Instituto Superior de
Engenharia de Coimbra

Departamento
de Engenharia Eletrotécnica

Tecnologias de Apoio à Monitorização de Fluxos de Pessoas e Controlo de Acessos Versátil de Baixo Custo

Trabalho de Projeto apresentado
para a obtenção do grau de Mestre em
Automação e Comunicações em Sistemas de Energia

Autor
Fábio Jorge Gomes da Costa

Orientador
Doutor Nuno Miguel Fonseca Ferreira
Doutor Victor Daniel Neto dos Santos

Coimbra, Dezembro, 2013

Viver é enfrentar um problema atrás do outro. O modo como você o encara é que faz a diferença.

Benjamin Franklin

Aprender é a única coisa de que a mente nunca se cansa, nunca tem medo e nunca se arrepende.

Leonardo Da Vinci

Agradecimentos

Este trabalho não representa apenas o resultado de extensas horas de estudo, pesquisa, investigação, reflexão e trabalho durante as diversas etapas que o constituem. É igualmente o culminar de um objetivo académico a que me propus e que não seria possível alcançar sem ajuda.

Estou particularmente agradecido ao Professor Doutor Nuno Ferreira pela sua orientação, pela fonte de ideias e pela possibilidade de desenvolver o presente trabalho no âmbito de uma bolsa de investigação aplicada, que surgiu do dinamismo da presidência do ISEC e do objetivo de promover a realização de projetos tecnológicos dentro do instituto.

Estou também agradecido ao Professor Doutor Victor Santos, pela sua vasta perspicácia, conhecimentos e sugestões transmitidas durante a elaboração do trabalho, pelos seus sábios conselhos e apoio na superação dos diversos obstáculos.

Ao Eng.º Tiago Figueira, do Gabinete de Informática (GI) do ISEC, pela realização de uma plataforma de registos no servidor do ISEC a qual possibilitou a execução de testes da plataforma de registo dos trabalhadores.

A todos os colegas do RoboCorp, em especial ao Doutor Micael Couceiro pela orientação na escrita de um artigo científico e pela motivação que me incutiu e ao Eng.º Samuel Pereira pelo auxílio no desenvolvimento da plataforma Web.

Aos trabalhadores do GTE (Gabinete Técnico de Eletrotécnica), o Sr. Francisco Dias, o Eng.º João Queiró e a Eng.ª Sónia Branco, pelo apoio e motivação, pela disponibilidade na execução das placas de circuito impresso e na obtenção de componentes, e pelas críticas construtivas que muito contribuíram para o desenvolvimento do projeto.

À Família, pelo apoio incondicional e em particular aos meus pais que estiveram sempre presentes e me encorajaram e apoiaram incondicionalmente durante toda a minha vida.

À minha namorada Tatiana, pela amizade, carinho, paciência e sacrifício imposto pelo tempo dedicado ao desenvolvimento deste trabalho.

O meu profundo e sentido agradecimento a todas as pessoas que contribuíram para a concretização desta dissertação, estimulando-me intelectual e emocionalmente.

Resumo

A gestão das identificações e acessos é um grande desafio no ensino superior desde que foram exigidas diferenciações de diferentes utilizadores, bem como implementações políticas de acesso a diferentes zonas, dependendo do utilizador, estudante ou funcionário da instituição (como trabalhadores ou professores). Além disso, somente pessoas autorizadas (e.g., professores) podem efetuar algumas ações ou ter acesso a zonas específicas dentro do ambiente educacional.

Este projeto de tese (Costa, et al., 2010) apresenta o desenvolvimento de um sistema de controlo e gestão de acessos de baixo custo integrado no ambiente educacional. Além de descrever o controlo de acesso a salas de aulas, laboratórios e departamentos, este trabalho é focado no registo das presenças dos alunos em aulas e nalguns espaços controlados, no registo de presença de trabalhadores da instituição, bem como a gestão do parque de estacionamento dentro do campus do ISEC.

Depois de desenvolvidos os protótipos, estes foram implementados no Instituto Superior de Engenharia de Coimbra, onde vieram a ser aperfeiçoados e testados e se revelaram fiáveis, com um bom funcionamento e capaz de monitorizar e fazer a gestão do controlo de acessos.

Palavras-chave: Gestão de Acessos, Baixo-Custo, RFID, Monitorização de Acessos

Abstract

The identification and management of access is a major challenge in higher education since they were required differentiations of different users, and policy implementations of access to different areas , depending on the user , student or employee of the institution (as employees or teachers) . Furthermore, only authorized personnel (eg, teachers) are allowed to perform some action or have access to specific areas within the educational environment.

This thesis project (Costa, et al., 2010) presents the development of a system of control and management of access to low cost integrated into the educational environment. Besides describing the access control classrooms, laboratories and departments, this work is focused on the record of attendance of students in classes and some controlled spaces, the registration of the presence of officials of the institution as the good management of parking within ISEC campus.

Once developed prototypes, these were implemented at the Instituto Superior de Engenharia de Coimbra, which came to be refined and tested and proved reliable, with a good functioning and able to monitor and manage access control.

Keywords: Access Management, Low-Cost, RFID, Monitoring Assiduity

Índice

Lista de Figuras

Índice de Tabelas

Acrónimos

ADC	*Analog-to-Digital Converter*
ANACOM	Autoridade Nacional de COMunicações
ANSI	American National Standards Institute
ARP	Address Resolution Protocol
ASP	*Active Server Pages*
CGD	Caixa Geral de Depósitos
CSS	*Cascading Style Sheets*
CUP	Cartão Universidade Politécnico
DEC	Departamento de Engenharia Civil
DEE	Departamento de Engenharia Eletrotécnica
DEIS	Departamento de Engenharia Informática e de Sistemas
DEM	Departamento de Engenharia Mecânica
DEQB	Departamento de Engenharia Química e Biológica
DFM	Departamento de Física e Matemática
DHCP	*Dynamic Host Configuration Protocol*
DNS	*Domain Name System*
DoD	*Department of Defense*
EAS	*Electronic Article Surveillance*
EMV	*Europay / MasterCard/Visa*
ETSI	*European Telecommunications Standards Institute*
FCFS	*First-Come, First-Served*
FEUP	Faculdade de Engenharia da Universidade do Porto
HTML	*HyperText Markup Language*
I^2C	*Inter-Integrated Circuit*

ICMP	*Internet Control Message Protocol*
IDE	*Integrated Development Environment*
IEC	*International Electrotechnical Commission*
IEEE	*Institute of Electrical and Electronics Engineers*
IES	Instituições de Ensino Superior
IFF	*Identify Friend or Foe*
IGMP	*Internet Group Management Protocol*
IPC	Instituto Politécnico de Coimbra
ISEC	Instituto Superior de Engenharia de Coimbra
ISM	*Industrial, Scientific and Medical*
ISO	*International Standards Organization*
ITU	*International Telecommunication Union*
LCD	*Liquid Crystal Display*
MACSE	Mestrado de Automação e Controlo em Sistemas de Energia
NFC	*Near Field Communication*
PCB	*Printed Circuit Board*
PHP	*Hypertext Preprocessor*
PPPoE	*Point-to-Point Protocol over Ethernet*
PWM	*Pulse-Width Modulation*
RFID	*Radio-Frequency IDentification*
RTC	*Real Time Clock*
SGBD	Sistema de Gestão de Base de Dados
SMTUC	Serviços Municipalizados de Transportes Urbanos de Coimbra
SPI	*Serial Peripheral Interface*
SQL	*Structured Query Language*
TCP	*Transmission Control Protocol*
TTL	*Transistor-Transistor Logic*

UART	*Universal Asynchronous Receiver Transmitter*
UDP	*User Datagram Protocol*
USB	*Universal Serial Bus*
WDT	*Watch Dog Timer*

1. INTRODUÇÃO

As Instituições de Ensino Superior (IES) têm como principais atribuições proporcionar aos seus alunos um ensino de elevada qualidade, realizar investigação fundamental e aplicada; promover a transferência de conhecimento e tecnologia para a indústria e finalmente produzir e difundir conhecimento e cultura.

Para se alcançarem os referidos objetivos é necessário que os órgãos de gestão das instituições disponham de ferramentas de apoio, que em interligação com os gabinetes de qualidade, de gestão de recursos humanos e de economato e inventário, sejam capazes de uma forma contínua aferir a qualidade do processo educativo e dos custos que lhes estão associados.

Assim, para otimizar a logística associada ao processo educativo é indispensável um correto planeamento e análise de custos, sendo necessário conhecer e registar, por exemplo a utilização das salas de aulas, dos anfiteatros e dos laboratórios, bem como dos equipamentos neles contidos. É igualmente necessário, de uma forma simples e transparente, registar informação relativa à assiduidade e pontualidade dos intervenientes no processo (alunos, docentes e trabalhadores), e informação relativa às aulas, na forma de sumários.

Antes de se propor uma nova solução técnica, foi dada particular atenção a projetos desenvolvidos noutras IES para o mesmo efeito.

Para o controlo de acesso de veículos no campus da Faculdade de Engenharia da Universidade do Porto (FEUP) em (Correia, Carvalho, & Nunes, 2002) desenvolveu-se uma solução que utiliza o cartão de estudante como identificação, através de leitura do código de barras nele contido, ajudando dessa forma o porteiro a registar corretamente as entradas numa base de dados. Uma solução mais complexa, baseada em tecnologia *Radio-Frequency IDentification* (RFID) combinada com uma rede *wireless* é proposta em (Lourenço & Almeida, 2009). O sistema localiza as pessoas e objetos num local de trabalho, simplificando a burocracia associada ao controlo de pessoas e materiais.

De uma forma similar, em (Madeira, Antunes, Morgado, & Pereira, 2008) é apresentada uma solução para monitorizar e controlar a presença dos alunos, usando uma plataforma de ensino à distância. Os participantes registam a sua presença através de um objeto virtual que contém informação, o qual é enviado para uma base de dados acedida por professores e outro pessoal administrativo.

Em (Kabir, Huang, Wu, & Rapajic, 2007), foi desenvolvido um sistema baseado em RFID, que identifica automaticamente os utilizadores e controla o acesso a espaços, nomeadamente salas de aula, laboratórios, bibliotecas, bem como o acesso remoto a computadores. A aplicação proposta utiliza uma base de dados *Structured Query Language* (SQL) alojada num servidor, com o objetivo de substituir um sistema mais antigo baseado em código de barras.

Finalmente, (Akpinara & Kaptan, 2010) descreve uma solução que reduz o trabalho administrativo da escola, assim como a execução automática de tarefas diárias, tais como: identificação de pessoa em aulas / laboratórios, gestão de atendimento da biblioteca, e-money, avisos, anúncios e registos de atividades. O sistema utiliza diversos computadores interligados entre si e equipados com leitores de RFID, com o objetivo de executar cada uma das tarefas acima mencionadas.

1.1. Implementação de um Sistema de Controlo de Acessos no ISEC

As limitações das soluções anteriores apontam para o desenvolvimento de uma nova solução de controlo de acesso, a qual tem por objetivo acomodar com segurança o maior número de carros no *campus*, tendo em conta a existência de utilizadores com privilégios (membros dos órgãos de gestão do Instituto Superior de Engenharia de Coimbra (ISEC), deficientes e pessoas em cadeiras de rodas).

No parque de estacionamento da presidência, que possui um controlo de acesso adicional, existem seis lugares de estacionamento reservados aos titulares dos cargos de gestão do ISEC, um lugar reservado exclusivamente a deficientes, os restantes lugares de estacionamento são utilizados usando uma política *First-Come*, *First-Served* (FCFS) entre os trabalhadores e os professores. No campus do ISEC existem outras áreas de estacionamento, nas quais os lugares de estacionamento estão aglomerados em parques (*clusters*) ou reservados ao longo das estradas existentes dentro do *campus* do ISEC.

Outro aspeto importante no mundo académico é o controlo de presenças dos alunos nas aulas. Atualmente, na maioria das instituições de ensino superior, o processo de registo de assiduidade é efetuado a partir das folhas de presença em formato papel. Este método simples é efetivo para efeitos de registo da assiduidade, no entanto somente utilizando um suporte digital será possível implementar novas funcionalidades tais como: registo do instante de chegada dos alunos, duração da aula, penalização da classificação dos alunos baseada na pontualidade e na assiduidade.

A informação em formato digital possibilita também a justificação da ausência de um aluno num dado período quando devidamente justificada com uma certidão válida (médica, do empregador ou outra). Para além destas funcionalidades é igualmente desejável: registar as tarefas de manutenção efetuadas pelos técnicos nos laboratórios; identificar o número de horas que os alunos dedicam à realização dos projetos nos laboratórios e identificar os alunos que não estão a frequentar uma determinada unidade curricular, apesar de estarem inscritos.

Com estes dados é possível adequar o número de estudantes à capacidade das salas; prever as necessidades em termos do número de turmas por unidade curricular e produzir estatísticas para o diretor de curso, presidente de departamento, conselho pedagógico e para a presidência da instituição relacionadas com o sucesso, presenças e abandono dos alunos.

Durante o processo de conceção do sistema de gestão e controlo de acessos, foram analisados cenários distintos para aplicação no ISEC, mas também noutras escolas do Instituto Politécnico de Coimbra (IPC). Numa primeira fase, pretendeu-se desenvolver um sistema totalmente novo, o qual compreendia a utilização de *tags* desenvolvidas para o efeito. No entanto, tendo em conta a dimensão da comunidade estudantil do IPC e o facto da maior parte dos custos associados à implementação de uma solução advir do custo das *tags* RFID, optou-se pela utilização dos cartões de estudantes, de professores e de trabalhadores existentes, fornecidos pela Caixa Geral de Depósitos (CGD) (Caixa Geral de Depositos, 2012), os quais incorporam tecnologia RFID Mifare.

1.2. Objetivos e Metodologia

Tendo por base os requisitos enunciados, a presidência do ISEC lançou um projeto (Costa, et al., 2010) com vista ao desenvolvimento de uma solução tecnológica vocacionada à monitorização de fluxos de pessoas e controlo de acessos ao campus do ISEC, que se pretende versátil e de baixo custo.

Assim sendo, foi desenvolvida uma plataforma capaz de efetuar a leitura de cartões que fazem parte do quotidiano dos utilizadores (cartão bancário de utilizador da instituição) e comunicar com um servidor os dados nele contidos, para este efetuar a identificação do usuário. Outro requisito importante do sistema a desenvolver, contempla a criação de diferentes variantes do sistema baseada na mesma plataforma (fixa e/ou móvel), que utilizassem o mesmo servidor e base de dados, sendo assim possível criar múltiplos sistemas baseados na mesma plataforma, como, sejam o sistema de gestão de presença de alunos em aulas, controlo e gestão de acessos às salas de aulas, laboratórios e ao campus do ISEC.

1.2.1. Objetivos

De acordo com o enunciado anterior, o presente projeto tem como objetivo desenvolver um sistema de controlo de acessos no ISEC unificado, o qual deve ter as seguintes funcionalidades:

- Controlo de acessos aos laboratórios e salas de aulas;
- Gestão do registo de ponto dos trabalhadores da instituição;
- Gestão de assiduidade dos alunos nas aulas (móvel ou fixo);
- Controlo de acessos ao campus do ISEC;
- Gestão integrada da informação numa plataforma web.

1.3. Estrutura da Dissertação

Neste capítulo introdutório é efetuada uma descrição geral dos sistemas de controlo de acessos e da sua aplicação em instituições de ensino superior (IES), sendo analisada a problemática que lhe está associada. Apresentam-se também algumas soluções tecnológicas já implementadas e em funcionamento em IES de referência em Portugal. Adicionalmente mencionam-se os objetivos e requisitos gerais subjacentes ao desenvolvimento da plataforma que se pretende construir, bem como à estrutura da presente dissertação.

No capítulo II – Identificação Por Rádio Frequência – é feita uma breve introdução histórica da tecnologia RFID. Nessa descrição é apresentada a evolução desde as primeiras experiências de comunicações por rádio frequência, até às aplicações presentes no nosso quotidiano e aos últimos desenvolvimentos desta tecnologia. Neste capítulo apresenta-se também as partes constituintes de um sistema de RFID e suas características, bem como o seu funcionamento, normas e regulamentos existentes para a utilização da tecnologia RFID;

No capítulo III – Conceção do Sistema – apresenta-se a localização dos edifícios e dos laboratórios no campus do ISEC. Com base nesta informação, são tomadas algumas decisões relativas ao sistema de controlo de acessos a implementar. Neste capítulo apresenta-se também o funcionamento do sistema global de gestão dos acessos e da assiduidade, tendo por base as ideias analisadas durante a conceção do sistema, bem como as funcionalidades que se pretendem implementar em cada uma das plataformas a desenvolver.

No capítulo IV – "Desenvolvimento de Plataforma" – apresentam-se as escolhas efetuadas relativas aos componentes constituintes da plataforma desenvolvida, de forma a cumprir os requisitos do sistema. Além disso, apresenta-se o resultado do processo de seleção do microcontrolador, do módulo RFID, do módulo Ethernet, do módulo de gestão de energia, da bateria e respetivo carregador. Refira-se que estas escolhas foram efetuadas sempre tendo em conta o compromisso existente entre o desempenho da solução desenvolvida e o preço de mercado dos seus componentes. Uma vez escolhidos os componentes, apresenta-se em detalhe a incorporação dos mesmo nas placas de circuito impresso projetadas e na interligação dos diferentes módulos comerciais e/ou desenvolvidos. Posteriormente apresenta-se o *software* desenvolvido, na forma de fluxograma onde se explica o algoritmo e/ou através do código das funções mais relevantes. No fim deste capítulo apresentam-se os protótipos implementados e realiza-se uma análise crítica dos mesmos.

No capítulo V intitulado – "Controlo e Gestão do Sistema" – faz-se referência à plataforma de gestão *online* do sistema de controlo de acesso e da base de dados desenvolvida, para ser utilizada nas múltiplas vertentes da plataforma desenvolvida (controlo de acesso ao campus, controlo de assiduidade dos trabalhadores e acesso aos laboratórios). Inicialmente são enunciadas as funcionalidades pretendidas, de seguida são referidas as tecnologias utilizadas no seu desenvolvimento. No final é demonstrado o resultado final da plataforma, bem como a descrição do seu funcionamento.

Finalmente no capítulo VI – Conclusões – procedeu-se à discussão dos resultados finais, resultantes das plataformas desenvolvidas, bem como das conclusões retiradas dos testes efetuados ao sistema. Neste capítulo são apresentados os pontos a melhorar e pontos de continuidade deste projeto na forma de trabalho futuro.

2. IDENTIFICAÇÃO POR RÁDIO FREQUÊNCIA

A identificação por radiofrequência ou RFID é um método de identificação automática através de sinais rádio, que permite nalguns casos ler e gravar dados remotamente de/para uma etiqueta RFID.

Através de um cartão que incorpore a tecnologia RFID é possível identificar, localizar e monitorizar posições de qualquer objeto ou pessoa (Chiesa, et al.) (Carlosle).

Tendo em conta a sua capacidade de localização e identificação, a tecnologia RFID sofreu múltiplos desenvolvimentos, sendo utilizada na indústria, movimentando atualmente vários milhares de milhares de dólares (Chiesa, et al.).

As aplicações em RFID têm um número elevado de aplicações, onde a imaginação é o limite. De entre elas refira-se identificação e localização de objetos (organização de stocks), controlo e gestão de mercadorias, monitorização de pessoas e bens, entre outros, para além da sua utilização para alarme antí-furto em lojas de roupa ou em bibliotecas. O RFID pode ser usado em diagnósticos médicos, na indústria química, farmacêutica e têxtil, mostrando a diversidade de áreas onde pode ser aplicado.

2.1. História

Nos finais do século XIX, Heinrich Rudolf Hertz, Nikola Tesla, Guglielmo Marconi e outros cientistas experimentaram a transmissão e receção de ondas eletromagnéticas. Em 1887, Heinrich Rudolf Hertz confirma experimentalmente as leis de Maxwell, sobre ondas eletromagnéticas ao conseguir transmitir e receber ondas de rádio em laboratório de uma forma rudimentar. Mais tarde em 1896, Guglielmo Marconi consegue efetuar a primeira transmissão de informação via rádio, naquela que seria a primeira comunicação transatlântica sem fios. Assim sendo, o nascimento das comunicações rádio é geralmente aceite como tendo ocorrido aquando do registo da patente do telégrafo sem fios por Marconi.

Em 1948, Harry Stockman publica um artigo (Harry, 1948) intitulado "*Communication by Means of Reflected Power*" no qual se apresenta a exploração do uso da potência refletida como forma de comunicação.

A tecnologia RFID teve assim origem nos sistemas de radares utilizados na segunda guerra mundial (entre 1939 a 1945), nos quais alterações nas transmissões rádios, resultantes da presença dos aviões alemães e das manobras efetuadas pelos seus pilotos, serviam de identificação para os operadores de radar. Este simples método é considerado o primeiro sistema passivo de RFID.

Os ingleses, por sua vez, desenvolveram o primeiro identificador ativo de aliado ou inimigo *Identify Friend or Foe* (IFF), colocando um transmissor em cada avião britânico. Quando esses transmissores recebiam sinais das estações de radar no solo, começavam a transmitir um sinal de resposta, que identificava a aeronave como amigável.

A partir da década de 60, cientistas e académicos de todo o mundo realizaram pesquisas para idealizar soluções para que a energia da radiofrequência possa ser utilizada para identificar objetos remotamente.

Mais tarde nos finais dos anos 60 e início da década de 70, foram desenvolvidos sistemas para controlar se um produto já tinha sido pago ou não (*Electronic Article Surveillance* (EAS)) (Gines & Tsai, 2007). Ainda nesta década várias entidades aperceberam-se do enorme potencial desta tecnologia. A partir daí observou-se uma explosão no desenvolvimento da tecnologia do RFID. A primeira patente foi registada a 23 de Janeiro de 1973 de uma *tag* ativa e de uma *tag* passiva que permitia destrancar uma porta de um carro sem a utilização da chave, seguindo-se das primeiras aplicações de sistemas RFID para animais (Cravo Gomes, 2007).

Os sistemas RFID atuais funcionam tendo por base o mesmo princípio. Um sinal é enviado por um transmissor a uma *tag* (etiqueta ou *transponder*) que retransmite de volta o sinal com determinadas modificações, correspondendo a sistemas passivos. Nos sistemas ativos é o próprio identificador que transmite o seu próprio sinal.

A partir da década de 80, o RFID entra definitivamente na indústria e no comércio a nível mundial. Nos Estados Unidos da América (EUA), em controlo de mercadorias, meios de transporte, acesso de pessoas e identificação animal. Na Europa, no uso da tecnologia para identificação animal, atividades industriais e controlo de acesso em rodovias (Gines & Tsai, 2007) .

Finalmente, a partir da década de 90, o RFID torna-se presente e largamente comum no dia-a-dia das pessoas, com o surgimento de normas reguladoras e aplicações comerciais a custos reduzidos. Na Tabela 1 encontra-se um resumo dos marcos mais relevantes no desenvolvimento do RFID ao longo dos tempos.

Tabela 1 – Evolução do RFID (Jandt, 2005).

Intervalo de Tempo	Ocorrências
1940-1950	Invenção e rápido desenvolvimento do radar durante a 2ª Guerra Mundial Início de funcionamento do RFID em 1948
1950-1960	Primeiras explorações da RFID e experimentações laboratoriais
1960-1970	Desenvolvimento da teoria da RFID e primeiras aplicações experimentais no terreno
1970-1980	Explosão no desenvolvimento da RFID e aceleração dos testes Implementações embrionárias de RFID
1980-1990	Aplicações comerciais de RFID entram no mercado
1990-2000	Surgimento de normas RFID é largamente utilizado começando a fazer parte das nossas vidas

Nos dias de hoje, a tecnologia de RFID está em quase todo o lado. O seu uso é tão usual que já nem se dá conta da sua presença. Por exemplo a tecnologia está integrada nos passaportes, portagens, bibliotecas, lojas, metro e outros transportes públicos. A adesão a esta tecnologia tem vindo a crescer, sendo atualmente usada a nível mundial em múltiplas aplicações.

A título de exemplo, a Figura 1, representa o identificador do sistema de portagem eletrónica Via Verde®. Refira-se que a Via Verde foi pioneira no mundo ao implantar RFID para implementar um sistema completamente inovador de cobrança automática de portagens.

Figura 1 – Identificador Via Verde.

A Figura 2, representa uma *tag* andante e respetivo leitor utilizado para controlar as entradas nos serviços intermodais da área metropolitana do Porto (STCP, metro e CP). Finalmente, a Figura 3, apresenta o cartão convida utilizado nos autocarros (bilhetes e passe) dos Serviços Municipalizados de Transportes Urbanos de Coimbra (SMTUC).

Figura 2 – Exemplo de utilização de cartão andante no metro do Porto.

Figura 3 – Cartão conVIDA utilizados nos autocarros SMTUC de Coimbra.

Um dos últimos desenvolvimentos na área do RFID é o *Near Field Comunication* (NFC) (NFC-Forum, 2012), este desenvolvimento tem como fundamento a integração da tecnologia RFID, em telemóveis, para permitir o seu uso como carteira eletrónica de forma a permitir o pagamento do táxi, dos combustíveis, do café, entre outras pequenas despesas.

A comunicação NFC veio modernizar o processo de controlo de acesso baseado em tecnologias sem fios. Este método utiliza chaves de criptografia entre dispositivos, através de um canal do campo-próximo, que, quando limitado a 20 cm, permite uma segurança física adicional ao conjunto. O padrão NFC foi desenvolvido para ser compatível com a *tag* RFID ISSO 15693 que opera na banda de 13.56 MHz, possibilitando também que dispositivos móveis comuniquem com outros protocolos de *tags* e sejam também compatíveis com os protocolos de *smartcard* FeliCa e Mifare.

Os intervenientes no desenvolvimento desta nova tecnologia englobam operadores de telecomunicações, fabricantes de equipamento eletrónico, entre outros, designadamente: American Express, Anadigm, France Telecom, Innovision, Inside, LG, Logitech, Motorola, RFMD, SK Telecom, Skidata, Vodafone; e como patrocinadores oficiais: MasterCard International, Matsushita Electric Industrial, Microsoft, Nokia, NEC, Renesas Technology, Royal Philips Electronics, Samsung, Sony, Texas Instruments e Visa International.

Em 2004 a Nokia lançou um equipamento (3200 GSM), que incorpora um sistema NFC, o telefone pode fazer pagamentos eletrónicos e fazer chamadas baseadas na deteção de *tags* de RFID. Um exemplo possível da sua aplicação, compreende o pedido de um táxi para qualquer sítio ao colocar um telemóvel próximo a uma *tag* RFID localizada na placa de uma praça de táxis. O telemóvel efetua uma comunicação para a companhia de táxis a solicitar um táxi para a localização pretendida.

2.2. Constituição do Sistema

Os sistemas RFID são essencialmente constituídos por dois blocos distintos: uma *tag* (etiqueta RFID que pode ser passiva ou ativa) e o leitor de *tags*.

A *tag* é um pequeno dispositivo que serve de identificador do objeto no qual foi colocado ou na pessoa que o possui. Este dispositivo pode ter diversas formas, desde um pequeno chip que será implantado em animais (ter menos de 3 mm de diâmetro e 10 mm de comprimento – *tag* passiva) até ao tamanho de um livro (*tag* ativa). A Figura 4 apresenta uma *tag* para aplicação num animal e respetivo aplicador. O tamanho e forma de cada *tag* variam consoante a aplicação à qual se destina.

Figura 4 – Exemplo de tag para aplicação em animais.

O leitor (13.56 MHz Mifare Read/Write Module, 2012) solicita informação à *tag* e esta devolve a informação contida dentro dela (microchip), podendo esta informação ser só um bit (caso dos sistemas de identificação de venda "antirroubo"), ou um conjunto de informações mais complexa de uma base de dados com o histórico de uma informação associada ao objeto.

Apesar desta ser a maneira mais comum de obter informação, existem *tags* ativas que transmitem informação sem a presença do leitor.

O leitor pode ser considerado um dos componentes, que tem como papel a gestão de um sistema RFID porque, é de sua responsabilidade a ligação entre as *tags* e o restante sistema. É igualmente responsável pela gestão de presença de varias *tag's* em simultâneo, rejeição de repetições de dados, correção de erros, entre outros.

O processamento de dados é colocado no leitor, uma vez que a *tag* é um dispositivo de tamanho reduzido e de baixa complexidade que se pretende de baixo custo, visto que uma dada aplicação pode ter milhares de unidades de *tags*.

Para que um sistema de RFID funcione é necessário que estes incorporem também antenas e aplicações de *software* (necessárias para o seu correto desempenho).

As antenas são o elemento responsável pela transmissão de dados entre o leitor e as *tags*, "convertendo" os sinais elétricos em eletromagnéticos para que estes sejam transmitidos pelo ar em ambos os percursos do leitor para a *tag* e vice-versa.

2.3. Tipos de Tags

As *tags* podem ser classificadas em 3 grupos: passivas, ativas e semi-ativas (ou semi-passivas).

As *tags* passivas são aquelas que se limitam a "responder" a sinais provenientes de um transmissor, não tendo capacidade de emitir um sinal autonomamente e não possuem qualquer sistema de alimentação. As *tags* passivas usam a energia enviada pelo leitor para alimentar os circuitos e transmitir os dados armazenados. Assim, uma *tag* passiva deve ter uma constituição muito simples e um reduzido número de elementos, como se pode observar na Figura 5. Uma vez que estes tipos de *tags* não necessitam de alimentação, não é necessário qualquer tipo de manutenção, carregamento ou troca de bateria. As *tags* passivas podem resistir a condições extremas sem que o seu funcionamento seja prejudicado.

Figura 5 – Tag RFID passiva.

Além disso, as *tags* passivas são geralmente menores que as *tags* ativas e a sua produção em massa reduz significativamente os seus custos. Sendo que as *tags* passivas mais simples podem custar entre 0.05 a 0.15 $USD cada, ao passo que cada *tag* ativa custa a partir de 25 $USD atingindo facilmente os 100 $USD (RFIDjournal). No entanto, é também de notar que a área de cobertura de uma *tag* passiva é menor do que a área de cobertura de uma *tag* ativa (Finkenzeller, 2003).

Na comunicação entre o leitor e a *tag*, o leitor terá sempre que comunicar em primeiro lugar, pois o *tag* necessita da potência recebida do leitor para funcionar. Por este motivo, neste sistema, o leitor tem que estar constantemente a emitir sinais RF para o campo de ação, para ser possível detetar a presença de uma *tag*.

As *tags* ativas por sua vez possuem alimentação, normalmente por meio de uma bateria, como se pode visualizar na Figura 6. Ao contrário das *tags* passivas, estas não precisam de receber energia do leitor para funcionarem. Uma vez que estas *tags* possuem alimentação, o raio de alcance será muito superior ao das *tags* passivas. Em função do tipo de alimentação utilizada, este tipo de *tags* pode possuir outras funcionalidades, como sejam, o processamento ou registo e medição de parâmetros através da incorporação de sensores, maior capacidade de memória, "independência" e capacidade de iniciar uma transmissão autonomamente mesmo sem a presença de um leitor. Dadas estas características, é espectável que este tipo de *tags* tenha maiores dimensões devido à presença de uma bateria e de mais circuitos e componentes para outras funcionalidades.

A principal desvantagem das *tags* ativas compreende a necessidade de manutenção, tendo em conta o fim do ciclo de vida das baterias. Quanto mais funções tenha a *tag*, maior será o seu consumo e custo, para além disso as *tags* ativas não suportam condições ambientais (de temperatura, humidade e pressão) tão extremas como as *tags* passivas.

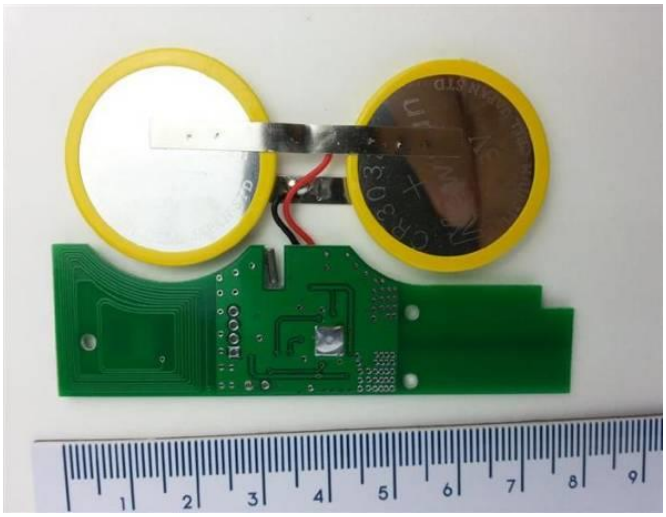

Figura 6 – Tag RFID ativa.

Tal como nas *tags* passivas as respostas das *tags* semi-ativas (ou semi-passivas), para o leitor, é realizada utilizando a energia recebida pelo leitor, no entanto estas *tags* tal como as *tags* ativas possuem uma bateria de alimentação. Esta bateria serve neste tipo de *tags* somente para alimentar os competentes eletrónicos e os sensores que possam estar presentes na própria *tag*. Dadas as suas características, este tipo de *tag* é equipado com antenas iguais às antenas presentes nas *tags* passivas.

Para além das diferenças já referidas, as *tags* ativas têm uma maior capacidade de armazenamento, maior rapidez de acesso múltiplo e melhores soluções e aplicações de segurança. Contudo, pelo facto de serem mais dispendiosas para muitas aplicações, usam-se as *tags* passivas que são consideravelmente mais baratas e igualmente fiáveis.

2.4. Princípios de Funcionamento

Quando uma antena de um sistema RFID transmite sinais de rádio, esses sinais são detetados pela *tag*, a qual emite um sinal de rádio como resposta. Esse sinal é posteriormente interpretado pelo recetor. Caso a *tag* tenha alguma capacidade de processamento, esta pode efetuar encriptação e desencriptação de dados. Algumas *tags* só permitem a leitura aos seus dados, enquanto outras permitem leitura e escrita.

Existem dois grupos de tipos de funcionamento (Finkenzeller, 2003):

- *Bit Transponder*;
- *Full e Half Duplex.*

No grupo do *bit transponder*, é feita a transmissão de apenas um bit ou de uma sequência, de informação entre o *tag* e o leitor, após a qual a comunicação é terminada. Neste conjunto, estão os sistemas tipo On-Off, típicos dos sistemas de alarme das lojas comerciais ou em sensores de movimento. Por exemplo, a *tag* de um produto numa loja responde sempre um bit que significa que não foi vendida até esta ser registada como vendida. Assim ao passar pelo leitor situado na saída da loja, se o produto não estiver marcado como vendido este emite um sinal de aviso. Este tipo de funcionamento é caracterizado por ser rápido e apenas precisar de uma resposta da *tag*. As *tags* deste grupo não precisam de muita memória, nem de eletrónica complexa para funcionar.

No grupo de *full duplex e half duplex,* existe uma maior quantidade de informação transmitida entre a *tag* e o leitor e maiores períodos de tempo de comunicação.

Dentro do grupo "*Full e Half Duplex*", subdividem-se outros 3 grupos:

- *Half Duplex* (HDX);
- Sequencial (SEQ);
- *Full Duplex* (FDX).

No funcionamento em *half-duplex,* a transmissão de dados entre o leitor e a *tag* é feita alternadamente, isto é, cada elemento transmite a informação de uma forma alternada. Este método permite uma grande simplificação da eletrónica envolvida. Neste método o leitor envia constantemente energia à *tag*, para que esta tenha energia para responder.

O funcionamento em modo sequencial é semelhante ao funcionamento em *half duplex*, a principal diferença reside no facto de agora o leitor apenas enviar energia quando está a transmitir para a *tag*. Neste cenário a *tag* tem capacidade de armazenar energia e utilizá-la na transmissão.

No modo de funcionamento *full duplex*, tanto o leitor como a *tag* estão a transmitir simultaneamente, comunicando dados nos dois sentidos. Para que este método seja possível é necessário que a transmissão seja realizada numa frequência diferente da do leitor, sendo pois necessário que a *tag* tenha um emissor próprio.

2.5. Bandas de Operação, Protocolos e Normas

Existe um número considerável de frequências, ou bandas de frequência que os sistemas RFID podem utilizar. A frequência utilizada por um sistema RFID determina muitas das características do seu funcionamento. Assim sendo, a escolha correta da banda de frequência a usar, é uma decisão importante no processo de desenvolvimento de uma solução tecnológica. A tabela seguinte apresenta três bandas distintas, bem como as características e aplicações típicas que lhe estão associadas.

Tabela 2 – Frequências de operação RFID.

Banda de Frequência	Características	Aplicações Típicas
Baixa: 100 a 500 kHz	- Faixa de curta até média leitura - Baixo custo - Baixa velocidade de leitura	- Controle de acesso - Identificação animal - Controle de inventário
Média: 10 a 15 MHz (também chamada Alta)	- Faixa de curta até média leitura - Potencialmente de baixo custo - Média velocidade de leitura	- Controle de acesso - Smart cards
Alta: 850 a 950 MHz e 2.4 a 5.8 GHz (também chamada Ultra Alta)	- Faixa larga de leitura - Alta velocidade de leitura - Alto custo - Linha de visão requerida	- Monitoração de veículos em estradas

As bandas de frequência de RFID estão ligadas às bandas de frequência *Industrial, Scientific and Medical* (ISM) como se apresenta na Figura 7. Embora as bandas ISM sejam bandas sem licenciamento, cada pedaço de espectro é disputado ao máximo o que provoca um controlo rigoroso do cumprimento das normas associadas à sua utilização em cada país.

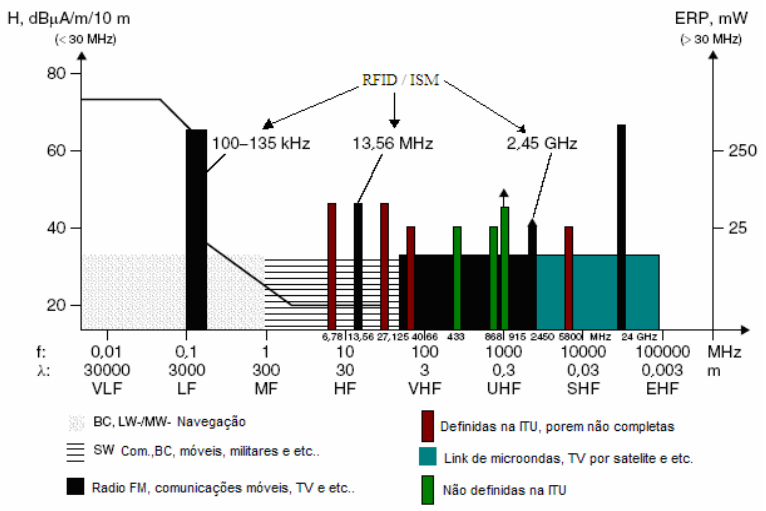

Figura 7 – Frequências de funcionamento.

Para que os equipamentos RFID (*tags* e leitores) sejam compatíveis entre si, é necessário que existam protocolos e/ou normas, para que qualquer *tag* de um determinado padrão e frequência, seja reconhecida por qualquer leitor dessa mesma frequência. Assim sendo, é fundamental a existência de regras (protocolos de comunicação), para o efeito.

As normas são criadas por diversas organizações para facilitar a interoperabilidade entre os componentes que formam um produto e que depois será fabricado por diferentes empresas. As normas definem não só o projeto de *hardware*, *software* mas também definem a sua utilização.

Existem diversos organismos responsáveis pela normalização e regulamentação, de entre eles, refira-se aqueles que se dedicam exclusivamente à criação de normas como são o *International Standards Organization* (ISO), *Internacional Electrotechnical Comission* (IEC), o *American National Standards Institute* (ANSI) e o *European Telecommunications Standards Institute* (ETSI).

A ISO e a IEC criaram uma comissão chamada de SC31 que trata especificamente dos aspetos de normalização dos protocolos e normas RFID.

As normas criadas por empresas são propriedade intelectual destas empresas, assim os fabricantes que as queiram usar terão que pagar *royalties*. Estes tipos de normas são chamados de normas proprietárias como o caso Mifare, da NXP Semiconductors. As normas criadas por organizações como a ISO são abertas e estão disponíveis sem custo ou a um custo muito reduzido.

Em sistemas RFID são usados tipicamente quatro tipos de categorias para qualificar as normas: normas envolvendo informação, normas de conformidade, normas aplicativas e normas tecnológicas.

A norma tecnológica define as especificações do *hardware* e o *software*. Esta fornece detalhes sobre a comunicação entre a *tag* e o leitor, a banda de frequência, a modulação dos sinais e esquemas de codificação da informação digital.

A norma envolvendo informação define o processo de leitura da informação das *tags* RFID e a forma como esta deve ser apresentada nas aplicações. As normas de conformidade definem métodos de teste para determinar a conformidade dos dispositivos (*tags* e leitores) a um tipo de norma. As normas aplicativas definem como, por exemplo, poderá ser colocada uma *tag* RFID num contentor.

Algumas das normas referentes a tecnologia RFID são:

- Normas envolvendo informação:
    - ISO 15424 – Identificadores de portadoras de informação;
    - ISO 15962 – Protocolo de informação: codificação;
    - ISO 15963 – ID único na *tag*.
- Normas de conformidade:
    - ISO 18046 – Métodos de teste no desempenho de dispositivos RFID;
    - ISO 18047 – Métodos de teste na conformidade de dispositivos RFID.
- Normas aplicativas:
    - ISO 10374 – Identificação automática de contentores de carga;
    - ISO 18185 – Comunicação RF para o selo eletrónico em contentores de carga.
- Normas tecnológicas:
    - ISO 18000 – Define normas na *interface* ar entre *tags* e leitores a várias frequências.

A norma ISO 18000 divide-se em sete partes, cada uma delas aborda temas diferentes e que especificam matérias diferentes como:

- ISO 18000-1: *standard*;
- ISO 18000-2: para a comunicação a frequências abaixo 135 kHz;
- ISO 18000-3: para uma frequência operacional em 13,56 MHz;
- ISO 18000-4: para uma frequência de 2,45 GHz;
- ISO 18000-6: para frequências entre 860 e 930 MHz;
- ISO 18000-7: para uma operação em 433 MHz.

Os regulamentos relativos ao espectro eletromagnético são elaborados por entidades internacionais, de que são exemplo o *International Telecommunication Union* (ITU) e o ETSI. As diretivas emanadas são posteriormente aplicadas pela autoridade reguladora das comunicações eletrónicas. Em Portugal estas funções são exercidas pela Autoridade Nacional de Comunicações (ANACOM).

Os regulamentos variam de país para país devido à diferença no uso de várias porções do espectro eletromagnético. Para os cartões e leitores de RFID os regulamentos englobam os seguintes fatores:

- Potência do campo eletromagnético – é usada na medição o *Effective Isotropic Radiated Power* (EIRP) em watts;
- Uso do espectro eletromagnético – a gama de frequências atribuídas;
- Espaçamento de canais – como atribuir /dividir os canais de comunicação na faixa de frequência;
- *Duty cycle* – Percentagem do tempo que o leitor pode efetivamente transmitir.

No Anexo I, é possível consultar um excerto do Quadro Nacional de Atribuição de Frequências (QNAF), onde podem ser consultados, entre outros, os valores máximos da potência e/ou campo transmitida, consoante a frequência de operação e o tipo de aplicação.

2.6. Conclusões do Capítulo

Neste capítulo foi realizada uma introdução histórica do aparecimento da tecnologia RFID, desde os seus primórdios, durante a segunda guerra mundial até aos sistemas atuais mais complexos de localização de produtos, controlo de acessos, sistemas anti-roubo e sistemas de portagem eletrónica. Adicionalmente foram apresentadas as características dos diversos tipos de *tags*, dos diversos modos de funcionamento para além dos aspetos regulamentares associados a esta tecnologia.

3. CONCEÇÃO DO SISTEMA

Neste capítulo pretende-se caracterizar a IES onde se pretende implementar o sistema para de seguida definir as funcionalidades do mesmo com base nas especificidades da instituição em análise, nomeadamente no que concerne à localização dos diferentes edifícios, a localização das entradas (pontos de acesso para o exterior do *campus*), a localização dos laboratórios e a caracterização dos seus utilizadores habituais: docente e discentes. Baseado nesta informação são tomadas algumas decisões relativas à informação que se pretende armazenar em suporte digital durante o processo de controlo e gestão de acessos e assiduidade dos trabalhadores e dos docentes e alunos nas aulas.

Para definir as funcionalidades que se pretendem implementar e os procedimentos associados a cada uma delas, nas diferentes vertentes do sistema, foram concebidos diversos fluxogramas nos quais se representa a tomada de decisões em função das variáveis do sistema. Desta forma simples e gráfica será possível consultar o funcionamento do sistema, e assim, antever otimizações e efetuar as modificações necessárias para que este seja o mais funcional, fiável e seguro possível.

3.1. O Instituto Superior de Engenharia de Coimbra – ISEC

O Instituto Superior de Engenharia de Coimbra (ISEC) é uma unidade orgânica do IPC sediado na Quinta da Nora, Coimbra num campus com cerca de 92100 m2 usualmente frequentado por 3500 pessoas (alunos e docentes e trabalhadores). No interior do campus existe um conjunto de edifícios, parques de estacionamento, espaços verdes e arruamentos com a distribuição presente na Figura 8, obtida a partir da aplicação Google Earth.

O sistema que se pretende desenvolver tem por objetivo efetuar o controlo de acessos ao campus do ISEC, o controlo de entrada/saída dos trabalhadores (pica-ponto) e o controlo de assiduidade dos alunos.

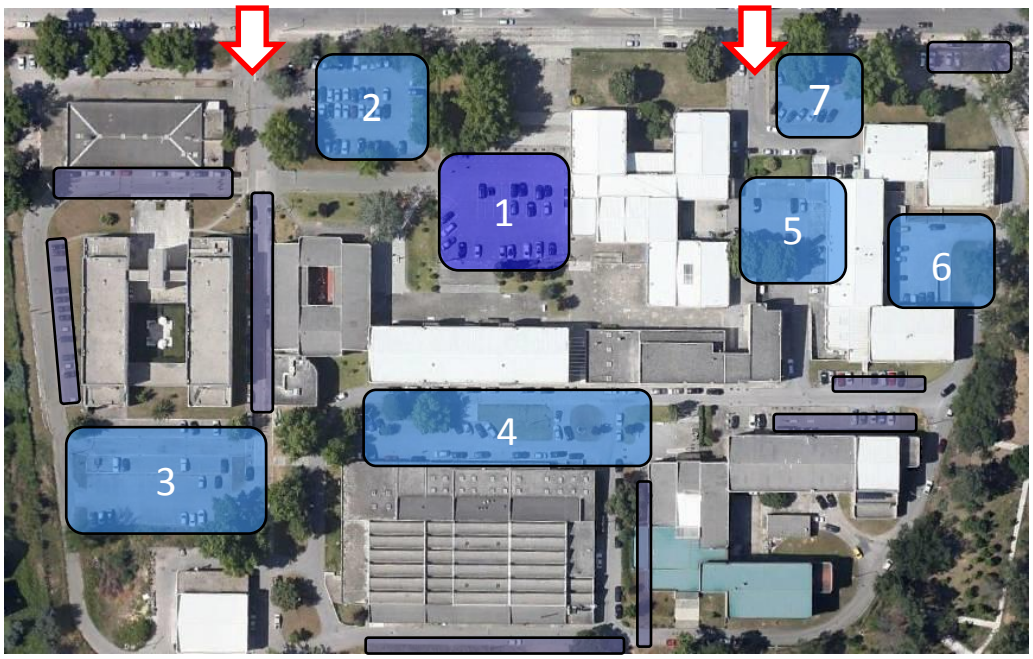

Figura 8 – Vista aérea do campus do ISEC.

Na Figura 8 pode-se observar a existência de duas entradas principais com ligação à estrada e cancelas de acesso ao campus, 12 edifícios, 7 parques de estacionamento e diversos outros lugares de estacionamento nos arruamentos existentes no interior do *campus*.

No que concerne aos edifícios situados no campus do ISEC refira-se os seguintes:

- Auditório;
- Cantina;
- Edifício do Departamento de Engenharia Civil (DEC);
- Edifício do Departamento de Engenharia Eletrotécnica (DEE);
- Edifício do Departamento de Engenharia Informática e de Sistemas e biblioteca (DEIS);
- Edifício do Departamento de Engenharia Mecânica (DEM);
- Edifício de Engenharia Eletromecânica;
- Edifício do Departamento de Engenharia Química e Biológica (DEQB);
- Edifício dos Gerais - Departamento de Física e Matemática (DFM)
- Oficinas de Civil & CET automóvel – antiga fundição;
- Oficinas de Mecânica;
- Presidência, serviços académicos, bar e associação de estudantes.

O acesso à maioria dos edifícios do campus é atualmente realizado por um sistema de acessos integrado numa plataforma de controlo de intrusão. Nesta tese propõe-se o desenvolvimento de uma nova solução integrada de controlo de acessos e sua gestão. Como ponto de partida escolheu-se a entrada do Auditório pela porta da cave como zona de testes.

No ISEC existem os seguintes parques de estacionamento:

1. Parque de estacionamento da Presidência;
2. Parque de estacionamento da entrada Oeste (DEC & Cantina);
3. Parque de estacionamento do DEIS;
4. Parque de estacionamento do DEM, oficinas e DFM (Gerais);
5. Parque de estacionamento do DEQB, DEE, bar;
6. Parque de estacionamento do DEQB (Este);
7. Parque de estacionamento da entrada Este (Associação de Estudantes);

3.2. Controlo de Acessos aos Laboratórios e Salas de Aulas

O sistema de controlo de acessos a laboratórios, permite o controlo dos acessos aos laboratórios dos diferentes departamentos nomeadamente aos professores, alunos e técnicos. Além disso permite que os alunos registados acedam aos mesmos fora do período de aulas, em períodos previamente definidos, desde que devidamente autorizados.

Na Figura 9 é possível verificar que o funcionamento deste sistema consiste na verificação constante da presença de uma *tag*, em dois leitores de RFID, um associado à entrada de um utilizador e outro à sua saída. Quando se verifica a presença de uma *tag*, o sistema envia para o servidor o ID da *tag* em conjunto com o ID do leitor onde esta foi lida. O servidor recebe a informação enviada pela plataforma e de acordo com os dados anteriormente registados na base de dados, responde para o local onde foi identificada a *tag*, autorizando ou não a entrada do utilizador. A plataforma recebe a resposta do servidor e de acordo com o seu conteúdo, dá ao utilizador acesso físico ao laboratório caso este tenha autorização de acesso ao local pretendido, registando a atividade relativa aos dados enviados pela plataforma.

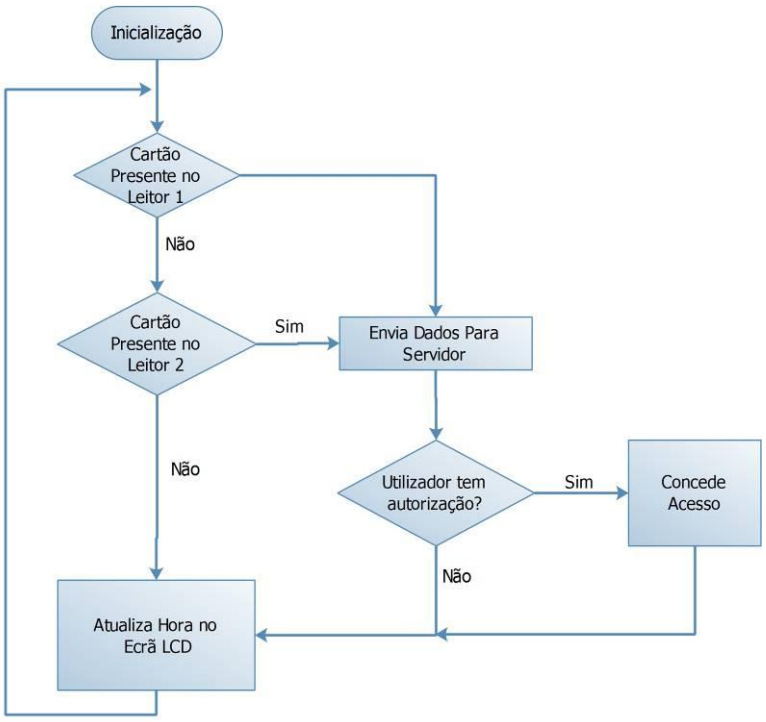

Figura 9 – Fluxograma de funcionamento do sistema de controlo de acessos.

3.3. Gestão de Presenças de Alunos

O sistema de registo de assiduidade às aulas por parte dos alunos foi idealizado para dois cenários destintos de duas formas diferentes. Uma vertente fixa, integrada no sistema de controlo de acesso anteriormente descrito, ou uma vertente móvel utilizada somente para monitorização de presenças. Assim, os dois sistemas servirão para registar e monitorizar as presenças de alunos nas aulas, evitando a utilização de folhas de registo em papel, e evitando fraudes na utilização do cartão bancário da CGD.

A vertente fixa da plataforma de gestão de presenças de alunos em aulas, consiste na fixação a título permanente da plataforma nas entradas das salas de aulas. Assim, aquando da entrada na sala de aula, o docente regista o início da referida aula, seguido dos alunos, os quais registam a sua presença nessa aula específica. Assim também é possível fazer a gestão dos acessos à sala de aula através da instalação de um trinco elétrico na porta da sala de aula, combinando a gestão de presenças à gestão de acessos a um espaço restrito.

Por outro lado, na vertente móvel da gestão de presenças de alunos em aulas, a partir da ideia da evolução das folhas de aulas para um livro de presenças digital, o sistema móvel consiste na utilização da plataforma de uma maneira *offline*.

O sistema é agora portátil e equipado com uma bateria o que possibilita o seu funcionamento sem alimentação externa. Após a identificação de um docente, o sistema inicia o registo dos alunos num cartão de memória. No fim da aula, o processo é concluído pelo docente. Após a conclusão da aula, os registos criados serão descarregados para o servidor através de uma ligação à rede. Durante esse período a bateria será carregada, bem como nos períodos que antecedem a primeira aula do dia e sucedem à última aula do dia.

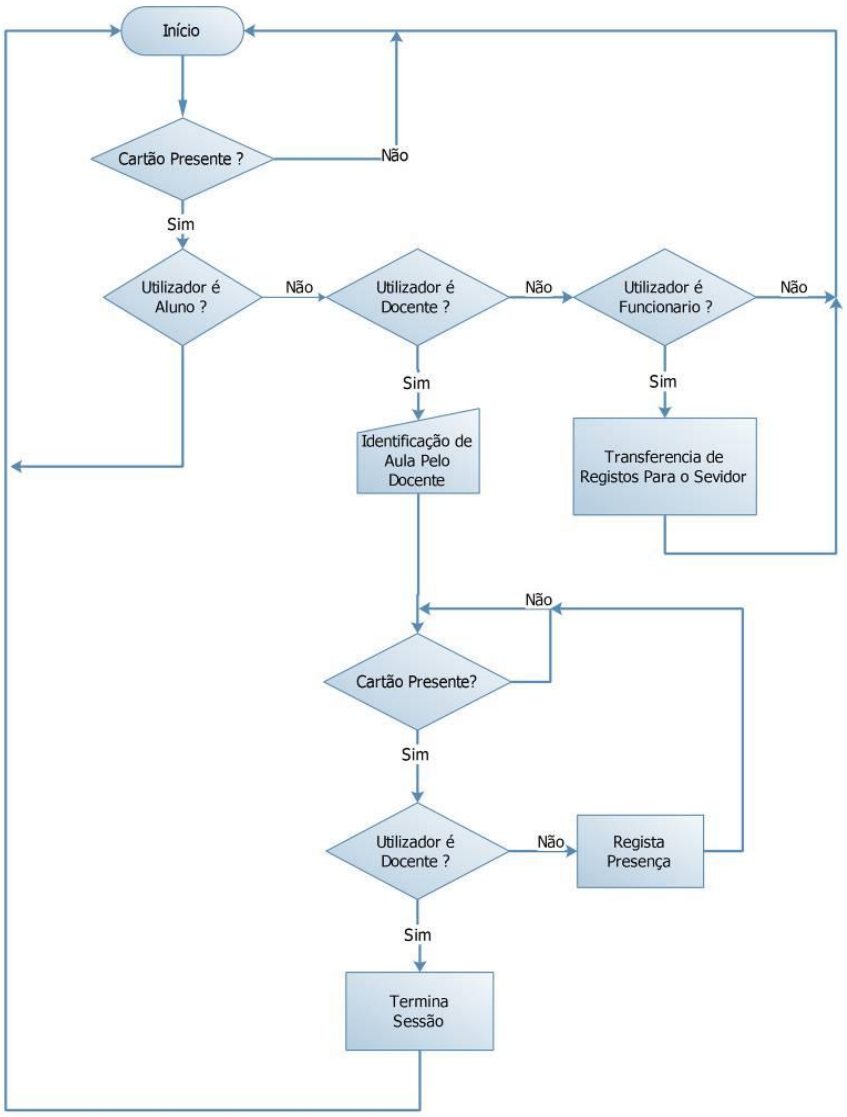

Figura 10 – Fluxograma do funcionamento do livro de ponto móvel.

No que concerne ao funcionamento do livro de ponto móvel, este segue as operações presentes no fluxograma da Figura 10.

Na prática, o sistema está constantemente a verificar se existe *tag* de RFID presente no leitor. Se não existir nenhuma *tag*, o sistema continua a verificação, mas se por outro lado, existir uma tag presente, este vai identificar que tipo de utilizador apresentou a *tag*.

Se a *tag* inicialmente identificada não pertencer a um funcionário (docente ou trabalhador), o sistema ignora a *tag* apresentada e recomeça o processo de procura de *tag*. Por outro lado, se o utilizador for um docente, o sistema pede a identificação da aula através do teclado e de seguida começa a detetar e registar as *tags* dos alunos até ao final da aula. Caso o docente volte a apresentar a sua *tag* o sistema pressupõe que a aula foi terminada. Depois o sistema volta ao início verificando constantemente a presença de uma *tag* no leitor.

Por outro lado, se a *tag* inicialmente detetada for de um trabalhador, o sistema faz ligação ao servidor para transferir os registos, que foram anteriormente efetuados em aulas, para o servidor para estes serem posteriormente analisados e utilizados.

3.4. Gestão de Registo de Ponto de Trabalhadores na Instituição

O sistema de registo de ponto dos trabalhadores, deverá identificar e registar a entrada e saída dos mesmos na instituição, enviando essa informação para um servidor dedicado. Posteriormente uma aplicação desenvolvida para o efeito contabilizará as horas de trabalho, as faltas e respetivas justificações, bem como o estado atual da presença dos trabalhadores na instituição (presente, ausente).

O funcionamento deste sistema está representado na Figura 11. A cada ciclo é atualizada a data e hora no display, de seguida é verificada a presença de *tag*, se existir uma *tag* presente, esta presença é registada. O registo desta *tag* é feito no servidor caso exista ligação ao mesmo, caso contrário, este registo é feito no cartão de memória e o sistema fica com a indicação de que existiu um problema de ligação ao servidor. Ainda no mesmo ciclo, caso tenha existido uma falha de ligação anteriormente e caso não tenha sido feito nenhum teste de conectividade no último minuto, é invocada uma função que testa a conectividade com o servidor.

Para finalizar o ciclo, caso não tenha existido uma verificação de registos no cartão de memória nos últimos 5 minutos, e caso o sistema tenha indicação que existe ligação ao servidor, o sistema verifica se existem registos guardados no cartão de memória. Caso essa verificação seja positiva, os registos são enviados para o servidor e posteriormente apagados do cartão de memória, voltando assim ao início do ciclo.

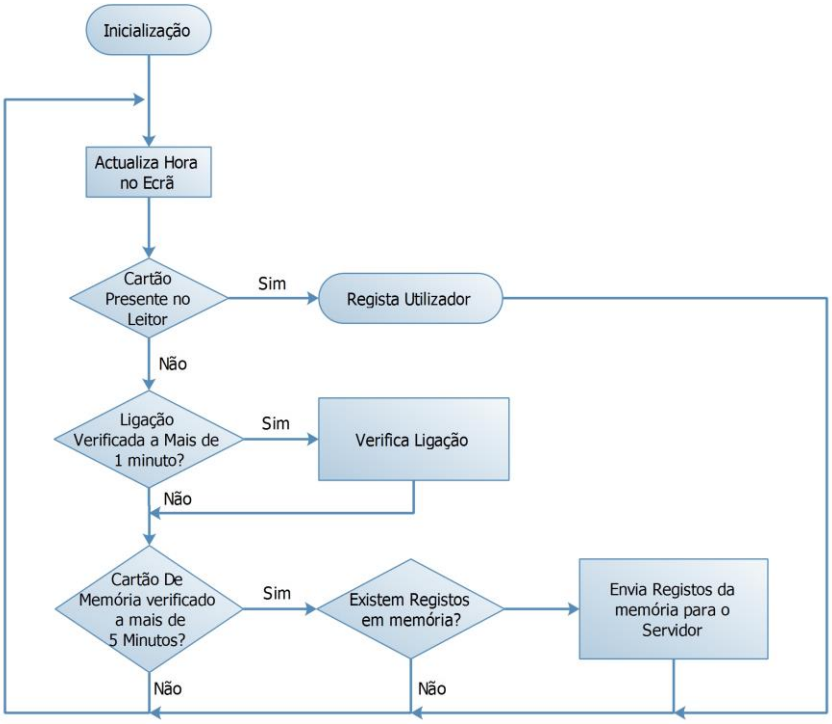

Figura 11 – Fluxograma de funcionamento de registo de ponto de trabalhador.

3.5. O Controlo de Acessos ao Campus do ISEC

A gestão do campus do ISEC é um dos aspetos mais importantes para garantir o bom funcionamento de toda a instituição. Assim sendo, e de forma a contribuir para a sua gestão desenvolveu-se uma plataforma capaz de monitorizar e controlar os acesos ao campus. Este sistema deverá ser capaz de identificar todos os utilizadores aquando da sua entrada, particularmente aqueles que o façam usando automóvel próprio.

Uma vez que os lugares nos parques de estacionamento são em número limitado, existe a necessidade de restringir o acesso entre os seus utilizadores e evitar a sua utilização a pessoas exteriores ao instituto, ressalvando os convidados e distribuidores de equipamento, materiais, gás e/ou outros serviços necessários ao funcionamento do ISEC. A restrição de acesso física é feita por meio de cancelas (Figura 12 e Figura 13) e com pontos de leitura de *tag's* RFID, ambos existentes no local, mas que necessitarão de ajustes e intervenções para os adaptar à nova plataforma.

O funcionamento deste sistema é idêntico ao do sistema descrito na secção 3.2, alterando apenas o tipo de barreira física.

Figura 12 – Entrada Oeste do campus.

Figura 13 – Entrada Este do campus.

3.6. Conclusões do Capítulo

Neste capítulo é apresentada a conceção do sistema, primeiro com a descrição mais pormenorizada do meio onde este sistema pretende ser implementado, bem como o conjunto de necessidades à qual esta proposta pretende demonstrar ser solução. Neste capítulo é também descrito o princípio de funcionamento do sistema, utilizando fluxogramas relativos ao funcionamento de cada uma das plataformas. Esses fluxogramas descrevem com detalhe acrescido, as ações e decisões tomadas na implementação desses subsistemas de controlo e gestão de acessos ao campus e laboratórios, e controlo da assiduidade dos trabalhadores e dos docentes e alunos nas aulas.

4. DESENVOLVIMENTO DE PLATAFORMA

Um dos aspetos fundamentais no desenvolvimento de uma dada solução tecnológica, compreende a seleção do *hardware* e *software* a utilizar. Assim sendo, as secções seguintes descrevem os diferentes passos que compreenderam a seleção de componentes e a sua interligação de forma a que estes cumpram os requisitos do sistema enunciados anteriormente.

4.1. Seleção de Componentes

Relativamente à escolha de *hardware* os fatores mais relevantes compreendem o custo (fator económico), a fiabilidade, a funcionalidade, a segurança e a flexibilidade de forma a que seja possível realizar futuras melhorias no sistema (*upgrades*).

No desenvolvimento do sistema de controlo e gestão de acessos, foi escolhida uma placa de desenvolvimento com recurso ao microcontrolador ATMEL (Arduíno), em conjunto com um leitor RFID (13.56 MHz Mifare Read/Write Module, 2012) que conecta ao servidor, através de um módulo de comunicação Ethernet (Arduíno *Ethernet Shield*), que possui uma base de dados (MYSQL), para armazenar todos os registos e informações necessárias para efetuar a gestão de todo o sistema.

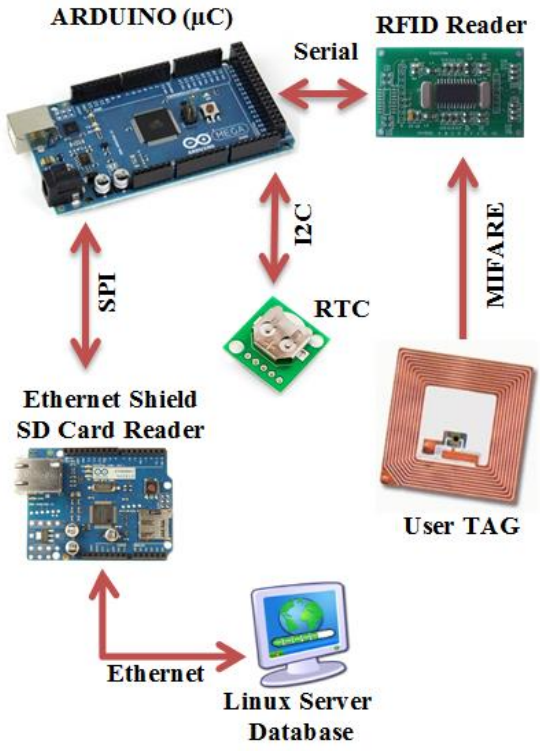

Figura 14 – Arquitetura do sistema.

Na Figura 14 é ilustrada a arquitetura do sistema que se pretende desenvolver, os componentes utilizados e os protocolos através dos quais os módulos comunicam entre si. Para o processamento foi escolhida a placa de desenvolvimento "Arduíno MEGA", que através de comunicação série com o leitor de RFID YHY502 da EHUOYAN faz a leitura das *tags* Mifare (13,56 MHz). Para que seja possível ao microcontrolador ter uma referência temporal exata, existe um Relógio de Tempo Real, "*Real Time Clock Module*" da SparkFun, que comunica com o microcontrolador através do protocolo *Inter-Integrated Circuit* (I2C). A informação do sistema é transmitida ou armazenada num cartão de memória, utilizando para o efeito o módulo de comunicação "Arduíno Ethernet Shield".

4.1.1. RFID

Como foi dito anteriormente, optou-se pela utilização do sistema embebido nos cartões de estudante, professores e trabalhadores existentes, os quais são fornecidos pela Caixa Geral de Depósitos (CGD) (Caixa Geral de Depositos, 2012), e incorporam tecnologia RFID Mifare.

Importa referir que o Cartão Universidade Politécnico (CUP) recebeu um galardão internacional de inovação tecnológica, entre cerca de 500 cartões de débito e crédito. Este reconhecimento é devido às 4 tecnologias embutidas: código de barras, banda magnética, *chip* de leitura de contacto *Europay, MasterCard and Visa* (EMV) e *chip* de leitura sem contacto (Mifare). Este cartão foi idealizado especialmente para o meio académico com a agregação das vertentes bancárias e identificação, destinado a todos os alunos, docentes e trabalhadores das Instituições do Ensino Superior (IES) que tenham protocolos válidos com a CGD.

Com a utilização das tecnologias de EMV e Mifare, e com o aumento da capacidade de armazenamento de informação, este cartão tornou-se um cartão multisserviços, ou seja, para além da sua função bancária (chip EMV), este cartão pode ser utilizado como diferentes funções, nomeadamente (Caixa Geral de Depositos, 2012):

- Cartão de identificação académica;
- Cartão de acessos à escola e/ou espaços dentro da escola (ex. parques de estacionamentos, laboratórios);
- Cartão de biblioteca, através da utilização do código de barras impresso no cartão;
- Possibilidade de incorporar aplicações específicas das próprias escolas;

- Eliminação da circulação de dinheiro no espaço da escola, através de uma solução de pagamentos de baixo valor;
- Possibilidade de incorporar aplicações específicas fora do âmbito da escola, tais como os sistemas de bilhética das mais variadas áreas, sistemas de controlo de acessos autorizados e outras aplicações cuja parametrização se enquadre na arquitetura tecnológica do cartão.

Tendo por base o requisito de utilização dos cartões CGD, foi iniciada a procura de um leitor RFID compatível. As primeiras pesquisas mostraram que os leitores disponíveis no mercado provenientes de fornecedores portugueses, teriam um custo demasiado elevado, assim a pesquisa passou a abranger fornecedores internacionais.

O módulo escolhido foi o "YHY-502CTG" do fornecedor "EHUOYAN" (13.56 MHz Mifare Read/Write Module, 2012) (Figura 15), o qual tem a capacidade de ler e escrever. Com este dispositivo, é possível ler o número de série do cartão para identificação do utilizador e ler e escrever na memória das *tags*.

Este leitor é extremamente compacto, uma vez que a antena encontra-se integrada no módulo e funciona sobre a norma ISO144443A (Mifare). O controlo desde módulo é feito pelo microcontrolador através de uma *interface Universal Asynchronous Receiver/Transmitter* (UART), com um débito de 19200 bps. O leitor necessita de uma fonte de alimentação de 5 V, sendo que em aplicações de curta distância, esta alimentação pode ser obtida diretamente da placa de microcontrolador.

Figura 15 – Módulo leitor RFID.

4.1.2. Microcontrolador

O passo seguinte compreendeu a seleção do microcontrolador, o qual será o elemento mais importante no cumprimento dos requisitos da plataforma que se pretende desenvolver. Este processo de seleção compreendeu um estudo de mercado que comparou, o preço de custo, as funcionalidades, a informação disponível e a aplicação do sistema. Como resultado desse estudo, foi escolhido o Arduíno como plataforma de trabalho, para fazer todo o controlo do *hardware* e a comunicação com o servidor local utilizando para o efeito um *shield* específico.

O Arduíno é uma plataforma *open-Soure* e *open-hardware* que revolucionou o desenvolvimento de aplicações baseadas em microcontroladores devido ao facto do seu *software* ser livre e do *hardware* ser de fácil utilização e aplicação.

Este equipamento permite um rápido e eficiente desenvolvimento de projetos, uma vez que permite uma programação simples em linguagem C, ultrapassando questões inerentes à complexidade da programação de baixo nível (e.g., *Assembly*).

O primeiro modelo escolhido, foi o Arduíno UNO, cuja ilustração se apresenta na Figura 16.

O Arduíno UNO (Arduino, 2011) é uma placa de desenvolvimento de microcontrolador, baseada no microcontrolador ATmega328. Nela existem 14 entradas / saídas digitais (das quais 6 podem ser usadas como saídas *Pulse With Modulation* (PWM), 6 entradas analógicas, 1 UART's (porta série) usada para ligar o leitor RFID, uma porta *Serial Peripheral Interface* (SPI) (utilizada para comunicar com o módulo Ethernet e cartão SD), um *bus* I2C (utilizado para ligar o microcontrolador ao relógio de tempo real), um cristal oscilador de 16 MHz de frequência e uma ligação USB (que utiliza a porta UART).

Figura 16 – Arduíno UNO.

Após a definição de todos os componentes, verificou-se que o Arduíno UNO poderia não ter memória suficiente para todo o código, tendo em conta que o programa utilizado utiliza um conjunto significativo de bibliotecas responsáveis pela interligação com o LCD, RTC, e módulo Ethernet. Assim sendo, foi escolhido o Arduíno Mega como plataforma de trabalho, de forma a possibilitar futuros *upgrades*.

O Arduíno Mega 2560 (Arduino, 2011) é uma placa de desenvolvimento de microcontrolador, baseada no ATmega2560. A referida plataforma inclui 45 entradas / saídas digitais (das quais 14 podem ser usadas como saídas PWM), 16 entradas analógicas, 4 UART's (portas série), 4 portas SPI, um *bus* I2C, um cristal oscilador a 16 MHz de frequência e uma porta USB.

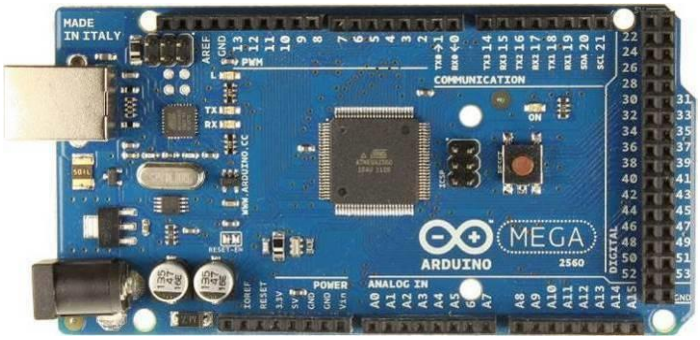

Figura 17– Arduíno MEGA 2560.

O Arduíno Mega, presente na Figura 17, foi escolhido essencialmente devido às limitações do Arduíno UNO. Uma vez que o Arduíno Mega tem 4 portas série (UART's), permite processar "simultaneamente" até 4 leitores RFID, ou 3 leitores RFID mais uma porta USB (para programação e *debugging*). Esta particularidade torna assim possível uma redução de custo nalgumas aplicações. Por exemplo, a mesma unidade de controlo (Arduíno Mega), pode ser utilizada para gerir e controlar as presenças e o acesso dos estudantes em duas salas contíguas, com portas adjacentes estando em cada uma delas dois leitores RFID (um de entrada e um de saída).

A Tabela 3, sumariza as principais características dos microcontroladores Atmega 328 e Atmega 2560 presentes respetivamente nas placas de desenvolvimento Arduíno Uno e Arduíno Mega.

Tabela 3 – Características microcontroladores Atmega 328 e Atmega 2560 (Atmel).

	Atmega 328	Atmega 2560
Flash (KBytes)	32 KBytes	256 KBytes
Numero de Pinos do Integrado	32	100
Frequência Máxima	20 MHz	16 MHz
Processador	8-bit AVR	8-bit AVR
Numero de I/O's	23	86
Interrupções Externas	24	32
Portas SPI	2	5
Portas TWI (I2C)	1	1
Portas UART	1	4
Entradas analógicas (ADC)	8	16
Resolução das ADC (bits)	10	10
Velocidade das ADC (ksps)	15	15
SRAM (KBytes)	2	8
EEPROM (Bytes)	1024	4096
Tensão de Operação (Vcc)	1.8 - 5.5	1.8 - 5.5
Temporizadores	3	6
PWM	6	15
Watchdog	Sim	Sim

A seleção do Arduíno Mega deveu-se essencialmente à quantidade de memória que ele possui, a qual permite a implementação de vários tipos de protocolos de comunicação como Ethernet, série, e I2C. Testes preliminares com o Arduíno UNO mostraram que a memória disponível era limitada para suportar todos estes protocolos.

4.1.3. Implementação do Módulo *Real Time Clock* (RTC)

O módulo de relógio de tempo real RTC é um integrado, usualmente utilizado em sistemas cujo funcionamento necessite de uma referência temporal exata. Isto é, sistemas que se mantenham sincronizados com a hora atual. Para poupar energia e para libertar recursos, esta função é assumida isoladamente por um circuito integrado específico o qual tem um ciclo de vida de pelo menos 9 anos e tem um consumo energético muito pequeno (500 nA) valores referência para o integrado DS1307 da Maxim, utilizado neste projeto. O processador principal poderia ter esta função, mas iria requerer processamento adicional e alimentação interrupta.

Este módulo pode ser opcional, dependendo do sistema ser tolerante a falhas ou não. No entanto, para que o sistema possa saber sempre qual é a hora atual e funcionar sem "perguntar" a hora ao servidor, torna-se necessário utilizar este módulo, embora nalguns sistemas não seja fundamental a sua existência.

No caso do sistema não ser resistente a falhas, o sincronismo de data e hora, é garantido pela ligação ao servidor no instante de cada leitura de uma *tag* RFID.

No desenvolvimento do sistema, o circuito foi inicialmente testado numa placa *BreadBoard* para testes de compatibilidade e programação. Após se terem obtidos bons resultados, foi desenvolvida a placa PCB presente na Figura 18 e na Figura 19, com todos os componentes necessários à sua implementação.

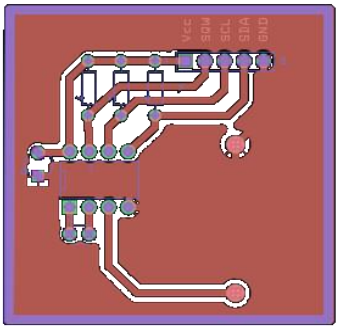

Figura 18 – Segunda versão do PCB do módulo de tempo real.

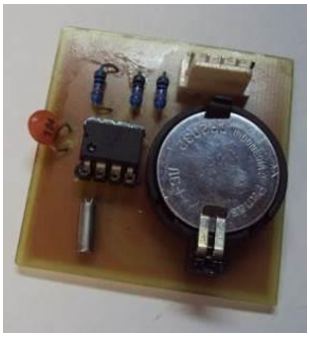

Figura 19 – Segunda versão do PCB desenvolvida com DS1307.

Numa fase mais avançada do desenvolvimento da plataforma, tanto por questões de custo de produção, como pela fiabilidade da construção e tamanho do módulo, foi utilizado um módulo RTC fabricado pela empresa SparkFun (SparkFun)(Figura 20). Este módulo traduz-se numa solução mais económica em comparação com a solução do (DS1307 V2) caso sejam considerados, os custos de produção da placa PCB, os componentes utilizados e a sua montagem.

Figura 20 – Módulo RTC SparkFun.

4.1.4. Display LCD

Outro requisito de um sistema desta natureza contempla a existência de um *display* onde se apresenta a informação relativa ao sistema, permitindo dessa forma visualizar as ações que são necessárias empreender e/ou a informação recolhida.

Para implementar esta *interface*, foi escolhido um ecrã de cristais líquidos (LCD) alfanumérico. Este tipo de ecrãs permite representar letras e números, e ainda nalguns casos específicos alguns caracteres especiais personalizados.

Existe uma grande variedade de *displays* LCD, no mercado, no que concerne a tamanhos e cores. Normalmente estes possuem de 1 a 4 linhas e de 8 a 40 caracteres por linha e encontram-se disponíveis em cores diversas como verde, vermelho, azul, amarelo e branco. A maioria dos LCDs possui a mesma lógica de funcionamento, sendo esse designado por HD44780. Os *display* LCDs possuem uma transparência controlada eletricamente que possibilita que a sua informação seja visível e define a forma de apresentar os caracteres. Os modelos mais simples usam a luz ambiente, no entanto existem outros que possuem uma iluminação traseira (*backlight*). O *backlight* é controlado simplesmente alimentando dois pinos que ligam um LED normalmente. Para economizar energia, é comum usar um pino do microcontrolador para ligar ou desligar esta alimentação.

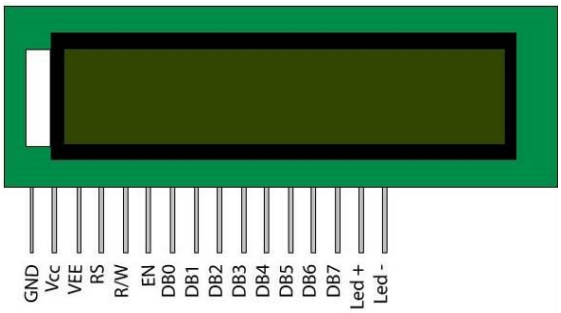

Figura 21 – Ligações ecrã alfanumérico HD44780.

Como se pode observar na Figura 21, a comunicação entre o ecrã e o microcontrolador é do tipo "paralela", em que um conjunto de bits (1 nible = 4 bits ou 1 byte = 8 bits) é transferido simultaneamente.

Os dados podem ser transferidos do microcontrolador para o ecrã (envio dados) ou no sentido contrário (receção de dados). O pino R/W determina a direção da comunicação sendo que o nível alto indica leitura e o nível baixo indica escrita.

Uma das funcionalidades do ecrã é a possibilidade de registar em memória própria até oito caracteres especiais. Para efetuar essa operação é necessário mudar o modo de funcionamento do ecrã. Normalmente para utilizar o ecrã, apresentar e/ou limpar caracteres de visualização, o pino "RS" encontra-se ativo (nível lógico alto), para que o ecrã entre no modo de escrita em memória é necessário desativar o pino "RS" (nível lógico baixo), de modo a que o ecrã interprete os comandos enviados como comandos de escrita em memória.

Por último, e mais importante, existe o pino E (*Enable*), que é quem comanda a execução de uma leitura ou escrita. Normalmente este sinal encontra-se no nível lógico baixo e os pinos de dados estão desconectados. Quando é necessário comunicar com o ecrã, o pino *Enable* cria o sincronismo entre o microcontrolador e o ecrã.

Inicialmente os testes de ligações e de códigos utilizaram um ecrã de 16 x 2 (16 caracteres e 2 linhas), representado na Figura 22, o que se revelou ser insuficiente para mostrar todas as informações necessárias.

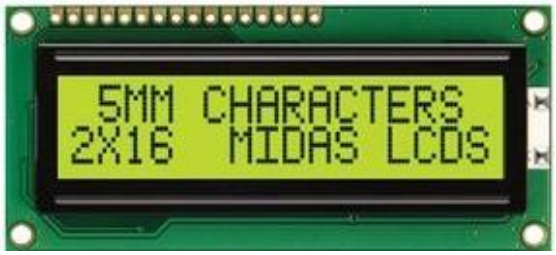

Figura 22 – LCD 2x16.

Para apresentar toda a informação necessária, foi escolhido o ecrã de 20 x 4 (20 caracteres e 4 linhas), representado na Figura 23, onde é possível organizar toda a informação que se pretenda visualizar. Numa segunda fase, foi ainda escolhido um terceiro modelo, também de 20 x 4, mas mais pequeno, o qual usa caracteres de 3 mm em vez dos caracteres de 5 mm usados pelos outros LCD's, possibilitando assim a implementação de plataformas mais pequenas com as mesmas funcionalidades.

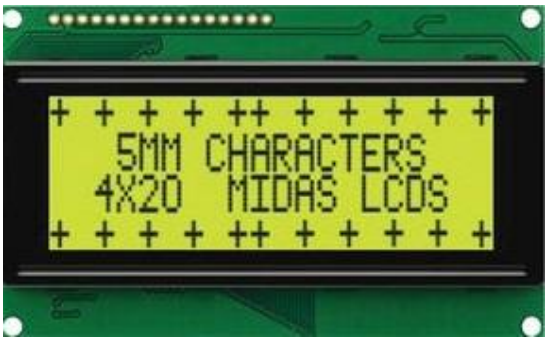

Figura 23 – Ecrã alfanumérico 20x4 5mm.

Para uma efetiva utilização do LCD, é necessário proceder a um conjunto de ligações entre o ecrã e o microcontrolador implementados. Assim sendo, foi desenvolvida para o efeito uma placa PCB cujo *layout* se apresenta na Figura 24, onde se soldam diretamente os pinos do ecrã. Esta placa tem como funções, efetuar as ligações necessárias para que o ecrã funcione corretamente e acomodar o potenciómetro responsável pelo controlo do contraste do ecrã, adicionalmente irá conter fichas para a ligação de cabos, que por sua vez ligam ao microcontrolador.

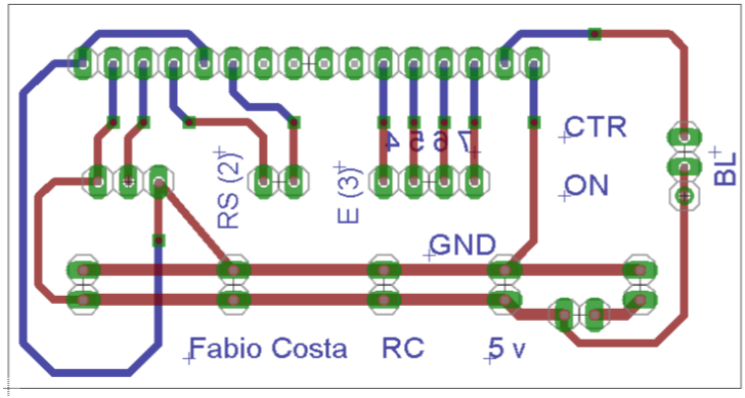

Figura 24 – Desenvolvimento CAD do módulo de controlo LCD.

Uma vez que foi necessário desenvolver um PCB de raiz, foi incluído nele um barramento de alimentação aproveitando os recursos existentes o que não originou um aumento de custos. Este barramento além de alimentar o ecrã LCD é composto por um conector de entrada de energia, quatro conectores de saída de energia e um conector, para ligação opcional de um interruptor, para que as saídas de energia possam ser desligadas da alimentação e deste modo desligar o sistema. Devido a questões de segurança, foi escolhido um interruptor com chave para ligar e desligar o circuito, representado na Figura 25.

Figura 25 – Módulo de controlo LCD instalado no LCD.

4.1.5. Módulo de Energia e Bateria

Para as plataformas portáteis que executam o controlo de acessos nas diferentes aulas foi necessário encontrar uma solução técnica para realizar a sua alimentação de uma forma autónoma. Essa solução compreende a existência de uma unidade de armazenamento de energia (bateria), um carregador para a referida bateria e um controlador da alimentação fornecida pela bateria.

O primeiro aspeto a ser analisado contemplou a seleção do tipo de bateria a usar, tendo por base o consumo da plataforma e o tempo de disponibilidade entre carregamentos. As baterias de iões de lítio (Li-Ion) possuem uma grande capacidade de armazenamento de energia por unidade de peso, superior às baterias de níquel e às baterias de chumbo.

Em comparação com as baterias de hidreto metálico de níquel (Ni-MH), as baterias de iões de lítio armazenam o dobro de energia de uma bateria de hidreto metálico de níquel e três vezes mais que uma bateria de níquel cádmio (ou NiCd). Outra diferença da bateria de iões de lítio é a ausência do efeito memória (não vicia) ao contrário da bateria de NiCd.

Uma bateria de iões de lítio pode armazenar cerca de 150 Wh/kg. Já uma bateria de NiMH pode armazenar até 100 Wh/kg, embora seja mais comum que estas tenham uma capacidade de 60 a 70 Wh. Refira-se que uma bateria de chumbo-ácido tem a capacidade de armazenar apenas 25 Wh/kg. Assim, usando a tecnologia chumbo-ácido, em termo comparativo, são necessários 6 kg de baterias do tipo chumbo-ácido, para armazenar a mesma quantidade de energia que uma bateria de iões de lítio de 1 kg.

As baterias de lítio têm uma capacidade de manter a carga durante mais tempo. Um conjunto de baterias lítio perde apenas cerca de 5 % da sua carga por mês, enquanto as baterias NiMH perdem 20 % no mesmo período.

As baterias de iões lítio conseguem suportar centenas de ciclos de carga/descarga, têm uma vida útil de aproximadamente 400-500 ciclos, desde que se tenham alguns cuidados específicos. As baterias iões lítio não podem ser descarregadas completamente, razão pela qual precisam de um controlador interno que gere os níveis de carga e de utilização, tornando-as portanto ainda mais caras.

As baterias de lítio são muito mais leves do que outros tipos de baterias recarregáveis do mesmo tamanho, tendo em conta que os seus elétrodos são feitos de lítio e carbono leve.

Este tipo de baterias é extremamente sensível a temperaturas altas. O calor faz com que as baterias de iões de lítio se decomponham muito mais rapidamente do que o normal. Assim a bateria selecionada, ilustrada na Figura 26, foi uma bateria de lítio de 3.7 V, com uma capacidade de 4400 mAh da GP (Farnell).

Figura 26 – Bateria de Lítio utilizada na plataforma (Farnell).

Após a decisão de utilizar baterias de lítio, foi necessário escolher o carregador e controlador responsável pela gestão de utilização de bateria. Logo o passo seguinte compreendeu a seleção de um dispositivo comercial que realizasse as funções de controlo e gestão da utilização da carga da bateria.

Figura 27 – Controlador de bateria "Lipo Rider".

Como primeira opção, foi escolhido o Lipo Rider (SeeedStudio, Lipo Rider) desenvolvido pela SeeedStudio e que se encontra representado na Figura 27. Esta solução possui um *chip* que faz a gestão da carga da bateria de lítio e na saída tem um *boost* que aumenta a tensão da bateria dos 3.7 V para os 5 V necessários para alimentar o sistema.

Este controlador permite que o sistema consuma um máximo de 400 mA da bateria, e a bateria carregue com uma corrente máximo de 300 mA. O limite de 400 mA demonstrou ser um entrave à sua utilização, uma vez que o sistema tem um consumo superior a 400 mA.

A Tabela 4 apresenta os valores mínimos, normais e máximos de tensão e corrente na utilização e carga de uma bateria recorrendo ao Lipo Rider.

Tabela 4 – Características controlador "Lipo Rider".

Característica	Mínimo	Normal	Máximo
V in	4.8 V	5.0 V	6.0 V
I in	5 mA	400 mA	-
I out	0 mA	-	300 mA
V out		5.0 V	

Para solucionar o problema da limitação de corrente, usou-se o controlador de energia Lipo Rider Pro (SeeedStudio, Lipo Rider Pro) representado na Figura 28, idêntico ao apresentado anteriormente mas mais evoluído. Este controlador possui um circuito integrado responsável pela gestão de carga e um conversor DC-DC *boost* de modo a obter na sua saída a tensão necessária. O limite de utilização de bateria é de 1000 mA e o limite de carga é de 600 mA. Além desta diferença nas características técnicas, o Lipo Rider Pro, inclui 4 LED's que possibilitam a leitura aproximada do nível de carga da bateria.

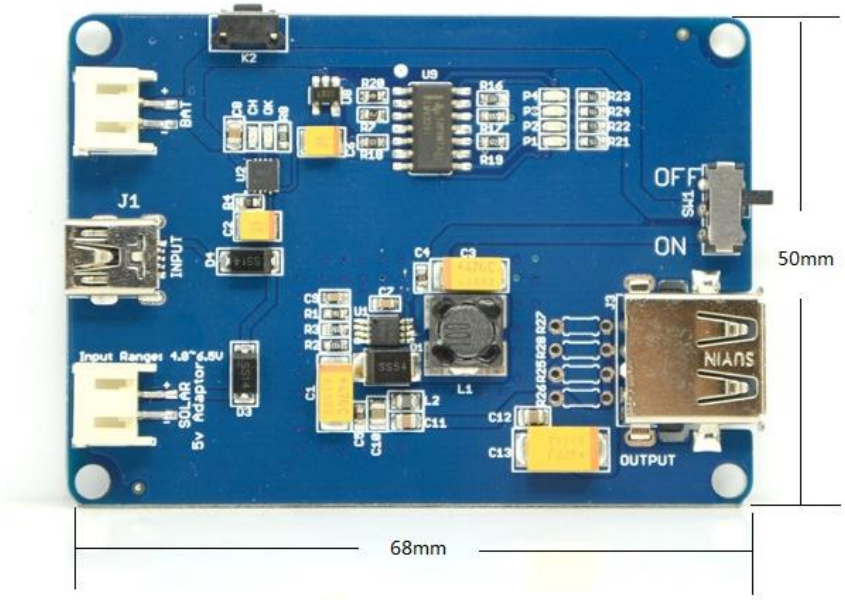

Figura 28 – Controlador de bateria "Lipo Rider Pro".

As principais características do controlador Lipo Rider Pro encontram-se descritas na Tabela 5.

Tabela 5 – Características controlador "Lipo Rider Pro".

Característica	Mínimo	Normal	Máximo
Vin	4.8 V	5.0 V	6.5V (10s)
I in(R=3.9 kΩ)	400 mA	500 mA	600 mA
I out	0 mA	-	1000 mA
V out		5.0 V	

4.1.6. Módulo de Comunicação

Uma das especificações do sistema pressupunha a comunicação entre a plataforma e um servidor. Para o referido efeito utilizou-se o módulo *Arduíno Shield Ethernet* (Arduino, 2011)(Figura 30) o qual possibilita a comunicação da Placa de desenvolvimento Arduíno à Internet. O referido *shield* utiliza o *chip* Wiznet W5100, que permite ligar a uma rede (IP), é capaz de criar ligações *Transmission Control Protocol* (TCP) e User Datagram Protocol (UDP) (Kurose & Ross, 2005) e permite até 4 ligações em simultâneo. O *shield* Ethernet comunica com o Arduíno através de *headers* com pinos compridos que atravessam o s*hield Ethernet*, isto permite uma ligação direta ao Arduíno Mega mantendo a disponibilidade de ligações a outros *shields* em camadas por cima da camada atual.

O integrado W5100 da WIZnet é um integrado de Ethernet completo, desenvolvido para aplicações embebidas, onde a facilidade de integração, estabilidade, desempenho, tamanho e controlo de custos do sistema, são as características mais importantes. Este integrado foi projetado para facilitar a implementação de conectividade com a Internet de forma a que não seja necessário um sistema operativo. Este integrado cumpre as normas "IEEE 802.3 10BASE-T" e "802.3u 100BASE-TX".

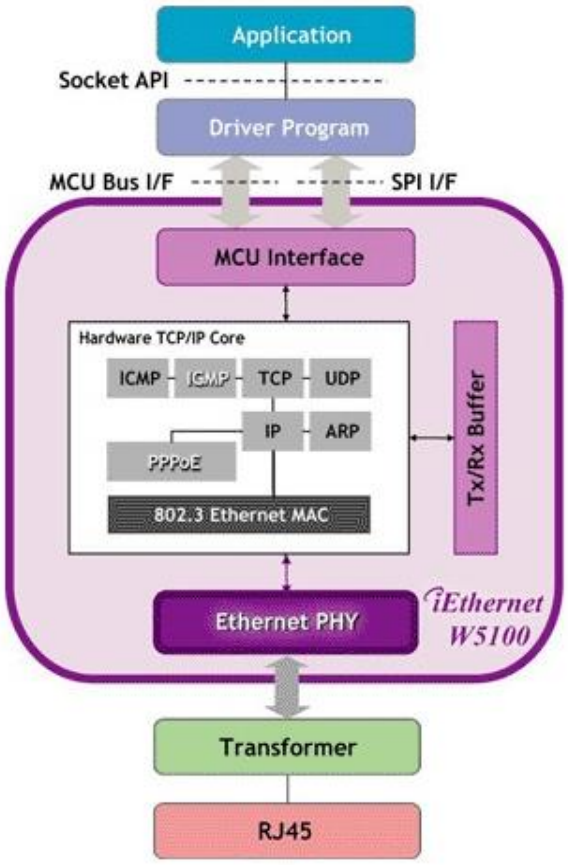

Figura 29 – Esquema de funcionamento do integrado W5100 (WizNet, 2012).

O W5100 engloba um conjunto de integrados completamente interligados de modo a que não seja necessário utilizar mais nenhum *hardware* (representado na Figura 29). As capacidades destes integrados compreendem entre outras as ligações Ethernet MAC, PHY, um buffer de 16 kBytes para a transmissão de dados e ligações TCP / IP as quais suportam TCP, UDP, IPv4, *Internet Control Message Protocol* (ICMP), *Address Resolution Protocol* (ARP), *Internet Group Management Protocol* (IGMP) e *Point-to-Point Protocol over Ethernet* (PPPoE).

Estas características fazem do W5100 uma ótima solução para o desenvolvimento de equipamentos, incluindo dispositivos de rede, serial-to-Ethernet, USB-to-Ethernet, sistemas de segurança, automação em casa, servidores incorporados e outros mais.

Para que a integração seja fácil, este integrado dispõe de diferentes *interfaces*, entre elas o protocolo SPI.

Além disso, este *shield* possui um *slot micro-SD*, que possibilita a coneção de um cartão de memória SD, cartão este que comunica com o microcontrolador através da mesma porta SPI, utilizada pelo módulo de Ethernet presente no mesmo *shield*. Este cartão pode ser usado para armazenar dados na forma de ficheiros, os quais, quando o sistema não estiver ligado à rede de comunicação, funciona como sistema de *backup*, sistema esse que pode ser implementado no cartão de memória.

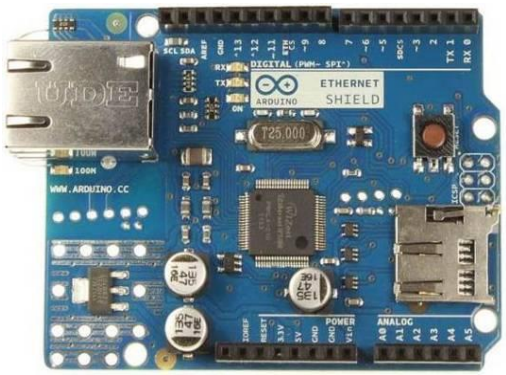

Figura 30 – Ethernet Shield para Arduíno.

4.1.7. Teclado Numérico

As plataformas portáteis necessitam que o seu operador efetue algumas operações, o que pressupõe a existência de um dispositivo de entrada de informação, para iniciar e / ou finalizar a leitura dos cartões e introduzir manualmente o número de aluno cuja *tag* está inoperacional.

Para o efeito é necessário recorrer a um teclado numérico matricial de 16 teclas, o qual está organizado numa matriz formada por quatro linhas e quatro colunas. A pressão sobre uma tecla provoca a ligação de uma linha a uma coluna.

A Figura 31 ilustra a organização interna de um teclado matricial de 16 teclas, enquanto que na Figura 32 se apresenta o teclado matricial utilizado.

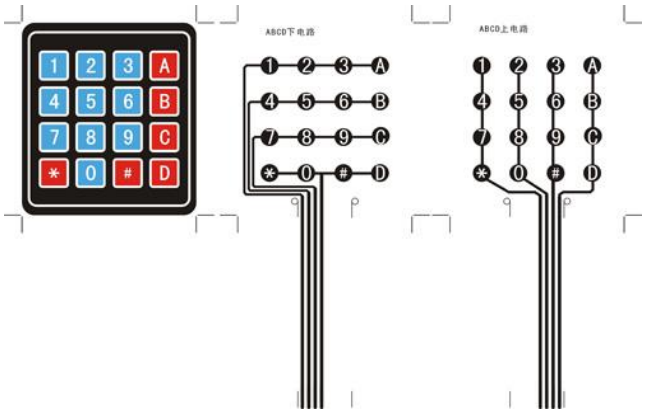

Figura 31 – Esquema genérico de ligação teclado matricial.

Figura 32 – Teclado matricial utilizado.

O princípio de funcionamento do teclado matricial, baseia-se em alimentar um "eixo" da matriz, ou seja, linhas ou colunas, e "ler" o estado do "eixo" oposto, através de varrimentos de leituras. O microcontrolador alimenta, sequencialmente, uma linha de cada vez e ao mesmo tempo verifica o estado de cada coluna. Assim, se uma tecla estiver a ser pressionada, é possível saber que se encontra na posição correspondente à intersecção da linha alimentada, com a coluna onde se lê o sinal.

Uma vez que o microcontrolador tem uma velocidade de ciclo na ordem dos 16 milhões de ciclos por segundo, é possível alternar a alimentação das linhas e verificar o estado das colunas com rapidez suficiente para detetar o toque nos botões da matriz.

4.1.8. Expansor de I/O para o Teclado Matricial

Como referido anteriormente, o teclado escolhido tem 8 ligações necessárias para o seu funcionamento, uma vez que em primeira opção foi utilizado o Arduíno Uno, não existiam I/O's suficientes para a ligação do teclado, foi desenvolvido, uma placa PCB que possibilitou o aumento de I/O's e desta forma tornou possível a ligação do microcontrolador ao teclado. O circuito é baseado no integrado PCF8574 (Instrumens) da *Texas Instruments*, a sua função é precisamente expandir I/O's remotamente através do protocolo I2C.

O referido integrado é alimentado a 5 V e apenas está ligado ao microcontrolador através de um barramento I^2C, barramento este que possui 2 ligações *Serial Clock* (SCL) e *Serial Data* (SDA). Para funcionar apenas é necessário configurar o endereço do dispositivo através dos pins A0, A1 e A2 (Figura 33).

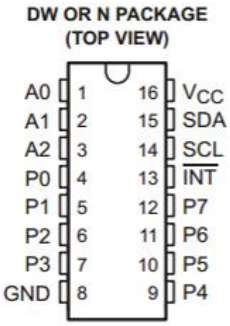

Figura 33 – Pinagem do integrado PCF8574.

O circuito desenvolvido inclui quatro conectores para que o bus I^2C do microcontrolador possa ligar também a outros dispositivos I^2C, o mesmo acontecendo com a alimentação de 5 V que liga da placa de alimentação. Também estão incluídas quatro resistências que fazem parte da ligação do teclado, uma vez que se trata de um teclado matricial, este exige a necessidade da existência de resistências de *pull-up*. Na Figura 34 é possível verificar o resultado final do projeto da placa PCB relativa ao expansor I/O com as respetivas ligações.

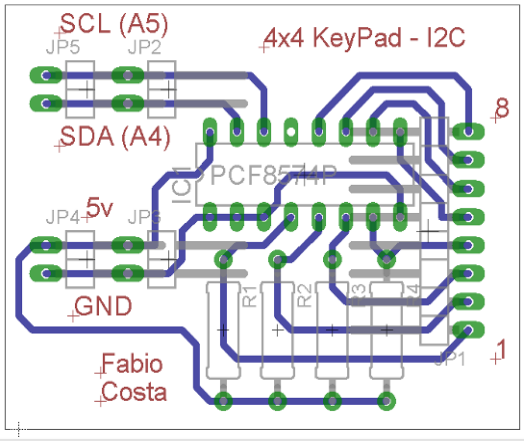

Figura 34 – Expansor de I/O I2C - desenvolvimento PCB.

Tal como enunciado antes, os 4 conectores permitem a ligação de outros dispositivos I2C, na Figura 35 é possível verificar a ligação do expansor de I/O desenvolvido, conectado a um dispositivo RTC através do bus I2C e alimentação, ficando disponível dois conectores (I2C e alimentação), para conectar estes dois dispositivos ao sistema.

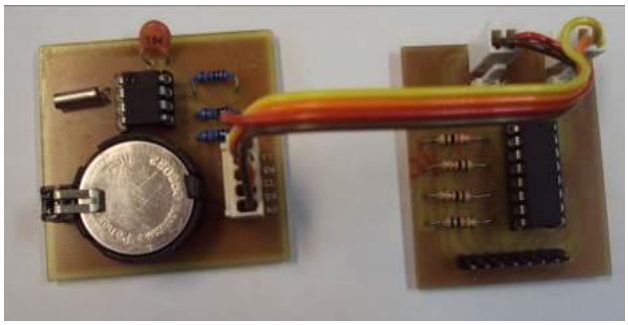

Figura 35 – Expansor de I/O I2C - PCB final.

4.1.9. Relé de Potência

Em muitas aplicações eletrónicas a energia fornecida pelo microcontrolador é suficiente para ativar os componentes conectados, uma vez que as correntes necessárias são muito pequenas. No entanto, quando é necessário interagir com dispositivos que necessitam de maior potência, é obrigatório introduzir uma *interface* entre o microcontrolador e o dispositivo periférico, dada a incapacidade dos microcontroladores fornecerem diretamente a corrente de saída pretendida.

Dada a necessidade de ativar trincos elétricos e outros dispositivos de maior potência, foi desenvolvida uma placa com um circuito de ativação de um relé que pode ser aplicado a outros dispositivos que tenham um consumo de corrente até 10 Amperes.

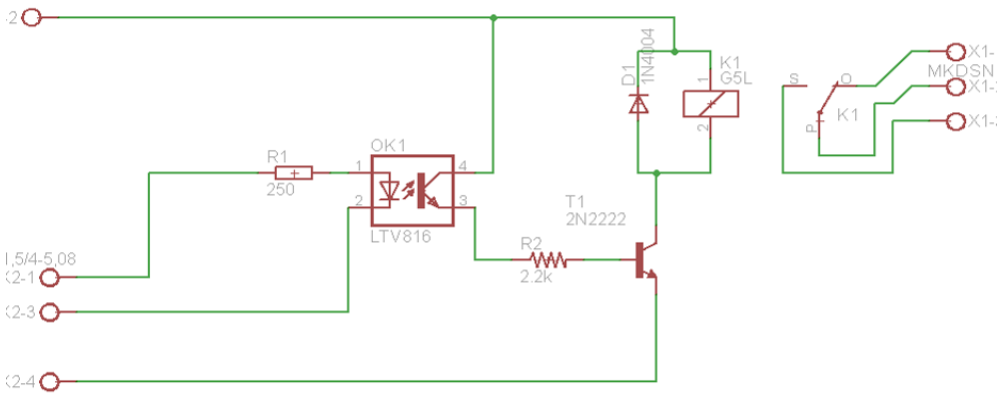

Figura 36 – Esquema elétrico de placa de potência.

O circuito implementado utiliza um foto-acoplador que separa o sinal elétrico vindo do microcontrolador, do circuito de potência, que por sua vez utiliza uma fonte de tensão diferente e alimenta o transístor que atua sobre o relé. Assim sendo, o sinal proveniente do

microcontrolador fica imune a ruídos provocados pela abertura e fecho do relé. É possível consultar o esquema elétrico na Figura 36 e o resultado da placa PCB na Figura 37.

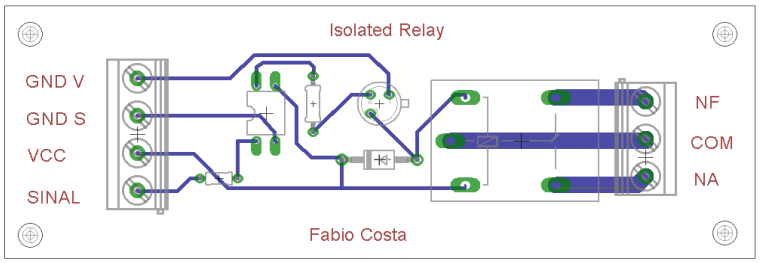

Figura 37 – PCB de placa de potência.

4.1.10. Cabo Adaptador de Programação

A placa de desenvolvimento utilizada (Arduíno Mega) dispõe de uma porta USB B, que pode ser utilizada para alimentação da própria placa, programação, e comunicação com um computador através de uma porta série virtual (que funciona através da porta USB).

Uma vez que a placa de desenvolvimento é um dos componentes que fazem parte da plataforma desenvolvida, esta ficará alojada no interior de uma caixa pelo que não existirá acesso direto à referida placa a partir do exterior. Para que esta possa ser programada e ajustada, quando se verificar necessário, foi introduzido no sistema uma ficha universal do tipo DB9, para que seja possível a programação e *debugging* do sistema (Figura 38 e Figura 39).

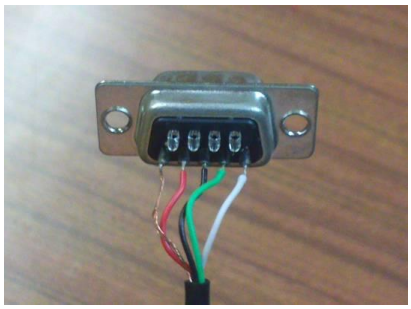

Figura 38 – Cabo adaptador de programação.

Figura 39 – Ficha de programação aplicada numa plataforma.

4.2. Funções de Programação Utilizadas

Finda a construção da plataforma a partir do equipamento selecionado anteriormente, foi necessário proceder à sua programação para que todos os módulos funcionem e interajam de acordo com o esperado.

Para a programação do microcontrolador, foi utilizado o *software* "Arduíno IDE" (Arduino, 2011). Esta plataforma foi desenvolvida em Java baseada no "Processing" e no "Wiring", o que permite a sua execução em diversos sistemas operativos. O desenvolvimento da plataforma de programação foi pensado para que as pessoas não familiarizadas com o desenvolvimento de *software* pudessem iniciar-se na sua programação. Algumas das características do *Integrated Development Environment* (IDE) são: o realce da sintaxe, parênteses com identificação automática e possibilidade de compilar e programar apenas com um botão. Assim, não é preciso editar ficheiros de programação e correr outras plataformas de programação após o desenvolvimento do código. O desenvolvimento de código pode ser feito em C/C++, o que permite criar com facilidade funções e iniciar a programação, de acordo com a aplicação pretendida.

4.2.1. Comunicação com Módulo RFID

A comunicação com o módulo RFID leitor de *tags*, tal como referido anteriormente, é feita através de uma porta série, com níveis lógicos TTL (sinais lógicos 0 V e 5 V). Para esta comunicação foram desenvolvidas funções de comunicação com o módulo. Os referidos *scripts* utilizam as funções *read* (), *write* () e *available* (), incluídas no *software* Arduino IDE, que implementa a comunicação série.

A comunicação série é caracterizada pelo envio sequencial da informação, um *bit* de cada vez (em cada sentido). Na comunicação série são utilizados dois condutores, um para envio e outro para receção de dados, comparando com a porta paralela onde vários bits são enviados ao mesmo tempo e onde cada um é enviado através de um condutor individual.

A comunicação entre o microcontrolador e o módulo RFID é feita através dos comandos definidos no *datasheet* do fabricante (13.56 MHz Mifare Read/Write Module, 2012). Nesta aplicação em concreto, apenas foi necessário analisar e desenvolver código para aplicar um comando (leitura do número de série da *tag* apresentada) e o respetivo processamento da resposta.

Uma vez que a comunicação é do tipo série, é necessário estabelecer uma organização nas mensagens transferidas. O formato da mensagem trocada entre o leitor RFID e o microcontrolador é definido pelo fabricante. Como se pode observar na Tabela 6 retirada do *datasheet* do fabricante, todas as mensagens tem início com um cabeçalho com 2 *bytes* (0xAA, 0xBB), seguido de um *byte* com a informação do comprimento total da mensagem enviada. Depois segue-se um *byte* com a identificação do comando que se pretende transmitir ao leitor RFID. Caso o comando necessite de alguma informação adicional, esta informação é adicionado de seguida. Para que seja possível verificar a integridade da mensagem enviada, é enviado como último *byte* o *cheksum* da mensagem.

Tabela 6 – Formato de mensagem de RFID.

Cabeçalho	Comprimento	Comando	Informação	CSUM
2 Byte	1 Byte	1 Byte	N Bytes	1 Byte

O comando utilizado na comunicação com o módulo de RFID, é o comando que efetua a leitura do número de série da *tag* apresentada no cartão. Caso não seja solicitado ao leitor nenhuma ação (envio de comando), este (o leitor) não reage à apresentação da *tag*. Assim, é necessário que seja enviado um comando ao leitor para que ele informe da presença e número de série da *tag*.

O comando a enviar para o leitor de RFID é o "0x20", que em conjunto com o cabeçalho, tamanho da mensagem e *cheksum*, forma a seguinte mensagem completa "0xAA 0xBB 0x20 0x22". Esta mensagem tem que ser enviada para o leitor para que se obtenha uma resposta. Caso exista cartão o comando de resposta do leitor é o mesmo "0x20", caso não exista cartão presente no leitor o comando de resposta será "0xDF". Um exemplo de uma transmissão de resposta do leitor RFID quando uma *tag* está presente será "0xAA 0xBB 0x06 0x20 0xAB 0xCD 0xEF 0xGH 0x20", onde o número de série do cartão detetado será ABCDEFGH, por outro lado caso não exista *tag* a resposta será "0xAA 0xBB 0x02 0xDF 0xDD".

Na implementação da comunicação com o módulo, foi necessário desenvolver um algoritmo para enviar o pedido, e depois receber a resposta e processá-la. Na Figura 40 é possível analisar o funcionamento do algoritmo de comunicação com o módulo RFID.

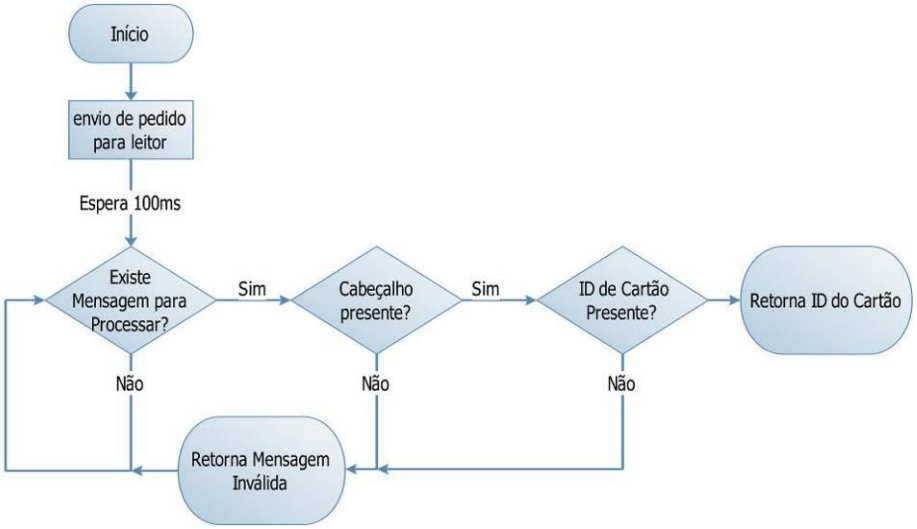

Figura 40 – Fluxograma de algoritmo de comunicação com leitor RFID.

A utilização das funções de comunicação série, pressupõe a inicialização das portas série a utilizar no início do programa. Uma vez que é possível utilizar 4 portas série, estas serão numeradas de 0 a 3 na inicialização. Na função de inicialização da porta série *begin*(*baud*,*setup*), *baud* representa a velocidade de transmissão de dados em bits por segundo e *setup* é um parâmetro opcional de comunicação dos parâmetros que define o: número de bits a enviar de cada vez, a utilização de bit de paridade e a duração do Stop bit. Por defeito a inicialização da porta série define o envio de 8 *bits*, sem bit de paridade e com um Stop bit. Esta será a definição utilizada, pois é a configuração necessária para se efetuar a comunicação com o leitor RFID, definição apresentada no *datasheet* do fabricante.

No excerto de código seguinte, é possível consultar um exemplo da iniciação de duas portas série para comunicação com leitores de RFID.

```
/*** Inicialização das Portas Serie *** /
Serial1.begin(19200);
Serial2.begin(19200);
```

As funções mais utilizadas para a implementação do fluxograma acima apresentado, foram *serial.write* () e *serial.read* (). A função *write* () é usada para enviar *bytes* do microcontrolador para o leitor RFID. A função *read* (), efetua a leitura *byte* a *byte* a informação recebida pelo leitor.

```
/*** Script de Pedido de Leitura ao leitor RFID *** /
char comandoProcuraRF[] = { 0xAA, 0xBB,0x02, 0x20, 0x22};
Serial.write(comandoProcuraRF, 5);
```

Para o processamento das mensagens trocadas com o leitor de RFID foi desenvolvido um *script*. Nesse script, a seguir representado, enquanto existe mensagem no buffer de entrada da porta série, o ciclo *while* mantém-se em repetição, dentro deste ciclo a cada repetição é lido um *byte*, de seguida através de uma "máquina de estados" é verificada a estrutura da mensagem recebida. A "máquina de estados" tem como função ler e verificar a estrutura e conteúdo da mensagem. Se o primeiro *byte* recebido, tal como descrito no *datasheet*, for "0xAA", a "máquina de estados" avança para o próximo estado, verificando então o segundo *byte* que deverá ser "0xBB", caso contrário a "máquina de estados" volta ao início onde permanece até verificar a existência do primeiro *byte* de uma transmissão "0xAA". Se a mensagem recebida não verifica a estrutura esperada, significa que não foi recebida uma resposta pelo leitor de RFID com um cartão presente na mensagem anterior. Quando a máquina de estados verifica todos os bytes da mensagem e estes estão de acordo com o esperado, o ID da *tag* é obtido e assim é realizada a identificação de uma *tag* presente no leitor RFID.

Para que o microcontrolador identifique uma passagem de uma *tag*, é necessário invocar o *script* de leitura do leitor RFID com uma frequência suficientemente elevada para que não possa ocorrer a presença de uma *tag* sem que esta seja detetada.

```
/*** Excerto de Processamento de Resposta do Leitor RFID *** /
Int estado = 0;
while (Serial.available() > 0) {
      val = Serial1.read();
      switch (estado) {
      case 0:  // verifica 0xAA
        if (val == 0xAA) estado = 1;
        break;
      case 1: // verifica 0xBB
        if (val == 0xBB) estado = 2;
        else return -1;
        break;
      case 2: // verifica comprimento
        if (val == 0x06) estado = 3;
        else return -1;
        break;
      (...)
}
```

Numa comunicação série, os dados são usualmente enviados em pequenos grupos de 10 ou 11 bits, dos quais 8 constituem a mensagem propriamente dita. Quando o canal está em repouso, o sinal eléctrico presente tem um nível lógico '1'. O pacote de dados começa sempre com o nível lógico '0' *start bit* para indicar o início da transmissão. Após o envio do *start bit*, seguem-se 8 bits de dados de mensagem os quais são enviados a uma taxa de transmissão previamente especificada. A transmissão é concluída com o envio de um stop bit.

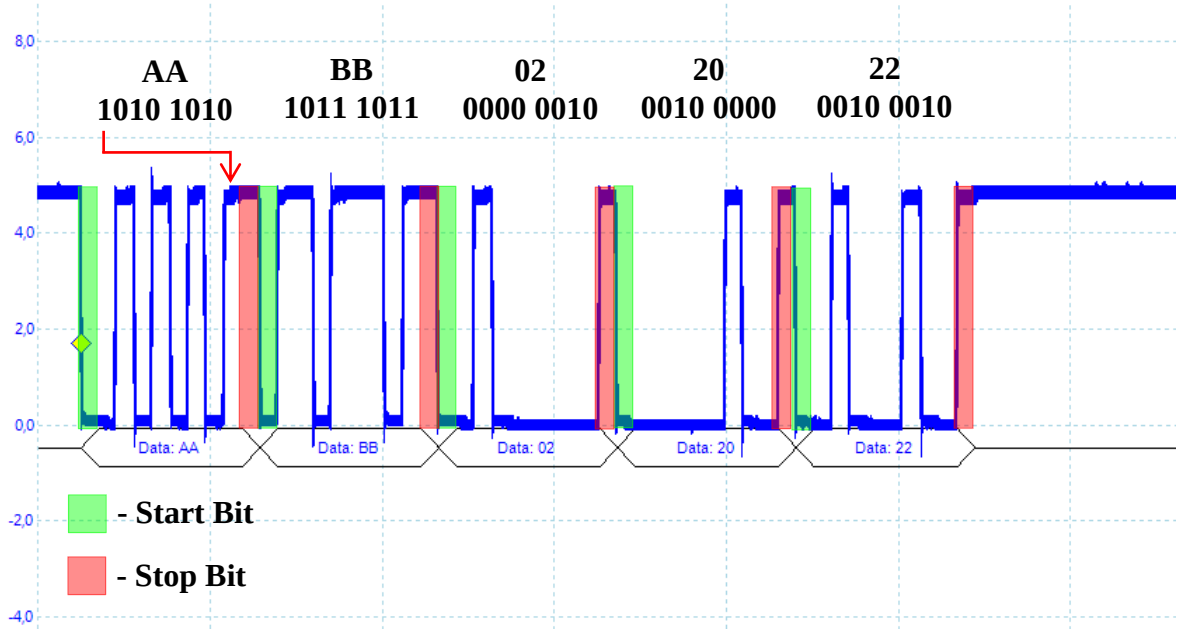

Figura 41 – Mensagem enviada pelo microcontrolador ao leitor RFID.

Na Figura 41, pode-se observar a mensagem enviada pelo microcontrolador ao leitor RFID. Como resposta do leitor, obtêm-se a mensagem representada na Figura 42 quando não existe qualquer *tag* presente ou a resposta representada na Figura 43 correspondente à sequência de bits de informação enviada pelo leitor quando existe uma *tag* presente.

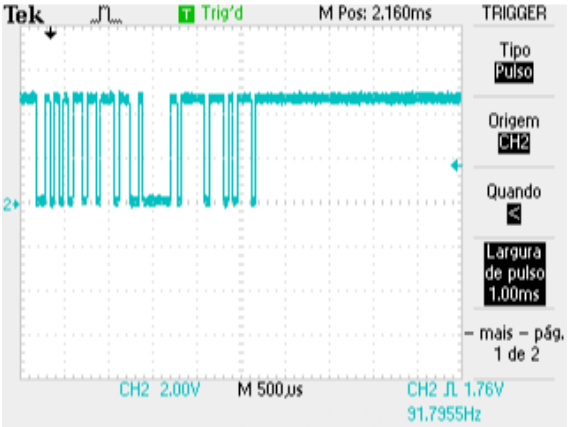

Figura 42 – Mensagem de resposta do Leitor RFID quando não existe *tag* presente.

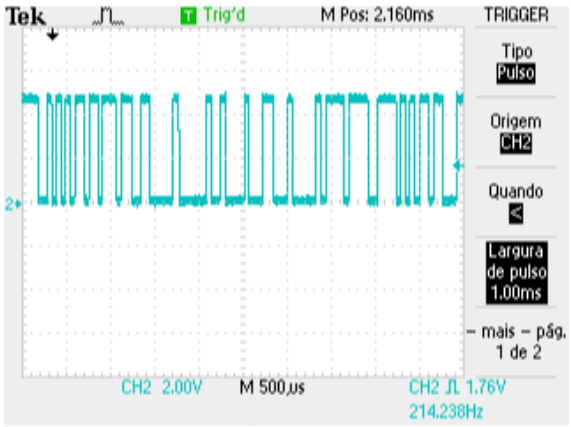

Figura 43 – Mensagem de resposta do Leitor RFID com existência de *tag* presente.

4.2.2. Módulo de Tempo Real (RTC)

A comunicação entre o módulo de RTC (Maxim DS1307) e o microcontrolador, é realizada usando o protocolo I^2C que específica dois sinais de comunicação, um sinal de *clock* e outro de dados bidirecional. Para a gestão dos sinais trocados entre eles, é necessária uma biblioteca específica, sendo que para o efeito foi escolhida a biblioteca DS1307new. Esta biblioteca, além de fazer toda a gestão da comunicação I^2C, permite a consulta e o acerto da hora, e permite também o acesso à memória interna do integrado Maxim DS1307, o mesmo que processa o tempo real no módulo RTC.

No desenvolvimento do *software* da plataforma de gestão e controlo de acessos, foi necessário utilizar funções incluídas na referida biblioteca. Um dos conjuntos de funções permite a consulta da data e da hora, o outro conjunto de funções é usado para atualizar a hora caso seja necessário.

Para a consulta da data e da hora foi usada a função *getTime* (), a qual utiliza a informação relativa à data e hora contida no módulo RTC e a coloca em variáveis apropriadas definidas na biblioteca, as quais podem ser facilmente consultadas posteriormente.

```
/************ Funcao Mostra Hora ********************/
void mostraHora () {
  RTC.getTime ();
  if (RTC.hour < 10) {
    lcd.print ("0");
    lcd.print (RTC.hour);
  }
  else {
    lcd.print (RTC.hour);
  }
  lcd.print (":");
…
```

Como é possível ver no exemplo anterior (excerto da função que mostra a hora no ecrã LCD), inicialmente é invocada a função *getTime* (), para que esta consulte a informação do módulo RTC. Tendo em conta que a representação da hora é realizada sempre com dois dígitos (para as horas) seguida de dois pontos e dois dígitos (para os minutos), é necessário verificar se o valor da hora (RTC.hour) a representar é maior ou menor que 10, adicionando-se um zero à esquerda caso o valor da hora seja inferior a 10.

Este procedimento foi também aplicado à representação dos minutos (RTC.minute), dos segundos (RTC.second), e dos dias do mês (RTC.day) no ecrã LCD. Para representar os meses (RTC.month), foi implementada uma estrutura *switch-case* que traduz o número

representativo da ordem do mês no seu nome descrito por extenso. Este procedimento é usado de uma forma semelhante para a representação dos dias da semana (RTC.dow), uma vez que o módulo RTC informa também qual é o dia da semana em formato numérico.

Para realizar a atualização da hora e da data no módulo RTC é necessário utilizar outro conjunto de funções. A sequência de acerto da data e da hora consiste em parar o relógio, preencher os novos dados de data e hora, seguido da ordem de acerto de hora e por fim, reiniciar o relógio como se pode ver na função acertar a hora que a seguir se representa.

```
/************ Funcao Acertar Hora ********************/
void acertarRTC (int hora, int mi, int seg, int ano, int mes, int dia){
  RTC.stopClock ();
  RTC.fillByHMS (hora, mi, seg);
  RTC.fillByYMD (ano, mes, dia);
  RTC.setTime ();
  RTC.startClock ();
  delay (200);
}
```

4.2.3. Ecrã Alfanumérico

A comunicação entre o ecrã e o microcontrolador é do tipo "paralela", tendo em conta que vários bits são transferidos simultaneamente. Esta comunicação envolve *bytes*, cada um com 8 bits. A forma mais simples de conexão é usar um sinal para cada bit, correspondendo aos pinos DB0 a DB7 do *display*. Para utilizar menos pinos de entradas e saídas do microcontrolador, o LCD possui um modo em que cada transferência de *byte* é dividida em duas transferências de 4 bits (*nibble*). Neste caso somente os pinos DB4 a DB7 são usados para a transferência de dados.

Para que seja possível a comunicação entre o microcontrolador e o LCD, é utilizada a biblioteca LiquidCrystal.h. Esta biblioteca permite que o microcontrolador controle ecrãs de cristais líquidos (LCD) baseados num *chipset* da Hitachi HD44780 (ou compatível), o qual é encontrado na maioria dos LCD's alfanuméricos.

Esta biblioteca permite após a inicialização do LCD, que com apenas três comandos simples seja possível utilizar o ecrã e representar o texto pretendido.

As funções de inicialização *begin* () e *LiquidCrystal* () são responsáveis respetivamente pela definição das dimensões (largura e altura) do *display* utilizado e pela definição dos pinos I/O usados na placa de desenvolvimento para a comunicação paralela com o LCD. Estas são as primeiras funções a ser invocadas como se descreve de seguida.

```
/******** CONFIGURACAO DO LCD *************

LiquidCrystal lcd (26, 28, 32, 34, 36, 38);
lcd.begin (20,4);
```

O possível posicionamento do cursor no ecrã, é realizado usando a função *setCursor* (*y*, *x*), onde *y* representa posição vertical e *x* a posição horizontal. A instrução mais importante da biblioteca é o *print* (*data*), esta função imprime no LCD a informação contida na variável *data* do tipo *string* a partir da posição atual do cursor no LCD.

Para limpar a informação presente no ecrã de uma só vez, é invocada a função *clear* () que limpa totalmente o ecrã. A sua utilização simplifica o trabalho do programador que teria, caso esta função não existisse, de inserir espaços em branco em todas as posições do LCD, para que este não expusesse qualquer caractere.

Para suportar a representação do logótipo do ROBOCORP (Figura 44) foi necessário definir caracteres especiais. Nos ecrãs LCD´s de caracteres alfanuméricos é possível definir até oito caracteres especiais, dependendo do LCD em uso.

O primeiro passo compreendeu a definição dos pixéis a ligar e desligar para cada caractere especial. Para o efeito foi definido um conjunto de *arrays* de bytes rc[] um para cada caractere especial. Em cada byte do *array*, está definida uma linha de pontos na qual o valor "0" significa que o pixel estará desligado e o valor "1" significa que o pixel estará ligado.

Para auxiliar a definição dos diferentes carateres e pixéis do símbolo da RoboCorp foi utilizada numa folha quadriculada, representada na Figura 45, onde é possível pré-visualizar o resultado dos caracteres especiais, e facilitar a definição dos diferentes bytes associados a cada um dos caracteres.

Figura 44 – Logotipo da RoboCorp.

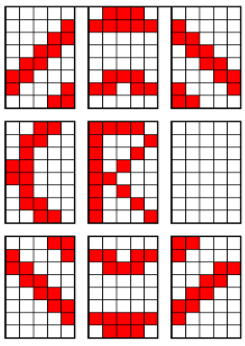

Figura 45 – Simulação do logotipo.

Após a definição dos caracteres especiais, é necessário registar esta informação no microcontrolador. Como se representa no exemplo seguinte, primeiro são definidos os *arrays* de caracteres especiais, de seguida estes são registados no LCD, usando a função *createChar* (*num*, *data*), onde a variável *num* representa o número do caractere especial de 0 a 7 e variável *data* é o *array* de bytes que definem o caractere.

```
/************ ROBOCORP Symbol ******************/
byte rc1[8] = { B00000,B00000,B00011,B00111,B01111,B11110,B11100,B11100};
byte rc2[8] = { B00000,B00000,B11111,B11111,B11111,B00000,B00000,B00000};
byte rc3[8] = { B00000,B00000,B11000,B11100,B11000,B00000,B00000,B00000};
byte rc4[8] = { B11100,B11100,B11100,B11100,B11100,B11100,B11100,B11100};
byte rc5[8] = { B11110,B11011,B11010,B11100,B11110,B11011,B11011,B11011};
byte rc6[8] = { B11100,B11110,B01111,B00111,B00011,B00000,B00000,B00000};
byte rc7[8] = { B11110,B11001,B11010,B11100,B11010,B11001,B11001,B11001};

lcd.createChar (1, rc1);
lcd.createChar (2, rc2);
lcd.createChar (3, rc3);
lcd.createChar (4, rc4);
lcd.createChar (5, rc5);
lcd.createChar (6, rc6);
lcd.createChar (0, rc7);

delay (500);
```

Para utilizar os caracteres especiais foi necessário recorrer à função *write* (*num*), onde a variável *num* indica a numeração do caractere especial, assim é possível que este seja escrito no ecrã, como no exemplo seguinte.

```
/****** Codigo para imprimir logo ******/
lcd.setCursor (2, 0);
lcd.write (01);
lcd.write (02);
lcd.write (03);

lcd.setCursor (2, 1);
lcd.write (04);
lcd.print ("R");

lcd.setCursor (2, 2);
lcd.write (06);
lcd.write (02);
lcd.write (03);
```

4.2.4. Utilização de Cartão SD

O *shield Ethernet* para além do conector RJ45 e do controlador Ethernet possui um leitor de cartões de memória SD que permite a gravação de dados em ficheiros. A comunicação com o cartão de memória é realizada (tal como para o módulo de Ethernet), através do protocolo SPI.

Para utilizar o cartão de memória SD é necessário que exista sincronismo com a porta SPI, sendo também necessárias as funções específicas para criar, editar e apagar ficheiros no cartão SD. Estas funções são asseguradas pela biblioteca SD.h, a qual deve ser incluída no programa usando o seguinte código "#include <SD.h>".

O procedimento normal para aceder a um ficheiro num cartão de memória consiste numa inicialização única através da função *begin* (*pin*), em que *pin* é o número do pino onde está ligado o sinal de *enable* do leitor de cartões na comunicação SPI. Este requisito deriva do facto de na comunicação SPI para cada dispositivo ligado por este protocolo, ter de existir um sinal de *enable*, para que cada uma das comunicações se efetue corretamente.

Depois da sua utilização deve-se fechar o ficheiro, para que não ocorram erros na sua formatação e seja possível utilizar outros ficheiros diferentes. Para a abertura de um ficheiro é necessário a declaração de uma variável do tipo ficheiro e abrir um ficheiro através da função *open* (). Após essa atribuição a função *open* abre o ficheiro caso este exista, caso este não exista a função gera automaticamente o ficheiro pretendido. A partir daí será possível trabalhar no mesmo, como mostra o exemplo seguinte. Este demostra que à variável *ficheiro* é atribuído o nome "teste" com extensão "txt" através da função *open* (), de seguida é verificado se o ficheiro "teste" existe, sendo depois inserida a informação contida na variável *data*, para terminar o processo é encerrado o ficheiro.

```
/****** Exemplo de Escrita em Cartao de Memoria ******/

File ficheiro = SD.open("caminho/teste.txt");
if (ficheiro) {
   SD.println(data);
}
SD.close();
```

Seguindo o procedimento de escrita acima descrito, foi desenvolvido um *script* de escrita no cartão de memória que segue um formato de escrita, referido à frente para que exista uma estrutura coerente. A Figura 46 apresenta o fluxograma de escrita no cartão de memória.

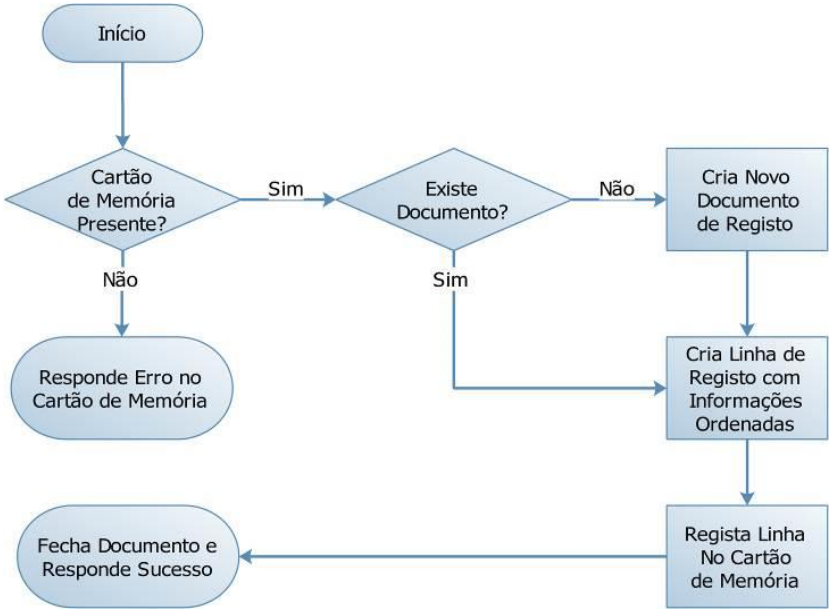

Figura 46 – Fluxograma de registo em memória.

Os registos realizados no cartão de memória devem estar organizados de acordo com uma estrutura pré-definidas de forma a que seja possível a quando da leitura, verificar se a integridade dos dados está preservada. Assim os registos feitos no cartão de memória devem ter a seguinte estrutura.

```
/*****Estrutura dos registos efetuados*****/
#12345678-HHMMSS-AAAAMMDD*
```

Onde "12345678" representa o ID da *tag* a registar, seguido da hora e da data separadas por "-". Para representar o início da linha um utiliza-se "#" e no fim da linha usa-se um "*".

No processamento da informação contida no cartão de memória, são utilizados ciclos de repetição, lendo cada um dos registos nela contido caractere a caractere. Em simultâneo verifica-se a integridade da estrutura dos registos efetuados e atribuem-se às variáveis os valores registados em cada linha. Após cada linha ter sido processada, a informação é enviada para o servidor, e caso esta seja registada na base de dados com sucesso, a linha é eliminada. O procedimento descrito encontra-se representado no fluxograma presente na Figura 47.

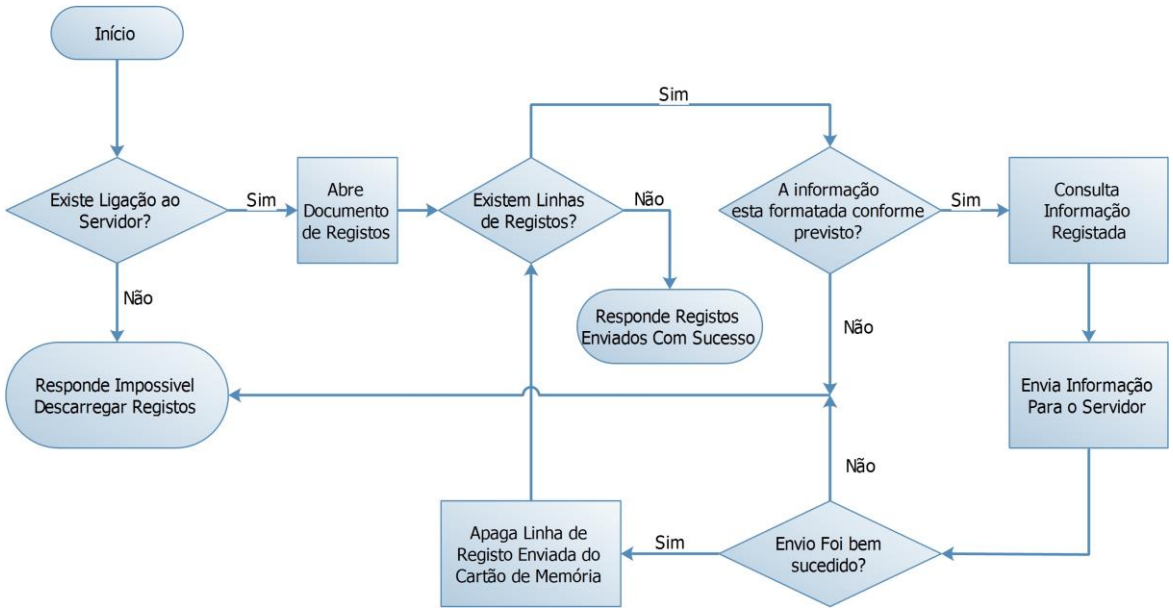

Figura 47 – Fluxograma de processamento de registos feitos no cartão de memória.

4.2.5. Comunicação com Servidor

Para que seja possível a transmissão de dados entre o microcontrolador e o servidor, além do *shield Ethernet*, referido anteriormente, é necessário utilizar funções de *software* para controlar e gerir todas as comunicações internas, entre microcontrolador e *shield Ethernet* e a comunicação externa, entre *shield Ethernet* e servidor, o qual usa protocolo *Ethernet* IEEE 802.3.

A gestão, é efetuada usando a biblioteca Ethernet.h. Esta biblioteca tem um conjunto de funções que permite a comunicação com outro dispositivo via *Ethernet*, possibilitando comunicações cliente-servidor, através dos protocolos de transporte TCP ou UDP. Internamente são criadas ligações SPI necessárias para a comunicação entre o *shield Ethernet* e o microcontrolador.

A biblioteca tem capacidade para adquirir um endereço IP automaticamente da rede, através de *Dynamic Host Configuration Protocol* (DHCP) caso esta assim esteja configurada. Caso se utilize o serviço DHCP (no *router* ou *switch*) não é necessário configurar o IP de rede manualmente pois este será atribuído automaticamente. Outro serviço que é possível utilizar com recurso às funções presentes na biblioteca é o serviço de *Domain Name System* (DNS), para que seja possível utilizar o "nome" (endereço *web*) do servidor de destino e assim, este ser resolvido pelo serviço DNS diretamente em vez de configurar o IP de destino.

A utilização das funções associadas à transmissão de dados sobre Ethernet pressupõe a inclusão das seguintes bibliotecas no programa principal.

```
/****** Declaração de Bibliotecas Necessarias ******/

#include <Dhcp.h>
#include <Dns.h>
#include <Ethernet.h>
#include <EthernetClient.h>
#include <EthernetServer.h>
#include <EthernetUdp.h>
```

No exemplo seguinte é possível verificar que para além da biblioteca principal Ethernet.h, são também incluídas outras bibliotecas que poderão ser necessárias para a implementação de ligações como os serviços DHCP e DNS e também EthernetClient e EthernetServer para que seja possível criar ligações como Servidor ou Cliente.

Além da referência das bibliotecas é necessário realizar a configuração das variáveis, onde serão armazenados os parâmetros de rede e ligação, de entre eles refira-se:

- Endereço *Media Access Control* (MAC), que é um endereço único associado a cada *hardware*;
- Endereço de servidor (poderá ser um domínio ou um Endereço IP);
- Endereço IP do módulo Ethernet (caso não seja utilizado o serviço de DHCP);
- *Gateway*, que é o IP do equipamento que liga a rede onde esta o módulo de *Ethernet* e o exterior;
- Máscara de rede que é um número que permite a rede fazer a separação da rede interior de uma rede exterior;
- Endereço IP de DNS (caso seja necessário utilizar o serviço DNS, o módulo irá ligar ao servidor DNS definido para fazer a resolução de domínios).

De seguida é possível ver um exemplo de declaração das variáveis necessárias para a inicialização do módulo *Ethernet*.

```
/************ CONFIGURACAO ETHERNET ***************/

byte mac[] = {0x90, 0xAA, 0xAA, 0x00, 0x1F, 0xAD };
//char server[] = "cda.electrocosta.pt";
IPAddress server(94,126,173,183);
IPAddress ip(192,168,100, 62);
IPAddress gateway(192,168,100,254);
IPAddress mydns(192,168,120,20);
IPAddress subnet(255, 255, 255, 0);
```

Para a utilização desde módulo é ainda necessário criar um objeto a partir do qual serão dadas as instruções de comunicação. A criação de um objeto, é idêntica à declaração de uma variável, primeiro é definido o tipo do objeto e a seguir o nome do objeto. Assim, quando o objeto cliente for invocado os comandos irão fazer com que o módulo de Ethernet reaja como um cliente numa rede *Ethernet*. No exemplo seguinte apresenta-se a declaração de um objeto *Ethernet*.

```
/** Declaracao de Objecto Cliente Ethernet **/

EthernetClient client;
```

A inicialização do módulo de *Ethernet* pode ser feita de diversas formas sempre através da função *begin* () mas com diferentes parâmetros relativos ao endereço MAC do equipamento, IP da rede, do servidor DNS e da *gateway*.

Quando é utilizado o serviço DHCP é apenas necessário referir o endereço MAC. No caso de se usar a configuração manual é necessário referir os restantes parâmetros. O código seguinte apresenta um exemplo de uma função que faz uma tentativa de ligação por DHCP, se esta tentativa falhar utiliza os parâmetros previamente definidos. Desta forma existe uma segunda possibilidade de configuração caso a atribuição automática de IP falhe.

```
/****** Inicialização de Módulo Ethernet ******/

if (Ethernet.begin(mac) == 0) {
   Ethernet.begin(mac, ip, mydns,gateway,subnet);
}
```

Depois do módulo estar inicializado, é possível comunicar com outro equipamento através de comandos usando o objeto anteriormente criado. Os comandos utilizados no desenvolvimento de um programa são os seguintes: *connect* (), *print* (), *available* (), *read* (), *flush* (), *find* () e *stop* ().

Para comunicar com o servidor, primeiro é necessário ligar ao servidor através da função *connect* (*serv*, 80), onde a variável *serv* é o endereço do servidor (IP ou domínio). Para além de estabelecer uma ligação, a função devolve informação relativa ao sucesso do estabelecimento da ligação. Caso a ligação tenha ocorrido com sucesso é enviado um comando de protocolo HTTP usando a função *println* ().

Após o envio do comando é aguardada uma resposta do servidor através da função *available* (), para que depois seja possível chamar a função *read* () e assim ler e processar a resposta enviada pelo servidor. Depois da leitura da resposta é verificado se existe mais alguma mensagem a ler do servidor, caso não exista mais nenhuma a ligação é fechada através da função *stop* ().

O excerto seguinte mostra uma função simples onde é feita uma ligação ao servidor e é esperada uma resposta específica, para verificar se o servidor se encontra preparado e ativo.

```
/** Verificacao de Servidor **/

int verificaServidor(){

  Ethernet.begin(mac, ip, mydns,gateway,subnet);
  if (client.connect(server, 80)) {
    client.println("GET /arduino_test.php?pass=010101");
  }
  else {
    Serial.println("Falhou a Ligacao");
  }
  if(client.find("80") == true){
    client.flush();
  }
  client.stop();

}
```

Para realizar as tarefas necessárias na plataforma *web* foram desenvolvidos *scripts*, com base nas funções referidas. A título de exemplo, refira-se, a verificação da ligação ao servidor, o registo de utilizador na base de dados (presença ou registo de ponto) e a verificação/registo de gestão de acessos.

A verificação de ligação ao servidor consiste numa tentativa de conexão ao servidor a solicitar para que este lhe responda uma mensagem pré-definida. O fluxograma apresentado na Figura 48 correspondente ao funcionamento deste *script* de verificação de ligação ao servidor.

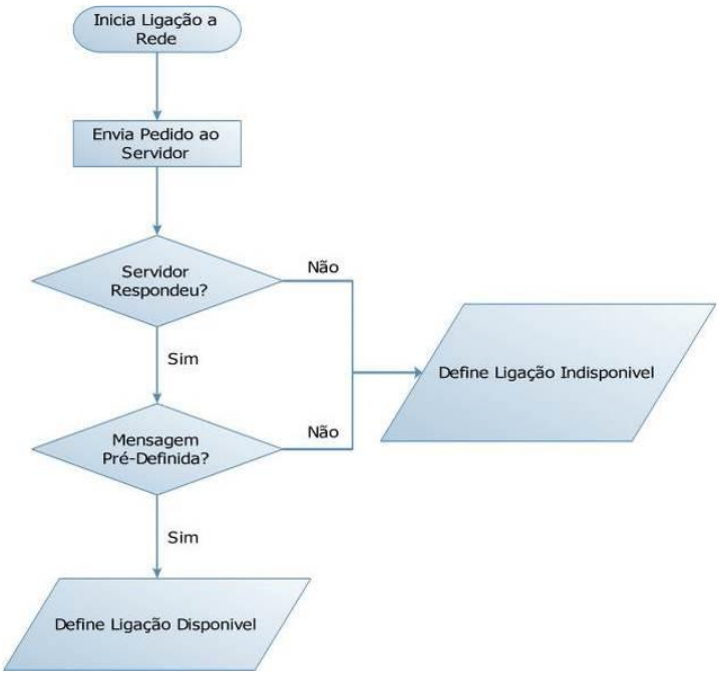

Figura 48 – Fluxograma de função de verificação de ligação ao servidor.

A função responsável pelo envio de informação para o servidor primeiro verifica se o servidor está disponível. De seguida envia a informação relativa à identificação de uma dada *tag* e do leitor que efetuou a leitura e consulta às autorizações de cada utilizador. Caso o utilizador anteriormente identificado tenha privilégio de acesso autorizado à zona pretendida, o microcontrolador atua o relé associado ao trinco da porta para dar acesso físico ao utilizador a um dado espaço. Durante esse processo, o servidor regista a assiduidade e/ou presença consoante a aplicação da plataforma em causa. Na Figura 49 é possível verificar o fluxograma de funcionamento do *script* de envio ao servidor.

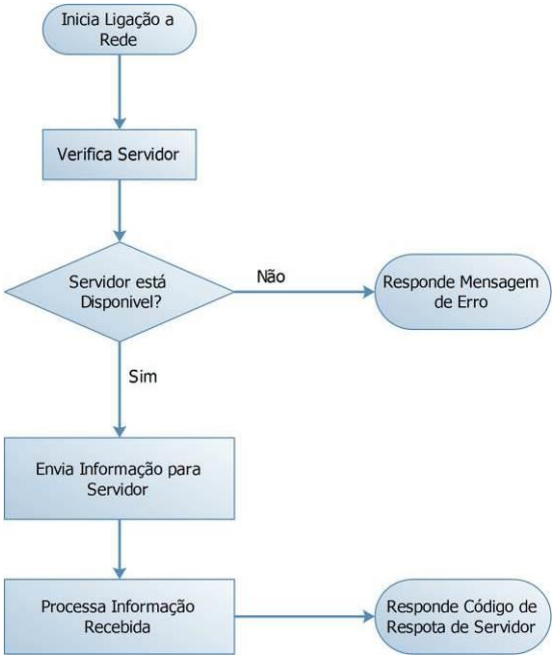

Figura 49 – Fluxograma de função de envio de informação para servidor.

4.2.6. MD5 (Message-Digest algorithm 5)

Muitas vezes a integridade e segurança das comunicações é posta em causa, utilizando programas como o *wireshark*, os quais permitem consultar os dados que entram e saem de um equipamento numa rede específico. Assim, como medida de segurança, foi implementado nas comunicações *Ethernet* a codificação MD5 para que, em caso de alguma comunicação ser intercetada na rede, as informações importantes estarem codificadas.

A escolha da codificação MD5, deve-se à existência de bibliotecas específicas para o efeito, o que permite diminuir o tempo despendido a desenvolver uma técnica de codificação, e também devido à facilidade da sua utilização no servidor, uma vez que já, existem funções diretas em PHP que convertem MD5.

Em virtude, deste método ser tão conhecido, poderia haver o perigo da informação enviada ser descodificada, perigo esse que é muito reduzido uma vez que o algoritmo MD5 é unidirecional, isto é, é possível codificar, mas não é possível descodificar. Para descodificar uma mensagem enviada com MD5, é necessário que o servidor tenha pré-registado a informação codificada. Assim tem que existir uma base de dados com a informação relativa aos dados enviados, para que depois estes sejam codificados no servidor. A identificação dos dados enviados é realizada por comparação entre os dados recebidos e os dados codificados existentes.

O resultado da codificação MD5 é sempre uma *string* alfanumérica com 32 caracteres, independentemente do número de caracteres da informação codificada. Por exemplo o resultado da codificação de um único espaço " " é o seguinte: "7215ee9c7d9dc 229d2921a40e899ec5f" enquanto que da frase "texto de exemplo" é "423108ca42bf6 28147920b264b4707ef".

Para aumentar a segurança deste método, e uma vez que existem bases de dados na internet com códigos e mensagens registadas na tentativa de descodificar o maior número de combinações possíveis, a cada mensagem a codificar, é adicionado um código que modificará por completo o resultado codificado. A título de exemplo, como anteriormente referido o resultado da codificação MD5 de "texto de exemplo" é "423108ca42bf6281 47920b264b4707ef" se juntarmos "A" no fim do texto a codificar o resultado é "689d351e4ef2bf18f272c9275c8a32c9", modificando completamente o resultado e assim o nível de segurança é maior.

A implementação deste sistema de codificação é feita através da utilização das funções presentes na biblioteca. Para codificar informação apenas é necessário invocar a função do_md5(*string*), onde a variável *string* é a variável a codificar.

O resultado da codificação é obtido a partir da própria função e pode ser representado num LCD como se observa no código seguinte.

```
/*** Exemplo de Script para Codificar Mensagens ***/

String aCodificar;
String resultado;

aCodificar = "texto de exemplo";

resultado=do_md5(aCodificar);

lcd.print(resultado);
```

4.2.7. Implementação de *Watch-Dog*

Para aumentar a fiabilidade do sistema é utilizado um mecanismo de *WatchDog*. Este pode ser considerado um mecanismo de emergência, uma vez que a sua função, é reiniciar todo o sistema quando este deixa de responder.

Os microcontroladores ATMega têm um temporizador *Watch Dog Timer* (WDT) integrado. O WDT é um temporizador que conta ciclos de um oscilador de 128 kHz incluído

no integrado. O WDT provoca uma interrupção ou um *reset* forçado ao sistema quando o contador atinge um valor pré-definido. O funcionamento normal de um sistema que usa WDT consiste na invocação de um comando específico, antes que o contador atinja o valor pré-definido. Se o sistema não reiniciar o contador, irá acontecer a interrupção forçada do sistema.

Tabela 7 – Intervalos de tempo de WatchDog.

Intervalo de Tempo	Nome da Constante	Suportado pelos integrados:
15 ms	WDTO_15MS	ATMega 8, 168, 328, 1280, 2560
30 ms	WDTO_30MS	ATMega 8, 168, 328, 1280, 2560
60 ms	WDTO_60MS	ATMega 8, 168, 328, 1280, 2560
120 ms	WDTO_120MS	ATMega 8, 168, 328, 1280, 2560
250 ms	WDTO_250MS	ATMega 8, 168, 328, 1280, 2560
500 ms	WDTO_500MS	ATMega 8, 168, 328, 1280, 2560
1 s	WDTO_1S	ATMega 8, 168, 328, 1280, 2560
2 s	WDTO_2S	ATMega 8, 168, 328, 1280, 2560
4 s	WDTO_4S	ATMega 168, 328, 1280, 2560
8 s	WDTO_8S	ATMega 168, 328, 1280, 2560

Os valores do contador são definidos ao inicializar do WDT e encontram-se restringidos aos valores definidos pelo fabricante (Tabela 7).

A função *wdt_enable* (*var*) estabelece o valor do *WatchDog*, sendo que o seu argumento é o nome da constante de tempo definida pelo fabricante.

O tempo a definir no *WatchDog*, depende do tempo máximo que uma função crítica pode demorar a executar. No exemplo, seguinte a função demora 200 ms a executar, então o *WatchDog* deve ter um valor superior a 200 ms, para que o microcontrolador não reinicie inadvertidamente. Se por algum motivo a função bloquear, após o tempo definido, o WDT irá reiniciar o sistema e assim voltar a funcionar, não sendo necessário a intervenção do utilizador.

```
/****** Exemplo de Utilização de WatchDog ******/

#include <avr/wdt.h>  // Referencia da biblioteca necessaria para uso de
WDT

void setup (){
  wdt_enable (WDTO_250MS);  //Inicializacao do WatchDog
}

void loop ()
{
  funcao();
  wdt_reset (); // recomeco do contador de WDT
}
```

4.3. Protótipos Desenvolvidos

No decurso do trabalho foram desenvolvidos diversos protótipos, com o objetivo de validar a solução tecnológica pretendida e que serviram de plataforma de desenvolvimento de novas funcionalidades.

Numa primeira fase foi criada uma plataforma fixa, cujo objetivo, compreendia o controlo e gestão de acessos a laboratórios de investigação do ISEC, escolhidos para o efeito. Para garantir a fiabilidade do sistema final, foi implementado um protótipo de testes o qual foi instalado na entrada do laboratório da RoboCorp.

Para esta plataforma, foi concebido um programa que permite a leitura de múltiplas *tags*, a representação de informação num ecrã LCD e o estabelecimento de comunicações cliente-servidor suportadas pela rede Ethernet existente. Este protótipo serviu também de base à realização de diferentes testes que permitiram a identificação de erros de *hardware* e *software* existentes, os quais foram posteriormente eliminados.

Figura 50 – Sistema de gestão de acessos no interior do RoboCorp.

Figura 51 – Sistema de gestão de acessos no exterior do RoboCorp.

Refira-se que este primeiro protótipo, representado na Figura 50 e Figura 51, inclui diversos módulos, de entre eles refira-se: uma placa de desenvolvimento (Arduíno Mega), um módulo de Ethernet (Arduíno Shield), dois leitores RFID, um ecrã LCD e respetivo PCB e um teclado bem como todas as ligações necessárias. Tendo em conta que o objetivo principal desta primeira versão do sistema compreendia o controlo do acesso a uma zona restrita, foi necessário proceder à instalação de um trinco elétrico e de dois leitores de RFID colocados um no interior e outro no exterior da porta de acesso ao referido laboratório.

O passo seguinte no trabalho desenvolvido compreendeu a implementação de uma plataforma fixa, a qual é responsável pela monitorização da assiduidade dos trabalhadores do ISEC (Figura 52). A plataforma encontra-se localizada no edifício da Presidência, no *hall* de acesso à secretaria, recursos humanos e serviços da Presidência.

O protótipo desenvolvido teve por base o *hardware* da solução anterior tendo sido incluído um cartão de memória SD e retirado o teclado. Apesar das especificidades da solução pretendida foram utilizadas a maioria das funções anteriormente desenvolvidas. Dada a importância dos dados recolhidos foram acrescentadas novas funcionalidades que tendem a salvaguardar possíveis falhas de comunicação e de energia.

Esta funcionalidade de salvaguarda de possíveis falhas de comunicação, consiste na implementação de uma função que verifica se existe ligação ao servidor aquando da presença de uma *tag*, se esta não existir, o registo é gravado num cartão de memória. Caso existiam falhas de comunicação com o servidor, o módulo tenta efetuar uma ligação ao servidor com algum tempo de intervalo. Quando a ligação estiver reposta, o sistema faz o *upload* dos registos dos trabalhadores que ocorreram durante a falha de ligação para o servidor e apaga os mesmos do cartão de memória após confirmação desse registo.

Numa primeira fase, os dados recolhidos pela plataforma fixa foram enviados para uma tabela de registos, de modo a testar a conectividade e o mecanismo de salvaguarda de falhas da ligação de rede.

Figura 52 – Sistema de registo de assiduidade de trabalhadores.

Numa fase posterior foi desenvolvido um protótipo portátil, representado na Figura 53, com o qual se pretendia efetuar o registo de assiduidade, dos alunos e docentes, durante as aulas como de um livro de ponto digital se tratasse. Dada a natureza portátil da referida unidade, foi necessário implementar um meio de gravação dos dados recolhidos para posterior *upload* para o servidor por ação de um trabalhador. Assim sendo, para além dos módulos mencionados anteriormente: placa de desenvolvimento, módulo de Ethernet, leitor de RFID, RTC, LCD e teclado existe uma bateria de lítio e respetivo carregador.

Figura 53 – Sistema de registo de assiduidade móvel.

4.4. Conclusões do Capítulo

Neste capítulo foram apresentadas as escolhas dos componentes e módulos do sistema que se pretendiam desenvolver, nomeadamente, o microcontrolador, módulo de gestão de energia, bateria, módulo RFID, módulo Ethernet e carregador. Estas escolhas foram efetuadas de forma a encontrar um compromisso entre o desempenho do sistema e o custo (baseado nos preços do mercado). Uma vez escolhidos os componentes, passou-se a apresentar em detalhe a implementação do *hardware* e depois do *software*, com apresentação de todos os scripts, funções de inicialização e programas utilizados para cada um dos componentes utilizados. No fim do capítulo apresenta-se os protótipos desenvolvidos e implementados no decurso do presente projeto de mestrado.

5. CONTROLO E GESTÃO DO SISTEMA

Após se ter desenvolvido o *hardware*, foi necessário implementar um sistema de informação onde seja possível ao utilizador, aceder, modificar e alterar os parâmetros, registar utilizadores, cartões e permissões, bem como consultar todos os registos efetuados e processados. Para servir de suporte a estas funcionalidades é utilizado um servidor *web*, onde se encontra alojada uma base de dados (MySQL) para consulta e registo de toda a informação e para servir de *interface* com o utilizador. Para o efeito foram desenvolvidas páginas em HTML com estilos em CSS, para controlar a página e ser possível a ligação desta e do microcontrolador à base de dados, foi utilizado PHP.

5.1. Funcionalidades

Todos os sistemas de controlo e gestão de acessos têm normalmente um *interface* com o utilizador, onde é possível consultar registos de uma base de dados e interagir com o sistema. Para além dos pontos de acesso, onde é controlado o acesso e monitorizada a assiduidade, é necessário centralizar toda a informação num servidor, para que seja possível editar os diferentes elementos e regras de utilização do sistema. Assim na página de controlo e gestão, será possível:

- Adicionar / Remover Utilizadores;
- Adicionar / Remover Permissões de Acesso;
- Adicionar / Remover *Tags* de Utilizadores;
- Consultar Registos de Entrada e Saída;

No que concerne à consulta de registos de entrada e saída de um dado laboratório estes podem ser apresentados na forma de uma tabela ou num diagrama de Gant (Figura 54), tendo esta última funcionalidade sido desenvolvida pelo Eng.º Samuel Pereira.

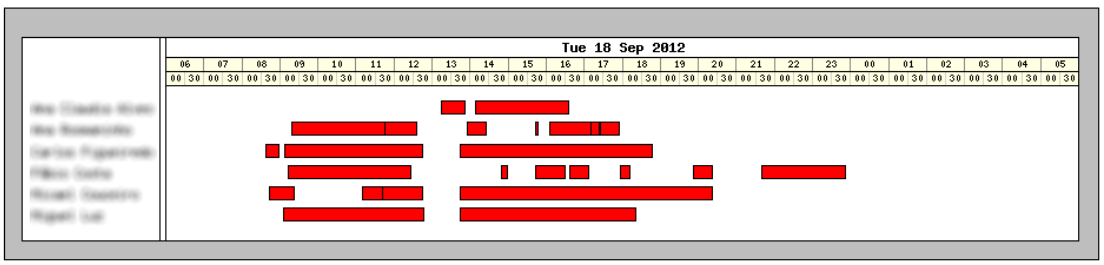

Figura 54 – Exemplo de diagrama de gant.

5.2. Tecnologias Utilizadas

Para o desenvolvimento do sistema e testes foi desenvolvida uma aplicação baseada em tecnologias web, a qual implementa as funcionalidades necessárias para o funcionamento do sistema. A aplicação foi desenvolvida em HTML e PHP. Os dados relativos aos acessos dos laboratórios, registos de assiduidade dos trabalhadores e os relativos às aulas são registados numa base de dados MySQL.

5.2.1. HTML

O *HyperText Markup Language* (HTML) é uma linguagem utilizada para produzir páginas na web que em português significa linguagem de marcação de hipertexto. Os documentos HTML são interpretados por *browsers*, que a partir do código presente no documento cria uma página web conforme esperado. Todos os documentos em HTML tem etiquetas, palavras entre parênteses angulares (< e >), sendo que essas etiquetas são os comandos de formatação da linguagem. Um elemento é formado por um nome de etiqueta, atributos e valores (que podem ser outros elementos ou texto). Os atributos modificam os resultados padrões dos elementos e os valores caracterizam essa mudança.

As etiquetas básicas de HTML, são:

<html>: define o início de um documento HTML e define ao navegador que todo conteúdo é uma série de códigos HTML;

<head>: define o cabeçalho de um documento HTML, onde estão presentes informações sobre o referido documento;

<body>: define o conteúdo da página. Esta é a parte do documento HTML que é mostrada no *browser*. No corpo podem-se definir atributos comuns a toda a página, como cor de fundo, margens, e outras formatações.

Para o desenvolvimento de páginas HTML são utilizados editores HTML. Embora a edição em linguagem HTML de uma página *web* possa ser feita com qualquer editor de texto (notepad, notepad++, Word), editores HTML específicos oferecem vários recursos extras para auxiliar na criação de páginas, além disso, acrescentam outras funcionalidades, e muitos dão a opção de visualização do projeto, tanto em linhas de código HTML quanto o resultado delas no *design* da página.

5.2.2. CSS

O *Cascading Style Sheets* (CSS) é uma linguagem utilizada para definir a apresentação de documentos escritos em linguagens como HTML ou XML. A principal vantagem decorrente da sua utilização é a separação dos aspetos relacionados com a formatação com o conteúdo de um documento.

Em vez de colocar a formatação dentro do documento, o programador cria uma ligação para uma página que contém os diferentes estilos, procedendo de forma idêntica para todas as páginas. O CSS tem uma sintaxe simples e utiliza uma série de palavras em inglês para especificar os nomes de diferentes estilos de propriedade de uma página.

Exemplificando, numa página *html*, são inseridos dois títulos e um texto simples, para diferenciar os três, com *html* simples, seria necessário criar *tags* de formatação em cada um deles, cada uma das *tags* seria necessário inserir as formatações para cada um dos tipos de texto, quando for necessário voltar a formatar um título com a mesma formatação seria necessário voltar a formatar o título. Com a utilização da formatação CSS, apenas é necessário incluir uma *tag* de identificação do tipo de texto inserido, e no ficheiro CSS fica especificado a formatação pretendida.

Uma folha de estilo, consiste de uma lista de regras. Quando for preciso modificar o aspeto visual da página basta modificar apenas o ficheiro de estilo (CSS).

5.2.3. PHP

O PHP ("PHP: *Hypertext Preprocessor*) é uma linguagem *open-source*, usada no desenvolvimento de aplicações processadas no lado do servidor, com a capacidade de gerar conteúdo dinâmico que se adapta e modifica em diferentes cenários. O código é interpretado no lado do servidor pelo módulo PHP, que gera a página *web* a ser visualizada no lado do cliente. Atualmente é possível instalar o módulo PHP na maioria dos sistemas operativos de uma forma, gratuita.

Ao contrário de muitos comandos HTML, as páginas PHP contém código HTML juntamente com o código PHP, responsável pelas ações que variam consoante o valor das variáveis apresentadas. O código PHP é delimitado por *tags* iniciais e finais “<?php” e “?>” que lhe permitem utilizar linguagem PHP e ao fechar a *tag*, utilizar linguagem HTML.

O que distingue o PHP de algo como a linguagem Javascript é o código ser executado no servidor, assim o módulo PHP gera código HTML que é então enviado para o cliente. O cliente recebe os resultados da execução desse script, mas não tem como saber como é o

código fonte. É possível configurar o servidor para processar todos os arquivos HTML como PHP, assim não haverá nenhuma maneira dos utilizadores descobrirem se é utilizado PHP ou HTML numa determinada página.

Uma grande vantagem decorrente da utilização do PHP é o facto de ser extremamente simples para um iniciante, e ao mesmo tempo oferecer muitos recursos para o programador profissional.

5.2.4. MY SQL

O MySQL é um sistema de gestão de base de dados (SGBD), que utiliza a linguagem de consulta estruturada SQL (*Structured Query Language*) como *interface*. É atualmente uma das bases de dados mais populares, com mais de 10 milhões de instalações pelo mundo (Why MySql, 2012). O sucesso do MySQL é em grande parte devido à fácil integração com o PHP, além de possibilitar a interação com outras linguagens como Delphi, Java, C/C++, C#, Visual Basic, Python, Perl, e ASP) e quase todos os pacotes de hospedagem de internet incluem PHP em conjunto com MySQL.

5.3. Base de Dados (MySQL)

Para registar, guardar e consultar toda a informação do sistema é necessário uma base de dados. Dadas as vantagens anteriormente descritas, foi selecionada a base de dados MySQL, tendo em conta os requisitos fundamentais do sistema. Para o seu desenvolvimento foi utilizado o *software* "MySQL Workbench", esta plataforma, permite projetar a estruturar uma base de dados *offline* e posteriormente construi-la *online*.

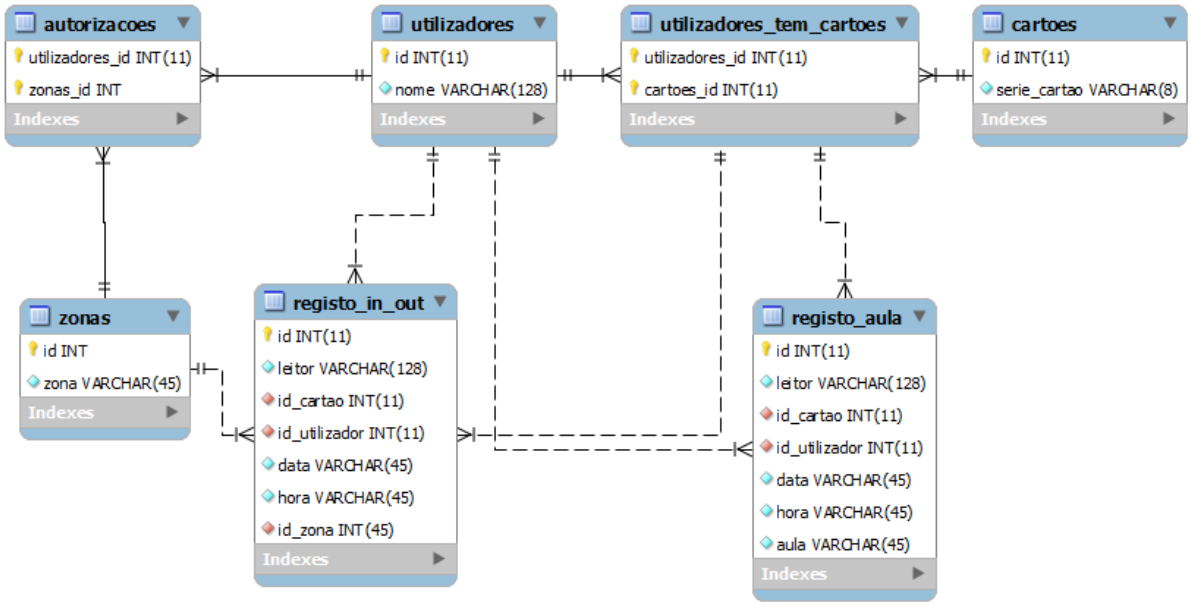

Figura 55 – Estrutura base de dados.

No desenvolvimento da estrutura da base de dados, foram tidos em conta quais os campos de informação fundamentais para que seja possível realizar a gestão do sistema, assim, foram criadas as tabelas de registo (Figura 55):

- Utilizadores – Esta tabela tem como função registar os utilizadores do sistema, e associar a cada um deles as salas e laboratórios a que têm acesso, bem como os períodos de acesso;
- Cartões – Nesta tabela são armazenados todos os IDs únicos de cada *tag* usada no sistema, identificada por uma chave primária única;
- Utilizadores_tem_cartoes – Com os registos feitos nesta tabela é possível cruzar o ID do utilizador com o ID do cartão (*tag*), e assim, associar as *tags* registadas aos utilizadores registados;
- Registo_in_out – Nesta tabela são registados todos os acessos a espaços, sendo assim possível consultar os acessos a salas de aulas, laboratórios e outros;
- Registo_Aula – Todas as aulas cujos acessos foram registadas na plataforma portátil são registadas na tabela, de forma a ser possível consultar as presenças;
- Zonas – Esta tabela contem as zonas monitorizadas com o sistema;
- Autorizações – As autorizações de cada utilizador a cada uma das zonas existentes é registada nesta tabela.

5.4. Aplicação Web Desenvolvida

Para realizar a *interface* da plataforma de gestão e controlo de acessos com os seus utilizadores, foi desenvolvida uma aplicação baseada em PHP e HTML. Esta aplicação permite de uma forma simples e intuitiva que o técnico superior designado pela Presidência do ISEC para essas funções, aceda aos registos da base de dados, aos registos de utilizadores e cartões, bem como consulte e edite as autorizações. A aplicação permite também a consulta dos registos de entradas e saídas de salas de aulas, dos registos de presenças em aulas bem como do registo de assiduidade dos trabalhadores.

Figura 56 – Página de autenticação de portal Web.

Na página de entrada do portal (Figura 56), está identificada a instituição onde o sistema está aplicado, neste caso, o Instituto Superior de Engenharia de Coimbra, e a informação necessária para fazer login e aceder às funcionalidades da aplicação.

Após o login, é mostrada a barra de navegação (Figura 57), onde é possível escolher a operação que se pretende realizar e/ou conjunto funcional onde se pretende atuar, entre elas refere-se:

- Início;
- Utilizadores;
    - Mostrar utilizadores;
    - Inserir Utilizadores;
- Cartões;
    - Mostrar Cartões;
    - Inserir Cartões;
- Registos;
    - Consultar Registos de Entradas/Saídas;
    - Consultar Registos de Pica-Ponto;
    - Consultar Registos de Assiduidade em Aulas;
    - Registar Entradas/Saídas;
    - Registas Pica-Ponto;
    - Registar Aulas;
    - Registo Manual;
- Terminar Sessão.

Figura 57 – Barra de menu.

5.4.1. Inserir, Consultar e Editar Utilizadores

Para inserir, consultar e editar a informação relativas aos utilizadores, foram desenvolvidas três páginas específicas para esse fim.

A zona de inserção de utilizadores é acessível pelo submenu "Utilizadores", disponível na barra de navegação, "Inserir Utilizadores", bem como a página de consulta de utilizadores "Consultar Utilizadores.

Na página "Consulta de Utilizadores" (Figura 58), é possível eliminar um utilizador (se este não tiver cartões associados nem registos), consultar e editar o utilizador.

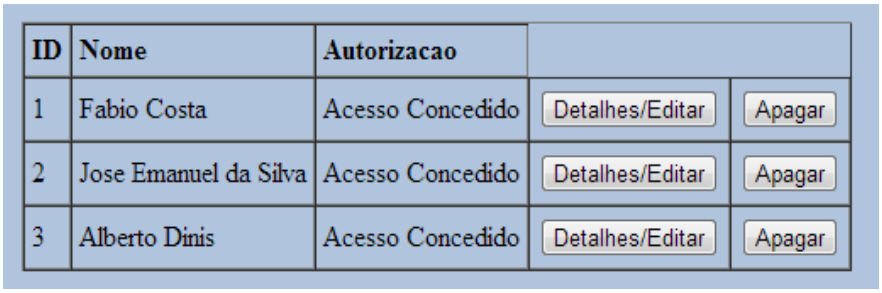

Figura 58 – Exemplo consulta de utilizadores.

A página de "Edição de Utilizador" (Figura 59), permite além de editar os dados de utilizador, consultar se existem cartões registados no utilizador atual, e se sim quais os números de série de cada um deles.

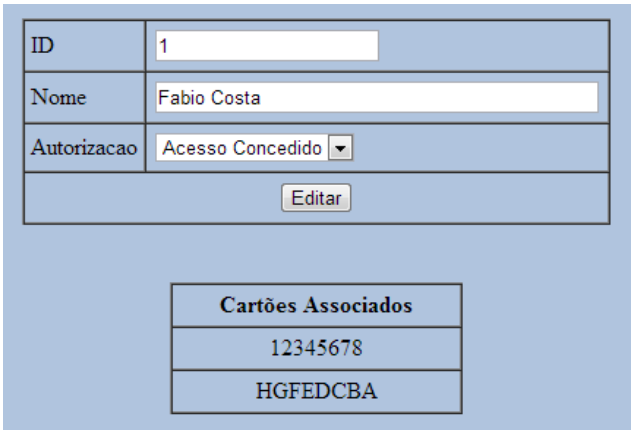

Figura 59 – Exemplo edição de utilizador.

5.4.2. Adicionar, Consultar e Eliminar *Tag's* e Associar Utilizador

Para que a plataforma funcione é indispensável, a inserção de cartões na base de dados (Figura 60), sendo apenas necessário para esta ação inserir o número de série de cada *tag*.

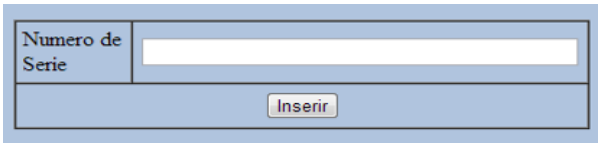

Figura 60 – Inserir cartões.

Após a inserção a partir da página de consulta de cartão (Figura 61), é possível consultar: o utilizador associado a cada *tag*, caso exista algum e eliminar o *tag* se este não tiver um utilizador associado, eliminar um utilizador associado ao *tag* (Eliminar Relação); adicionar um utilizador a uma *tag* (Adicionar Relação, Figura 62) e também trocar o utilizador associado a uma *tag*.

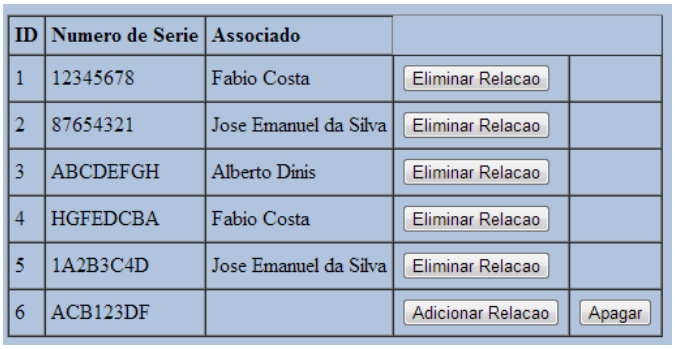

Figura 61 – Exemplo consulta de cartões registados.

Figura 62 – Exemplo de adicionar/editar relação Cartão-Utilizador.

5.4.3. Consulta de Registos

Através do menu "Registos" é possível consultar os registos feitos na base de dados. As funcionalidades existentes permitem consultar os registos de entradas/saídas provenientes do controlo de acessos, os registos de assiduidade de discentes e as presenças dos alunos nas aulas utilizando respetivamente as opções "Consultar Entradas/Saídas", "Consultar Registos de "Pica-Ponto" e "Consultar Registos de Aulas".

O sistema de gestão e controlo de acessos registou de 2 de abril até 31 de outubro de 2012, cerca de 7700 registos de entradas e saídas ao laboratório da RoboCorp na base de dados, como se pode observar na Figura 63.

ID	Zona	Leitor	Cartao	Utilizador	Data	Hora
7692	RC	RC_OUT	FCC4F48A	Tiago Crespo	31-10-2012	20:47:48
7691	RC	RC_OUT	4ECE9EE7	Ana Claudia Alves	31-10-2012	20:25:16
7690	RC	RC_IN	4ECE9EE7	Ana Claudia Alves	31-10-2012	20:25:11
7689	RC	RC_IN	FCC4F48A	Tiago Crespo	31-10-2012	20:00:36
7688	RC	RC_OUT	FCC4F48A	Tiago Crespo	31-10-2012	19:57:47
7687	RC	RC_OUT	4ECE9EE7	Ana Claudia Alves	31-10-2012	19:50:53
7686	RC	RC_IN	4ECE9EE7	Ana Claudia Alves	31-10-2012	19:43:01
7685	RC	RC_OUT	9ECDDD4F	Micael Couceiro	31-10-2012	19:37:02
7684	RC	RC_OUT	5DF6CF38	Miguel Luz	31-10-2012	19:35:17
7683	RC	RC_OUT	884EF5E	Fabio Costa	31-10-2012	18:58:09
7682	RC	RC_OUT	CEEEC8A	Carlos Figueiredo	31-10-2012	18:55:38
7681	RC	RC_OUT	7E10DB4F	Filipe Clemente	31-10-2012	18:47:12
7680	RC	RC_IN	CEEEC8A	Carlos Figueiredo	31-10-2012	18:39:20
7679	RC	RC_IN	5DF6CF38	Miguel Luz	31-10-2012	18:15:53
7678	RC	RC_IN	FCC4F48A	Tiago Crespo	31-10-2012	18:11:10

Figura 63 – Exemplo consulta de registos de Entradas/Saídas.

5.5. Conclusões do Capítulo

Este capítulo abordou o desenvolvimento da plataforma de gestão do sistema *online*. Inicialmente são enunciadas as funcionalidades pretendidas, bem como a importância da página *web* no desempenho do sistema. De seguida são referidas as tecnologias utilizadas para o seu desenvolvimento e no final é demonstrado o resultado final, bem como a descrição do seu funcionamento.

6. CONCLUSÕES

A presente tese de mestrado teve como principal objetivo o desenvolvimento de uma plataforma integrada para controlo e gestão de acessos vocacionada para instituições de ensino superior, usando a tecnologia RFID incorporada nos cartões de utilizador existentes.

Numa primeira fase estudou-se os requisitos da instituição de ensino superior para a qual o sistema foi concebido. Nomeadamente identificam-se os pontos de acesso ao campus do ISEC e aos seus edifícios, os dados necessários para proceder ao registo de assiduidade dos trabalhadores e ao registo da presença nas aulas por parte dos alunos.

Analisados os requisitos, procedeu-se à conceção do sistema. Para o efeito foram desenvolvidos fluxogramas de funcionamento e descritas as funcionalidades que se pretendem para cada uma das plataformas a desenvolver. Entre elas refira-se o controlo e gestão de acessos a laboratórios e salas de aulas, onde se pretende possibilitar aos alunos, docentes e não docentes, o acesso físico às instalações, caso estes utilizadores tenham autorização para o efeito. Ao mesmo tempo pretende-se manter um registo de movimentos e acessos efetuados. A segunda plataforma desenvolvida compreendeu um livro de ponto digital portátil cujo objetivo contempla o registo da hora de início e fim de cada aula, registo de todos os alunos presentes (para controlo de assiduidade) e posterior inserção da informação numa base de dados. A presidência do ISEC dispõe assim de uma ferramenta de gestão de recursos humanos de suporte à elaboração da distribuição de serviço docente que permite o ajuste da utilização das salas de aulas das diferentes unidades curriculares. Outra funcionalidade que se pretendia implementar compreendeu o sistema de registo de assiduidade dos trabalhadores, situado no acesso à presidência do ISEC, onde é registada a hora de entrada e saída dos mesmos. O sistema de gestão realiza um relatório com o número de horas diárias, semanais e mensais dos referidos trabalhadores. Este sistema contempla ainda a capacidade de criar alarmes, estabelecer regras e aceitar faltas justificadas de acordo com a lei.

Posteriormente foram escolhidos os componentes a utilizar nas diversas plataformas, entre eles refira-se: o microcontrolador, o leitor RFID, o módulo *Ethernet*, o ecrã LCD, o módulo de RTC, o teclado, o módulo de gestão de energia da bateria e respetiva bateria e carregador. Após o processo de seleção dos componentes desenvolveram-se as placas PCB necessárias para o funcionamento do sistema e interligação dos módulos. De entre elas refira-se o *interface* do ecrã LCD, o expansor de I/O para o teclado e a que implementa o módulo de potência.

Concluída a implementação do *hardware*, procede-se ao desenvolvimento de *software*, utilizando bibliotecas específicas bem como o *software* desenvolvido para todo o funcionamento da plataforma.

De seguida são indicados os protótipos desenvolvidos, refira-se entre eles o protótipo de controlo e gestão de acessos ao espaço RoboCorp, onde foram efetuadas cerca de 7700 registos, o protótipo de registo de ponto dos trabalhadores e o protótipo de livro de ponto digital portátil, que efetua o registo de assiduidade dos alunos nas aulas.

Na presente tese, foi também desenvolvida uma plataforma web, que consiste num *interface* onde serão efetuados os registos provenientes das plataformas anteriormente mencionadas. A mesma aplicação permite a consulta dos registos efetuados, bem como o registo de novos utilizadores e respetivas *tags* e a definição das autorizações individuais.

Em suma a plataforma desenvolvida provou ser capaz de satisfazer as necessidades pretendidas, revelando ser uma ferramenta polivalente de apoio à gestão das IES e versátil.

6.1. Perspetivas de Trabalho Futuro

Face ao trabalho realizado no projeto, que foi descrito na presente tese e na evolução que se pretende para versões vindouras do protótipo desenvolvido, sugerem-se os seguintes tópicos de melhoramentos e trabalho futuro:

- Desenvolvimento de um único PCB que integrasse todos os componentes;
- Desenvolver um novo módulo de energia para as unidades móveis;
- Redesenhar a base de dados;
- Desenvolver o portal *web* em *javascript*;
- Implementar a plataforma para a gestão das entradas e saídas no campus;
- Incorporar de *cookies* e outros sistemas de segurança;
- Implementar um sistema de login diferenciado por tipo de utilizador

7. BIBLIOGRAFIA

(13.56 MHz Mifare Read/Write Module, 2012) – *13.56 MHz Mifare Read/Write Module*. (Julho de 2012). (EHOUYAN)

(Akpinara & Kaptan, 2010) – Akpinara, S., & Kaptan, H. (2010). Computer aided school administrarion system using RFID tecnology. *Procedia - Social and Behabioral Sciences, Vol2, Issue2*, pp. 4392-4397.

(Arduino, 2011) – *Arduino*. (2011). Obtido de http://arduino.cc/

(Atmel) – Atmel. (s.d.). *Atmel AVR 8-bit and 32-bit Microcontrollers - Datasheet*. Obtido de http://www.atmel.com/products/microcontrollers/avr/default.aspx?tab=documents

(Caixa Geral de Depositos, 2012) – Caixa Geral de Depositos. (2012). *Cartão da Caixa Premiado na OSCARDS 2008*. Obtido de https://www.cgd.pt/Institucional/Sala-Imprensa/2008/Pages/Cartao-CUP.aspx

(Carlosle) – Carlosle, T. (s.d.). Radio Frequency Identification - RFID ... Coming of Age. *Information Technology Association of America (ITAA).*

(Chiesa, et al.) – Chiesa, M., Genz, R., Heubler, F., Kim, M., Noessel, C., Sopieva, N., . . . Tester, J. (s.d.). *RFID - a week long survey on the technology and its potencial.* (H. T. Project, Ed.) Obtido de http://www.interaction-ivrea.it

(Correia, Carvalho, & Nunes, 2002) – Correia, A. I., Carvalho, S. G., & Nunes, S. S. (2002). *Parque de estacionamento da FEUP, Aplicação da Soft Systems Methodology, Especificação de Requisitos e Modelação de um Sistema de Informação.* Universidade do Porto, FEUP.

(Costa, et al., 2010) – Costa, F. J., Pereira, S., Rosmaninho, A., Couceiro, M. S., Figueiredo, C. M., Santos, V., & Ferreira, N. F. (2010). Low-Cost Access Management System in an Educational Environment. *9th WSEAS International Conference on EDUCATION and EDUCATIONAL TECHNOLOGY (EDU '10)*, (pp. 146-151). Porto.

(Cravo Gomes, 2007) – Cravo Gomes, H. M. (2007). *Construção de um sistema de RFID com fins de localização.* Dissertação de Mestrado, Universidade de Aveiro.

(Farnell) – Farnell. (s.d.). *Datasheet da Bateria de Litio*. Obtido de http://www.farnell.com/datasheets/810103.pdf

(Finkenzeller, 2003) – Finkenzeller, K. (2003). *RFID Handbook: Fundamentals and Applications in Contact-less Smart Cards and Identification (2nd Ed.).* J. Wiley and Aons.

(Gines & Tsai, 2007) – Gines, F. G., & Tsai, T. T. (2007). *Projecto e Implementação de um Sistema de identificação por RFID para uma aplicação de automação residencial.* Relatório de Projecto de Final de Licenciatura, Escola Politécnica Universidade São Paulo.

(Harry, 1948) – Harry, S. (Outubro de 1948). Communication by Means of Reflected Power. (A. N. Goldsmith, Ed.) *Proceedings Of The I.R.E., 36*, pp. 1196-1204.

(Instrumens) – Instruments, T. (s.d.). *Remote 8-Bit I/O Expander for I2C-Bus (PCF8574)*. Obtido de http://www.ti.com/product/pcf8574

(Jandt, 2005) – Jandt, J. (2005). *The history of RFID* (Vol. Volume 24). IEEE.

(Kabir, Huang, Wu, & Rapajic, 2007) – Kabir, A., Huang, K. C., Wu, R., & Rapajic, P. (31 de Dezembro de 2007). Next Generation Identity Card: RFID-bases Automatic Access Control System for Universities. *6th WSEAS Internacional Conference on CIRCUITS, SYSTEM, ELECTRONICS, CONTROL & SIGNAL PROCESSING*, pp. 480-483.

(Kurose & Ross, 2005) – Kurose, J., & Ross, K. (2005). *Computer Networking: A Top-down Approach Featuring the Inthernet* (Vol. Third Edition). Addison Wesley.

(Lourenço & Almeida, 2009) – Lourenço, F., & Almeida, C. (Setembro de 2009). RFID based Monitoring and Access Control System. *INFORUM Simpósio de Informática*.

(Madeira, Antunes, Morgado, & Pereira, 2008) – Madeira, A., Antunes, R., Morgado, L., & Pereira, A. (2008). Controlo de assiduidade em aulas efectuadas em mundos virtuais@ Secound Life (R). *III Conferencia Ibérica de Sistemas y Tecnologías de la información, CESTI*.

(NFC-Forum, 2012) – *NFC-Forum*. (Fevereiro de 2012). Obtido de http://www.nfc-forum.org

(RFIDjournal) – RFIDjournal. (s.d.). Obtido de How much does an RFID tag cost today?: http://www.rfidjournal.com/faq/show?85

(SeeedStudio, Lipo Rider) – SeeedStudio. (s.d.). *Lipo Rider*. Obtido de http://www.seeedstudio.com/depot/lipo-rider-p-710.html

(SeeedStudio, Lipo Rider Pro) – SeeedStudio. (s.d.). *Lipo Rider Pro*. Obtido em Abril de 2012, de http://www.seeedstudio.com/depot/lipo-rider-pro-p-992.html

(SparkFun) – (SparkFun. (s.d.). *Real Time Clock Module*. Obtido de https://www.sparkfun.com/products/99

(Why MySql, 2012) – *Why MySql*. (Setembro de 2012). Obtido de http://www.mysql.com/why-mysql/

(WizNet, 2012) – WizNet. (2012). *W5100 Official Page*. Obtido de http://www.wiznet.co.kr/Sub_Modules/en/product/Product_Detail.asp?cate1=5&cate2=7&cate3=26&pid=1011

8. ANEXOS

8.1. Excerto da Adenda do Quadro Nacional de Atribuição de Frequências (QNAF) em 2013

Tipo de equipamento de curto alcance	Faixa de frequências	Limite de potência de emissão/limite de intensidade de campo/limite de densidade de potência	Parâmetros adicionais	NOTAS
Aplicações Indutivas[1]	9 – 90 kHz	72 dBµA /m a 10 metros		Decisão 2011/829/UE
	90 – 119 kHz	42 dBµA /m a 10 metros		Decisão 2011/829/UE
	119 – 135 kHz	66vdBµA /m a 10 metros		Decisão 2011/829/UE
	135 – 140 kHz	42 dBµA /m a 10 metros		Decisão 2011/829/UE
	140 – 148,5 kHz	37,7 dBµA /m a 10 metros		Decisão 2011/829/UE
	148,5 – 5000 kHz Nas faixas específicas abaixo indicadas, aplicam-se limites mais elevados para a intensidade de campo e restrições de utilização adicionais:	– 15 dBµA/m a 10 metros em qualquer largura de banda de 10 kHz Adicionalmente, a intensidade de campo total é – 5 dBµA/m a 10 metros para os sistemas que operam em larguras de banda superiores a 10 kHz.		Decisão 2011/829/UE
	400 – 600 kHz	– 8 dBµA/m a 10 metros		Decisão 2011/829/UE
	3155 – 3400 kHz	13,5 dBµA/m a 10 metros		Decisão 2011/829/UE
Aplicações indutivas[1] (cont.)	5 000 – 30 000 kHz Nas faixas específicas abaixo indicadas, aplicam-se limites mais elevados para a intensidade de campo e restrições de utilização adicionais:	– 20 dBµA/m a 10 metros em qualquer largura de banda de 10 kHz Adicionalmente, a intensidade de campo total é – 5 dBµA/m a 10 metros para os sistemas que operam em larguras de banda superiores a 10 kHz.		Decisão 2011/829/UE
	6 765 – 6 795 kHz	42 dBµA /m a 10 metros		Decisão 2011/829/UE
	7 400 – 8 800 kHz	9 dBµA /m a 10 metros		Decisão 2011/829/UE

	10 200 – 11 000 kHz	9 dBµA/m a 10 metros		Decisão 2011/829/UE
	13 553 – 13 567 kHz	42 dBµA /m a 10 metros		Decisão 2011/829/UE
		60 dBµA /m a 10 metros		Decisão 2011/829/UE
	26 957 – 27 283 kHz	42 dBµA /m a 10 metros		Decisão 2011/829/UE
Identificação por radiofrequências (RFID)	2446 – 2454 MHz	500 mW p.i.r.e.		Decisão 2011/829/UE
	865,0 – 865,6 MHz	100 mW p.a.r.	Espaçamento entre canais: 200 kHz	Recomendação 70-03
	865,6 – 867,6 MHz	2 W p.a.r.	Espaçamento entre canais: 200 kHz	Recomendação 70-03
	867,6 – 868,0 MHz	500 mW p.a.r.	Espaçamento entre canais: 200 kHz	Recomendação 70-03

Nota

[1] Esta categoria abrange, por exemplo, os dispositivos para imobilização de veículos, identificação de animais, sistemas de alarme, deteção de cabos, gestão de resíduos, identificação pessoal, ligações de voz sem fios, controlo do acesso, sensores de proximidade, sistemas anti-roubo, incluindo os sistemas anti-roubo RF por indução, transferência de dados para dispositivos de mão, identificação automática de artigos, sistemas de controlo sem fios e portagem rodoviária automática.

[2] Esta categoria abrange as aplicações indutivas utilizadas na identificação por radiofrequências (RFID).

[3] Esta categoria abrange as aplicações indutivas utilizadas na vigilância eletrónica de artigos (EAS).