



INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Interfaces para Acesso Remoto a Sistemas Domóticos

Bruno Moita Fazenda Beirão Monteiro

Dissertação para obtenção do Grau de Mestre em

Engenharia Informática e de Computadores

Júri

Presidente: Prof. João António Madeiras Pereira

Orientador: Prof. Renato Jorge Caleira Nunes

Vogal: Prof. Alberto Manuel Ramos da Cunha

Outubro 2010

Agradecimentos

Quero expressar o meu sincero agradecimento ao meu orientador, Prof. Renato Nunes, pelo seu interesse e acompanhamento, mesmo fora de horas, no desenvolvimento e escrita desta dissertação.

Quero agradecer à Dália Pedro pelas inúmeras sugestões e acompanhamento no desenvolvimento da aplicação e pelas revisões dos diversos textos.

Quero também agradecer à minha mãe pela revisão desta dissertação e diversas sugestões dadas.

Quero ainda agradecer ao Filipe Pereira e Pedro Ribeiro pela revisão e sugestões dadas relativas ao artigo escrito em Inglês.

Quero ainda deixar um agradecimento à empresa Safira e mais especificamente ao Paulo Dias e José Neves pela compreensão e pelo tempo disponibilizado para a realização do projecto e desta dissertação.

Resumo

A área da domótica tem evoluído ao longo dos últimos anos, registando cada vez mais aderentes às soluções de automação residencial existentes no mercado. No entanto, na sua maioria, estas soluções encontram-se limitadas ao controlo de dispositivos de uma única tecnologia domótica. O sistema domótico DomoBus, desenvolvido no âmbito de um projecto académico, tem como objectivo ultrapassar esta limitação usando uma especificação genérica dos dispositivos que podem existir num sistema. Esta permite representar qualquer habitação e distribuição de dispositivos, escondendo os detalhes das tecnologias específicas que controla.

O objectivo deste projecto é a criação de uma interface de acesso remoto para o sistema domótico DomoBus, que permita ao utilizador monitorizar e controlar todos os aspectos da sua habitação, em qualquer momento ou localização. A interface web tem ainda o objectivo de ser configurável pelo utilizador e adaptável ao tamanho e resolução de ecrã do dispositivo onde este lhe acede. No cumprimento destes objectivos foi implementada uma aplicação que disponibiliza uma interface web, acessível através de qualquer dispositivo móvel ou fixo que aceda à Internet através de um *browser*.

A interface funciona de forma idêntica nos *browsers* mais usados e permite ao utilizador configurar aspectos como favoritos, macros, modo de navegação e propriedades que se mantêm sempre visíveis no ecrã. Em adição, é possível especificar a linguagem e o esquema de cores da aplicação. Todos estes pontos foram implementados e testados junto de utilizadores, concluindo-se que o projecto foi desenvolvido com sucesso e que se atingiram os objectivos propostos para o mesmo.

Palavras-Chave: Interface Web, Domótica, DomoBus, Dispositivos, Controlo

Abstract

Domotics has shown great advances during the last few years and residential automation solutions in the market have become more and more widespread. However, the majority of these solutions are limited to controlling devices of a single domotic technology. DomoBus domotic system, developed as an academic project, aims to overcome this limitation using a generic specification of the devices that can be available in a domotic system. This specification allows the system to represent any house and device disposition, while hiding specific details of the controlled technologies.

This project's goal is creating a remote access interface to the domotic system DomoBus that enables the user to monitor and control all aspects of his home at any time or place. Another goal of this web interface is to be easily configured and adaptable to size and screen resolution of the device where it's being accessed. A web interface was developed to fulfill all these goals, accessible through any device mobile or not that connects to the Internet via browser.

The interface runs equally in the most used browsers and enables its user to define bookmarks, macros, navigation type and properties always visible in all screens. In addition, it's possible to specify the application's language and theme. All these topics were implemented and tested with users, therefore concluding that this project was developed successfully and all its goals were reached.

Keywords: Web Interface, Domotics, DomoBus, Devices, Control

Índice

| | |
|--|----|
| Introdução | 1 |
| 1.1 Motivação | 2 |
| 1.2 Contexto | 2 |
| 1.3 Objectivos | 3 |
| 1.4 Estrutura da Dissertação | 5 |
| Trabalho Relacionado | 7 |
| 2.1 Interfaces Comerciais | 7 |
| 2.2 DomoBus | 9 |
| 2.3 Web Browser | 10 |
| 2.4 Servidor e tecnologias <i>Web</i> | 11 |
| Arquitectura | 13 |
| 3.1 Base de Dados | 14 |
| 3.1.1 Habitação | 14 |
| 3.1.2 Dispositivos | 15 |
| 3.1.3 Propriedades dos Dispositivos | 16 |
| 3.1.4 Utilizadores | 17 |
| 3.2 Interface | 18 |
| 3.2.1 Componente Principal | 19 |
| 3.2.2 Componente Ajax | 20 |
| 3.2.3 Global | 21 |
| 3.3 Simulador DomoBus | 22 |
| Implementação | 25 |
| 4.1 Funcionalidades da Interface | 25 |
| 4.1.1 Sistema de Navegação | 27 |
| 4.1.2 Propriedades de dispositivos | 28 |
| 4.1.3 Permissões | 32 |
| 4.1.4 Favoritos | 32 |
| 4.1.5 Macros | 33 |
| 4.1.6 Propriedades em Destaque | 35 |
| 4.1.7 Temas | 38 |
| 4.1.8 Linguagens | 40 |
| 4.1.9 Adaptabilidade à resolução de ecrã | 41 |
| 4.1.10 Adaptabilidade do texto | 44 |

| | |
|--|----|
| 4.1.11 Especificações XML | 45 |
| 4.2 Bibliotecas da Interface | 48 |
| 4.2.1 Comunicações | 49 |
| 4.2.2 Base de Dados | 50 |
| 4.2.3 General | 50 |
| 4.2.4 General Javascript | 51 |
| 4.2.5 jQuery | 54 |
| 4.3 Simulador | 54 |
| 4.4 Diagrama de Interação | 57 |
| Avaliação da Aplicação | 59 |
| 5.1 Testes de utilização..... | 59 |
| 5.1.1 Resultados das tarefas | 60 |
| 5.1.2 Análise de resultados | 61 |
| 5.1.3 Classificação da interface pelo utilizadores | 63 |
| 5.2 Testes de desempenho | 64 |
| 5.2.1 Comunicações Servidor / Simulador | 64 |
| 5.2.2 Tamanho das páginas | 66 |
| 5.2.3 Tempo de geração de ecrã..... | 68 |
| Conclusão | 71 |
| 6.1 Trabalho Futuro | 72 |
| Referências | 75 |
| Anexo A – Formato do documento XML | 78 |
| Anexo B - Casos de Teste | 80 |

Índice de Figuras

| | |
|---|----|
| Figura 1 – Ilustração do sistema | 4 |
| Figura 2 – Alguns ecrãs do PowerHome 2 | 8 |
| Figura 3 – Home Monitor por David Fox | 9 |
| Figura 4 – Modelo do funcionamento da aplicação | 13 |
| Figura 5 – Modelo simplificado da arquitectura da base de dados em UML | 14 |
| Figura 6 – Modelo detalhado dos dados da habitação | 15 |
| Figura 7 – Modelo detalhado dos dispositivos | 15 |
| Figura 8 – Modelo detalhado das propriedades dos dispositivos | 16 |
| Figura 9 – Modelo detalhado dos dados associados aos utilizadores | 18 |
| Figura 10 – Representação dos diversos módulos do componente principal. | 19 |
| Figura 11 – Representação dos diversos módulos das páginas acedidas por <i>Ajax</i> | 20 |
| Figura 12 – Representação dos diversos sub-módulos do módulo Global..... | 21 |
| Figura 13 – Representação dos módulos do Simulador do DomoBus | 22 |
| Figura 14 – Formato das mensagens recebidas pelo Simulador DomoBus | 23 |
| Figura 15 – Estrutura global das páginas da interface | 26 |
| Figura 16 – Menu da aplicação, acessível pelo ponto 4 da figura 15. | 27 |
| Figura 17 – Navegação por divisões (esquerda) e por serviços (direita). | 27 |
| Figura 18 – Lista de dispositivos da sala (esquerda) e lista de propriedades da televisão (direita). | 29 |
| Figura 19 – Representação de propriedades do tipo enumerado. | 30 |
| Figura 20 – Propriedade escalar com um máximo de cinco valores possíveis. | 30 |
| Figura 21 – Controlo de propriedades escalares. | 31 |
| Figura 22 – Representação do bloqueio de edição pelo sistema de permissões. | 32 |
| Figura 23 – Ecrãs dos favoritos do utilizador. | 33 |
| Figura 24 – Ecrãs de algumas secções das macros. | 34 |
| Figura 25 – Ecrãs de gravação (esquerda) e edição (direita) da macro. | 35 |
| Figura 26 – Ecrãs de listagem e remoção das propriedades em destaque. | 36 |

| | |
|--|----|
| Figura 27 – Ecrãs de configuração das propriedades em destaque. | 37 |
| Figura 28 – Exemplo do aspecto de um outro tema. | 39 |
| Figura 29 – Opções de configuração em inglês. | 40 |
| Figura 30 – Interface no PDA HTC Touch. | 42 |
| Figura 31 – Ecrã das divisões na consola PSP [46]. | 42 |
| Figura 32 – Ecrã das propriedades da televisão na consola PSP. | 43 |
| Figura 33 - Ecrã das propriedades da televisão no Mozilla Firefox. | 43 |
| Figura 34 – Algumas representações de adaptabilidade do texto. | 45 |
| Figura 35 – Ecrãs de exportação e importação. | 46 |
| Figura 36 – Ecrã com nuvem em processamento de um pedido <i>Ajax</i> | 52 |
| Figura 37 – Manipulação do canal de televisão. | 53 |
| Figura 38 – Consola do Simulador (Fundo Branco) | 56 |
| Figura 39 – Interacção entre os diversos componentes do projecto | 58 |
| Figura 40 – Representação do número de cliques por cada tarefa | 60 |
| Figura 41 – Representação dos segundos despendidos por tarefa | 61 |
| Figura 42 – Representação das classificações atribuídas à interface | 63 |
| Figura 43 – Lista de propriedades da televisão em ecrã de maiores dimensões | 65 |
| Figura 44 – Tamanho das páginas de início de sessão | 67 |
| Figura 45 – Tamanho das páginas após início de sessão | 68 |
| Figura 46 – Tempo despendido na geração de ecrãs | 69 |

Lista de Abreviaturas

XML – Extensible Markup Language

TCP/IP – Transmission Control Protocol / Internet Protocol

Ajax – Asynchronous JavaScript and XML

HTTP – Hypertext Transfer Protocol

WWW – World Wide Web

PHP – PHP: Hypertext Preprocessor

ASP – Active Server Pages

HTML – HyperText Markup Language

CSS – Cascading Style Sheets

WAMP – Windows, Apache, MySQL and PHP

UML – Unified Modeling Language

IEM6 – Internet Explorer Mobile 6

Capítulo 1

Introdução

Nos últimos anos tem-se assistido a uma grande evolução da tecnologia presente em aparelhos domésticos, dispositivos móveis e redes de informação. Esta constante evolução tem tornado a área da domótica, ou automação residencial, cada vez mais diversificada, abrangente e acessível economicamente ao público em geral.

Há cerca de duas décadas atrás a maioria dos aparelhos domésticos possuíam apenas interfaces mecânicas, reduzidas e limitadas apenas a alguns botões e rodas de configuração. Estas apenas permitiam controlar as funcionalidades principais dos aparelhos, como por exemplo numa máquina de lavar roupa era possível especificar o estado (on/off) e o programa de lavagem. Hoje em dia não só podemos configurar o programa, mas também o enxaguamento, a centrifugação e a temperatura da água desejados, podendo ainda configurar a fragilidade dos tecidos e agendar o seu funcionamento para uma hora desejada. Estas novas possibilidades trazem uma maior complexidade aos aparelhos e por isso requerem interfaces mais sofisticadas.

Um outro conjunto de dispositivos disponível aos utilizadores em geral é o que se pode designar por dispositivos móveis, tais como telemóveis, *Personal Digital Assistants* (PDA), computadores portáteis e *Tablet PC's*. Estes aparelhos tornaram-se cada vez mais leves, portáteis, com maior poder de computação, maior autonomia e mais acessíveis para a generalidade da população, levando à sua distribuição em massa. Na actualidade podemos considerar que estes dispositivos se tornaram indispensáveis à actividade diária dos seus utilizadores, disponibilizando um conjunto de funcionalidades e utilitários em constante crescimento.

Em paralelo com estas evoluções, assistimos também ao crescimento de diversos tipos de redes de informação, tanto internas a uma casa ou edifício como redes globais, mais especificamente a Internet. Numa vertente interna, estas redes permitem que os diversos dispositivos existentes numa habitação se encontrem interligados entre si. Externamente à habitação, verificou-se uma redução de custo e aumento de velocidade da Internet. Esta maior acessibilidade permitiu um alargamento da Internet a praticamente todas as residências e dispositivos móveis, fazendo com que existam cada vez mais funcionalidades e conteúdos disponíveis.

A Domótica tem usufruído de todas estas evoluções na tentativa de tornar as casas cada vez mais “inteligentes” e cómodas para os seus habitantes. Actualmente existem diversas tecnologias domóticas

(entre elas o X10 [1], LonWorks [2], KNX [3], CEBus [4]) tendo como principal finalidade facilitar diversas tarefas diárias tais como controlar as luzes, a temperatura, subir e descer os estores a determinadas horas do dia, regar o jardim, controlar a segurança da sua casa. Apesar das diversas soluções existentes, estas são incompatíveis entre si o que dificulta a adesão e distribuição da domótica numa maior escala [5].

1.1 Motivação

No cenário actual, onde uma habitação é constituída por uma grande quantidade de dispositivos, todos eles com interfaces cada vez mais complexas e com funcionalidades e propriedades específicas a si próprios, surge a necessidade de um sistema domótico que permita monitorizar e controlar todas estas particularidades. Uma vez que a casa deve ser controlada pelos seus habitantes, este sistema tem que disponibilizar uma interface robusta e completa com todos os dispositivos e funcionalidades, mas ao mesmo tempo simples e intuitiva de utilizar e compreender pelo utilizador.

A possibilidade de controlo permitida por esta interface não é por si só suficiente. Tendo em conta que a maioria das pessoas passa grande parte do dia fora da sua residência e considerando a grande mobilidade destas no mundo actual, verifica-se que uma das necessidades desta interface é que esteja sempre disponível para acesso ao utilizador. Ou seja, não apenas em casa, mas em qualquer local em que o utilizador sinta a necessidade de verificar ou controlar o estado de qualquer dispositivo na sua habitação.

A interface desenvolvida no âmbito deste projecto pretende fornecer esta acessibilidade aos utilizadores permitindo que, independentemente da localização dos mesmos, estes possam monitorizar e controlar a sua casa. Com esta interface o utilizador apenas necessita de um dispositivo com acesso à Internet via *browser* e poderá estar ligado ao seu lar sempre que desejar.

1.2 Contexto

No panorama actual existem diversas tecnologias domóticas, no entanto existem algumas limitações que impedem a sua distribuição pela generalidade das habitações. Destas limitações destacam-se a incompatibilidade entre os diferentes sistemas domóticos e a inexistência de aplicações genéricas de monitorização e controlo capazes de se adaptar a novas situações, como por exemplo a adição de novos tipos de dispositivos [6]. Por fim, destaca-se ainda a necessidade de recorrer a pessoal técnico quando o utilizador quer configurar e ajustar o comportamento do sistema às suas preferências.

O sistema domótico DomoBus [7] [8] foi construído com o objectivo de superar as limitações descritas, para isso tem uma especificação genérica dos dispositivos que podem existir num sistema. Esta especificação consegue representar as propriedades e valores de dispositivos físicos de diferentes tecnologias, num único sistema. A abstracção entre os dispositivos físicos e a representação genérica do DomoBus tem a grande vantagem de disponibilizar uma tecnologia única, sobre a qual se podem desenvolver aplicações de controlo do sistema domótico, independentemente das diversas tecnologias que realmente se encontram a ser controladas. Tornando-se possível a criação de uma interface única para todos os dispositivos disponíveis numa habitação.

Na área dos dispositivos móveis também se tem assistido a diversas evoluções. Desde o aparecimento dos telemóveis que estes têm vindo a integrar cada vez mais funcionalidades e aplicações, em conjunto com uma maior autonomia que permite ao utilizador um uso destes dispositivos para inúmeras tarefas além do contacto telefónico. Em adição a este conjunto de dispositivos, as consolas portáteis também têm vindo a disponibilizar cada vez mais funções para além de jogos. Uma destas funcionalidades, que se encontra em ambas as categorias de dispositivos, é o acesso à Internet através de um *browser*; este permite aceder a qualquer página na Internet como se estivessemos a fazê-lo a partir de um computador pessoal. Desta forma, os utilizadores estão não só familiarizados com a Internet como com diversos dispositivos móveis através dos quais se podem manter ligados a qualquer conteúdo.

O projecto aqui apresentado centra-se no desenvolvimento de uma interface de acesso remoto para o DomoBus, permitindo visualizar e controlar os dispositivos de uma habitação através de telemóvel ou computador com ligação à Internet. Assim, foi criada uma aplicação que interpreta todas as informações fornecidas pelo DomoBus, como os dispositivos existentes na casa, a localização de cada um ou a forma de serem controlados e ao mesmo tempo comunica com o sistema para obter e alterar os valores actuais das propriedades dos diversos dispositivos.

1.3 Objectivos

O propósito deste trabalho é a criação de uma interface fácil e intuitiva de manipular que permita aos habitantes de uma casa, com o sistema domótico DomoBus, monitorizar e controlar todos os aspectos da sua casa “inteligente”. Esta interface deve permitir visualizar e alterar as diversas propriedades dos dispositivos disponíveis na habitação e deve ainda permitir um conjunto de configurações específicas a cada utilizador / habitação como por exemplo a criação de favoritos e macros e a validação de permissões de utilizador.

Um dos principais objectivos para esta interface é que esteja acessível ao utilizador a qualquer momento e em qualquer localização. Assim, a aplicação a desenvolver deverá estar disponível

localmente ou através de uma ligação à Internet e ser compatível com quaisquer dispositivos móveis ou fixos, permitindo ao utilizador usufruir de todas as suas ferramentas independentemente do local onde se encontra e do dispositivo utilizado.

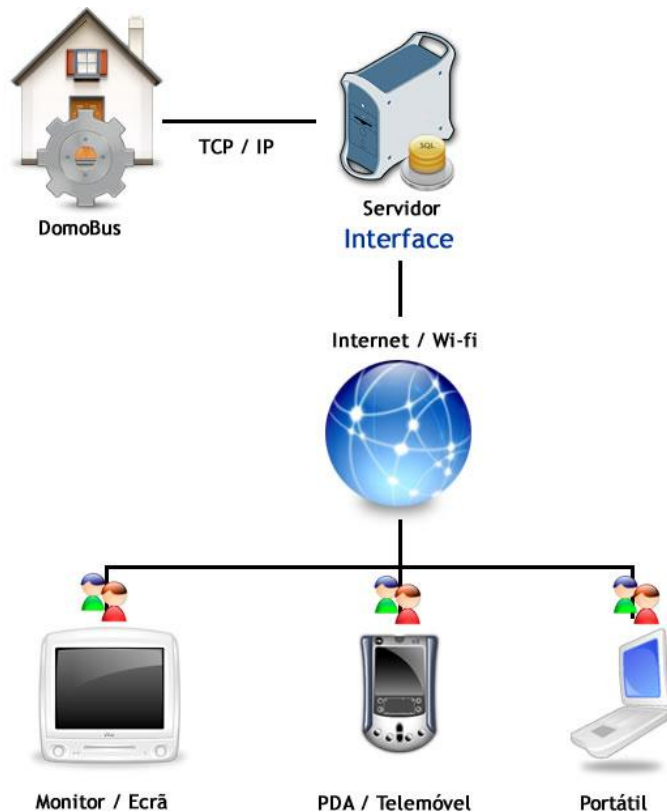


Figura 1 – Ilustração do sistema

A figura anterior demonstra a ligação entre os vários componentes deste projecto. Como se pode verificar, a interface pode ser acedida através de diversos dispositivos sendo manipulada por teclado e rato ou caneta e dedo. Estes dispositivos podem aceder ao servidor localmente (por Wi-Fi [9]) ou remotamente (por GPRS [10], UMTS [11], entre outros) Por sua vez o servidor, que implementa a interface, comunica com o DomoBus através de uma ligação TCP/IP [12].

O uso do modelo DomoBus, como base da aplicação a desenvolver, fornece uma camada de abstracção entre as tecnologias específicas usadas na automação residencial e a interface usada para controlar todas essas tecnologias. No entanto, a aplicação a desenvolver não é parte integrante do sistema DomoBus, assim para que possa deter toda a informação sobre a habitação e dispositivos disponíveis para controlo é necessário que tenha a capacidade de receber e interpretar um ficheiro XML [13], gerado pelo DomoBus [14]. Após a leitura e análise de um ficheiro XML que descreve um sistema em concreto a interface passará a conhecer a estrutura da habitação, que divisões e outros espaços possui, e a composição do sistema, dispositivos que estão disponíveis e qual a sua localização. No

acesso aos dispositivos é gerada automaticamente uma interface adequada à visualização e controlo das suas propriedades, permitindo absoluto controlo do dispositivo. Desta forma, consegue-se uma aplicação genérica e adaptável a qualquer tipo de habitação, que modela a interface a apresentar conforme a informação recebida.

O utilizador, ainda assim, não ficará restrito à interface gerada pela interpretação da especificação de um sistema DomoBus, existindo diversas particularidades passíveis de configuração. Uma destas é o modo de navegação pela aplicação, sendo possível escolher entre uma estrutura que simula a organização natural de uma habitação – divisão a divisão – e uma estruturação por serviços que agrupa dispositivos do mesmo tipo – iluminação, climatização, entre outros. Outras configurações possíveis contemplam a criação de favoritos, de macros e a definição de propriedades em destaque.

Os favoritos disponibilizam acesso directo a dispositivos pré-determinados pelo utilizador. As macros são funcionalidades pré-programadas que através de um só clique executam um conjunto de acções previamente definido pelo utilizador. Por sua vez, as propriedades em destaque podem ser quaisquer propriedades de qualquer dispositivo que o utilizador tenha escolhido para visualizar sempre o seu estado no fundo do ecrã.

O ficheiro XML recebido também caracteriza os utilizadores da aplicação e os níveis de acesso de cada um. Deste modo a aplicação irá possuir um sistema de sessões (*log in*) e de permissões. Assim, consoante o utilizador que acede à aplicação, esta deve avaliar o nível de acesso do mesmo e determinar a sua capacidade de visualização e modificação perante cada divisão e dispositivo, adaptando a interface às permissões encontradas. Todos estes dados, incluindo configurações de visualização, devem ser exportados para um ficheiro XML no mesmo formato que o gerado pelo DomoBus.

A adaptabilidade da aplicação não se resume à estrutura das habitações e aos diversos tipos de dispositivos que são apresentados; é também um objectivo que a aplicação funcione de forma similar em todos os aparelhos móveis e fixos, adaptando-se aos diversos tamanhos e resoluções de ecrã disponíveis. Consegue-se desta forma um controlo universal adaptável tanto aos dispositivos que controla como aos dispositivos que apresentam a interface de controlo, algo que nunca seria possível com um aparelho físico.

1.4 Estrutura da Dissertação

O restante documento desta dissertação segue a seguinte estrutura de capítulos. No capítulo 2 apresenta-se a análise realizada sobre as interfaces de sistemas domóticos actuais, o sistema DomoBus, os dispositivos disponíveis para acesso à Internet e a tecnologia disponível para implementar um

servidor. Depois, no capítulo 3 descreve-se a arquitectura da interface de acesso remoto para o DomoBus. Seguido do capítulo 4 onde se encontra a descrição dos detalhes de implementação da solução e das abordagens seguidas. No capítulo 6 são apresentados os resultados obtidos na avaliação qualitativa e quantitativa da interface. Por fim, no capítulo 7 apresentam-se as conclusões sobre o trabalho realizado e apresenta-se também o trabalho futuro.

Capítulo 2

Trabalho Relacionado

As interfaces são parte integrante de qualquer aplicação disponível ao público em geral e é através destas que se possibilita a interação entre o utilizador e a máquina. Dos diversos tipos de interfaces, as que se encontram disponíveis através da Internet permitem que os utilizadores lhes acedem em qualquer momento e local. Na realização de uma interface web para um sistema domótico podemos destacar quatro áreas de maior interesse: as interfaces de sistemas domóticos comerciais disponíveis hoje em dia, o sistema domótico para o qual a interface será desenvolvida - Domobus, as aplicações que permitem o acesso a estas interfaces - *web browsers* - e por fim as aplicações que permitem a disponibilização da interface em redes locais ou através da Internet.

2.1 Interfaces Comerciais

Os sistemas domóticos, sendo controlados pelos habitantes das casas onde estão instalados, tentam apresentar interfaces intuitivas e fáceis de manipular. Estas interfaces começaram por ser apenas controlos físicos como botões e interruptores, evoluindo gradualmente para o uso de aplicações em software que permitem a visualização e controlo de algumas propriedades do sistema de uma forma mais intuitiva para o utilizador. Hoje em dia estas interfaces (tais como o ActiveHomePro [15] e o INSTEON [16]) são maioritariamente acedidas dentro das próprias casas que controlam, não permitindo o controlo nem acesso aos dispositivos através da Internet.

A análise realizada sobre interfaces comerciais existentes hoje em dia encontra-se restrita a interfaces gratuitas e disponíveis na Internet. Esta situação resulta do facto da maioria das interfaces disponíveis serem propriedade de alguma empresa e por consequência só estão disponíveis após compra, o que impede o acesso directo às mesmas. No entanto foi possível constatar que a maioria das interfaces disponíveis possuem quatro características base:

- Navegação – listagem dos dispositivos organizada por divisões, adaptável a qualquer estrutura de habitação;
- Visualização e controlo dos dispositivos – o estado actual geralmente é identificado por uma imagem verde/vermelha (*on* e *off*) ou um valor numérico (por exemplo a temperatura) disponibilizando a possibilidade do utilizador o alterar;

- Macros – possibilidade de definição e execução de um conjunto de tarefas previamente definido pelo utilizador;
- Cenários (*Triggers*) – possibilidade de definição de uma condição temporar (data e/ou hora) ou de uma mudança de estado de outro dispositivo que desencadeia uma determinada acção. Por exemplo, quando o sensor de iluminação detectar que é noite o sistema manda fechar as cortinas.

As interfaces Web actualmente encontram-se em expansão devido à evolução do processamento e aumento da velocidade de acesso à Internet nos aparelhos móveis dos últimos tempos. Hoje em dia ainda poucos sistemas possuem uma interface Web e a maioria dos que possuem apenas dão acesso à visualização em tempo real das câmaras de vigilância que a habitação possui. Alguns exemplos de interfaces Web são:

PowerHome 2 [17] – embora permita acesso completo ao sistema pela Internet, é necessário instalação de uma aplicação específica para esse propósito. Esta abordagem implica a existência de uma versão desta aplicação para cada sistema operativo e dispositivo móvel com diferentes características.

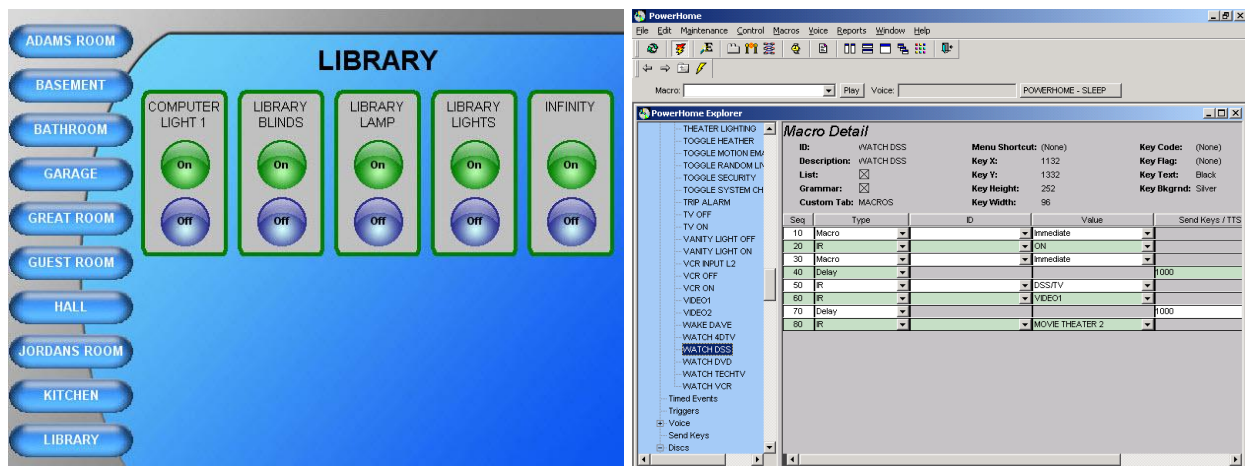


Figura 2 – Alguns ecrãs do PowerHome 2

Home Monitor [18] – trata-se de uma aplicação desenvolvida no âmbito de um concurso sem nunca chegar a ser aplicada a um sistema de automação residencial. Esta aplicação permite visualizar a planta da casa, as imagens das câmaras e controlar certos dispositivos via checkboxes, tudo através de uma aplicação desenvolvida em Java 2 Micro Edition [19].



Figura 3 – Home Monitor por David Fox

iHome [20] – é uma aplicação desenvolvida para o iPhone [21] que permite controlar dispositivos X10 remotamente e programar tarefas diárias com base em temporizadores. Esta aplicação utiliza a interface disponibilizada pela Apple para a construção de aplicações para os seus dispositivos [22].

A primeira interface, PowerHome 2, permite um controlo sobre todos os dispositivos do sistema domótico; enquanto as duas últimas, Home Monitor e iHome, apenas permitem controlar certos dispositivos. No entanto, todas as interfaces apresentadas têm a mesma desvantagem, são específicas para um conjunto de dispositivos móveis. Estas limitações impedem um controlo absoluto através de um qualquer dispositivo com acesso à Internet.

2.2 DomoBus

O DomoBus [6] é um sistema domótico, criado no âmbito de um projecto académico, que tem como objectivos permitir interoperabilidade entre dispositivos de diferentes tecnologias, oferecer mecanismos que permitem a utilizadores comuns adaptar o comportamento do sistema às suas preferências, apresentar um baixo custo e ser escalável. Este sistema usa um modelo de dados genérico para representar todas as características de uma habitação e todos os seus dispositivos constituintes, escondendo os detalhes da tecnologia específica de cada dispositivo a ser controlado. Esta representação pode ser exportada do DomoBus usando um ficheiro *XML* [14] com todos os dados presentes no sistema, incluindo a lista de divisões, dispositivos e respectivas propriedades e ainda uma lista de utilizadores e os seus respectivos níveis de acesso.

No caso dos dispositivos domóticos o modelo de dados [8] inclui o seu nome, a divisão onde se encontra, o tipo de dispositivo, o nível de acesso de leitura e escrita e uma lista de propriedades. Cada propriedade representa uma funcionalidade do dispositivo, como por exemplo o “Canal” numa Televisão ou o “Volume” do som. A lista de propriedades pode conter um conjunto de estados (Ligado / Desligado) ou um conjunto de escalares que obedecem a determinadas regras, tais como não ultrapassar um valor mínimo ou máximo, entre outras.

O sistema DomoBus suporta comunicações baseadas no protocolo *TCP/IP* permitindo que uma aplicação possa monitorizar ou alterar o estado de uma propriedade de um dispositivo. As mensagens trocadas apresentam um formato simples. Em caso de leituras basta especificar os identificadores do dispositivo e da propriedade; para realizar modificações, além dos identificadores referidos, é ainda necessário indicar o novo valor pretendido. As comunicações são fiáveis suportando mensagens de resposta de confirmação ou de erro, conforme a alteração tenha sido ou não efectuada com sucesso.

A interface a desenvolver irá interpretar o ficheiro *XML* com a especificação de um sistema e construir toda a estrutura nele definida. Este processo é efectuado apenas uma vez ou sempre que existirem alterações à sua estrutura. Uma vez importada a especificação, a aplicação gera a interface correspondente e permite consultar e alterar as diversas propriedades dos dispositivos sempre que desejado. Desta forma, será desenvolvida uma aplicação mas esta permitirá monitorizar e controlar qualquer sistema domótico independentemente da sua tecnologia base.

2.3 Web Browser

Um *Web Browser* [23], também conhecido por Navegador de Internet, é uma aplicação de *software* que obtém e apresenta informações directamente da Internet. Esta permite visualizar páginas *Web*, imagens, vídeos e outros tipos de conteúdo. Devido ao constante crescimento da Internet e na procura de melhorar e apresentar novas formas de acesso aos dados estas aplicações têm sofrido constantes actualizações e melhoramentos.

Há alguns anos atrás os *web browsers* estavam apenas disponíveis para navegação na Internet através de um computador. No entanto nos últimos anos já é possível aceder e navegar através de telemóveis, consolas e até mesmo televisões. Esta evolução fez surgir a necessidade de um novo conceito de *browser* que se adequasse às diferentes dimensões e resoluções do ecrã, à velocidade de ligação e à capacidade de processamento do dispositivo usado para aceder à Internet.

No cenário actual, com toda a diversidade de *browsers* disponível e onde se pretende que todos eles mostrem da melhor forma os conteúdos a que acedemos, houve a necessidade de construir normas comuns que todos sigam. Actualmente a maior diferença reside nas edições *Mobile*, para dispositivos móveis, onde temos alguns *browsers* que mantêm o mesmo comportamento que num PC e outros que tentam transformar as informações de forma a serem adequadas ao aparelho e ligação. No entanto, estas diferenças são cada vez menores dado que quer as ligações à Internet quer os aparelhos móveis são cada vez mais rápidos.

As páginas *web* fazem uso de diversas tecnologias. O Ajax (*Asynchronous Javascript and XML*) [24] é uma delas que surgiu para mitigar limitações de tráfego existentes e com o intuito de criar páginas mais

interactivas. Esta técnica permite que um *browser* interaja com um servidor sem necessidade de se mudar ou recarregar totalmente todas as informações de uma página. Apenas o *browser* tem conhecimento do uso desta tecnologia, pois para o servidor um pedido iniciado por *Ajax* é igual a qualquer outro pedido *HTTP* [25] que o mesmo receba. A título de exemplo, suponha-se um jornal electrónico onde uma página mostra os cabeçalhos das últimas 10 notícias. Quando se clica num dos cabeçalhos abre-se uma secção, por baixo do respectivo cabeçalho, com o conteúdo da notícia. Neste caso, o *Ajax* permite que com o clique se carregue apenas o conteúdo da notícia escolhida. Sem *Ajax* seria necessário carregar novamente toda a página, onde apenas uma pequena parte era nova ou, em alternativa, no primeiro acesso a esta teria de se carregar todo o conteúdo das 10 notícias, mesmo que o utilizador não chegasse a escolher nenhuma para ler.

Uma vez que neste projecto se pretende uma interface que seja acedida através da Internet e que possa ser executada em qualquer dispositivo, optou-se por desenvolvê-la como uma página *Web* tirando partido da grande divulgação dos *browsers* e da sua constante evolução. A aplicação será testada em diversos dispositivos tais como computador pessoal, PDA da marca *HTC* [26] e *iPhone*, usando diferentes *browsers* tais como *Internet Explorer Mobile* (v6), *Internet Explorer* (v7 e 8) [27], *Mozilla Firefox* (v3) [28] e *Safari* (v5) [29]. Esta variedade de testes tem o intuito de garantir a compatibilidade da interface com os *browsers* dos diversos dispositivos, isto porque cada *browser* implementa diferentes motores de *Javascript* [40] e por vezes apresentam limitações no uso das funcionalidades de *Ajax*. Apesar de algumas diferenças particulares entre *browsers*, todas as tecnologias utilizadas neste projecto são compatíveis com os *browsers* criados desde 1996, altura em que o *Ajax* foi implementado no *Internet Explorer*.

2.4 Servidor e tecnologias *Web*

Um Servidor *Web* é um programa de computador que disponibiliza conteúdos, tais como páginas *Web*, através da *World Wide Web* [30]. Este tipo de aplicação, em conjunto com outros programas, permite a disponibilização de conteúdos em diversas linguagens de programação tais como *PHP* [31], *ASP* [32] ou *Java* [33]. Estas aplicações podem ser executadas em qualquer computador, permitindo que através da introdução de um endereço num *browser* se aceda aos conteúdos quer localmente (numa rede interna) quer através da Internet (sistema global de redes de computadores interligadas) [34].

Existem diversos programas para criar um Servidor *Web* e um dos que possui maior adesão é o *Apache HTTP Server* [35] que, para além de gratuito, é compatível com diversos sistemas operativos, dos quais se destacam Microsoft Windows [36], Linux [37] e Mac OS [38]. O *Apache* permite disponibilizar conteúdos em *HTML* [39] / *Javascript* e com a instalação de módulos adicionais, *PHP* e muitos outros. O *HTML* é uma linguagem de marcação utilizada para produzir páginas web. Esta

linguagem faz uso de *tags* que definem elementos *HTML*. Esta pode conter *scripts*, tal como o *Javascript* e estilos como o *CSS* [41]. Enquanto os primeiros permitem controlar o comportamento dos diversos elementos, os estilos definem a aparência dos conteúdos da página.

O *PHP* é uma linguagem de programação orientada para a Internet e tal como o *Apache* é um software gratuito e compatível com diversos sistemas operativos. Enquanto no *HTML* as páginas são estáticas, isto é uma página obtida por diferentes utilizadores em diferentes alturas é sempre a mesma, o *PHP* permite páginas dinâmicas que podem ser geradas de forma diferente consoante o utilizador e o contexto, entre outros. Esta linguagem pode ser integrada com diversas bases de dados, tais como o *MySQL* [42], que é um sistema de gestão de base de dados relacional.

Na maioria dos casos, uma empresa ou um particular opta por alugar um servidor a uma empresa especializada na área. Deste modo pode alojar as suas “*web pages*” num local seguro, disponível 24 horas por dia, com tempos de acesso reduzidos e sem ter que se preocupar com a instalação e configuração da máquina. Como as aplicações aqui referidas são todas gratuitas a maioria das opções de alojamento disponíveis, incluem-nas. Quem opta por instalar numa máquina privada pode escolher instalar um programa que já inclui todas estas aplicações e que as configura automaticamente de forma a ficar tudo pronto a usar. Existem diversas versões deste programa para cada sistema operativo, no caso do Microsoft Windows a versão denomina-se *WAMP* (*Windows, Apache, Mysql e PHP*) [43].

Neste projecto pretende-se uma interface web que, após o carregamento inicial, utilize uma base de dados com todos os utilizadores e as propriedades da casa e de cada um dos seus dispositivos. Uma vez que necessitamos de páginas dinâmicas e se pretende uma solução fácil de encontrar no mercado e de baixo custo, optou-se por criar um servidor *web* com as tecnologias explicadas, nomeadamente *Apache, HTML, Javascript, CSS, PHP e MySQL*.

Capítulo 3

Arquitectura

O objectivo do projecto é o desenvolvimento de uma interface Web para o sistema o DomoBus. A aplicação desenvolvida para atingir este objectivo divide-se em três módulos: a base dados, a interface e o simulador DomoBus. A base de dados armazena todas as informações referentes à habitação e configurações da interface. A interface é responsável por gerar todos os ecrãs visíveis ao utilizador e pela monitorização e controlo de todos os dispositivos apresentados. Por fim, desenvolveu-se um simulador para facilitar os testes à aplicação e permitir que estes sejam o mais exaustivos possível sem se ficar dependente do uso de um sistema real. Deve notar-se que o uso do simulador não limita de forma alguma a aplicação ou o seu funcionamento, sendo possível realizar a mudança para o sistema real com um mínimo de esforço.

Na figura 4 é possível visualizar-se a decomposição do projecto nos seus diversos componentes e uma possível localização física dos mesmos. O *web browser* encontra-se sempre em qualquer dispositivo, móvel ou fixo, que o utilizador utiliza para aceder às páginas da interface. A aplicação que produz a interface e a base de dados encontram-se na mesma máquina, no entanto poderiam ser separadas alterando apenas a indicação do *host* da base de dados na aplicação. O simulador é representado numa máquina distinta da anterior e uma vez que simula um sistema domótico encontra-se na habitação. A distribuição da interface, base de dados e simulador não é fixa, podendo estes encontrar-se todos na mesma máquina ou em máquinas distintas.

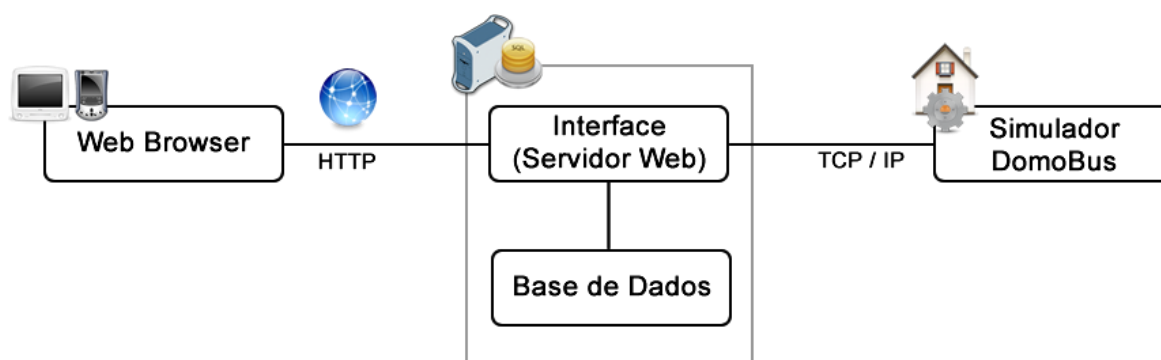


Figura 4 – Modelo do funcionamento da aplicação

3.1 Base de Dados

O propósito da base de dados é o armazenamento da estrutura da casa e composição do sistema domótico, informação sobre os diversos utilizadores e suas preferências e configurações. Estes dados provêm de um ficheiro *XML* gerado pelo Sistema DomoBus que contém as diversas divisões da casa, os seus dispositivos, propriedades e utilizadores. O modelo de dados construído baseia-se nestes quatro componentes e pode ser representado pelo seguinte modelo em UML [44]:

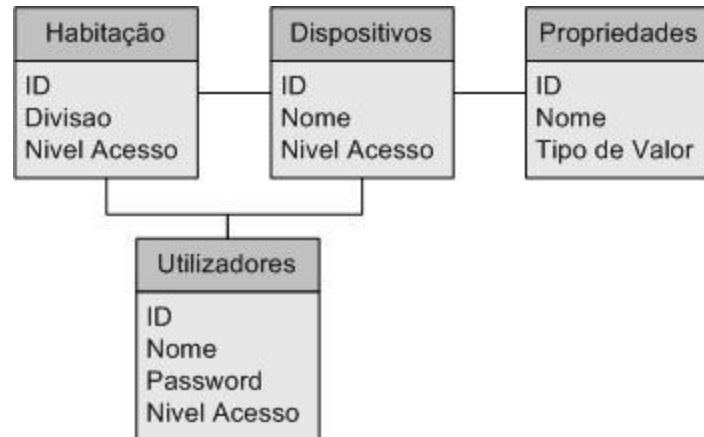


Figura 5 – Modelo simplificado da arquitectura da base de dados em UML

Neste modelo simplificado é possível visualizar que uma habitação possui várias divisões, cada uma com um nível de acesso que define se um determinado utilizador lhe pode aceder. Cada divisão pode possuir vários dispositivos e cada um destes pode apresentar várias propriedades configuráveis.

A estrutura de dados da aplicação teve em conta as diversas informações provenientes do *XML*. Alguns dos campos, embora de momento não sejam utilizados ou tragam informações úteis à interface, foram criados para manter um registo das informações iniciais importadas. Assim, torna-se possível exportar as informações num *XML* contendo o estado actual da aplicação em conjunto com toda a informação adicional existente num ficheiro gerado pelo DomoBus. De seguida, explica-se mais detalhadamente cada um dos quatro componentes identificados acima.

3.1.1 Habitação

Uma habitação (*House*) possui diversas divisões (*Division*) e é caracterizada pelo nome, morada e telefone. O campo *XML* serve para guardar uma cópia do último ficheiro importado, sendo possível recuperar o estado inicial. Uma divisão possui um nome, um andar, uma imagem representativa e um

nível de acesso que restringe a visualização por parte de utilizadores com um nível inferior. O único campo obrigatório é o nome da divisão uma vez que é utilizado para a representar.

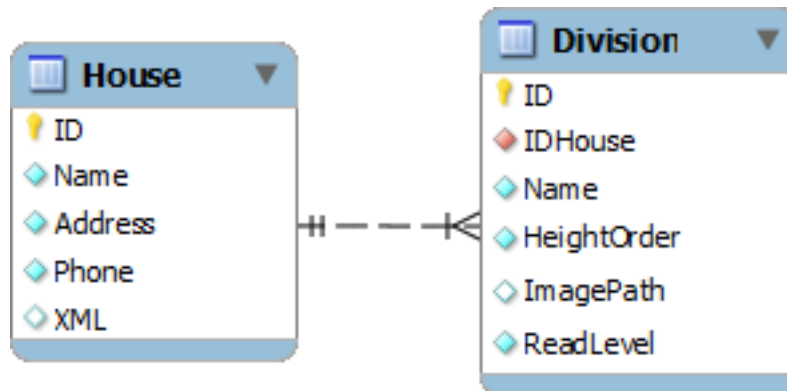


Figura 6 – Modelo detalhado dos dados da habitação

3.1.2 Dispositivos

Um dispositivo (*Device*) é caracterizado por um nome, níveis de acesso de leitura e escrita, identificação da divisão onde se encontra e identificação do tipo de dispositivo a que pertence. O nome e divisão do dispositivo são utilizados para identificação e localização deste na interface. Por sua vez, os níveis de acesso indicam o nível mínimo que um utilizador necessita ter para que possa visualizar ou editar o dispositivo em questão. O tipo de dispositivo (*Device Type*) indica a categoria a que o dispositivo pertence, como por exemplo “Lâmpada Regulada”, sendo que cada categoria existente pode abranger diversos dispositivos. O tipo exemplificado pode estar associado ao “Candeeiro de Mesa” e ao “Candeeiro de Tecto” da sala.

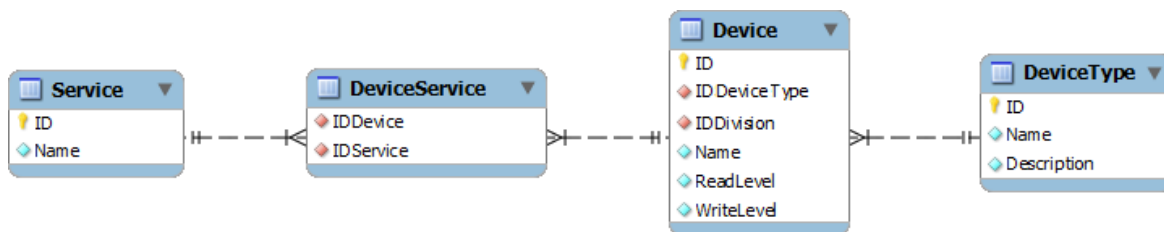


Figura 7 – Modelo detalhado dos dispositivos

No modelo de dados dos dispositivos encontramos ainda o conceito de serviço (*Service*). Este é utilizado para agrupar os diversos dispositivos associados a uma funcionalidade. Como é natural, um

serviço pode ter diversos dispositivos associados e um dispositivo pode pertencer a diversas funcionalidades, por exemplo um detector de presença relaciona-se tanto com segurança como com iluminação. Na primeira funcionalidade serve para detecção de intrusos na habitação, na segunda serve para controlo da iluminação poupando trabalho e energia aos habitantes desta.

3.1.3 Propriedades dos Dispositivos

As propriedades dos dispositivos (*DeviceProperty*) são os atributos que caracterizam um dispositivo. Cada propriedade possui um nome, o modo de acesso, o tipo de valor e um identificador para o conjunto de valores admissíveis. O modo de acesso especifica se a propriedade é apenas de consulta ou se também pode ser modificada; este indicador é independente do utilizador. O tipo de valor identifica a forma como o estado da propriedade é representado, quer por um valor escalar ou por um tipo de enumerado.

Os dois tipos de representação do estado de uma propriedade têm por sua vez um conjunto de valores possíveis. Uma grandeza escalar (*ScalarValue*) é caracterizada por um valor numérico inteiro que pode encontrar-se associado a um mínimo e máximo e a um valor mínimo de incremento (*Step*). Enquanto o mínimo e máximo limitam o intervalo de valores admissíveis para a propriedade, o valor mínimo de incremento representa a unidade de alteração que uma propriedade pode sofrer, por exemplo no caso de uma lâmpada reguladora o utilizador apenas conseguir alterar a intensidade de dez em dez por cento.

Os tipos de enumerado (*EnumType*) são categorias que agrupam um conjunto de pares nome/valor e designados por enumerados (*Enumerated*). Cada tipo é representado por um nome e pode possuir um ou mais enumerados associados. Considere-se por exemplo uma televisão, esta pode ter como propriedades o estado ligado / desligado (enumerado), o volume e o canal (escalares).

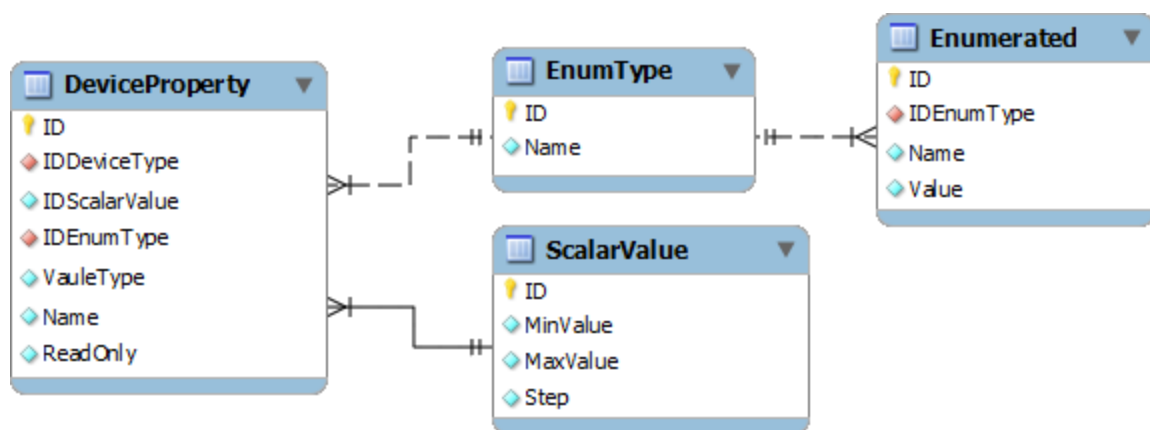


Figura 8 – Modelo detalhado das propriedades dos dispositivos

3.1.4 Utilizadores

Os utilizadores (*User*) são constituídos por um nome, uma palavra-chave, um nível de acesso e um conjunto de opções relativas à interface. Associados a estes existem ainda favoritos (*Bookmark*), macros (*Macro*) e propriedades em destaque (*ImportantProperty*). Enquanto as primeiras três propriedades do utilizador – nome, palavra-chave e nível de acesso – são provenientes do ficheiro XML gerado pelo DomoBus e importado na aplicação aqui descrita, as propriedades restantes provêm da interacção do utilizador com a aplicação e podem constar numa versão mais actualizada do XML exportada pela Interface (estrutura do XML explicada no capítulo 5 – Implementação).

O nome e palavra-chave identificam univocamente um utilizador perante a aplicação. Após a introdução destes dados, é através do nível de acesso que a aplicação determina se o utilizador pode visualizar cada divisão e dispositivo, por comparação com o nível mínimo de acesso a cada um dos componentes da habitação.

O utilizador pode costumizar a aplicação conforme as suas preferências, por esta razão existem diversas propriedades associadas ao mesmo, como é visível na figura 9 em *User*. O tipo de navegação (*NavigationType*) determina o modo como os dispositivos estão organizados na interface por omissão, se por divisão ou por serviço. O aspecto (*Skin*) e a linguagem (*Language*) determinam o esquema de cores e a linguagem em que a interface é apresentada.

A aplicação, além das preferências, permite que o utilizador especifique favoritos, acções compostas e propriedades em destaque, como é possível visualizar no modelo representado na figura 9. Os favoritos são uma lista de dispositivos, divisões e serviços a partir da qual o utilizador pode aceder directamente ao componente seleccionado sem necessitar de navegar pelas divisões ou serviços. Um favorito é constituído por três identificadores que correspondem a um dispositivo, divisão ou serviço. No entanto para cada favorito apenas um dos identificadores se encontra em utilização, especificando o tipo escolhido.

As acções compostas, designadas por macros, são um conjunto de acções pré-programadas pelo utilizador e onde cada uma tem associada uma ou mais acções (*MacroAction*). Cada uma destas acções é caracterizada pelo dispositivo a que se refere, pela propriedade deste que se pretende alterar e pelo valor a atribuir à propriedade.

As propriedades em destaque são uma qualquer propriedade que o utilizador escolhe destacar na interface e ter sempre visível o seu estado em qualquer ecrã. No modelo de dados, estas são constituídas pelos identificadores do dispositivo e da propriedade que o utilizador deseja destacar. Cada uma destas propriedades pode ainda possuir uma imagem associada para facilitar a sua identificação.

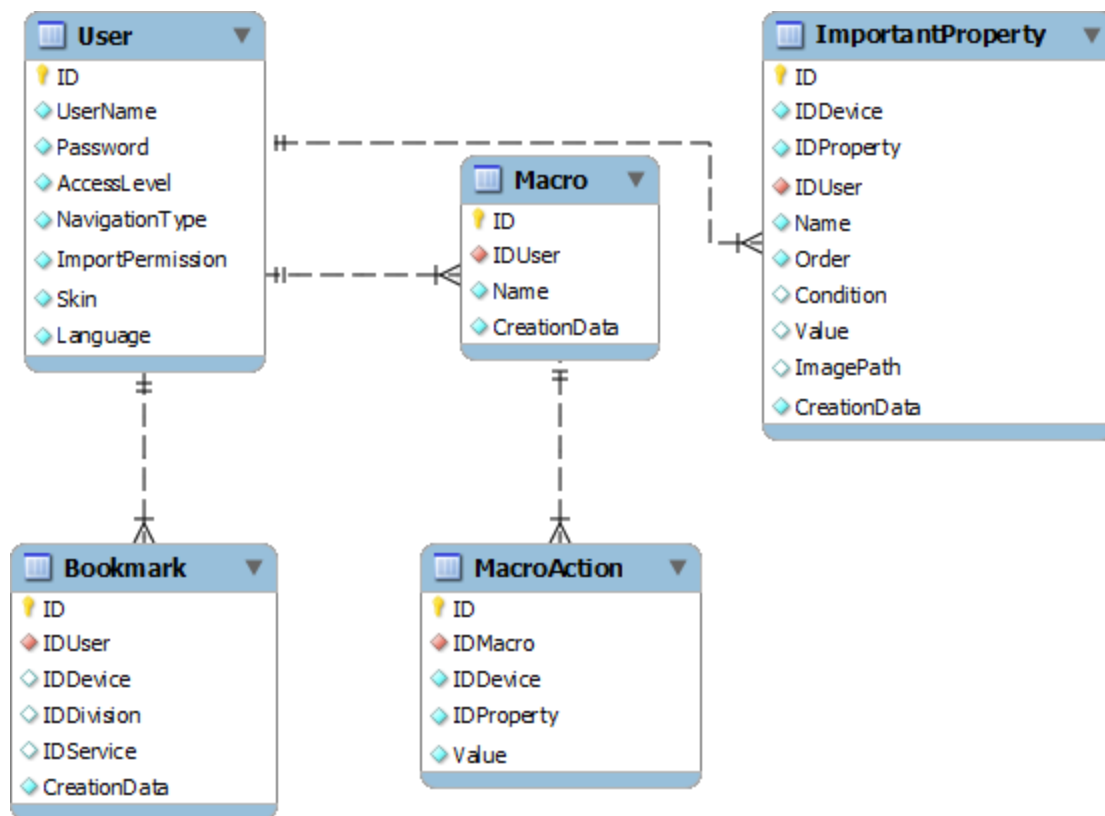


Figura 9 – Modelo detalhado dos dados associados aos utilizadores

3.2 Interface

A interface é responsável por apresentar um conjunto de páginas web que permitam ao utilizador visualizar e controlar a sua habitação. Uma página *web*, ao contrário de uma aplicação de *desktop*, não tem um fluxo que esteja sempre em execução. Quando um pedido é efectuado ao servidor, o mesmo cria uma tarefa com o intuito de processar o pedido e lhe responder. Uma vez enviada a resposta, a tarefa é concluída e termina. Deste modo, existem certas inicializações e validações que têm de ser realizadas em cada pedido.

A arquitectura desta interface *web* divide-se em três grandes componentes: o módulo global, responsável pelas inicializações e validações referidas acima; o componente principal, responsável pelo acesso e construção da estrutura base da aplicação e o componente *Ajax*, responsável por responder aos pedidos *Ajax*.

3.2.1 Componente Principal

O componente principal é responsável pela primeira página da aplicação apresentada ao utilizador. Esta página consiste sempre num de três conteúdos: o carregamento de XML, o início de sessão ou a moldura envolvente das restantes páginas da aplicação. Em qualquer um destes casos, o componente principal faz uso de dois módulos, o “Cabeçalho” e o módulo correspondente ao conteúdo a ser apresentado na primeira página. A arquitectura deste componente apresenta-se na figura seguinte.

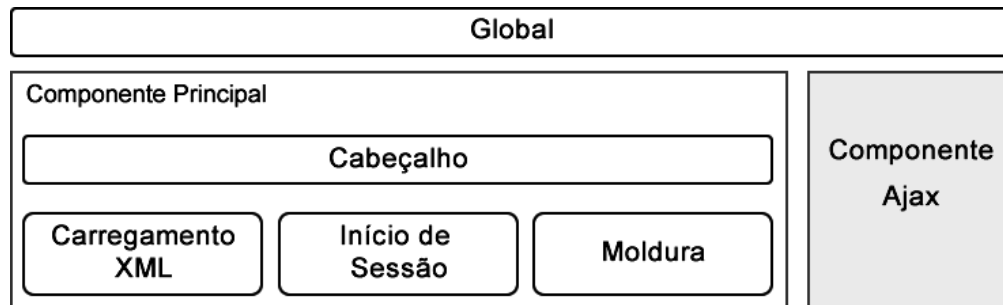


Figura 10 – Representação dos diversos módulos do componente principal.

O “Cabeçalho” é o módulo responsável por carregar os *scripts client-side (Javascript)*, os estilos (CSS) e as imagens. Neste último caso, no primeiro acesso à aplicação, este módulo carrega o conjunto das imagens mais visualizadas de forma a evitar constantes carregamentos do mesmo conteúdo. Após os carregamentos do “Cabeçalho”, o módulo seguinte depende do estado da aplicação.

A primeira vez que um qualquer utilizador acede à aplicação, esta ainda não contém informações sobre a habitação ou os seus habitantes; desta forma neste primeiro acesso o componente principal disponibiliza uma página com um formulário onde é possível importar um ficheiro XML com os dados da habitação. O conteúdo deste ficheiro é interpretado pelo módulo denominado “Carregamento XML” e mapeado na base de dados; a partir desse momento a aplicação passa a ter conhecimento de todos os dados necessários ao seu funcionamento, nomeadamente divisões, dispositivos e utilizadores.

Em futuros acessos após esta inicialização, e uma vez que a aplicação já reconhece os seus utilizadores, este componente disponibiliza a página de início de sessão. Esta página é apresentada sempre que a aplicação detecta um acesso não autenticado e permite que o utilizador, através da introdução do seu identificador e da palavra-chave, inicie a sua sessão e aceda à aplicação. Após a autenticação do utilizador, este componente fica responsável por criar uma moldura envolvente usada em todas as restantes páginas da aplicação.

Em adição aos módulos descritos existe ainda o módulo “Global” e o “Componente Ajax”. Ambos estão representados na figura 10, para uma melhor percepção da arquitectura da interface desenvolvida e são explicados nas secções seguintes.

3.2.2 Componente Ajax

O utilizador ao interagir com a aplicação e ao aceder às divisões e aos dispositivos, faz com que esta necessite de carregar os novos dados a apresentar. Numa página normal este tipo de interacção requeria que a página web fosse inteiramente recarregada, tanto os conteúdos novos como aqueles que já era possível visualizar na página anterior. Com o intuito de diminuir o tráfego e actualizar apenas as informações necessárias à interface, após o carregamento inicial da aplicação esta passa a funcionar com base em pedidos *Ajax*. Em termos práticos, com o uso deste componente, evita-se o carregamento dos módulos “Cabeçalho” e “Moldura” do componente principal a cada pedido e carrega-se apenas o conteúdo novo produzido pelos módulos deste componente. Estes módulos podem ser identificados na figura seguinte.

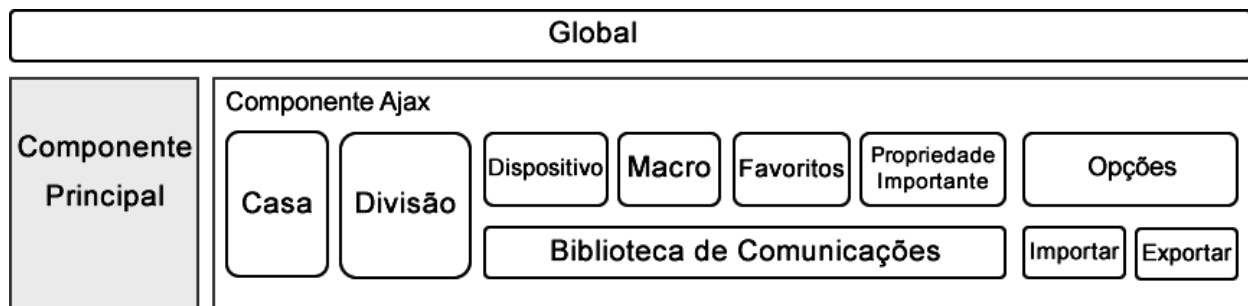


Figura 11 – Representação dos diversos módulos das páginas acedidas por *Ajax*.

O componente Ajax é assim responsável, tal como identificado acima, pelo carregamento de novos conteúdos para uma página que já está a ser visualizada. Sempre que o utilizador navega entre ecrãs da aplicação, o pedido de novos dados é enviado do *browser* para este componente que, consoante os argumentos recebidos, determina qual o conteúdo a devolver para ser apresentado no ecrã. Este conteúdo depende do tipo de informação a ser apresentada e é produzido pelo módulo responsável por esse conteúdo. Por exemplo, a página que apresenta os dispositivos de uma divisão é produzida pelo módulo “Divisão”.

Na figura 11 é ainda possível verificar a existência de alguns módulos que são auxiliares àqueles que produzem o conteúdo das páginas. Estes módulos são a “Biblioteca de Comunicações”, o “Exportar” e o “Importar”. Enquanto o primeiro contém todos os métodos que permitem a comunicação TCP/IP da

Interface com o DomoBus, o módulo “Exportar” permite guardar todos os dados e configurações da interface num ficheiro XML. O último módulo permite carregar um ficheiro XML com dados e configurações para a aplicação. Estes dados apenas actualizam os já existentes não modificando as configurações próprias que a interface já possua. O ficheiro pode ser gerado pelo DomoBus ou pela própria aplicação. O módulo “Global” é descrito na secção seguinte.

3.2.3 Global

Os componentes descritos anteriormente fazem uso do módulo “Global” que é responsável pelas validações e carregamentos necessários em todos os cliques, como por exemplo validar que o acesso está a ser realizado por um utilizador autenticado. Este módulo é, por sua vez, constituído por cinco sub-módulos apresentados na figura 12.

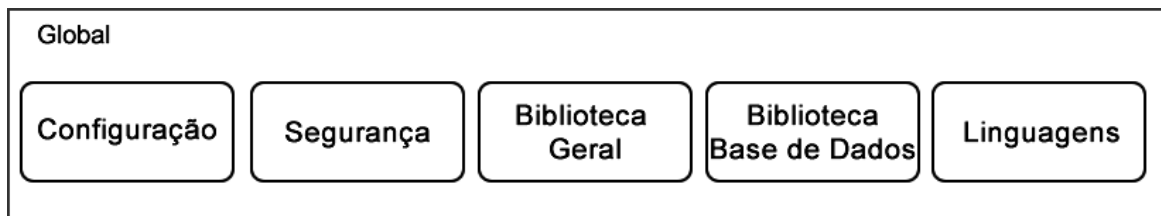


Figura 12 – Representação dos diversos sub-módulos do módulo Global

O sub-módulo “Configuração”, tal como o nome indica, é constituído por configurações aplicacionais, como a autenticação para acesso à Base de Dados, o domínio e o porto para a ligação TCP/IP, entre outros. A “Biblioteca Base de Dados” contém todos os métodos necessários para consultar e modificar os dados que constam na base de dados. Uma vez carregado este sub-módulo qualquer módulo pode aceder facilmente à base de dados através da invocação de métodos simples e intuitivos. Uma grande vantagem de utilizar este sub-módulo é permitir a independência entre a aplicação e a tecnologia de Base de Dados utilizada, pois desde que se mantenha a interface de métodos, a implementação deste sub-módulo pode ser a que o programador desejar.

A “Biblioteca Geral” contém uma lista de funções globais à aplicação. Estas funções encontram-se aqui pois são comuns à maioria dos ecrãs apresentados na aplicação; desta forma, independentemente da navegação efectuada pelo utilizador, estas funções estão sempre disponíveis para execução. O sub-módulo “Segurança” é responsável por validar os dados dos utilizadores e respectivas permissões de acesso, permitindo identificar se o utilizador é conhecido e os seus dados devem ser carregados ou se, pelo contrário, a aplicação deve ser redireccionada para a página de início de sessão.

Por fim, é carregado o sub-módulo responsável pelas linguagens que contém um ficheiro para cada linguagem suportada pela aplicação. Este sub-módulo identifica qual a linguagem especificada pelo utilizador e carrega o ficheiro respectivo, que contém todas as mensagens apresentadas na Interface.

3.3 Simulador DomoBus

O projecto desenvolvido no âmbito desta dissertação, tem por objectivo fornecer uma interface remota ao sistema domótico DomoBus. No entanto, este sistema é ele próprio um projecto académico em desenvolvimento e para efeitos de implementação desta aplicação eram necessárias duas condições. Por um lado, um ambiente estável onde fosse possível testar os diversos comandos e interacções do utilizador com a interface e, por outro lado, um ambiente simulado onde os comandos enviados pela interface não fossem directamente enviados a dispositivos físicos. O simulador não pertence à aplicação final e uma vez que esta foi construída de forma modular, o simulador pode ser substituído por uma ligação real ao DomoBus sem alterar o comportamento da interface.

O simulador DomoBus tem assim o objectivo de permitir testar e validar, de uma forma rápida, simples e realista, a aplicação desenvolvida. Para proporcionar esta funcionalidade o simulador é constituído por três módulos organizados tal como se apresenta na figura 13.

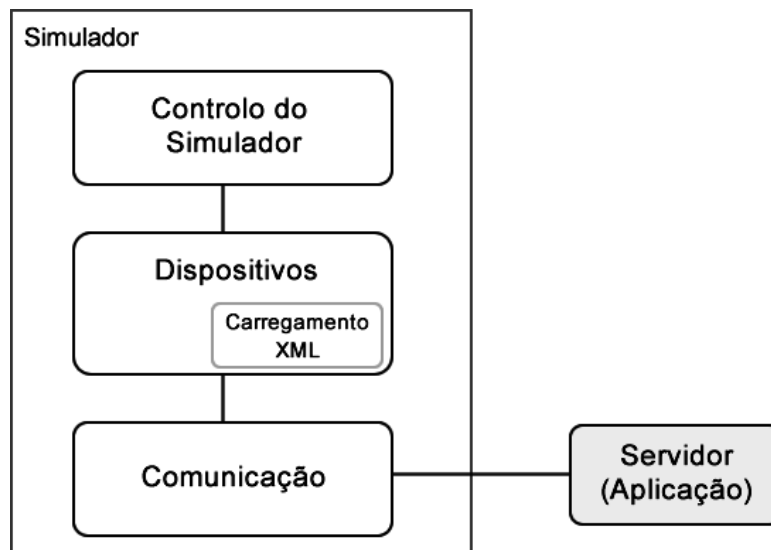


Figura 13 – Representação dos módulos do Simulador do DomoBus

Toda a informação do simulador é estruturada no módulo "Dispositivos". Contudo, tal como na Interface, é necessário interpretar o ficheiro XML gerado pelo DomoBus por forma a conhecer a estrutura

da habitação, os seus dispositivos e a sua localização. No entanto, ao contrário da Interface, o simulador não contém uma base de dados pelo que necessita de realizar esta interpretação no arranque sempre que é executado. O sub-módulo responsável por esta interpretação é o “Carregamento XML”. Uma vez que a informação gerada pelo DomoBus não apresenta o estado dos dispositivos, este sub-módulo também interpreta um outro ficheiro XML com esta informação. Este segundo processo de interpretação traz a vantagem de ser possível inicializar o simulador com qualquer estado de dispositivos sem ser necessário alterar a implementação do mesmo.

Uma vez terminado o carregamento e a interpretação de dados, o módulo “Dispositivos” mantém a informação necessária para simular todos os dispositivos da casa e disponibiliza os métodos que permitem a consulta e alteração desses dados. Estes métodos correspondem aos pedidos que os módulos “Comunicação” e “Controlo do Simulador” podem efectuar.

O módulo “Comunicação” é responsável por receber as mensagens provenientes da aplicação e traduzi-las em pedidos para o módulo “Dispositivos”, por forma a obter ou modificar o estado dos dispositivos no simulador. Para conseguir receber as mensagens da Interface, o módulo “Comunicação” cria um socket [45] e aguarda novas mensagens desta. Sempre que uma mensagem é recebida este módulo valida os seus argumentos e invoca as acções desejadas.

As mensagens TCP/IP enviadas pela Interface correspondem sempre a um de dois tipos, um pedido de estado de uma propriedade de um dispositivo ou uma alteração do respectivo estado. No sentido de diminuir o tráfego de informação entre as aplicações e, uma vez que a maioria das vezes se pretende obter ou alterar mais que uma propriedade no mesmo instante, as mensagens foram estruturadas para possuir múltiplos argumentos identificando os diferentes dispositivos e propriedades que se pretende obter ou modificar. Na figura seguinte pode verificar-se a estrutura de ambos os tipos de mensagens.

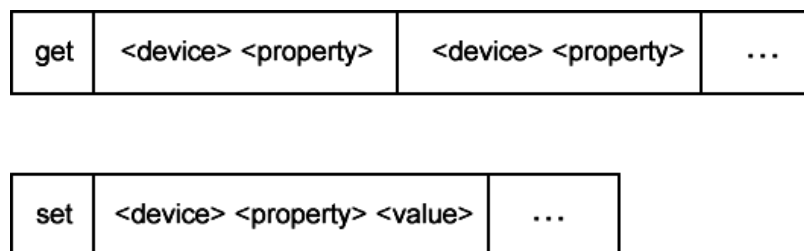


Figura 14 – Formato das mensagens recebidas pelo Simulador DomoBus

A resposta do simulador aos pedidos recebidos é efectuada quando o processamento destes termina. No caso de um pedido para obtenção de estado, o simulador responde com o estado actual do dispositivo. Se o pedido é referente a uma modificação o simulador responde apenas com zero ou um

indicando o respectivo sucesso ou erro na realização da acção. Em caso de ocorrência de um erro de origem desconhecida optou-se por enviar um carácter especial, neste caso a arroba, como resposta. Todas as mensagens trocadas via TCP/IP terminam com o carácter “\n”.

Um dos objectivos deste simulador é validar que a Interface está a apresentar o estado correcto dos dispositivos e que quando é pedida uma alteração de estado pelo utilizador, existe realmente um estado alterado numa aplicação exterior à Interface. O “Controlo do Simulador” é o módulo através do qual se pode efectuar esta validação, pois permite consultar os diversos dispositivos existentes no simulador e os seus estados. Desta forma, e em paralelo com a interacção que a Interface faz com o simulador, é possível monitorizar os estados dos dispositivos e verificar que estão realmente a ser modificados conforme as acções do utilizador sobre a interface.

Além da monitorização, o módulo “Controlo do Simulador” também permite manipular o estado dos dispositivos disponíveis no simulador. Assim, é possível validar que um utilizador que está a aceder através da Interface tem acesso ao estado real da habitação, mesmo que alguém na habitação esteja a alterar directamente o estado dos dispositivos. Numa outra óptica, o simulador mantém uma estatística de quantas mensagens da Interface já foram processadas e pode também ser usado para validar a robustez e tolerância a falhas da Interface, pois permite configurar períodos de lentidão na resposta e mesmo de erro. A configuração de lentidão e erros de comunicação possível no “Controlo do Simulador”, permite simular momentos em que estes problemas ocorram por falha do DomoBus ou dos dispositivos na execução das acções.

O simulador DomoBus é uma aplicação Java que pode ser executada em qualquer computador, com qualquer sistema operativo e, uma vez que a sua comunicação é feita através de mensagens TCP/IP, esta aplicação pode ser executada na mesma máquina que a Interface ou numa diferente. Em adição, a Interface não tem noção se esta se encontra a comunicar com o simulador ou com o DomoBus, tornando assim os testes o mais realistas possível.

Capítulo 4

Implementação

Neste capítulo descreve-se em detalhe as diversas funcionalidades da aplicação implementada no âmbito desta dissertação. Começa-se por abordar as funcionalidades da Interface desenvolvida e os ecrãs que as disponibilizam. Segue-se a apresentação das bibliotecas auxiliares criadas para promover o funcionamento eficiente e modular da Interface. Posteriormente, são descritas as funcionalidades do simulador de DomoBus e a forma como este pode ser usado para testar a Interface. Concluindo-se com a apresentação de um diagrama de interacção entre todos estes elementos.

A Interface foi desenvolvida para ser acedida através de qualquer *browser* e ser executada num servidor *web* de baixo custo. Desta forma, as tecnologias escolhidas para a sua implementação são gratuitas, compatíveis com os sistemas operativos mais comuns e disponibilizadas pela maioria das empresas de alojamento. Assim, as principais tecnologias deste projecto são PHP [31], para desenvolver as diversas páginas da aplicação e MySQL [43], para gerir a base de dados relacional. Em complemento ao HTML [39] gerado pelo PHP, é também usado *Javascript* [40], *Ajax* [24] e CSS [41]. Deve ainda destacar-se que o simulador de DomoBus foi implementado em Java [33].

4.1 Funcionalidades da Interface

O objectivo principal da aplicação desenvolvida é a criação de uma interface de acesso remoto para monitorizar e controlar um sistema domótico. Desta forma, a interface pode ser acedida a qualquer momento a partir de um dispositivo móvel com acesso à Internet. Na demonstração das funcionalidades desta interface, serão apresentados os ecrãs onde as mesmas podem ser acedidas e com as quais o utilizador pode interagir.

Na sua maioria, todos os ecrãs que compõem interface obedecem a uma estrutura global. Assim, antes da apresentação de cada funcionalidade, começa-se por explicar a estrutura dos ecrãs. Na figura 15 cada número a vermelho corresponde a um componente ou zona do ecrã, que será explicado de seguida.

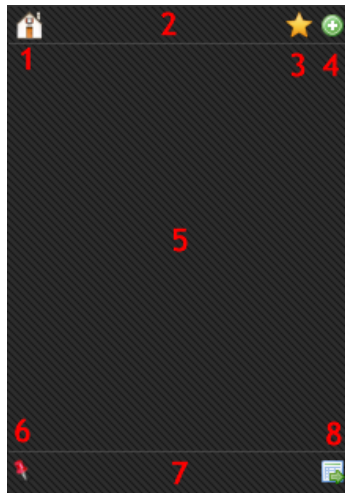


Figura 15 – Estrutura global das páginas da interface

Legenda:

1. Acesso ao ecrã principal da aplicação – permite voltar ao ecrã principal a partir de qualquer ponto da aplicação, normalmente designado por *Home*.
2. Barra de endereço – apresenta o caminho desde a página principal até ao local onde o utilizador se encontra. Todos os pontos no caminho dão acesso, por *link*, à página a que correspondem.
3. Acesso aos favoritos – listagem e adição de favoritos.
4. Abertura de menu – atalho que permite aceder ao menu apresentado na figura 16.
5. Área de conteúdo – zona onde se apresentam os diversos ecrãs da aplicação, tais como lista de divisões, dispositivos, serviços, opções, entre outros.
6. Acesso às propriedades em destaque – listagem e configuração destas propriedades.
7. Propriedades em destaque – zona onde se apresentam as propriedades destacadas pelo utilizador e os seus valores.
8. Acesso às macros – listagem, configuração e execução de macros.

De forma a otimizar o espaço disponível no ecrã foi criado um menu, acessível a partir do ponto quatro da figura 15, que disponibiliza um conjunto de funcionalidades usadas com menor frequência que as restantes. Quando o utilizador acede a este menu é apresentada uma barra azul, com a estrutura que se apresenta na figura seguinte.



Figura 16 – Menu da aplicação, acessível pelo ponto 4 da figura 15.

Neste menu é possível aceder às opções, lista de serviços ou sair da aplicação. Nas opções o utilizador acede a um ecrã onde é possível especificar o tipo de navegação, a linguagem e o tema da aplicação. É possível ainda exportar e importar dados em formato XML, para guardar ou actualizar o estado da aplicação. Todos estes itens são explicados em maior detalhe nas secções seguintes.

4.1.1 Sistema de Navegação

A interface apresenta ao utilizador a possibilidade de monitorizar e manipular as propriedades de todos os dispositivos domóticos instalados na sua habitação. Para aceder a cada uma destas propriedades a interface disponibiliza dois modos de navegação, por divisões e por serviços. Na figura seguinte podem visualizar-se ambos os modos, tal como o utilizador lhes acede.

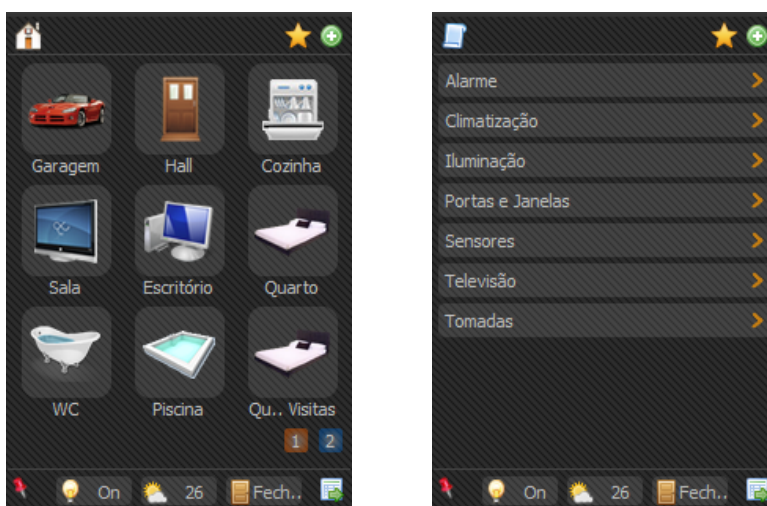


Figura 17 – Navegação por divisões (esquerda) e por serviços (direita).

Na imagem da esquerda apresenta-se a navegação por divisões. Este é o modo mais natural de visualizar a aplicação, pois representa a estrutura física da habitação, em que todos os dispositivos estão na divisão da casa onde realmente podem ser encontrados. Devido a ser o modo mais familiar, quer para um utilizador novo como um experiente, este foi definido como o modo de navegação por omissão. A página principal da interface, quando neste modo, apresenta as divisões organizadas em tabela onde cada divisão é representada por uma imagem e uma designação. O número de divisões apresentadas no ecrã depende das dimensões do mesmo e quando não existe espaço suficiente, a interface apresenta um

sistema de paginação. O acesso às diferentes páginas deste sistema é realizado através da numeração no canto inferior direito da área de conteúdo (imagem da esquerda).

A imagem da direita apresenta o modo de navegação por serviços. Este modo lista todos os serviços existentes no sistema, agrupando-os segundo as funcionalidades de cada um. Enquanto no modo anterior um dispositivo apenas se pode encontrar numa divisão, neste modo o mesmo dispositivo pode fazer parte de vários serviços. Considere-se o exemplo de um sensor de presença, este pode funcionar como parte do serviço de iluminação, controlando as luzes conforme a presença de habitantes e ao mesmo tempo fazer parte do serviço de alarme, na detecção de intrusos.

O modo de navegação escolhido por omissão determina o primeiro ecrã da aplicação apresentado ao utilizador após o início de sessão. Em qualquer outro ecrã, a única diferença entre estes dois modos é a imagem de acesso ao ecrã principal da aplicação, diferença apresentada nas imagens da figura 17 (canto superior esquerdo). Nas opções é possível alternar entre os dois modos de navegação, escolhendo o modo de omissão. No entanto, se o utilizador pretender aceder temporariamente ao modo que não escolheu, pode fazê-lo através da abertura de menu, atalho Serviços ou Divisões.

No caso da navegação por divisões, no ficheiro XML importado, pode dar-se o caso de não terem sido especificadas imagens para todas as divisões. Neste caso, estas divisões, são representadas por uma imagem genérica. Se eventualmente não foram especificadas imagens para nenhuma das divisões, então a interface apresenta-as em lista, tal como na navegação por serviços.

Em ambos os tipos de navegação, após a escolha de uma divisão ou de um serviço, a interface apresenta a lista de dispositivos que corresponde ao item seleccionado.

4.1.2 Propriedades de dispositivos

Um dispositivo é constituído por um conjunto de propriedades, onde cada uma tem um estado/valor associado. São estes valores que caracterizam o dispositivo e as suas funcionalidades e são também estes valores que o utilizador pode visualizar e manipular através da interface. Tal como indicado na secção acima, quando se selecciona uma divisão ou um serviço acede-se à lista de dispositivos da selecção; nesta lista o utilizador pode escolher um qualquer dispositivo e após a sua escolha é-lhe apresentada a lista de propriedades do dispositivo seleccionado.

Manipular um dispositivo corresponde a manipular as suas propriedades, no entanto algumas destas não podem ser alteradas pelo utilizador. A apresentação da lista de propriedades no ecrã distingue entre estes dois tipos, propriedades só de leitura ou de leitura e escrita, como se pode ver na figura seguinte (à direita) onde por exemplo a “Temperatura” só pode ser monitorizada e o “Canal” pode ser monitorizado e alterado.

Na figura 18, à esquerda, podem visualizar-se os dispositivos existentes na sala, onde a televisão se encontra em último. À sua direita encontram-se as propriedades da televisão, apresentadas quando esta é seleccionada. A lista quando acedida encontra-se fechada, isto é, podem visualizar-se as propriedades e os seus estados, mas não se apresentam comandos para efectuar alterações. Estes comandos são acessíveis por selecção da propriedade, que faz o ecrã “abrir” uma secção entre a propriedade seleccionada e a seguinte na lista, como se mostra nas figuras 19 e 20.



Figura 18 – Lista de dispositivos da sala (esquerda) e lista de propriedades da televisão (direita).

A forma de manipular as propriedades de um dispositivo difere em relação ao tipo de propriedade seleccionada e desta forma, também diferem os comandos para realizar as alterações. Existem dois tipos de propriedades, os enumerados e as grandezas escalares, e os dois métodos de configuração respectivos são descritos nas subsecções seguintes.

4.1.2.1 Enumerados

Uma propriedade do tipo enumerado tem um conjunto fixo de valores que podem representar o seu estado. O utilizador apenas pode escolher um destes valores, pelo que a interface os apresenta sobre a forma de botões. Cada botão tem a designação de um estado e o seu tamanho é influenciado pelo número de caracteres da designação. Quando o conjunto de botões não consegue ser apresentado lado a lado num linha, a interface estrutura-o para ser apresentado em coluna.

Na figura seguinte apresenta-se a lista de propriedades do ar condicionado da sala, com os comandos para efectuar alterações visíveis. Neste caso os comandos são apenas botões estruturados em linha (à esquerda) e em coluna (à direita). Em ambas as configurações o estado seleccionado é facilmente identificado, correspondendo ao botão preenchido.

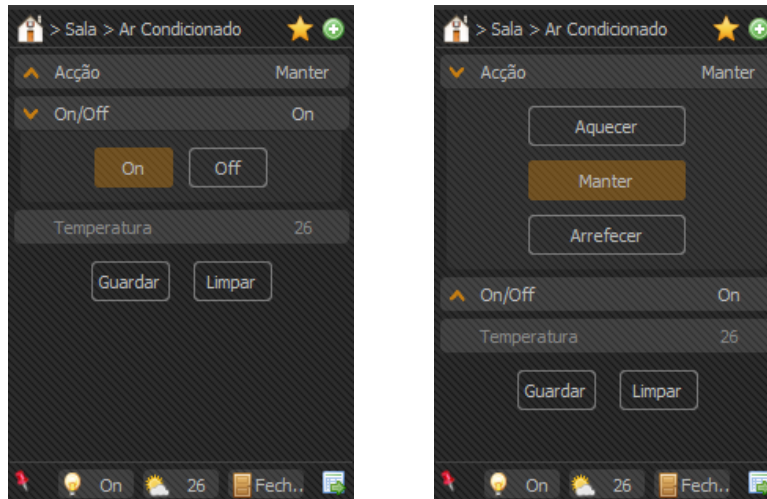


Figura 19 – Representação de propriedades do tipo enumerado.

4.1.2.2 Grandeza Escalar

Uma grandeza escalar é um valor numérico. Assim, as propriedades deste tipo são sempre associadas a estados representados por números, que podem ser limitados por um valor mínimo e/ou máximo. Estas propriedades podem estar associadas a conjuntos de valores de tamanho variável, por sua vez este tamanho determina a forma como a interface apresenta o modo de manipulação do estado. Quando o número de valores possíveis, de uma propriedade deste tipo, é inferior ou igual a cinco a interface apresenta-os numa linha como demonstrado na figura seguinte.

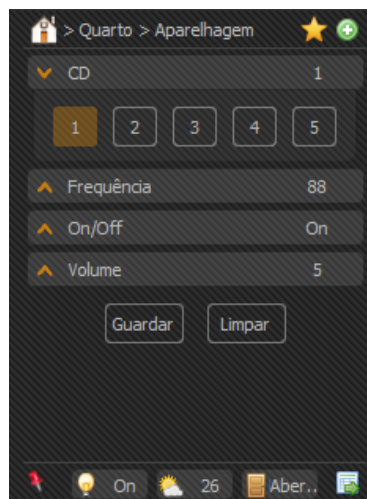


Figura 20 – Propriedade escalar com um máximo de cinco valores possíveis.

No caso do número de valores possíveis ser superior a cinco, a interface apresenta um controlo mais elaborado para a definição do valor pretendido. Na figura 21, em todas as imagens, pode visualizar-se este controlo, constituído por duas colunas. Na coluna da esquerda encontra-se o mostrador do novo valor, o incremento e decremento do valor, a identificação dos valores máximo e mínimo possíveis. Esta identificação também permite a selecção num único clique dos valores correspondentes. Na coluna da direita encontra-se o teclado numérico para a definição de valores, o remover e a alteração de sinal. A alteração de sinal só se encontra disponível se a propriedade suportar valores tanto positivos como negativos.



Figura 21 – Controlo de propriedades escalares.

Na figura anterior pode ainda visualizar-se três aspectos de controlo de propriedades escalares: a definição de valores restrita a intervalos, a identificação de propriedades modificadas de forma apropriada e a identificação de erros. O primeiro caso são propriedades cujos valores não são especificados de um em um mas sim num outro intervalo como por exemplo de dez em dez. O controlo de propriedades com estes intervalos é demonstrado na imagem do meio, onde é apresentada uma mensagem identificativa do intervalo a respeitar e onde os botões do mostrador especificam o incremento e decremento mínimo que pode ser realizado.

A interface tenta prevenir que o utilizador incorra em erros ao definir os valores das propriedades, no entanto nem sempre é possível fazê-lo. A imagem da direita apresenta tanto o caso em que propriedades existem bem definidas como possíveis situações de erro. Sempre que uma propriedade é bem definida, a designação da mesma passa a apresentar a cor verde. Caso contrário, o utilizador é avisado do erro através da cor vermelha da propriedade, do novo valor e da infracção existente. É ainda

apresentada uma mensagem de aviso junto ao botão de gravação que identifica ao utilizador a existência de um ou mais erros nas propriedades modificadas que serão ignoradas em caso de gravação.

4.1.3 Permissões

No sistema domótico DomoBus, cada utilizador, divisão e dispositivo tem associado a si um nível de acesso entre zero e dez. Estes níveis são utilizados para determinar as permissões de leitura e escrita do utilizador em relação a cada divisão e dispositivo. No caso das divisões, um utilizador só visualiza aquelas cujo nível de acesso é inferior ou igual ao nível detido pelo próprio. No caso dos dispositivos existem dois níveis, um de acesso e outro de edição. Enquanto no nível de acesso é aplicada a mesma regra que no caso das divisões; no nível de edição o utilizador consegue visualizar as propriedades dos dispositivos e os seus valores não conseguindo, no entanto, alterá-las. As propriedades nestas condições são identificadas através de um cadeado como se pode visualizar na figura 22.

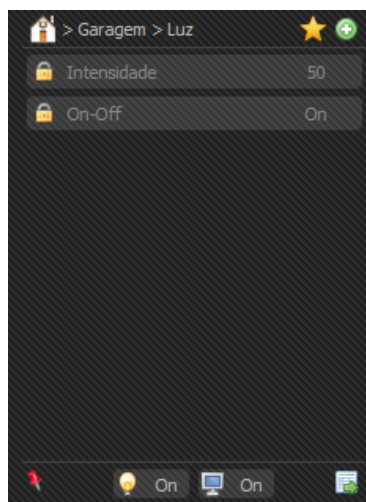


Figura 22 – Representação do bloqueio de edição pelo sistema de permissões.

4.1.4 Favoritos

A interface desenvolvida no âmbito desta dissertação, permite manipular as propriedades de todos os dispositivos domóticos existentes numa habitação. No entanto, o utilizador tem tendência a aceder a determinadas divisões, serviços ou dispositivos com maior regularidade que outros. Assim sendo, a interface fornece ao utilizador a possibilidade de configurar uma lista de favoritos, permitindo um acesso mais rápido e orientado a este conjunto de itens.

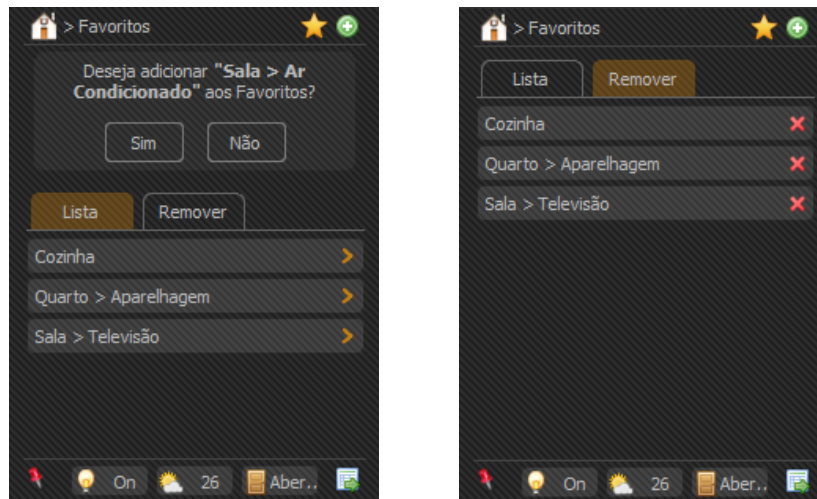


Figura 23 – Ecrãs dos favoritos do utilizador.

Os favoritos são constituídos por duas secções: a lista e o remover. Na primeira são listados todos os favoritos permitindo ao utilizador aceder a estes através de um só clique. No remover apresenta-se a mesma lista, no entanto serve para o utilizador indicar o item a remover sendo redireccionado para um ecrã de confirmação desta operação. Quando um utilizador se encontra num item que pretende adicionar como favorito necessita apenas de aceder aos mesmos, nesse acesso é-lhe apresentada a mensagem visível na figura 23 à esquerda a confirmar a adição do favorito. Esta mensagem só é apresentada se o item em questão ainda não fizer parte da lista de favoritos.

4.1.5 Macros

Os utilizadores de sistemas domóticos, por norma, efectuem determinadas tarefas em conjunto. Por exemplo, quando vão dormir ou se ausentam de casa podem trancar todas as portas, janelas e portões da habitação, desligar todas as luzes e activar o alarme. Este tipo de tarefa, mesmo com um sistema domótico, requer o acesso a diversas divisões e dispositivos obrigando o utilizador a efectuar diversas acções que despendem o seu tempo. As macros ajudam o utilizador a realizar este tipo de tarefas compostas, agrupando num só local um conjunto de acções pré-programadas que este pode mandar executar através de um só clique. Cada acção deste conjunto define o valor de uma propriedade de um dispositivo domótico.

Os ecrãs das macros estão organizados em três secções “Executar”, “Detalhes” e “Remover”. Em todos os casos é sempre apresentada uma lista das macros existentes no sistema, mas a função de cada uma varia entre cada secção. Na primeira, o clique ou toque na macro promove a execução da mesma, fazendo com que todas as acções associadas à macro sejam executadas nos dispositivos a que se referem. Nos “Detalhes” é possível aceder a um ecrã para editar a macro escolhida, onde se encontra

uma lista das diversas acções que compõem a macro, entre outras configurações. Por fim, na secção “Remover” o utilizador pode eliminar qualquer macro da sua aplicação, esta remoção é confirmada num ecrã adicional após o clique. Em qualquer uma das secções, após a realização de uma tarefa na mesma, é sempre apresentado no topo uma mensagem de sucesso ou erro de acordo com o resultado da tarefa efectuada.

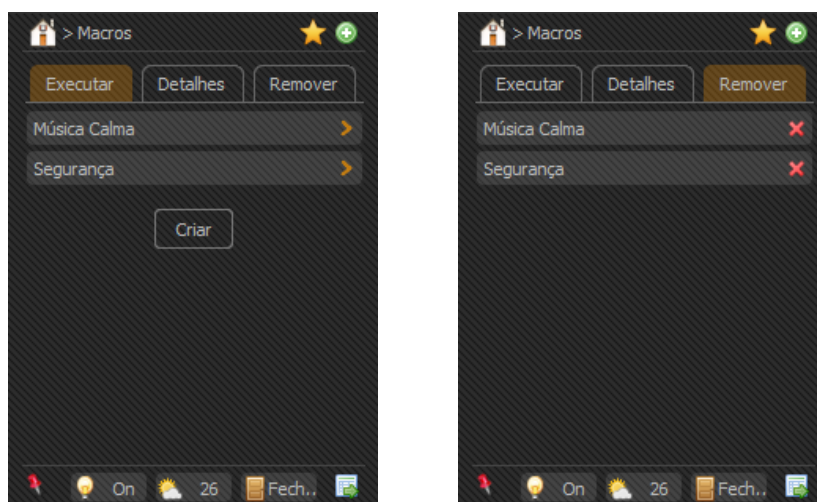


Figura 24 – Ecrãs de algumas secções das macros.

Na secção “Executar” existe ainda uma função adicional, a criação de novas macros, através do botão “Criar” tal como se apresenta na imagem à esquerda da figura 24. O processo de criação de uma macro consiste na gravação de todas as acções que o utilizador realiza durante um determinado período de tempo. Este período inicia-se quando o utilizador clica no botão “Criar” e é redireccionado para o ecrã principal da aplicação e termina quando o mesmo acede novamente às macros ou clica no link que é apresentado na área das propriedades em destaque, visível na figura 25 imagem à esquerda. Durante a gravação o utilizador navega normalmente pela aplicação e efectua as alterações que deseja gravar na macro, sendo-lhe apresentado uma mensagem de sucesso de registo dos dados em memória por cada alteração efectuada. Quando o utilizador termina a gravação é-lhe pedido o nome a atribuir à nova macro, após o qual a macro fica pronta a ser executada.

O processo de gravação de uma macro, embora possua diversos passos, é realizado apenas uma vez e quando terminado permite a execução de todas as acções da macro de forma rápida e fácil, num único clique. As macros criadas não são entidades fixas, podem ser alteradas e ajustadas a novas necessidades e preferências do utilizador, pelo que através da secção “Detalhes” o utilizador pode aceder ao ecrã de edição de cada macro. Este ecrã é apresentado na figura 25, imagem à direita, e nele

é possível modificar o nome da macro, adicionar ou remover acções da mesma e executar todas ou uma parte das acções que constituem a macro.



Figura 25 – Ecrãs de gravação (esquerda) e edição (direita) da macro.

A alteração do nome é realizada por clique na primeira opção do ecrã “Alterar Nome”, que abre uma secção onde se pode definir o novo nome e gravar o mesmo. A opção “Adicionar Acções”, uma vez clicada, inicia um novo período de gravação que quando termina adiciona as novas acções registadas à macro em questão. O final deste ecrã apresenta a lista de acções que constituem a macro, organizadas pela divisão dos dispositivos em que cada acção actua. Nesta lista o utilizador pode seleccionar um qualquer número de acções e dos botões disponíveis pode escolher “Executar”, “Remover” ou “Limpar” a selecção. Enquanto o primeiro botão executa, só e apenas, as acções seleccionadas; o segundo permite remover essas acções da macro. O botão “Limpar”, como o nome indica, limpa a selecção.

4.1.6 Propriedades em Destaque

A interface apresenta ao utilizador uma forma de controlar as propriedades de todos os dispositivos da sua habitação. No entanto, algumas das propriedades são mais importantes para o utilizador do que outras. Tendo em conta o grande número de dispositivos que podem existir numa casa e mesmo com a possibilidade de definir favoritos, é sempre necessário alguma navegação até se poder visualizar o estado de qualquer propriedade. Na realização desta interface considerou-se que esta situação não era satisfatória e que as propriedades importantes para o utilizador deveriam estar sempre acessíveis a este, incluindo os seus valores. A solução para esta situação foi a criação do conceito das propriedades em destaque.

Uma propriedade em destaque é uma propriedade de um qualquer dispositivo, cujo valor pode ser acedido à distância máxima de um clique. Os utilizadores podem ter um qualquer número destas propriedades e na secção “Propriedades em Destaque”, representada na figura 26, podem visualizar a lista das propriedades com todos os seus valores actuais. O acesso a esta secção é realizado através da imagem no canto inferior esquerdo (o pionés) de qualquer ecrã.

Este ecrã possui três secções: lista das propriedades importantes, detalhes e remover. Na primeira secção, além da visualização das propriedades e dos seus valores, é possível clicar em qualquer propriedade e aceder ao seu respectivo dispositivo. Quando o utilizador acede ao ecrã de um dispositivo deste modo, os controlos da propriedade em destaque seleccionada são “abertos” automaticamente na geração do ecrã. A secção detalhes permite a configuração das propriedades destacadas. Por fim, a secção remover, visível na figura 26 à direita, possibilita ao utilizador eliminar propriedades desta lista, após confirmação num ecrã adicional.



Figura 26 – Ecrãs de listagem e remoção das propriedades em destaque.

O acesso através de um clique à secção das propriedades em destaque não se revelou suficiente para a importância que algumas propriedades possam ter. Desta forma e para fornecer ainda um maior destaque às propriedades escolhidas pelo utilizador foi criada, no fundo de todos os ecrãs, uma secção que lista as diversas propriedades escolhidas. Esta listagem mostra a imagem associada e o valor/estado actual da propriedade, permitindo ainda o acesso directo ao dispositivo que a contém.

O número de propriedades apresentadas no fundo do ecrã depende da resolução do dispositivo móvel que acede à interface. Assim, uma vez que podem existir mais propriedades em destaque do que o número apresentado no ecrã, é dada a possibilidade ao utilizador de ordenar as diversas propriedades

de forma a serem apresentadas pela ordem escolhida. Na figura 27, do lado esquerdo, encontra-se o mecanismo de ordenação das diversas propriedades.

Cada uma destas propriedades é caracterizada por um nome e pode ter associada uma imagem e uma regra. Ao centro e à direita da figura 27, é possível visualizar a edição e caracterização de uma propriedade de destaque. Na configuração destas propriedades é possível definir um valor ou intervalo de valores em que cada propriedade se deve manter. Esta definição é considerada uma regra e sempre que o valor especificado não seja respeitado é alterada cor, no fundo do ecrã, da propriedade correspondente. O utilizador fica, desta forma, avisado sobre a situação. Na figura 27 ao centro indica-se que a porta da “Garagem” se deve manter fechada, como é possível validar no fundo de todas os ecrãs apresentados a propriedade encontra-se aberta e em consequência apresenta o fundo vermelho.



Figura 27 – Ecrãs de configuração das propriedades em destaque.

O utilizador pode adicionar qualquer propriedade a esta lista necessitando apenas de navegar até ao dispositivo com a propriedade em questão e clicando no pionés. Ao fazê-lo é-lhe apresentado um ecrã para especificar o nome a dar ao destaque e a propriedade a ser destacada, entre as disponíveis no dispositivo que ainda não se encontrem nesta situação. Na maioria das aplicações Web as informações contidas nas páginas só são actualizadas por acção do utilizador, no entanto e uma vez que estas propriedades se encontram sempre visíveis foi-lhes atribuído um modo de actualização via *Ajax*. Esta actualização ocorre de 10 em 10 segundos, sendo que este tempo é parametrizável no ficheiro de configurações da interface *pagecfg.php*.

4.1.7 Temas

Na realização deste projecto, mais propriamente no aspecto gráfico, foi utilizado CSS para definir classes e atributos associados a determinados elementos *HTML*. Esta tecnologia permite definir um conjunto de atributos, entre os quais a cor, o tamanho e o tipo de letra, para um determinado elemento ou conjunto de elementos com as mesmas características. Nesta interface foram criados três documentos CSS: um que contém todas as classes utilizadas, um que possui extras de visualização para *browsers* executados através de um computador e outro, mais simples, destinado aos *browsers* mais antigos que apresentam funcionalidades mais limitadas.

O uso de CSS permite isolar as configurações respeitantes ao aspecto gráfico num único módulo, tornando simples a criação de outros aspectos apenas pela substituição das configurações neste módulo. Em seguida apresenta-se, a título de exemplo, um conjunto simples de alterações que resultaram num novo aspecto para duas páginas da aplicação.

Considere-se os *selectors* *body* e *table*. No caso do *body*, o tema criado possui apenas a propriedade *background-image* que corresponde ao *url* da imagem a usar como fundo dos ecrãs. No caso da *table* foram definidas três propriedades: uma para o tipo de letra, outra para o tamanho do texto e outra para a sua cor. A definição destas propriedades é feita através do seguinte código:

```
body {  
    background-image: url("../images/background.gif");  
}  
  
table {  
    font-family: Tahoma, Trebuchet MS;  
    font-size: 11px;  
    color: #BBBBBB;  
}
```

A especificação de uma nova imagem de fundo ou um novo tipo de letra pode ser realizada pela alteração do código acima. O tipo de *selectors* usados afectam todos os elementos *HTML* com o mesmo nome, por exemplo no caso da *table* as propriedades definidas são aplicadas a todas as tabelas *HTML* existentes na aplicação. No entanto na maioria das vezes é necessário afectar apenas uma das tabelas, nestes casos no elemento *table* que queremos afectar define-se o atributo *class*, no código *HTML*, com um dado valor e no CSS restringe-se a aplicação das propriedades às tabelas que possuem esse valor. Por exemplo, considerando as barras que mostram a lista dos dispositivos, na figura 28 à direita. Estas barras são feitas através de elementos *td*, mas existem muitos outros elementos iguais que não queremos representar desta forma. Assim, foi definido o atributo *class* com o valor "bar" para os

elementos *td* a afectar e no CSS foram aplicadas as configurações estes elementos. De seguida encontra-se o respectivo código.

```
Código HTML:
<td class='bar'>...</td>

Código CSS:
td.bar {
    background: url("../images/barback.png") no-repeat;
    width: 215px;
}
```

O código CSS apresentado especifica que os elementos *td* com a *class* “bar”, definidos em código *HTML*, vão ter como imagem de fundo a “barback.png” e uma largura 215 pixeis. Manipulando apenas os dois conjuntos de dados indicados anteriormente e criando as imagens necessárias, foi possível alterar o aspecto gráfico da interface para o que se apresenta na figura abaixo.



Figura 28 – Exemplo do aspecto de um outro tema.

A arquitectura e implementação seguidas neste projecto permitem que com pouco trabalho se criem novas versões de temas e se mude radicalmente o aspecto gráfico da aplicação. Estas novas versões não têm que substituir as anteriores, podem ser apenas adicionadas à aplicação possibilitando que esta apresente diversos temas, de onde o utilizador pode escolher a que mais lhe agrada a cada momento. A adição de novos temas é conseguida através de dois passos. Primeiro é necessário criar um ficheiro na pasta “css” com as classes e atributos desejados. Por fim, é necessário adicionar o nome do novo tema à lista das existentes, isto é realizado no módulo “Configuração” (apresentado na secção 3.2.3) mais especificamente no ficheiro *pagecfg.php*.

4.1.8 Linguagens

A linguagem em que uma interface é disponibilizada determina o conjunto de utilizadores que conseguem compreender as suas designações, mensagens e funcionalidades. Com o intuito de alargar este conjunto de utilizadores o mais possível e estender o uso da interface a qualquer país, a aplicação desenvolvida no âmbito desta dissertação tem a capacidade de gerar uma interface multi-linguagem. Na figura seguinte apresenta-se o ecrã onde o utilizador pode configurar a linguagem da interface. Neste momento a aplicação encontra-se traduzida em duas linguagens, português e inglês.

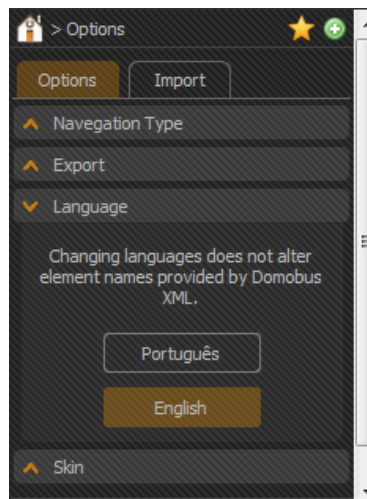


Figura 29 – Opções de configuração em inglês.

Este objectivo foi conseguido através da criação de ficheiros de linguagens, que contém uma lista de pares <código, mensagem> com todas as mensagens apresentadas em ecrãs da aplicação. O código da lista de pares é um identificador único da mensagem a apresentar e é igual em todos os ficheiros de linguagens existentes na aplicação. A mensagem, por sua vez, é diferente em cada um dos ficheiros e representa a tradução da mensagem na linguagem a que este se refere. As mensagens podem ser simples ou complexas. No primeiro caso são constituídas por uma designação ou frase completa. No segundo caso são frases que contém *tokens*, representados pelos caracteres “{#<número>}”, e que na execução da aplicação são substituídos por dados que dependem do utilizador e das interacções deste com a interface. A título de exemplo, no ficheiro de linguagens existe “Macro {#1} criada com sucesso”, esta mensagem é apresentada quando uma macro é inserida com sucesso mas a sua apresentação depende do nome que o utilizador lhe atribui, como por exemplo “Macro Luzes criada com sucesso”.

A adição de novas linguagens à aplicação é conseguida através de dois passos. Primeiro é necessário traduzir as mensagens de um dos ficheiros já existentes, para uma nova linguagem, e gravar esse ficheiro na pasta *languages* da aplicação. Depois, é necessário adicionar o nome da nova

linguagem à lista das que já são suportadas, isto é realizado no módulo “Configuração” (apresentado na secção 3.2.3) mais especificamente no ficheiro *pagecfg.php*. Na tradução de qualquer um dos ficheiros já existentes é necessário garantir que a estrutura do mesmo se mantém inalterada.

A multi-linguagem disponível na aplicação permite ao utilizador alternar entre as linguagens existentes. Contudo, os dados provenientes no ficheiro XML do DomoBus não são afectados pela alteração. Isto acontece porque não há modo de prever que nomes ou constantes serão recebidos no ficheiro. No entanto, se o sistema domótico se encontrar instalado numa habitação de outro país, como por exemplo na França, seria de esperar que os nomes das divisões, dispositivos, entre outros, se encontrassem na linguagem nativa desse país, no exemplo dado seria o francês, pelo que o ficheiro XML importado na interface teria os nomes na linguagem correcta.

4.1.9 Adaptabilidade à resolução de ecrã

A interface permite acesso remoto a um sistema domótico e permite fazê-lo através de qualquer dispositivo móvel ou fixo que consiga aceder à Internet por um *browser*. Existe, desta forma, uma grande diversidade de dispositivos que podem apresentar a interface e que vão desde telemóveis ou PDA's com ecrã relativamente pequeno, a consolas de jogos portáteis e computadores pessoais com ecrãs gradualmente maiores. A aplicação foi principalmente desenvolvida para ecrãs pequenos e médios, que o utilizador transporta facilmente consigo, no entanto esta pode ser executada em qualquer dispositivo, independentemente da dimensão e resolução do seu ecrã.

Na gama dos ecrãs pequenos e médios, a aplicação consegue adaptar-se às dimensões que tem disponíveis. No caso de ecrãs de tamanho superior, a aplicação adapta o seu conteúdo até um máximo definido, a partir do qual se mantém inalterada. Em qualquer ecrã, sempre que é necessário recorre-se a um *scroll* vertical, deste modo se existir uma divisão com um número elevado de dispositivos, a página gerada cresce em altura e apresenta uma listagem dos dispositivos em conjunto com o *scroll*. Com a apresentação de *scroll* vertical, qualquer página pode mostrar mais conteúdo do que a altura que tem disponível; no entanto, não se permitiu o mesmo relativamente à largura dos ecrãs. A adaptabilidade à resolução do ecrã, encontra-se assim ligada aos pixels em largura que o ecrã do utilizador disponibiliza.

As páginas da interface, com excepção da página principal em navegação por divisões, seguem um modelo genérico que apresenta os diversos serviços, dispositivos, propriedades e opções numa lista de barras com 215 pixels de largura. A largura destas barras é fixa, assim quando um ecrã apresenta uma largura maior do que a necessária para uma coluna de barras, a aplicação calcula o novo número de colunas que pode apresentar. Desta forma, para um ecrã maior em largura não é apresentado *scroll* vertical, mas sim um conjunto de colunas. Tal como indicado anteriormente, a adaptação foi desenvolvida para ecrãs pequenos e médios, pelo que o número de colunas apresentadas varia entre o mínimo de

uma e o máximo de três. A página principal em navegação por divisões, nunca apresenta *scroll* vertical, pois tem integrado um sistema de paginação, nesta página a adaptação à largura dos ecrãs é realizada através da apresentação de um maior número de divisões em linha e coluna. Nas figuras 30 a 33, apresenta-se o exemplo de todas estas adaptações em três dispositivos diferentes.



Figura 30 – Interface no PDA HTC Touch.



Figura 31 – Ecrã das divisões na consola PSP [46].



Figura 32 – Ecrã das propriedades da televisão na consola PSP.

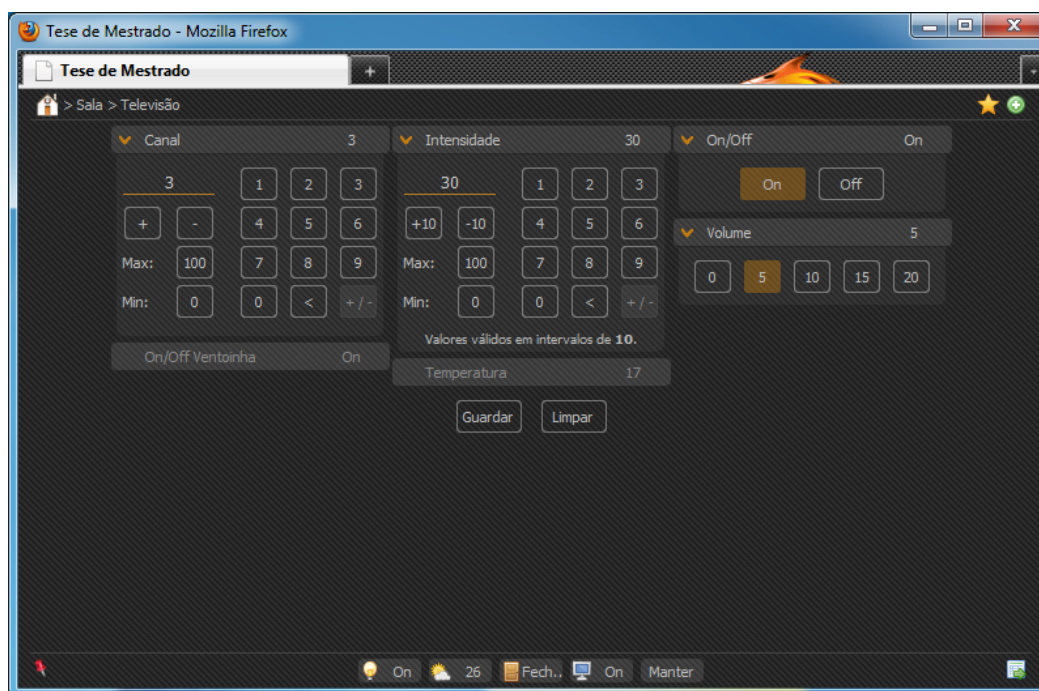


Figura 33 - Ecrã das propriedades da televisão no Mozilla Firefox.

A aplicação a ser executada é a mesma em todas as figuras, no entanto como está a ser acedida a partir de dispositivos com diferente largura de ecrã, a interface ajusta-se a cada um mostrando o mesmo conteúdo de formas diferentes. Deve indicar-se ainda o seguinte factor, estas figuras são simulações dos dispositivos reais constituídas por uma moldura e uma *iframe* que abre a aplicação real. Esta adaptabilidade foi testada no PDA HTC, no iPhone e em computador através da manipulação da dimensão da janela do *browser*.

4.1.10 Adaptabilidade do texto

Numa aplicação que permite múltiplas linguagens e que adapta o seu conteúdo à resolução e dimensão do ecrã em que é visualizada, surgiu a necessidade de flexibilizar o texto apresentado nas diversas designações, botões e na barra de endereço. A funcionalidade a desenvolver para cobrir esta necessidade tem como objectivo adaptar um qualquer texto a um qualquer tamanho, tendo sempre em vista que o resultado final seja o mais compreensível possível para o utilizador. Desta forma, foi desenvolvido um método que calcula a redução ou abreviação a aplicar a um dado o texto dependendo dos pixeis disponíveis no local onde o mesmo deve ser apresentado. Este método calcula os pixeis necessários para apresentar o texto completo e se estes ultrapassam os disponíveis no local, procede à redução do texto seguindo os passos apresentados, entre cada um é sempre validado se a redução já se encontra dentro do limite disponível:

1. Valida o texto como um todo e elimina palavras internas, aquelas que são precedidas por um espaço, iniciadas por minúsculas;
2. Nas palavras restantes, da primeira para a última, reduz carácter a carácter, do fim da palavra para o início, deixando sempre pelo menos a primeira letra de cada palavra;
3. No final, se a última abreviação realizada resultou numa palavra com mais que uma letra são colocados dois pontos no final da mesma.

Este processo é válido para o texto de todas as designações e botões apresentados, no caso do texto na barra de endereço é realizado um passo adicional, antes dos apresentados acima. O método de adaptação do texto, quando é executado para uma barra de endereço, começa por dividir o mesmo nas suas partes constituintes (separadas pelo símbolo '>') e é para cada uma dessas partes que executa os passos 1 a 3 explicados. Cada vez que uma parte é concatenada é validada a necessidade de concatenar as próximas, no final a barra de endereço apresenta a concatenação das diversas páginas desde a principal até ao local onde nos encontramos.

Na figura seguinte apresenta-se exemplos das abreviações efectuadas na interface, quando a mesma se encontra em português e é acedida pelo PDA HTC Touch. É possível verificar o texto abreviado na barra de endereço, no nome da divisão "Quarto de Visitas", na designação de uma acção de uma macro e numa propriedade em destaque.

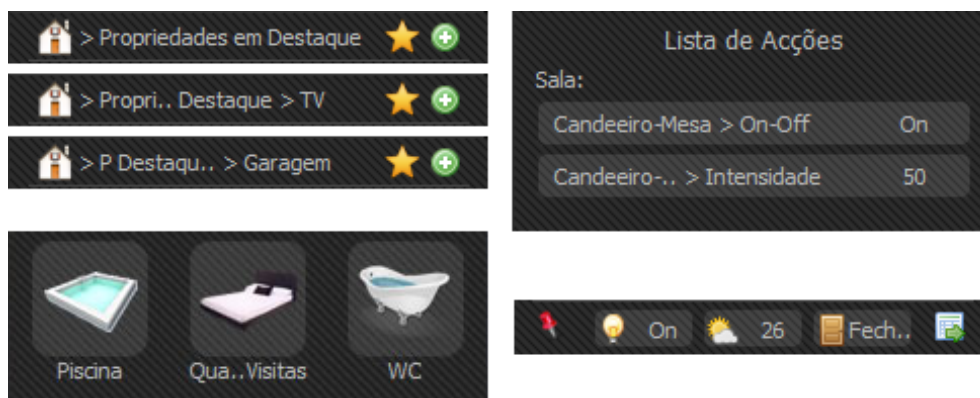


Figura 34 – Algumas representações de adaptabilidade do texto.

Na figura anterior apresenta-se a barra de endereço em três situações distintas, que influênciam o resultado do texto apresentado. Enquanto na primeira situação não foi necessário reduzir o texto e na segunda foi suficiente eliminar alguns caracteres da palavra “Propriedades”; na última situação o espaço disponível obrigou à redução máxima da palavra “Propriedades”.

4.1.11 Especificações XML

O sistema domótico DomoBus, como indicado anteriormente, exporta um ficheiro XML com os dados presentes no sistema, incluindo a lista de divisões, dispositivos e respectivas propriedades e ainda uma lista de utilizadores e os seus respectivos níveis de acesso. Estes dados são importados para a interface, por forma a inicializá-la com toda a estrutura do sistema domótico que irá controlar. Uma estrutura de exemplo deste ficheiro XML encontra-se no ‘Anexo A’. Nesta secção apresenta-se uma descrição das funcionalidades responsáveis por importar e exportar dados da interface e os possíveis formatos que o ficheiro XML pode apresentar.

4.1.11.1 Importar / Exportar

A interface pode importar e exportar dados no formato de um ficheiro XML. A funcionalidade de importação permite inicializar e actualizar a interface com toda a informação necessária ao seu funcionamento, nomeadamente as divisões, dispositivos e utilizadores existentes na habitação a controlar. A funcionalidade de exportação, permite guardar a qualquer momento estes mesmos dados e ainda configurações específicas que o utilizador tenha realizado na interface, como definição de favoritos, macros e propriedades em destaque. Na figura 35 apresentam-se os ecrãs onde é possível realizar as operações indicadas. Estas funcionalidades só podem ser usadas em dispositivos que disponibilizam gestores de ficheiros; no caso do iPhone em que não é possível armazenar ficheiros XML sem auxílio de uma aplicação própria, as secções de importar e exportar não são visíveis no ecrã.



Figura 35 – Ecrãs de exportação e importação.

No caso da importação, é disponibilizada a opção de manter as configurações específicas à aplicação, já existentes no momento em que o novo ficheiro XML é importado. Se o utilizador escolher não manter estas configurações, todas as informações existentes na Interface são perdidas e são carregadas novas, sendo que a aplicação fica inicializada com os dados provenientes no ficheiro XML.

4.1.11.2 Formato Original do ficheiro

O ficheiro XML proveniente do sistema DomoBus [14] encontra-se organizado em diversas secções. Em seguida são explicados os campos das secções relevantes para a interface.

- *House* – caracterizado por um nome e por uma lista de divisões agrupadas nos diversos andares da habitação. Cada divisão é caracterizada por um nome e um nível de acesso mínimo para consulta.
- *ScalarValueTypeList* – lista de definições de valores de grandeza escalar, sendo caracterizado obrigatoriamente pelo nome e opcionalmente por valores mínimo e máximo e valor mínimo de incremento (*step*).
- *EnumValueTypeList* – lista de enumerados, cada um caracterizado por um nome e uma lista de estados possíveis. Esta última é caracterizada pelo nome e valor interno associado.
- *DeviceTypeList* – lista dos diversos tipos de dispositivos existentes, considere-se um dispositivo do tipo Lâmpada-Regulada. Cada um possui uma lista dos tipos de propriedades associadas, para este caso o Estado e a Intensidade. Para cada propriedade é especificado o modo de acesso (só leitura ou leitura e escrita) e o tipo de valor (grandeza escalar ou enumerado), no exemplo apresentado o Estado é do tipo enumerado (*On* e *Off*) e a Intensidade é uma grandeza escalar (entre 0 e 100).

- *ServiceList* – lista de serviços existentes, tais como Segurança, Entretenimento, entre outros.
- *DeviceList* – lista dos diversos dispositivos existentes no sistema DomoBus. Estes são caracterizados pelo nome, divisão em que se encontram, níveis mínimos de acesso e escrita e a lista de serviços a que pertencem.
- *UserList* – contem a lista de utilizadores caracterizada pelo nome, palavra-chave e nível de permissões. A palavra-chave encontra-se codificada segundo uma *hash*, que para este projecto foi considerado o *md5* [47].

4.1.11.3 Novo Formato do ficheiro

O formato original do ficheiro de XML possui os dados conhecidos pelo DomoBus do sistema domótico presente na habitação. Estes dados referem-se aos utilizadores, às divisões, aos serviços e aos dispositivos e suas propriedades. No entanto, como referido anteriormente, um dos objectivos da interface é oferecer ao utilizador a possibilidade de configurar e personalizar o sistema domótico de acordo com as suas necessidades e preferências. A informação referente às configurações do utilizador, uma vez definidas por este, não deve ser perdida. Desta forma, optou-se por modificar o formato original do ficheiro XML para contemplar as configurações definidas na interface. Todos os campos adicionados são opcionais, tornando a aplicação compatível com ambos os formatos.

Na interface existem três grandes componentes de configuração: as opções, os favoritos, as macros e as propriedades em destaque. Cada um destes componentes originou um elemento novo no XML, como é explicado de seguida.

- *OptionsList* – este elemento contém as opções dos utilizadores, através de uma lista de elementos *Option*. Cada linha indica o modo de navegação, o tema (*skin*) e a linguagem de um utilizador.
- *MacroList* – neste elemento são guardadas as macros de cada utilizador, através de uma lista de elementos *Macro* caracterizados pelo nome. Por sua vez, cada um destes elementos possui uma lista de elementos *Action*. Uma *Action* é caracterizada pelos identificadores do dispositivo e propriedade a alterar e pelo valor a atribuir à propriedade.
- *BookmarkList* – a lista de favoritos de cada utilizador é guardada neste elemento. Cada item da lista corresponde a um elemento *Bookmark* e é caracterizado pelo identificador do dispositivo, divisão ou serviço a que se refere (em cada elemento apenas um destes identificadores é usado).
- *ImpPropertyList* – este elemento guarda as propriedades em destaque de todos os utilizadores e é constituído por uma lista de elementos *ImpProperty*. Cada um destes elementos é caracterizado pelo nome e imagem associada, pelo dispositivo e propriedade a que se refere, pela ordem de visualização e pela condição de validação e valor a validar.

Os quatro elementos apresentados referem-se sempre a um utilizador. Assim, apesar de não ser referido na lista anterior, cada um dos elementos *Option*, *Bookmark*, *Macro* e *ImpProperty* contem também um identificador do utilizador a que se encontra associado e uma data de criação. Além dos elementos novos identificados, foram também adicionados alguns atributos a elementos já existentes.

Na lista de divisões, elemento *Division*, foi adicionado o atributo *Image* onde é especificado o nome de uma imagem ou um URL. O primeiro caso é usado se a imagem existir no caminho “/images/divisions/” a partir da localização onde se encontra a aplicação. O URL define o endereço da imagem, caso a mesma se encontre na Internet. Como explicado no Sistema de Navegação (em 4.1.1), se nenhuma divisão possuir este campo a aplicação mostra as divisões numa lista de texto e se apenas algumas o possuírem, nas restantes é utilizada uma imagem genérica.

Nos dispositivos foi adicionado mais um parâmetro ao atributo *AcessLevel*. Este atributo passa a poder ter três valores que representam os níveis mínimos exigidos ao utilizador para acesso, modificação e modificação via interface remota do dispositivo. O parâmetro adicionado permite limitar o controlo de alguns dispositivos via interface remota, mesmo que o utilizador tenha permissão de modificação do dispositivo não a poderá exercer pela interface. Para exemplificar esta situação considere-se um forno, embora dentro de casa possa fazer sentido ser controlado por vários utilizadores, através da internet deve ter um acesso mais restrito devido aos perigos associados com este tipo de dispositivo. Para a interface o valor a considerar para as permissões de modificação é sempre o máximo entre os valores definidos.

Nos utilizadores foi adicionado o atributo *ImportAccess*, este pode receber “0” (por omissão) ou “1” e determina se o utilizador pode ou não importar novos ficheiros XML. Esta restrição foi definida devido à possibilidade de eliminação de dados que a funcionalidade de importação comporta, pelo que não deverá estar disponível para todos os utilizadores.

4.2 Bibliotecas da Interface

Na implementação deste projecto foram seguidas várias técnicas de engenharia de software, das quais se destacam modularização e hierarquia. O objectivo da modularização é tornar cada classe como uma “caixa negra” onde se conhecem os dados de entrada e os possíveis dados de saída, mas não é necessário conhecer o seu processo interno. Uma hierarquia é uma estrutura em camadas onde as representações mais abstractas de conceitos se encontram no topo e representações incrementalmente mais detalhadas e especializadas no fundo [48]. Tendo em conta estas técnicas, foram criadas as bibliotecas de “Comunicações” e “Base de Dados” detalhadas de seguida. Foram ainda criadas as

bibliotecas “General” e “General Javascript” que contém as funções mais utilizadas da aplicação. Por fim, introduz-se a framework jQuery [49] referindo-se a necessidade da sua utilização.

4.2.1 Comunicações

As comunicações entre o servidor e o simulador são efectuadas através do protocolo TCP/IP. Por forma a separar o modo de comunicar com o simulador, da lógica associada à interface foi criada uma biblioteca *comclass.php* com o método *send*. Este método recebe a mensagem que a aplicação necessita de enviar e é responsável pelos passos que garantem o envio da mensagem e recepção da resposta à mesma; nomeadamente cria um canal de comunicação TCP/IP – designado por *socket* – estabelece a conexão, envia a mensagem, aguarda resposta e uma vez recebida retorna-a.

A inexistência de lógica aplicacional na biblioteca referida permite alterar facilmente o modo de comunicação da aplicação, sendo apenas necessário substituir esta biblioteca por outra que forneça a mesma assinatura – nome do método e parâmetros iguais. Esta abordagem, no entanto, tornou necessária a criação de uma segunda biblioteca com a lógica aplicacional associada às comunicações. Esta designa-se por *communications.php* e estende a funcionalidade da *comclass.php*. Esta segunda biblioteca é constituída por três métodos distintos:

- *getPropValues* – este método recebe uma lista de propriedades e constrói a mensagem *get*, detalhada na secção 3.3 Simulador DomoBus, que obtém o estado das propriedades dos diversos dispositivos. De seguida é invocado, na classe original, o método de envio com a mensagem construída e uma vez recebidos os resultados, o método retorna a lista do estado das propriedades. Caso ocorra algum problema é retornada uma lista de erros.
- *validateNewValue* – recebe as configurações de uma propriedade e o valor que se pretende atribuir a essa propriedade. O método é responsável por validar se o valor recebido se encontra de acordo com as regras da propriedade.
- *setPropValues* – este método recebe uma lista de propriedades e uma lista de valores e para cada propriedade invoca o método *validateNewValue*. De seguida, constrói a mensagem *set*, detalhada na secção 3.3 Simulador DomoBus, contendo as instruções das propriedades que respeitam as suas regras. Por fim, envia a mensagem através do método *send* da biblioteca original e uma vez recebidos os resultados retorna uma lista com todas as propriedades alteradas com sucesso. Neste método, antes da mensagem ser enviada, é validado se a aplicação está num período de gravação de macro e em caso afirmativo as instruções são apenas gravadas em memória.

4.2.2 Base de Dados

A base de dados guarda toda a informação apresentada na interface, com excepção dos valores das propriedades. Desta forma, sempre que existem pedidos ao servidor é necessário aceder à base de dados para obter os dados necessários à geração da nova página. Todos estes acessos requerem métodos específicos que dependem da tecnologia de base de dados usada. Assim, foi criada a biblioteca *dbclass.php* que possui métodos de conexão, início e fim de transacções, execução de pedidos e tratamento de erros. Esta biblioteca não possui qualquer lógica aplicacional, pelo que pode ser substituída por outra que forneça a mesma assinatura de métodos e utilize um qualquer sistema de base de dados.

Da mesma forma que foi necessário expandir a biblioteca *comclass.php*, na secção anterior, também se estendeu a biblioteca *dbclass.php*. A extensão criada designa-se por *database.php* que para além de adicionar novos métodos à biblioteca original, também redefine alguns. Esta biblioteca possui três métodos: o primeiro redefine o tratamento de erros original, o segundo é responsável por invocar o método de conexão com dados específicos da aplicação e o terceiro é um método que valida se a base de dados da aplicação se encontra criada.

No primeiro caso procedeu-se à redefinição do tratamento de erros por forma a realizar-se uma maior análise destes e ser possível apresentar mensagens mais detalhadas dos problemas ocorridos. O segundo método foi criado devido ao uso de configurações específicas da aplicação, que não poderiam existir no método original, como o endereço, *user* e *password* da base de dados utilizada. Finalmente, o último método valida se a base de dados já se encontra criada como forma de prevenir erros e identificar prontamente ao utilizador a acção que o mesmo tem de tomar para colocar a aplicação em funcionamento.

4.2.3 General

A biblioteca *General* agrupa diversas funções globais a toda a aplicação. Em seguida apresenta-se uma descrição sucinta das funções mais importantes:

- *debug* – esta função tem o objectivo de facilitar a depuração da aplicação. Recebe uma lista de variáveis e, independentemente do tipo, apresenta o seu valor no ecrã da aplicação de uma forma legível ao programador. Na solução final esta função não é executada, mas optou-se por manter a sua presença na lista na eventualidade de surgir algum problema ou de surgir algum interesse em expandir este projecto.
- *isNumeric* – recebe uma lista de uma ou mais variáveis e valida se todas correspondem a valores numéricos. Esta função é utilizada na validação dos argumentos das chamadas *HTTP*,

aumentando o nível de segurança da aplicação, uma vez que estas chamadas podem ser manipuladas por utilizadores mais experientes.

- *show* – é responsável por receber uma mensagem do ficheiro de linguagens e uma lista de valores e imprimir essa mensagem no ecrã com os *tokens* substituídos pelos valores recebidos. A explicação dos *tokens* encontra-se na secção 4.1.8 Linguagens. A título de exemplo, na mensagem “Macro {#1} executada com sucesso” é responsável por substituir o “{#1}” pelo nome da macro e imprimir a mensagem no ecrã.
- *showWarnings* – recebe uma lista de mensagens e imprime-as no topo da página, sobre um fundo amarelo, alertando o utilizador para o sucedido.
- *showTabs* – responsável por construir as áreas de *tabs* (secções). Recebe uma lista com os nomes das diversas secções a apresentar e qual a *tab* seleccionada.
- *showHoverCSS* – esta função valida se a aplicação está a ser acedida através de um *browser* de um computador pessoal e em caso afirmativo carrega um conjunto de efeitos adicionais que destacam os locais por onde o ponteiro do rato passa.
- *parseSting* – função responsável pela funcionalidade descrita no ponto 4.1.10 Adaptabilidade do texto. Recebe um texto e o número de pixels disponíveis, e retorna o texto adaptado aos pixels disponíveis.

4.2.4 General Javascript

O código da aplicação foi desenvolvido, na sua maioria, em *PHP*. Este é interpretado pelo servidor gerando um documento *HTML* que é enviado para o *browser* do utilizador. Este processo produz uma página estática, no entanto na aplicação pretende-se uma interface dinâmica, rápida e de agradável manipulação. Com esta finalidade, foi desenvolvido código complementar em *Javascript*. Este código é executado no *browser* do utilizador e pode depender das acções deste ou da passagem de um determinado período de tempo.

Na aplicação foram criados dois ficheiros que contêm todo o código *Javascript* utilizado pela aplicação. O primeiro é um ficheiro genérico com a definição de todas as funções que a aplicação utiliza nesta linguagem. O segundo é incluído apenas em *browsers* mais antigos e destina-se a redefinir algumas das funções do primeiro ficheiro de modo a que, embora mais limitadas, funcionem.

Em seguida destacam-se algumas das funções mais relevantes presentes nestes ficheiros, com uma pequena descrição associada:

- *displayLoading* – esta função coloca uma cortina/nuvem sobre a página e uma imagem identificativa de pedido em processamento, como é visível na figura 36. Este método é redefinido para *browsers* mais antigos, pois devido às suas limitações não conseguem apresentar uma nuvem semitransparente por cima de elementos já existentes. Para estes casos a nuvem não tem transparência.



Figura 36 – Ecrã com nuvem em processamento de um pedido *Ajax*.

- *getData* – função responsável por efectuar pedidos *Ajax*. Esta função recebe um argumento que envia no pedido *HTTP* ao servidor, identificando o tipo de acção realizada pelo utilizador. Ao ser executada esta função invoca o *displayLoading* detalhado no ponto anterior.
- *ajaxDone* – esta função é invocada quando se recebe resposta ao pedido feito pela *getData*. Esta função retira a cortina e actualiza a área de conteúdo e a barra de endereço.
- *getImpProp* – responsável pela constante actualização das propriedades em destaque, através de pedidos *Ajax* de 10 em 10 segundos, tempo que pode ser parameterizado nas configurações da aplicação.
- *impPropValues* – esta função é chamada quando se recebe resposta ao pedido feito pela *getImpProp*. Esta função interpreta os resultados e actualiza a secção das propriedades em destaque para mostrarem os novos valores.
- *menuToggle* – responsável pela abertura e fecho do menu. Nesta função invoca-se uma animação da biblioteca *jQuery*, explicada na próxima secção. No caso de incompatibilidade, esta

função é redefinida por uma que apenas mostra ou esconde o menu, sem qualquer tipo de animação.

- *getWindowSize* – função responsável por identificar a largura disponível em pixels, do ecrã onde a interface é acedida. Sendo informada do “espaço” disponível a interface consegue aproveitá-lo da melhor forma, como explicado no ponto 4.1.9.

Destacam-se de seguida as funções responsáveis pela edição das propriedades de qualquer dispositivo, como por exemplo o canal de uma televisão presente na figura 37.



Figura 37 – Manipulação do canal de televisão.

- *objProp* (classe) – no *Javascript* cada propriedade foi definida como instância deste objecto. Esta classe é caracterizada por um identificador único, podendo também ser definidos valores mínimo e máximo e o intervalo (*step*).
- *parseSpecialCommand* – esta função recebe uma propriedade, sob a forma de um *objProp*, e um comando “especial” e retorna o valor final da propriedade tendo em conta o comando efectuado e as regras da mesma. O comando “especial” são operações a efectuar à propriedade, que se podem visualizar na figura 37, e inclui os seguintes casos: uma mudança de sinal, o incremento ou decremento de uma unidade, a definição directa para o valor mínimo ou máximo e a remoção de um dígito.
- *getNewValue* – esta função é responsável por calcular o novo valor a apresentar no mostrador da propriedade, representado pelo “3” sobre a barra laranja na figura 37. A função valida se o utilizador pressionou um dígito entre “0” e “9” ou se utilizou um dos outros comandos. No primeiro caso calcula o valor final da propriedade tendo em conta as regras existentes, no segundo caso invoca a função *parseSpecialCommand*.

- *addPropValue* – invocada quando o utilizador carrega em qualquer botão para manipulação de uma propriedade escalar. Esta função invoca o *getNewValue* e define as cores a apresentar no ecrã conforme seja uma modificação válida (verde) ou inválida (vermelho).

4.2.5 jQuery

O *jQuery* é uma *framework* gratuita de *Javascript*, compatível com todos os *browsers* mais recentes. Entre outras funcionalidades, esta *framework* disponibiliza um conjunto variado de animações pronto a utilizar em qualquer página *web* [49]. No caso específico desta interface, a *framework* é utilizada para produzir efeitos como abrir e fechar os controlos de uma propriedade ou abrir e fechar o menu de topo. No primeiro caso, quando o utilizador expande uma propriedade de um dispositivo, os detalhes deslizam de cima para baixo, dando a sensação de uma abertura fluida ao utilizador. No segundo caso, o menu ao aparecer desliza da esquerda para a direita e desaparece posteriormente num efeito inverso.

Uma vez que foi identificado que *browsers* mais antigos e algumas edições para telemóvel não suportam as funcionalidades desta *framework*, optou-se por, quando identificados estes casos, excluí-la por completo dos dados enviados para o *browser*, poupando tráfego e processamento. Nestes casos a animação é substituída pelo aparecimento / desaparecimento simples dos detalhes e menu.

4.3 Simulador

A interface foi desenvolvida para controlar remotamente uma habitação com o sistema domótico DomoBus. No entanto, como o DomoBus é também um projecto em desenvolvimento, na realização desta interface optou-se pela criação e uso de um simulador. Este simulador foi desenvolvido em Java e permite a realização de testes à aplicação sem se depender do uso de um sistema real. Note-se que a sua utilização não altera o comportamento da interface, esta não tem conhecimento de que se encontra a comunicar com um simulador. De facto, o uso do simulador torna o comportamento da aplicação o mais similar possível ao uso do sistema DomoBus, incluindo a comunicação via protocolo TCP/IP.

O simulador, como referido na secção da arquitectura, é constituído por três módulos: Controlo, Dispositivos e Comunicações. Destacando-se ainda o sub-módulo responsável pelo carregamento de XML, que faz parte dos Dispositivos. De seguida apresenta-se uma descrição sucinta da implementação destes módulos.

O módulo Dispositivos contém toda a lógica aplicacional do simulador. No entanto, para conseguir simular o DomoBus necessita de carregar um conjunto de informação contida em dois ficheiros XML – o Dados.xml e o Defaults.xml – existentes na raiz da aplicação. O primeiro documento, exportado pelo

DomoBus ou pela Interface, fornece ao simulador os dados relativos aos dispositivos e suas propriedades. Com o carregamento deste documento o módulo Dispositivos constrói em memória uma lista de todos os dispositivos e propriedades a simular, no entanto nenhuma propriedade possui valor associado. O segundo documento, Defaults.xml, tem como objectivo inicializar todas estas propriedades com um valor por omissão. A resolução desta situação poderia passar pela geração aleatória de um valor para cada propriedade, contudo esse processo poderia gerar valores irrealistas e o objectivo do simulador é tornar o funcionamento da interface o mais aproximado possível da realidade. Considere-se por exemplo a temperatura de duas divisões ligadas, com valores aleatórios uma poderia encontrar-se a 5º e outra a 40º influenciando negativamente a credibilidade e coerência da aplicação.

Os dados carregados são organizados, no módulo Dispositivos, na seguinte estrutura de classes:

- *DeviceProperty* – esta classe representa uma propriedade sendo caracterizada pelo valor actual e pelas regras associadas à mesma.
- *Device* – esta classe é a representação de um dispositivo e é constituída por uma lista de propriedades *DeviceProperty*. Esta classe é responsável por disponibilizar métodos de leitura e de alteração para cada propriedade.
- *DeviceList* – responsável por invocar o sub-módulo de carregamento dos ficheiros XML e criar os diversos *Device* e *DeviceProperty* correspondentes à informação obtida. Esta classe mantém uma lista de todos os dispositivos simulados e disponibiliza acesso aos métodos de leitura e alteração existentes em cada *Device*

O módulo Controlo é a interface do simulador, permitindo listar e manipular os diversos dispositivos simulados. Este módulo disponibiliza uma consola e através do comando *list*, primeiro comando representado na figura 38, é possível visualizar todos os dispositivos, propriedades e valores que o simulador possui em memória.

A consola permite ainda efectuar diversas operações que acrescentam realismo à simulação em curso. Estas operações incluem alterar o valor de uma propriedade (*set <DevId> <PropId> <Value>*), simular lentidão nas respostas (*slow <milliseconds>*), obter algumas estatísticas de utilização (*stats*) e definir se é pretendido causar alguns erros nas respostas enviadas à interface (*error <0 or 1>*). Na figura seguinte é possível verificar o resultado dos comandos *list* e *stats*.

```

> list
DevId  PropId  Device / Prop                                Value
-----
   1           Luz
       1      * On-Off                          1
       2      * Intensidade                     50
           ...

  10           Aparelhagem
       17     * On/Off                          1
       18     * CD                             1
       19     * Frequência                       88
       20     * Volume                          5
-----
Set Value:  set <DevId> <PropId> <Value>
> stats
Messages received: 9
           Gets: 38           Sets: 8
Sleep Time: 0      (slow <milliseconds>)
Error Mode: Off   (error <0-1>)
-----
>

```

Figura 38 – Consola do Simulador (Fundo Branco)

O módulo Comunicações é responsável por criar uma ligação TCP/IP e aguardar pedidos com instruções de alteração ou leitura de propriedades. É neste módulo que são recebidas as instruções de *get* e *set* de propriedades provenientes da interface e é aqui que as mensagens são interpretadas para que o módulo Dispositivos reconheça os valores a retornar como resposta. Uma vez determinados estes valores, este módulo é responsável por responder à interface. É através deste módulo que o simulador fica disponível para comunicar com a interface.

O simulador foi construído para simular o DomoBus a nível de comunicações. Isto incluiu, manter um conjunto de dispositivos com propriedades e valores, receber e responder a pedidos de leitura ou modificação de propriedades. A implementação realizada cumpre com este objectivo, não tendo sido consideradas lógicas adicionais como a localização dos dispositivos na habitação ou uma interface gráfica para monitorização e controlo do simulador. Destaca-se ainda que com esta implementação é possível simular qualquer habitação, com qualquer conjunto de dispositivos e estados desejados.

4.4 Diagrama de Interação

Nesta secção demonstra-se uma interacção típica entre os vários elementos que compõem este projecto. O *Browser* que apresenta a interface ao utilizador em qualquer dispositivo com conexão à Internet, o *Web Server* que executa a aplicação responsável por gerar a interface e o Simulador DomoBus que contém os dispositivos domóticos e os seus estados como se de uma habitação se tratasse. A figura 39 apresenta o diagrama em questão, com o utilizador João a interagir com a interface no seu PDA.

O utilizador João navegou pela interface e encontra-se na página da “Sala” onde decide visualizar as propriedades da televisão e possivelmente altera-las. O diagrama de interacção na figura 39 pode ser decomposto em duas fases. Na primeira o utilizador acede à página com as propriedades da televisão e os valores correspondentes têm de ser lidos do simulador. Na segunda fase o utilizador realiza uma alteração na interface e grava essa alteração que por sua vez tem de ser gravada no simulador, no dispositivo correspondente. Começa-se por apresentar os passos relativos à consulta da televisão:

1. Na lista de dispositivos da sala o João clica na “Televisão”
2. A interface, no *browser*, invoca um url com indicação do dispositivo escolhido pelo utilizador
3. O *web server* recebe o pedido e interpreta-o. Como o utilizador está a aceder às propriedades de um dispositivo é necessário consultar todos os valores correspondentes às mesmas. O *web server* envia uma mensagem ao simulador com o *get* de todas as propriedades.
4. O simulador analisa a mensagem e valida a ordem em que as propriedades são pedidas, consulta os estados correspondentes e constrói uma resposta com os valores de cada propriedade na mesma ordem que o pedido. O simulador responde ao *web server* com a mensagem construída.
5. O *web server* constrói a nova página a mostrar ao utilizador, com as propriedades da televisão e os valores respectivos. No final envia o código HTML da página para o *browser*.

Neste momento o utilizador tem na sua interface a lista de propriedades da televisão da sala. Uma vez que decide alterar o canal e o volume da televisão inicia uma nova interacção como se descreve de seguida:

1. Na lista de propriedades da televisão o João altera os valores do canal e do volume e grava.
2. O *browser* invoca um URL com indicação do dispositivo escolhido e os novos valores das propriedades.
3. O *web server* recebe e interpreta o pedido e envia uma mensagem ao simulador com o *set* dos novos valores das propriedades.

4. O simulador analisa a mensagem e altera os valores indicados na mesma. Quando termina responde com sucesso ou não da operação para cada propriedade modificada.

A aplicação a executar no *web server* recebe as confirmações e inicia um novo processo de consulta. Após este momento as interações são idênticas ao processo de consulta na primeira fase, realizando-se os passos 3 a 5 da fase de consulta de propriedades.

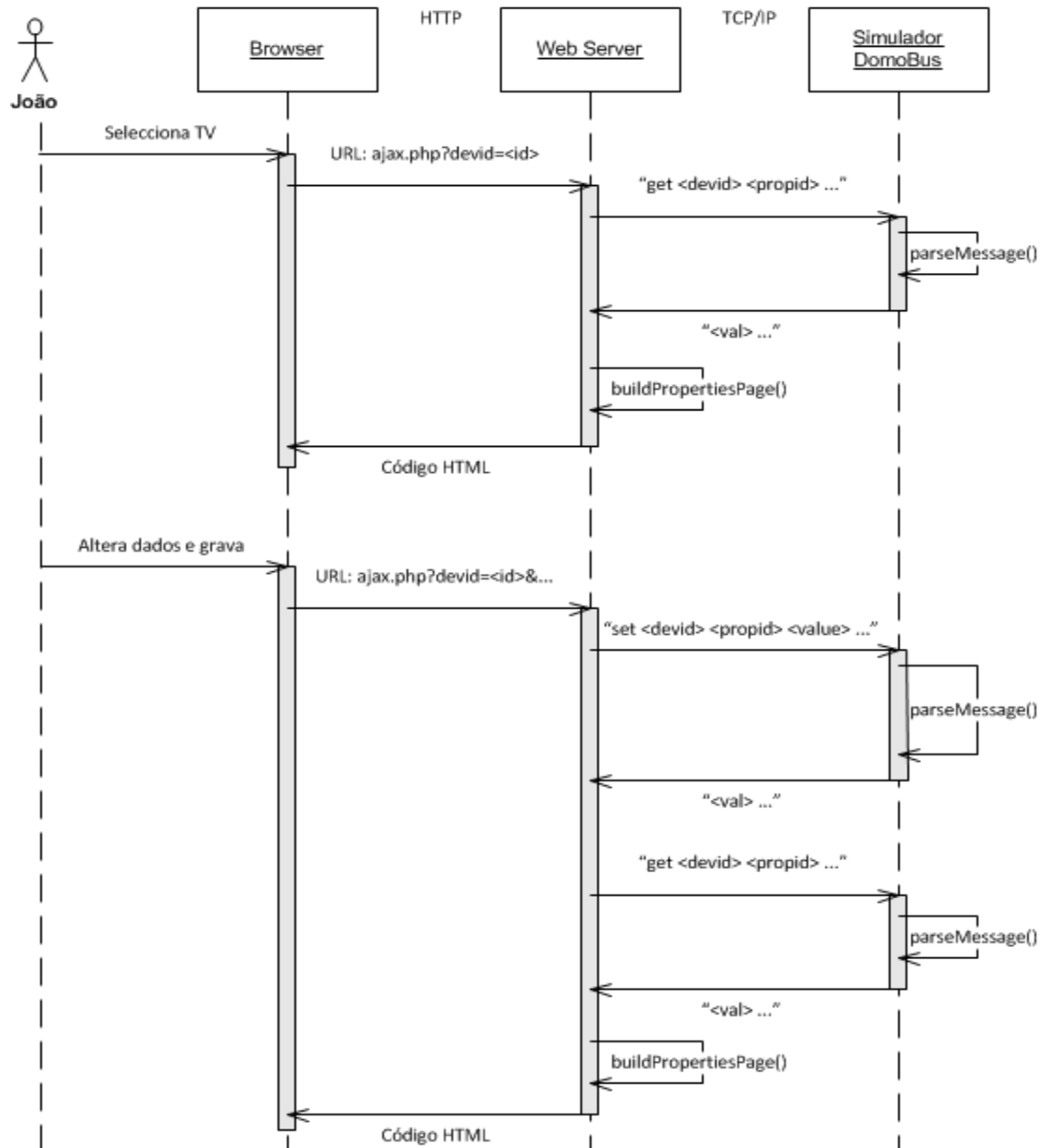


Figura 39 – Interação entre os diversos componentes do projecto

Capítulo 5

Avaliação da Aplicação

A aplicação desenvolvida no âmbito desta dissertação tem como principal objectivo fornecer uma interface remota para o sistema domótico DomoBus. Como qualquer interface esta aplicação é o elo de ligação entre os utilizadores e a máquina. Desta forma consideram-se dois grupos de teste a realizar à aplicação: testes de utilização e de desempenho. Os primeiros focam a interacção dos utilizadores com a interface e as suas opiniões e constatações durante a utilização da aplicação. Nos testes de desempenho e tendo em conta que a aplicação é uma interface remota idealizada para dispositivos móveis, foram testados factores relacionados com o processamento dos dispositivos e a sua ligação à Internet.

5.1 Testes de utilização

A opinião dos utilizadores é o factor mais importante numa interface, pois são estes que decidem usar ou não a aplicação. Desta forma, durante o desenvolvimento contou-se com um conjunto de utilizadores que interagiram com a interface e aos quais foi pedido que avaliassem a sua usabilidade, aspecto gráfico e utilidade. As opiniões recebidas destas interacções permitiram uma evolução da aplicação entre alguns protótipos, até se ter atingido o aspecto e funcionalidade final que se apresenta nesta dissertação.

Uma vez atingido o estado final da aplicação, a mesma foi testada com um novo conjunto de utilizadores (num total de 10) que nunca a tinham visualizado nem experimentado. A estes utilizadores explicou-se o projecto e os seus objectivos e foi-lhes entregue um questionário organizado em três secções: contexto, tarefas e classificação. Para a realização deste questionário entregou-se aos utilizadores um iPhone, onde estes interagiram com a interface por forma a completar as quatro tarefas especificadas.

O questionário completo encontra-se no ‘Anexo B’, apresentando-se de seguida as tarefas que se pediu a cada utilizador para cumprir:

1. Mudar o canal da Televisão da Sala para o canal 6.
2. Visualizar todas as luzes da Casa.

3. Colocar a luz do Hall nas propriedades em destaque com o nome "AB", propriedade "On-Off" e com a regra de manter-se "Off".
4. Criar Macro que liga as luzes da Garagem e do Hall com o nome "GH" e executá-la.

O iPhone foi entregue aos utilizadores com o *browser* a apresentar a página principal da aplicação, após o início de sessão. Esta página corresponde à visão da casa por omissão, ou seja organizada por divisões. Na realização das tarefas foram consideradas duas medidas os cliques necessários para a conclusão com sucesso da tarefa e o tempo despendido pelo utilizador nesta realização. Note-se que no iPhone o utilizador interage com a interface através de toques de dedo, mas para este teste consideram-se como cliques.

5.1.1 Resultados das tarefas

Apresenta-se de seguida dois gráficos com os resultados das tarefas realizadas. No primeiro gráfico pode visualizar-se o número mínimo de cliques necessários para a conclusão de cada tarefa e o número médio que o grupo de utilizadores, na primeira interação, necessitou. No segundo gráfico pode verificar-se o tempo mínimo que um utilizador experiente necessita para completar cada tarefa e o tempo médio que o grupo de utilizadores inexperientes despendeu.

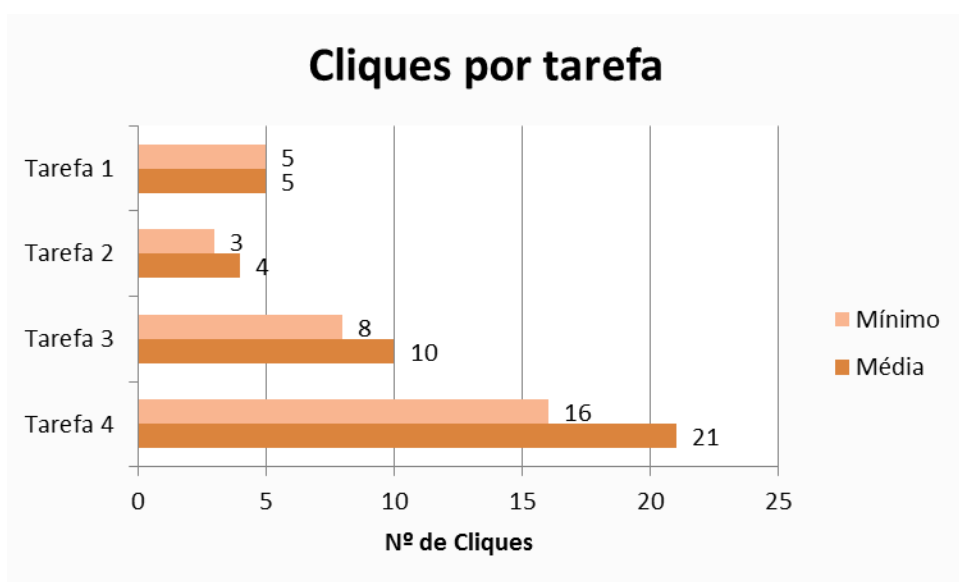


Figura 40 – Representação do número de cliques por cada tarefa

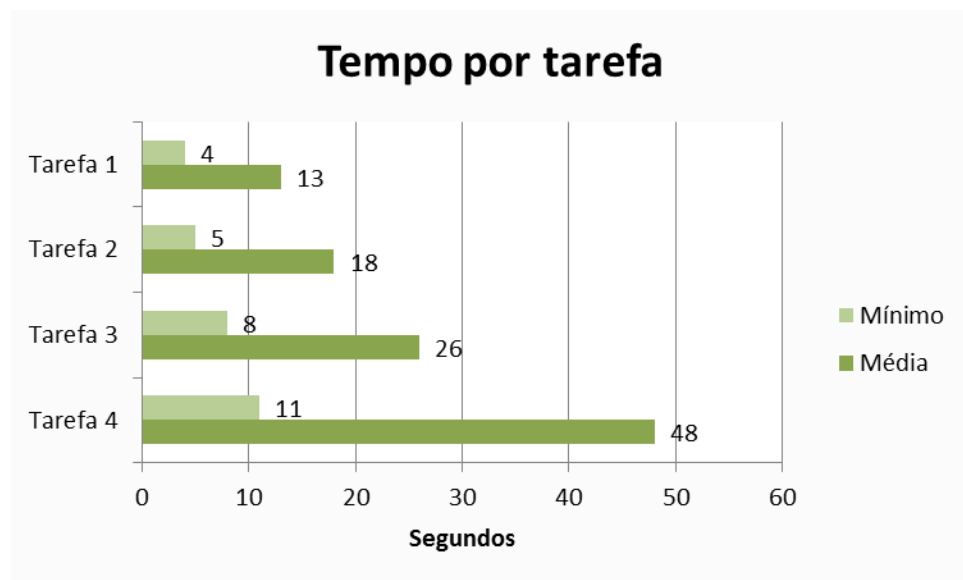


Figura 41 – Representação dos segundos despendidos por tarefa

Note-se que os tempos apresentados neste segundo gráfico, contam não apenas com o tempo que o utilizador demora entre todos os cliques, mas também com o tempo que os ecrãs demoram a ser apresentados e com as comunicações efectuadas entre a interface e o simulador DomoBus.

5.1.2 Análise de resultados

As quatro tarefas especificadas encontram-se organizadas por ordem crescente de complexidade e na ordem inversa de frequência. Enquanto as duas primeiras são tarefas que o utilizador poderá utilizar no seu dia-a-dia, as duas últimas são tarefas que por norma o utilizador realiza apenas uma vez na configuração da sua interface.

Na tarefa 1, verificou-se que todos os utilizadores conseguiram realizar a tarefa com o número mínimo de cliques. Tal como identificado previamente, a aplicação foi apresentada aos utilizadores na página das divisões pelo que a tarefa é considerada simples. Relativamente ao tempo demorado na realização da tarefa, verificou-se que os utilizadores sendo a primeira vez que visualizavam a interface despendiam algum tempo a verificar toda a informação apresentada no ecrã, mesmo depois de terem encontrado o local em que deviam clicar. Este ponto foi considerado positivo, pois apesar de estender as tarefas por mais tempo, significa que os utilizadores estavam interessados no aspecto da interface e no seu conteúdo e não apenas na realização da tarefa.

A tarefa 2 é considerada simples na sua realização, no entanto tinha como objectivo obrigar o utilizador a alterar o modo de navegação da interface por omissão. Ou seja, requeria que o utilizador navegasse através da lista de serviços e não pelas divisões. Na sua maioria os utilizadores compreenderam o conceito, tendo apenas o conhecimento fornecido no contexto do questionário e realizaram a tarefa no número mínimo de cliques ou quase. O tempo despendido na realização da tarefa revelou esta necessidade do utilizador compreender a mudança de apresentação da interface.

A tarefa 3 foi a primeira tarefa mais complicada, tanto no número de cliques como no conceito envolvido na sua realização. Esta tarefa requeria navegação até ao dispositivo cuja propriedade se pedia para destacar e só então aceder às propriedades em destaque, em detrimento de aceder directamente às propriedades. Como este ponto chave estava identificado no contexto do questionário, os utilizadores concluíram a tarefa quase sem erros. O tempo gasto na tarefa pelo grupo de utilizadores destaca-se relativamente ao tempo mínimo pelo facto de que na sua maioria os utilizadores consideravam a tarefa concluída assim que destacavam a propriedade, só se apercebendo após revisão da tarefa que ainda faltava configurar uma regra para a propriedade.

Na tarefa 4 os utilizadores depararam-se com a tarefa mais complexa e por consequência foi aquela em que houve uma maior diferença entre o número de cliques e a média do grupo de utilizadores. Aqui requeria-se ao utilizador a criação de uma Macro – conjunto de acções agrupadas e executadas de uma só vez. Esta é uma tarefa de configuração que o utilizador “programa” na interface apenas uma vez e após esta “programação” o utilizador apenas necessita de executar a Macro, para que todas as acções especificadas na mesma sejam cumpridas. Na realização desta última tarefa os utilizadores necessitaram de aceder às macros para iniciar uma sessão de gravação e navegar pela interface a executar as tarefas que desejavam gravar, da mesma forma como se eles próprios estivessem a realizar as acções. Esta navegação e a forma de parar a gravação foram dos principais factores que contribuíram para a diferença entre o tempo mínimo da tarefa e a média apresentada. A conclusão da tarefa era a execução da Macro criada e este ponto foi realizado por todos os utilizadores sem erros.

Os tempos de execução apresentados devem ainda ter em conta o facto de que as tarefas foram realizadas num dispositivo com o qual a maioria dos utilizadores não tem experiência de utilização. Este facto foi principalmente importante na realização das tarefas 3 e 4, pois incluído no seu processo estava a escrita dos nomes da propriedade destacada e da macro. Deve ter-se também em conta que tanto os tempos como os cliques foram contabilizados sem recorrer ao uso de atalhos que o próprio utilizador pode configurar sobre a forma de favoritos ou propriedades em destaque entre outros.

5.1.3 Classificação da interface pelo utilizadores

O questionário apresenta uma terceira secção para os utilizadores classificarem a interface. Esta secção foi preenchida pelos utilizadores após a conclusão das quatro tarefas e tem como objectivo classificar a interface quando à sua usabilidade, aspecto gráfico, utilidade e desempenho. Cada uma destas características pôde ser classificada pelos utilizadores num intervalo de cinco valores possíveis 'Muito Bom', 'Bom', 'Médio', 'Mau' e 'Péssimo'. A classificação geral atribuída pelo grupo de utilizadores é apresentada no gráfico seguinte:

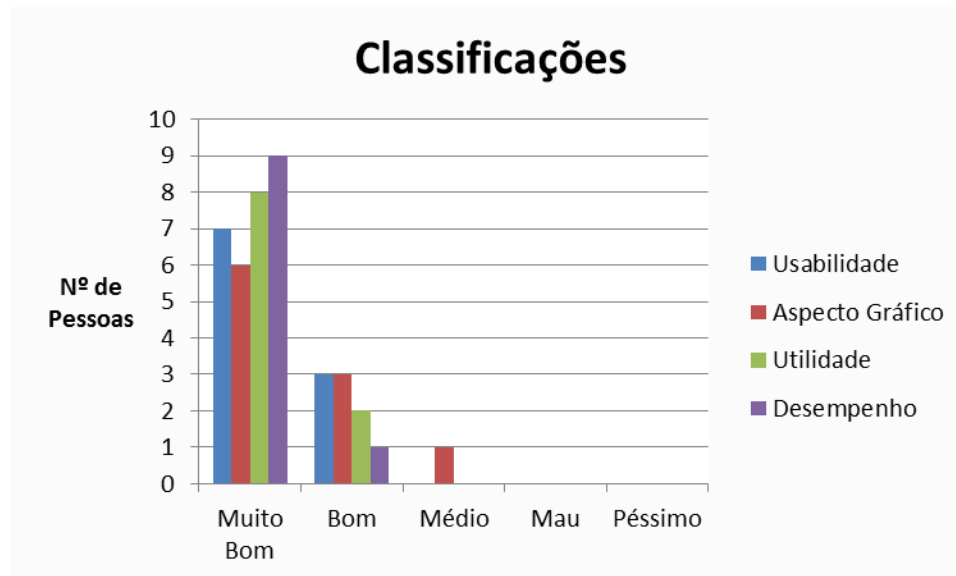


Figura 42 – Representação das classificações atribuídas à interface

Na generalidade os utilizadores classificaram a interface com 'Muito Bom' ou 'Bom', existindo apenas um utilizador que classificou o aspecto gráfico com 'Médio'. Note-se ainda que foi pedido aos utilizadores para considerarem as suas respostas como se eles próprios possuísem um sistema domótico na sua habitação, sendo o mesmo compatível com a interface e permitindo todo o controlo nela apresentado.

Na realização do questionário pelos utilizadores verificou-se ainda um ponto importante que merece algum destaque. Após o término das tarefas e da classificação da aplicação, a maioria dos utilizadores despenderam ainda algum tempo a explorar a interface e as funcionalidades que não haviam sido abrangidas nas tarefas. Este factor permite concluir que enquanto interface, a aplicação não só permite realizar as funcionalidades para as quais foi concebida, mas também apela ao interesse do utilizador para a explorar e customizar.

Analisando os resultados obtidos concluiu-se que, de forma geral, a interface desenvolvida cumpre os objectivos propostos. Sendo classificada pelos utilizadores como intuitiva e fácil de navegar, apesar das diversas e complexas funcionalidades que a compõem. Para além da importância dos testes de utilização e da avaliação positiva dos utilizadores é também importante garantir o bom desempenho da aplicação, pelo que se apresentam de seguida os testes realizados neste âmbito.

5.2 Testes de desempenho

O bom desempenho numa interface reflecte-se na satisfação do utilizador pois uma aplicação lenta e com problemas não captura o seu interesse. Na avaliação desta aplicação teve-se em conta que a interface é acedida no *browser* de um dispositivo móvel que comunica pela Internet com o servidor que a disponibiliza e este por sua vez comunica com o simulador de DomoBus através de TCP/IP, quer o servidor e o simulador estejam no mesmo computador ou não. Dada esta disposição de máquinas e pontos de comunicação considerou-se que existiam três aspectos importantes a avaliar: as mensagens trocadas entre servidor e simulador, o tamanho das páginas enviadas para o dispositivo e o tempo de geração das mesmas pelo *browser*.

5.2.1 Comunicações Servidor / Simulador

As mensagens trocadas entre estes dois componentes servem para consulta (*get*) e alteração (*set*) do estado dos dispositivos no simulador. Em ambos os tipos, cada mensagem pode ser apenas para uma propriedade ou pode conter várias propriedades de vários dispositivos. Esta estratégia de mensagens foi uma implementação realizada para promover um maior desempenho da comunicação e por consequência um menor tempo de espera para o utilizador que dá as instruções, tal como explicado no capítulo da Arquitectura na secção 3.3.

Nesta avaliação realizaram-se diversas trocas de mensagens, de ambos os tipos e com tamanhos variados, determinando-se a seguinte média para cada tipo:

- *Get* – Cabeçalho fixo 4 *bytes*, informação sobre cada propriedade a consultar 4 *bytes*.
- *Set* – Cabeçalho fixo 4 *bytes*, informação sobre cada propriedade a modificar 6 *bytes*.

Apresenta-se de seguida um cenário onde se pretende configurar as seis propriedades da televisão da sala, consultar o estado das luzes de um quarto e fechar o portão da garagem. A primeira interacção é o acesso à televisão onde se vêem, tal como representado na figura 43, as seis propriedades desta e os seus respectivos valores, que são consultados através de uma única mensagem do tipo *get*. Posteriormente o utilizador realiza as suas alterações e é enviada uma única mensagem do tipo *set* com

informação sobre as seis propriedades a modificar. Por fim, para garantir que a operação foi efectuada com sucesso, a aplicação envia uma nova mensagem de consulta das mesmas propriedades.

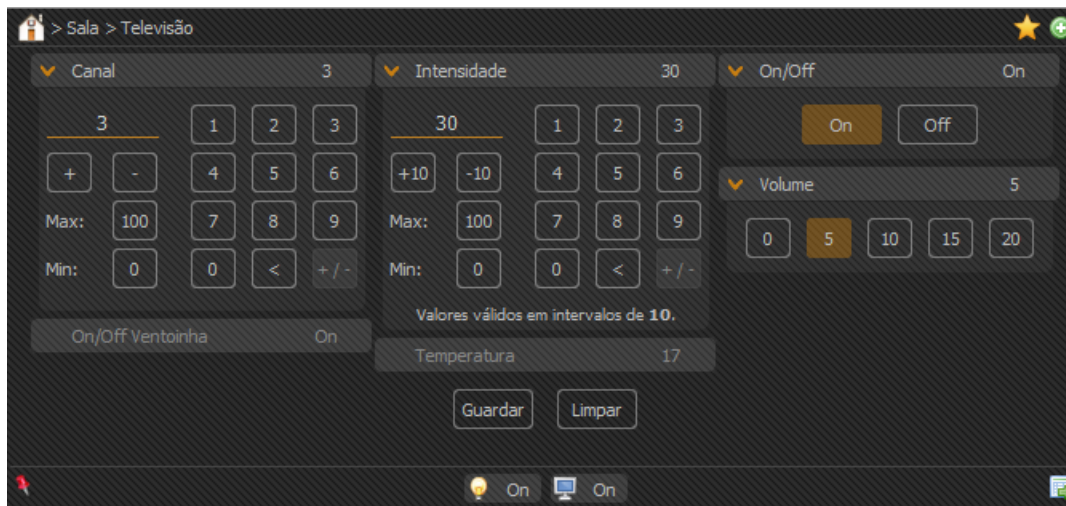


Figura 43 – Lista de propriedades da televisão em ecrã de maiores dimensões

No caso do quarto, basta aceder à divisão onde a propriedade luz é consultada através de um pedido *get* simples. Por fim, para fechar o portão, é necessária uma sequência de mensagens *get*, *set* e *get*, para consultar a propriedade, alterá-la e consultá-la novamente.

No total deste cenário tem-se 7 mensagens trocadas, que na realidade realizaram 19 acções de leitura e escrita. No total de mensagens enviadas, as mensagens de leitura contabilizaram 62 bytes e as alterações 32 bytes. A nível das mensagens de resposta foram transferidos um total de 36 bytes, estas mensagens correspondem aos valores das propriedades em caso de consultas e à identificação de sucesso ou erro no caso das alterações de valores. O total de bytes trocado em todas estas mensagens é praticamente insignificante, pelo que o tempo dispendido neste processo irá depender mais da latência das comunicações do que qualquer factor do próprio sistema.

Note-se que existe ainda forma de reduzir o número de mensagens e por sua vez o número de *bytes* transaccionados, através do uso de macros. As acções de alteração de propriedades de diversos dispositivos, admitindo que são realizadas com frequência e em conjunto, podem usufruir do uso de macros. Deste modo, ao executar-se uma macro que agrupa diversas acções é apenas enviada uma mensagem de modificação, em detrimento de uma mensagem por cada dispositivo e respectivas leituras no acesso e após alteração.

Para teste das operações indicadas o simulador foi instalado num servidor disponibilizado pelo IST, um AMD 2.2Ghz com 2Gb de memória, denominado Sigma. Por sua vez a interface foi colocada numa máquina Intel 2.66Ghz, com 2Gb de memória e o sistema operativo Windows XP. Com estas

configurações os tempos de comunicação entre a interface e o simulador nunca superaram os 0.05 segundos. Considerando a velocidade das redes actuais este valor não tem visibilidade significativa, pelo que se considera apropriado para a aplicação.

Por fim, foram realizados testes de carga com diversos utilizadores a efectuarem constantes alterações a diversos dispositivos. Nem o tempo das comunicações nem o desempenho da interface no servidor sofreram alterações. No caso específico do desempenho do servidor que possui a interface, destaca-se ainda que o processamento dos pedidos originados pelo *browser* do cliente não causaram, para além dos recursos consumidos constantemente pelo Apache, qualquer peso no CPU ou na memória da máquina.

Numa interface remota desenvolvida principalmente para dispositivos móveis, com um processamento menor e uma ligação à Internet mais lenta, é necessário ter em atenção dois aspectos importantes: o tamanho das páginas e o tempo de geração das mesmas nos diversos dispositivos.

5.2.2 Tamanho das páginas

No mundo dos dispositivos móveis com acesso à Internet disponíveis no mercado pode identificar-se uma grande variedade de tipos de conexão e de velocidades. Como o objectivo da interface é ser acedida por qualquer um destes dispositivos, desde que este possua um *browser*, um dos aspectos importantes a considerar é o tamanho das páginas que são apresentadas ao utilizador no seu dispositivo.

Nos gráficos apresentados de seguida pode verificar-se os diversos tamanhos conseguidos na aplicação. Estes dependem da ordem de acesso que se está a realizar e num dos casos depende do browser usado para aceder à aplicação. O browser que se destaca no gráfico, designado por IEM6, é o Internet Explorer Mobile 6. Adicionalmente é utilizado o conceito de *cache*, este conceito no âmbito dos *browsers* refere-se a guardar determinadas informações que se prevêem ser novamente necessárias num futuro / acesso próximo.

Analisando o gráfico da figura 44, verifica-se que o primeiro acesso à aplicação é aquele que comporta um maior número de KBs, nomeadamente uma média de 126KBs. Este acesso corresponde à página de início de sessão e só ocorre caso a aplicação não se encontre em *cache* no *browser*, caso contrário o carregamento é apenas de 2,3KBs. O valor superior do primeiro acesso era já esperado, pois é aqui que se carregam os diversos estilos CSS, os ficheiros *javascript* e as imagens mais comuns da interface. Estes três componentes são usados por toda a aplicação nos restantes acessos e como já se encontram carregados permitem que as páginas seguintes tenham um menor peso.

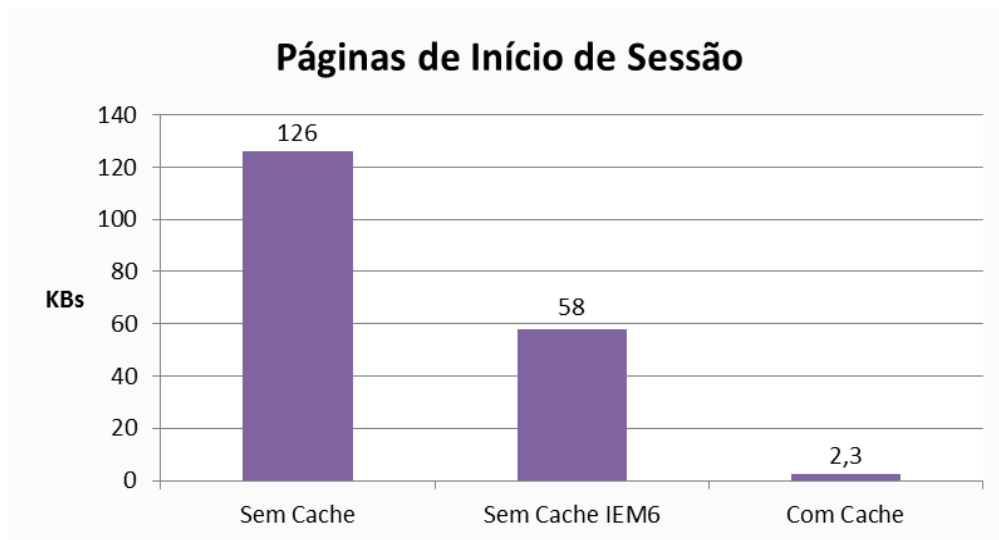


Figura 44 – Tamanho das páginas de início de sessão

Neste primeiro acesso sem cache existe ainda uma dependência do *browser* usado. A diferença entre o Internet Explorer Mobile 6 (IEM6) e qualquer outro *browser*, é que no primeiro não se carrega o componente *jQuery*. Este componente é uma *framework* que permite, entre outros, realizar efeitos de imagens através de *javascript*, tornando a apresentação das páginas da aplicação mais dinâmica e visualmente mais estimulante para o utilizador. No entanto o IEM6 e *browsers* mais antigos não suportam este componente, pelo que a interface identificando esta situação exclui esta *framework* dos dados enviados.

Após o início de sessão da aplicação, o utilizador acede à página inicial da aplicação que corresponde à primeira página de navegação por divisões ou serviços, conforme a opção por omissão. O gráfico da figura 45 ilustra o carregamento desta página e das restantes que correspondem à resposta de pedidos *Ajax*.

A página inicial da interface apresenta 4,6KBs e carrega a moldura apresentada ao utilizador, contendo as barras de topo e fundo da página. Estas barras contêm o menu e o acesso às propriedades em destaque e não voltam a ser carregadas durante a navegação do utilizador, apenas actualizadas por pedidos de *Ajax*. Tal como a actualização das barras, todo o restante conteúdo das páginas por onde o utilizador navega são carregadas através de pedidos *Ajax*, comportando apenas uma média de 2,5KBs.

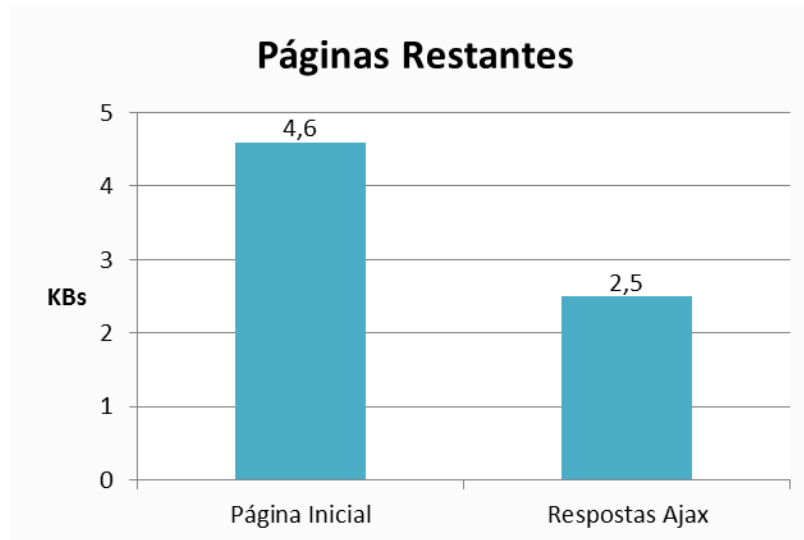


Figura 45 – Tamanho das páginas após início de sessão

O maior tamanho identificado neste teste foi de 126KBs no primeiro acesso sem cache e com excepção deste valor que se encontra na ordem das centenas de KBs, todos os outros acessos estão abaixo dos 5KBs. Os sites do Google¹ e do Sapo² que possuem versões específicas para dispositivos móveis, apresentam valores médios de uma e duas dezenas de KBs; no caso do “Jornal de Negócios”³ que também possui uma versão mobile a média de valores foi sempre superior a 170KBs. Em comparação com estes sites e outros do mesmo tipo, conclui-se que a interface desenvolvida apresenta valores abaixo da média dos outros sites, destacando-se apenas o primeiro acesso sem cache que no entanto não se considera significativo por ocorrer no máximo uma vez em cada sessão de uso da interface.

5.2.3 Tempo de geração de ecrã

O tempo de geração de um ecrã ou como é mais conhecido, tempo de *render*, é o intervalo de tempo desde que os dados de uma página chegam ao *browser* e esta é de facto mostrada ao utilizador com todo o conteúdo carregado. O processamento do dispositivo influencia largamente este tempo, pelo que considerou-se ser um ponto a ser testado em diversos dispositivos. Para comparação foram escolhidos um PC (2.66GHz), um iPhone 3GS (600MHz) e um HTC Touch (200MHz), cobrindo tanto os dispositivos fixos como os dispositivos móveis mais recentes e mais antigos.

Os tempos recolhidos destes três dispositivos apresentam-se no gráfico seguinte.

¹ Google – www.google.com

² Sapo – www.sapo.pt

³ Jornal de Negócios – www.jornaldenegocios.pt

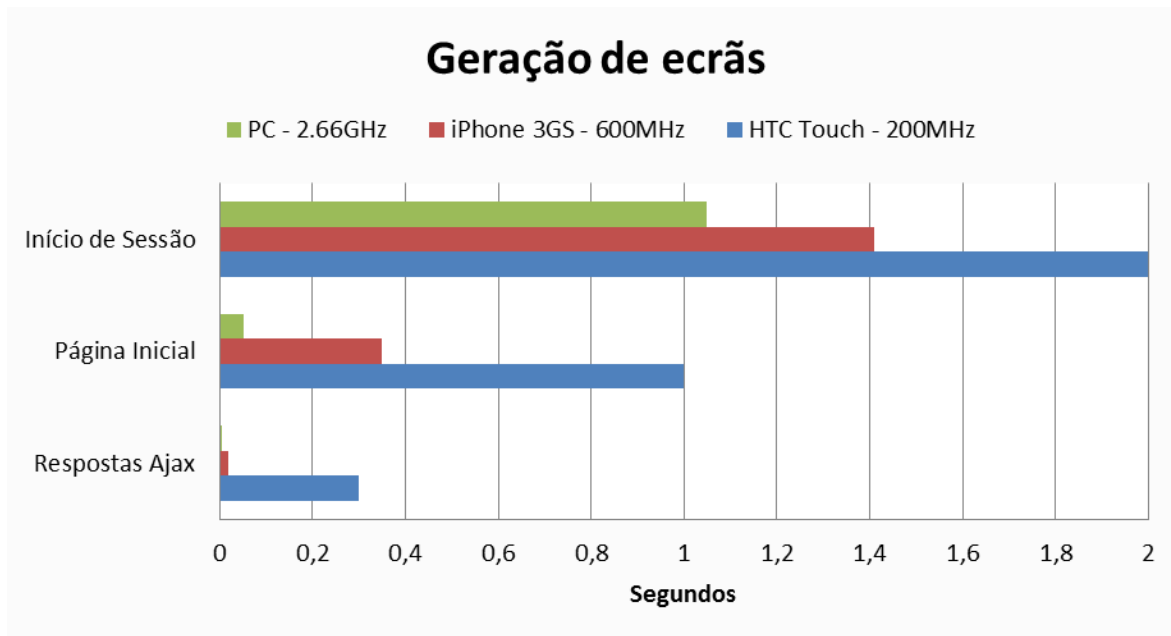


Figura 46 – Tempo despendido na geração de ecrãs

No gráfico pode verificar-se novamente uma divisão pelo início de sessão, a página inicial e as restantes páginas resultantes de pedidos *Ajax*. No início de sessão, tal como identificado na secção anterior, são recebidas as imagens mais usadas nos restantes ecrãs da aplicação e é o *render* de todas estas imagens que torna a geração destes ecrãs o mais lento dos três grupos apresentados. Note-se que apesar deste *render* as imagens não são apresentadas ao utilizador.

A geração da página inicial inclui o cabeçalho, a construção da moldura e o posicionamento de todos estes elementos na página. As restantes páginas geradas a partir de pedidos *Ajax*, apenas necessitam carregar um conjunto limitado de dados e gerá-los nas posições já computadas pela página inicial. Desta forma, a página inicial demora mais que as restantes uma vez que já construiu a arquitectura de ecrã também usada pelas páginas seguintes.

Como era esperado, verificou-se que quanto maior é o processamento do dispositivo mais rápida é a geração da página. De destacar ainda, que no servidor cada um destes pedidos tem um processamento médio de 0.13 segundos. Neste tempo o servidor interpreta os pedidos recebidos da interface, consulta a base de dados e o simulador DomoBus se necessário e constrói a resposta que finalmente envia para o *browser*.

Os testes de performance demonstram que embora a primeira página seja a mais pesada, as páginas restantes, devido ao uso de *Ajax*, apresentam valores bastante baixos tanto no tamanho como no tempo de geração. Considerando todos estes aspectos conclui-se que a interface é fácil de usar e promove uma boa interactividade com os utilizadores.

Capítulo 6

Conclusão

A evolução tecnológica dos últimos anos tem abrangido diversos sectores além dos sistemas informáticos. A domótica tem sido uma das áreas que beneficia com esta evolução, à medida que os dispositivos domésticos se tornam mais sofisticados e passíveis de controlo remoto e computadorizado. Na área dos dispositivos móveis assistiu-se também a diversas melhorias desde aparelhos que, apesar de mais compactos, possuem uma capacidade de processamento superior aos seus antecessores, à maior autonomia dos dispositivos, que permite ao seu utilizador usufruir de novas funcionalidades com melhor qualidade e durante mais tempo.

A acompanhar estas evoluções verificou-se também uma gradual redução de custo e aumento de velocidade da Internet, o que levou à sua massificação pelo público em geral, de tal forma que hoje em dia liga pessoas e dispositivos independentemente da distância que os separa. Estas três áreas em conjunto levantam a atractiva possibilidade de uma habitação equipada com uma solução domótica, poder ser controlada remotamente através de um telemóvel, PDA ou qualquer dispositivo fixo ou móvel com acesso à Internet. Hoje em dia, podemos encontrar diversas soluções domóticas disponíveis no mercado, no entanto na sua generalidade cada uma apresenta uma tecnologia própria, fechada e compatível apenas com os seus produtos e dispositivos. O sistema domótico DomoBus é um projecto académico e está a ser desenvolvido para colmatar esta limitação das soluções existentes, implementando um único sistema domótico que através do uso de *gateways* se torna compatível com dispositivos de diversas tecnologias.

Da reunião de todas as condições descritas acima, surgiu o objectivo principal deste projecto, desenvolver uma interface remota para o sistema domótico DomoBus. Este objectivo pode dividir-se em muitos outros que necessitam de ser cumpridos para que se possa atingir um resultado positivo no projecto. Uma vez que a aplicação a desenvolver é uma interface domótica e é através desta que os utilizadores interagem com o sistema, pretende-se que o resultado seja algo simples, agradável e intuitivo, mas ao mesmo tempo robusto e completo, que nos permita controlar todas as funcionalidades dos dispositivos.

No controlo de todos os dispositivos e propriedades disponíveis ao utilizador, foram determinadas certas funcionalidades específicas que a Interface deveria cumprir. A funcionalidade principal era a possibilidade de visualização e manipulação das propriedades dos dispositivos existentes na habitação,

em qualquer momento e a partir de qualquer local. Em adição, a interface deve também permitir um conjunto de configurações específicas a cada utilizador / habitação como definição do tipo de navegação por omissão, criação de favoritos, macros, destaque de propriedades e validação de permissões de utilizador. Por fim, e como o utilizador pode escolher qualquer dispositivo com ligação à Internet para aceder à interface, esta tem por objectivo ser adaptável ao ecrã em que está a ser visualizada, fornecendo um maior conforto a quem a manipula.

A interacção da aplicação com o sistema domótico externo foi testada sobre um simulador do DomoBus, que foi desenvolvido também no âmbito deste projecto, por forma a permitir um ambiente estável e simulado. Este simulador teve por objectivo ser o mais realista possível, pelo que é acedido por ligação TCP/IP tal como seria o DomoBus e apenas recebe as mesmas mensagens de obter e alterar propriedades.

Os objectivos descritos foram todos enquadrados no âmbito deste projecto, pelo que se conclui que o mesmo foi implementado com sucesso. Um utilizador, com um dispositivo móvel com acesso à Internet, pode aceder à interface e monitorizar o estado dos diversos dispositivos apresentados, tal como manipular o seu estado directamente ou por execução de macros. Foi também possível testar esta aplicação em diversas localizações, desde a própria habitação, o local de emprego ou enquanto se espera que uma peça de teatro se inicie.

6.1 Trabalho Futuro

Embora os objectivos definidos tenham sido atingidos com sucesso, existem diversos pontos que poderiam ser explorados mais profundamente e os quais seria interessante expandir no futuro. De seguida apresenta-se uma lista desses pontos:

- Uma vez que se trata de uma interface para o sistema DomoBus e, pelas razões já enunciadas, foi necessário recorrer a um simulador. Um dos pontos mais interessantes seria realmente efectuar a ligação entre a interface e o DomoBus e controlar uma habitação real. Para este ponto ser cumprido, da parte da interface é apenas necessário alterar o módulo de comunicações de forma a adaptar a actual estrutura das mensagens enviadas e recebidas para o formato específico do DomoBus [50].
- Uma expansão interessante seria, em adição às macros, a possibilidade de construir cenários (*triggers*). Um cenário corresponde à definição de uma condição temporal (data e/ou hora) ou de uma mudança de estado de outro dispositivo, que desencadeia uma determinada acção. Por exemplo, quando o sensor de iluminação detecta que é noite o sistema procede ao fecho das janelas. Para tal se concretizar seria necessário que a aplicação permitisse criar estes cenários

e enviá-los ao DomoBus. Este envio é necessário porque a interface Web, por não se encontrar sempre em execução, não possui a capacidade de análise permanente necessária para desencadear as acções correspondentes.

- Possibilidade de adicionar e remover dispositivos, tal como editar as diversas propriedades de cada um. Actualmente, a única solução é exportar um novo ficheiro XML com o estado mais actual da habitação e importá-lo na aplicação, de forma a interface proceder à sua actualização.
- Para além de poder adicionar e remover dispositivos, teria interesse a adição e remoção de utilizadores, tal como a edição das suas permissões. No caso das permissões, um utilizador poderia alterar o nível de acesso de qualquer outro com um nível inferior ao seu, para um valor igual ou inferior àquele que ele próprio detém.
- Uma das funcionalidades mais requeridas no acesso remoto a um sistema domótico é a possibilidade de visualizar em tempo real as imagens das diversas câmaras de vigilância da casa. Esta funcionalidade permite ao utilizador uma maior confiança no sistema de segurança e sempre que exista alguma situação inesperada possa aceder às imagens actuais da habitação. Este ponto não foi contemplado neste projecto, pois seria necessário expandir também o próprio sistema domótico para facultar esta possibilidade.

Referências

- [1] Tecnologia X10, URL: <http://www.x10.com>
- [2] Tecnologia LonWorks, URL: <http://www.echelon.com>
- [3] Tecnologia KNX, URL: <http://www.knx.org>
- [4] Tecnologia CEBus, URL: <http://www.cebuse.org>
- [5] Lisa Montgomery. 10 Key Features in a Home Automation System, 2009. Electronic House.
URL: http://www.electronichouse.com/article/10_features_to_look_for_in_a_home_automation_system/
- [6] Renato Nunes. DomoBus – A New Approach to Home Automation, 2003. 8CIEEE – 8th Internacional Congress on Electrical Engineering, Portugal.
URL: <http://www.domobus.net/papers/03-CIEEE03.pdf>
- [7] Renato Nunes. Decentralized Supervision for Home Automation, 2006. MELECON 2006 – The 13th IEEE Mediterranean Electrotechnical Conference, Espanha.
URL: <http://www.domobus.net/papers/06-MELE06.pdf>
- [8] Renato Nunes. Modelo de Especificação e Programação de um Sistema Domótico, 2004. IADIS Conferência Ibero-Americana WWW/Internet 2004, Espanha.
URL: <http://www.domobus.net/papers/04-CIAWI04.pdf>
- [9] Wi-Fi – Rede local sem fios, URL: <http://www.wi-fi.org/?mobile=no>
- [10] GPRS – General Packet Radio Services, URL: <http://www.3gpp.org/>
- [11] UMTS - Universal Mobile Telecommunications System, URL: <http://www.umtsworld.com/>
- [12] TCP/IP, URL: <http://www.w3schools.com/tcpip/default.asp>
- [13] André Bergholz. Extending Your Markup: XML Tutorial. 2000. Stanford University.
URL: <http://xml.coverpages.org/BergholzTutorial.pdf>
- [14] Renato Nunes. Especificação XML de um Sistema DomoBus, 2006. Instituto Superior Técnico.
URL: http://www.domobus.net/papers/DomoBus_XML_Spec_v16.pdf
- [15] ActiveHomePro, URL: www.activehomepro.eu
- [16] INSTEON, URL: <http://www.smarthome.com/1411/INSTEON-Compatible-mControl-Software/p.aspx>
- [17] PowerHome 2, URL: <http://www.smarthome.com/1440/PowerHome2-INSTEON-Compatible-Home-Automation-Software/p.aspx>
- [18] David Fox - Home Monitor, URL: <http://www.oreillynet.com/pub/a/wireless/2001/07/23/monitor.html>
- [19] Java 2 Micro Edition, URL: <http://www.oracle.com/technetwork/java/javame/overview/index.html>

- [20] iHome, URL: <http://www.smarthome.com.au/homeautomation/iphone/>
- [21] Telemóvel iPhone, URL: <http://www.apple.com/iphone/>
- [22] iPhone iOS SDK – Plataforma de desenvolvimento, URL: <http://developer.apple.com>
- [23] Web Browser, URL: http://www.webopedia.com/quick_ref/internet_browsers.asp
- [24] Jesse James Garret. Ajax: New Approach to Web Applications, 2005.
URL: <http://www.adaptivepath.com/ideas/essays/archives/000385.php>
- [25] HTTP, URL: <http://www.w3.org/Protocols/HTTP/AsImplemented.html>
- [26] HTC Phones, URL: <http://www.htc.com/pt/>
- [27] Internet Explorer Browser, URL: <http://www.microsoft.com/windows/internet-explorer/default.aspx>
- [28] Mozilla Firefox Browser, URL: <http://www.mozilla.com/en-US/firefox/personal.html>
- [29] Safari Browser, URL: <http://www.apple.com/safari/>
- [30] Web Server, URL: <http://www.w3.org/TR/webarch>
- [31] PHP Language, URL: <http://www.php.net/>
- [32] ASP Language, URL: <http://www.asp.net/>
- [33] Java Language, URL: <http://www.java.com/en/>
- [34] Internet, URL: <http://www.internet.com/>
- [35] Apache Program, URL: <http://www.apache.org/>
- [36] Sistema Operativo Microsoft Windows, URL: <http://www.microsoft.com/windows/>
- [37] Sistema Operativo Linux, URL: <http://www.linux.com/>
- [38] Sistema Operativo Mac OS, URL: <http://www.apple.com/macosx/>
- [39] HTML, URL: <http://www.w3.org/>
- [40] Javascript, URL: <https://developer.mozilla.org/en/JavaScript>
- [41] Cascading Style Sheets – CSS, URL: <http://www.w3.org/Style/CSS/>
- [42] MySQL Database, URL: <http://www.mysql.com/>
- [43] WAMP, URL: <http://www.wampserver.com/en/>
- [44] Grady Booch, Ivar Jacobson, James Rumbaugh. The Unified Modeling Language User Guide, 1998. Addison-Wesley.
- [45] Andrew S. Tanenbaum. Computer Networks Internet Socket, 2003. Prentice Hall.

[46] Playstation Portable – PSP, URL: <http://www.psp.com>

[47] MD5, URL: http://www.w3.org/TR/1998/REC-DSig-label/MD5-1_0

[48] Steve McConnell. Code Complete, 2004. Microsoft Press

[49] jQuery, URL: <http://jquery.org/>

[50] Renato Nunes. Comunicação no sistema DomoBus - Application Programming Interface, 2005. Instituto Superior Técnico.

URL: http://www.domobus.net/papers/DomoBus_Comm-API_v12.pdf

Anexos

Anexo A – Formato do documento XML

```
<?xml version="1.0" encoding="UTF-8"?>
<HouseConf>
  <House ID="1" Name="Lisbon House">
    <FloorList>
      <Floor ID="1" Name="P01" HeightOrder="-1">
        <DivisionList>
          <Division ID="1" Name="Garagem" AccessLevel="0" Image="garage.png"/>
          ...
        </DivisionList>
      </Floor>
      ...
    </FloorList>
  </House>

  <ScalarValueTypeList>
    <ScalarValueType ID="1" Name="Porcentagem (0-100)" MinValue="0" MaxValue="100" Step="10"/>
    ...
  </ScalarValueTypeList>

  <EnumValueTypeList>
    <EnumValueType ID="1" Name="On-Off">
      <Enumerated Name="On" Value="1"/>
      <Enumerated Name="Off" Value="0"/>
    </EnumValueType>
    ...
  </EnumValueTypeList>

  <DeviceTypeList>
    <DeviceType ID="1" Name="Lâmpada-Regulada">
      <PropertyTypeList>
        <PropertyType ID="1" Name="On-Off" AccessMode="RW" ValueType="ENUM" RefValueType="1"/>
        <PropertyType ID="2" Name="Intensidade" AccessMode="RW" ValueType="SCALAR" RefValueType="1"/>
      </PropertyTypeList>
    </DeviceType>
    ...
  </DeviceTypeList>

  <ServiceList>
    <Service ID="1" Name="Climatização"/>
    ...
  </ServiceList>
```

```

<DeviceList>
  <Device ID="1" RefDeviceType="1" Name="Luz" RefDivision="1" AccessLevel="3,5,6">
    <DeviceServiceList>
      <DeviceService RefService="2"/>
      ...
    </DeviceServiceList>
  </Device>
  ...
</DeviceList>

<UserList>
  <User ID="1" Name="admin" Password="xx" AccessLevel="10" ImportAccess="1"/>
  ...
</UserList>

<OptionList>
  <Option ID="1" RefUser="1" NavType="0" SkinID="default" LanuageID="pt"/>
  ...
</OptionList>

<MacroList>
  <Macro ID="6" RefUser="2" Name="Completa" CreationData="1284322495" >
    <Action ID="29" RefDevice="5" RefProperty="1" Value="0" />
    ...
  </Macro>
  ...
</MacroList>

<BookmarkList>
  <Bookmark ID="1" RefUser="2" RefDevice="0" RefDivision="4" RefService="0" CreationData="1283623763" />
  ...
</BookmarkList>

<ImpPropertyList>
  <ImpProperty ID="11" RefUser="1" RefDevice="3" RefProperty="1" Name="Quarto" Order="1" Image="light.png"
CreationData="1286019835" />
  ...
</ImpPropertyList>
</HouseConf>

```

Anexo B - Casos de Teste

Contexto

A aplicação apresentada de seguida é uma interface para um sistema domótico, permitindo controlar os diversos dispositivos distribuídos por uma habitação.

O ecrã da aplicação divide-se em três secções principais:

- No topo, encontra-se o acesso à página inicial, o endereço actual, os Favoritos e mais opções (+).
- Na secção principal podem ser visualizados, entre outros: as divisões, dispositivos (ex: Portão) e as propriedades (ex: Estado Aberto / Fechado).
- No fundo encontram-se as Propriedades em Destaque e o acesso às Macros.

As Macros são um conjunto de acções executadas de uma só vez. Para criar uma macro necessita de iniciar um período de gravação e realizar todas as acções que deseja incluir na mesma.

As Propriedades em Destaque são propriedades de dispositivos que se encontram sempre visíveis e podem ter associadas uma regra. Sempre que esta regra não se verifica a propriedade é assinalada com fundo vermelho. Para destacar propriedades é necessário navegar até ao dispositivo e aceder às propriedades em destaque.

Existem dois tipos de listagem: um por divisão (omissão) e outro por serviço. Os serviços agrupam os diversos dispositivos por funcionalidade.

Tarefas

1º Mudar o canal da Televisão da Sala para o canal 6.

2º Visualizar todas as luzes da Casa.

3º Colocar a luz do Hall nas propriedades em destaque com o nome "AB", propriedade "On-Off" e com a regra de manter-se "Off".

4º Criar Macro que liga as luzes da Garagem e do Hall com o nome "GH" e executá-la.

Classificação

Após executar as tarefas anteriores classifique de Muito Bom a Péssimo as seguintes áreas:

Usabilidade

☐ Muito Bom ☐ Bom ☐ Médio ☐ Mau ☐ Péssimo

Aspecto Gráfico

☐ Muito Bom ☐ Bom ☐ Médio ☐ Mau ☐ Péssimo

Utilidade (admitindo que possui um sistema domótico)

☐ Muito Bom ☐ Bom ☐ Médio ☐ Mau ☐ Péssimo

Desempenho

☐ Muito Bom ☐ Bom ☐ Médio ☐ Mau ☐ Péssimo