UNIVERSIDADE NOVA DE LISBOA

Faculdade de Ciências e Tecnologia

Departamento de Engenharia Electrotécnica

Programa de Controlo de Acessos com Configuração de Regras em XML

Volume I

Por

Pedro Nuno Extreia Ribeiro Semeano

Dissertação apresentada na Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa para obtenção do grau de Mestre em Engenharia Electrotécnica e de Computadores

Orientador: Prof. José Manuel Fonseca

Lisboa, 2009

Aos meus amigos.

Sumário

Nunca antes como nos dias de hoje se deu tanta importância à segurança. Em muitas empresas é uma questão que não é deixada ao acaso e novas soluções surgem para tomar o lugar de uma simples fechadura que por vezes não é suficiente para proteger empregados ou documentos importantes. A tecnologia das fechaduras electrónicas têm vindo a evoluir e é uma opção preferencial não só de grandes empresas como também de alguns dos maiores hotéis, oferecendo maior nível de segurança e permitindo também um registo de "movimentos".

No entanto, na maior parte dos casos, o controlo de acessos é feito de um modo estático, ou seja, um cartão de acesso é registado para um conjunto de portas a que tem acesso e assim permanece até nova alteração. Com este trabalho foi possível realizar um software de controlo de acessos mais dinâmico, que permite criar regras de acesso de acordo com as necessidades do cliente, algo que no momento actual não existe no mercado e que pode ser utilizado para aumentar os padrões de segurança no software de controlo de acessos.

Abstract

Never like today security was so important. In many enterprises it is a major issue that can't be forgotten and newer solutions come to take place of a simple lock when sometimes is not enough to protect employees or important documents. Electronic locks technology has evolved and is preferred by the major multinationals and also by the greatest hotels, offering increased security levels also allowing to trace every move.

Nevertheless, in most cases, the access control is very static, that is, an access card gets authorized for a bunch of doors and stays that way until rules are changed. In this work a more dynamic access control software was built, allowing to create rules of access accordingly the needs of the client, something that wasn't done already and that can change the security patterns of access control software.

Índice Geral

Índice de figuras

Índice de quadros

1. Introdução

Hoje em dia a questão da segurança é algo que interessa a qualquer cidadão. Todos queremos andar completamente à vontade em qualquer lugar com a certeza de que não corremos qualquer tipo de perigo. Todos nos preocupamos com a nossa integridade física e não queremos que ela seja comprometida por acções de terceiros as quais não podemos controlar, seja na rua ou dentro de um edifício, num espaço público ou no local de trabalho, em casa ou no carro. É por isso que recorremos a ferramentas ou serviços que nos oferecem segurança, como por exemplo o alarme que colocamos no carro e em casa, um cão ou até mesmo um guarda-costas (para os mais abastados). Há também quem prefira melhorar a sua condição física e auto-estima frequentando ginásios ou praticando artes marciais. Enfim, o leque é vasto e todos eles apontam num único objectivo: segurança.

Obviamente que a questão da segurança não passa só pela protecção da integridade física. Existem outros tipos de ameaças que põem em causa a segurança, como por exemplo um ataque por parte de um *hacker*. Um *hacker*, sem sequer sair do conforto da sua cadeira em frente ao computador, pode pôr em causa a segurança de dados informáticos confidenciais, o que pode gerar grande alarme e inúmeros prejuízos. Este tipo de ataque não é tão inofensivo quanto parece porque se alguém conseguir obter acesso a várias contas bancárias facilmente consegue arruinar a vida dos seus donos legais ou até mesmo da instituição bancária que foi alvo de ataque. Ficar sem as poupanças de uma vida de trabalho é algo que certamente ninguém deseja. Nunca como hoje a questão da segurança informática foi tão levada a sério. Antivírus, *firewalls* e *anti-spywares* fazem parte de um conjunto de ferramentas de protecção que oferecem o mínimo de segurança que hoje em dia um computador deve ter de forma a proteger-se dos vários ataques que acontecem todos os dias milhares de vezes por dia. E muitas das vezes não é suficiente. Existem pessoas com grandes conhecimentos informáticos que facilmente ultrapassam estas barreiras. E não só conseguem atacar facilmente mas como

também conseguem proteger-se muito bem, tornando difícil seguir o seu rasto e impossibilitando a encontrar o autor do ataque.

O tema deste trabalho enquadra-se tanto no campo da protecção da integridade física como no de protecção de dados. O software de controlo de acessos desenvolvido (para o Gabinete Nacional de Segurança) permite controlar quando e onde é permitido o acesso a cada utilizador, que possui um determinado cartão que lhe foi entregue, tornando possível vigiar o seu comportamento, registando e alertando situações não desejadas ou suspeitas.

A tecnologia de hoje em dia evoluiu de forma nunca antes vista. Basta olhar para um passado não muito distante e ver o que era a tecnologia *state of the art* e que hoje está completamente ultrapassada. Na verdade hoje em dia o desenvolvimento tecnológico cresce a um ritmo tão acelerado que o que é hoje topo de gama daqui a poucos meses pode estar já obsoleto. A sua evolução tem permitido realizar ideias e transformar simples objectos em tecnologia de ponta como, por exemplo, as fechaduras. Até há poucos anos as fechaduras não eram mais que meros objectos puramente mecânicos, que de acordo com um segredo impresso numa chave, abriam o trinco. Hoje em dia temos fechaduras bem mais evoluídas, onde parte da sua estrutura mecânica ainda se mantém mas cuja abertura é efectuada a partir da leitura de um cartão (por banda magnética, RFID ou chip) ou através de dados biométricos, utilizando sistemas para a leitura da impressão digital, íris, padrão da mão, ou até mesmo o padrão da face.

Em seguida ir-se-á abordar alguns destes sistemas de modo a manter o leitor a par da tecnologia utilizada na realização desde trabalho, nomeadamente RFID e sistemas biométricos de leitura de impressão digital.

2. Sistemas de identificação pessoal

Hoje em dia é obrigatório andar pela rua com pelo menos um documento que nos identifique, seja o bilhete de identidade, carta de condução ou outro. Sem identificação não somos ninguém, ou pior, podemos ser qualquer um. Todos nós já vimos alguma reportagem de detenção de grupos criminosos de falsificação e reprodução de documentos de identificação tais como passaportes e bilhetes de identidade. É por isso que o uso do tradicional bilhete de identidade está a chegar ao fim, visto que facilmente é reproduzido. Para ocupar o seu lugar estão a aparecer SmartCards que para além de serem mais difíceis de reproduzir, podem guardar diversa informação pessoal dentro do chip que contem. Mesmo os novos passaportes já vêm equipados com tecnologia RFID que, juntamente com tecnologia biométrica, permite identificar se realmente aquele passaporte pertence à pessoa que o detém.

Cada vez mais e com cada vez mais força a tecnologia se envolve nas nossas vidas, trazendo diversas vantagens em relação a sistemas mais simples e rudimentares. No entanto nem todos vêm com bons olhos esta nova era electrónica visto que muito facilmente se consegue obter informação sobre qualquer um de nós.

2.1. RFID

RFID significa *Rádio-Frequency IDentification* (identificação por rádio-frequência) e, tal como o nome indica, trata-se de um tipo de identificação que utiliza como meio de comunicação as ondas de rádio. Na 2ª Guerra Mundial os radares eram uma ferramenta essencial para a detecção de aviões quando estes ainda se encontravam a uma distância considerável, de forma a evitar ataques surpresa. O problema estava em identificar se os aviões a caminho eram inimigos ou não. Sob o comando de Sir Robert Alexander Watson-

Watt, o mesmo físico Escocês que melhorou a tecnologia do radar, deu-se início a um projecto secreto que visava a construção de um identificador activo de amigo. Este identificador era colocado nos aviões e quando estes recebiam os sinais da estação de radar emitiam um sinal de resposta que os identificava como amigos. A este dispositivo foi dado o nome de IFF – *Identify Friend or Foe*[1].

A identificação no RFID é feita utilizando um *transponder* (acrónimo de *Transmitter-Responder*) ou etiqueta, um pequeno dispositivo electrónico que tem uma

Figura 2.1 – Etiqueta de segurança[1].

pequena antena e que lhe permite responder aos sinais de rádio. Um exemplo bastante simples é o que vemos em praticamente todas as lojas de roupa. Cada peça de roupa vem com uma etiqueta com um aviso para remover antes de lavar ou usar. Dentro dessa etiqueta está uma peça de plástico. Essa peça é o *transponder* ou a etiqueta de RFID. À entrada da loja existem uns "postes" que mais não são do que a antena transmissora. Quando passamos pela porta da loja sem retirar a etiqueta, a antena emissora, que está constantemente a enviar um sinal de rádio, vai detectar a presença da etiqueta, fazendo soar o alarme. Este é um exemplo de uma etiqueta passiva (não tem bateria). No entanto, existem algumas que são activas (alimentadas por uma bateria) e que não necessitam de um sinal para responder, enviando o seu próprio sinal. Não só como método de segurança e alarme, as etiquetas RFID podem também servir para a gestão de stocks, em qualquer loja. Se ao entrar na loja ou armazém o produto já tiver a

etiqueta RFID, basta fazer passar o produto pela zona de leitura que este dá entrada automática no sistema de gestão de stocks. À medida que os diversos produtos se vão vendendo consegue-se ter um controlo da quantidade que ainda existe em stock e saber quando fazer novas encomendas.

Um outro exemplo é o que é utilizado nas provas de fórmula 1, de rally ou até mesmo de ciclismo. Na zona da meta existe uma antena que vai receber o sinal da etiqueta RFID, onde cada etiqueta tem o seu próprio código de identificação, que faz parte do sinal que é enviado, e é com base neste código que são calculados os tempos de cada um. Este é o funcionamento básico de todo o sistema RFID. Existe uma etiqueta que está à espera de ser lida; quando esta recebe o sinal do receptor, utilizando a energia da bateria interna ou a energia obtida através do campo electromagnético induzido pelo receptor, a etiqueta envia um sinal de rádio como resposta que vai ser interpretada pelo receptor.

O tamanho do *transponder* pode variar, de acordo com a aplicação. Por exemplo, para a identificação de livros numa biblioteca, não é necessário um *transponder* muito pequeno, mas convém que a sua espessura seja mínima, como também na sua implementação em cartões, como o cartão de aluno da faculdade, que permite, por exemplo, o acesso ao parque de estacionamento. No entanto, existem identificadores ainda mais pequenos; em 2009 investigadores da Universidade de Bristol, na Inglaterra, conseguiram colar pequenos *transponders* em formigas vivas, de modo a estudar o seu comportamento[2]. Estes não tinham mais que 3 mm de largura. No entanto, o recorde pertence à empresa de dispositivos electrónicos Hitachi, que conseguiu com sucesso criar um *transponder* com as dimensões de apenas 0,15 mm por 0,15 mm, e com 7,5 micrómetros de espessura! Para além do seu reduzido tamanho ainda consegue espaço para uma ROM de 128 bits que permite armazenar um código de identificação de 38 dígitos[3].

O RFID tem hoje em dia imensas aplicações. Eis algumas:

- Controlo de acessos a parques de estacionamento, edifícios, zonas privadas/exclusivas, etc.
- Localização de mercadorias, animais ou pessoas
- Pagamento em portagens, parques de estacionamento e bombas de gasolina

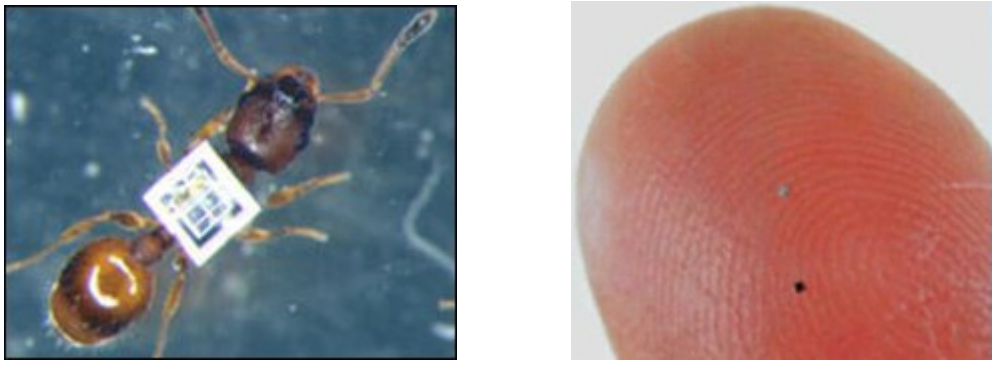


Figura 2.2 – Etiquetas RFID de reduzido tamanho.

- Leitura de documentos tais como passaportes ou outro tipo de cartões como modo de identificação pessoal
- Identificação de objectos tais como livros, DVD's, embalagens, etc.

Um dos pontos que gera alguma controvérsia é a implantação de chips RFID em humanos. Apesar de ter os seus pontos positivos, tais como o pagamento de bens com a simples aproximação da mão de um leitor, esta aplicação provoca preocupações no que diz respeito à privacidade de cada um, visto que o seu rasto e localização torna-se uma informação facilmente adquirida, o que poderia levar a abusos por parte de um governo autoritário.

Neste trabalho este tipo de tecnologia é utilizado nos cartões de acesso (com identificadores passivos) que são facultados a cada funcionário do Gabinete Nacional e que estão identificados com um número único e que dão acesso a determinados pontos do edifício, consoante as regras definidas no software de gestão de controlo de acessos.

2.2. Sistemas biométricos

Os sistemas biométricos são equipamentos electrónicos que, a partir das características de uma dada parte do corpo humano, efectuam uma série de cálculos e como resultado devolvem um código (*minutæ*) que vai identificar essa pessoa. Como cada pessoa é diferente da outra, e como esse código varia consoante as características de cada pessoa, então

teoricamente existirá uma minuta por pessoa. A maior parte dos sistemas biométricos existentes no mercado baseiam-se na análise da impressão digital (a forma mais popular), do padrão da mão, da face, da íris ou da voz.

No entanto, seja qual for o método utilizado para o reconhecimento de uma dada pessoa, os sistemas biométricos actualmente são apontados como um dos sistemas preferenciais para o controlo de acessos, visto que, ao contrário dos cartões, a informação necessária para permitir o acesso muito dificilmente se perde ou se torna alvo de roubo. Além disso, a pessoa não necessita de transportar cartões, decorar códigos ou qualquer outra chave de identificação; neste caso a chave é a própria pessoa, e cada pessoa é única.

No entanto, tal como qualquer outro sistema, a biometria não é infalível. Se perdermos um cartão de acesso ou se o PIN de acesso é descoberto por alguém, facilmente obtemos um novo cartão ou mudamos o código. Mas se alguém nos rouba a impressão digital não há praticamente nada a fazer, o que nos pode condenar para a vida toda, pois não poderíamos utilizar a nossa impressão digital como forma de identificação até que as cópias fossem destruídas.

A popularidade dos sistemas biométricos tem vindo a aumentar principalmente na área de sistemas controlo de entradas e saídas, onde o seu aparecimento veio reforçar (ou substituir) os métodos tradicionais, oferecendo maior segurança, conforto e robustez. Os sistemas biométricos não guardam uma amostra biométrica mas sim uma representação digital. Desta forma caso alguém tenha acesso ao registo do sistema, não conseguirá recriar os dados biométricos originais, nem sequer utilizá-los noutro sistema porque os sistemas biométricos não fazem uma digitalização da imagem obtida mas sim uma "codificação" dos dados calculados a partir da imagem. Para além do mais, qualquer sistema biométrico só funciona num sentido, ou seja, a partir dos dados biométricos recolhidos é possível chegar a um código que os representam (*template*); o inverso não é possível. Também, cada método de cálculo do *template* é único do fabricante e não está acessível às entidades que fornecem ou adquirem os equipamentos, assegurando a privacidade dos dados recolhidos.

Os sistemas biométricos têm basicamente duas formas de funcionamento: verificação ou identificação. A verificação é feita juntamente com qualquer outro tipo de identificação, como um código PIN, um SmartCard, nome de utilizador, etc. Após a identificação (com base nos métodos referidos) a utilização do sistema biométrico servirá apenas para verificar que o *template* gerado coincide com o que está registado para a pessoa que o utilizador afirma ser.

No método de identificação a autenticação é feita com base apenas nas características biométricas, ou seja, o *template* gerado irá ser comparado com todos os que estão guardados no registo até se encontrar um *template* idêntico (ou praticamente idêntico). Praticamente porque nem sempre, para as mesmas características, é gerado exactamente o mesmo *template* de identificação. Por exemplo, no caso de autenticação utilizando a impressão digital, por vezes basta uma colocação do dedo numa posição diferente da original (na altura do registo) para que o código gerado seja ligeiramente diferente. Por isso, no software de identificação é dado um intervalo de erro permitido, e caso a diferença entre o *template* obtido e o guardado no registo esteja dentro desse intervalo, então a autenticação é considerada válida.

Durante a execução desta tese de mestrado foi utilizado um sistema biométrico de leitura da impressão digital, e daí a decisão de dar maior ênfase a esta área dos sistemas biométricos.

2.2.1. Reconhecimento vocal

A identificação da pessoa através da sua voz é um pouco diferente dos outros métodos que se baseiam nas formas de partes do corpo humano. No entanto, a voz é influenciada pela estrutura da garganta, da boca, pela cavidade nasal, pelas cordas vocais, pela forma como mexemos a boca e a língua quando falamos, etc.

Existem duas formas de fazer a identificação: ou através de uma frase-chave que foi previamente guardada nos registos do sistema, e que será comparada com a frase que a pessoa terá que repetir para ter acesso; com base numa análise da "estrutura" da voz. A voz é analisada através da frequência e ritmo, criando um espectrograma da mesma através da qual é feita a identificação. Um espectrograma é basicamente um gráfico que no eixo vertical abrange um intervalo de frequências (normalmente até 8000 kHz) e no eixo horizontal é feita a sua representação ao longo do tempo. A maior parte dos espectrogramas são em tons de cinza onde os sítios mais escuros representam a maior concentração de energia. Também existem espectrogramas a cores onde o vermelho representa a maior concentração de energia, decrescendo para laranja, amarelo, verde, ciano, azul, magenta, cinza e finalmente branco onde os níveis de energia são considerados nulos.

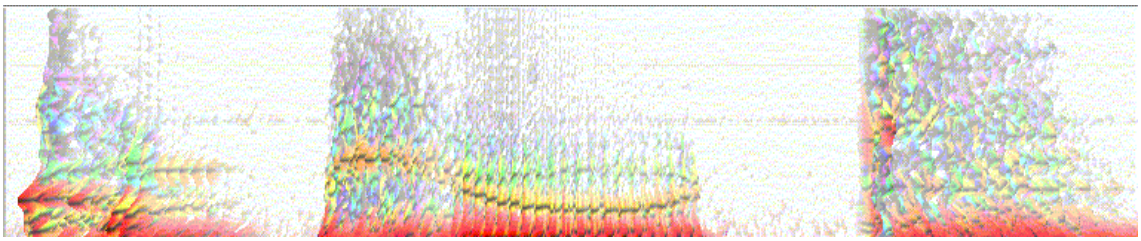


Figura 2.3 – Espectrograma a cores da palavra "compute"[25].

Um sistema de reconhecimento vocal tem uma particularidade diferente de todos os outros, caso se possa considerar uma vantagem ou uma desvantagem, mas é possível dar autorização de acesso sem que a própria pessoa esteja presente, fazendo-o, por exemplo, através de uma chamada telefónica. No entanto, esta situação pode ser uma desvantagem visto que facilmente se consegue "roubar" a chave com o auxílio de um gravador de voz.

2.2.2. Íris

Sem dúvida que o melhor método para distinguir indivíduos é a da leitura da íris, a parte colorida do olho. O desenvolvimento da íris começa no terceiro mês de gestação e termina ao oitavo mês, ou seja, esta não se altera ao longo da vida, ao contrário dos outros casos referidos (a voz altera-se com o passar do tempo, assim como a cara, a geometria e tamanho das mãos e dos dedos). Para além do mais, não segue nenhum padrão genético pelo que o seu desenvolvimento é totalmente aleatório. A probabilidade de falso positivo é bastante baixo (mais baixo que 10^{-11})[8], o que em teoria garante que não existirão duas pessoas com o mesmo padrão de íris, sobre a alçada do mesmo sistema de segurança.

Os leitores de íris são bastante simples. Consistem numa câmara que com o auxílio de uma fonte de luz e de infravermelhos consegue captar uma imagem do padrão da íris. Posteriormente essa imagem é analisada por um software que efectua uma série de cálculos. Caso o resultado final seja semelhante ao registo no sistema, a autenticação é bem sucedida.

Outra vantagem da íris é que é um órgão que está bem protegido, ao contrário das impressões digitais que podem deteriorar-se ao longo do tempo ou devido a execução de trabalhos manuais pesados. Para além disso, a forma da íris nunca se altera de pessoa para pessoa, ao contrário do que acontece por exemplo com a face.

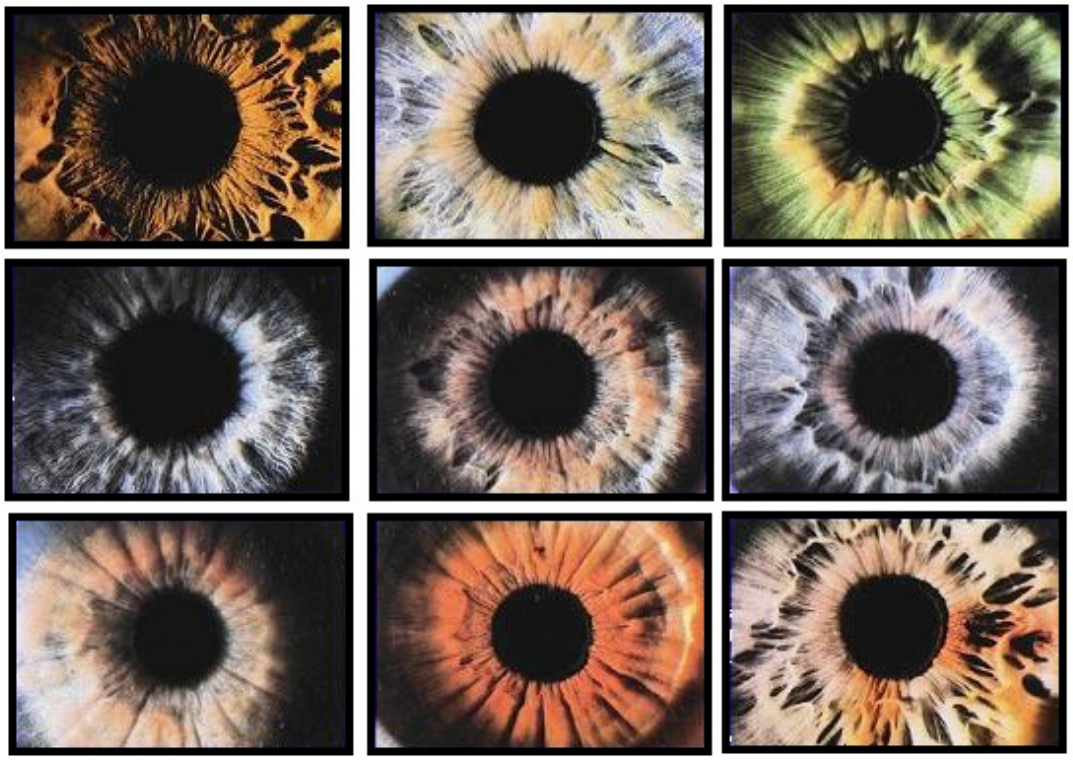


Figura 2.4 – Diferentes padrões da íris do olho humano.

No entanto, nem tudo são vantagens. A superfície do olho humano é curva e húmida, o que pode dificultar a captura da imagem da íris, não só por causa da sua geometria mas também por causa dos reflexos luminosos ao nível da córnea. Doenças como cataratas, conjuntivite ou outras doenças oftalmológicas impedem certas pessoas de serem identificadas por este tipo de equipamento.

2.2.3. Reconhecimento facial

O reconhecimento facial é outro dos métodos utilizados nos sistemas biométricos. Neste campo existem sistemas 2D e 3D.

Os sistemas 2D utilizam como base de comparação fotografias das pessoas a identificar necessitando desta forma de uma base de dados algo volumosa. O reconhecimento é feito a partir de uma fotografia de teste (tirada na altura) sobre a qual é feita a comparação com todos os registos na base de dados e aquela que mostrar mais semelhanças (acima de um

dado nível) é a escolhida. As medidas utilizadas na comparação são normalmente a distância entre os olhos, a largura da boca, a largura do nariz, a largura do maxilar, entre outras. No entanto, este método apresenta falhas visto que normalmente a tentativa de identificação é feita num ambiente não controlado, onde uma simples variação na luz ou postura da face facilmente leva o sistema a não encontrar nenhuma correspondência. O sistema 2D, no entanto, não é totalmente inútil. Recentemente a Toshiba mostrou na Automotive Engineering Expo, no Japão, um sistema de reconhecimento facial para detectar situações de distracção ou adormecimento ao volante[26]. Para além disso também pode ser utilizado para mudar a estação de rádio ou até mesmo ligar e desligar luzes.

Os sistemas 3D já são mais eficientes porque além das características detectadas pelo sistema 2D conseguem ter a percepção do relevo da face conseguindo obter a noção da profundidade das cavidades oculares, tamanho das bochechas, tamanho do nariz, etc.; tudo características que se mantêm praticamente inalteradas ao longo do tempo. Além do mais, a maior parte dos dispositivos de leitura não necessita de condições especiais de iluminação e a cara pode estar virada até 90º (vista de perfil) que a sua autenticação/identificação é conseguida com sucesso. A sua evolução tem sido notável ao ponto da identificação ser conseguida em menos de 1 segundo, mesmo entre gémeos idênticos[27].

2.2.4. Geometria da mão e dedos

Os sistemas biométricos que utilizam o padrão da mão como controlo de acessos normalmente são mais utilizados em escolas ou em empresas onde os níveis de segurança exigidos são mais baixos. O padrão da mão é utilizado na identificação de indivíduos, no entanto, este padrão não é tão único como a impressão digital ou a íris. Neste caso existe uma maior probabilidade de semelhança entre indivíduos diferentes.

O leitor é composto por uma fonte de luz e uma câmara digital que capta uma imagem da mão. O software utiliza a imagem para determinar o comprimento e largura da mão, espessura e curvatura dos dedos entre outros. Claro que uma das desvantagens deste sistema é que pode não funcionar tão bem quando a geometria normal da mão é alterada devido a ferimentos ou fracturas.

Figura 2.5 – Base de leitura da geometria da mão e dedos
HandKey II da empresa Schlage[28].

2.2.5. Impressão digital

Há mais de um século que a impressão digital é utilizada na identificação de pessoas. Assim que cientistas descobriram que esta era uma característica única de cada ser humano, a polícia percebeu que isto era algo que os poderia ajudar a desvendar crimes revelando os seus autores. É um método de identificação que é ainda utilizado e eficiente mas que hoje em dia tem também outras finalidades tais como obter acesso em determinadas portas, fazer *login* no computador pessoal, efectuar transferências bancárias, etc.

Este método de identificação tornou-se popular também pela facilidade com que qualquer pessoa deixa um rasto da sua impressão digital por onde passa. Quando agarramos ou tocamos em algum objecto a nossa impressão digital fica marcada nesse objecto devido a algo que não controlamos: o suor. O facto de por vezes tocarmos no cabelo, na cara, no nariz, vai transferir óleo que é segregado pela pele para os nossos dedos, o que aumenta a possibilidade de deixar a marca da impressão digital.

O padrão da impressão digital, tal como a íris, forma-se durante os primeiros meses de gestação do embrião e também não depende do DNA, sendo um padrão totalmente aleatório

mesmo entre gémeos idênticos. No entanto, os padrões existentes nas impressões digitais dividem-se em oito tipos[9]:

- *Arch* (arco): quando as linhas entram de um lado e flúem para o outro com uma pequena elevação ao centro;
- *Tented Arch* (arco em tenda): é semelhante ao primeiro mas com uma elevação mais acentuada e ao centro as linhas formam uma zona "pontiaguda" em forma de delta;
- *Ulnar Loop* (gancho ulnar): neste tipo de padrão as linhas fazem uma espécie de gancho em que a curva desta está direccionada para o dedo mindinho;
- *Radial Loop* (gancho radial): é o espelho do caso anterior, ou seja, neste caso o gancho está direccionado para o dedo polegar;
- *Plain Whorl* (espiral): tem a forma de uma espiral com duas formações em delta;
- *Double Loop Whorl* (espiral com duplo gancho): quando as linhas fazem um padrão em forma de espiral mas formado por duas formações de ganchos;
- *Central Pocket Loop Whorl* (espiral com centro): é uma espécie de formação em gancho com uma pequena espiral na ponta;
- *Accidental Whorl* (gancho combinado): é um padrão com dois ou mais deltas, e uma combinação de um ou mais tipos de padrão. Esta classificação também engloba todos os outros tipos de padrões não-usuais que não se enquadram nas outras classes.

Apesar de os tipos de padrões serem limitados, a verdade é que o seu tamanho, forma e número de linhas varia de pessoa para pessoa, sendo essas características que permitem distinguir as impressões digitais. Para além disso, também são analisadas pequenas características chamadas de *minutiae*, sendo estas[11]:

- *Ridge ending* (fim de linha): é o ponto onde uma linha acaba;
- *Bifurcation* (bifurcação): é o ponto onde uma linha se separa em duas;

- *Short Ridge* (linha curta): são linhas que têm um comprimento muito pequeno em comparação com as outras;
- *Island* (ilha): uma pequena linha dentro de uma linha curta;

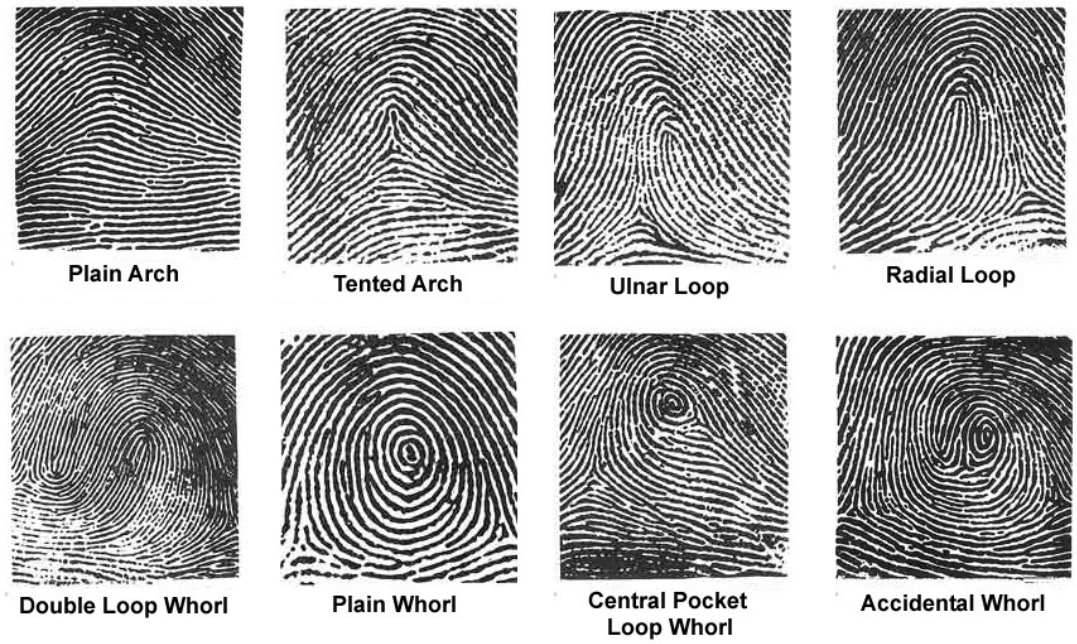
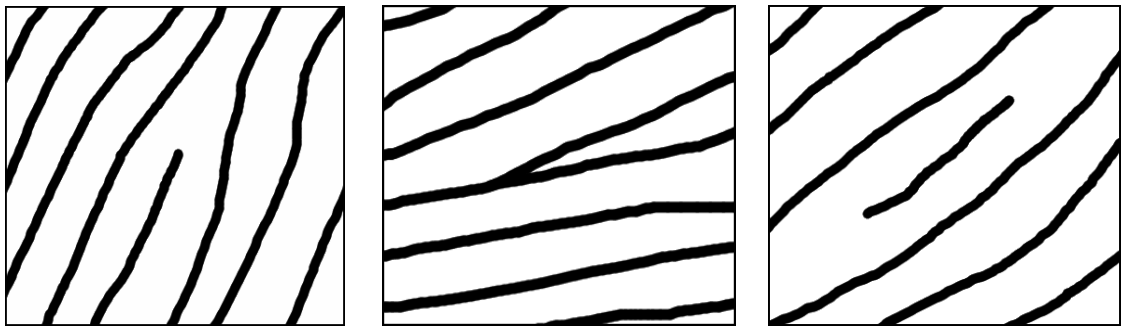


Figura 2.6 – Diferentes padrões de impressões digitais[9].

- *Ridge Enclosure* (linha enclausurada): uma bifurcação de linha que após um
- a curta distância se volta a reunir;
- *Spur* (pico): uma bifurcação onde uma das linhas é muito curta e a outra longa;
- *Bridge* (ponte): uma linha curta que corre entre duas linhas paralelas;
- *Delta* : uma união de linhas em forma de Y;
- *Core* (núcleo): uma curva em forma de U no padrão das linhas.



a) b) c)

Figura 2.7 – Alguns tipos de *minutiae*: a) Fim de linha; b) Bifurcação; c) Ponte.

É com base nestas características, a forma e distância entre elas, que é possível distinguir as várias impressões digitais, e é exactamente neste tipo de características que os leitores electrónicos se baseiam para identificar a pessoa.

2.2.5.1. Leitores de impressão digital

Não há muito tempo os leitores de impressão digital eram algo que só fazia parte dos filmes de ficção científica. No entanto, hoje é algo que se tornou completamente banal. Basta irmos a uma loja de produtos informáticos para podermos comprar um leitor para o nosso computador pessoal por menos de 50 Euros. Existem vários tipos de leitores de impressão digital, e os que mais se destacam são os leitores ópticos e os capacitivos.

Os leitores ópticos utilizam um CCD (Charge Coupled Device), o mesmo sensor de luz que é utilizado em máquinas fotográficas e de filmar, ou seja, basicamente este tipo de leitor não é mais que uma máquina fotográfica digital mas com um propósito diferente. Por debaixo da superfície onde é colocado o dedo existe uma fina camada de fósforo que, assim que é iluminada, realça as linhas da impressão digital (parte que está em contacto com a superfície de leitura); o sensor CCD tira uma imagem digital da impressão digital e faz um pequeno pré-processamento, rejeitando a imagem caso esta esteja muito escura, muito clara, ou com má definição. Caso isso aconteça o leitor ajusta os parâmetros de captura da imagem e tenta novamente até que esta esteja em condições e possa depois ser enviada para o software que irá fazer a leitura das *minutiaes* e a devida identificação. No entanto é necessário que a superfície de leitura esteja limpa e livre de riscos, pois isso pode dificultar uma boa leitura da

impressão digital, assim como também é necessário que o dedo não esteja sujo ou com algum tipo de marca.

Alguns destes leitores vêm equipados com algo a que se chama de "*live finger detector*", ou seja, um detector que permite ver através da temperatura emitida pelo dedo, se este se encontra anexado a um corpo humano. Pode parecer estranho mas não é de excluir casos de dedos cortados para obter acesso a algo vedado com um leitor biométrico (ver referência bibliográfica 13).

Os leitores capacitivos, em vez de utilizarem a luz para ler as linhas da impressão digital, utilizam uma pequena corrente eléctrica. Na superfície de um leitor capacitivo existem pequenas placas (mais pequenas que o tamanho de uma linha), que ao colocar-se o dedo nessa superfície, as linhas da impressão digital irão fazer uma ponte de contacto entre algumas placas e as zonas entre as linhas não. Desta forma, na zona onde existe contacto, a pele vai funcionar como o dieléctrico de um condensador, e na zona onde não existe contacto o dieléctrico é o ar, o que faz com que o valor de capacitância seja diferente. É com base nestas diferenças que se fará um "mapa" da impressão digital.

Dentro dos leitores capacitivos existem dois tipos: activo e passivo. Os leitores activos são parecidos com os passivos, mas precisam de um ciclo de carga onde aplicam uma dada voltagem à epiderme antes de se iniciar o cálculo da capacitância. Estes não trazem vantagens significativas sobre os leitores passivos e normalmente consomem mais energia que estes.

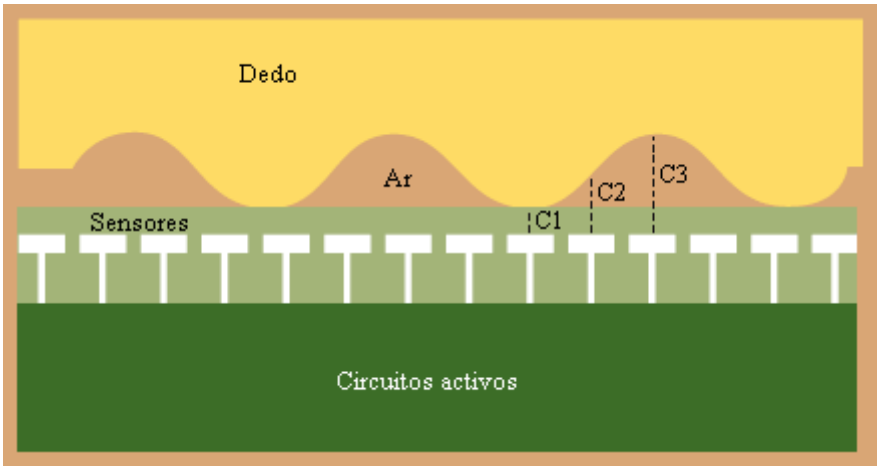


Figura 2.8 – Detalhe de um sensor de um leitor de impressão digital capacitivo[14].

A grande vantagem deste tipo de leitores é que trabalha sobre a forma real da impressão digital, ao contrário dos leitores ópticos que trabalham sobre uma imagem, tornando-se mais difícil de iludir o leitor. Além disso não precisa de uma superfície limpa e livre de riscos, e o facto de não precisar de um sensor CCD também torna este tipo de leitor mais compacto, menos dispendioso, e apresentando um consumo inferior aos leitores ópticos. No entanto, os leitores capacitivos são mais sensíveis à electricidade estática e ao desgaste.

3. Sistemas de controlo de acessos

Existem no mercado variados tipos de sistemas de controlo de acessos. De modo a perceber o que estaria em falta ou o que se poderia melhorar foi feita uma análise pelos actuais sistemas de controlo de acessos, procurando as características de cada um.

3.1. Access Control Scotland

A Access Control Scotland[15] assume-se como líder na Escócia na área de sistemas de controlo de acessos equipados com leitores de smart cards e leitores biométricos. A tecnologia utilizada é de última geração e está disponível ao alcance de qualquer empresa, desde uma com menos de 250 empregados até ao nível de grandes multinacionais, oferecendo total controlo sobre quem tem acesso em qualquer porta, a qualquer hora do dia.

Nos acessos com smart cards, estes colocam à disposição uma série de cartões com diferentes tamanhos e feitios. Também colocam à disposição, em alternativa aos cartões, pequenos porta-chaves apetrechados com etiquetas RFID que podem ser usados pelo mesmo sistema. O único senão destes porta-chaves é que, sendo mais pequenos, a distância ao leitor terá que ser menor para que este consiga proceder à sua autenticação.

Nos pontos de acesso mais críticos, é normal a utilização de leitores biométricos juntamente com os leitores de smart-cards. Quando o utilizador desejar abrir a porta passa em primeiro lugar com o cartão pelo leitor RFID e em seguida coloca o seu dedo no leitor biométrico de forma a garantir que a pessoa que tem o cartão é realmente a pessoa a quem esse cartão pertence (situação de autenticação que foi vista na introdução deste trabalho). Os leitores biométricos utilizados são do tipo capacitivo pelo que não estão sujeitos a erros de

leitura devido à presença de sujidade ou marcas na impressão digital, visto que a sua leitura é determinada pela camada de pele inferior à epiderme, a derme.

Figura 3.1 – Porta chaves com etiqueta RFID utilizado em sistemas de controlo de acessos[15].

Todo o equipamento é configurado e supervisionado através de um software que se encontra instalado num computador central. Neste programa está registado o perfil de cada utilizador e é nele que são configurados os pontos onde este tem acesso e a que horas o acesso é permitido. Todos os acessos, autorizados ou não, são registados de modo a que se possa ter um registo de todo os movimentos por parte de cada utilizador. As principais características do software são:

- Criação de um perfil para cada empregado/utilizador;
- Criação de relatórios com os acessos autorizados e negados em dado ponto a dada hora;
- Acesso limitado de acordo com o estatuto da pessoa;
- Controlo de *anti-passback*, garantindo que um cartão apenas é utilizado por um empregado de cada vez;
- Visualização em tempo-real, mostrando as portas que estão a ser abertas, os alarmes e entradas forçadas;
- Alarmes disparados automaticamente se uma entrada forçada for detectada.

Também existe outro tipo de equipamentos, para além dos leitores e actuadores, que pode ser interligado com a rede de segurança tais como botões de alarme e sirenes que podem ser actuados a partir do software. Este pode ser programado para abrir as portas automaticamente em caso de alarme.

3.2. Amano – Christy Industries

Os sistemas de controlo de acessos oferecidos Christy Industries[16] são bastante semelhantes aos da Access Control Scotland, no entanto o seu software tem outras características interessantes tais como:

- Aplicação integrada para a criação de cartões de identificação;
- Sistema integrado de gravação de vídeo digital (DVR – Digital Vídeo Recorder);
- Aplicação criada em JAVA o que permite correr em diversos sistemas operativos;
- Suporte multilingue;
- Compatibilidade com SQL Server e Oracle.

Esta empresa oferece outras soluções de software, à parte do software de controlo de acessos, e de hardware para monitorizar o cumprimento de horários por parte dos empregados.

3.3. Card Access 3000 – Continental Access

O Card Access 3000 (AC3000) da Continental Access[17] é mais um excelente software de controlo de acessos capaz de oferecer um vasto leque de ferramentas integradas de modo a garantir a segurança do local onde este possa ser instalado.

Uma das suas características é a compatibilidade com câmaras de vigilância VDR, NVR (Network Vídeo Recorders) ou CCTV (Closed-Circuit Television). Com o AC3000 o utilizador tem à sua disposição diversas opções tais como:

- Associar uma câmara a um alarme, e sempre que o alarme for accionado, a imagem é guardada;
- Iniciar gravação de vídeo quando um alarme dispara;
- Iniciar gravação de vídeo quando é detectado movimento na área abrangida pelo "campo de visão" da câmara;
- Visualizar várias câmaras simultaneamente;
- Suporte de câmaras com controlo PTZ (Pan, Tilt and Zoom);
- Procura de gravações de vídeo;
- Monitorização de vídeo protegida por password.

No que toca a outros aspectos do programa de controlo de acessos podem-se destacar os seguintes:

- 5 níveis de ameaça, o que permite ao administrador alterar o nível em situações de perigo ou de ameaça, desactivando privilégios de acesso por identificação individual ou de grupo;
- Registo de acessos e alarmes;
- Registo de cumprimento/incumprimento do horário de trabalho;
- Notificação por e-mail e SMS;
- Criação de relatórios de acessos/alarmes detalhados-

3.4. Honeywell NStar – Bass Home Security Systems

O sistema de controlo de acessos Honeywell NStar[18] é outro exemplo de integração de soluções de hardware e software. As suas controladoras utilizam leitores RFID, suportam até 2 portas, 10000 utilizadores e 100000 eventos. Quanto ao software, preparado para trabalhar com as controladoras NStar, permite a configuração dos acessos, visualização das

actividades em tempo-real, notificação de alarmes, e integração de vídeo entre outros. Em seguida apresentam-se as características principais deste sistema de segurança:

- Até 10000 utilizadores;
- Permissão a convidados;
- 32 períodos de férias com várias definições;
- Visualização de eventos;
- Activação/Desactivação de cartão por dia/mês/ano;
- Atribuição de múltiplos cartões a uma mesma pessoa;
- Agendamento de backup da base de dados;
- Criação de relatórios de eventos com base na hora, data, leitor, empregado, tipo de evento;
- Criação de relatórios de administrador com base na hora, data, qual o administrador que efectuou as alterações e quais as operações efectuadas.

O sistema de segurança NStar é considerado especialmente indicado para pequenas aplicações. No entanto, está preparado para sofrer expansões e upgrades de modo a estar a altura das exigências de qualquer tipo de edifício.

3.5. Presidio – Synergistics

O software de controlo de acessos Presídio[19] da Synergistics que é alojado num servidor HTTP sendo acedido através de qualquer *browser* de Internet, sem ser necessário instalar ou carregar o software localmente, descartando assim a necessidade de um servidor ou de uma base de dados central. Qualquer computador com acesso à Internet consegue configurar os acessos deste sistema. De modo a dar acesso apenas a pessoas autorizadas, são utilizados protocolos de comunicação segura e encriptação SSL de 128 bits.

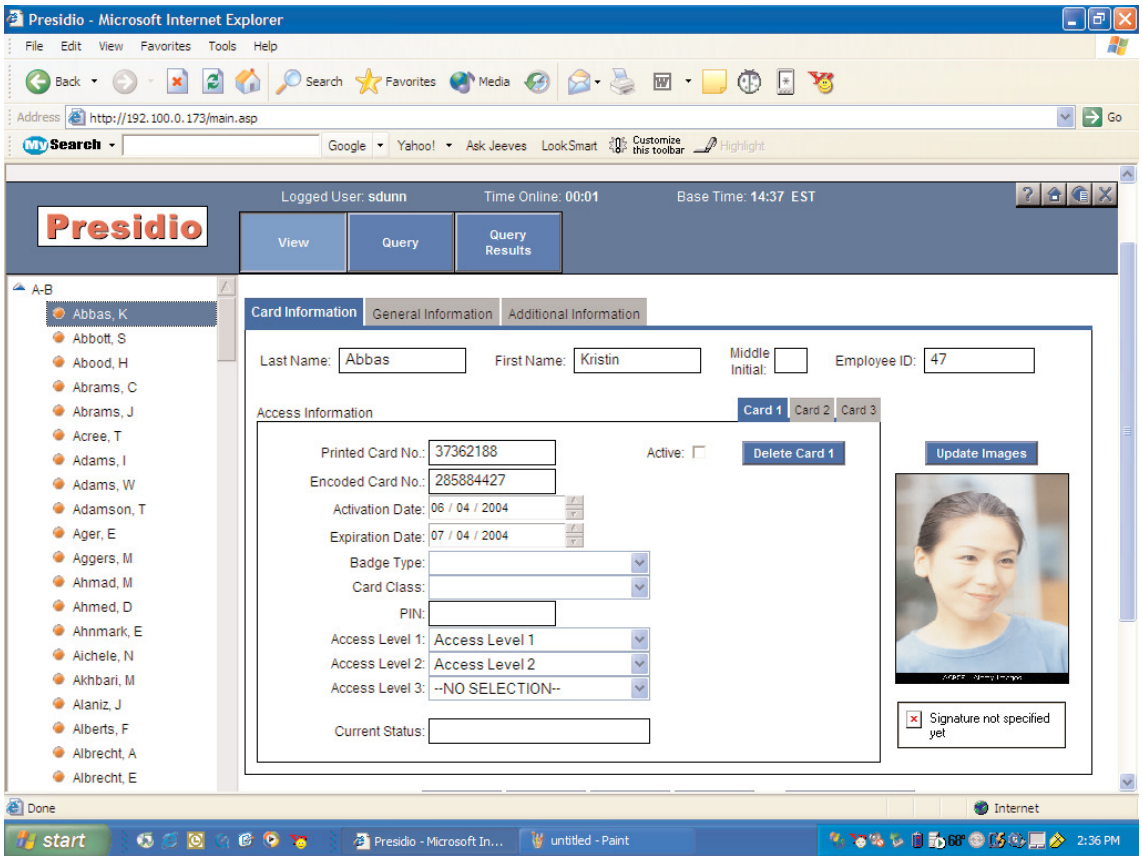


Figura 3.2 – Configuração de uma controladora utilizando o software Presídio[20].

Em seguida apresentam-se as principais características do software Presidio:

- Suporte para 10000 utilizadores;
- 255 níveis de acesso (3 para cada utilizador);
- Exportação e importação de configurações;
- Criação de vários tipos de relatórios;
- Reencaminhamento de eventos;
- Controlo de elevadores;
- Período de férias configuráveis;

- 3 níveis de operador (assim que é introduzido o nome de utilizador e a palavra-passe o utilizador é redireccionado para uma dada página consoante o seu nível de operador, i.e., administrador, operador ou guarda);
- Arquivo automático de dados;
- Anti-passback;
- Monitorização de eventos em tempo-real;
- Notificação por email/WAP;
- Suporte para leitores biométricos.

3.6. WAPAC – Synergistics

O WAPAC[21] é outra solução da Synergistics que é baseada numa arquitectura cliente-servidor, oferecendo todos os benefícios de um sistema de última geração com a complexidade reduzida ao mínimo. Ao contrário da solução anterior, esta utiliza um computador central onde irá estar instalado o software responsável pela configuração das controladoras. A flexibilidade é o ponto mais forte deste software tornando-o num sistema de controlo de acessos tanto para um edifício como para um complexo de edifícios. Muitos sistemas WAPAC podem até estar separados por vários quilómetros, sendo possível controlar o servidor central através de máquinas ligadas a este remotamente. Além disso é capaz de suportar diversos tipos de leitores: de banda magnética, RFID, teclados numéricos ou leitores biométricos. Em seguida mostram-se as principais características do sistema WAPAC:

- Compatível com Windows NT, 2000 e XP;
- Processamento distribuído. As decisões de acesso são feitas localmente (na controladora);
- Suporta até 80000 cartões por controladora;
- Múltiplos códigos de acesso por utilizador;

- Sincronismo automático com a base de dados;
- Backup automático;
- Arquivamento automático;
- Notificação de eventos em tempo-real;
- Opção para verificar quem se encontra ainda dentro do edifício;
- Diferentes níveis de utilizador;
- Activação/Desactivação temporária de cartão;
- Acesso com horário definido;
- Acesso temporário de cartão ou grupo de cartões;
- Anti-passback local;
- Autorização global de abertura/fecho de portas;
- Número ilimitado de andares/planos por sistema;
- Edição de campos da base de dados;
- Permite comentar os alarmes ocorridos;
- Controlo de elevadores;
- Visualização da planta do andar com os pontos de acesso.

3.7. Starwatch iTDC PRO II – IDTECK

A IDTECK é uma das empresas líderes mundiais no que se refere à produção de sistemas de segurança para edifícios, industria ou casas particulares. Todo o tipo de tecnologia tal como leitores biométricos, RFID, Smart Cards, controladoras e o software de controlo de acessos é produzida por esta empresa que já conta com 20 anos de experiência na área da segurança empresarial e pessoal.

Esta mesma empresa apresenta várias soluções de software, cada uma preparada para funcionar com uma dada família de controladoras e com ligeiras diferenças entre elas (consultar Anexo I). Como já existia alguma familiaridade com a solução iTDC PRO II, que está preparada para funcionar com as controladoras utilizadas para o desenvolvimento deste trabalho, fez todo o sentido escolher este software como objecto de análise.

O Starwatch iTDC PRO II[22] é compatível com as últimas plataformas Windows base NT, suporta comunicação TCP-IP e série (RS-232) com as controladoras, e vem disponível em várias línguas, apresentando também a possibilidade de edição de modo a dar ao utilizador a possibilidade de traduzir para uma língua que não esteja disponível. Eis algumas das suas principais características:

- Monitorização de eventos, alarmes, último acesso e sua identificação e localização no mapa. Tudo em tempo-real;
- Configuração de mapa/planta do edifício de modo a mostrar através de ícones as portas onde se dão os acessos;
- Visualização/Gravação de vídeo nos momentos de acesso ou alarme;
- Configuração de mapa/planta do edifício de modo a mostrar através de ícones a localização de câmaras de vigilância;
- Controlo de acessos por cartão, grupo ou horário;
- Activação/Desactivação de cartão;
- Possibilidade de atribuir uma data de validade a um cartão;
- Anti-passback;
- Menu administrativo protegido por password;
- Exportação de relatórios de eventos para formato Excel, HTML e PDF;
- Envio de eventos por SMS;
- Pesquisa de movimentos de uma dada pessoa;
- Registo de horas de trabalho, faltas, horas extraordinárias por utilizador;

- Definição de férias por utilizador;
- Controlo de fechadura de uma dada porta ou as fechaduras de uma área ou andar;
- Controlo de alarmes, de modo individual, por área, ou andar;
- Possibilidade de configurar a abertura de portas ao sinal de alarme de fogo;
- Configuração de ronda de vigia do guarda de segurança;
- Configuração de visitantes; registo de movimentos e de dados pessoais, configuração do dia de visita e remoção automática.

3.8. C•Cure 9000 – Software House

O C•Cure 9000[23] da Software House é uma poderosa ferramenta de controlo de acessos que foi desenhada de raiz utilizando o Framework 3.5 da Microsoft, e tem também uma arquitectura baseada no modelo servidor-cliente. Em cada zona de segurança existe um servidor que é responsável pela programação das controladoras podendo o acesso a este ser feito a partir de qualquer máquina com o programa cliente. A base de dados é única e é acedida por todos os servidores existentes no sistema. Uma particularidade deste tipo de arquitectura é que permite fazer a instalação nas máquinas-cliente a partir do servidor, eliminando a necessidade de fazer upgrade em cada cliente separadamente.

Em seguida apresentam-se as principais características do C•Cure 9000:

- Área de visualização configurável, de modo a observar vários tipos de informação tal como acessos, configuração ou vídeos em tempo-real;
- Suporte de mapas de edifícios através de ficheiros CAD ou imagens (BMP, JPEG, etc.). Permite também adicionar pontos de acesso e câmaras de vigilância em determinados pontos do mapa;
- Partição da base de dados. Se uma base de dados for partilhada por várias empresas esta característica permite definir quais as partições que são partilhadas por todas;
- Suporta SQL Server e Oracle;

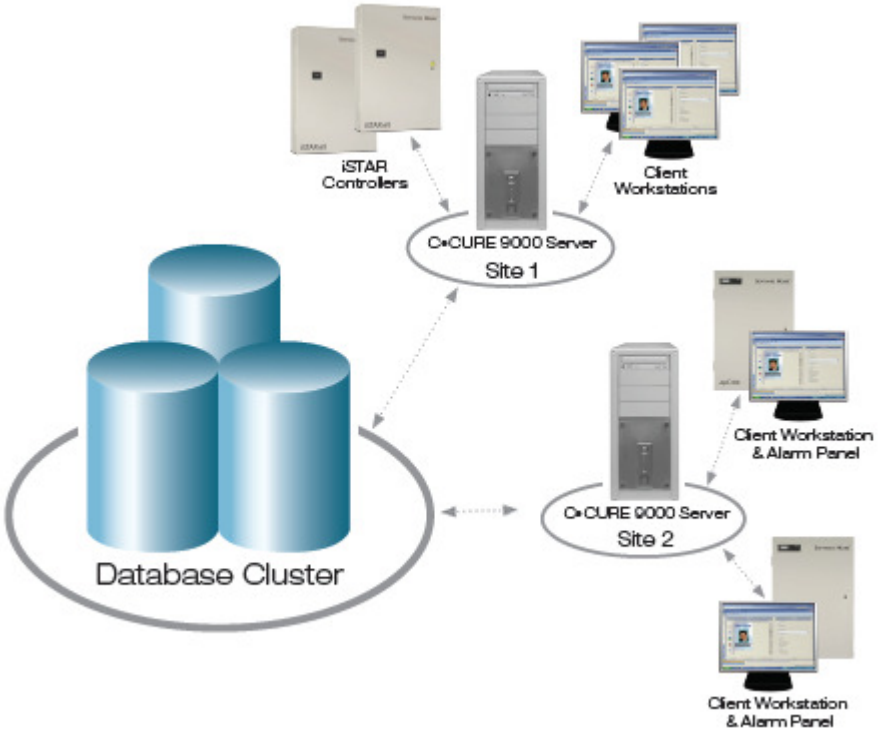


Figura 3.3 – Diagrama do sistema.

- Utilização de métodos de encriptação de modo a criar uma ligação segura entre o servidor e cliente;
- A sua arquitectura distribuída permite uma comunicação mais robusta e eficiente através de uma WAN tornando mais fácil a configuração do sistema e tornando-o mais flexível em caso de expansão do local de vigilância;
- Permite fácil migração a partir de versões anteriores;
- Inclui uma pequena ferramenta para a criação de cartões de identificação.

3.9. Access Enterprise – Tensor PLC

O Tensor Access Enterprise[24] é a última e mais evoluída ferramenta de controlo de acessos, tendo sido desenhada especificamente para grandes empresas. Esta funciona de maneira um pouco diferente das anteriores. Existe um computador central onde é feita a

configuração dos empregados e o controlo dos acessos, onde este por sua vez está ligado a várias *clocking stations*, que estão colocadas normalmente à entrada do edifício ou andar, e que controla o número de horas de trabalho de cada empregado. Ligada à *clocking station* estão as controladoras das portas e os respectivos leitores. Cada controladora suporta até 2 leitores (de proximidade ou de impressão digital), cada *clocking station* suporta até 12 controladoras e o servidor central suporta até 31 *clocking stations*, o que perfaz um total de 372 pontos de acessos. O número de utilizadores é ilimitado. Pode parecer muito mas quando comparado com o sistema da IDTECK, fica muito aquém (o sistema da IDTECK suporta até 999 controladoras onde cada uma pode controlar até 4 portas, ou seja, 8 leitores, o que no total dá 3996 pontos de acesso).

Eis as principais características do sistema da Tensor PLC:

- Perfiz de controlo de acesso;
- Janela de eventos mostra todas as portas que foram abertas/fechadas e quais as que foram sujeitas a abertura forçada praticamente em tempo-real;
- As portas podem ser configuradas para abrir automaticamente mediante alarme de fogo, e fechar automaticamente em caso de roubo;
- Anti-passback;
- Suporte para câmaras CCTV com a possibilidade de gravar situações de entradas forçadas, alarmes, acessos negados ou até mesmo acessos autorizados;
- O acesso pode ser alterado consoante o estatuto da pessoa;
- O sistema continua a funcionar mesmo que o servidor central deixe de funcionar;
- Sensores para detecção de entrada forçada;
- Mostra quem e onde teve acesso.

3.10. Comparação

Como se pode observar pela descrição das diversas aplicações de controlo de acessos, existem algumas características que são gerais e outras que as distinguem entre si. No entanto, não existe uma única aplicação que consiga abranger todas as funcionalidades. Alguns dos requisitos podem não ser suportados por limitação do hardware que faz parte do sistema de segurança, outros ou não vêm especificados ou simplesmente não fizezam parte dos planos na altura de implementação do software.

A tabela colocada em anexo (Anexo 2) exibe uma lista de características comuns nas aplicações de controlo de acessos e, dentro das aplicações analisadas, quais as que cumprem cada uma dessas características.

4. Funcionamento dos sistemas de controlo de acessos

A maioria dos sistemas de controlo de acessos são compostos por uma parte de software e outra de hardware. O hardware é a parte que interage com o ser humano, ou seja, o que lê os cartões, destranca ou tranca as portas, despoleta alarmes, etc. O software serve para gerir os utilizadores, controlar horas de trabalho, registar horas de entrada e saída, registar situações de alarme, e claro, configurar o hardware de modo a que este saiba quando deve ou não conceder acesso.

4.1. Controladora (hardware)

Nos sistemas de controlo de acessos o controlo das portas é efectuado com base em dispositivos, designados por controladoras, que não são mais que pequenos computadores munidos de alguma inteligência. De modo a termos alguma referência, e visto estarmos também familiarizados com este modelo, decidiu-se utilizar uma controladora da empresa IDTeck (modelo Star iTDC) para mostrar quais as principais características[29] de uma controladora de acessos:

• **Modo de funcionamento** – Este modelo permite controlar 2 a 4 portas. A controladora recebe do leitor de proximidade o número de identificação do cartão e verifica se deve ou não destrancar a porta. Quando é sentido algum sinal de entrada, de um sensor ou de um botão de saída de emergência, a controladora gera e guarda o evento. Todos os eventos são guardados em memória e posteriormente enviados para o computador central. Cada controladora é independente pelo que, em caso de avaria, não afectará as outras

controladoras mesmo quando estas estão todas ligadas em série (a explicação da ligação em série encontra-se no ponto seguinte).

• **Interligação entre várias controladoras** – Para edifícios com diversos pontos de controlo (portas) serão necessárias várias controladoras e todas estas devem estar ligadas ao mesmo computador central. Uma forma de ligar as diversas controladoras é todas elas ligarem por TCP-IP a um *switch*, onde a diferenciação entre elas é feita através do seu endereço IP, que será único dentro da rede. Outra forma possível é fazer uma ligação em série, onde apenas uma controladora liga ao computador central por TCP-IP e a controladora seguinte liga a esta através de uma linha de comunicação RS-422. A terceira controladora liga-se a esta última do mesmo modo e por aí em diante até à última controladora. A este tipo de ligação chama-se ligação em série, e neste caso a diferenciação entre as diversas controladoras é feita através de um endereço que é modificado num pequeno conjunto de interruptores (8-bit DIP Switch) que está na *board* da controladora.

• **Interacção com o computador central** – Todos os eventos ocorridos podem ser analisados a partir do computador central, visto que todos os eventos decorridos na controladora são enviados para este através do protocolo de comunicação (RS-232 ou TCP-IP). No caso de utilização da comunicação TCP-IP é necessário instalar um módulo que é vendido separadamente. Esta ligação é bidireccional, visto que o computador central faz o envio para a controladora dos dados para a sua configuração, e esta envia para o computador central todos os registos de eventos ocorridos naquele local onde a controladora se encontra instalada. Mesmo que haja alguma falha no computador central a controladora armazena os eventos em memória e quando a ligação é novamente estabelecida é feito o envio dos eventos que entretanto ocorreram. Também é possível exportar relatórios de eventos de determinados utilizadores.

• **Memória não-volátil** – Se ocorrer alguma falha de energia todos os eventos e informação de utilizadores autorizados não serão apagados porque a controladora usa memória flash.

• **Teclado e display LCD** – Se a controladora não estiver ligada a um computador os módulos de teclado e display LCD podem ser utilizados na sua configuração.

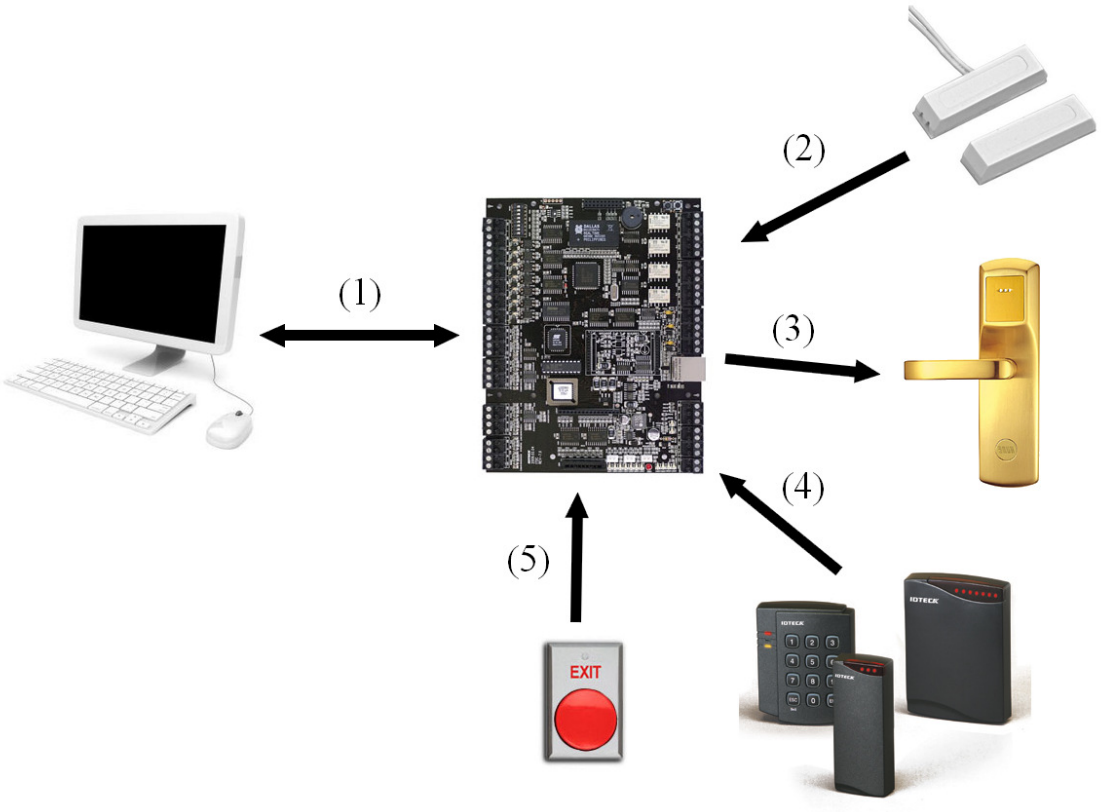


Figura 4.1 – Esquema simples de ligações com a controladora. (1) Ligação bidireccional entre o computador. (2) Ligação unidireccional aos sensores magnéticos das portas. (3) Ligação unidireccional para controlo da fechadura. (4) Ligação aos leitores. (5) Ligação ao botão de saída de emergência.

• **Anti-Pass-Back** – Utilizando 2 leitores de proximidade para cada porta é possível utilizar o método de Anti-Pass-Back (APB). Quando activado previne que qualquer utilizador passe duas vezes na mesma entrada ou na mesma saída sem que tenha feito o percurso inverso. Ou seja, se um utilizador passou o seu cartão no leitor de entrada, abriu e fechou a porta mas não passou para o outro lado, quando passar o cartão novamente

esta já não vai abrir. A regra de APB obriga a que seja cumprida a regra de "uma entrada, uma saída".

• **Número de portas sob controlo** – Este modelo (Star iTDC da IDTeck) de controladora suporta até 4 portas com 4 leitores. No caso de controlar apenas 2 portas, os leitores 1 e 2 ficaram ligados à porta 1 e os leitores 3 e 4 à porta 2. Neste caso o Anti-Pass-Back pode ser aplicado. Para se usar mais portas por controladora é necessário um módulo de expansão mas, no caso de 4 portas, cada porta terá apenas um leitor pelo que a regra de APB já não é possível aplicar.

• **Entradas e saídas (sensores e actuadores)** – Cada controladora tem 4 saídas para actuadores que são accionados através de relés que permitem destrancar a fechadura das duas portas e activar os dois alarmes associados a cada porta. Em termos de entradas para os sensores existem também 4 (mais 3 TTL), onde são ligados os sensores magnéticos e os botões de saída de emergência. Os sensores magnéticos são sensores que ficam na porta e que indicam ao sistema quando é que esta se encontra fechada ou aberta, permitindo detectar aberturas forçadas. Além disso também permite detectar quando uma porta fica demasiado tempo aberta após a abertura, alertando o computador central com um aviso.
Além disso, existem as entradas para a ligação dos leitores dos cartões. Estes leitores lêem o código que é característico de cada cartão, envia-o para a controladora através do protocolo RS-232 ou Wiegand e esta verifica se este cartão se encontra nos seus registos. Se se encontrar a autorização de passagem será concedida e a porta correspondente será destrancada. Os identificadores podem ser leitores RFID com ou sem teclado numérico (para introduzir uma password em conjunto com a passagem do cartão), leitores biométricos de impressão digital ou de reconhecimento facial.

4.2. Software

Tendo em conta que o problema apresentado para a elaboração deste trabalho partiu do facto de o software de controlo de acessos da IDTeck, o Starwatch iTDC PRO, não conseguir responder às necessidades do cliente final (neste caso o Gabinete Nacional de Segurança), ir-se-á fazer uma pequena análise deste, desde a sua estrutura, passando pela configuração do sistema até à fase final de funcionamento com todo o sistema de segurança operacional.

4.2.1. Estrutura

Em seguida ir-se-á analisar detalhadamente todas as características do software Starwatch de modo a compreender melhor a sua organização/estrutura desta aplicação e de modo a que se possa mais à frente fazer comparações em relação ao software desenvolvido.

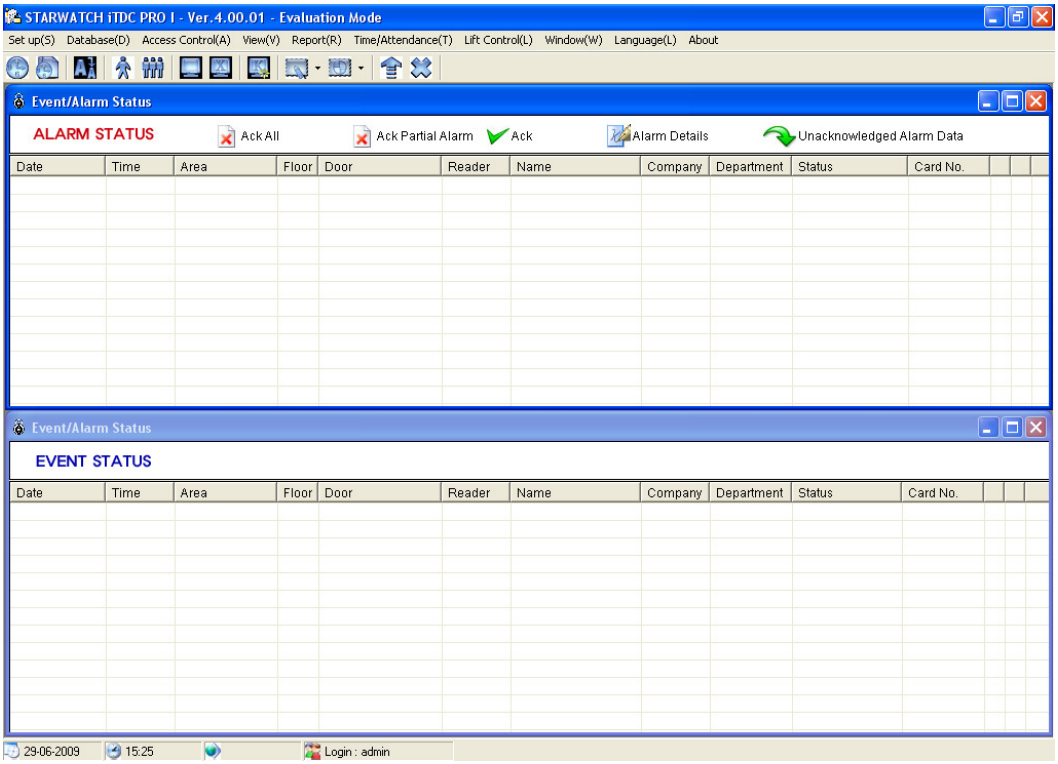


Figura 4.2 – Janela principal do programa de controlo de acessos Starwatch iTDC PRO.

A aplicação de segurança Startwatch iTDC PRO está dividida em 4 partes. A parte superior consiste numa série de menus que dão acesso a diversas opções de configuração do programa, existindo por debaixo destes menus um conjunto de botões de atalho que dão acesso directo aos menus de configuração mais comuns. Na zona central aparecem duas janelas, onde numa aparecem todos os alarmes e na outra todos os eventos (ditos normais) reportados pelas controladoras para o computador central. Nestas janelas em forma de lista podemos ver o dia, a hora, o local, e a pessoa que gerou tal evento ou alarme, ficando todos estes eventos registados para que se possa mais tarde exportar em forma de relatório.

• Configurações (Set-Up)

Neste menu podemos configurar tudo o que diz respeito à comunicação com as controladoras, definição de portas e leitores, anti-pass-back, configuração da planta do edifício, configuração de notificações por SMS, e configurações gerais do programa.

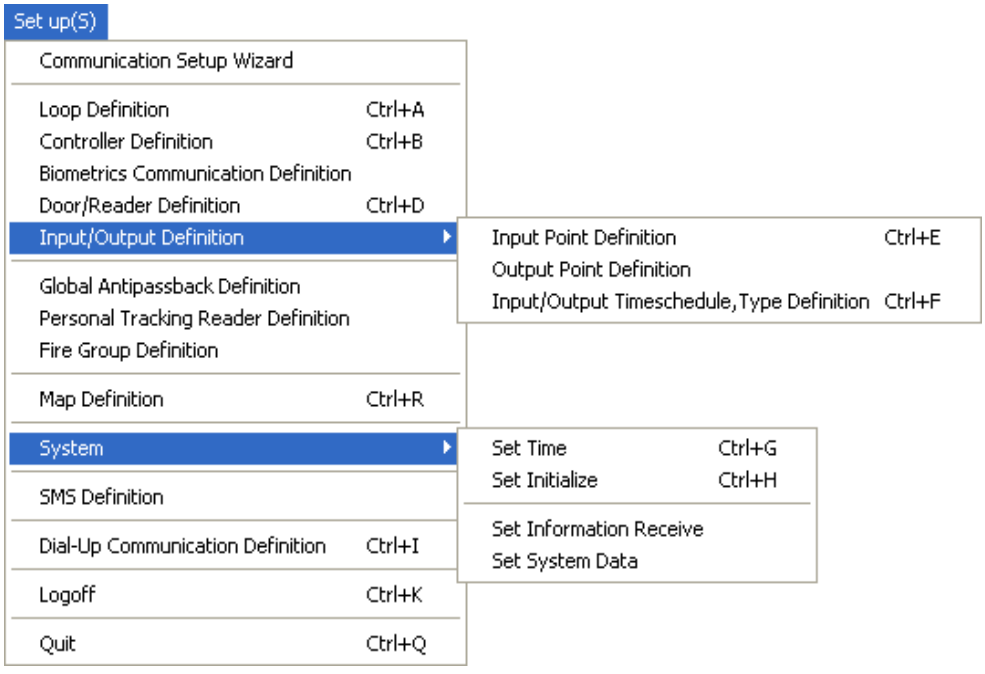


Figura 4.3 – Menu de configurações (Set-Up).

• Base de dados (Database)

Aqui para além de podermos alterar algumas opções que dizem respeito à gestão da base de dados do sistema tais como criar backups e restaurar dados anteriormente guardados, podemos também alterar outro tipo de informações tais como o nome da empresa ou empresas que estão sob a alçada do sistema de segurança, quais os diversos departamentos existentes, definir áreas de acesso constituídas por um ou mais andares, entre outros.

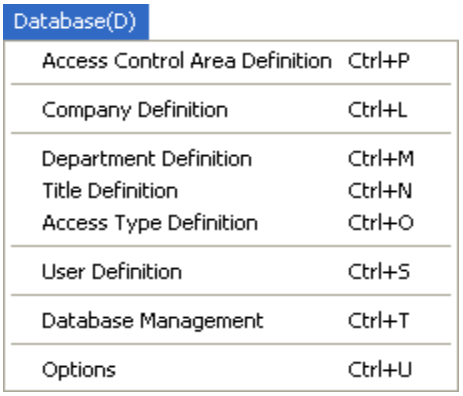


Figura 4.4 – Menu da base de dados (Database).

• Controlo de acessos (Access Control)

É no menu de controlo de acessos que se faz o registo dos utilizadores ou grupo de utilizadores, onde se definem as regras de acesso, quais as portas onde cada um tem autorização de acesso, a que horas e a que dias. Também existe uma opção para a criação dos cartões de identificação, cartões de visitantes, cartão do guarda e definição das zonas de vigia.

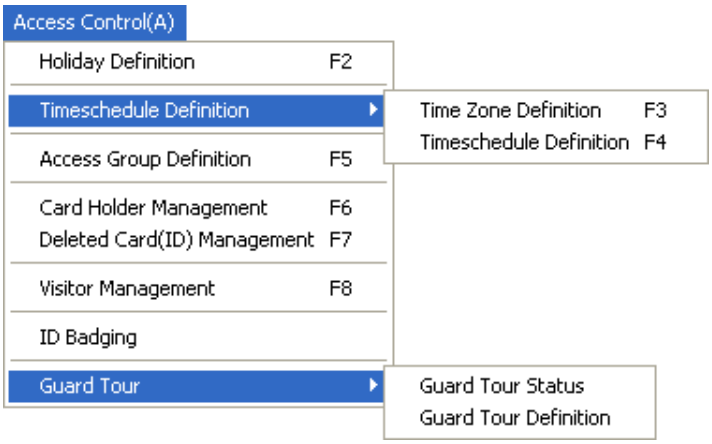


Figura 4.5 – Menu de controlo de acessos (Access Control).

• Ver (View)

O menu ver permite mostrar detalhes de acontecimentos dentro do sistema tais como os eventos de entradas e saídas, alarmes, detalhes da comunicação com as controladoras, configuração das câmaras de vídeo, e também controlo das portas, alarmes, botões de emergência e sensores magnéticos.

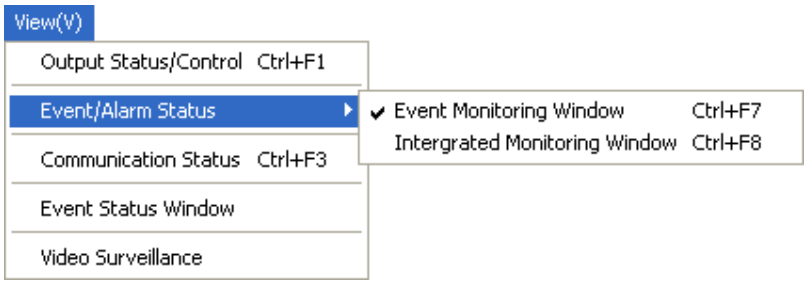


Figura 4.6 – Menu ver (View).

• Criação de relatórios (Report)

Neste menu são apresentadas diversas opções para a criação de relatórios, desde relatórios por cartão, por evento, alarme, porta, visitante, últimos acessos, notificações de SMS e vigia do segurança.

Figura 4.7 – Menu de criação de relatórios (Report).

• Criação de horários (Time/Attendance)

Como por vezes nem todos os trabalhadores de uma empresa têm o mesmo horário de trabalho aqui poderá definir diversos horários, quais os dias de descanso, e criação de relatórios para verificação de cumprimento/incumprimento dos horários estabelecidos.

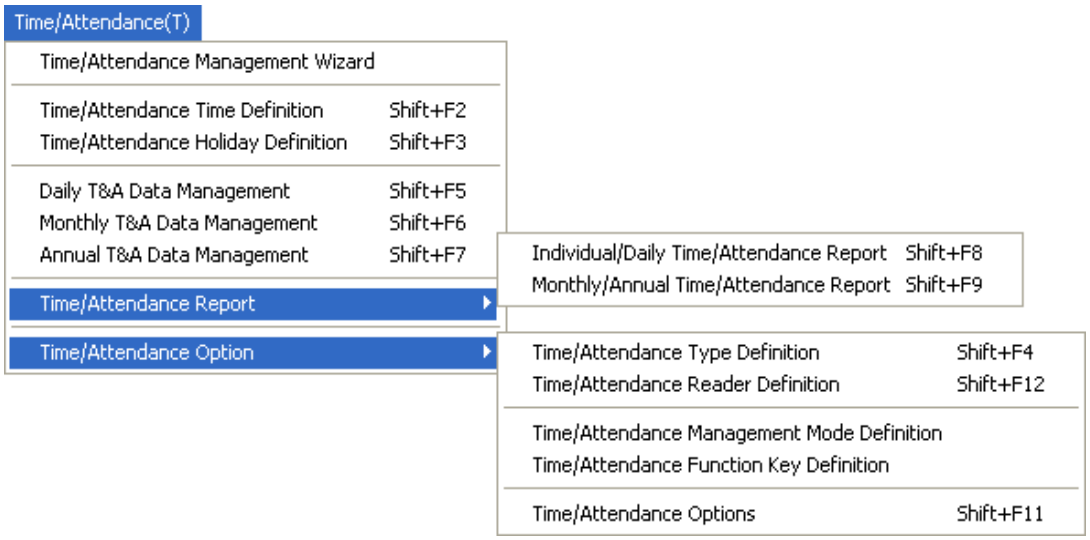


Figura 4.8 – Menu de criação de horários (Time/Attendance).

• Controlo de elevadores (Lift Control)

O acesso aos diversos andares de um edifício também é algo que se deve ter em conta. O acesso ao elevador é feito com uma controladora que apenas chama o elevador se a pessoa tiver privilégio para tal.

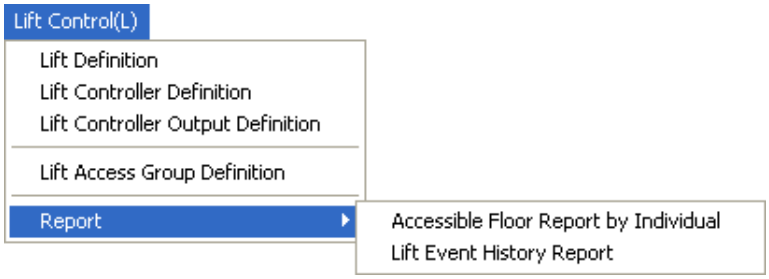


Figura 4.9 – Menu de controlo de elevadores (Lift Control).

• Posição das janelas (Window)

Este menu serve apenas para dispor as janelas que se encontram abertas dentro do programa em disposição vertical ou horizontal.

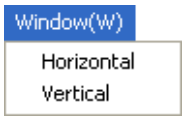


Figura 4.10 – Menu de posicionamento de janelas (Window).

• Língua (Language)

O Starwatch iTDC PRO vem com duas línguas predefinidas: Inglês e Coreano. No entanto, é dada a possibilidade ao utilizador de definir a sua língua, bastando para isso fazer a tradução de todas palavras ou frases que compõem o programa (utilizando o *Language Converter*).

Figura 4.11 – Menu de escolha de língua (Language).

• **Sobre (About)**

Este não tem qualquer utilidade, apenas serve para informar o utilizador qual a versão do programa, qual a empresa que o criou, os contactos para suporte, etc.

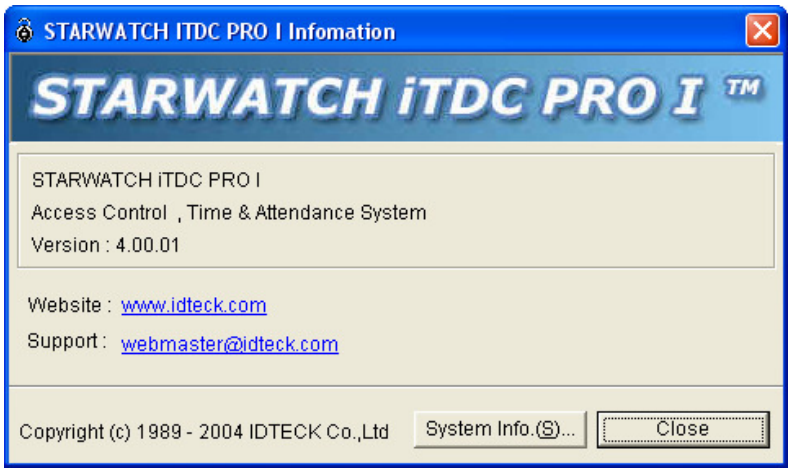


Figura 4.12 – Informações do programa (About).

4.2.2. Configuração do edifício

De modo a fazer uma comparação do software Starwatch com o software desenvolvido, ir-se-á demonstrar como é feita a configuração do edifício, ou seja, como dizer ao sistema quantas portas, leitores, controladoras existem e qual a sua localização.

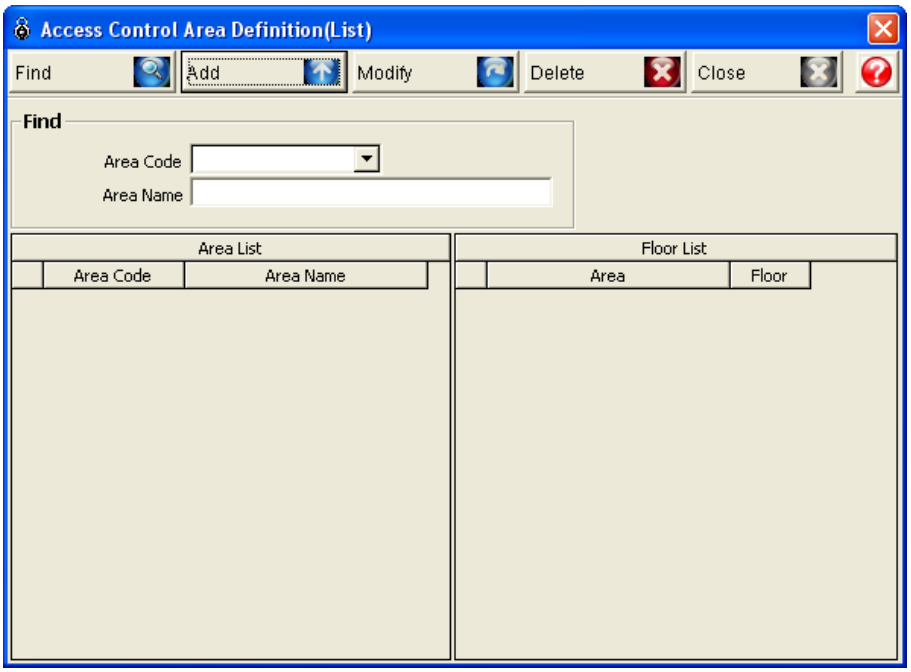


Figura 4.13 – Janela de lista de áreas/andares.

O primeiro passo será definir as áreas ou andares do edifício. No menu *Database* acede-se à opção *Access Control Area Definition*, o que fará aparecer a janela que está representada na Figura 4.13. Nesta janela aparece uma lista com todas as área definidas, e como se pode ver, neste momento não existe nenhuma. Carregando no botão *Add* irá aparecer uma nova janela para adicionar uma nova área.

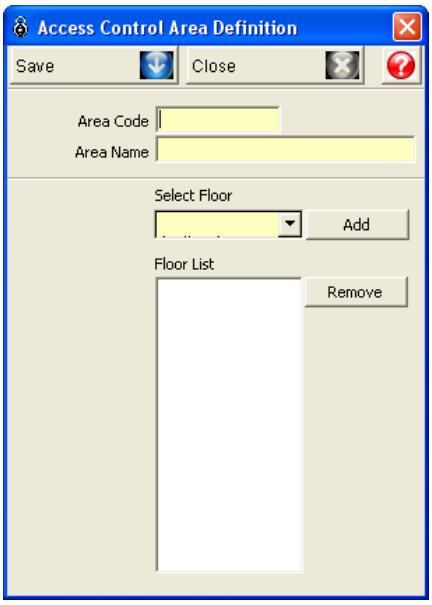


Figura 4.14 – Definição de áreas.

Nesta janela atribuímos um código que vai identificar a área, e um nome para a área. Podemos atribuir o código 000 e o nome "Rés-do-Chão", seleccionando na lista de andares (que estão pré-definidos) o andar GF (de *Ground Floor*). Se desejarmos adicionar mais áreas no mesmo andar ou em andares diferentes basta carregar novamente no botão *Add*. A janela de lista de áreas permite também procurar por uma determinada área (através do seu código ou nome), modificar ou apagar uma área já existente.

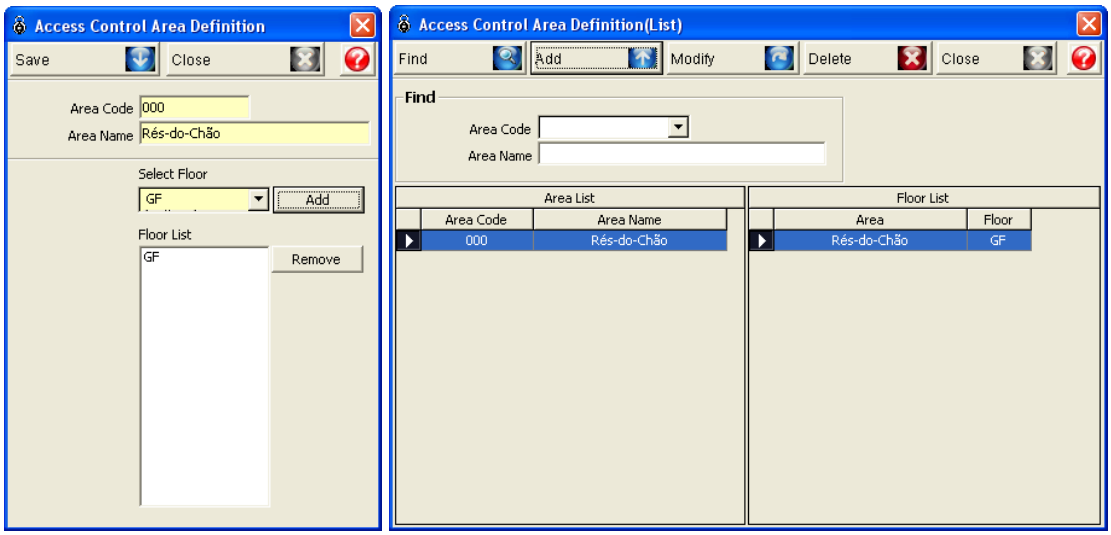


Figura 4.15 – Introdução de uma nova área.

4.2.3. Configuração das ligações com as controladoras

Em seguida, com as áreas do edifício já definidas, é necessário configurar a ligação com as controladoras que já se encontram instaladas. O primeiro passo é definir as ligações, ou *Loops*, através da opção *Loop Definition* no menu *Set-Up* que irá abrir uma lista parecida

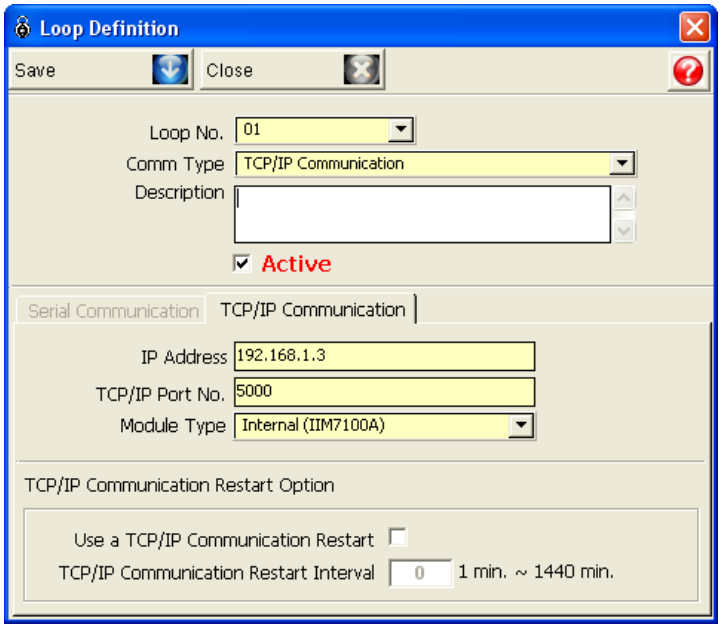


Figura 4.16 – Parâmetros de uma ligação TCP-IP.

com a anterior (lista das áreas) mas com a lista das ligações, onde já aparece uma ligação com o código 00 que é a ligação por defeito. Esta não é possível apagar da lista, mas é possível alterar os seus parâmetros. No entanto, esta é uma ligação feita através da porta série (RS-232) e como durante a execução deste trabalho se utilizou sempre uma ligação TCP-IP (e também porque é como as ligações estão feitas no GNS), adicionou-se uma nova ligação. Depois de colocados todos os parâmetros esta ligação pode ser adicionada ao sistema. No entanto, ainda não existe nenhuma comunicação com as controladoras. Como no edifício do GNS a comunicação das controladoras com o computador central é feita através de uma ligação TCP-IP, a sua distinção é feita pelo seu endereço IP, logo apenas existirá uma controladora por ligação (*Loop*).

Para adicionar uma controladora na ligação criada anteriormente, acede-se novamente ao menu de configurações (*Set-Up*) e à opção *Controller Definition*. Aparece novamente uma

lista com as controladoras instaladas (lista essa que inicialmente está vazia) e adiciona-se uma nova controladora, fazendo aparecer a janela com os respectivos parâmetros.

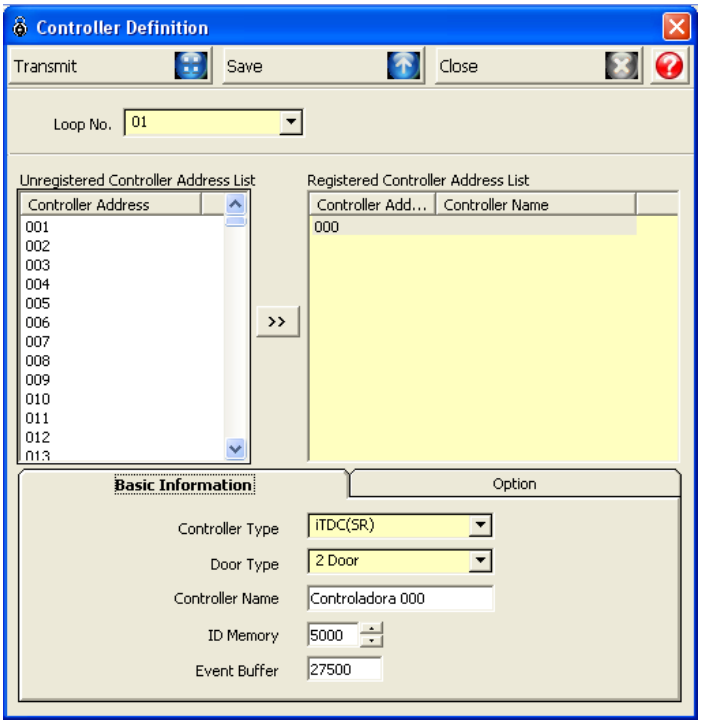


Figura 4.17 – Parâmetros de uma controladora.

Como se pode verificar na figura 4.18 foi apenas adicionada à ligação 01 (criada anteriormente) uma controladora com o número de identificação 000. A partir do momento do registo da controladora é iniciada a comunicação com esta, ou seja o computador central irá começar a receber mensagens de eventos ou alarmes desta controladora. No entanto, esta não está associada a nenhuma porta dentro da aplicação, nem muito menos configurada para abrir mediante a presença de qualquer cartão.

Novamente no menu de configuração ao aceder à opção *Door/Reader Definition* é apresentada uma lista com todas as portas e leitores associados a estas que estão definidos no sistema. Ao adicionar uma nova porta é pedido a ligação, controladora, área e andar (tudo informação que já foi previamente adicionada ao sistema), e também um nome para a porta e os respectivos leitores de entrada e saída.

O próximo ponto trata do controlo de acessos em concreto. Iremos ver como configurar esta nova controladora para abrir quando um determinado cartão é colocado perto do leitor.

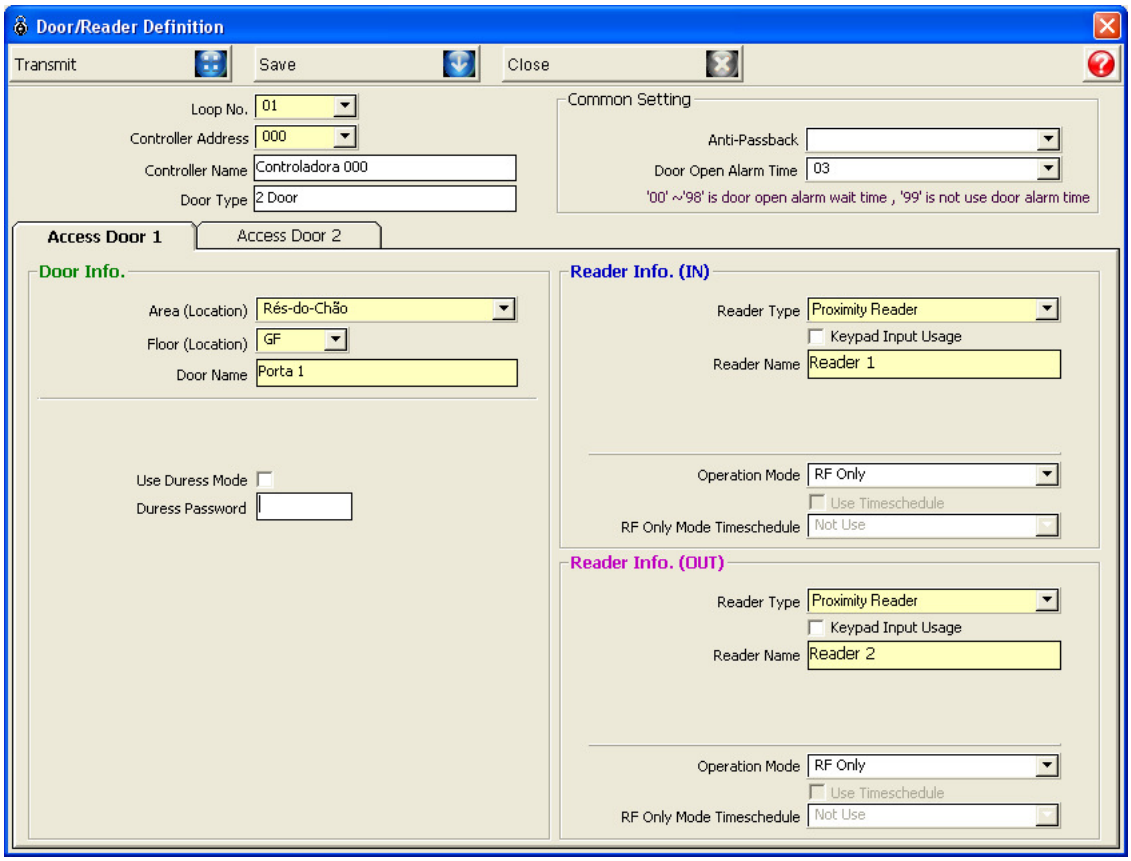
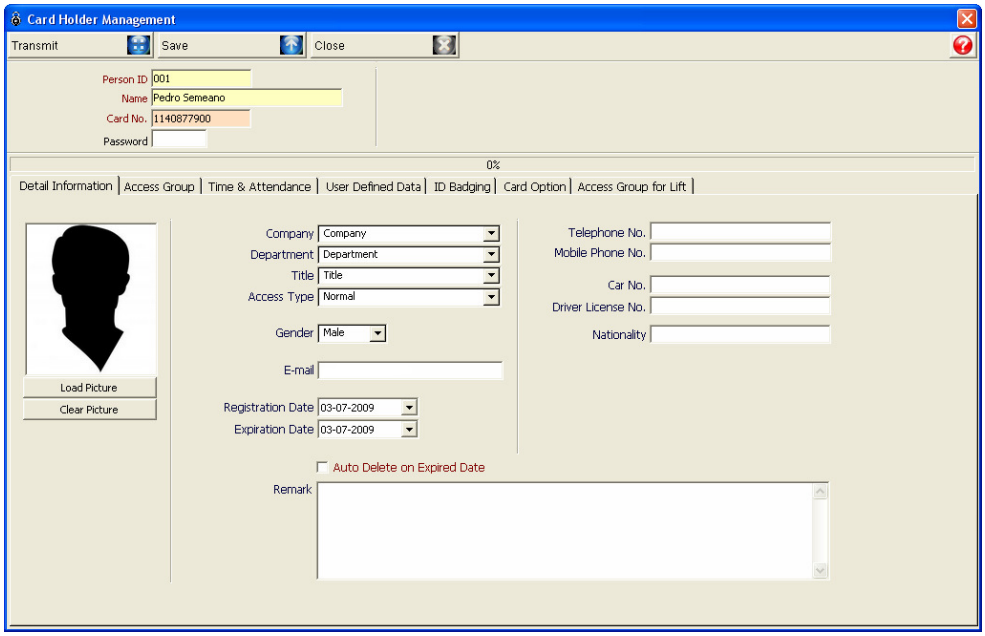


Figura 4.18 – Parâmetros de uma nova porta e dos seus respectivos leitores de entrada e saída.

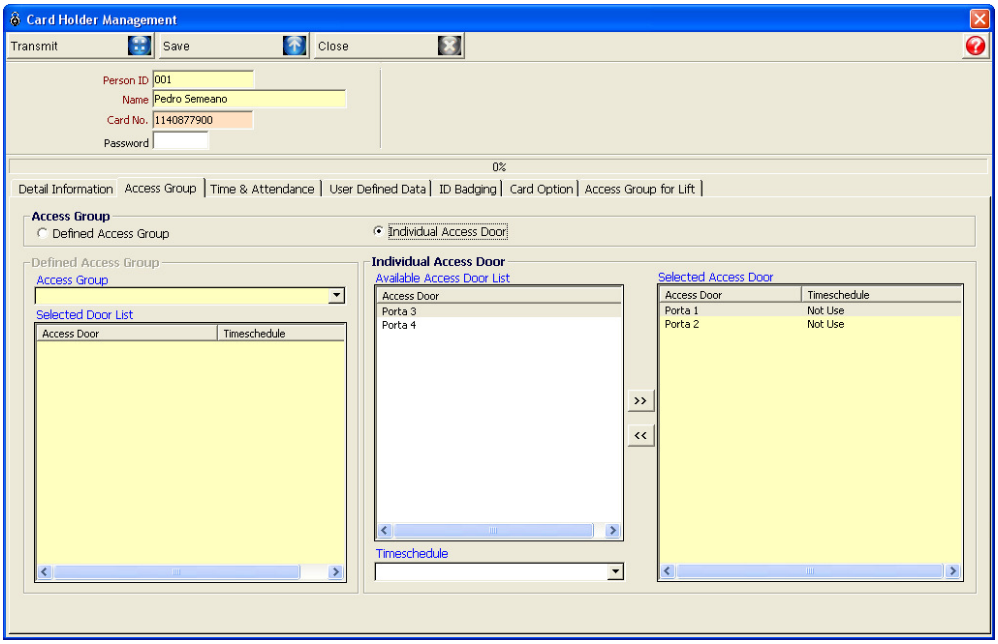
4.2.4. Configuração de acessos

Nesta secção é onde são definidos os utilizadores, qual o número de identificação (que será o número do cartão) e quais as portas a que cada um tem acesso. A opção *Card Holder Management* do menu *Access Control* mostra-nos uma lista de todos os utilizadores registados, e ao adicionar uma nova entrada na lista aparece um conjunto de campo a preencher com os dados do novo utilizador. No separador seguinte, *Access Group*, é onde se definem quais as zonas onde este utilizador tem acesso, sendo possível dar as devidas autorizações através de um grupo predefinido (e.g. "Pessoal do Departamento de Finanças", onde quem pertence a este departamento só tem autorização de acesso pelo caminho que dá até ao Departamento de Finanças) ou simplesmente dando a autorização por porta. Na

imagem que se segue foi definido que o utilizador Pedro Semeano com o número de cartão 1140877900 tem acesso apenas às portas 1 e 2.



(a)



(b)

Figura 4.19 – (a) Registo de um utilizador; (b) Configuração dos acessos permitidos ao mesmo utilizador.

5. Protocolos de comunicação

Durante a execução deste trabalho foi necessário lidar com alguns tipos de protocolos de comunicação entre o computador central e as controladoras, pelo que neste capítulo irá ser feita uma pequena abordagem ao método de transmissão de dados de cada um destes, nomeadamente RS-232, Wiegand e o protocolo de comunicação utilizado entre a controladora e a aplicação. Inicialmente nada se sabia sobre este protocolo que era utilizado na comunicação entre a aplicação e a controladora. Sabia-se que a ligação seria sobre TCP-IP, mas nada se sabia sobre a informação que circulava, pelo que esta foi alvo de estudo e análise mais detalhada.

5.1. RS-232

No protocolo de comunicação RS-232 os dados são enviados um byte de cada vez. A codificação utilizada é o *Start-Stop Assíncrono*, ou seja, por cada byte, existe um bit (start bit) que indica o início, seguido de 8 bits de dados, 1 bit de paridade e um ou dois stop bit que indica o final da transmissão do byte em questão. A velocidade de transmissão pode variar desde 1200 até 115200 bits por segundo. Pode-se dividir as linhas utilizadas pelo RS-232 em três tipos: a referência, as linhas de comunicação de dados e as linhas de *handshake*. A linha de referência serve para que tanto no transmissor como no receptor haja o mesmo valor de 0 Volts. As linhas de comunicação são duas, uma de transmissão e outra de recepção. Obviamente que a linha de transmissão do emissor irá estar ligada à linha de recepção do receptor e vice-versa. As linhas de handshake servem para que haja entendimento entre as duas partes na transmissão de dados, de modo a que não haja colisões.

Sinal	Significado	Explicação
GND	Ground	Referência (0 Volts)
TX	Serial Data Output	Linha de transmissão de dados
RX	Serial Data Input	Linha de recepção de dados
DTR	Data Terminal Ready	Avisa o modem que o terminal UART está pronto
DSR	Data Set Ready	Avisa o terminal UART que o modem está pronto
RTS	Request To Send	O terminal está pronto para receber os dados
CTS	Clear To Send	O modem está pronto para enviar os dados
DCD	Data Carrier Detected	Portadora detectada (quando existe uma ligação entre dois modems)
RI	Ring Indicator	Fica activo quando o modem detecta um sinal de toque de telefone

Quadro 5.1 – Pinos de um conector RS-232 de nove pinos[33].

5.2. Wiegand

O Wiegand é um protocolo de comunicação que utiliza três linhas de transmissão, onde uma destas é apenas a referência (0V) e as outras duas de transmissão de dados (Data 0 e Data 1). Quando nenhuma informação está a ser transmitida, tanto a linha Data 0 como a Data 1 encontram-se com valor lógico 1 (ou seja, +5V). Quando se quer transmitir o valor lógico 0 a linha Data 0 muda o seu valor lógico para 0 (0 Volts); e quando se quer transmitir o valor lógico 1 a linha Data 1 muda o seu valor lógico para 0. Quando uma linha está a 0 a outra está obrigatoriamente a 1, e vice-versa. A mudança de valor lógico ocorre em forma de pulso que tem a duração normal de 100 us (Tpw – ver figura 2.3) e entre cada pulso deve existir um mínimo de 1 ms de intervalo (Tpi).

O formato inicial do protocolo tinha 1 bit de paridade, 8 bits de *facility code* (que identifica o leitor), 16 bits do número de identificação (que identifica o cartão) e finalmente um stop bit, perfazendo um total de 26 bits. No entanto existem variações de 34, 39 e 44 bits.

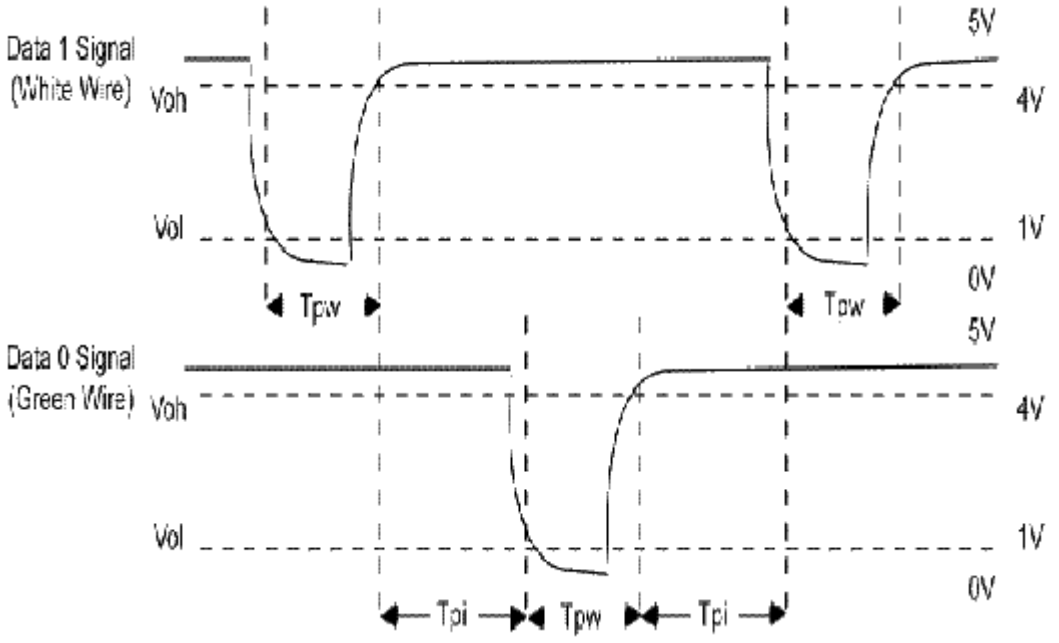


Figura 5.1 – Padrão de transmissão de dados Wiegand[32].

A grande vantagem do Wiegand em relação ao RS-232 é que este permite comunicações em linhas com um maior comprimento (até 150 metros; o RS-232 não é aconselhado para distancias superiores a 15 metros, isto à velocidade máxima e com um cabo série vulgar).

5.3. Protocolo de comunicação com as controladoras

De modo a analisar os dados que eram enviados e recebidos da controladora foi utilizada uma ferramenta (Wireshark) que mostra o conteúdo dos pacotes IP que são transaccionados numa ligação.

5.3.1. Estabelecimento de ligação

O início da ligação entre o programa e a controladora é uma simples ligação TCP. Antes de esta se iniciar é necessário que o servidor (a parte que vai receber a ligação, neste caso a controladora) esteja associado a uma porta e que a abra para receber ligações. Assim que a porta do servidor está aberta, o cliente (a aplicação de controlo de acessos) pode iniciar o processo. A ligação TCP utiliza um *handshake* de 3 passos:

1. O cliente envia para a porta aberta pelo servidor um pacote de sincronismo.
2. O servidor envia um outro pacote de resposta ao pedido.
3. O cliente responde ao servidor notificando-o que recebeu a sua resposta.

Neste ponto, tanto o cliente como o servidor receberam uma resposta de início de ligação. Mas tudo isto e a estrutura dos pacotes estão escondidos do utilizador visto que isto é tratado numa camada inferior à camada de aplicação do modelo OSI (Open Systems Interconnection).

Na aplicação, os passos dados não foram mais que a configuração de uma ligação, indicando o endereço IP do servidor, o porto ao qual o servidor está à escuta dos pedidos de novas ligações, sendo em seguida iniciada a ligação. No entanto, com a ligação estabelecida nada acontece, mesmo que haja alguma actividade junto da controladora. Isto acontece porque a controladora não é proactiva, mas sim reativa. Para que esta responda, precisa de receber uma pergunta, neste caso uma mensagem de *polling*.

5.3.2. Polling

A mensagem de *polling* é uma mensagem que tem como objectivo perguntar à controladora se existiu alguma actividade desde a última vez que recebeu este tipo de mensagem. Quando se passa um cartão pelo leitor de proximidade de uma controladora o número do cartão fica registado na memória desta juntamente com a data e a hora em que tal acontecimento se deu, assim como qual o leitor que fez a leitura e uma *flag* que indica se o acesso foi concedido ou não. Quando a controladora recebe a mensagem de *polling* o registo é enviado para o computador central, libertando a memória local da controladora.

Eis um exemplo do conteúdo de uma mensagem de *polling*:

```
01 30 30 33 56 39 30 31 32 32 0D
```

Quadro 5.2 – Exemplo de uma mensagem de *pooling*.

Cada um dos tuplos representados é 1 byte apresentado em valor hexadecimal, e consultando a tabela ASCII (Anexo 3) podemos fazer a sua tradução para a sua representação em caracteres ASCII, resultando na seguinte cadeia:

```
[SOH]003V90122[CR]
```

Quadro 5.3 – Exemplo de uma mensagem de *pooling* em ASCII.

O primeiro e últimos caracteres representam um *Start of Heading* e um *Carriage Return*, respectivamente. Ou seja, o primeiro indica o início da informação contida no pacote, e o último o fim deste.

Os três caracteres seguintes ("003") dizem respeito ao número de identificação da controladora, ou seja, neste exemplo a mensagem era destinada à controladora 003 (como já foi dito anteriormente este número pode ir de 000 a 255). Alterando o destinatário da mensagem vemos que estes caracteres mudam consoante o número de identificação da controladora.

Os dois caracteres seguintes ("V9") são os que indicam o tipo da mensagem, neste caso "V9" é o identificador de uma mensagem de *polling*. Mais adiante iremos ver que consoante o tipo de mensagem este identificador também muda.

Finalmente, os últimos quatro caracteres que antecedem o caracter de fim de mensagem (*Carriage Return*) não são mais que o valor de *checksum*, onde este é calculado

fazendo a soma em hexadecimal do conteúdo (tudo excepto o *Start Of Heading*, o *Checksum* e o *Carriage Return*), ou seja:

```
003V9
```

Quadro 5.4 – Conteúdo de uma trama de *pooling*.

Conteúdo este que tem a seguinte representação em hexadecimal:

```
30 30 33 56 39
```

Quadro 5.5 – Conteúdo de uma trama de *pooling* em hexadecimal.

Ao fazer-se a soma (em hexadecimal) do conteúdo obtém-se:

```
122
```

Quadro 5.6 – Valor do *checksum* de uma trama de *pooling*.

O *checksum* é dado sempre por quatro caracteres, sempre. Deste modo terá que se adicionar um zero à esquerda ficando:

```
0122
```

Quadro 5.7 – Valor normalizado do *checksum* de uma trama de *pooling*.

Convertendo cada caracter acima para o seu representante hexadecimal resulta em:

```
30 31 32 32
```

Quadro 5.8 – Valor do *checksum* de uma trama de *pooling* em hexadecimal.

Que é o valor que está na trama inicial.

Por cada trama de *polling* enviada pelo computador central a controladora responde com outra trama que neste caso poderia ser algo como o exemplo seguinte:

```
01 30 30 33 56 31 31 31 31 31 31 31 31 31 31
31 31 31 31 31 31 30 30 30 30 30 30 30 30 30
30 30 30 30 30 30 30 36 3C 39 0D
```

Quadro 5.9 – Resposta da controladora à trama de *pooling*.

Novamente, fazendo a correspondência para caracteres ASCII, obtém-se:

```
[SOH]003V1111111111111111000000000000006<9[CR]
```

Quadro 5.10 – Resposta da controladora à trama de *pooling* ASCII.

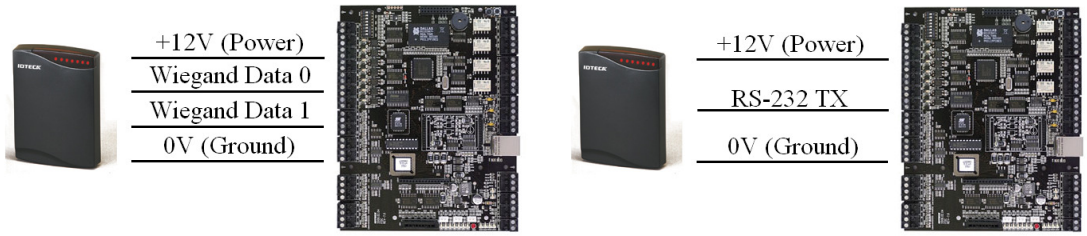
Os três primeiros caracteres, tal como na trama de *polling*, correspondem ao número de identificação da controladora. Se a trama anterior foi enviada para a controladora 003 é normal que a trama de resposta seja da mesma controladora.

Em seguida vem a identificação do tipo de trama, para que o computador central saiba o que está a ler, neste caso é uma trama de *status* ("*V1*"), porque o seu conteúdo indica o status dos sensores e actuadores da controladora. Os zero (0) e um (1) que se vêm no resto da trama é nada mais que o indicador do estado destes, onde as entradas (sensores) são *active-*

low e as saídas (actuadores) *active-high*. No fim da trama vemos novamente o valor de *checksum* e *Carriage Return*.

5.3.3. Passagem de cartão

O leitor de proximidade não faz parte da controladora, é algo que é adicionado à controladora, ou seja, é um sistema independente. Como tal, sendo a controladora e o leitor dois sistemas separados, necessitam de um protocolo de comunicação para que se possam "entender". Neste caso existem dois protocolos que podem ser utilizados para que o leitor comunique com a controladora: RS-232 ou Wiegand.



(a) (b)

Figura 5.2 – Ligação do leitor: a) Protocolo Wiegand; b) Protocolo RS-232.

Seja qual for o protocolo utilizado (a transmissão é unidireccional pelo que apenas é utilizada a linha de transmissão no RS-232) os dados transmitidos assemelham-se com o seguinte exemplo:

```
[STX]1140877900[ETX][EOT]
```

Quadro 5.11 – Dados transmitidos entre o leitor de cartões e a controladora.

Onde em hexadecimal corresponde a:

```
02 31 31 34 30 38 37 37 39 30 30 03 04
```

Quadro 5.12 – Dados transmitidos entre o leitor de cartões e a controladora.

O primeiro caracter corresponde a um *Start Of Text*, e onde o penúltimo e último caracter correspondem a um *End Of Text* e um *End Of Transmition*, respectivamente. Os caracteres numéricos que estão pelo meio correspondem ao número de identificação do cartão que foi lido pelo leitor.

O envio de uma trama de *status* é o comportamento normal de uma controladora que não tem nada a reportar ao computador central. Quando se passa um cartão no leitor da controladora esta dá a resposta automaticamente (abre ou não a porta caso o cartão esteja registado ou não) e o evento é guardado em memória. Quando esta recebe a trama de polling responde com uma trama, diferente da trama de status, onde reporta o evento. Eis um exemplo de uma resposta de evento:

```
[SOH]005V2111408779002009011921123491A06<1[CR]
```

Quadro 5.13 – Trama de *status* enviada pela controladora.

Desta vez o conteúdo é bastante diferente, e pode-se ver que não se trata já de uma trama de status ("*V2*").

Separando as partes já anteriormente analisadas (*Start Of Heading*, número da controladora, tipo de trama, *checksum* e *Carriage Return*) do resto dos dados da trama, obtemos o seguinte:

```
1114087790020090119211234 91A
```

Quadro 5.14 – Conteúdo da trama de *status* enviada pela controladora.

O primeiro caracter ("*1*") corresponde ao identificador do leitor, que neste caso pode variar entre 1 e 4. Os dez números seguintes ("*1140877900*") correspondem ao número de identificação do cartão. Os oito números seguintes ("*20090119*") correspondem à data do evento pela ordem ano-mês-dia; neste caso o evento que está a ser analisado ocorreu no dia 19 de Janeiro de 2009. O caracter seguinte indica o dia da semana; neste caso aparece o número 2, e se formos ver ao calendário o dia 19 de Janeiro de 2009 é uma segunda-feira. Em seguida aparece a hora do evento pela ordem horas-minutos-segundos; este deu-se às 11 horas, 23 minutos e 49 segundos. O caracter seguinte indica o status do evento de acordo com a seguinte tabela:

Valor	Significado
0	Identificação válida
1	Identificação inválida
2	Horário inválido
3	Erro de *antipassback* (entrada)
4	Erro de *antipassback* (saída)
5	Erro de *password*
6	Erro de acesso (acesso negado)
7	Erro de impressão digital
8	*Duress mode* (IDTeck)

Quadro 5.15 – Identificação dos eventos da controladora.

Após o envio desta trama de evento, mais umas mensagens são trocadas entre a controladora e o computador central de modo a garantir que a transmissão foi bem sucedida. O computador envia uma trama de *acknowledge* para avisar a controladora que recebeu bem a mensagem do evento e a controladora responde com uma outra mensagem de *acknowledge* para avisar o computador central que recebeu a confirmação. A partir deste ponto a controladora apaga o evento da memória local e a comunicação volta à normalidade (recebe

tramas de *polling* respondendo com tramas de status). Caso alguma mensagem se perca na comunicação esta é reenviada até que cada um receba a devida resposta.

5.3.4. Monitorização dos sensores e actuadores

A abertura da porta não é algo que depende do computador central. A decisão é feita localmente na controladora, e esta irá abrir a porta se o cartão que foi apresentado no leitor está registado na sua memória interna. Tal como já foi dito no tópico anterior, este evento é apenas comunicado ao computador central através de uma trama de evento. Além disso, depois do evento reportado, a controladora irá responder às tramas de *polling* que chegam até ela, e nas tramas de status enviadas pode-se ver o estado dos sensores e actuadores. Com base nisto é possível saber se uma dada porta está aberta ou fechada, se foi aberta sem a actuação do trinco, etc.

Em anexo (Anexo 4) apresenta-se um exemplo de um evento que começa com uma situação de *polling* normal, passando pela apresentação do cartão, abertura e fecho da porta, onde se consegue ver a mudança dos valores (bits) referentes ao actuador da fechadura e ao sensor de abertura da porta.

6. Organização de dados

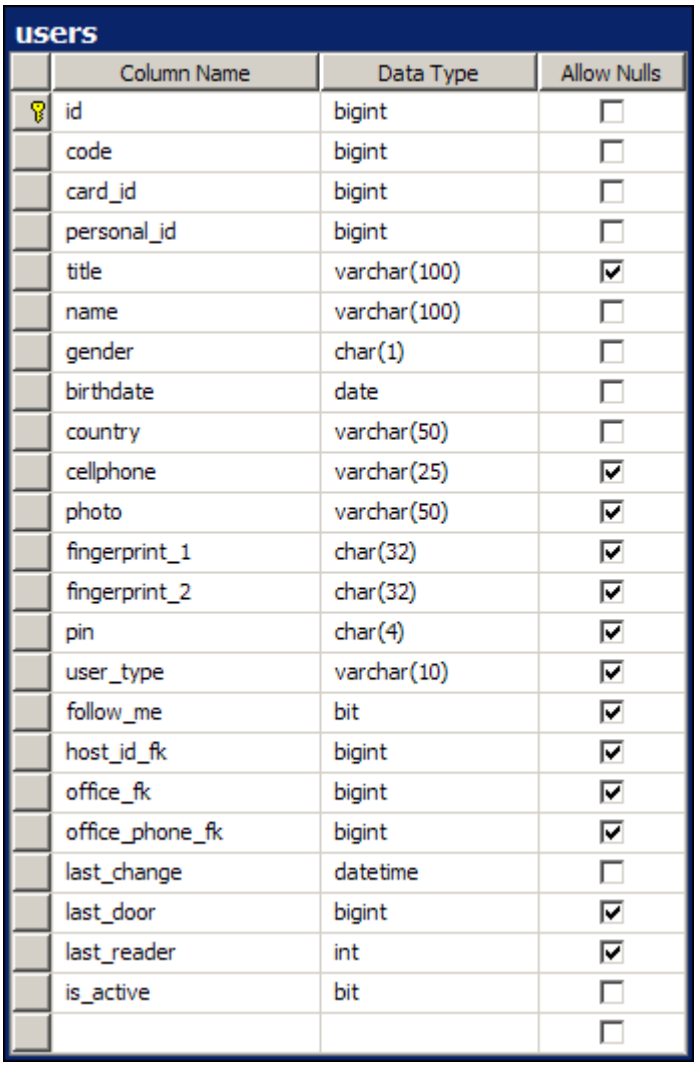
Este capítulo mostra de uma forma muito simples como foi organizada a estrutura da base de dados para a aplicação de controlo de acessos desenhada. Será onde ficará guardada toda a informação relativa aos utilizadores, eventos, alarmes, regras especiais, informação sobre o edifício, localização de portas, etc.

6.1. Utilizadores

É sempre necessário, seja em que empresa for, ter alguma informação sobre os seus empregados. Não só informação relacionada com a empresa, como o cargo que ocupa, qual o seu gabinete, contacto, etc., mas também informação pessoal como por exemplo o seu nome completo, número do bilhete de identidade, data de nascimento, etc.

Como é normal existe sempre um campo que será a chave primária, e neste caso é o campo "id" que não é mais que um campo automático que incrementa cada vez que se adiciona um novo utilizador. Em segundo lugar na estrutura aparece um campo com o nome "code". O "code" funciona de forma idêntica ao "id" mas com a particularidade de que o "code" é um campo que entre todas as entradas não tem falhas pelo meio. Mesmo que hajam falhas, ou entradas que posteriormente são apagadas, o último campo "code" corresponderá sempre ao número de entradas na base de dados; se existem 10 utilizadores então o último utilizador que foi introduzido no sistema tem o campo "code" igual a 10, o que pode não acontecer com o campo "id" que é gerado automaticamente.

Também aparecem na tabela de utilizadores algumas chaves estrangeiras, nomeadamente do gabinete e do número do gabinete. Estas são chaves primárias nas tabelas de gabinetes e números de gabinetes, respectivamente.



users

Column Name	Data Type	Allow Nulls
id	bigint	☐
code	bigint	☐
card_id	bigint	☐
personal_id	bigint	☐
title	varchar(100)	☑
name	varchar(100)	☐
gender	char(1)	☐
birthdate	date	☐
country	varchar(50)	☐
cellphone	varchar(25)	☑
photo	varchar(50)	☑
fingerprint_1	char(32)	☑
fingerprint_2	char(32)	☑
pin	char(4)	☑
user_type	varchar(10)	☑
follow_me	bit	☑
host_id_fk	bigint	☑
office_fk	bigint	☑
office_phone_fk	bigint	☑
last_change	datetime	☐
last_door	bigint	☑
last_reader	int	☑
is_active	bit	☐
		☐

Figura 6.1 – Tabela de utilizadores.

Existe um outro campo no fim da tabela que, tal como os campos “id” e “code”, aparece em todas as outras tabelas incluídas na base de dados do programa. O campo “is_active” serve para indicar se uma dada entrada está ou não activa. Ou seja, sempre que se queira apagar um registo, na verdade esse registo não é apagado; o campo “is_active” é colocado a zero (visto que é um campo booleano) dando a indicação que aquele registo não está activo, logo é como se tivesse sido apagado. Não se apagam permanentemente os registos porque nunca se sabe se no futuro será necessário verificar registos passados.

6.2. Edifício

Num edifício existe muita informação que é necessária guardar e que é imprescindível para um software de controlo de acessos. A zona "mãe" é provavelmente os andares (floors) que fazem parte do edifício, onde depois dentro de cada um iremos encontrar as portas, departamentos, gabinetes, etc. Para cada andar é guardado o seu nome e o nome de ficheiro da imagem da planta desse andar.

floors

Column Name	Data Type	Allow Nulls
id	bigint	☐
code	bigint	☐
floor_name	varchar(50)	☐
floor_plan	varchar(50)	☐
is_active	bit	☐
		☐

(a)

doors

Column Name	Data Type	Allow Nulls
id	bigint	☐
code	bigint	☐
name	varchar(50)	☐
map_coordinates	varchar(10)	☐
controller_id	char(3)	☐
controller_ip	varchar(15)	☐
controller_port	varchar(5)	☐
reader_in	char(1)	☐
reader_in_name	varchar(50)	☐
reader_out	char(1)	☐
reader_out_name	varchar(50)	☐
exit_door	bit	☐
arinca	bit	☐
floor_fk	bigint	☐
arinca_fk	bigint	☑
poller_index	int	☑
is_active	bit	☐
		☐

(b)

Figura 6.2 – (a) Tabela de andares; (b) Tabela de portas.

As portas (doors) irão ter um campo com uma chave estrangeira do andar a que pertencem ("floor_fk"). O campo "map_coordinates" serve para localizar o ícone da porta na imagem da planta do andar, e os campos "arinca" e "exit_door" para indicar se se trata de uma maquina dispensadora de cartões ou de uma porta de saída. Caso nenhum destes campos esteja marcado então trata-se de uma porta normal. Em cada porta também ficam guardados os dados sobre a controladora que é responsável pela sua abertura e fecho. Note-se que os campos "id", "code" e "is_active" estão sempre presentes.

Em cada andar podem existir um ou vários departamentos (departments), logo a tabela de registo de departamentos irá ter uma chave estrangeira indicando qual o andar a que pertence. Cada gabinete (offices) pertencerá a um departamento e da mesma forma cada número (office_phones) pertencerá a um gabinete. Desta forma obtém-se a relação que se mostra na figura 6.4.

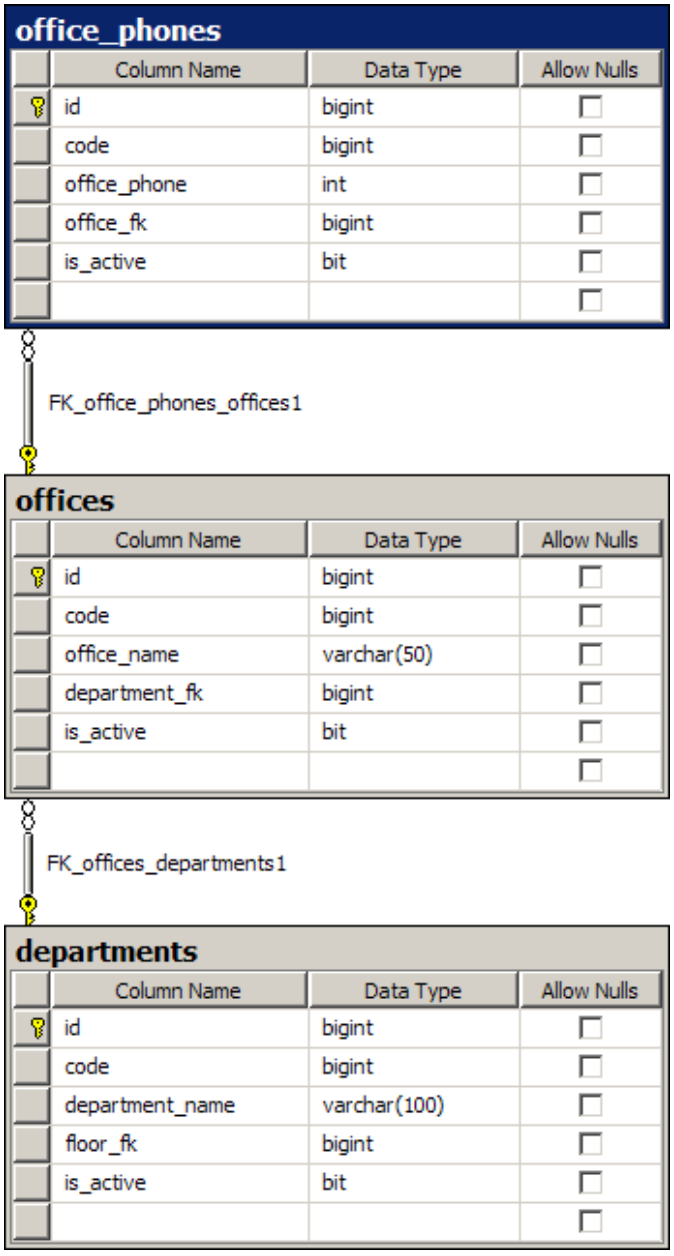
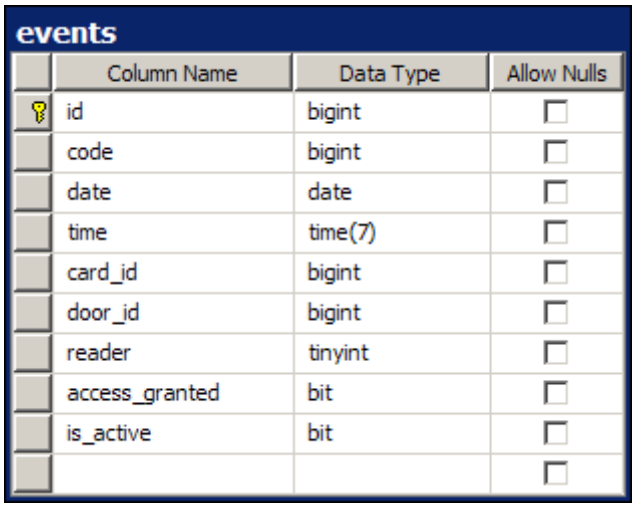


Figura 6.4 – Relação entre as tabelas de departamentos, gabinetes e números de gabinete.

6.3. Eventos

Sempre que alguém passa o seu cartão por um leitor é gerado um evento que é reportado ao computador central onde este o irá registar na sua base de dados. Seja este um evento de acesso autorizado ou negado, todos os eventos são registados. O registo de um evento contém o dia e a hora do evento, o cartão que despoletou esse evento, a porta que foi aberta, o leitor que leu o cartão e finalmente o resultado desse evento (se abriu a porta ou não).



events

Column Name	Data Type	Allow Nulls
id	bigint	☐
code	bigint	☐
date	date	☐
time	time(7)	☐
card_id	bigint	☐
door_id	bigint	☐
reader	tinyint	☐
access_granted	bit	☐
is_active	bit	☐
		☐

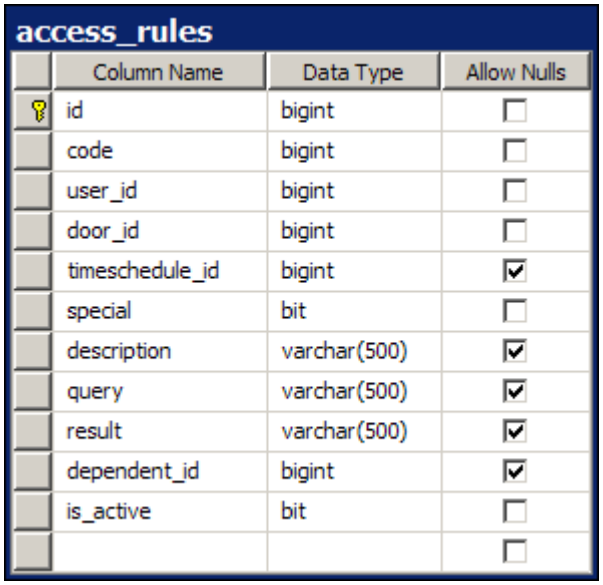
Figura 6.5 – Tabela de registo de eventos.

O registo destes eventos irão permitir a criação de relatórios de acesso detalhados. Será possível saber quem entra atrasado, quem cumpre todas as horas de trabalho semanais, quem cumpre a mais, quem tentou ter acesso onde não devia, qual o trajecto que fez durante o dia, etc.

6.4. Regras de acesso

Quando se define uma regra de acesso para qualquer utilizador é necessário guardar essa configuração. Não só em caso de utilização das regras especiais mas também no caso configuração de acesso simples numa porta. Para cada porta que cada utilizador tem acesso é

feito um registo na tabela de regras de acessos. Se se tratar de uma regra de acesso especial então o campo "special" virá marcado (com o valor *true*) o



access_rules

Column Name	Data Type	Allow Nulls
id	bigint	☐
code	bigint	☐
user_id	bigint	☐
door_id	bigint	☐
timeschedule_id	bigint	☑
special	bit	☐
description	varchar(500)	☑
query	varchar(500)	☑
result	varchar(500)	☑
dependent_id	bigint	☑
is_active	bit	☐
		☐

Figura 6.6 – Tabela de regras de acesso.

que significa que os campos "query" e "result" não estão vazios. Nestes virão as *queries* e os resultados das mesmas que farão validar a regra especial que foi atribuída. Cada *query* virá separada por um ponto e vírgula da seguinte, assim como os resultados. As *queries* e os resultados encontram-se ordenados, logo, a primeira *query* terá que devolver o valor do primeiro resultado para permitir acesso.

Se se tratar de um acesso simples (abrir uma porta sem ser necessária a validação de uma regra) então nesse caso o campo "special" não estará validado e os campos "query" e "result" serão nulos.

O campo "dependent_id" serve para indicar se aquela regra de acesso está dependente de outro utilizador. Por exemplo, se o User001 tiver uma regra com o "dependent_id" do User007, cada vez que o User007 gerar um evento, a regra especial do User001 vai ser verificada, e se for validada, este irá obter autorização de acesso. A vantagem de se fazer a verificação desta forma é que torna a resposta de configuração de acesso mais rápida. Imagine-se o exemplo de o User001 apenas poder entrar na porta 007 quando o User007 passar nesta. É mais rápido, no ponto de vista do utilizador, a autorização ser dada quando o User007 passa pela porta, do que ser dada a autorização quando o User001 tenta passar pela

porta. Isto é, quando o User007 passa pela porta, o sistema verifica que o User001 tem uma regra dependente do User007 e dá-lhe a devida autorização. E assim, quando o User001 passar o cartão pelo leitor da porta esta vai abrir no mesmo instante, porque o acesso já tinha sido concedido.

De outra forma, a regra teria que ser verificada "on-the-fly", ou seja, quando o User001 passasse o cartão pelo leitor, o sistema teria que validar a regra antes de dar a autorização de abertura, o que iria fazer com que o utilizador tivesse que esperar alguns segundos pela abertura da porta.

Para além destas regras especiais, também é possível limitar o acesso de acordo com um horário predefinido, e é para isso que serve o campo "timeschedule_id", que está relacionado com a tabela de horários (timeschedule).

time_schedule

Column Name	Data Type	Allow Nulls
id	bigint	☐
code	bigint	☐
timeschedule_code	char(2)	☐
name	varchar(50)	☐
sunday	char(3)	☐
monday	char(3)	☐
tuesday	char(3)	☐
wednesday	char(3)	☐
thursday	char(3)	☐
friday	char(3)	☐
saturday	char(3)	☐
is_active	bit	☐
		☐

Figura 6.7 – Tabela de horários.

Em cada horário podemos definir diferentes intervalos de acesso para cada dia. Mais concretamente, é possível definir até cinco intervalos (time_zone). Por exemplo, uma *time zone* pode ter os seguintes intervalos: das 11:00 às 12:00, das 14:00 às 16:00 e das 20:00 às 21:00. Apenas estão definidos 3 intervalos mas poder-se-ia ter definido mais dois. E agora,

tendo esta *time zone* assim definida, posso aplicá-la a qualquer dia de semana, e durante esses dias apenas ter-se-ia acesso dentro dos intervalos indicados.

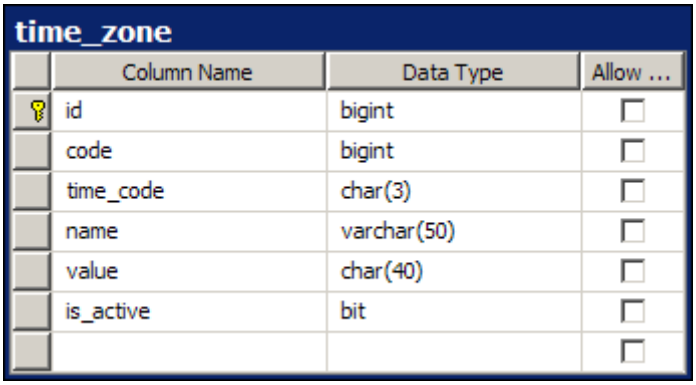


Figura 6.8 – Tabela de intervalos de acesso.

Os intervalos estão indicados no campo "value" que tem o comprimento exacto de 40 caracteres. Para o exemplo dado acima, este campo teria o seguinte valor:

```
value = '1100120014001600200021000000000000000000'
```

Quadro 6.1 – Exemplo do valor do campo "value" na tabela *time_zone*.

7. Aplicação de controlo de acessos com personalização de regras de acessos

Neste capítulo explica-se porque surgiu a ideia de criar um programa de controlo de acessos com personalização de regras de acesso, quais as opções que surgiram e o porquê da escolha do XML como solução ao problema apresentado.

7.1. O problema

Tal como já foi dito anteriormente, o Gabinete Nacional de Segurança está equipado com um sistema de controlo de acessos da IDTeck que permite controlar de uma forma eficaz quem tem acesso a determinados pontos do edifício, podendo também ter um registo de todos os acessos ou tentativas de acesso. No entanto, existia uma falha na segurança: cada utente levava consigo para fora do edifício o cartão de identificação, podendo facilmente ir parar às mãos erradas.

Este problema foi apresentado ao Professor José Manuel Fonseca a quem ficou incumbida a responsabilidade de construir uma máquina dispensadora de cartões, de modo a que os cartões de acesso nunca saíssem do edifício. Esta máquina entrega um cartão mediante a apresentação de um código e da impressão digital, de forma a garantir que o cartão é entregue à pessoa a quem este pertence.

Depois de instalada a máquina no Gabinete Nacional de Segurança, altos responsáveis apresentaram mais umas questões que gostavam de ver melhoradas de forma a garantir o máximo de segurança dentro do edifício. Eis algumas:

- Acessos especiais. Um dos exemplos dados é a situação de uma dada pessoa ter acesso a um gabinete apenas se a pessoa a quem o gabinete pertence estiver presente. Caso contrário o acesso não é permitido;

- Integração com a máquina dispensadora de cartões. A entrada no edifício é feita apenas com recurso a leitura da impressão digital. No entanto a entrada apenas deverá ser concedida se o cartão da pessoa identificada estiver presente na máquina dos cartões;

- Cartões de convidados com opção de *follow-me*. O *follow-me* implica que o convidado só tenha acesso à última porta que o anfitrião abriu, fazendo com que o convidado tenha que andar sempre acompanhado do seu anfitrião;

- Definição de zonas no edifício. Desta forma é possível atribuir zonas de acesso a diversos utilizadores, sem ter que definir cada porta uma a uma.

Tal como foi demonstrado na pesquisa e análise aos diversos programas de controlo de acessos, nem o actual software de segurança (Starwatch PRO II) nem qualquer outro permite introduzir este tipo de regras no sistema. A única solução encontrada para satisfazer tais necessidades foi a de criar um software de raiz que o permitisse.

7.2. A solução

Para além do que o actual software permitia, seria necessário que a nova aplicação permitisse edição de regras de acesso. Inicialmente poderiam haver poucas regras a implementar, mas poderia surgir o caso de mais tarde ser necessário incluir uma nova regra no sistema, e para não fazer uma alteração sobre o código fonte da aplicação, foi necessário encontrar um método de criação de regras suficientemente flexível de modo a adaptar-se a novas situações.

Inicialmente pensou-se numa espécie de painel gráfico onde seria possível escolher o cartão do utilizador a editar, a porta onde aplicar a regra, o tipo de regra, o horário de

aplicação da regra, etc. O problema desta solução é que não é suficientemente flexível, porque para adicionar uma nova regra de acesso torna-se necessário editar o código fonte da aplicação.

Assim, criando um ficheiro de descrição de regras que estariam definidas numa espécie de meta-linguagem e que fosse interpretada pela aplicação, esta poderia moldar-se de acordo com o seu conteúdo sem necessidade de alterar o código fonte. Esta solução necessita que se compreenda a lógica da meta-linguagem, que pode não estar ao alcance de qualquer um, mas é uma solução bem mais flexível que a primeira.

No que diz respeito à meta-linguagem, não traz qualquer vantagem inventar algo de novo quando se pode muito bem utilizar uma linguagem já existente e que seja, até certo ponto, intuitiva. Um exemplo disso é o XML. Criando um conjunto de regras de acesso num ficheiro XML, e tendo um *parser* na aplicação que as interprete, torna-se possível modelar a aparência da aplicação gráfica de acordo com o seu conteúdo. Utilizando este novo método para criar regras de acesso consegue-se englobar praticamente qualquer cenário, como irá ser explicado em detalhe mais adiante.

7.3. A aplicação

O objectivo inicial era criar uma aplicação que conseguisse fazer tudo ou praticamente tudo que a aplicação anterior (Starwatch PRO II) faz em termos de controlo de acessos. No entanto, a criação de raiz de uma aplicação desta envergadura implica muitos passos intermédios como a criação de uma base de dados relacional, a ligação com esta, a comunicação com as controladoras, etc. Esteve também sempre presente a preocupação de criar uma aplicação o mais simples e *user-friendly* possível, o que implica ter sempre uma perspectiva global de modo a agrupar o maior número de acções numa só. Como iremos ver mais adiante neste capítulo, o processo de adicionar um ponto de acesso no Starwatch exige mais passos e não é tão simples como fazê-lo nesta nova aplicação.

De forma a proteger as configurações de acessos, quando a aplicação inicia pede uma palavra-passe, e assim, só quem sabe a *password* é que está autorizado a fazer as devidas alterações. Após a correcta introdução dos dados, é apresentada a janela principal da

aplicação. Esta é constituída por vários menus, onde se encontram as diversas opções do programa que vão desde a gestão de utilizadores do sistema, configuração dos diversos pontos de acessos espalhados pelo edifício, criação de relatório de eventos, e alteração do idioma.

Figura 7.1 – Introdução da *password* de acesso.

Também se pode ver uma janela de eventos onde, à medida que estes ocorrem, aparecem indicando o ponto de acesso, a identificação do utilizador, assim como a data e hora do evento. Nesta janela existem alguns separadores que permite visualizar diferentes tipos de evento:

- **Todos os eventos**: neste separador aparecem todos os eventos registados, independentemente do tipo;
- **Acesso permitido**: neste separador apenas aparecem os eventos que resultaram na abertura de uma porta;
- **Acesso negado**: apenas aparecem os eventos que resultaram na não abertura de uma porta;
- **Arinca**: apenas aparecem os eventos de recolha ou entrega de cartões nas máquinas dispensadoras de cartões;
- **Porta de saída**: eventos de passagem nas portas de saída.

No final existe mais um separador mas não se relaciona apenas com os eventos de entradas e saída. Este separador mostra uma consola onde podemos ver com algum detalhe o

que está a ser feito por detrás como a comunicação e programação das controladoras, horários de acesso, conecção com a base de dados, etc. Cada vez que o programa é encerrado, todos os dados que foram escritos na consola, são guardados num ficheiro de texto (LOG).

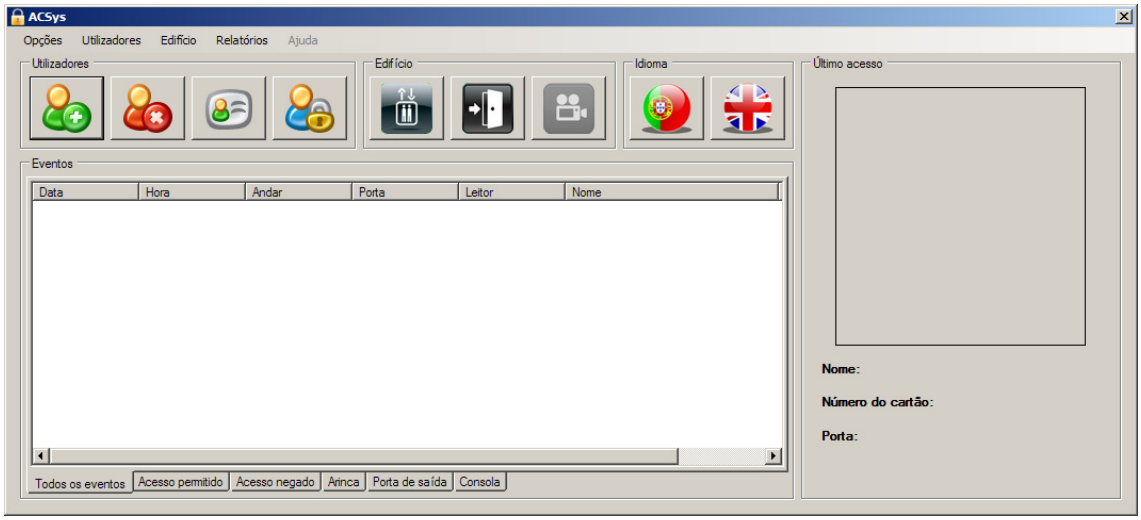


Figura 7.2 – Janela principal da aplicação de controlo de acessos.

7.3.1. Menus

• Opções

Neste menu podemos alterar as opções gerais da aplicação que, numa primeira versão, se resume apenas ao idioma de apresentação. Todas as palavras utilizadas na aplicação estão num ficheiro *Managed Resource File*, o que faz com que, para adicionar uma nova língua, basta adicionar um novo ficheiro deste tipo, com as devidas traduções, e adicionar ao projecto.

Para tornar mais simples a migração de outro software para o ACSys (nome dado a este projecto) foi criada uma opção de importação de dados de utilizadores. Para isso, basta preencher os campos de um ficheiro Excel predefinido, que se encontra na pasta do programa, e carregar em "Importar dados dos utilizadores". Qualquer utilizador que tenha o mesmo cartão de identificação de um outro utilizador já existente, não será adicionado à base de

dados. No final da operação são indicados quantos utilizadores foram adicionados e quantos foram ignorados.

Neste menu também podemos trancar ou encerrar a aplicação. Ao trancar a aplicação faz-se com que só seja possível voltar a fazer alterações, depois de introduzir novamente a palavra-passe.

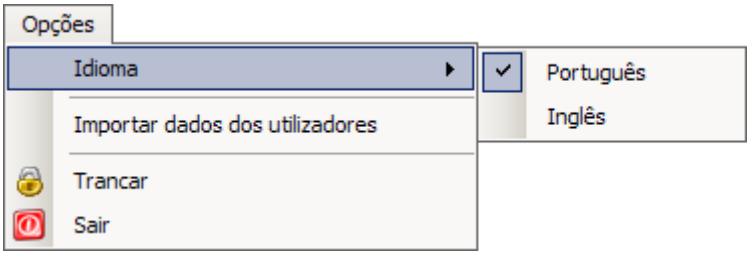


Figura 7.3 – Menu de opções.

• Utilizadores

É no menu dos utilizadores é que se encontram as opções para adicionar, remover ou alterar os dados destes. No caso de se querer remover ou alterar dados, é apresentado ao gestor do programa uma janela para efectuar a procura do utilizador que deseja remover ou alterar. Também se podem alterar as permissões de acesso e predefinir horários de acesso onde é possível, posteriormente, atribuir um destes a uma das portas onde o utilizador queira ter acesso.

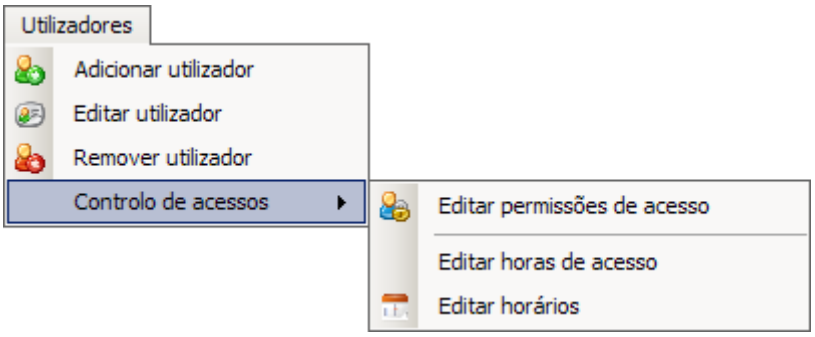


Figura 7.4 – Menu de utilizadores.

• Edifício

Neste menu é onde se encontram as opções para configuração dos andares e portas (pontos de acesso). Encontra-se também um pequeno sub-menu que irá permitir (numa versão posterior) fazer a integração com câmaras IP que estejam instaladas no edifício. Alguns testes chegaram a ser feitos e com sucesso no entanto esta opção foi deixada em segundo plano nesta primeira versão.

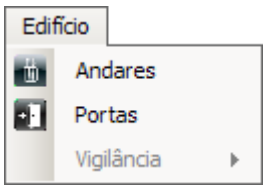


Figura 7.5 – Menu de configuração do edifício.

• Relatórios

No menu de relatórios podem-se criar listagens de todos os eventos ocorridos, dentro de um espaço de tempo escolhido, e guarda-los em formato PDF. Também existe a opção de criar um relatório diário, ficando estes guardados numa pasta predefinida.

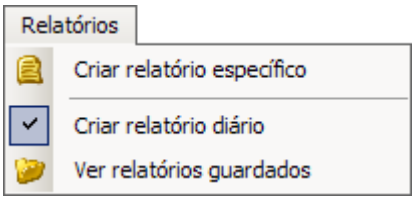


Figura 7.6 – Menu de criação de relatório de eventos.

7.4. Adicionar novos utilizadores

Para adicionar novos utilizadores ao sistema, basta utilizar o botão de atalho que se apresenta na barra de ferramentas da janela principal ou então através do menu "Utilizadores". Irá aparecer uma nova janela onde devem ser introduzidos dados pessoais

(nome, fotografia, número do bilhete de identificação, data de nascimento, etc.) e relativos à empresa (número de identificação do cartão, contacto, gabinete, etc.). Alguns campos não são obrigatórios como por exemplo a localização do gabinete, visto poder existir um utilizador que faça parte da empresa mas que não tem um gabinete próprio, como pode ser o caso de uma empregada de limpeza ou de um segurança.

Nesta parte apenas se configura a informação dos utilizadores, a configuração de acessos é feita numa outra parte da aplicação. No entanto, assim que um novo utilizador é adicionado à base de dados do sistema, é dada a hipótese ao gestor de configurar logo de seguida os pontos onde este terá acesso dentro do edifício. Se mais tarde for necessário alterar os privilégios de acesso basta aceder ao menu de configuração de acessos dos utilizadores e fazer as devidas alterações.

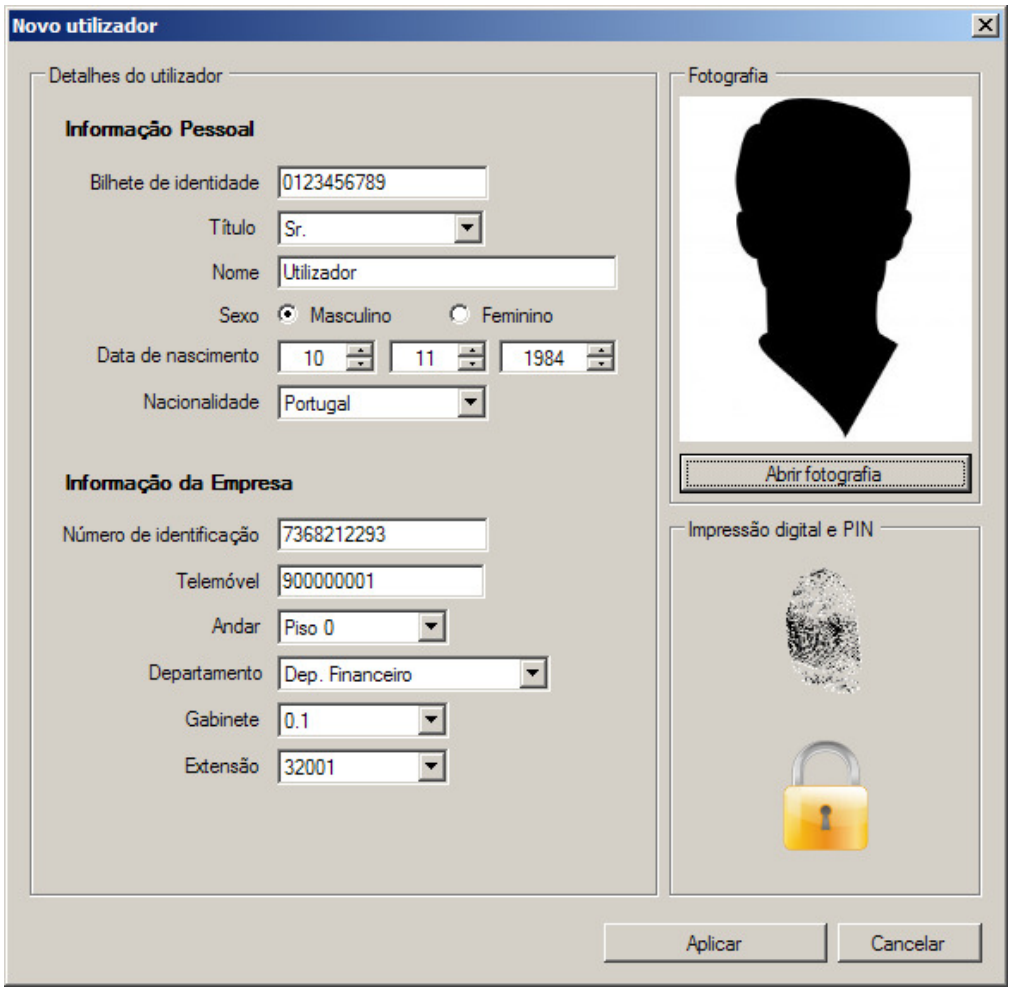


Figura 7.7 – Janela de novo utilizador.

7.5. Adicionar pontos de acesso

Ao longo do edifício existem várias controladoras que são responsáveis pela abertura das portas, e para que estas possam ser configuradas é necessário introduzi-las no sistema. O primeiro passo a dar é adicionar os andares que fazem parte do edifício. Para cada andar é dado um nome, os departamentos e gabinetes que o compõem. Os departamentos e gabinetes apenas servem para complementar a informação dos utilizadores na altura do seu registo. Para além disto também deve ser adicionada uma imagem com a planta do andar.

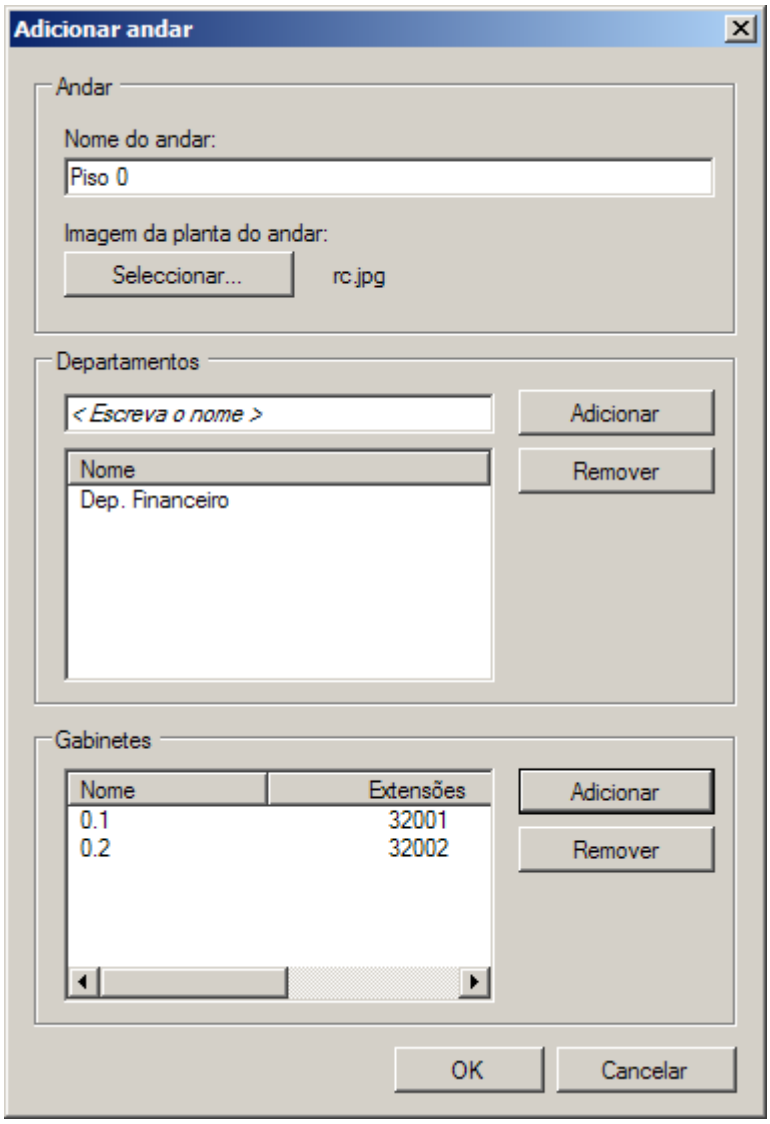


Figura 7.8 – Janela de novo andar.

Esta imagem servirá para posicionar as portas que irão ser adicionadas e ver em tempo real os eventos que nelas ocorrem (abertura e fecho de porta, abertura e fecho de trinco, abertura forçada).

Depois de adicionado o andar pode-se adicionar as portas (pontos de acesso) desse mesmo andar. Escolhendo o andar na lista, é apresentada a planta deste com um tamanho predefinido e será sobre este que iremos colocar as portas e fazer a devida configuração. A imagem seguinte mostra um andar sem qualquer porta.

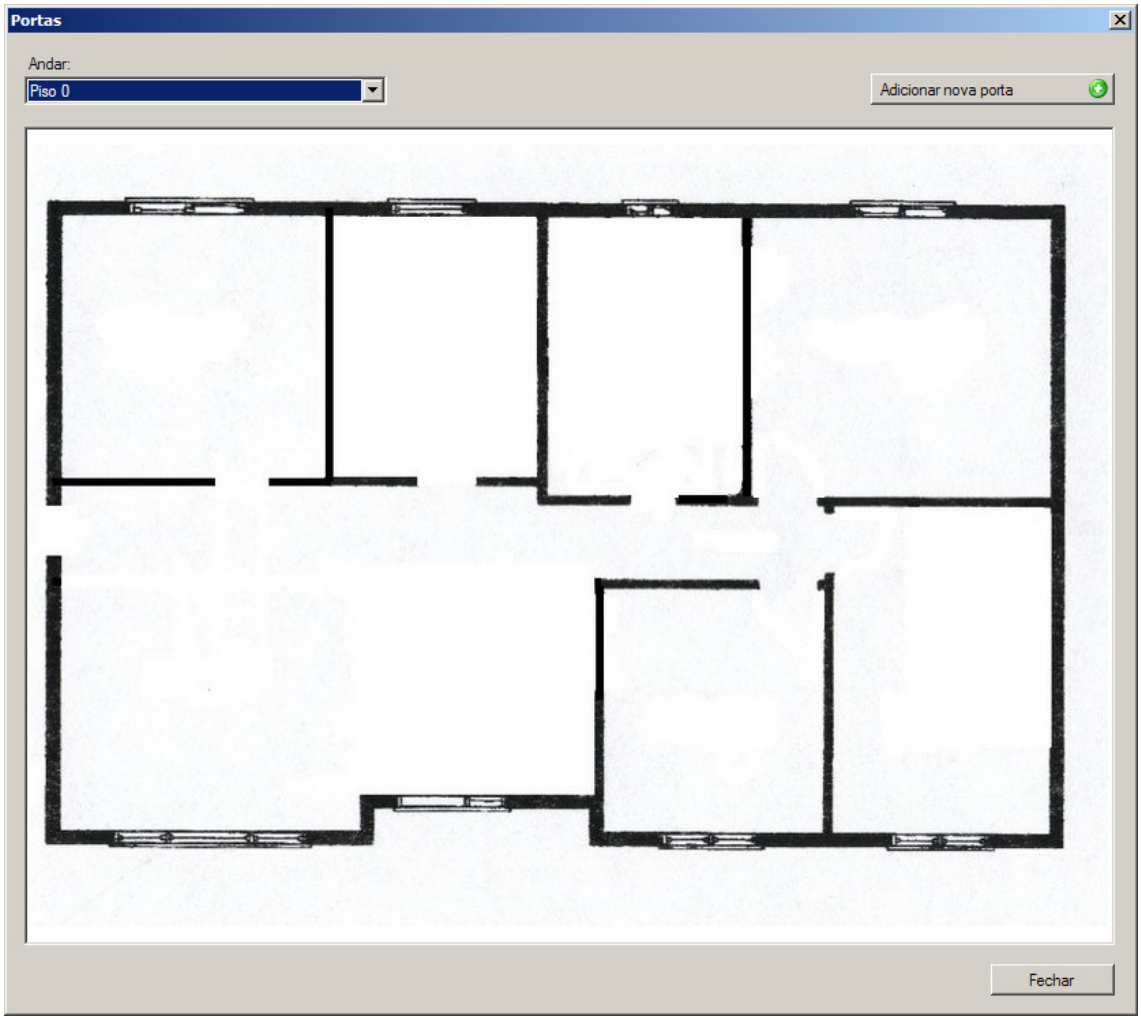


Figura 7.9 – Andar sem portas.

Para adicionar uma nova porta basta pressionar o botão com o sinal de adição "+" que está no canto superior direito e seleccionar o local onde se deseja colocar a porta. Naquele lugar irá aparecer um ícone que irá representar a porta e em seguida irá aparecer uma nova

janela onde deve ser introduzida a informação da controladora responsável por esta porta: ID da controladora, IP da controladora, porto, leitor de entrada e saída. Assim que a porta é adicionada ao sistema, o programa automaticamente começa a comunicar com a controladora responsável, podendo logo de seguida começar a receber eventos desta.

Outra coisa que se pede é o tipo de porta, isto é, é necessário indicar se se está a adicionar uma porta normal, uma porta de saída do edifício ou se é uma Arinca. A Arinca é uma máquina que tem como função guardar os cartões que dão acesso dentro do edifício, tal como já foi referido anteriormente.

Caso se queira adicionar uma porta de saída, é necessário indicar também qual a Arinca a que essa porta está associada. Isto serve para que quando alguém deixa o seu cartão numa Arinca, as portas de saída associada a ela sejam programadas para permitir a saída do edifício.

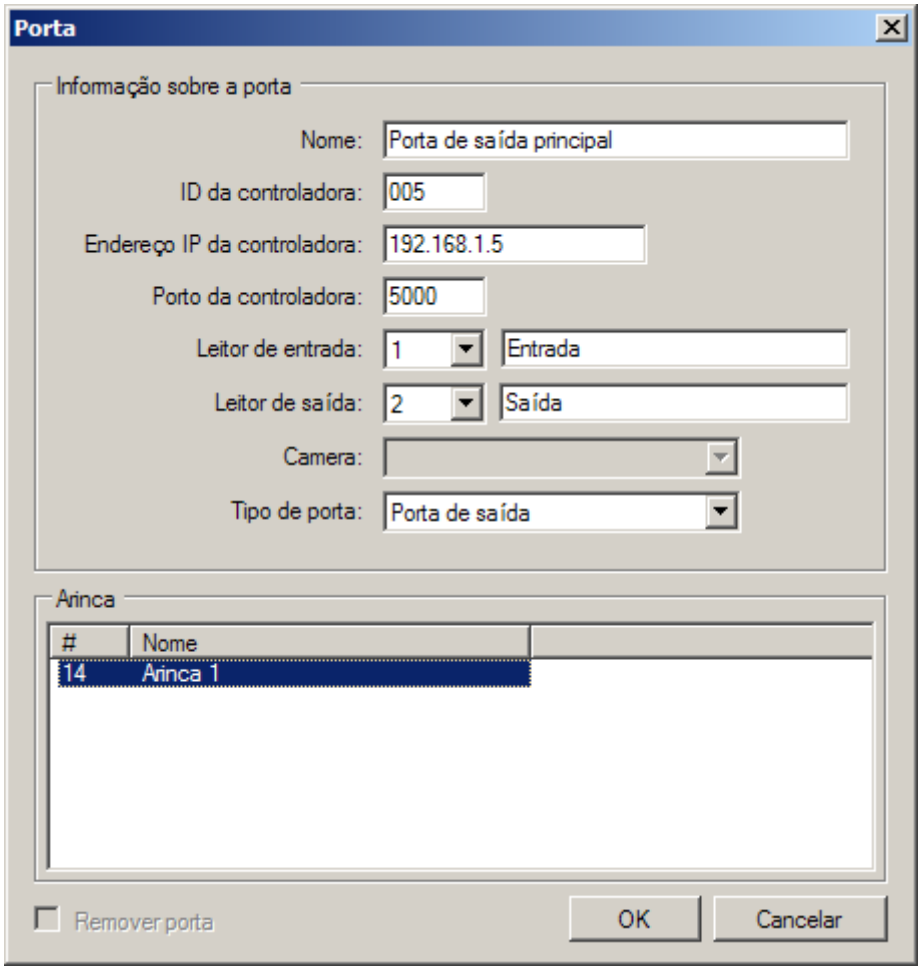


Figura 7.10 – Janela de configuração de uma porta de saída.

Após a introdução da porta ela já vai aparecer na planta do respectivo andar. Para alterar os seus dados basta clicar em cima do ícone.

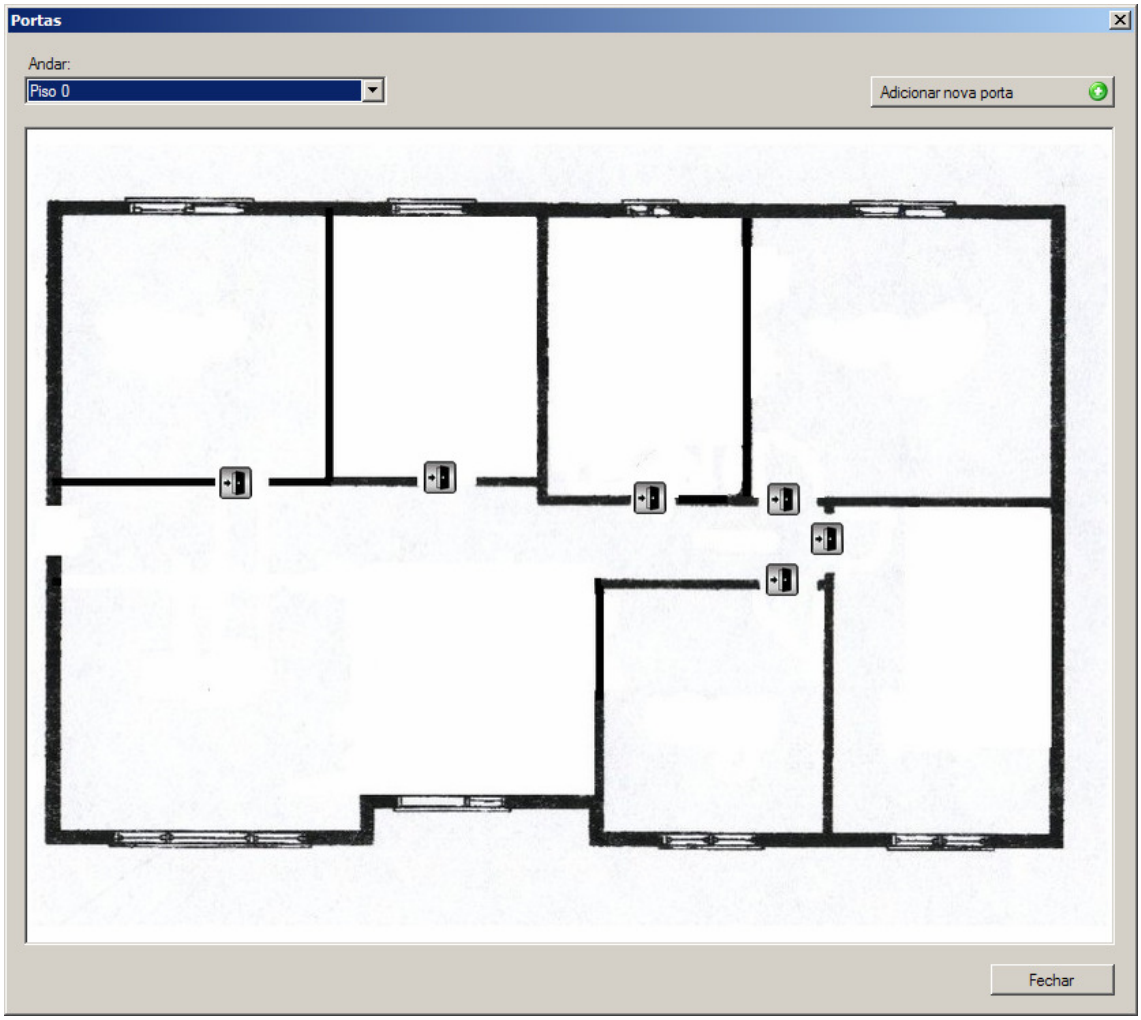


Figura 7.11 – Planta de um andar com diversas portas.

Comparando com o Starwatch PRO II foram eliminados uma série de passos intermédios para que se consiga adicionar um novo ponto de acesso. Para além do mais, no Starwatch, se se quiser adicionar uma planta para ver os eventos em tempo real é necessário fazê-lo à parte, e desta forma fica tudo configurado num único passo.

7.6. Configurar regras de acesso

Para configurar as regras de acesso deve-se aceder ao menu de utilizador e seleccionar a opção "Editar permissões", ou carregar no botão de atalho que se encontra na barra de ferramentas na janela principal. Depois de seleccionado o utilizador que se deseja configurar, irá aparecer uma janela idêntica à da figura 7.12.

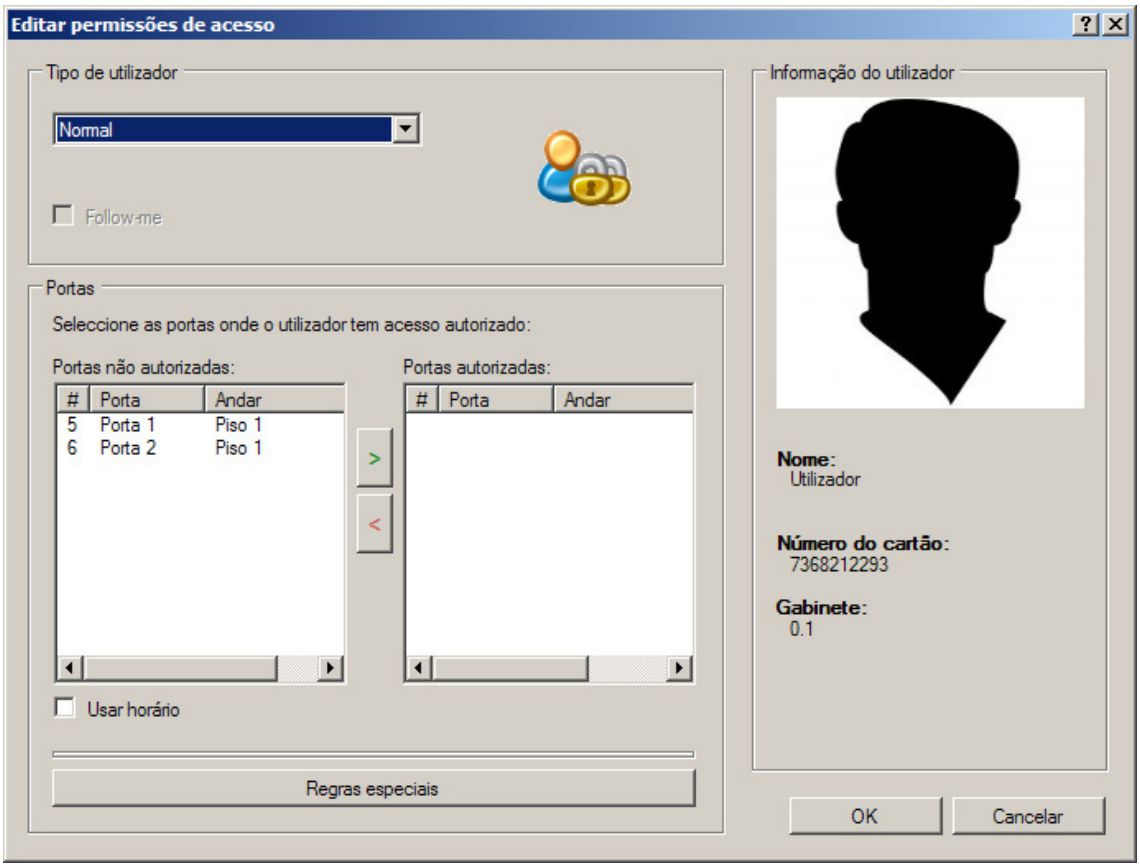


Figura 7.12 – Janela de configuração das permissões de acesso.

Cada utilizador terá que ser inserido dentro de uma categoria, pois consoante cada categoria a permissões são diferentes. Existem 3 tipos de categorias de utilizador:

- **Super** – um utilizador com esta categoria, tem acesso a qualquer ponto de acesso. Consegue abrir todas as portas, a qualquer hora, em qualquer dia, sem estar sujeito a qualquer tipo de regras;

- **Normal** – um utilizador normal apenas conseguirá abrir certas portas. Este pode ser sujeito a qualquer tipo de regra de acesso;

- **Restrito** – um utilizador restricto andará sempre em modo *follow-me*. Quando se escolhe esta opção, é necessário escolher quem será o anfitrião do utilizador, e neste modo o utilizador apenas terá acesso na última porta aberta pelo anfitrião.

Tal como foi dito acima, apenas um utilizador Normal está sujeito a regras de acesso, visto que um utilizador Super pode aceder a qualquer porta, independentemente da hora, e um utilizador Restrito está sempre sujeito ao comportamento do seu anfitrião. Para definir as portas onde um utilizador normal pode ter acesso basta "transferir" os pontos de acesso desejados da lista de portas não autorizadas para a lista das portas autorizadas. Caso se queira limitar esse acesso num determinado horário, basta seleccionar a opção "Usar horário" e escolher o horário de acesso pretendido. Para além disso, ainda é possível atribuir regras de acesso especiais, tal como a regra de apenas poder entrar num gabinete caso a pessoa a quem esse gabinete pertence estiver presente. No capítulo seguinte ir-se-á abordar este assunto com mais detalhe.

7.6.1. Desenvolvimento de regras especiais de acesso em XML

Tal como já foi dito, a ideia de criar uma lista de regras em XML tem o objectivo de tornar uma aplicação de controlo de acessos mais flexível e ajustável às necessidades do cliente. Ao ter uma lista de regras que é parte integrante do software mas que não está embutido neste, torna mais fácil o seu acesso e a sua edição. Como iremos ver mais adiante, é necessário ter conhecimentos não só de XML mas também da estrutura da base de dados do programa para conseguir criar uma regra capaz de funcionar. Ou seja, o cliente muito possivelmente não será capaz de criar regras sempre que o desejar. Para tal terá que contactar alguém com conhecimento suficiente que possa criar a regra para si. Esta situação pode ser

encarada como uma desvantagem ou como uma vantagem. Desvantagem para o cliente que sempre que quiser uma nova regra terá que pedir (e possivelmente pagar), vantagem para o criador que, para além de ser uma oportunidade para receber um pouco mais pelo "add-on", tem de certa forma um maior controlo sobre a sua aplicação e o seu trabalho.

A ideia base passa por ter um ficheiro XML que irá conter a descrição das diversas regras de acesso que irá ser validada pela aplicação. Para melhor se entender a estrutura do ficheiro de descrição XML seguidamente irá ser efectuada uma descrição pormenorizada da criação de uma regra de acesso simples.

O ficheiro XML irá ter um conjunto de regras, por isso faz todo o sentido que a etiqueta principal seja para agrupar as regras. De forma a simplificar o *parser* na aplicação acrescentou-se um atributo que indica o número de regras que estão descritas no ficheiro (ver Figura 7.12).

```
<rules size="0">
</rules>
```

Quadro 7.1 – Etiqueta que vai englobar a descrição das regras.

Dentro destas etiquetas ir-se-á colocar cada uma das regras. Como exemplo ir-se-á demonstrar a criação de uma regra que apenas permitirá acesso à porta de um gabinete, se a pessoa a quem esse gabinete pertence, estiver lá dentro. Neste caso, basta verificar onde foi a última passagem da pessoa a quem o gabinete pertence. Se essa última passagem foi no leitor de entrada da porta desse gabinete, o acesso é concedido.

Como parte da regra terão que existir alguns campos que façam a sua descrição para que se saiba na aplicação qual a regra a escolher. São portanto adicionadas duas etiquetas onde a primeira será o nome ou título da regra, e a segunda será a descrição que irá explicar o que a regra faz.

Após a descrição fica a faltar o principal: o funcionamento da regra. Para tal, é necessário indicar qual a porta, e quem é o "dono" do espaço. É então necessário introduzir

```
<rules size="1">
        <rule>
                <title>Autorização dada a presença</title>
                <description>Só é permitida a entrada se estiver presente a pessoa a quem o
                        gabinete pertence.
                </description>
        </rule>
</rules>
```

Quadro 7.2 – Título e descrição de uma regra.

dados para que a regra possa funcionar. Esses dados não são introduzidos no ficheiro XML mas sim na aplicação de modo a que possam ser guardados na base de dados. No entanto o ficheiro XML deve transmitir à aplicação a necessidade de obter dois argumentos de entrada para validar a regra. Desta forma, deve ser colocada na regra um campo onde irão ficar os argumentos de entrada da regra, assim como a sua descrição e tipo de argumento.

```
<rules size="1">
        <rule>
                <title>Autorização dada a presença</title>
                <description>Só é permitida a entrada se estiver presente a pessoa a quem
                        pertence o espaço.
                </description>
                <inputs size="3">
                        <input name="i1" caption="Porta do gabinete" type="door"/>
                        <input name="i2" caption="Leitor" type="reader"/>
                        <input name="i3" caption="Utilizador que deve estar presente"
                        type="user"/>
                </inputs>
        </rule>
</rules>
```

Quadro 7.3 – Argumentos de entrada de uma regra.

Como se pode observar, exactamente como foi feito para as regras (*rules*), é criada uma etiqueta que vai agrupar todos os argumentos de entrada, e da mesma forma é colocado um atributo que indica o total de argumentos de entrada da regra (*input size*).

Para cada regra é indicado o nome da variável (*input name*), uma palavra representativa (que vai aparecer na aplicação) e o tipo. Neste caso o primeiro argumento irá indicar ao programa que é do tipo "*door*", ou seja a escolha deve de incidir sobre todas as portas existentes no edifício. Da mesma forma, o segundo argumento é do tipo "*user*" o que irá fazer com que o programa mostre uma lista de todos os utilizadores disponíveis.

Tal como já foi dito, para verificar esta regra basta verificar se a última passagem de um dado utilizador foi na porta indicada. Existe um registo permanente sobre todos os utilizadores de qual foi o último ponto por onde passaram, e se esse ponto corresponder com a entrada no gabinete referido na regra então o acesso é concedido. No entanto, a aplicação necessita de saber qual a *query* que tem que enviar à base de dados. Para tal basta adicionar mais um campo no ficheiro de regras com as *queries* que têm que ser executadas para validar a regra.

```
<rules size="1">
        <rule>
                (…)
                <queries size="2">
                        <query name="q1" value="select last_door from users where id=_i3 and
                        is_active=1"/>
                        <query name="q2" value="select last_reader from users where id=_i3
                        and is_active=1"/>
                 </queries>
        </rule>
</rules>
```

Quadro 7.4 – *Queries* que vão ser usadas para validar a regra.

Exactamente como se fez para o conjunto de regras ou de argumentos de entrada, colocou-se uma etiqueta que agrupa todas as *queries* e que tem um atributo que indica o seu total (*querie size*). Neste caso apenas é necessário uma *query*, no entanto podiam ser

utilizadas muitas mais. A aplicação, ao aplicar esta *query* à base de dados, irá receber um resultado que terá que ser comparado com um dado valor. Como resultado desta comparação teremos a validação ou não validação da regra. Ou seja, para cada *query* é necessário um valor de comparação.

```
<results size="2">
    <result name="r1" value="_i1"/>
    <result name="r2" value="_i2"/>
</results>
```

Quadro 7.5 – Resultados da aplicação das *queries* da regra.

Note-se que tanto nos resultados como nas *queries* não aparecem valores. Em vez disso aparece o nome dos argumentos de entrada precedidos de um "_". Isto serve para o *parser* identificar as variáveis dentro da frase e substituí-las pelo valor que foi escolhido. Assim, se na aplicação for escolhido o utilizador com o número de identificação "123456", a *query* que será enviada para a base de dados será:

```
SELECT last_door FROM users WHERE id=123456 AND is_active=1
```

Quadro 7.6 – Exemplo de um *query* utilizando a regra.

Onde o resultado devolvido, para validar a regra, terá que ser igual ao resultado indicado, ou seja, igual ao primeiro argumento de entrada (i1). Desta forma garante-se a passagem se e só se a pessoa se encontrar dentro do gabinete.

Por último, caso a regra criada dependa das acções de outro utilizador, basta indicar no fim com a etiqueta "dependent_id" com a devida indicação de qual dos argumentos de entrada corresponde ao número de identificação do utilizador.

Como resultado final, para a regra de conceder a entrada num gabinete dada a presença do seu utilizador, obtém-se o seguinte:

```
<rules size="1">
	<rule>
		<title>Autorização dada a presença</title>
		<description>Só é permitida a entrada se estiver presente a pessoa a quem
		o gabinete pertence.</description>
		<inputs size="3">
			<input name="i1" caption="Porta do gabinete" type="door"/>
			<input name="i2" caption="Leitor" type="reader"/>
			<input name="i3" caption="Utilizador que deve estar presente"
			type="user"/>
		</inputs>
		<queries size="2">
			<query name="q1" value="select last_door from users where
			id=_i3 and is_active=1"/>
			<query name="q2" value="select last_reader from users where
			id=_i3 and is_active=1"/>
		</queries>
		<results size="2">
			<result name="r1" value="_i1"/>
			<result name="r2" value="_i2"/>
		</results>
		<dependent_id value="_i3"/>
	</rule>
</rules>
```

Quadro 7.7 – Exemplo do ficheiro XML com a descrição das regras.

Um pormenor importante a salientar é o facto de ser necessário que as *queries* e os resultados respeitem sempre a sua ordem. Ou seja, a primeira *query* deve ter o nome igual a "q1", a segunda *query* "q2", e por aí em diante. Os resultados a mesma coisa; o primeiro resultado deve ser "r1", o segundo "r2", etc. É necessário que sigam esta regra para que o resultado devolvido pela base de dados à primeira *query* seja comparado com o primeiro resultado, e a segunda *query* com o segundo resultado, etc.

A grande vantagem deste método está na facilidade com que se pode criar uma grande variedade de regras. Qualquer regra terá que ter em conta certas condições no sistema, isto é,

na regra acima criada para que o acesso fosse concedido era necessário que a última passagem de um outro utilizador fosse a entrada no seu gabinete. No entanto, podem-se adicionar outras condições tais como "o chefe ainda não entrou ao serviço", e para isso basta verificar se o cartão do chefe já saiu da maquina dispensadora de cartões. Seja qual for a condicionante da regra é possível de verificar através de registos guardados na base de dados. O que não é possível é criar regras estáticas (regras que são criadas no código fonte do programa) e garantir que todas elas irão satisfazer as exigências do cliente.

7.7. Ferramentas de desenvolvimento

Neste capítulo encontram-se descritas, de forma muito sucinta, as ferramentas utilizadas para o desenvolvimento do programa de controlo de acessos, quais as diferenças em relação a outras opções e o porquê da sua escolha na realização deste trabalho.

7.7.1. Microsoft Visual Studio .NET e a plataforma .NET

O Microsoft Visual Studio .NET é uma ferramenta de desenvolvimento integrada que permite desenvolver aplicações em modo gráfico ou consola, assim como páginas, aplicações ou serviços web, sendo possível executá-las em plataformas Windows (Windows XP, Windows Vista, Windows 7, Windows CE, Windows Mobile) e .NET (Framework .NET, Compact Framework .NET). O Visual Studio tem diversas funcionalidades que tornam o desenvolvimento de uma aplicação mais rápida e simples, como é o caso de escrita inteligente, recriação de código e outras ferramentas para a criação de aplicações gráficas.

A plataforma .NET é de facto importantíssima não só pela sua profundidade tecnológica, que permitiu quebrar barreiras de integração, mas também pela abrangência às diversas linguagens de programação existentes. À semelhança do que já existia com o Java onde as aplicações JAVA correm sobre uma máquina virtual, qualquer aplicação desenvolvida com o Visual Studio .NET irão correr sobre uma espécie de máquina virtual da plataforma .NET, que irá gerir todos os requisitos necessários à sua execução. Esta é denominada por *Common Language Runtime* (CLR), e providência importantes serviços tais

como segurança, gestão de memória e tratamento de excepções. Além disso a plataforma .NET oferece um conjunto de bibliotecas estandardizadas que permitem acesso a ficheiros, acesso a bases de dados, criação de listas, manipulação de ficheiros (TXT ou XML), criação de interfaces, entre outros.

7.7.2. A linguagem C#

O C# é uma linguagem de programação introduzida pela Microsoft com o objectivo de tornar a programação mais simples e produtiva, à semelhança do Visual Basic, sem no entanto perder o poder que é característico do C++. Eis algumas características principais desta linguagem[30]:

- **Orientada aos componentes**: Um dos grandes avanços na engenharia do software foi a criação do conceito de "componente". Um componente é uma unidade de código que pode ser incluída numa aplicação, como se se tratasse de uma peça que foi construída à parte e que facilmente se encaixa no nosso programa. O C# incluí características de programação e de desenvolvimento de componentes directamente na linguagem tornando-a muito prática tanto para a construção de aplicações baseadas em componentes, como para o desenvolvimento dos próprios componentes.

- **Robusta e moderna**: O C# é uma linguagem orientada aos objectos, possuindo mecanismos como: *garbage collection*, que liberta o programador da gestão explicita da memória; excepções, que permitem uma gestão robusta dos erros nos programas; gestão de versões de módulos, que permite que as classes e os programas evoluam ao longo do tempo; e introspecção, que permite determinar os tipos dos objectos dinamicamente e realizar conversões entre eles. Estas características e muitas outras permitem ao programador construir aplicações robustas e de uma forma muito mais segura do que tipicamente acontece com linguagens como o C++.

- **Familiar**: O C# baseia a sua sintaxe na linguagem C++, Visual Basic e, em certa medida, na linguagem Java, tornando a transição para o C# bem mais cómoda e rápida.

7.7.3. Microsoft SQLServer

O SQL Server da Microsoft é um servidor de bases de dados relacionais que vem integrado com o seu programa gestor que nos permite criar, alterar e apagar dados, a partir de qualquer computador, em qualquer parte, a qualquer altura. Este oferece os mais altos níveis de segurança, fiabilidade e escalabilidade em qualquer tipo de aplicação. Além do mais este oferece todas as bases de interoperabilidade com o Visual Studio .NET, que torna a sua utilização a partir da aplicação desenvolvida bem mais simples. Esta é uma das vantagens para a utilização do SQL Server, no entanto existem outras que o tornam na solução preferencial para aplicações na área de controlo de acessos:

- **Escalabilidade**: O SQL Server já foi colocado à prova com as mais exigentes aplicações de gestão de base de dados sendo capaz de suportar até 400 TB de dados.

- **Performance**: Este consegue também endereçar até 256 núcleos de CPU (que é o limite imposto pelo Windows Server 2008 r2) onde o MySQL apenas consegue endereçar até 4 núcleos[31].

- **Segurança**: O SQL Server vem equipado ferramentas que garantem uma ligação e autenticação segura, forte encriptação de chaves e dados.

8. Conclusão

A segurança é algo que é bastante prezado por qualquer ser humano, seja na defensa de informação, bens ou a sua própria integridade física. Os métodos para, por exemplo, garantir que uma transacção electrónica não falhe e que não permita fuga de informação, sofrem com o passar do tempo melhorias de forma a garantir sempre o máximo de segurança. No caso do Gabinete Nacional de Segurança foram pedidas melhorias no controlo de acessos, no entanto o software instalado anteriormente não suportava tais exigência, devido à sua inflexibilidade na configuração de regras. Cedo se percebeu que era algo que afectava a grande parte dos softwares, o que tornava a lista de possíveis configurações bastante curta, e incapaz de satisfazer os desejos de clientes mais exigentes. A aplicação desenvolvida para além de usar o método tradicional de configuração de acessos (em que se escolhe para cada cartão, quais as portas que este tem acesso) apresenta uma solução que permite a criação de diversas regras, de acordo com os desejos do cliente.

A aplicação desenvolvida não se encontra no seu estado final. Pela frente ainda existe muito trabalho, testes e melhorias a fazer. Tem algumas limitações em relação aos outros softwares de controlo de acesso que foram aqui analisados. Não dispõe, por exemplo, de um *display* onde mostre em tempo real as acções que estão a decorrer, como o fecho e abertura de portas. No entanto, são questões que com mais tempo e dedicação ficariam resolvidas. Não convém esquecer que este tipo de software é desenvolvido por uma equipa de programadores, grande parte deles já com um elevado conhecimento e experiência no ramo. No entanto, a principal "arma" deste software está realmente na versatilidade ao permitir um vasto leque de possibilidades de regras de acesso. Essa é a principal inovação em relação a qualquer outro software de controlo de acessos.

Este, durante o seu desenvolvimento, sofreu várias alterações de forma a tentar encontrar a melhor solução para implementar a configuração das regras especiais de acesso.

Inicialmente pensou-se em tornar as decisões de abertura de porta totalmente centralizada, ou seja, a cada evento o computador iria verificar se existiam permissões de acesso e, com base nos registos, abria ou não as portas. O problema desta solução é o facto de a resposta não ser imediata. Como é um caso especial, a resposta de abertura da porta não virá da controladora mas sim do computador central. Essa tarefa vai introduzir atraso na resposta visto que o software terá primeiro que fazer a validação da regra, caso ela exista. É impossível fazer a validação das regras especiais na controladora visto que esta não foi desenhada para desempenhar tais funções. O processamento local é bastante mais simples e limitado.

Entretanto surgiu uma solução que anulou por completo esta latência na resposta de abertura da porta. Pegando no exemplo já conhecido de apenas dar entrada se a pessoa a quem o gabinete pertence estiver presente, quando se passa o cartão pelo leitor o que o software recebe é um evento de que não foi permitida a abertura da porta. É feita uma busca para verificar se essa pessoa que passou o cartão tem essa regra especial para essa porta e se tiver é feita a sua validação. Em caso positivo a porta é aberta através de um comando que é enviado pelo software para a respectiva controladora. O que se pode fazer para anular o atraso da validação da regra é fazer a essa mesma validação antes. Ou seja, quando o "dono" entra no seu gabinete, é feita uma busca pela lista de regras e verifica se algum cartão tem uma regra especial para aquela porta e para aquele "dono". Se sim, então a controladora é programada para permitir a abertura com esses cartões. Deste modo, quando um desses cartões a quem foi alterada a permisão de acesso passar pelo leitor, a abertura será imediata visto que a controladora já se encontra programada para abrir na presença desses mesmos cartões. Quando o "dono" abandona o seu gabinete esses cartões, que estavam autorizados a entrar mediante a sua presença, são "desautorizados" deixando de poder abrir a porta desse gabinete.

No entanto esta solução também tem um contra. Na primeira solução as regras especiais só eram pesquisadas em caso de uma não-abertura de porta, mas neste caso cada vez que se dá um evento é necessário verificar se esse evento vai despoletar mudanças nas autorizações de acesso de outros cartões. Esta constante verificação (à base de dados) ajuda a sobrecarregar o sistema. No entanto, este atraso é bastante ínfimo quando comparado com a operação de reconfiguração das controladoras. Configurar uma controladora para permitir a entrada de um dado cartão, utilizando as controladoras da IDTeck, demora cerca de 1 segundo. Imagine-se agora uma regra especial que irá despoletar a permissão de acesso de 50 cartões. Durante este tempo a aplicação poderá deixar de responder, e é normal que isso

aconteça. Só quando as operação de reconfiguração de todas as controladoras terminar é que esta pode voltar a receber novos pedidos, o que pode introduzir algum atraso na resposta e aplicação da regra.

Esta é uma questão que muito dificilmente ficará resolvida por uma simples razão: as controladoras (da IDTeck) não estão preparadas para serem constantemente reprogramadas, logo é natural que este processo não seja o mais eficiente. No entanto, foi a melhor, e possível, solução que se encontrou para implementar utilizando o equipamento disponível. Com um conjunto software-hardware especificamente desenhado para o efeito com certeza que se obteriam melhores resultados em termos de performance.

Outro pormenor de salientar na elaboração deste trabalho é a atitude do Gabinete Nacional de Segurança que, mesmo com a existência de diversas empresas no mercado que podiam desenvolver uma solução para esta questão, aceitou recebê-la de uma empresa que em colaboração com a Faculdade de Ciências e Tecnologia, iria utilizar o desenvolvimento do software para dar apoio ao desenvolvimento desta tese de mestrado. É uma atitude que é de louvar não só pela paciência que tiveram enquanto se fizeram os testes ao software, onde por vezes teve que envolver mais que um funcionário, mas também pelo facto de terem escolhido uma empresa portuguesa. Os incentivos dentro do mercado nacional à criação de empresas direccionadas para o desenvolvimento de software são poucos, e são atitudes como a do Gabinete Nacional de Segurança que ajudam na proliferação de novas empresas e maior desenvolvimento nacional nesta área.

9. Bibliografia

1. BONSOR, Kevin and KEENER, Candace – How RFID Works [em linha]. 5 Nov. 2007. [Consult. 20 Mai. 2009]. Disponível em WWW: <URL: http://electronics. howstuffworks.com/gadgets/high-tech-gadgets/rfid.htm>. ISBN 1-55653-236-9.

2. BBC News – Ants' home search habit uncovered [em linha]. 22 Abr. 2009. [Consult. 20 Mai. 2009]. Disponível em WWW: <URL: http://news.bbc.co.uk/2/hi/uk_news/ england/bristol/somerset/8011998.stm>. ISBN 1-55653-236-9.

3. MILLER, Paul – Hitachi's RFID powder freaks us the heck out [em linha]. 14 Fev. 2007. [Consult. 21 Mai. 2009]. Disponível em WWW: <URL: http://www.engadget.com/ 2007/02/14/hitachis-rfid-powder-freaks-us-the-heck-out>. ISBN 1-55653-236-9.

4. Wikipedia [et al.] – Biometrics [em linha]. 6 Ago. 2003. [Consult. 27 Mai. 2009]. Disponível em WWW: <URL: http://en.wikipedia.org/wiki/Biometrics>. ISBN 1-55653-236-9.

5. Wikipedia [et al.] – Fingerprint Recognition [em linha]. 13 Dez. 2005. [Consult. 29 Mai. 2009]. Disponível em WWW: <URL: http://en.wikipedia.org/wiki/ Fingerprint_recognition>. ISBN 1-55653-236-9.

6. Comissão Nacional de Protecção de Dados – Princípios sobre a utilização de dados biométricos no âmbito do controlo de acessos e de assiduidade [em linha]. 26 Fev. 2004. [Consult. 30 Mai. 2009]. Disponível em WWW: <URL: http://www.cnpd.pt/bin/ orientacoes/principiosbiometricos.htm>. ISBN 1-55653-236-9.

7. VALADÃO, Renan Bernardo – Reconhecimento de Íris [em linha]. 3 Jun. 2008. [Consult. 2 Jun. 2009]. Disponível em WWW: <URL: http://www.gta.ufrj.br/grad/08_1/iris/index.html>. ISBN 1-55653-236-9.

8. DAUGMAN, John – Professor of Computer Vision and Pattern Recognition. *How Íris Recognition Works*, IEEE Trans. CSVT 14(1), pp. 21-30. Jan. 2004 – Cambridge, UK.

9. GERMAN, Eduard – Fingerprinting [em linha]. Actual. 21 Jun. 2008. [Consult. 6 Jun. 2009]. Disponível em WWW: <URL: http://onin.com/fp/fpmeritbdg.html>. ISBN 1-55653-236-9.

10. [et al.] – What sort of surface do we leave fingerprint on? [em linha]. 12 Abr. 2008. [Consult. 6 Jun. 2009]. Disponível em WWW: <URL: http://uk.answers.yahoo.com/question/index?qid=20090412050154AARnmPA>. ISBN 1-55653-236-9.

11. Wikipedia [et al.] – Minutae [em linha]. 5 Mai. 2005. [Consult. 8 Jun. 2009]. Disponível em WWW: <URL: http://en.wikipedia.org/wiki/Minutiae>. ISBN 1-55653-236-9.

12. HARRIS, Tom – How Fingerprint Scanners Work [em linha]. 24 Set. 2004. [Consult. 8 Jun. 2009]. Disponível em WWW: <URL: http://computer.howstuffworks.com/fingerprint-scanner.htm>. ISBN 1-55653-236-9.

13. BBC News – Malaysia car thieves steal finger [em linha]. 31 Mar. 2005. [Consult. 8 Jun. 2009]. Disponível em WWW: <URL: http://news.bbc.co.uk/2/hi/asia-pacific/4396831.stm>. ISBN 1-55653-236-9.

14. HARRIS, Daniel – Fingerprint Authentication [em linha]. 7 Jun. 2007. [Consult. 10 Jun. 2009]. Disponível em WWW: <URL: http://electronicdesign.com/Articles/Index.cfm?AD=1&ArticleID=15658>. ISBN 1-55653-236-9.

15. Access Control Scotland – Smart Card and Biometric Access Control Systems [em linha]. [Consult. 12 Jun. 2009]. Disponível em WWW: <URL: http://www.accesscontrol-scotland.co.uk/>. ISBN 1-55653-236-9.

16. Amano – Access Control Systems [em linha]. [Consult. 12 Jun. 2009]. Disponível em WWW: <URL: http://christy-ind.com/amano/amano_access.php>. ISBN 1-55653-236-9.

17. CardAccess 3000 – Access Control Software & Vídeo Integration [em linha]. [Consult. 13 Jun. 2009]. Disponível em WWW: <URL: http://www.cicaccess.com/software.html>. ISBN 1-55653-236-9.

18. Honeywell NStar – Access Solution [em linha]. [Consult. 14 Jun. 2009]. Disponível em WWW: <URL: http://www.basshome.com/nstar_honeywell_nstar_109009_prd1.htm>. ISBN 1-55653-236-9.

19. Synergistics Presidio – Web-Based Access Control: System Description [em linha]. [Consult. 15 Jun. 2009]. Disponível em WWW: <URL: http://www.synergisticsinc .com/?view=Web>. ISBN 1-55653-236-9.

20. Synergistics Presidio – Presídio Features [em linha]. [Consult. 15 Jun. 2009]. Disponível em WWW: <URL: http://www.synergisticsinc.com/Documents/presidio_cutsheet.pdf>. ISBN 1-55653-236-9.

21. Synergistics Presidio – Client/Server Based Access Control [em linha]. [Consult. 15 Jun. 2009]. Disponível em WWW: <URL: http://www.synergisticsinc.com/default .asp?LINKNAME=SERVER-BASED-ACCESS-CONTROL>. ISBN 1-55653-236-9.

22. IDTeck ITDC PRO II – Access Control and Time Attendance Software [em linha]. [Consult. 15 Jun. 2009]. Disponível em WWW: <URL: http://www.idteck.com/product/ product_view.asp?productcode=SI%20%20&productidx=197&category=Software&categ oryname=Access%20Control%20Software&categoryidx=14 >. ISBN 1-55653-236-9.

23. Software House – C.Cure 9000 Security and Event Management System [em linha]. [Consult. 17 Jun. 2009]. Disponível em WWW: <URL: http://www.swhouse.com/ products/software_CCURE9000.aspx>. ISBN 1-55653-236-9.

24. Tensor – Tensor Access Enterprise [em linha]. [Consult. 18 Jun. 2009]. Disponível em WWW: <URL: http://www.tensor.co.uk/access-enterprise>. ISBN 1-55653-236-9.

25. CARMELL, Tim – Spectrogram Reading – What are spectrograms? [em linha]. Actual. 19 Mar. 1997. [Consult. 25 Jun. 2009]. Disponível em WWW: <URL: http://cslu.cse.ogi.edu/tutordemos/SpectrogramReading/spectrogram.html>. ISBN 1-55653-236-9.

26. Wired – Toshiba brings facial recognition to cars [em linha]. 1 Jun. 2009. [Consult. 27 Jun. 2009]. Disponível em WWW: <URL: http://www.wired.com/autopia/2009/06/facial-recognition/>. ISBN 1-55653-236-9.

27. L-1 Identity Solutions – 3D FastPass Face Reader [em linha]. [Consult. 27 Jun. 2009]. Disponível em WWW: <URL: http://www.l1id.com/pages/404-3d-fastpass>. ISBN 1-55653-236-9.

28. Schlage – Recognition Systems [em linha]. [Consult. 30 Jun. 2009]. Disponível em WWW: <URL: http://recognitionsystems.schlage.com/products/>. ISBN 1-55653-236-9.

29. IDTeck, *Star iTDC Intelligent Multi Door Access Control Panel User's Manual*, Ago. 2006, Korea.

30. MARQUES, Paulo e PEDROSO, Hernâni. *C# 2.0*, 4ª edição, FCA, 2005.

31. FRYER, Andrew – SQL Server 2008 supports how many cores? [em linha]. 7 Nov. 2008 [Consult. 2 Jul. 2009]. Disponível em WWW: <URL: http://blogs.technet.com/andrew/archive/2008/11/07/sql-server-2008-supports-how-many-cores.aspx>. ISBN 1-55653-236-9.

32. ZHLAB – Pyramid Séries Wiegand Data Format [em linha]. [Consult. 6 Jul. 2009]. Disponível em WWW: <URL: http://www.zhlab.cn/technique/T0000006.htm>. ISBN 1-55653-236-9.

33. PEACOCK, Craig – Beyond Logic – Interfacing the Serial/RS-232 Port [em linha]. 15 Jun. 2005 [Consult. 11 Jul. 2009]. Disponível em WWW: <URL: http://www.zhlab.cn/technique/T0000006.htm>. ISBN 1-55653-236-9.

10. Anexos

10.1 Anexo I – Diversas soluções de software da IDTeck

IDTECK

STARWATCH
Software Selection Guide

Last updated in February, 2009

Product	STARWATCH ENTERPRISE	STARWATCH STANDARD	STARWATCH TIME PRO	STARWATCH DUAL PRO I	STARWATCH iTDC PRO I	STARWATCH DUAL PRO II	STARWATCH iTDC PRO II	STARWATCH LX ACCESS PRO II
Controller	iMDC	All IDTECK Controllers	iTDC Series FINGER007 Series FGR007 Series 505R Series iCON100 Series LX007 Series LX505 Series	iCON100 Series FINGER007 Series FGR007 Series 505R Series	iTDC Series	iCON100 Series FINGER007 Series FGR007 Series 505R Series	iTDC Series	LX007 Series LX505 Series
General								
Server Operating System	Windows 2000 / 2003 Server	Windows 2000 / 2003 Server	Windows 2000 / 2003 Server	Windows 2000 / 2003 Server	Windows 2000 / 2003 Server	Windows 2000 / 2003 Server	Windows 2000 / 2003 Server	Windows 2000 / 2003 Server
Client Operating System	Windows 2000 Pro / XP / Vista	Windows 2000 Pro / XP / Vista	Windows 2000 Pro / XP / Vista	Windows 2000 Pro / XP / Vista	Windows 2000 Pro / XP / Vista	Windows 2000 Pro / XP / Vista	Windows 2000 Pro / XP / Vista	Windows 2000 Pro / XP / Vista
Serial Communication	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
TCP/IP Communication	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Central Alarm Monitoring	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Central Database Management	Yes	Yes	Yes	No	No	Yes	Yes	Yes
Online Help Support	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Multi Language Support	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Redundant Server/Host	Yes	No	No	No	No	No	No	No
Web Browser Interface	No	No	Yes (Report Only)	No	No	No	No	No
System Capacity								
Maximum Client Workstation Definable	Unlimited	Unlimited	Unlimited	No	No	20 Clients	20 Clients	20 Clients
Maximum Cardholders	Umlimited	50,000 (iTDC)	50,000 (iTDC)	10,000 (iCON100)	50,000	10,000	50,000	20,000
Maximum Access Levels	Umlimited	Umlimited	999	999	999	999	999	999
Maximum Readers	255*255*32	255*99*4 (iTDC)	255*99*4 (iTDC)	32*99*2	32*99*4	32*99*2	32*99*4	32*99*2
Maximum Supervised Inputs	255*255*128	255*99*15 (iTDC)	255*99*15 (iTDC)	32*99*5	32*99*15	32*99*5	32*99*15	32*99*5
Maximum Outputs	255*255*128	255*99*15 (iTDC)	255*99*15 (iTDC)	32*99*5	32*99*15	32*99*5	32*99*15	32*99*5
Maximum Holidays	68*128	15*100 (iTDC)	15*100 (iTDC)	10*32	15*100	10*32	15*100	10*100

Access Control Specifications								
Visitor Management	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Duress Mode	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Anti-passback Protection (Zone)	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Elevator Control	No	No	No	No	Yes	No	Yes	No
Two Person Rule Access Control	Yes	Yes	No	No	Yes	No	Yes	Yes
Intercom System Interface	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Video Surveillance Interface	No	No	No	Yes	Yes	Yes	Yes	Yes
Fire/Burglar Alarm System integration	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Multiple Access Group	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Alarm for Power Failure	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Rolling Transaction Display	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Guard Tour	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Intrusion Zones	Yes	No	No	Yes	Yes	Yes	Yes	Yes
Occupancy Restriction	Yes	No	No	Yes	Yes	Yes	Yes	Yes
Asset Management 3rd Party Support	No	No	No	No	No	No	No	No
Building Lighting/HVAC Control Interface	No	No	No	No	No	No	No	No
Interface to Alarm Panels via ASCII text	No	No	No	No	No	No	No	No
Portable Readers	No	No	No	No	No	No	No	No
Cardholder Database								
Relational Database	MS-SQL 2005	MS-SQL 2005	MS-SQL 2005 / ORACLE 10g	MS ACCESS 97	MS ACCESS 97	MS-SQL2005	MS-SQL2005	MS-SQL2005
Database Import/Export	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Database Partitioning	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Search on any Field	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Automatic Card Activation/Deactivation	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Support for Badge Status (Lost, etc.)	Yes	Yes	No	No	No	Yes	Yes	Yes
Bulk Card Addition/Deletion	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Automatic Download to Controllers on Changes	No	No	No	No	No	No	No	No
Support of Multiple Badges per Person	Yes	Yes	No	No	No	No	No	No
Disability Designation for Extended Shunt	Yes	Yes	No	No	Yes	No	Yes	Yes
Extended Shunt by Cardholder & Door	Yes	Yes	No	No	Yes	No	Yes	Yes
Audit Trail	Yes	Yes	No	No	No	No	No	No
Controller Administration / Features								
Networking Protocols	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Self Diagnostic	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Embedded Processor	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Scalable Memory	Yes	No	No	No	No	No	No	No
Maximum Event Buffers per Controller	200,000	14,000 (iTDC)	14,000 (iTDC)	26,000 (FINGER007)	14,000	26,000 (FINGER007)	14,000	20,000
Flash Memory	Yes	Yes	Yes	No	Yes	No	No	No
Maximum Readers per Controller	32	4 (iTDC)	4 (iTDC)	2	4	2	4	2
Onboard Ethernet	Yes	Yes (iTDC / LX007 / LX505)	Yes (iTDC / LX007 / LX505)	No	Yes	No	Yes	Yes
Web-browser Remote Access / Diagnostics	No	No	Yes (Report Only)	No	No	No	No	No
Multi Format Support	Yes	No	No	No	No	No	No	No
Operating System	Embedded Linux	Proprietary	Proprietary	Proprietary	Proprietary	Proprietary	Proprietary	Proprietary

USB Slot	Yes	No	No	No	No	No	No	No
Dual Path Redundant Communications	Yes	No	No	No	No	No	No	No
Encryption Algorithm	3DES	3DES	3DES	3DES	3DES	3DES	3DES	3DES
Peer-to-Peer Communications	Yes	No	No	No	No	No	No	No
Fingerprint Capture	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Reporting								
Event /Alarm History Report	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Tracking & Mustering Report	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
User Design Report	Yes (Reporting Tool Embedded)	No	Yes (Required Cystral Report)	No	No	No	No	No
Excel, PDF Report Format	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
HTML Report Format	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
ID/Daily Attendance Report	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Monthly Attendance Report	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Time & Attendance Report Preview	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Alarm Management								
Alarm Instruction to the Operator	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Dynamic Alarm Monitoring / Control from Maps	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Audible Alarm	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
E-mail	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Event for Unacknowledged Alarm	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Alarm Routing	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
SMS Service	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
Badge Imaging System								
Digital Camera Support	No	No	No	Yes	Yes	Yes	Yes	Yes
Batch Mode Printing	No	No	No	Yes	Yes	Yes	Yes	Yes
Allows Printing of User Defined Fields	No	No	No	Yes	Yes	Yes	Yes	Yes
Printing on Both Sides	No	No	No	No	No	No	No	No
Multiple Badge Template Support	No	No	No	Yes	Yes	Yes	Yes	Yes
Image Capture/Import	No	No	No	Yes	Yes	Yes	Yes	Yes
Signature Capture/Import	No	No	No	No	No	No	No	No
Time and Attendance								
Work Type Management (Personal)	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Work Type Definition	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Work Holiday Definition	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
"Time and Attendance Reader Group Definition"	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Shift Time and Attendance	No	No	Yes	No	No	No	No	No
Break Time Definition	No	No	Yes	Yes	No	Yes	No	Yes
Function Key Definition	No	No	Yes	Yes	No	Yes	No	Yes
Meal Time Management	No	No	Yes	No	No	No	No	No
Over Time Work Management	Yes	Yes	Yes	No	No	No	No	No
Other								
Import / Export Maps	Import	Import	Import	Import	Import	Import	Import	Import
Biometrics 3rd Party Support	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand
Iris Scan Support	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand
Smart Card 3rd Party Integration	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand	Yes, Wiegand

10.2 Anexo II - Características das diversas aplicações de controlo de acessos analisadas

	Access Control Scotland	Amano	Card Access 3000	Honeywell Nstar	Presidio	WAPAC	Starwatch iTDC PRO II	C•Cure 9000	Access Enterprise
Independente do sistema operativo	✗	✓	✗	✗	✓	✗	✗	✗	✗
Diferentes perfis para os utilizadores	✓	✓	✓	✓	✓	✓	✓	✓	✓
Suporte para visitantes	✗	✗	✓	✗	✗	✗	✓	✗	✗
Suporte para guarda de vigia	✗	✗	✗	✗	✗	✗	✓	✓	✗
Criação de cartões de identificação	✗	✓	✓	✗	✗	✗	✓	✓	✗
Anti-Passback	✓	?	?	?	✓	✓	✓	?	✓
Visualização de eventos em tempo-real	✓	✓	✓	✓	✓	✓	✓	✓	✓
Detecção de entrada forçada	✓	✗	✗	✗	✗	✗	✓	✗	✓
Suporte para vídeo-vigilância	✗	✓	✓	✓	✗	✗	✓	✓	✓
Associação das câmaras de vídeo com os alarmes ou eventos	✗	?	✓	✓	✗	✗	?	?	✓

Controlo de fechaduras									
Abertura automática de portas em caso de incêndio									
Fecho automático de portas em caso de roubo									
Controlo de acessos por grupo									
Controlo de acessos consoante horário									
Supervisionamento das horas de trabalho									
Notificação de eventos por SMS									
Notificação de eventos por e-mail									
Relatório de eventos detalhados									
Controlo de elevadores									
Visualização de eventos num mapa do edifício									
Disponível em diversas línguas									

Legenda:


Característica presente


Característica ausente


Característica que não é referida especificamente. Pode ou não existir.

10.3 Anexo III – Mapa de caracteres ASCII

Regular ASCII Chart (character codes 0 – 127)

000d	00h	␀	(nul)	016d	10h	►	(dle)	032d	20h	␣	048d	30h	0	064d	40h	@	080d	50h	P	096d	60h	‘	112d	70h	p
001d	01h	☺	(soh)	017d	11h	◄	(dc1)	033d	21h	!	049d	31h	1	065d	41h	A	081d	51h	Q	097d	61h	a	113d	71h	q
002d	02h	☻	(stx)	018d	12h	↕	(dc2)	034d	22h	"	050d	32h	2	066d	42h	B	082d	52h	R	098d	62h	b	114d	72h	r
003d	03h	♥	(etx)	019d	13h	‼	(dc3)	035d	23h	#	051d	33h	3	067d	43h	C	083d	53h	S	099d	63h	c	115d	73h	s
004d	04h	♦	(eot)	020d	14h	¶	(dc4)	036d	24h	$	052d	34h	4	068d	44h	D	084d	54h	T	100d	64h	d	116d	74h	t
005d	05h	♣	(enq)	021d	15h	§	(nak)	037d	25h	%	053d	35h	5	069d	45h	E	085d	55h	U	101d	65h	e	117d	75h	u
006d	06h	♠	(ack)	022d	16h	▬	(syn)	038d	26h	&	054d	36h	6	070d	46h	F	086d	56h	V	102d	66h	f	118d	76h	v
007d	07h	•	(bel)	023d	17h	↨	(etb)	039d	27h	'	055d	37h	7	071d	47h	G	087d	57h	W	103d	67h	g	119d	77h	w
008d	08h	◘	(bs)	024d	18h	↑	(can)	040d	28h	(	056d	38h	8	072d	48h	H	088d	58h	X	104d	68h	h	120d	78h	x
009d	09h	○	(tab)	025d	19h	↓	(em)	041d	29h	)	057d	39h	9	073d	49h	I	089d	59h	Y	105d	69h	i	121d	79h	y
010d	0Ah	◙	(lf)	026d	1Ah	→	(eof)	042d	2Ah	*	058d	3Ah	:	074d	4Ah	J	090d	5Ah	Z	106d	6Ah	j	122d	7Ah	z
011d	0Bh	♂	(vt)	027d	1Bh	←	(esc)	043d	2Bh	+	059d	3Bh	;	075d	4Bh	K	091d	5Bh	[	107d	6Bh	k	123d	7Bh	{
012d	0Ch	♀	(np)	028d	1Ch	└	(fs)	044d	2Ch	’	060d	3Ch	<	076d	4Ch	L	092d	5Ch	\	108d	6Ch	l	124d	7Ch	|
013d	0Dh	♪	(cr)	029d	1Dh	↔	(gs)	045d	2Dh	-	061d	3Dh	=	077d	4Dh	M	093d	5Dh	]	109d	6Dh	m	125d	7Dh	}
014d	0Eh	♫	(so)	030d	1Eh	▲	(rs)	046d	2Eh	.	062d	3Eh	>	078d	4Eh	N	094d	5Eh	^	110d	6Eh	n	126d	7Eh	~
015d	0Fh	☼	(si)	031d	1Fh	▼	(us)	047d	2Fh	/	063d	3Fh	?	079d	4Fh	O	095d	5Fh	_	111d	6Fh	o	127d	7Fh	⌂

Extended ASCII Chart (character codes 128 – 255) Latin1/CP1252

128d	80h	€		144d	90h			160d	A0h	␣	176d	B0h	°	192d	C0h	À	208d	D0h	Ð	224d	E0h	à	240d	F0h	ð
129d	81h			145d	91h	‘		161d	A1h	¡	177d	B1h	±	193d	C1h	Á	209d	D1h	Ñ	225d	E1h	á	241d	F1h	ñ
130d	82h	‚		146d	92h	’		162d	A2h	¢	178d	B2h	²	194d	C2h	Â	210d	D2h	Ò	226d	E2h	â	242d	F2h	ò
131d	83h	ƒ		147d	93h	“		163d	A3h	£	179d	B3h	³	195d	C3h	Ã	211d	D3h	Ó	227d	E3h	ã	243d	F3h	ó
132d	84h	„		148d	94h	”		164d	A4h	¤	180d	B4h	´	196d	C4h	Ä	212d	D4h	Ô	228d	E4h	ä	244d	F4h	ô
133d	85h	…		149d	95h	•		165d	A5h	¥	181d	B5h	µ	197d	C5h	Å	213d	D5h	Õ	229d	E5h	å	245d	F5h	õ
134d	86h	†		150d	96h	–		166d	A6h	¦	182d	B6h	¶	198d	C6h	Æ	214d	D6h	Ö	230d	E6h	æ	246d	F6h	ö
135d	87h	‡		151d	97h	—		167d	A7h	§	183d	B7h	·	199d	C7h	Ç	215d	D7h	×	231d	E7h	ç	247d	F7h	÷
136d	88h	ˆ		152d	98h	˜		168d	A8h	¨	184d	B8h	¸	200d	C8h	È	216d	D8h	Ø	232d	E8h	è	248d	F8h	ø
137d	89h	‰		153d	99h	™		169d	A9h	©	185d	B9h	¹	201d	C9h	É	217d	D9h	Ù	233d	E9h	é	249d	F9h	ù
138d	8Ah	Š		154d	9Ah	š		170d	AAh	ª	186d	BAh	º	202d	CAh	Ê	218d	DAh	Ú	234d	EAh	ê	250d	FAh	ú
139d	8Bh	‹		155d	9Bh	›		171d	ABh	«	187d	BBh	»	203d	CBh	Ë	219d	DBh	Û	235d	EBh	ë	251d	FBh	û
140d	8Ch	Œ		156d	9Ch	œ		172d	ACh	¬	188d	BCh	¼	204d	CCh	Ì	220d	DCh	Ü	236d	ECh	ì	252d	FCh	ü
141d	8Dh			157d	9Dh			173d	ADh		189d	BDh	½	205d	CDh	Í	221d	DDh	Ý	237d	EDh	í	253d	FDh	ý
142d	8Eh	Ž		158d	9Eh	ž		174d	AEh	®	190d	BEh	¾	206d	CEh	Î	222d	DEh	Þ	238d	EEh	î	254d	FEh	þ
143d	8Fh			159d	9Fh	Ÿ		175d	AFh	¯	191d	BFh	¿	207d	CFh	Ï	223d	DFh	ß	239d	EFh	ï	255d	FFh	ÿ

Hexadecimal to Binary

0	0000	4	0100	8	1000	C	1100
1	0001	5	0101	9	1001	D	1101
2	0010	6	0110	A	1010	E	1110
3	0011	7	0111	B	1011	F	1111

Groups of ASCII-Code in Binary

Bit 6	Bit 5	Group
0	0	Control Characters
0	1	Digits and Punctuation
1	0	Upper Case and Special
1	1	Lower Case and Special

10.4 Anexo IV – Exemplo de comunicação com a controladora

Descrição: Começa com um exemplo de polling, dando-se em seguida um evento de passagem de um cartão, abertura e finalmente o fecho da porta. As tramas estão ordenadas por ordem de envio. Por cada mensagem enviada pelo computador central existe sempre uma mensagem de resposta da controladora.

Número de identificação da controladora: 006

Número de identificação do cartão: 1140877900

Número do leitor: 1

Número da porta: 1

Origem da trama	**Conteúdo**	**Notas**
Computador **C**entral:	*[SOH]*006V90125*[CR]*	Trama de polling
Controladora:	*[SOH]*006V1111111111111111 00000000000000006<<*[CR]*	Trama de status com porta e trinco fechados
CC:	*[SOH]*006V90125*[CR]*	Trama de polling
C:	*[SOH]*006V211140877900 2009012841533220A06;?*[CR]*	Trama de evento
CC:	*[SOH]*006]10124*[CR]*	Trama de acknowledge
C:	*[SOH]*006]Z014=*[CR]*	Trama de acknowledge
CC:	*[SOH]*006V90125*[CR]*	Trama de polling
C:	*[SOH]*006V1111111111111111 **1**0000000000000006<=*[CR]*	Trama de status com porta fechada e trinco aberto(*)
CC:	*[SOH]*006V90125*[CR]*	Trama de polling
C:	*[SOH]*006V80220090128415332 5004;3*[CR]*	Trama de abertura de porta
CC:	*[SOH]*006]10124*[CR]*	Trama de acknowledge

C:	*[SOH]*006]Z014=*[CR]*	Trama de acknowledge
CC:	*[SOH]*006V90125*[CR]*	Trama de polling
C:	*[SOH]*006V11**0**1111111111111 10000000000000006<<*[CR]*	Trama de status com a porta aberta e trinco aberto(**)
CC:	*[SOH]*006V90125*[CR]*	Trama de polling
C:	*[SOH]*006V1101111111111111 00000000000000006<;*[CR]*	Trama de status com a porta aberta e trinco fechado(**)
CC:	*[SOH]*006V90125*[CR]*	Trama de polling
C:	*[SOH]*006V80220090128415332 6104;5*[CR]*	Trama de fecho de porta
CC:	*[SOH]*006]10124*[CR]*	Trama de acknowledge
C:	*[SOH]*006]Z014=*[CR]*	Trama de acknowledge
CC:	*[SOH]*006V90125*[CR]*	Trama de polling
C:	*[SOH]*006V1111111111111111 00000000000000006<<*[CR]*	Trama de status com porta e trinco fechados

(*) – O bit que se encontra a negrito é o que indica o actuador da fechadura.

(**) – O bit que se encontra a negrito é o que indica o sensor de abertura da porta.