



UNIVERSIDADE
LUSÓFONA

Aplicação de controlo de assiduidade

Trabalho Final de Curso

Relatório Intercalar 1º Semestre

Aluno: Daniel Martins

Orientador: Professor Rodrigo Correia

Trabalho Final de Curso | LEI | 2024-12-03

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)LaTeX e o modelo ULThesis (v1.0.0) [Mat24].

Resumo

Abstract

Índice

Resumo	2
Abstract	3
Índice	4
Lista de Figuras	5
Lista de Tabelas	6
1 Identificação do Problema	7
2 Benchmarking	8
2.1 Anviz CrossChex	8
3 Viabilidade e Pertinência	10
4 Solução Proposta	11
4.1 Exemplos de cenários melhorados	11
4.1.1 Permuta de turno sem formalização	11
4.1.2 Processamento do subsídio de natal	11
4.2 Prova de conceito	11
4.3 Requisitos de <i>hardware</i>	12
4.3.1 Terminal de picagem já existente no mercado, com equipamento para fornecer informação extra	12
4.3.2 Desenvolvimento do terminal de picagem com recurso a micropro- cessador	13
4.3.3 Desenvolvimento do terminal de picagem com recurso a microcon- trolador	17
4.3.4 Escolha final do <i>hardware</i>	17
4.4 Requisitos de <i>software</i>	17
5 Calendário	19
5.1 Primeira fase: extensão da prova de conceito	19
5.2 Segunda fase: componente lógica e de automatização	19
5.3 Terceira fase: documentos e <i>bug fixing</i>	19
Bibliografia	20
Glossário	21

Lista de Figuras

2.1	Primeiro ecrã mostrado ao abrir a aplicação CrossChex	8
2.2	Ecrã <i>User -> User Scheduling</i> mostrado na aplicação CrossChex	9
2.3	Ecrã <i>Shifts</i> mostrado na aplicação CrossChex	9
4.1	Placa de desenvolvimento LilyGo T-Display-S3, frente (esquerda) e verso (direita)	13
4.2	Diagrama de integração para fornecer feedback ao trabalhador após a picagem	13
4.3	Ecrã tátil Waveshare 5inch DSI LCD	15
4.4	Funcionamento do sensor de impressão digital ótico (esquerda) e capacitivo (direita)	15
4.5	Sensor de impressão digital ótico AS608	16
4.6	Coluna USB	16
4.7	Fonte de alimentação oficial para Raspberry Pi 5	17
4.8	Arquitetura para a solução escolhida	18

Lista de Tabelas

4.1	Especificações dos SBCs Raspberry Pi	14
4.2	Especificações dos sensores de impressão digital óticos	16

1 - Identificação do Problema

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 Idosos (ERPI). Perante o aumento do número de trabalhadores necessário para a instituição funcionar, resultante da abertura da ERPI, foi instalado num novo sistema de controlo de assiduidade que além de suportar mais funcionários registados, também permite o tratamento automático dos registos de início e fim de trabalho (picagem).

Até essa altura, a tarefa de identificar o vencimento mensal do trabalhador seguia a seguinte metodologia: perto do final do mês, na aplicação de gestão dos terminais de picagem, eram exportados para *Portable Document Format* (PDF) todos os registos existentes para o mês corrente. Recorrendo ao *template* criado em folha de cálculo, pré-preenchida com o nome e número dos funcionários, era inserido manualmente o horário de trabalho para cada dia do mês, de acordo com a escala publicada pela chefia da valência a que pertence, e as horas das picagens do trabalhador. Em caso de anomalia, eram verificados os pedidos no âmbito do trabalho (permuta 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. 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 ser superior a 1 dia de trabalho. Por fim, as informações de horas obtidas são transcritas para uma folha de registo de alterações mensais (em papel), isto é, horas extra, faltas e subsídios de turno, referente a cada trabalhador para poder constar em arquivo. Por fim, é feita a transcrição das alterações mensais, isto é, faltas (dias de falta, baixas médicas ou baixas de parto), remunerações (subsídios ou trabalho suplementar) e descontos (sindicato ou penhoras de vencimento), referente a cada funcionário para inserir no *software* de processamento de salários e, posteriormente, constar em arquivo em papel.

Após a instalação do novo sistema de controlo de assiduidade, a metodologia usada continuou a ser a descrita no parágrafo anterior, devido à quantidade de informação necessária de inserir à data de instalação do novo sistema (todas as escalas, falta, férias para cada trabalhador) e à interface gráfica pouco intuitiva, que torna difícil introduzir e manipular essas informações. Existindo uma aplicação adquirida para resolver um problema, parcialmente ou totalmente, que se encontra subaproveitada suscitou interesse da minha parte para criar algo que fosse ao encontro do pretendido.

2 - Benchmarking

Antes de propor o presente tema enquanto Trabalho Final de Curso, foi efetuada consulta de mercado para perceber se alguma das soluções existentes se demonstra adequada para as necessidades da instituição. Passam a ser listadas aquelas que foram analisadas em mais detalhe

2.1 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 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*.

Analisando 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, ações recorrentes. Na parte superior, é visível o friso, método de navegação característico da *suite* Microsoft Office, com todas as funcionalidades.

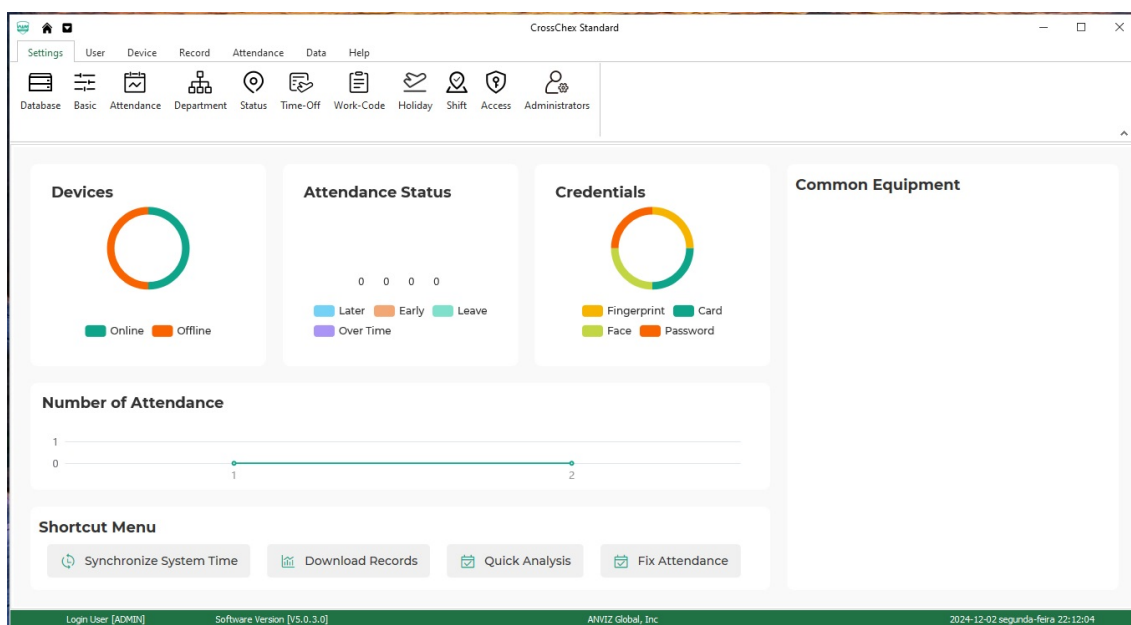


Figure 2.1: Primeiro ecrã mostrado ao abrir a aplicação CrossChex

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*, estão informações que dependem do país e do negócio, como o caso dos feriados e departamentos, no entanto são fáceis de inserir.

Já a gestão de utilizadores é apresentada numa janela simples, onde consta a listagem de todos os utilizadores existentes ou que correspondem ao filtro aplicado (departamento,

número/nome do funcionário) e na lateral direita o formulário de edição dos seus atributos.

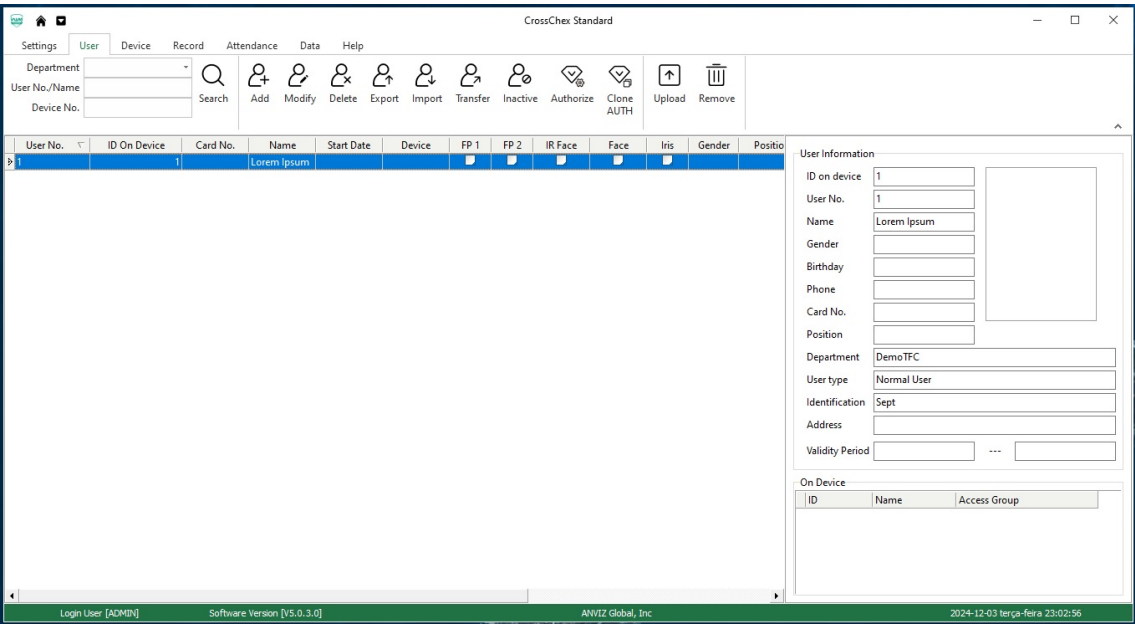


Figure 2.2: Ecrã *User* -> *User Scheduling* mostrado na aplicação CrossChex

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 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.

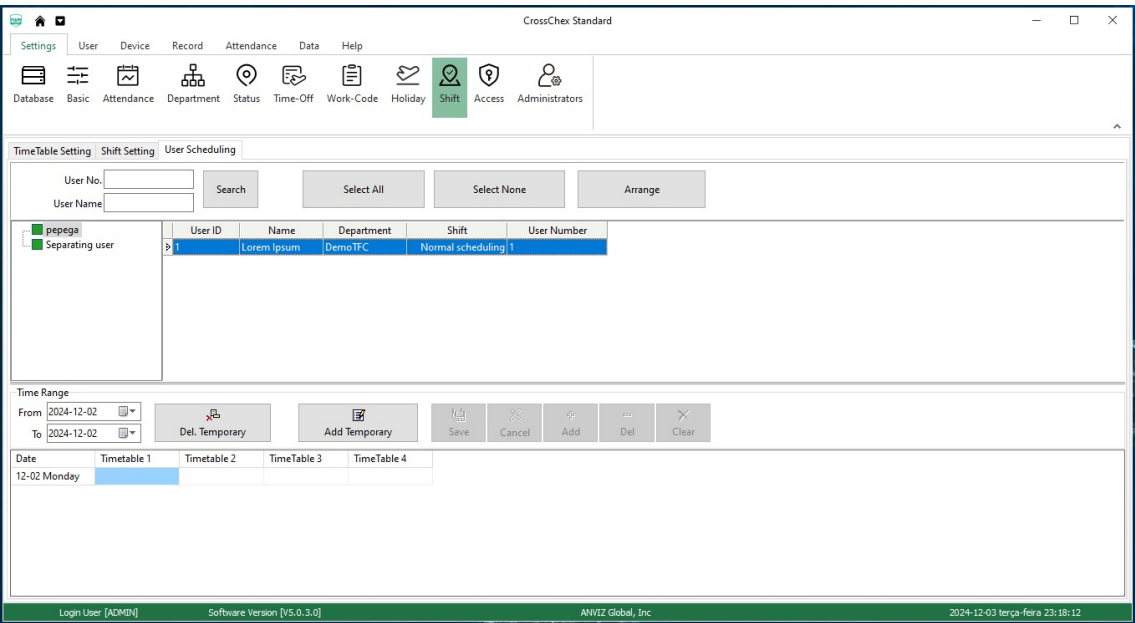


Figure 2.3: Ecrã *Shifts* mostrado na aplicação CrossChex

3 - Viabilidade e Pertinência

A pertinência deste projeto surge no seguimento do exposto no capítulo 1, a expansão das instalações da ARPI S. J. Tojal que resultará no aumento considerável de funcionários. Como a construção tem conclusão em março de 2025 e é esperado que se encontre totalmente em utilização em julho de 2025, o período de desenvolvimento enquadra-se bastante bem com estes prazos exigidos pela instituição.

Como suporte à validação, foi realizado um inquérito às funcionárias do 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 na reflexão de pedidos corretivos e demora desde que uma anomalia relativa ao horário de trabalho acontece até que é detetada. Também está planeado realizar um inquérito formal, abrangente de todos os funcionários, que não consta neste documento devido a apenas ter sido autorizado próximo da data em que é publicado.

Quanto à continuidade pós TFC, considerando que durante o TFC apenas será desenvolvido o fundamental para a solução existir a funcionar, existem funcionalidades adicionais que são úteis à ARPI S. J. Tojal e continuam a enquadrar-se no tema em questão, como é o caso da criação de escalas por parte do *software* e a integração com o *software* usado para processamento de salários. Para alcançar paridade de funcionalidades com outros sistemas no mercado, poderá ser também implementado o uso do terminal para controlo de acesso ou suporte para outros métodos de autenticação, como cartão *Radio-frequency identification* (RFID). Adicionalmente, a construção de um terminal próprio com *Printed Circuit Board* (PCB) desenhada de raiz é um ponto bastante relevante caso o resultado deste projeto motive a abertura de atividade empresarial, para consolidar as funcionalidades necessárias num equipamento visualmente idêntico a outros para o efeito e permitir ter uma base para a implementação de outras funcionalidades, como autenticação por aplicação para *smartphone* que comunica com o terminal por *Bluetooth* ou *Near Field Communication* (NFC), ou idealização de outros cenários de *deployment*, como numa localização remota que poderá exigir conectividade através de rede móvel e/ou *Virtual Private Network* (VPN).

4 - Solução Proposta

4.1 Exemplos de cenários melhorados

A solução a desenvolver terá de cobrir cenários que as mostradas em 2 não têm a capacidade de fazer, nomeadamente uma forma de conseguir dar informação ao funcionário relativamente a anomalias quanto às suas picagens, de forma célere (até 3 dias após ocorrerem). Passam a ser descritas as abordagens que serão adotadas para estes cenários

4.1.1 Permuta de turno sem formalização

De acordo com a escala, um funcionário no dia de hoje está a trabalhar no turno da tarde, 16:00 às 22:00. Fez permuta de turno com um colega para trabalhar antes no turno da manhã, 8:00 às 14:00, e teve permissão verbal da chefia. No entanto, nunca chegou a entregar o pedido de permuta de turno no 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 de alteração de turno. Apenas neste momento é desencadeado o processo de normalização da regularidade junto da chefia e trabalhador.

Com este projeto, esta anomalia passa a ser detetada no dia seguinte devido ao processamento automático de picagens. O alerta passa a ser dado através do equipamento onde é feita a picagem, com uma curta indicação sonora acrescida de informação para o trabalhador confirmar a anomalia com o RH

4.1.2 Processamento do subsídio de natal

Pelo código do trabalho[Por23], o subsídio de natal tem de ser pago até 30 de novembro, sendo que no caso das Instituição Particular de Solidariedade Social (IPSS) o subsídio corresponde a 2.5 dias por cada mês trabalhado na totalidade. Como o pagamento deste subsídio acontece na mesma altura do pagamento do vencimento do mês de novembro, existe um pico de trabalho.

Atualmente, é necessário consultar os registos do ano corrente (arquivados em papel) de cada funcionário para validar se o trabalhador não teve em baixa prolongada (superior a 30 dias). O processo torna-se demorado por ser uma tarefa inteiramente manual, apenas para identificar situações irregulares.

Com este projeto, uma vez que todos os pedidos de alteração de horário ou justificações de falta ficarão guardados na BD da solução, passa a existir uma forma automática de indicar os trabalhadores que estiveram ou se encontram em baixa prolongada e os meses afetados pela(s) mesma(s).

4.2 Prova de conceito

O repositório da prova de conceito desenvolvida encontra-se disponível em https://grupolusofona-my.sharepoint.com/:u:/g/personal/a22102503_alunos_ulht_pt/ERbb345P52VIjK5_EfKND8ABAtYDErnb6-T9dQkQP3sRJA?e=sG1cxJ

O vídeo a mostrar o ambiente de desenvolvimento está disponível em <https://youtu.be/uFecjalULP4>

4.3 Requisitos de *hardware*

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
- Teclado numérico com setas de navegação ou ecrã tátil, para permitir a introdução de PIN e navegação na interface
- Ecrã para visualização da interface, transmitir feedback visual da picagem e informações de irregularidades em dias anteriores
- Coluna para transmitir feedback sonoro da navegação na interface ou da picagem
- *Real Time Clock* (RTC), para que a data e hora se mantenha correta na próxima vez que o terminal volte a ser ligado
- 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 *deployment*.

4.3.1 Terminal de picagem já existente no mercado, com equipamento para fornecer informação extra

De modo a ser possível usar um equipamento já no mercado, para um ambiente desenvolvido por uma entidade que não o fabricante, é necessário que esteja disponível uma *Software Development Kit* (SDK). Partindo do pressuposto que a solução atualmente instalada usa um terminal fabricado por um *Original Equipment Manufacturer* (OEM) com o logótipo da marca que vendeu à ARPI S. J. Tojal, então o equipamento original do OEM estará à venda no mercado.

A pesquisa desse equipamento teve sucesso, descobrindo que trata-se do modelo HY-C280 fabricado pela marca Hysoon. No *website* do fabricante [Hys24], é possível encontrar uma página de transferências onde constam SDKs para os seus equipamentos.

Antes de considerar esta opção viável, a documentação e SDK foram exploradas de modo a identificar se é utilizável. Os recursos de exemplo encontram-se desenvolvidos para C# .NET Framework 4.8 com *Graphical User Interface* (GUI) para sistema operativo Windows, onde cada operação tem a sua própria função que corresponde a uma entrada na documentação. Apesar do uso de uma tecnologia antiga e específica para o sistema operativo Windows, a SDK pode ser facilmente atualizada para uma versão mais recente da linguagem de programação com suporte *Cross-platform*, para que a base funcional deste projeto seja moderna e enquadrada com os modelos de programação atualmente utilizados.

Visto que não é possível programar o que é mostrado no ecrã do terminal, terá de ser utilizado outro capaz disso. Existem no mercado placas de desenvolvimento com base no microcontrolador ESP-32 S3, que já têm um ecrã incorporado, como é o caso da LilyGo T-Display-S3.

Para que esta opção possa funcionar, é necessário que através da SDK do terminal de picagem seja possível obter as picagens em tempo real, para desencadear o processo de

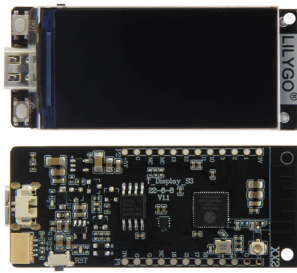


Figure 4.1: Placa de desenvolvimento LilyGo T-Display-S3, frente (esquerda) e verso (direita)

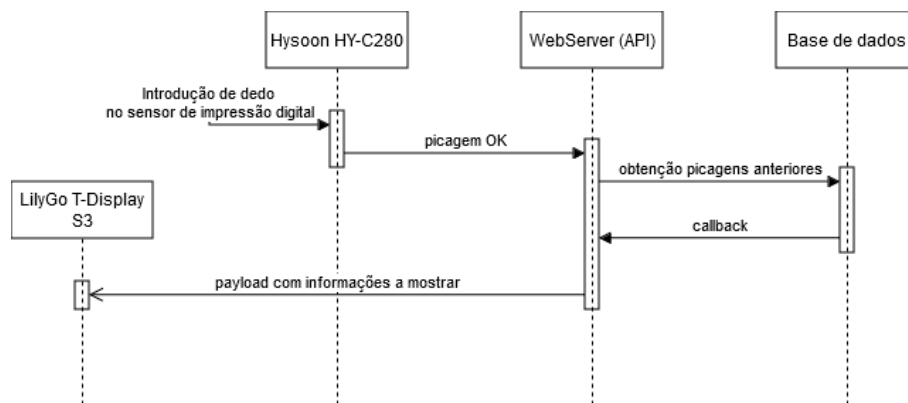


Figure 4.2: Diagrama de integração para fornecer feedback ao trabalhador após a picagem

obtenção de informações sobre os registos dos dias anteriores e, em caso de anomalia, enviá-las para o LilyGo T-Display-S3, de acordo com o seguinte diagrama de interação:

Um dos recursos de exemplo na SDK do Hysoon HY-C280 é exatamente um servidor que escuta todos os registos de picagem transmitidos pelo protocolo de rede *Transmission Control Protocol* (TCP) na porta indicada, sendo a pré-definida 7005.

Assim conclui-se que a solução Hysoon HY-C280 + LilyGo T-Display-S3 é viável, tendo como principais vantagens a instalação mais rápida do terminal, visto que é um produto já existente e não exige adaptação por parte dos trabalhadores por ser igual ao já instalado. Como principais desvantagem, está o tamanho físico da solução e o facto de não ser uniforme.

4.3.2 Desenvolvimento do terminal de picagem com recurso a microprocessador

Em alternativa da aquisição de um terminal de picagem já existente no mercado, poderá ser desenvolvido um de raiz, considerando que para este projeto existe um requisito que nenhum outro terminal já existente cumpre: *feedback* de anomalias face a picagens de dias anteriores.

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 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 relevante. As-

	Pi 3A+ [Ltd24a]	Pi Zero 2W [Ltd24b]	Pi 4B [Ltd24c]	Pi 5 [Ltd24d]
SoC	Broadcom BCM2837	Broadcom BCM2710	Broadcom BCM2711	Broadcom BCM2712
RAM	512MB LPDDR2	512MB LPDDR2	1/2/4/8GB LPDDR4	2/4/8GB LPDDR4X
Conectividade	Wifi, Bluetooth	Wifi, Bluetooth	WiFi, GbE, Bluetooth	WiFi, GbE, Bluetooth
Portas USB	1x USB 2.0	2x micro-USB 2.0	2x USB 2.0, 2x USB 3.0	2x USB 2.0, 2x USB 3.0
Alimentação	5V/2.5A	5V/2.5A	5V/3A	5V/5A
Funcionalidades especiais	N/A	N/A	N/A	Conector pilha RTC
Preço	30€	18€	40€-86€	57€-90€

Table 4.1: Especificações dos SBCs Raspberry Pi

sim, identifica-se que a base de hardware mais adequada é um *Single Board Computer* (SBC), sendo o mais comum o Raspberry Pi.

Devido a não existirem alternativas ao Raspberry Pi à venda em Portugal e estas terem fracos recursos para possibilitar o desenvolvimento (documentação e sistema operativo oficialmente suportado), a escolha passará por uma placa da família Raspberry Pi. Coloca-se a possibilidade de escolha entre o Raspberry Pi 3A, Raspberry Pi Zero 2W, Raspberry Pi 4B 2GB e Raspberry Pi 5 2GB, com as especificações:

Destes quatro SBC, é possível criar dois grupos: Pi 3A+ e Pi Zero 2W, Pi 4B e Pi 5. No primeiro grupo, as especificações são semelhantes e destaca-se o baixo custo, em sacrifício de uma porta Ethernet. No segundo grupo, as semelhanças são ao nível de conectividade, existindo no entanto uma diferença clara de desempenho, que se reflete na especificação recomendada do carregador, e nas funcionalidades especiais, em particular o conector para a pilha de alimentação do *Real Time Clock* (RTC).

Considerando as hipóteses possíveis, o primeiro grupo fica excluído devido à necessidade de usar adaptadores para o terminal ter uma porta Ethernet, que torna complexo concretizar o terminal com dimensões semelhantes ao produto comercial analisado em 4.2.1. No segundo grupo, dada a diferença de custo entre o Pi 4B e Pi 5 e a existência de um conector para a pilha de alimentação do RTC, que contribui para a simplificação de construção do equipamento, esta será a base de hardware selecionada.

Para os restantes componentes, ecrã e teclado (ou apenas ecrã tátil), sensor de impressão digital e coluna de som, serão escolhidas as opções que procurem diversificar a conectividade do Raspberry Pi 5.

Começando pelo ecrã, será utilizado um ecrã tátil ao invés de ecrã e teclado numérico devido ao tamanho das opções para teclado numérico existentes no mercado. O ecrã pode ser ligado via pinos GPIO, comunicando por *Serial Peripheral Interface* (SPI), via *High Definition Media Interface* (HDMI) ou via DSI. Visto que os ecrãs ligados via GPIO usam quase todos os disponíveis e não consideram o uso dos restantes por outros dispositivos, esta opção encontra-se excluída. No caso dos ecrãs *High Definition Media Interface* (HDMI), tipicamente apenas existem em dimensões de 7" ou superior, o que é demasiado grande para este caso. Os ecrãs ligados por DSI encontram-se disponíveis em dimensões a partir de 2.8" e usam um conector próprio pelo efeito, deixando os pinos GPIO livres para os restantes dispositivos. Sendo a última a melhor opção, a escolha passa pelo Waveshare 5inch DSI LCD [Lim24].

Passando para o sensor de impressão digital, os sensores existentes no mercado recorrem à tecnologia ótica ou capacitiva. No caso do sensor ótico, a biometria é obtida



Figure 4.3: Ecrã tátil Waveshare 5inch DSI LCD

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 FAR, mas o custo é mais elevado e a superfície do sensor é bastante sensível a danos e à água. [Yir+23]

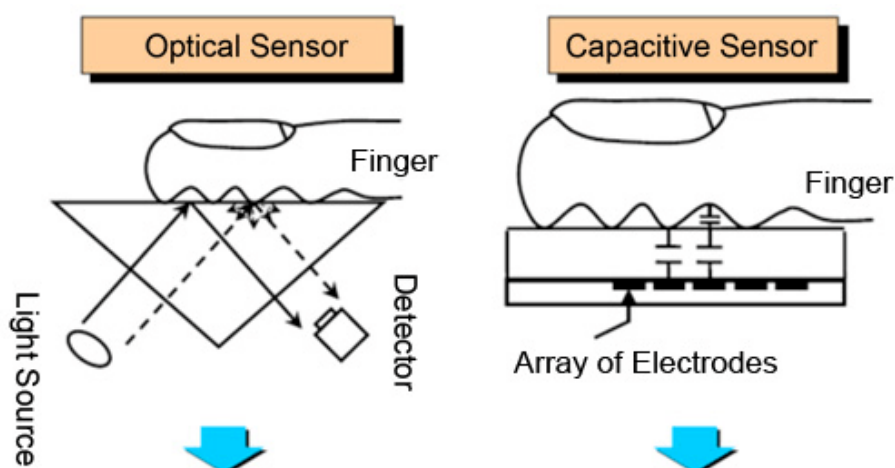


Figure 4.4: Funcionamento do sensor de impressão digital ótico (esquerda) e capacitivo (direita)

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:

Considerando que qualquer um destes sensores satisfaz o critério mencionado acima e as diferenças no FAR e FRR são desprezáveis 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.

Passando para o áudio, usado para fornecer *feedback* dos *inputs* fornecidos, visto

	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 <i>Direct Current</i> (DC)	DC 3.3V	DC 3.8-7.0V	DC 3.3-6V
Custo	28€	33€	55€

Table 4.2: Especificações dos sensores de impressão digital óticos

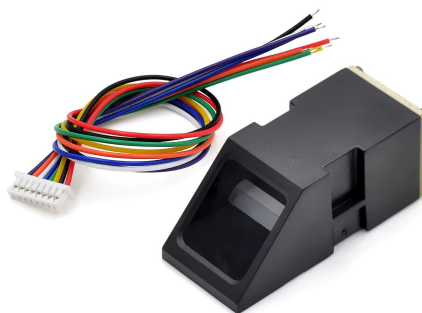


Figure 4.5: Sensor de impressão digital ótico AS608

que o pretendido é reproduzir sons além de um simples beep, a solução mais simples é uma coluna pequena ligada por conector de 3.5mm ou USB. Após pesquisa, a solução identificada como mais adequada é uma coluna ligada por USB, em particular a da imagem abaixo.



Figure 4.6: Coluna USB

Por fim a alimentação, uma vez que existe uma fonte de alimentação oficial para a base de hardware a usar, essa será a opção escolhida.

Assim conclui-se que a opção de construção de um terminal de picagem com base em microprocessador é viável, tendo como principais vantagens os recursos disponíveis e dificuldade intermédia de desenvolvimento do *software* necessário. Como desvantagem está o elevado custo da solução.



Figure 4.7: Fonte de alimentação oficial para Raspberry Pi 5

4.3.3 Desenvolvimento do terminal de picagem com recurso a microcontrolador

4.3.4 Escolha final do *hardware*

Colocadas as opções possíveis, tal como as vantagens e desvantagens de cada uma, no hardware será optado pelo 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 do que poderá ser esperado para a continuação do projeto.

4.4 Requisitos de *software*

Considerando a escolha de hardware feita em , a arquitetura da solução terá por base o que já foi desenvolvido na prova de conceito, sendo que passarão a existir vários terminais, logo serão fisicamente distintos do servidor seguindo o modelo cliente-servidor.

No caso do cliente, que é o terminal de picagem, o software que existe oficialmente com suporte é RaspberryOS, baseado em Debian 12, portanto a linguagem de desenvolvimento terá de funcionar nesta distribuição de Linux. 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 uma vez que a complexidade de desenvolvimento fica reduzida. Aqui existirá uma base de dados para guardar temporariamente os registos até que sejam enviados para o servidor. Para uniformizar as tecnologias utilizadas, o SGBD usado será mariaDB.

No caso do servidor, onde ficará uma API para servir todos os *endpoints*, o servidor web e o SGBD para a Base de dados central, à semelhança dos terminais, terá o SGBD mariaDB e os restantes componentes em Python. Por ter sido a usada na Unidade Curricular (UC) Programação Web e na prova de conceito, o servidor web escolhido é o Django. Para a API, visto que existe a framework *Django REST*, esta será a escolhida para o fim. Portanto, no servidor, o Django ficará como *fullstack*, respondendo aos pedidos *front-end* e *back-end*, tal como visível no diagrama da arquitetura.

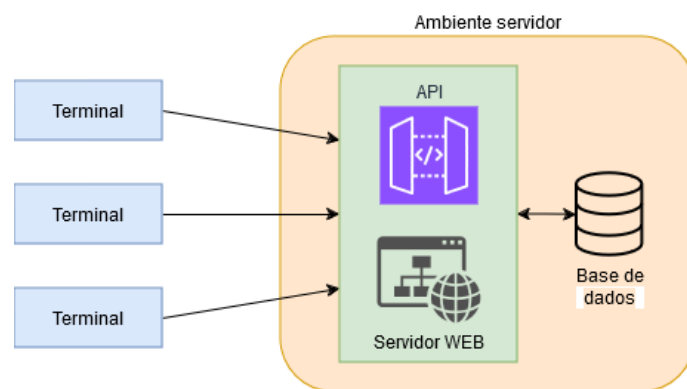


Figure 4.8: Arquitetura para a solução escolhida

5 - Calendário

Partindo das necessidades de desenvolvimento, segmenta-se os próximos 6 meses em desenvolvimento do terminal, desenvolvimento da aplicação web e correção de problemas e melhorias. É necessário ter em conta que por necessidade de negócio, a solução será colocada em produção assim que possível, ou seja, imediatamente depois da conclusão do TFC ou ligeiramente antes, caso se reúnam condições para tal. Adotando a metodologia ágil com sprints de 1 semana, focando em tarefas mais pequenas de modo a garantir conciliação com a vida académica, estas serão as fases de desenvolvimento.

5.1 Primeira fase: extensão da prova de conceito

Durante as semanas de 2024-12-08 e 2025-01-26 serão concluídas as funcionalidades do terminal de picagem, à exceção da indicação de avisos relativamente a picagens anteriores, e as funcionalidades na aplicação web para consultar e manipular funcionários e picagens.

Com o objetivo de testar o funcionamento persistente do terminal, no fim da última semana tem de estar concebido o protótipo para ser montado nas instalações da ARPI S. J. Tojal e usado por 2 funcionários selecionados, que farão o seu registo de assiduidade na solução antiga e nesta em simultâneo.

5.2 Segunda fase: componente lógica e de automatização

Durante as semanas de 2025-02-02 e 2025-04-06 será implementada a componente lógica da aplicação (importação de escalas, cálculo de horas) e de automatização (detecção de presenças ao trabalho). Aqui todas as funcionalidades do terminal devem estar concluídas e a maioria das funcionalidades da aplicação web devem estar implementadas, excluindo a inserção de documentos de alteração de turno e justificações.

Com o objetivo de obter mais dados para testes durante o desenvolvimento, na semana de 2025-03-02 será aumentado o número de funcionários a fazer o seu registo de assiduidade para 8 funcionários. Para preparar a disponibilização do ambiente de servidor, no fim da última semana tem de estar criado o *container* para Docker.

5.3 Terceira fase: documentos e *bug fixing*

Durante as semanas de 2025-04-13 e 2025-06-15 serão concluídas as restantes funcionalidades, inserção de documentos de alteração de turno e justificações, efetuadas correções de erros e implementadas melhorias de *Quality of Life* (QoL).

Com o objetivo de identificar padrões de erro dos funcionários, serão conduzidos testes com utilizadores no início da primeira semana. No fim desta semana será disponibilizado o ambiente de testes. Uma vez que nesta altura a aplicação já estará minimamente funcional e o tempo para observar o efeito das novas alterações em ambiente de testes é reduzido, a versão nesse ambiente passa a ser atualizada no início de cada semana com as alterações mais recentes.

Bibliografia

- [Mat24] João P. Matos-Carvalho. *The Lusófona L^AT_EX 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).
- [Por23] XXIII Governo Constitucional de Portugal. “Lei n.º 13/2023 - Código do Trabalho”. In: *Diário da República* (2023). URL: <https://portal.act.gov.pt/AnexosPDF/Legisla%C3%A7%C3%A3o%20nacional/C%C3%B3digo%20do%20trabalho.pdf>.
- [Hys24] Hysoon. *Hysoon Group*. 2024. URL: <http://www.hysoon.com> (visited on 09/2024).
- [Ltd24a] 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).
- [Ltd24b] 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).
- [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 (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 (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 (visited on 11/2024).

Glossário

API *Application Programming Interface*. 17

ARPI S. J. Tojal Associação de Reformados, Pensionistas e Idosos de São Julião do Tojal. 7, 10, 12, 15, 19

ATL Atividades de Tempos Livres. 7

BD Base de dados. 11, 17

BT *Bluetooth*. 8

DC *Direct Current*. 16

DSI *Display Serial Interface*. 14

ERPI Estrutura Residencial para Idosos. 7

FAR *False Acceptance Rate*. 15, 16

FRR *False Rejection Rate*. 15, 16

GPIO *General Purpose Input/Output*. 13, 14

GUI *Graphical User Interface*. 12

HDMI *High Definition Media Interface*. 14

IPSS Instituição Particular de Solidariedade Social. 11

NFC *Near Field Communication*. 10

OEM *Original Equipment Manufacturer*. 12

PCB *Printed Circuit Board*. 10

PDF *Portable Document Format*. 7

picagem Registo de início ou fim de trabalho. 7, 8, 11–13, 16, 17, 19

PIN Número de identificação pessoal. 8, 12

QoL *Quality of Life*. 19

RAM *Random Access Memory*. 14

REST *Representational State Transfer*. 17

RFID *Radio-frequency identification*. 8, 10

RH Recursos Humanos. 7, 10, 11

RTC *Real Time Clock.* 12, 14

SaaS *Software as a Service.* 8

SBC *Single Board Computer.* 6, 14

SDK *Software Development Kit.* 12, 13

SGBD *Sistema de Gestão de Bases de Dados.* 8, 17

sistema embebido coisa fixe que não é um computador e cumpre apenas uma função.
12, 13

SoC *System on a Chip.* 14

SPI *Serial Peripheral Interface.* 14

TCP *Transmission Control Protocol.* 13

TFC *Trabalho Final de Curso.* 8, 10, 19

UC *Unidade Curricular.* 17

VPN *Virtual Private Network.* 10