterior, segue-se a análise crítica, com suporte no calendário de execução.

O planeamento, de modo geral, é adequado em termos de fases a realizar para alcançar a solução final, mas otimista em termos do tempo alocado a cada uma. A criação da expectativa do tempo ideal para a realização de uma tarefa não ter em consideração o contexto do programador, isto é, o desenvolvimento deste projeto acontecer em simultâneo com a frequência de Unidade Curricular e outras responsabilidades pessoais, como o trabalho voluntário na Associação de Reformados, Pensionistas e Idosos, faz com que o desenvolvimento tenha de ser revisto negativamente, ao aceitar como bom algo que é satisfatório, em vez de positivamente, ao implementar melhorias de *Quality of Life* ou mesmo funcionalidades opcionais. Mais, ao não existir uma margem para garantir resiliência em caso de qualquer imprevisto em qualquer domínio, coloca em risco o cumprimento dos objetivos. Algo desta natureza ocorreu no 3º trimestre, como se verifica pelo contraste da execução na 6.3 e do planeamento na 6.2.

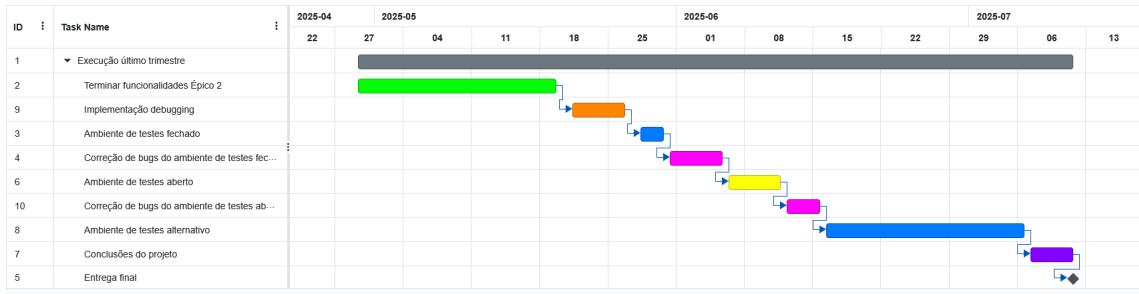


Figure 6.3: Execução do calendário no 3º trimestre

Ao longo de todo o desenvolvimento existiram 2 momentos marcantes dessa natureza:

O primeiro surgiu pouco após o início do desenvolvimento. Nesta altura apenas estava implementada a autenticação por biometria, recorrendo a um *websocket* para a interligação entre o *backend* Python e o *frontend* Javascript. Com a necessidade de implementar a autenticação por código PIN e registo de impressão digital, surgiram problemas a nível cognitivo e técnico, de como garantir que o sincronismo entre *backend* e *frontend* estava correto.

O *websocket* usado recorre à biblioteca "websockets (Python)", com implementação em *threads* e computação assíncrona. Até este ponto, a aplicação usou a implementação em *threads*. Após pesquisa na documentação desta biblioteca, surge um exemplo de abordagem do servidor enquanto produtor e consumidor, que recorre à computação assíncrona. Apesar da pouca experiência em desenvolvimento de computação assíncrona, foi feita uma tentativa com esta implementação, novamente sem sucesso, devido à complexidade de organização de produção e consumo para as 3 funcionalidades.

Recorrendo a nova pesquisa de abordagens para interligar o *backend* e *frontend*, foi descoberta a biblioteca "Eel (Python)" que implementa igualmente por *websockets* a execução de funções do *backend* Python no *frontend* Javascript e vice-versa, tal como a execução de callbacks. Esta acabou por ser a solução adotada visto que além de funcionar para as necessidades do projeto, constitui uma abstração ao uso de *websockets*, simplificando o desenvolvimento.

Apesar desta dificuldade ter consumido algumas semanas de trabalho e poder ter sido evitada caso a Prova de Conceito já permitisse a autenticação por impressão digital e código PIN (apesar da necessidade de prova incidir apenas na autenticação por impressão digital), constituiu uma oportunidade para exploração de opções alternativas e, portanto, uma experiência enriquecedora para o futuro.

O segundo momento surgiu durante os ambientes de testes. Apesar de durante o desenvolvimento não terem surgido problemas na funcionalidade, por terem acontecido durante períodos de curta duração, ao colocar o terminal de picagem em execução de longo prazo surgiram problemas relacionados com a temperatura do equipamento.

O Raspberry Pi, à semelhança de outros microcomputadores, tem um mecanismo de proteção térmica, onde bloqueia ao atingir temperaturas elevadas, para proteger os componentes. Após o acontecimento de alguns bloqueios, começaram a ser verificadas as temperaturas do SoC, sendo que tipicamente se encontravam nos 60°C após o arranque, subindo progressivamente até aos 70°C ou superior. A primeira intervenção foi alterar a curva de temperatura da ventoinha, para que funcione de forma mais agressiva, não tendo sido observadas melhorias relevantes. Adicionalmente, o dissipador de calor foi substituído por um com ventoinha axial (em vez da ventoinha *blower* no dissipador originalmente instalado) e maior área de superfície, que mesmo com curva

de temperatura da ventoinha mais agressiva não evitou o aumento progressivo da temperatura até o equipamento bloquear.

Devido a suspeitas da causa deste aumento de temperatura ser causado pela exigência da aplicação, foram aplicadas 3 otimizações para reduzir o tempo de utilização do processador: alteração da função que mostra o relógio, para que fosse atualizada apenas 1 vez por minuto (em vez de 1 vez por segundo); alteração da extensão dos tipos de letra, de *TrueType Font* (TTF) para *Web Open Font Format 2* (WOFF2), uma vez que é um formato especificamente concebido para *websites*; alteração da resolução da imagem de fundo de 4K para a resolução do ecrã, 480p, já que apenas será usada no terminal com esta resolução específica.

Novamente, apesar destas otimizações, o problema continuou a verificar-se. Recorrendo a uma câmara térmica, identificou-se que o ecrã é o componente do terminal de picagem que mais aquece, e, pela base do Raspberry Pi estar a poucos centímetros da base do ecrã, acabou por dissipar parte do calor para o SBC, fenómeno compatível com as observações notadas anteriormente. Posto esta descoberta, o Raspberry Pi foi reorientado para estar mais afastado do ecrã e evitar reter o calor do ecrã.

Após esta alteração, verificou-se que o SoC passou a ter temperaturas entre os 40°C e os 50°C, valores bastante confortáveis para o típico de um sistema embebido.

A abordagem adotada para a resolução desta dificuldade contribuiu para melhorias para o projeto, mas além de consumir vários dias de diagnóstico, condicionou o tempo em que o ambiente de testes aberto esteve a decorrer. Sendo que o problema estava relacionado com temperatura, ao não identificar qualquer tarefa computacionalmente exigente em execução prolongada, uma das primeiras fontes de diagnóstico deveria ter sido a câmara térmica, em vez da última.

# 7 - Resultados

## 7.1 Resultados dos testes

### 7.1.1 Preparação dos ambientes de testes

Antes de dar início aos ambientes de testes foi necessário preparar informações relativas aos funcionários e às escalas de trabalho. Algumas destas informações foram fáceis de importar, visto que já se encontram informatizadas em formatos de fácil importação (ficheiro *Comma Separated Values* (CSV)) ou encontram-se noutras formatos onde foi possível criar mecanismos para converter para CSV. No entanto, existiram dados que tiveram de ser recolhidos manualmente, sendo o caso da impressão digital. Este foi o dado com maior variabilidade na recolha, tendo existido casos onde não foi possível recolher a biometria devido ao dedo ter estado em contacto com água recentemente ou pelo elevado desgaste da impressão digital, causada pelo efeito de substâncias ou materiais abrasivos sem proteção das mãos. Nestes casos, a autenticação do trabalhador terá de ser feita com recurso a código PIN. Existem outras situações onde não foi possível recolher a biometria devido ao dedo ser demasiado grande para o compartimento onde o sensor foi colocado ou devido ao comprimento da unha fazer com que o dedo não ficasse posicionado adequadamente no sensor. Já nestes cenários, por se prenderem com a forma como o terminal está fisicamente construído, serão tomadas percauções no *design* do equipamento final para não terem impacto na funcionalidade do terminal de picagem.

### 7.1.2 Ambiente de testes fechado

O ambiente de testes fechado, planeado para os dias 13 a 15 de maio, decorreu entre 28 e 30 de maio. O atraso da realização destes testes deveu-se ao acréscimo da funcionalidade de registo de *logs*, para conseguir identificar melhor problemas reportados. Participaram 12 funcionários, um para cada escala de trabalho existente, abrangendo todas as respostas sociais.

Ao longo destes dias foram detetados problemas severos, que impediram a aplicação de executar no longo prazo, nomeadamente temperatura. Foram aplicadas correções ao nível físico, com a substituição do dissipador instalado para um com ventoinha axial (em vez de ventoinha *blower*) e reposicionamento do terminal de modo a que a base, único local com aberturas para ventilação, não estivesse completamente sobre a superfície que o apoia. Foram também aplicadas correções ao nível de software, com a alteração dos parâmetros de inicialização do "Eel", para que o *websocket* usado para comunicar entre o *backend Python* e o *frontend Javascript* não seja fechado poucos segundos após deixar de estar em uso. Com estas correções foram observadas melhorias de tempo de execução. Quanto à utilização, os botões do menu foram o aspeto que causaram maior dificuldade aos funcionários, visto que apenas o texto dentro do botão, em letra de tamanho normal, permitia aceder ao ecrã desejado. Também foi observado que o volume dos códigos de beeps era baixo, havendo trabalhadores a não reparar na sua existência.

Apesar das dificuldade sentidas, o balanço deste ambiente de testes foi positivo, visto que contribuiu para a resolução de problemas que impedem o funcionamento do protótipo no ambiente real.

### 7.1.3 Ambiente de testes aberto

O ambiente de testes aberto, planeado para os dias 19 de maio a 15 de junho, decorreu entre 6 e 11 de junho. No período desde o fecho do ambiente de testes fechado até à abertura do ambiente de testes aberto, foram aplicadas alterações para os problemas de utilização mencionados anteriormente: alteração dos botões do menu, para que o texto prenchesse o máximo espaço disponível e todo o botão permitisse aceder ao ecrã seguinte; aumento do ganho dos ficheiros de som.

Devido ao tamanho do edifício, atualmente existem 2 terminais de picagem onde os trabalhadores de cada resposta social apenas podem fazer o registo pontométrico naquele que corresponde ao seu local de entrada e saída. Visto que apenas existe um protótipo do terminal, o local escolhido como equilibrado foi o mesmo do ambiente de testes fechado, próximo da entrada principal da instituição. Por motivos de conformidade legal, foi exigido pelos RH que os funcionários fizessem as suas picagens no sistema atual e no de testes.

Mesmo com as correções aplicadas no ambiente de testes anterior, voltaram-se a verificar severas falhas de funcionamento no longo prazo, com maior frequência devido ao calor sentido nestes dias. Por este motivo, que transmitiu aos funcionários a sensação de muito provavelmente não conseguirem picar, e pela necessidade de trabalhadores de algumas respostas sociais terem de dispensar mais alguns minutos para se deslocarem ao terminal de testes, picar e voltar ao local onde desempenham funções, foram fatores que levaram ao cancelamento deste ambiente de testes pelo risco de não estarem a ser obtidos dados que permitam tirar conclusões do projeto.

### 7.1.4 Ambiente de testes alternativo

O ambiente de testes alternativo, substituto do ambiente de testes aberto, decorreu entre 16 de junho e 4 de julho. Com as lições aprendidas dos ambientes de testes anteriores, a abordagem adotada teve foco nos trabalhadores da resposta social com mais escalas de trabalho para permitir a instalação do protótipo do terminal ao lado do atualmente existente e desta forma aumentar a participação neste ambiente de testes.

O arranque do ambiente de testes alternativo apenas se deu quando ficou garantido que estaria funcional no longo prazo. Até isolar a causa do bloqueio frequente do Raspberry Pi devido à temperatura elevada, foram aplicadas 3 optimizações: alteração da função do relógio, para executar apenas 1 vez por minuto em vez de 1 vez por segundo; correspondência da resolução da imagem de fundo com a resolução do ecrã, para acelerar o carregamento da mesma; alteração do formato dos tipos de letra de *TrueType Font* (TTF) para *Web Open Font Format 2* (WOFF2), por ser recomendado para websites. Apesar de terem sido alterações positivas, não resolveram o problema em questão.

Por fim, com recurso a câmara térmica, observou-se que o ecrã é a maior fonte de calor do equipamento. Como o Raspberry Pi estava a menos de 1cm de distância do ecrã, acabava por também aquecer até bloquear devido à temperatura. Para corrigir esta lacuna, o SBC foi reorientado para estar mais afastado do ecrã, passando a ser verificadas temperaturas estáveis entre os 40°C e 50°C.

A comunicação foi um aspeto melhorado, onde passei a estar mais presente no ambiente da instituição e a ter interações com os funcionários onde ocasionalmente perguntava se estariam a enfrentar algum problema com o equipamento. Apesar de terem acontecido de forma informal, grande parte dos relatos foi de que estaria tudo a funcionar normalmente.

Com o decorrer de 2 semanas sem falhas a apontar e relatos de funcionários de outras respostas sociais que usaram o terminal de picagem com sucesso sem indicação para tal, o ambiente de testes foi alargado para abranger 3 respostas sociais.

## 7.2 Cumprimento dos requisitos

Recuperando a tabela de requisitos apresentada no capítulo 3, passa a ser apresentado o estado de desenvolvimento de cada um

Feature	User story	Tipo
E1F1	Enquanto presidente, quero que o terminal esteja conectado à internet, para que possa ser gerido remotamente	Realizado
E1F2	Enquanto presidente, quero que o terminal tenha alimentação de <i>backup</i> , para que se mantenha operacional durante falhas da rede de energia	Realizado
E1F3	Enquanto presidente, quero que seja possível autenticar com impressão digital, para que os funcionários possam registar a sua presença	Realizado
E1F4	Enquanto presidente, quero que seja possível autenticar com código PIN, para que os funcionários com impressão digital difícil de ler tenham uma forma alternativa de autenticação	Realizado
E2F1	Enquanto Recursos Humanos, queremos registar a impressão digital dos funcionários no sistema, para que se possam autenticar	Parcialmente realizado
E2F2	Enquanto Recursos Humanos, queremos registar uma impressão digital secundária, para que os funcionários tenham um dedo alternativo para autenticação	Parcialmente realizado
E2F3	Enquanto Recursos Humanos, queremos definir um código PIN para os funcionários, para que os funcionários com impressão digital difícil de ler tenham uma forma alternativa de autenticação	Não realizado
E2F4	Enquanto utilizador, quero autenticar-me com impressão digital, para que possa registar a hora em que compareci para início e fim de turno	Realizado
E2F5	Enquanto utilizador, quero autenticar-me com código PIN, para que possa registar a hora em que compareci para início e fim de turno de forma alternativa	Realizado
E2F6	Enquanto utilizador, quero ouvir uma sequência de beeps, para que possa rapidamente perceber o resultado da minha autenticação	Realizado
E2F7	Enquanto utilizador, quero ver informações no ecrã, para que possa ter mais detalhes do resultado da minha autenticação	Realizado
E2F8	Enquanto utilizador, quero ter informação de anomalias com picagens anteriores, para que possa verificar junto dos Recursos Humanos	Realizado
E2F9	Enquanto administrador de sistemas, quero que as credenciais sejam encriptadas, para que os dados se mantenham seguros em caso de acesso indevido	Realizado

E2F10	Enquanto programador, quero <i>logs</i> da aplicação, para que possa entender e replicar o que aconteceu quando um <i>bug</i> é reportado.	Realizado
E3F1	Enquanto Recursos Humanos, queremos registar funcionários, para que estes possam usar o sistema	Obrigatório
E3F2	Enquanto Recursos Humanos, queremos carregar escalas de trabalho, para que a aplicação possa saber quando é esperado que o funcionário faça o seu registo de assiduidade	Realizado
E3F3	Enquanto Recursos Humanos, queremos descarregar um relatório de horas de cada funcionário para um dado mês, para que possamos processá-lo e pagar salários	Realizado
E3F4	Enquanto Recursos Humanos, queremos registar códigos de trabalho, para que seja possível identificar o tipo de presença ou ausência esperada do trabalhador	Realizado
E3F5	Enquanto Recursos Humanos, queremos registrar cargos, para que seja possível representar a nossa estrutura institucional na aplicação	Realizado

Table 7.1: *User Stories*

Todos os requisitos foram realizados de acordo como inicialmente planeado, à exceção de E2F1, E2F2 e E2F3. No caso de E2F1 e E2F2, o desenvolvimento não foi concluído porque no único momento em que o registo de impressão digital foi necessário, tornou-se muito mais rápido fazê-lo diretamente no terminal Linux e com mais informação em caso de erro. No caso de E2F3, não foi desenvolvido pois os códigos PIN foram importados juntamente com outras informações dos funcionários no momento de realização dos ambientes de testes.

Salienta-se que as *user stories* E3F4 e E3F5 foram criadas posteriormente por não terem sido identificadas juntamente com as restantes, mas serem funcionalidades necessárias das quais dependem outras componentes da aplicação. Já a *user story* E2F10 foi criada posteriormente por constituir funcionalidade de grande utilidade para os ambientes de testes.

# **8 - Conclusão**

## **8.1 Conclusão**

Este projeto mostra as diversas nuances que podem surgir do desenvolvimento de uma solução informática, divulgadas nas diversas Unidade Curriculares (UCs longplural) do curso, mas agora sentidas na primeira pessoa. Mostra também que a tecnologia não é exclusivamente a solução para qualquer problema. No caso do incumprimento por falta de picagem, não foram verificadas alterações face ao que acontece atualmente, fenómeno cujas explicações possíveis podem ser a exigência de picar no sistema atual e no protótipo durante os ambientes de testes, existindo esquecimento de picar num, mas não no outro, ou a indicação de que houve uma anomalia não ser acompanhada do motivo para tal. Mesmo a indicação de anomalias ser uma funcionalidade positiva, terão de estar associadas medidas de negócio que sejam mais rígidas nesta matéria.

De todos os objetivos, aquele que no imediato pode ser dado como alcançado é a criação de um sistema de gestão de assiduidade que de facto será usado na instituição. Os restantes são dados como estando em curso visto que precisam de mais trabalho.

O resultado do trabalho pode não refletir a visão completa idealizada ao longo de vários meses a observar os problemas com os processos internos da entidade parceira, ou a proposta de projeto inicialmente apresentada para o Trabalho Final de Curso, mas constitui uma solução viável para um problema real com perspetiva de poder ser continuamente melhorada.

## **8.2 Trabalhos futuros**

Como mencionado no ponto anterior, este projeto não completa a visão inicial, pelo que existe trabalho futuro a fazer. Sendo o objetivo colocar a solução desenvolvida em ambiente de produção quando a entidade parceira inaugurar a 2<sup>a</sup> fase das suas instalações, terá de ser concluída a funcionalidade deixada como pendente e garantir paridade de funcionalidades com a metodologia atualmente utilizada para controlo de assiduidade. Poderão também ser incluídas as sugestões dadas no inquérito realizado e *feedback* transmitido, tal como melhorias de *Qualidade de Vida* identificadas durante o desenvolvimento.

Após as metas referidas serem alcançadas, será dada continuidade à expansão de funcionalidades, começando com o suporte para múltiplos terminais de registo pontométrico e migração da aplicação web e base de dados para um contentor docker, possibilitando a sua instalação num servidor dedicado. Posteriormente o terminal será reconstruído com base em microcontrolador, retirando as lições aprendidas do protótipo desenvolvido, de modo a dar a solução anterior como desativada. Em diante serão implementadas funcionalidades inicialmente planeadas.

# Bibliografia

- [Mat24] João P. Matos-Carvalho. *The Lusófona L<sup>A</sup>T<sub>E</sub>X Template User's Manual*. Lusófona University. 2024. URL: <https://github.com/jpmcarvalho/UL-Thesis>.
- [Inc24] Xthings Inc. *CrossChex access control and time management software solution*. 2024. URL: <https://www.anviz.com/CrossChex.html> (visited on 12/2024).
- [Ltd24a] Raspberry Pi Ltd. *Raspberry Pi Zero 2 W Product Brief*. 2024. URL: <https://datasheets.raspberrypi.com/rpizero2/raspberry-pi-zero-2-w-product-brief.pdf> (visited on 11/2024).
- [Ltd24b] Raspberry Pi Ltd. *Raspberry Pi 3 Model A+ Product Brief*. 2024. URL: <https://datasheets.raspberrypi.com/rpi3/raspberry-pi-3-a-plus-product-brief.pdf> (visited on 11/2024).
- [Ltd24c] Raspberry Pi Ltd. *Raspberry Pi 4 Model B Product Brief*. 2024. URL: <https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-product-brief.pdf> (visited on 11/2024).
- [Ltd24d] Raspberry Pi Ltd. *Raspberry Pi 5 Product Brief*. 2024. URL: <https://datasheets.raspberrypi.com/rpi5/raspberry-pi-5-product-brief.pdf> (visited on 11/2024).
- [Lim24] Waveshare International Limited. *5inch DSI/LCD - Waveshare Wiki*. 2024. URL: [https://www.waveshare.com/wiki/5inch\\_DSI\\_LCD](https://www.waveshare.com/wiki/5inch_DSI_LCD) (visited on 11/2024).
- [Yir+23] Xuyang Li Yirong Yu Qiming Niu et al. "A Review of Fingerprint Sensors: Mechanism, Characteristics and Applications." In: *Micromachines (Basel)* (2023). URL: <https://pmc.ncbi.nlm.nih.gov/articles/PMC10305017/>.
- [Han15] Ltd. Hangzhou Synochip Data Security Technology Co. *AS608 Processor Datasheet*. 2015. URL: [https://electronic724.com/wp-content/uploads/6814\\_synochip\\_as608.pdf](https://electronic724.com/wp-content/uploads/6814_synochip_as608.pdf) (visited on 11/2024).
- [Han19] Ltd Hangzhou Grow Technology Co. *R307 Fingerprint Module User Manual*. 2019. URL: <https://github.com/RalphBacon/Arduino-Fingerprint-Sensor-With-Lock/blob/e7cb40ce89dd121bc551251bd12e74ee51058766/R307%20fingerprint%20module%20user%20manual.pdf> (visited on 11/2024).
- [Ltd18] ADH Technology Co. Ltd. *Datasheet GT-521FX2*. 2018. URL: [https://cdn.sparkfun.com/assets/learn\\_tutorials/7/2/3/GT-521FX2\\_datasheet\\_V1.1\\_003\\_.pdf](https://cdn.sparkfun.com/assets/learn_tutorials/7/2/3/GT-521FX2_datasheet_V1.1_003_.pdf) (visited on 11/2024).
- [Esp25] Ltd. Espressif Systems (Shanghai) Co. *ESP32 S3 Datasheet*. 2025. URL: [https://www.espressif.com/sites/default/files/documentation/esp32-s3\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32-s3_datasheet_en.pdf) (visited on 04/2025).
- [Ltd25] Raspberry Pi Ltd. *Raspberry Pi Pico 2 Product Brief*. 2025. URL: <https://datasheets.raspberrypi.com/pico/pico-2-product-brief.pdf> (visited on 04/2025).
- [PJR25] LLC PJRC.COM. *Teensy® 4.0*. 2025. URL: <https://www.pjrc.com/store/teensy40.html> (visited on 04/2025).

# Glossário

**ARPI** Associação de Reformados, Pensionistas e Idosos. 11, 21

**ARPI S. J. Tojal** Associação de Reformados, Pensionistas e Idosos de São Julião do Tojal. 1, 8, 10, 35

**ATL** Atividades de Tempos Livres. 8

**BD** Base de dados. 16

**BT** *Bluetooth*. 32

**CRUD** *Create, Read, Update and Delete*. 16

**CSV** *Comma Separated Values*. 24

**DC** *Direct Current*. 11, 36

**DSI** *Display Serial Interface*. 35

**ERPI** Estrutura Residencial para Pessoas Idosas. 8

**FAR** *False Acceptance Rate*. 35, 36

**FRR** *False Rejection Rate*. 36

**GPIO** *General Purpose Input/Output*. 34

**PdC** Prova de Conceito. 16, 22

**PDF** *Portable Document Format*. 8

**picagem** Registo de início ou fim de trabalho. 8, 16, 18, 32, 34

**PIN** Número de identificação pessoal. 12, 13, 17, 22, 24, 26, 27, 32, 34

**QdV** *Qualidade de Vida*. 10, 28

**QoL** *Quality of Life*. 21

**RFID** *Radio-frequency identification*. 11, 32

**RH** Recursos Humanos. 8, 10, 12, 13, 19, 25–27

**RTC** *Real Time Clock*. 34, 35, 37

**SaaS** *Software as a Service*. 32

**SBC** *Single Board Computer*. 23, 25, 34, 35, 37

**SGBD** Sistema de Gestão de Bases de Dados. 17, 32

**sistema embebido** Sistema computacional resultante da integração de hardware com software desenvolvido para um caso de uso específico. 34

**SoC** *System on a Chip.* 17, 22, 23

**SPI** *Serial Peripheral Interface.* 37

**TFC** Trabalho Final de Curso. 11, 28

**TTF** *TrueType Font.* 23, 25

**UART** *Universal asynchronous receiver/transmitter.* 37

**UC** Unidade Curricular. 18, 21, 28

**UI/UX** *User Interface/User Experience.* 2, 3, 18

**WOFF2** *Web Open Font Format* 2. 23, 25

# A - Análise do sistema Anviz CrossChex

A Anviz dispõe um ecossistema de terminais para efeitos de gestão de assiduidade ou controlo de acesso, que permitem o funcionário autenticar-se através de PIN, cartão *Radio-frequency identification* (RFID), biometria (dedo ou palma da mão), reconhecimento facial ou emparelhamento à aplicação para Android e iOS via *Bluetooth* (BT). Para controlar estes equipamentos existe o software CrossChex [Inc24], fornecido de forma gratuita nas vertentes Standard, instalado e utilizado localmente numa máquina Windows, e Cloud, fornecido enquanto *Software as a Service* (SaaS), acessível através de qualquer *browser*.

Analizando a aplicação para *desktop*, Crosschex Standard, ao abri-la pela primeira vez é visível um *dashboard* com estatísticas pertinentes para quem usa a aplicação (Estado dos terminais, situação de picagens e, na parte inferior, menu de atalhos. Na parte superior, é visível o friso, método de navegação característico da *suite Microsoft Office*, com todas as funcionalidades.

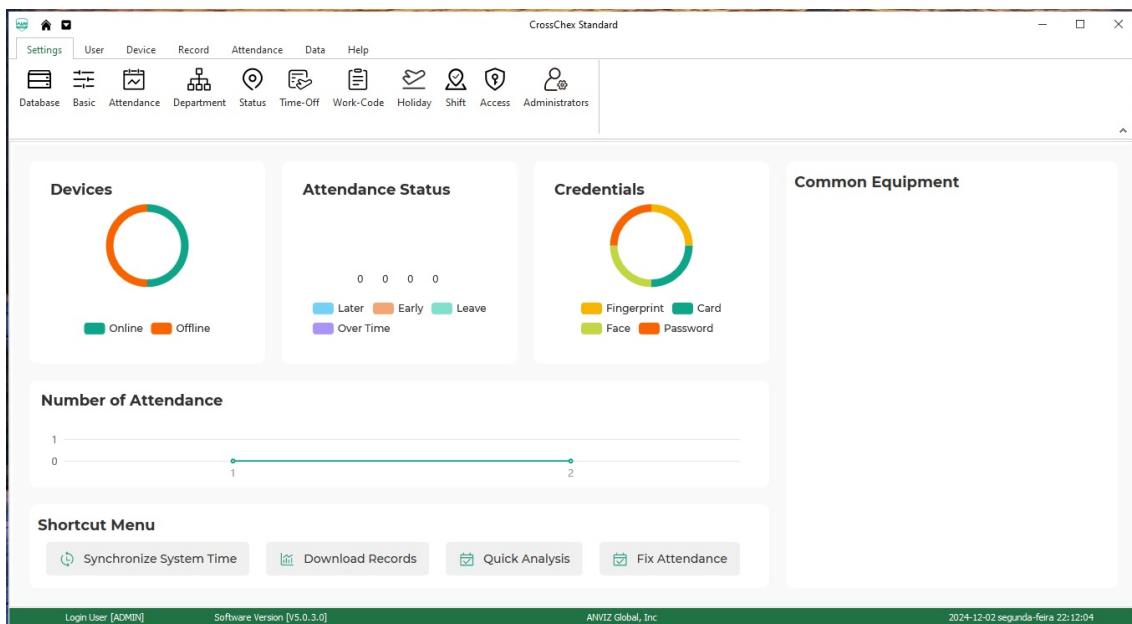


Figure A.1: Primeiro ecrã mostrado ao abrir a aplicação CrossChex

Acedendo a *Settings* -> *Database*, são listados os Sistema de Gestão de Bases de Dados (SGBD) disponíveis, Microsoft Access e Microsoft SQL Server. Nas restantes opções do separador *Settings*, constam informações que dependem do país e do negócio, como o caso dos feriados e unidades orgânicas, no entanto são fáceis de inserir.

Já a gestão de utilizadores é apresentada numa janela simples, onde consta a listagem de todos os funcionários existentes ou que correspondem ao filtro aplicado (departamento, número/nome do funcionário) e na lateral direita o formulário de edição dos seus atributos.

Na gestão de escalas, a interface deixa claro onde é que cada ação deve ser realizada, no entanto qualquer tarefa torna-se impossível de realizar de forma intuitiva já que ao tentar, por exemplo, introduzir um horário através dos campos *Timetable 1 a 4*

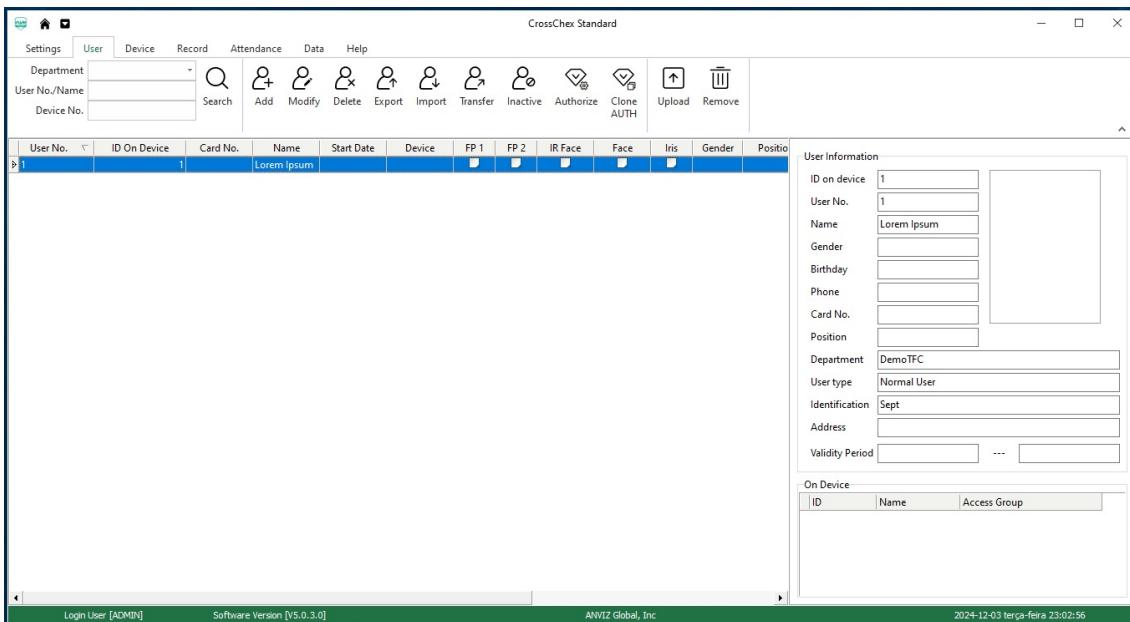


Figure A.2: Ecrã User -> User Scheduling mostrado na aplicação CrossChex

não tem o resultado pretendido. Para o tipo de dados a inserir, poderia ter sido adotado um método de introdução gráfico, como uma *sliding bar*, em vez de apenas campos de texto.

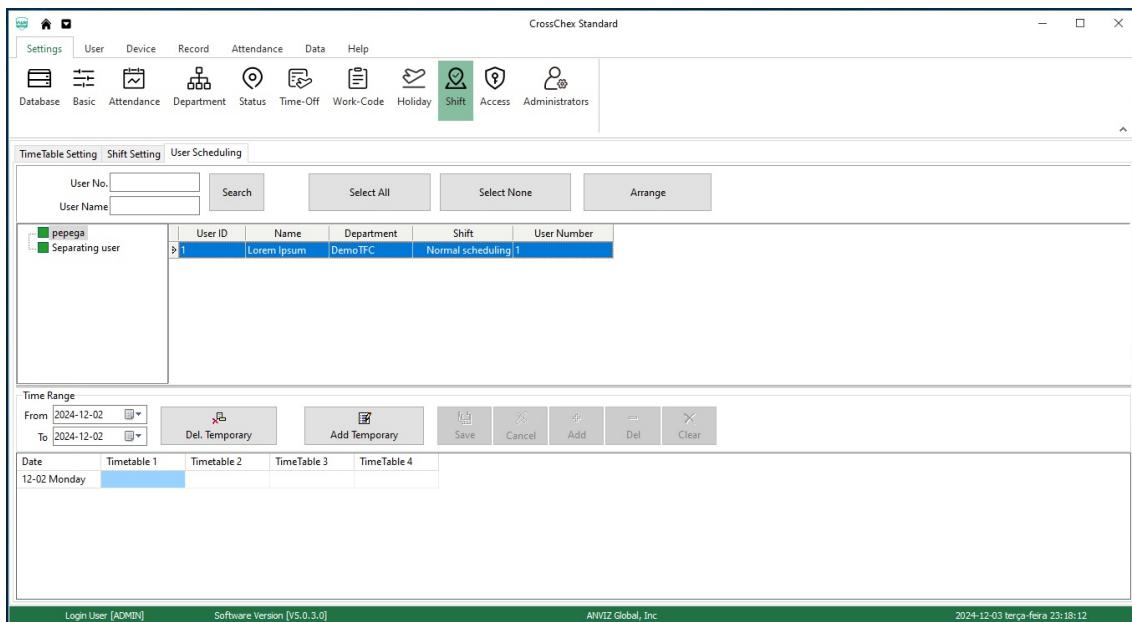


Figure A.3: Ecrã Shifts mostrado na aplicação CrossChex

## **B - Exploração de *hardware* para o terminal de picagem**

Em termos de *hardware*, um terminal de picagem é um sistema embebido. Em moldes gerais, o equipamento precisa de ter ligação à corrente elétrica e rede de dados, uma forma de *input* para o funcionário se puder autenticar perante o terminal e um mecanismo de feedback da autenticação. Para o contexto a abordar neste projeto, é necessário em específico:

- Sensor de impressão digital, para permitir a recolha de biometria;
- Ecrã para visualização da interface, transmitir feedback visual da picagem e informações de irregularidades em dias anteriores;
- Teclado numérico com setas de navegação ou ecrã tátil, para permitir a introdução de PIN e navegação na interface;
- Coluna para transmitir feedback sonoro da navegação na interface ou da picagem;
- *Real Time Clock* (RTC), para que a data e hora se mantenham corretas na próxima vez que o terminal volte a ser ligado à corrente elétrica;
- Bateria, interna ou externa, para que o terminal continue operacional durante uma falha elétrica no edifício onde está instalado.

Estando identificada a lista de componentes necessários para o terminal, passam a ser identificadas as soluções possíveis de construção do equipamento.

### **B.1 Terminal de picagem com recurso a microprocessador**

Considerando o tamanho que o terminal poderá ter e a necessidade de conectar dispositivos que são desenvolvidos com propósito de ligar a um microcontrolador, a base de *hardware* de desenvolvimento tem de ser compacta e capaz de efetuar ligações por *General Purpose Input/Output* (GPIO). Dado que o cenário de utilização do sistema embebido é altamente previsível e limitado, o nível de *performance* do *hardware* não é fator eliminatório. Assim, identifica-se que a base de *hardware* mais adequada é um *Single Board Computer* (SBC), sendo o mais comum o Raspberry Pi.

Com o objetivo de maximizar a diversidade de conexões para que todos os componentes possam ser ligados diretamente ao SBC, passam a ser listados os modelos em consideração:

Modelo	Pi Zero 2W [Ltd24a]	Pi 3A+ [Ltd24b]	Pi 4B [Ltd24c]	Pi 5 [Ltd24d]
Porta Ethernet	Não	Não	Sim	Sim
Porta USB	1x micro-USB 2.0	1x USB 2.0	2x USB 2.0, 2x USB 3.0	2x USB 2.0, 2x USB 3.0
Conecotor DSI	Não	Sim	Sim	Sim
RTC	hat GPIO	hat GPIO	hat GPIO	Conecotor específico
Alimentação	5V/2.5A	5V/2.5A	5V/3A	5V/5A
Preço	18€	30€	40€ (1GB RAM)	57€ (2GB RAM)

Table B.1: Especificações dos SBCs Raspberry Pi

Destes quatro SBC, o foco incide nos 2 que têm todas as conexões necessárias, Raspberry Pi 4 e Pi 5. Como a única diferença relevante é o *Real Time Clock* (RTC), que se for um *hat GPIO* exige configuração no sistema operativo, a escolha será o Raspberry Pi 5 por ser maioritariamente *plug and play*.

Quanto ao ecrã, será utilizado um ecrã tátil ao invés de ecrã e teclado numérico devido ao tamanho das opções para teclado numérico existentes no mercado. Como já foi identificado o conector *Display Serial Interface* (DSI) na tabela, que é específico para ecrãs que existem em dimensões a partir de 2.8", a escolha será o Waveshare 5inch DSI LCD [Lim24].

Passando para o sensor de impressão digital, os equipamentos existentes no mercado recorrem à tecnologia ótica ou capacitiva. No caso do sensor óptico, a biometria é obtida através da projeção de luz num prisma, paralelo à superfície onde o dedo está colocado, sendo projetada para uma câmara. Tem como vantagem o baixo custo dada a simplicidade do princípio de funcionamento, mas com a desvantagem de ter *False Acceptance Rate* (FAR) mais alta. Já o sensor capacitivo recorre às propriedades dos semicondutores para, através de um *array* de eletrólitos que ao toque permite a passagem de corrente elétrica, e desta forma, obtém a biometria pela identificação dos eletrólitos onde existe corrente. Tem como principal vantagem o tamanho reduzido e a menor *False Acceptance Rate* (FAR), mas o custo é mais elevado e a superfície do sensor é bastante sensível a danos e à água. [Yir+23]

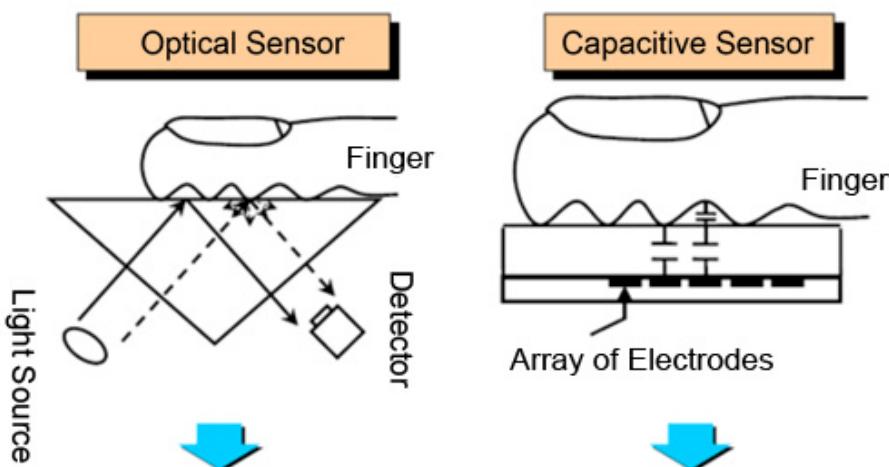


Figure B.1: Funcionamento do sensor de impressão digital óptico (esquerda) e capacitivo (direita)

O sensor de impressão digital terá de ser capaz de armazenar pelo menos 120 modelos de biometria, igual ao número esperado de trabalhadores na ARPI S. J. Tojal

no curto prazo. Isto exclui os sensores capacitivos, que apenas têm capacidade para 80 a 100 modelos. Pesquisando apenas por sensores óticos, é possível encontrar vários idênticos, onde apenas muda o microcontrolador, que influencia maioritariamente a capacidade de modelos:

	AS608 [Han15]	R307 [Han19]	GT521F52 [Ltd18]
Capacidade de modelos	300	1000	3000
Cor LED	Azul	Verde	Branco
<i>False Acceptance Rate</i> (FAR)	<0.001%	<= 0.0001%	<0.001%
<i>False Rejection Rate</i> (FRR)	<1.0%	<0.1%	<0.1%
Alimentação Direct Current (DC)	DC 3.3V	DC 3.8-7.0V	DC 3.3-6V
Custo	28€	33€	55€

Table B.2: Especificações dos sensores de impressão digital óticos

Considerando que qualquer um destes sensores satisfaz o critério mencionado acima e as diferenças no FAR e *False Rejection Rate* (FRR) são desrespeitáveis dado que se trata de uma percentagem bastante pequena e que a consequência de qualquer uma destas situações acontecer não resulta numa falha crítica (ao contrário de, por exemplo, um cenário onde o sensor de impressão digital é usado para acesso a um espaço restrito), o sensor escolhido será o AS608, por ter o menor custo.



Figure B.2: Sensor de impressão digital ótico AS608

Passando para o audio, usado para fornecer *feedback* dos *inputs*, visto que o pretendido é reproduzir sons além de um simples beep, a solução mais simples é uma coluna de som pequena ligada por USB, em particular a da imagem abaixo devido às suas dimensões.



Figure B.3: Coluna USB

Quanto à alimentação, uma vez que existe uma fonte de alimentação oficial para a

base de hardware a usar, essa será a opção escolhida.

No caso do terminal com recurso a microprocessador, as vantagens estão no facto de o poder de processamento ser bastante superior ao necessário, existir sistema operativo, que torna o modo de desenvolver aplicações igual a qualquer outro computador moderno e a diversidade de conectores tipicamente usados em microcontroladores que podem ser usados sem configuração (*plug and play*). As desvantagens são o custo total do *hardware* e a impossibilidade de expor apenas os conectores necessários, isto é, portas do SBC que não são necessárias têm de ficar expostas de qualquer modo, que pode facilitar a adulteração do equipamento.

## B.2 Terminal de picagem com recurso a microcontrolador

Como já referido anteriormente, as execuções de um terminal de picagem são altamente previsíveis, pelo que usar um microcontrolador como base funcional é viável, desde que seja estimado o desempenho necessário. Sendo as tarefas são maioritariamente sequenciais, no sentido em que apenas há uma pessoa a efetuar uma picagem de cada vez, os microcontroladores genéricos são suficientes.

Por serem uma base mais básica, é preciso definir primeiro que interfaces serão necessárias. Após pesquisa dos módulos RTC, ethernet e ecrãs táteis mais comuns, verifica-se que DS3231 comunica por I<sup>2</sup>C, W5500 comunica por *Serial Peripheral Interface* (SPI) e o ecrã tátil ILI9348 comunica por I<sup>2</sup>C. Acrescentando o sensor de impressão digital AS608 que comunica por *Universal asynchronous receiver/transmitter* (UART), são necessários 1 barramento UART, 1 barramento I<sup>2</sup>C e 1 barramento SPI, visto que os 2 últimos suportam várias comunicações por barramento.

Com o objetivo de maximizar a diversidade de conexões para que todos os componentes possam ser ligados diretamente ao SBC, passam a ser listados os modelos em consideração:

Modelo	ESP32 S3 [Esp25]	Pi Pico 2 [Ltd25]	Teensy 4 [PJR25]
Tipo de processador	Xtensa LX7	Cortex M33/Hazard3	Cortex M7
Memória Flash	8MB	4MB	2MB
Barramentos UART	3	2	7
Barramentos SPI	2	2	2
Barramentos I <sup>2</sup> C	2	2	3
RTC incorporado	Sim	Não	Sim
Preço	12€	7€	30€

Table B.3: Especificações dos microcontroladores selecionados

Uma vez que todos os microcontroladores listados têm um número de barramentos superior ao listado, o fator eliminatório será o preço. Desta forma, o microcontrolador selecionado será o Raspberry Pi Pico 2.

Em linha com os componentes a usar na opção com microprocessador, o sensor de impressão digital será o mesmo. Já o ecrã será o ILI9348, tátil de 2.8"; para substituir a coluna USB será usado um buzzer; e como fonte de alimentação será usado um transformador genérico de 5V 2A.

No caso do terminal com recurso a microcontrolador, as vantagens estão no custo total do *hardware* e no dimensionamento do equipamento exatamente às necessidades do projeto. Enquanto desvantagens, prendem-se com a dificuldade de programação, porque não existe sistema operativo e portanto todas as abstrações que fornece teriam de ser programadas; e a necessidade de construir a base física para ligar os

componentes compacta o suficiente para ter dimensões semelhantes ao terminal de picagem atualmente instalado.

## **C - Orçamento para terminal de picagem**

# Orçamento Aquisição Protótipo TFC

Nome	Preço un.	Qtd	Preço total
Raspberry Pi 5 2GB (Mauser.pt)	56,99 €	1	56,99 €
Bateria de backup RTC (Mauser.pt)	5,99 €	1	5,99 €
Sensor de impressão digital AS608 (botnroll.com)	28,55 €	1	28,55 €
Ecrã tátil 5" 800x480 DSI (botnroll.com)	42,90 €	1	42,90 €
Cabo FPC adaptador de display (Mauser.pt)	1,17 €	1	1,17 €
Cartão microSD 32GB Raspberry Pi (Mauser.pt)	5,75 €	1	5,75 €
Fonte de Alimentação para Raspberry Pi 5 (Mauser.pt)	13,30 €	1	13,30 €
Coluna USB (Worten.pt)	10,16 €	1	10,16 €
Conecotor NSR (Mauser.pt)	0,10 €	10	1,00 €
Terminal fêmea para conector NSR (Mauser.pt)	0,10 €	10	1,00 €
<b>Subtotal:</b>			<b>166,81 €</b>

**Subtotal:** **166,81 €**

**Portes de Envio Mauser.pt:** **0,00 €**

**Portes de Envio botnroll:** **0,00 €**

**Portes de Envio Worten.pt:** **0,00 €**

**Total:** **166,81 €**

## Notas

Todos os valores apresentados tem IVA incluído.

Preços consultados a 12/12/2024

## **D - Inquérito realizado**

# Aplicação de gestão de assiduidade

No âmbito do meu trabalho final de curso, encontro-me a desenvolver uma solução para a gestão de assiduidade a implementar na ARPI de São Julião do Tojal. Poderia ajudar-me com o preenchimento deste inquérito? Não demorará mais de 3 minutos!

\* Obrigatória

## Contextualização

1. Há quantos anos trabalha na instituição? \*

- Menos de 1 ano
- 1 a 3 anos
- 4 a 5 anos
- 5 ou mais anos

2. Em que resposta social desempenha funções (caso seja várias, escolha a principal) \*

- Apoio domiciliário
- Estrutura Residencial para Idosos
- Centro de Dia
- Atividades e Tempos Livres
- Serviços administrativos
- Outro

3. Para as funções que desempenha, usa o MySenior? \*

- Sim
- Não

## Sistema atual

4. Quando foi a última vez que teve de verificar uma anomalia relativamente às suas picagens ou pedidos/justificações em falta com os Recursos Humanos? \*

Menos de 1 mês

1 a 3 meses

4 meses a 1 ano

Mais de 1 ano

5. Caso tenha respondido "Menos de 1 mês", "1 a 3 meses" ou "4 meses a 1 ano" na questão anterior, a que se deveu essa anomalia?

Picagem em falta

Pedido/Justificação não entregue

Outro

6. Qual é o seu grau de satisfação com a forma como o registo de assiduidade é feito atualmente? \*

0	1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	---	----

Pouco satisfeito(a)

Muito satisfeito(a)

7. O que destaca de positivo do funcionamento de registo de assiduidade atual?

8. O que destaca de negativo do funcionamento de registo de assiduidade atual?

## Propostas de implementação para a nova solução

9. Relativamente à pertinência de funcionalidades acessíveis por aplicação web (no computador, telemóvel ou tablet), responda às seguintes perguntas \*

	Não pertinente	Pouco pertinente	Neutro	Algo pertinente	Muito per
Consultar o relatório de horas trabalhadas	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Submeter pedidos de alteração de escala ou justificações	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Consultar a minha escala de trabalho via aplicação web	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

10. Relativamente à pertinência de funcionalidades no terminal de picagem, responda às seguintes perguntas \*

	Não pertinente	Pouco pertinente	Neutro	Algo pertinente	Muito per
Poder associar mais que um dedo para fazer a picagem	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Receber feedback sonoro e visual de que a picagem foi bem/mal registada	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
Ter informação, no ecrã e por código de beeps, que existem anomalias com as minhas picagens de dias anteriores	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>

11. Para si, quais são as funcionalidades mais importantes? Ordene de forma decrescente (mais importante primeiro) \*

Consultar o relatório de horas trabalhadas

Submeter pedidos de alteração de escala ou justificações

Consultar a minha escala de trabalho via aplicação web

Poder associar mais que um dedo para fazer a picagem

Receber feedback sonoro e visual de que a picagem foi bem/mal registada

Ter informação, no ecrã e por código de beeps, que existem anomalias com as minhas picagens de dias anteriores

12. Existe alguma sugestão ou comentário que pretenda acrescentar?

---

Este conteúdo não foi criado nem é aprovado pela Microsoft. Os dados que submeter serão enviados para o proprietário do formulário.



Microsoft Forms

## **E - Resultados do inquérito realizado**

# Aplicação de gestão de assiduidade

42 Respostas

07:32 Tempo médio de conclusão

Fechado Estado

1. Há quantos anos trabalha na instituição?

Menos de 1 ano	7
1 a 3 anos	11
4 a 5 anos	4
5 ou mais anos	20

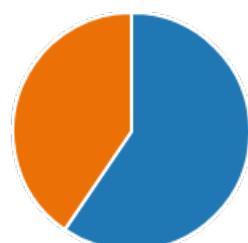


2. Em que resposta social desempenha funções (caso seja várias, escolha a principal)

Apoio domiciliário	12
Estrutura Residencial para Idosos	19
Centro de Dia	3
Atividades e Tempos Livres	1
Serviços administrativos	1
Outro	6

3. Para as funções que desempenha, usa o MySenior?

Sim	25
Não	17



4. Quando foi a última vez que teve de verificar uma anomalia relativamente às suas picagens ou pedidos/ justificações em falta com os Recursos Humanos?

Menos de 1 mês	11
1 a 3 meses	5
4 meses a 1 ano	5
Mais de 1 ano	21



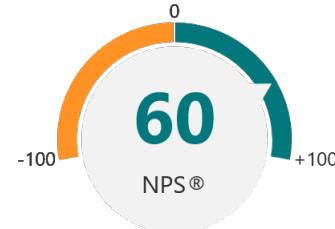
5. Caso tenha respondido "Menos de 1 mês", "1 a 3 meses" ou "4 meses a 1 ano" na questão anterior, a que se deveu essa anomalia?

● Picagem em falta	18
● Pedido/Justificação não entregue	5
● Outro	3



6. Qual é o seu grau de satisfação com a forma como o registo de assiduidade é feito atualmente?

Promotores	28
Passivos	11
Detratores	3



7. O que destaca de positivo do funcionamento de registo de assiduidade atual?

17  
Respostas

Respostas Mais Recentes  
"É fácil e rápido "

1 inquiridos (6%) responderam **fácil controlo** a esta pergunta.

**folgas** **bolsos** **ponto** **problemas**  
**dedo** **picagem** **fácil** **controlo** **cartão**  
**registro** **metodologia** **argur**

8. O que destaca de negativo do funcionamento de registo de assiduidade atual?

14  
Respostas

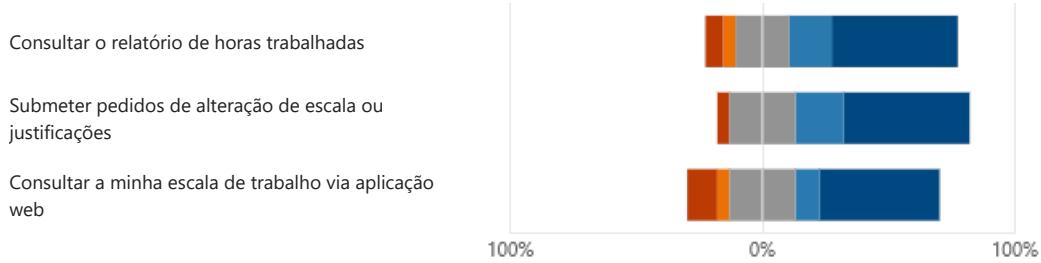
Respostas Mais Recentes  
"Não aceita a impressão digital "

6 inquiridos (43%) responderam **Nada** a esta pergunta.

**colaboradores**  
ok dúvidas  
cores vivas  
minuto 8h serviço  
**Problemas**  
**Nada** informação  
tempo Falt  
imp  
assidui

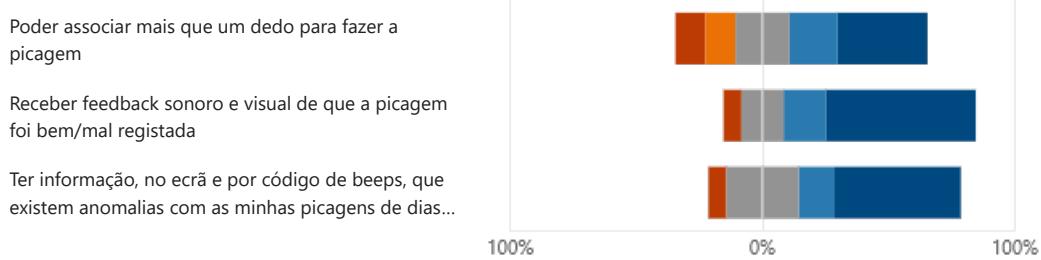
9. Relativamente à pertinência de funcionalidades acessíveis por aplicação web (no computador, telemóvel ou tablet), responda às seguintes perguntas

■ Não pertinente ■ Pouco pertinente ■ Neutro ■ Algo pertinente ■ Muito pertinente



10. Relativamente à pertinência de funcionalidades no terminal de picagem, responda às seguintes perguntas

■ Não pertinente ■ Pouco pertinente ■ Neutro ■ Algo pertinente ■ Muito pertinente



11. Para si, quais são as funcionalidades mais importantes? Ordene de forma decrescente (mais importante primeiro)



12. Existe alguma sugestão ou comentário que pretenda acrescentar?

8  
Respostas

Respostas Mais Recentes

**2** inquiridos (25%) responderam **Nao** a esta pergunta.

**Nao**