

Universidade da Beira Interior

Departamento de Informática



**Departamento de
Informática**

**Nº 194 - 2020: *CasaIn – aplicação para
acompanhamento de comportamentos domésticos***

Elaborado por:

Gonçalo Delgadinho

Orientador:

Professor Doutor Pedro Araújo

17 de maio de 2021

Agradecimentos

Gostaria de expressar os meus sinceros agradecimentos ao Sr.Professor Doutor Pedro Araújo pela oportunidade de desenvolver este projeto e disponibilidade e incentivo para uma boa causa.

Gostaria de agradecer à minha família, namorada e colega de casa por todo o apoio e incentivo nesta fase complicada.Tudo isto foi essencial no desenrolar do projeto tanto para fazer críticas como também para dar novas ideias para uma melhor aplicação.

Finalmente quero agradecer por me ter sido concedido um projeto que pode ajudar tanto no meu caso mas como em tantas outras famílias para uma vida melhor e saudável.

Conteúdo

Conteúdo	iii
Lista de Figuras	v
Lista de Tabelas	vii
1 Introdução	1
1.1 Enquadramento	1
1.2 Motivação	1
1.3 Objetivos	1
1.4 Organização do Documento	3
2 Estado da Arte	5
2.1 Introdução	5
2.2 Sistema de monitorização para pessoas desabilitadas ou envelhecidas	5
2.3 Privacidade/ Rejeição	6
2.4 Sistemas existentes	7
2.5 Comparação com o sistema implementado no projeto	7
2.6 Conclusão	9
3 Engenharia de Software	11
3.1 Introdução	11
3.2 Requisitos	11
3.3 Casos de Uso	13
3.4 Lista de Funcionalidades	14
3.5 Manual de Utilização	16
3.6 Conclusões	17
4 Tecnologias e Ferramentas Utilizadas	19
4.1 Introdução	19
4.2 <i>Android Studio</i>	19
4.3 <i>Arduino</i>	20
4.4 Alternativas para desenvolvimento	20

4.5	<i>VSCode com a extensão PlatformIO</i>	22
4.6	<i>Firebase</i>	22
4.7	Sensores utilizados	23
4.8	Preço estimado dos equipamentos	28
4.9	Conclusões	28
5	Implementação e Testes	29
5.1	Introdução	29
5.2	Preparação do Ambiente de Trabalho	29
5.3	Arquitetura da aplicação	29
5.4	Comunicação entre <i>Firebase</i> e <i>NodeMcu</i>	31
5.5	<i>Realtime Database</i>	32
5.6	Simulação da instalação de sensores	34
5.7	<i>Gauges</i>	35
5.8	Testes	36
5.8.1	Teste dos vários dispositivos	36
5.9	Situações típicas de utilização	40
5.10	Conclusão	41
6	Conclusões e Trabalho Futuro	43
6.1	Conclusões Principais	43
6.2	Trabalho Futuro	43
A	Funções e Excertos de Código	45
B	Anexos	50
	Bibliografia	57

Lista de Figuras

3.1	Caso de uso da aplicação cliente	14
4.1	<i>Sonoff T0EU1C</i>	23
4.2	<i>Sonoff TH10</i>	24
4.3	DHT11	24
4.4	ACS712	25
4.5	NodeMcu ESP8266	25
4.6	<i>Relay</i>	25
4.7	PIR	26
4.8	YF-S201C	26
4.9	WeMos D1 Mini WiFi	27
4.10	<i>Arduino</i>	27
4.11	<i>Foto da mesa de testes</i>	28
5.1	Diagrama do funcionamento da aplicação.	30
5.2	Diagrama de interligações entre dispositivos e sua função.	31
5.3	Regras presentes na base de dados	32
5.4	Estrutura da base de dados em tempo real.	32
5.5	Planta modelo de uma situação real.	34
5.6	Legenda da planta acima apresentada.	35
5.7	Captura de ecrã da aplicação onde mostra os <i>Gauges</i> em tempo real.	35
5.8	Fotografia real que tem presente o módulo YF-S201C.	36
5.9	Fotografia real tirada que tem presente os módulos descritos acima.	37
5.10	Fotografia real que contém o módulo do sensor de movimento e um <i>NodeMcu</i>	37
5.11	Fotografia real que contém <i>Sonoff T0EU1C</i> e uma lâmpada.	38
5.12	Fotografia real que contém <i>Sonoff TH10</i> e ao lado o módulo DHT11.	38
5.13	Captura de ecrã da aplicação onde mostra um exemplo de um fragmento da aplicação.	39

Listas de Tabelas

3.1 Requisitos Funcionais da aplicação.	12
3.2 Requisitos Não Funcionais da aplicação.	13

Listas de Excertos de Código

A.1 Função ConsumoEletricidade.ino	45
--	----

Acrónimos

ADC Águas da Covilhã

EDP Energias de Portugal

IDE Integrated Development Environment

NTP Network Time Protocol

BaaS Backend-as-a-Service

URL Uniform Resource Locator

Capítulo

1

Introdução

1.1 Enquadramento

Este projeto realizou-se no âmbito da unidade curricular Projeto. O mesmo foi desenvolvido no contexto da área de automação e monitoramento de sensores instalados numa casa, tendo em conta uma integração de uma aplicação móvel.

1.2 Motivação

A motivação presente no projeto deveu-se ao enorme gosto por automação, de casas inteligentes e toda a tecnologia utilizada nos mesmos. Estes contribuem para um estilo de vida mais facilitado tendo controlo praticamente de qualquer dispositivo. Com isto é possível tornar uma simples lâmpada normal só com controlo pelo interruptor numa atividade que possa ser controlada pelo telemóvel, o que é magnífico.

1.3 Objetivos

Este projeto tem três principais objetivos sendo estes uma investigação na área de monitoramento de atividades no caso dos idosos, uma aplicação móvel e por fim um manual de utilização.

O projeto deve conter os seguintes requisitos:

- Desenvolvimento de uma aplicação móvel;
- Base de dados(Firebase);

A investigação deve incluir também produtos existentes no mercado em termos de *hardware* e *software*.

O projeto consiste numa aplicação Android que transmite informação para a Firebase e que por sua vez comunica com o equipamento.

O projeto foi dividido em 4 tarefas principais: análise da aplicação a desenvolver, desenvolvimento da aplicação, Testes e ajustes e Elaboração do relatório. A investigação demorou cerca de um mês e meio, o desenvolvimento do protótipo cerca de dois meses e o relatório foi escrito ao longo do projeto (figura 1.1).

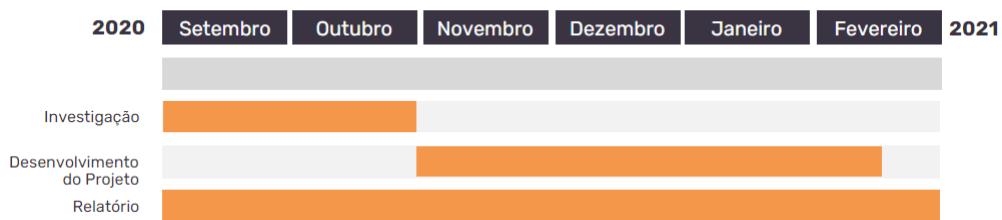


Figura 1.1: Cronograma e Tarefas Realizadas

1.4 Organização do Documento

De modo a refletir o trabalho que foi feito, este documento encontra-se estruturado da seguinte forma:

1. O primeiro capítulo – **Introdução** – apresenta o projeto, a motivação para a sua escolha, enquadramento , objetivos, cronograma e a respetiva organização do documento.
2. O segundo capítulo – **Estado de Arte** – descreve os conceitos mais importantes no âmbito deste projeto, bem como as tecnologias utilizadas durante do desenvolvimento da aplicação.
3. O terceiro capítulo – **Engenharia de Software** – descreve os requisitos funcionais e não funcionais acordados, bem como, os casos de uso da aplicação.
4. O quarto capítulo – **Tecnologias e Ferramentas Utilizadas** – apresenta as tecnologias utilizadas durante do desenvolvimento da aplicação.
5. O quinto capítulo – **Implementação e Testes** – descreve a implementação das tecnologias e ferramentas escolhidas, incluindo também testes.
6. O sexto capítulo – **Conclusões e Trabalho Futuro** – apresenta as principais conclusões recolhidas com o desenrolar do projeto, bem como perspetiva de um possível trabalho futuro.

Capítulo

2

Estado da Arte

2.1 Introdução

Este capítulo explica resumidamente o estado da arte do tema deste Projeto, apresentando assim trabalhos relacionados com o tema em interesse. Para uma melhor compreensão do ambiente que iremos trabalhar, serão revistos artigos desde os benefícios a formas de melhorar o dia-a-dia de pessoas idosas. O mesmo está dividido em 2 secções: secção 2.2 onde explica o sistema de monitorização das tarefas diárias ao longo do dia para pessoas incapacitadas ou envelhecidas. Na secção 2.3, é mostrada uma lista de resultados que foram encontrados e analisados de acordo com o tema abordado.

2.2 Sistema de monitorização para pessoas desabilitadas ou envelhecidas

O estudo de [1], demonstrou que a monitorização de pessoas idosas nos seus lares é feita através de uma rede de sensores, onde é enviada a informação através da linha de telefone para um centro de atendimento. Tanto a monitorização doméstica como o sistema de casas inteligentes são apostas que os serviços de saúde e investigadores estão a adotar para casos deste tipo. No caso das casas inteligentes suportam monitorização de saúde, segurança e actividades do dia a dia.

Existem vários tipos de sensores tais como sensores de infravermelhos, sensores utilizados no corpo/roupa, sensores de monitorização de vídeo, sensores de pressão e por reconhecimento de voz. Para cada um destes sensores cabe uma tarefa como por exemplo a deteção da atividade diária, a deteção

de quedas e alterações no estado de saúde.

Outro fator importante é a aceitação ou a funcionalidade dos dispositivos por utilizador. Ou seja, a dificuldade no uso do dispositivo e sua funcionalidade, o que causa pouca satisfação. A tecnologia mais aceite pelos pacientes é aquela que integra a ajuda de equipas especializadas como é o caso da monitorização de rede doméstica.

Através dos estudos feitos, a tecnologia de monitorização melhora a qualidade de vida do paciente, aumentando a sua confidência e fornece segurança [2], [3], [4]. Contudo através do estudo feito por [5], [4] indica-nos que por um lado os pacientes sentiam-se mais confiantes e estavam convencidos que iria ajudar quando necessitassem. Por outro lado alguns tinham bastante receio em praticar as suas atividades do dia a dia pelo que causava uma grande ansiedade.

A melhoria da qualidade de resposta/atendimento face às condições gerais leva a um aumento da esperança média de vida. Por exemplo, na Europa, existem cerca de 16% de pessoas com sessenta e cinco ou mais anos. Esta população necessita de mais acompanhamento afim de poder antecipar algo que aconteça sem estar um profissional 24h de vigilância. Com isto não significa que sejam apenas os idosos a receber assistência mas também para crianças/adultos que necessitem de cuidados especiais.

Um dos métodos utilizados para situações de emergência são os sensores de posição. Normalmente estão instalados no quarto, na sala, nos corredores, cozinha, etc. A informação é recebida através da deteção de calor do corpo humano enquanto as pessoas se movimentam dentro da habitação. O modelo é feito através do dia-a-dia da pessoa, conforme os seus hábitos, o tempo que passam em cada divisão e as tarefas ao longo do dia. Ou seja, cada caso é um caso e a monitorização tem de ser feita especificamente para o problema em questão de modo a visualizar anomalias presentes.

Para demonstrar o impacto das tecnologias presentes numa casa inteligente teremos de encontrar mais evidências para compreender que tecnologias podemos oferecer ao serviço de saúde como também aos pacientes. Para isso irá ser necessário encontrar mais oportunidades entre empresas para trazer mais qualidade e eficiência para o mercado.

2.3 Privacidade/ Rejeição

De acordo com [6], os dados estatísticos mostram um aumento de população envelhecida, cerca de 20% da população europeia irá atingir a idade de 65 anos no ano de 2025. A duração média de vida aumentou, mas por ou-

tro lado o aparecimento de doenças nestas idades também aumenta consideravelmente. Através da análise feita , podemos demonstrar que as razões mais importantes para a rejeição dos dispositivos são o medo de perder os aparelhos, a falta de consciência da sua necessidade que estão diretamente ligadas com o medo de perder a sua privacidade. Ao mesmo tempo, o grau de aceitação foi influenciado pelo tipo de cuidados prestados aos pacientes e também pelo seu grau económico. Por outro lado, outra grande diferença foi pelo género do paciente, ou seja, as mulheres geralmente sentem-se menos à vontade com a aplicação dos dispositivos no seu lar.

2.4 Sistemas existentes

De acordo com [2] existem várias tecnologias para assistência médica, mas dentro do leque de possibilidades, realçam-se 5 principais. Entre elas, sensores de infravermelhos, sensores para medições de pulsação, por exemplo, monitorização de vídeo , sensores de tensão arterial e reconhecimento de som. No caso de aplicação prática das categorias acima descritas, podem ser utilizadas em várias situações, focando-se principalmente em três objetivos nomeadamente a deteção de atividades diárias, deteção de quedas e finalmente a deteção de alterações no estado de saúde. Todas estas tecnologias são utilizadas mais frequentemente no ramo hospitalar, instituições de apoio ao domicílio e instituições de apoio a longo prazo.

2.5 Comparação com o sistema implementado no projeto

Como foi abordado acima, existem múltiplas formas de implementar a tecnologia introduzida no sistema em questão. Neste caso foi projetado para ambientes onde as pessoas se encontram mais debilitadas e necessitam de uma monitorização mais atenta e detalhada. A aplicação CasaIn foi implementada numa versão diferente para o acompanhamento de pessoas idosas. Neste caso temos vários sensores instalados dentro de casa para monitorização das atividades diárias. São utilizados vários equipamentos como por exemplo sensor de movimento para detetar a passagem do quarto para a sala, por exemplo, ter acesso ás luzes de casa(quarto) neste caso específico. Existe também a possibilidade de monitorizar o consumo de água, eletricidade, medir a temperatura e humidade de uma certa divisão. Para complementar existe a possibilidade de a qualquer altura do mês ver o consumo médio de água e luz como também o preço estimado conforme a zona territorial,

devido aos preços divergirem conforme o concelho. Por fim para as pessoas que por qualquer motivo se esqueçam de algum aparelho ligado, um aquecedor por exemplo, através da aplicação móvel recebe um aviso caso algum valor esteja fora do comum e facilmente pode ser desligado á distância.

Estatísticas

De acordo com [6], temos um gráfico que representa o grau de aceitação dos dispositivos nas habitações onde temos uma escala de 0-100% onde comparmos homens, mulheres e ambos.

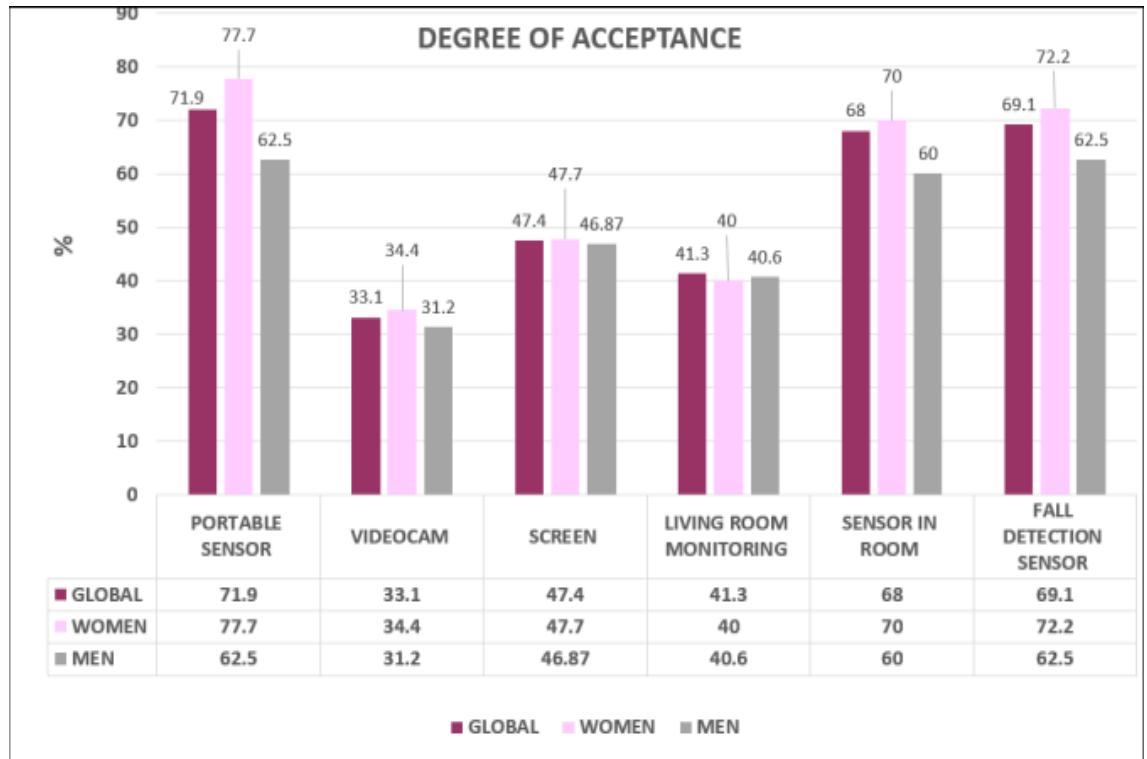


Figura 1 - Grau de aceitação dos dispositivos

2.6 Conclusão

Neste capítulo foram apresentadas as ideias mais importantes no ponto de vista da temática. Foram apresentados sistemas de hardware existentes no mercado como também a questão da privacidade e rejeição na aplicação de dispositivos nas habitações. Tudo isto vai ser importante para o processo de desenvolvimento da aplicação. Podemos afirmar com alguma certeza que os equipamentos influenciam na vida das pessoas mais debilitadas, podendo assim desfrutarem de uma vida melhor e mais saudável.

Capítulo

3

Engenharia de Software

3.1 Introdução

Neste capítulo irão ser apresentados os requisitos funcionais e não funcionais para a aplicação cliente no capítulo 3.2. Logo de seguida são apresentados os diagramas de casos de uso(capítulo 3.3).

3.2 Requisitos

Relativamente ao requisitos presentes na aplicação, funcionais e não funcionais, irão ser apresentadas nas tabelas seguintes, 3.1 e 3.2.

Os requisitos funcionais representam o que um *Software* faz, em relação a tarefas e serviços. Por outro lado, os requisitos não funcionais são os requisitos relacionados com o desempenho, usabilidade, segurança e manutenção das tecnologias presentes.

Requisitos Funcionais		
RF	Requisito	Descrição
1	Interface	Interface dentro de cada atividade onde mostra a informação armazenada na base de dados(Firebase), com recurso a sensores.
2	Base de Dados	Base de dados onde é guardada a informação de cada sensor em tempo real.
3	Home	Do topo para baixo: logo abaixo da mensagem inicial de saudação está presente o logótipo da aplicação CasaIn. A barra de menu será uma barra vertical, onde o utilizador , ao selecionar um dos menus pretendidos, será redirecionado para a página correspondente. Esta barra terá 4 menus , ordenados da seguinte forma: Home, Quarto, Sala, WC(ver RF3-RF6 (incluindo o menu Home)). Este menu será o mais suscetível a mudanças de acordo com novas tarefas, ou qualquer informação que seja necessária acrescentar.
4	Quarto	Logo abaixo do menu Home(RF3), estará o menu Quarto que dá acesso a funcionalidades tais como ligar/desligar luzes, ver o histórico de passagens numa certa divisão bem como verificar a temperatura e humidade no dado momento.
5	Sala	No menu está presente a Sala que contém informações acerca do consumo atual/total como também um preço estimado. No topo da página encontra-se a informação relativa à temperatura e humidade da Sala e por último no final da página existem 2 botões que controlam a medição e permitem ao utilizador o corte de energia de um equipamento.
6	WC	Neste menu está presente a monitorização do consumo/fluxo de água como também um preço estimado que permite ao utilizador fazer uma análise rápida.

Tabela 3.1: Requisitos Funcionais da aplicação.

Requisitos Não Funcionais		
RNF	Requisito	Descrição
1	Extensão	Possibilidade de evolução onde são acrescentadas novas funcionalidades.
2	Modificabilidade	Possibilidade de fácil alteração/modificação quando seja necessário.
3	Escalabilidade	Possibilidade de aumentar as funcionalidades em grande escala.
4	Versão a Aplicação Móvel	A aplicação móvel irá estar disponível em dispositivos <i>Android</i> .
5	Dispositivos <i>Android</i>	Dispositivos <i>Android</i> com versão 5 ou superior.
6	Tempo de Resposta e Desempenho	O tempo de resposta deverá ser o mais reduzido possível, enquanto o desempenho de aplicação deverá ser ótimo para que o utilizador possa ter um <i>feedback</i> do que esteja a acontecer no preciso momento.
7	Fiabilidade	A aplicação deverá manter os dados atualizados todos os dias do ano(24h/dia).
8	Relevância de dados	Os dados deverão ser atualizados em tempo real(o atraso não deverá ser superior a 30 segundos).

Tabela 3.2: Requisitos Não Funcionais da aplicação.

3.3 Casos de Uso

Através dos requisitos funcionais da aplicação, foram criados os seguintes diagramas de caso de uso. Na figura 3.1 podemos observar o caso de uso para a aplicação cliente.

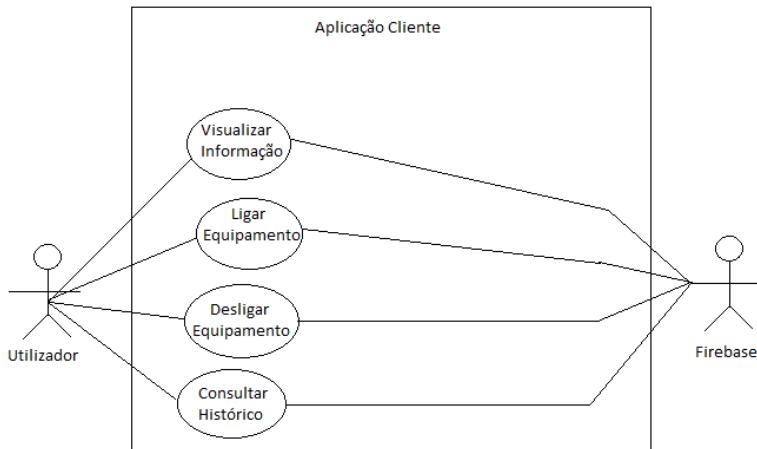


Figura 3.1: Caso de uso da aplicação cliente

3.4 Lista de Funcionalidades

A aplicação destinada para os utilizadores inicia-se com uma janela de boas vindas, onde começa por saudar os utilizadores de acordo com a hora que está a ser utilizada. Se o utilizador pretender selecionar a divisão pretendida basta então clicar no botão do menu (\equiv) e estarão disponíveis várias opções, entre elas, quarto, sala, WC. Dentro de cada fragmento, o utilizador tem várias opções de interação de acordo com a divisão escolhida.

Cada divisão terá as seguintes funcionalidades:

Quarto

- Histórico de passagens - Informação acerca da data e hora de passagens pelo sensor;
- Luz Quarto - Acende ou desliga a luz e está interligado como o interruptor, ou seja quando acende o botão ativa;
- Luz Corredor - Acende ou desliga a luz e está interligado como o interruptor, ou seja quando acende o botão ativa;
- Luz Candeeiro - Acende ou desliga a luz e está interligado como o interruptor, ou seja quando acende o botão ativa;
- Gauges - Manômetro utilizado para visualizar a temperatura e humidade no quarto.

Sala

- Histórico de passagens - Informação acerca da data e hora de passagens pelo sensor;
- *Gauge* Temperatura - Permite ao utilizador visualizar a temperatura em tempo real;
- *Gauge* Humidade - Permite ao utilizador visualizar a humidade em tempo real;
- Botão *ON* - Ativa a leitura do sensor de corrente e a energia através de um *relay*;
- Botão *OFF* - Desativa a leitura do sensor de corrente e a energia através de um *relay*;
- Consumo atual em *Watt* - Apresenta a leitura do sensor de corrente em *Watt*;
- Consumo total em *Watt* - Apresenta a soma de energia consumida pelo utilizador no equipamento em utilização;
- Preço estimado: Preço estimado da eletricidade de acordo com o tarifário da Energias de Portugal (EDP) com a potência contratada de 4.6.

WC

- *Gauge* do consumo de água - Manómetro que representa os litros de água consumidos;
- *Gauge* do fluxo de água - Manómetro que representa o fluxo de água em mililitros por segundo;
- Consumo atual em Litros - Representam consumo atual de água em litros;
- Preço estimado - Preço estimado do gasto de água de acordo com o tarifário disponibilizado pelas Águas da Covilhã (ADC).

3.5 Manual de Utilização

Através da aplicação móvel CasaIn é possível fazer várias opções, entre elas:

Home:

- Se o utilizador pretender visualizar os avisos basta abrir a aplicação e aparecerá na página principal os avisos referentes aos vários sensores da casa.

Quarto:

- Se o utilizador pretender visualizar o histórico de passagens tem de ir ao menu(≡) e selecionar o Quarto e de seguida clicar no botão referente ao histórico.
- Caso pretenda ligar/desligar a luz basta clicar no botão à frente de cada divisão.
- Para visualizar a temperatura e humidade o utilizador tem de aceder ao menu do Quarto onde pode visualizar os valores dos mesmos.

Sala:

- Se o utilizador pretender visualizar os consumos do equipamento ligado ao sensor basta voltar ao menu no ícone(≡) e selecionar a Sala, onde tem a possibilidade de visualizar os consumos bem como o preço estimado.
- Para o corte de energia através da aplicação é necessário clicar no botão *ON/OFF* dentro dessa divisão de acordo com o pretendido.
- Para verificar a temperatura da Sala tem disponível a informação referente à divisão.

WC:

- Se o utilizador pretender visualizar o consumo total e o fluxo de água é necessário aceder ao menu através do ícone(≡) e selecionar WC.
- Se o utilizador pretender visualizar o preço da água, encontra-se uma tabela com o valor estimado.

3.6 Conclusões

Este capítulo incide nos casos de uso da aplicação, no seus requisitos e, por último, o manual de utilização para o utilizador. Através destes, foi possível uma implementação das suas funcionalidades na íntegra como também o uso de algumas ferramentas.

Capítulo

4

Tecnologias e Ferramentas Utilizadas

4.1 Introdução

Neste capítulo são apresentadas as ferramentas utilizadas na realização do projeto, como por exemplo, *Arduino*, *VSCode* com integração da extensão *PlatformIO*, *AndroidStudio* e *Firebase*, como também foi necessário adicionar extensões e bibliotecas para o desenvolvimento do projeto.

4.2 *Android Studio*

O [7] é um Integrated Development Environment (IDE) utilizado para desenvolver aplicações para a plataforma *Android*. Este *Software* é baseado no [8], que tem funcionalidades tais como auto-completar o código, através da análise do mesmo, ajudas em caso de erros, mostrando várias opções, indentação mais facilitada, simples navegação no código, entre outros.

Em relação às características do *Android Studio*, temos suporte para compilações em Gradle, integração dos componentes mais utilizados em aplicações, um editor de *layout* que tem múltiplas opções que permite ao utilizador arrastar e pré-visualizar diferentes componentes e por fim Integração da [9] que nos permite fazer guardar informação numa base de dados.

4.3 *Arduino*

O [10] é uma plataforma de desenvolvimento de protótipos de hardware livre, ou seja, equipamentos que têm software de código livre, que podem ser executados, modificado e distribuído sem qualquer restrição. Um dos objetivos da criação deste programa foi tornar possível a qualquer pessoa criar conteúdo sem que seja necessário investir enormes quantias de dinheiro, visto que alguns controladores mais complexos não estão ao alcance de todos. Para termos uma ideia do funcionamento dos equipamentos e do resultado que transmitem, existem ferramentas para mostrar todo o tipo de resultados, como por exemplo, valores, gráficos, entre outros. As principais vantagens de utilizar esta plataforma são, entre elas, as bibliotecas disponíveis, a fácil programação e uma comunidade enorme face ao hardware existente. Quanto às desvantagens existem poucos recursos de termos de *DEBUG*.

4.4 Alternativas para desenvolvimento

Quanto à plataforma de desenvolvimento existem várias onde se destacam, por exemplo, *Sloeber*, *embedXcode* utilizado para computadores com *Mac OS X* e *Ktechlab*. A razão pela escolha do Arduino IDE foi simplesmente por já ter conhecimento prévio e pela facilidade de programação.

Bibliotecas Utilizadas

- NTPClient - Obtém o tempo(data/hora) de um servidor Network Time Protocol (NTP) e mantém sempre sincronizado.
- *EasyNTPClient* - Biblioteca utilizada para ler o tempo através de um servidor NTP e apresenta-o em formato UNIX.
- *DHT sensor library* - Biblioteca do Arduino para DHT11, DHT22, etc. Utilizado para sensores de temperatura e humidade.
- *Firebase ESP8266 Client* - Esta biblioteca permite-nos fazer operações tais como ler, escrever, modificar, eliminar, recuperar, guardar informação na *Firebase* no tempo real.
- *SoftwareSerial* - Permite que haja comunicação em serial noutras pinos digitais, utilizando o software para replicar a funcionalidade do Arduino.

- ACS712 - É utilizada para fazer medições da corrente no dispositivo ao qual está conectado.
- Arduino - Biblioteca nativa da plataforma.

4.5 VSCode com a extensão PlatformIO

O [11] é um editor de código desenvolvido pela Microsoft para Sistemas Operativos tais como *Windows*, *Linux* e *macOS*. Uma das funcionalidades mais importantes é precisamente o *IntelliSense*, o *Debugging* e o *Built-in Git*.

O *IntelliSense* é uma funcionalidade que permite auto-completar funções, realçar a sintaxe das mesmas e também a informação rápida enquanto desenvolvemos o código. Quanto ao *Debugging* é uma ferramenta que nos permite rapidamente identificar o erro, compilar e facilita no processo de edição do mesmo.

A extensão utilizada neste projeto foi como referida anteriormente, *PlatformIO*, que tem a possibilidade de integrar funções da biblioteca *Arduino* para o desenvolvimento de sistemas baseados em micro-controladores. Esta plataforma tem vantagens como por exemplo, para funções mais complexas permite-nos ter mais rapidamente acesso ao erro, caso haja, do que a plataforma *Arduino*, uma vez que o editor automaticamente deteta erros existentes no código. No ambiente de desenvolvimento do *Arduino* apenas após a sua compilação temos acesso aos erros existentes, o que leva a que seja uma desvantagem enorme em termos de desenvolvimento.

4.6 Firebase

Firebase é uma plataforma Backend-as-a-Service (BaaS) que foi desenvolvida Pela *Google* para a criação de aplicações *mobile* e aplicações *web*. As funcionalidades mais importantes para os utilizadores são entre elas, funcionamento em vários dispositivos, a sua infraestrutura, mecanismos de autenticação, rápida implementação, escalável, segurança, espaço de armazenamento, entre outras.

Para ter acesso a funcionalidades mais específicas os serviços podem ser pagos mas por outro lado têm funcionalidades que podem realmente interessar no ramo empresarial, p.e..

4.7 Sensores utilizados

Em relação aos sensores utilizados para a realização do projeto, que consistem em aparelhos que são capazes de monitorizar algumas das tarefas realizadas numa casa. Mais abaixo estarão representados todos os componentes necessários para simular essa mesma experiência. Os *DataSheet A* dos seguintes componentes foram retirados do seguinte *website* [12] e estão disponíveis em anexo.

Componentes utilizados

- *Sonoff T0EU1C* - Interruptor inteligente com integração de controlo através da aplicação;



Figura 4.1: *Sonoff T0EU1C*

Principais Características:

- Controlar dispositivos remotamente através de um *smartphone*;
 - Botões de toque para controlar a operação conveniente;
 - Sincronizar o estado do dispositivo em tempo real com o *smartphone*.
 - Preço: 16.20€
- *Sonoff TH10* - Permite monitorizar a temperatura e humidade;

Principais Características:

- Suporta 3 sensores de temperatura e humidade (AM2301, DS18B20, DHT11);
- Suporta temperatura e humidade predefinidas para ligar/desligar;
- Suporta monitorização de temperatura e humidade em tempo real.
- Preço: 18.99€



Figura 4.2: *Sonoff TH10*



Figura 4.3: *DHT11*

- DHT11 - Sensor de temperatura e umidade ligado diretamente ao TH10;

Principais Características:

- Grande capacidade anti-interferência e durável;
 - Faixa de medição de temperatura: 0 - 50 °C;
 - Alcance de transmissão do sinal: 20m.
 - Preço: 3.00€

- Sensor Corrente ACS712 - Sensor de corrente que permite efetuar medições da corrente.

Principais Características:

- Faixa de medição de corrente: 5A;
 - Tensão de alimentação do módulo: 5V.
 - Preço: 8.82€

- *NodeMcu ESP8266*- Módulo WiFi que permitiu ligar vários equipamentos;

Principais Características:



Figura 4.4: ACS712



Figura 4.5: NodeMcu ESP8266

- *Open-source*;
 - Arquitetura RISC de 32 bits;
 - Programável via USB ou WiFi(OTA).
 - Preço: 8.55€
- *Relay* - Interruptor electromecânico que permite desligar dispositivos;

Figura 4.6: *Relay*

Principais Características:

- Indicador de energia;
- Corte de energia ON/OFF.
- Preço: 1.25€

- Sensor de Movimento PIR - Sensor infravermelho passivo que permite monitorizar passagens próximas do mesmo;



Figura 4.7: PIR

Principais Características:

- Realiza a deteção através de sistema infravermelho;
 - Deteta o movimento numa área de até 7 metros;
 - Ajuste do tempo e sensibilidade.
 - Preço: 2.40€
- Sensor do fluxo de água YF-S201C - Sensor utilizado para calcular o fluxo de água baseado em fórmulas matemáticas;



Figura 4.8: YF-S201C

Principais Características:

- Teste da água;
- Sistema de refrigeração de água (Caudalímetro).
- Preço: 12.00€

- *WeMos D1 Mini WiFi* - Módulo WiFi utilizado para ligar equipamentos;



Figura 4.9: WeMos D1 Mini WiFi

Principais Características:

- Compatível com *MicroPython*;
- Antena embutida.
- Preço: 6.00€
- Arduino - Placa de micro-controlador utilizada para testes.



Figura 4.10: Arduino

Principais Características:

- O uso do controlador de microprocessador de alta velocidade (ATmega328);
- Memória Flash: 32 KB dos quais 2KB usados pelo bootloader.
- Preço: 20.00€

Através destes equipamentos foi possível simular uma situação real e verificar em tempo real na aplicação. De seguida deixo uma fotografia da mesa de testes onde é possível ver os equipamentos.

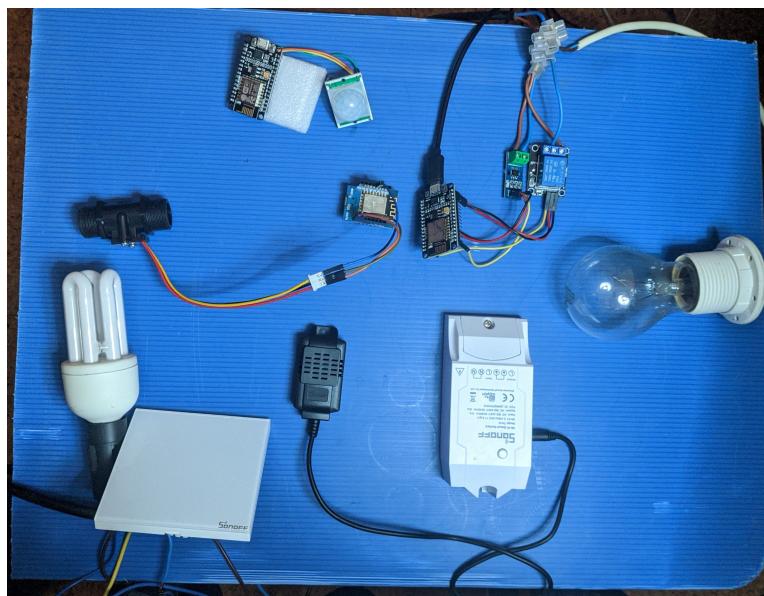


Figura 4.11: Foto da mesa de testes

4.8 Preço estimado dos equipamentos

Depois de feita a análise e compra dos equipamentos, neste caso prático, o preço total aproximado seria de 100€. Este preço inclui todos os equipamentos utilizados para desenvolver este projeto. Cajo seja necessário adicionar mais funcionalidades, o valor irá sofrer alterações.

4.9 Conclusões

Neste capítulo foram apresentadas todas a ferramentas e tecnologias estudadas e implementadas no projeto para obter o melhor desempenho possível. Através do que foi mostrado acima, são apresentadas as razões que levaram à utilização das plataformas mencionadas.

Capítulo

5

Implementação e Testes

5.1 Introdução

Este capítulo mostra a preparação do ambiente de trabalho necessário para a realizar do projeto e posteriores testes efetuados. Ao longo do capitulo vai ser explicado com mais detalhe a implementação da Firebase, como funciona de um modo geral a aplicação e alguns dos testes efetuados na aplicação.

5.2 Preparação do Ambiente de Trabalho

Para a realização do projeto será necessário implementar as tecnologias e ferramentas discutidas no Capítulo 4. O Sistemas Operativo utilizado para a realização de todos os passos necessários foi o *Windows 10*. Para uma melhor compreensão foram necessárias as seguintes etapas:

1. Instalação da plataforma Arduino bem como o editor de código *VSCode* com a extensão *PlatformIO* e por fim o *AndroidStudio* para a implementação do código a ser desenvolvido para cada equipamento utilizado;
2. Configuração das regras na *Firebase* e seleção da *Realtime Firebase* para visualização e armazenamento dos valores obtidos.

5.3 Arquitetura da aplicação

Em relação à arquitetura utilizada na aplicação, podemos observar na figura 5.1 que se trata de todo o processo necessário no processo de desenvolvimento. De acordo com este plano, temos um mapa de como todos os compo-

nentes do projeto irão ser organizados e interligados. Através deste processo de definição de uma solução estruturada poderá ter impacto a nível de qualidade, desempenho e facilidade de manutenção.

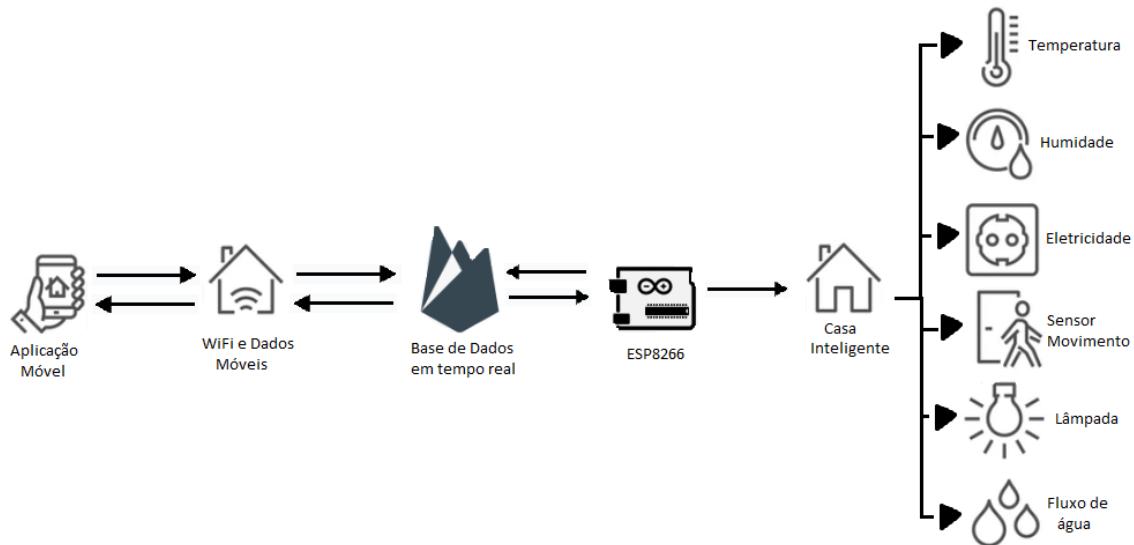


Figura 5.1: Diagrama do funcionamento da aplicação.

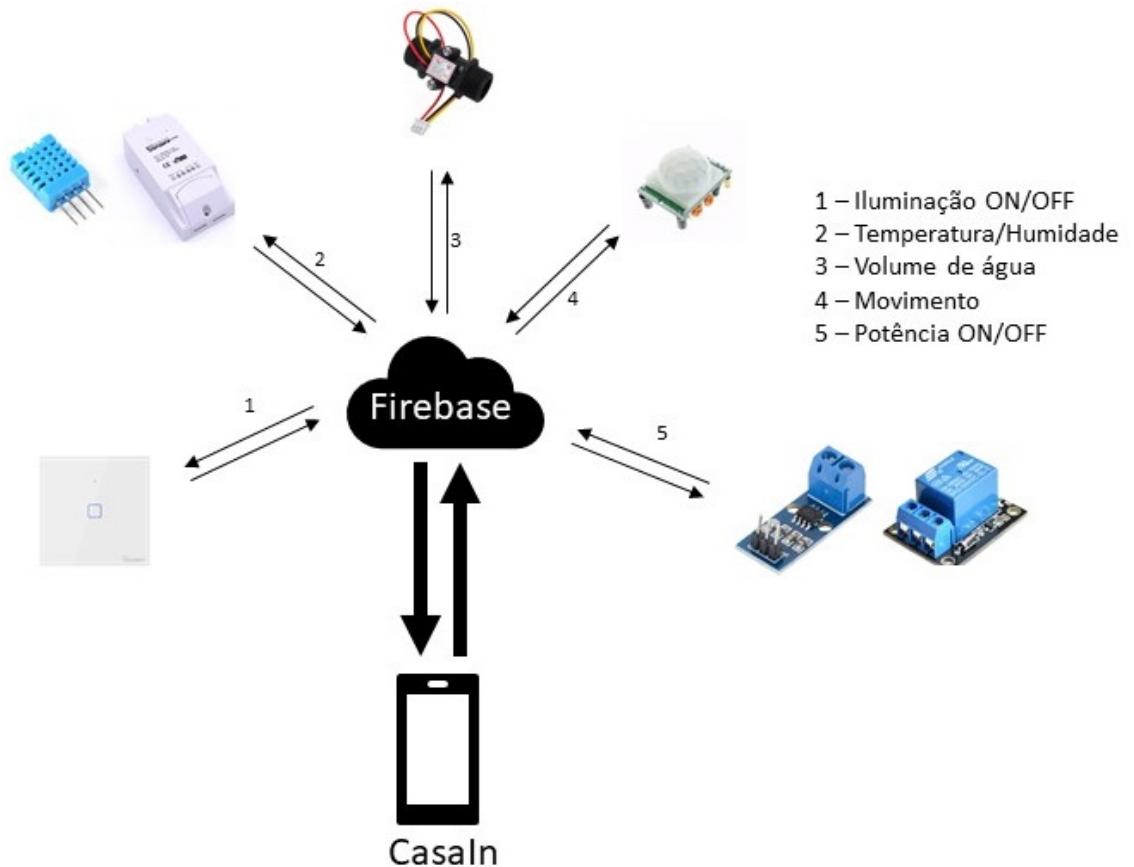


Figura 5.2: Diagrama de interligações entre dispositivos e sua função.

5.4 Comunicação entre *Firebase* e *NodeMcu*

Depois de ser criado o projeto na *Firebase*, as respetivas regras, é necessário configurar o dispositivo(*NodeMcu* com as bibliotecas necessárias para que haja uma ligação entre ambos. Após esse processo, é necessário colocar dentro do trecho de código a chave secreta da base de dados, o Uniform Resource Locator (URL), o nome da rede WiFi e por fim a palavra passe da mesma. Desta forma o equipamento pode conectar-se sem qualquer problema à rede.

5.5 *Realtime Database*

A base de dados em tempo real é um sistema hospedado numa nuvem que sincroniza os dados com os dispositivos em tempo real. Os dados são armazenados como JSON e sincronizados em tempo real com todos os clientes conectados. No projeto foi utilizada a base de dados em tempo real porque a informação está em constante mudança. As vantagens deste sistema são, entre elas, a consulta da informação em qualquer dispositivo, deteção de eventuais problemas e a informação está constatadamente a ser atualizada em tempo real.

As regras utilizadas na base de dados estão representadas na figura 5.2 e a estrutura de dados na figura 5.3.

```
{
  "rules": {
    ".read": true,
    ".write": true
  }
}
```

Figura 5.3: Regras presentes na base de dados

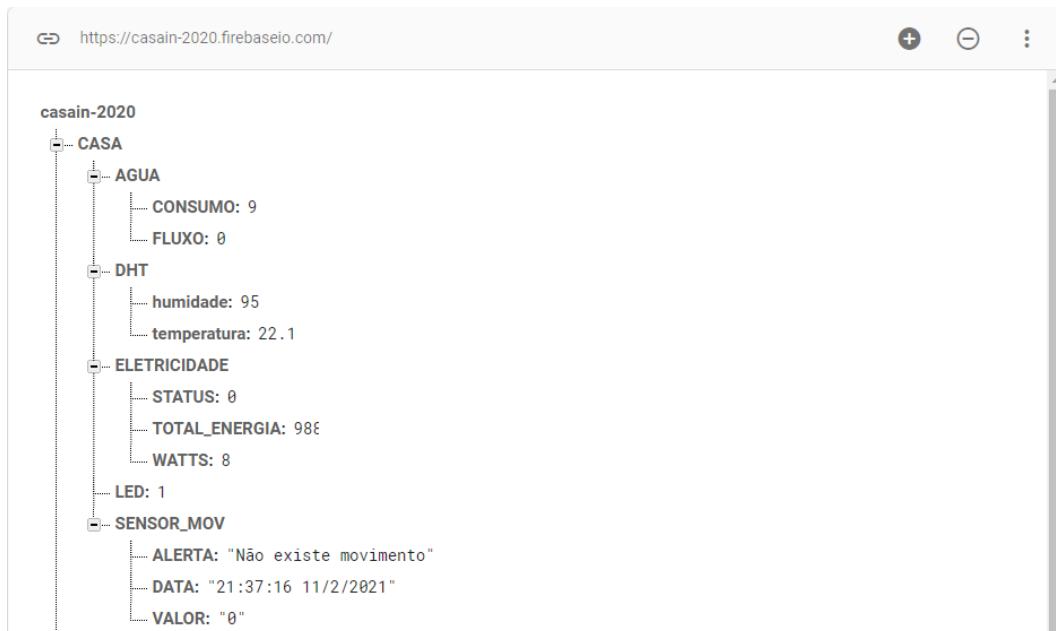


Figura 5.4: Estrutura da base de dados em tempo real.

Descrição da estrutura da base de dados

A estrutura de dados está dividida em várias secções, de acordo com os sensores presentes na habitação.

Água

- Consumo - Representa o consumo medido em L;
- Fluxo - Representa o fluxo de água medido em mililitros por segundo.

DHT

- Humidade - Representa a percentagem de humidade no momento atual;
- Temperatura - Representa o valor da temperatura no momento atual.

ELETRICIDADE

- Status - 1 representa o estado ligado e 0 o estado desligado;
- TotalEnergia - Total de Energia consumida;
- Watts - Consumo atual de energia.

Água

- Alerta - Informa se existe ou não movimento;
- Data - Data e hora da passagem pelo sensor;
- Valor - Quando o valor é 1 acende uma luz verde na aplicação, caso contrário luz vermelha.

Led

- Estado da lâmpada, 1 representa o estado ligado e 0 o estado desligado.

5.6 Simulação da instalação de sensores

Para uma melhor compreensão de uma situação real foi desenhada uma planta de uma casa modelo para mostrar um exemplo de posicionamento dos sensores pelas várias divisões da casa.

Na imagem abaixo podemos ver um exemplo do mesmo onde foi pensado a localização dos sensores para obter a posição ótima e de modo a obter os melhores resultados no caso de monitorização.

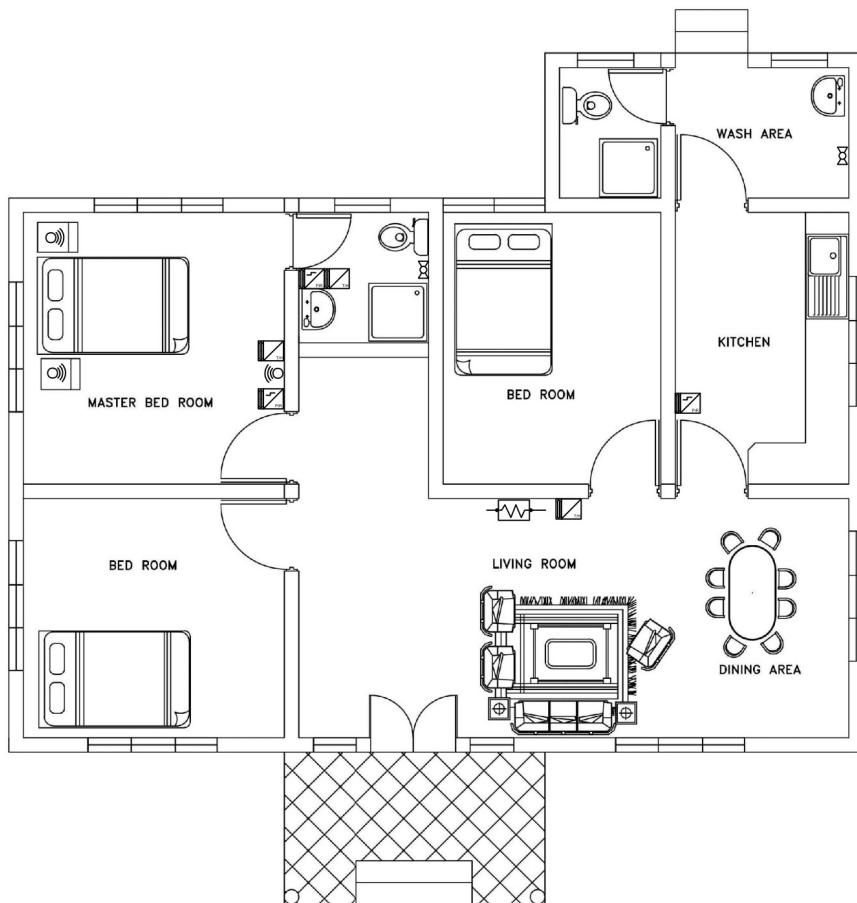


Figura 5.5: Planta modelo de uma situação real.

	Sensor de movimento
	Sensor de Temperatura e Humidade
	Interruptor com conexão WiFi 2.4Ghz
	Radiador elétrico
	Sensor de fluxo de água

Figura 5.6: Legenda da planta acima apresentada.

5.7 *Gauges*

Os manómetros(*gauges*) são utilizados para mostrar a informação em tempo real, ou seja, qualquer alteração na base de dados em tempo real irá alterar automaticamente ser mostrada nos manómetros. Neste caso os *gauges* são dinâmicos. Através destes o utilizador consegue visualizar os dados de forma rápida e ter saber exatamente o ambiente em que se depara. Os valores são recebidos através dos dispositivos que estão constantemente a transmitir informação para a *Firebase* e posteriormente mostrados na aplicação.

Como se pode observar na figura 5.6, temos um exemplo de um manómetro medindo a temperatura e humidade em tempo real.

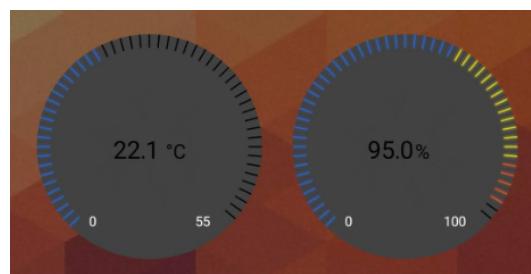


Figura 5.7: Captura de ecrã da aplicação onde mostra os *Gauges* em tempo real.

5.8 Testes

5.8.1 Teste dos vários dispositivos

1^a Fase - Teste do consumo de água:

- Através do equipamento YF-S201C interligado com o *Wemos D1 Mini WiFi* foi possível obter a leitura do fluxo de água. Para isso foi utilizado uma bicha para chuveiro enrolada diretamente no módulo YF-S201C e com a água a correr foi possível calcular o fluxo de água bem como os litros de água que por lá passavam.

Na fotografia 5.7 podemos observar como foi feita a ligação:



Figura 5.8: Fotografia real que tem presente o módulo YF-S201C.

2^a Fase - Teste do consumo da eletricidade

- Nesta fase foram necessários vários equipamentos tais como o *Relay*, o módulo de sensor de corrente(ACS712) e por fim o módulo *NodeMcu ESP8266*. Através destes equipamentos foi possível testar o consumo de energia, ligar e desligar o equipamento através do *Relay* e o equipamento utilizado para o teste foi uma lâmpada.

Conforme mostra a fotografia 5.8 abaixo podemos ver como foi tudo interligado.

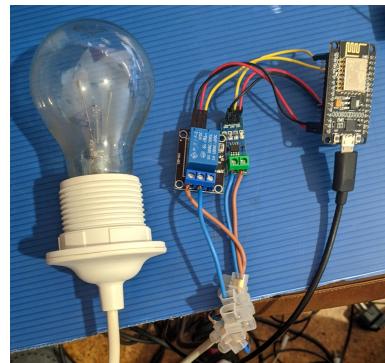


Figura 5.9: Fotografia real tirada que tem presente os módulos descritos acima.

3^a Fase - Teste do sensor de movimento

- Para testar o movimento das pessoas numa certa divisão foi colocado um módulo de sensor de movimento(PIR) ligado diretamente a um *NodeMcu* e depois de feitas as configurações manualmente foi possível ter uma leitura da hora e data que detetou um movimento.

A ligação entre ambos os dispositivos encontra-se na figura 5.9.



Figura 5.10: Fotografia real que contém o módulo do sensor de movimento e um *NodeMcu*.

4^a Fase - Testar o interruptor inteligente

- Para a realização do teste do interruptor foi necessário guardar o estado do dispositivo de forma a aplicação conseguir ler o valor da base da dados e a estar sempre atualizada conforme o estado da lâmpada, ligado/desligado.

A montagem encontra-se na fotografia 5.10 abaixo representada.

5^a Fase - Testar o Módulo de temperatura e humidade



Figura 5.11: Fotografia real que contém *Sonoff T0EU1C* e uma lâmpada.

- Neste caso foi necessário o interruptor sem fios que monitoriza remotamente a temperatura e umidade da divisão onde está colocado. Através do módulo DHT11 aplicado no mesmo foi possível ter acesso à temperatura e umidade.

Na imagem 5.11 abaixo encontra-se os módulos utilizados.



Figura 5.12: Fotografia real que contém *Sonoff TH10* e ao lado o módulo DHT11.

Durante a realização deste projeto foram feitos vários testes aos equipamentos para obter o melhor desempenho e para retirar as medições mais precisas no ambiente testado. Primeiramente foi feito o teste individual a cada equipamento e a cada caso em particular na aplicação Arduino apenas para mostrar no monitor de série os valores que iam sendo obtidos a partir dos aparelhos. Depois de visualizar o resultado de cada aparelho e confirmar a veracidade dos valores passaria à próxima etapa. De seguida o objetivo foi interligar o aparelho de desenvolvimento NodeMcu à base de dados(*Firebase*) para verificar se estava a receber os valores em tempo real e se estariam em constante mudança. Após todos estes passos foi necessário criar uma interface para poder visualizar todos estes valores e para uma melhor leitura dos mesmos. Com isto o próximo passo foi conseguir o melhor *layout* para ter uma leitura rápida e para ter controlo sobre certos dispositivos.

Para uma análise detalhada do código acerca do consumo de energia, encontra-se na apêndice A uma das funções utilizadas.

Um exemplo de um layout implementado na aplicação encontra-se na figura 5.12 presente abaixo.



Figura 5.13: Captura de ecrã da aplicação onde mostra um exemplo de um fragmento da aplicação.

5.9 Situações típicas de utilização

Conforme os testes feitos na aplicação, foi necessário implementar funções adicionais para o caso do utilizador não poder interagir e também para integrar informações relevantes acerca do dia-a-dia das pessoas. As funcionalidades extra presentes na aplicação são:

- Caso a luz do candeeiro tenha sido acesa entre as 7h e as 10h da manhã avisa que foi ligada uma luz, ou seja, as pessoas já se encontram acordadas. Caso contrário, aparece a mensagem que a luz se encontra desligada, o que pode ser mais preocupante.
- Caso o consumo de energia elétrica seja superior a 5000w o utilizador recebe um alerta a dizer que o consumo atual é elevado. Caso seja entre a 0h e 7h, é automaticamente desligado.
- Quanto à temperatura ambiente, caso exceda os 25°C o utilizador recebe um aviso na aplicação a indicar que a temperatura está muito elevada e, caso seja entre as 0h e as 7h da manhã, o equipamento que se encontra a fornecer calor é automaticamente desligado.
- Quanto ao elevado consumo de água, caso haja um consumo elevado novamente entre a 0h e 7h, o utilizador através da aplicação consegue visualizar a anomalia.

Nota: Para o corte de água caso houvesse algum tipo de fuga seria necessário uma válvula inteligente, o que neste caso, não foi possível concretizar devido a não possuir o equipamento necessário.

5.10 Conclusão

Após todos estes passos para o desenvolvimento do projeto, foi possível verificar os pontos cruciais para o tema em questão e tornar a informação acessível em qualquer lugar que esteja graças à *Firebase* que está 24h a sincronizar informação. A utilização da *PlatformIO* foi uma mais valia visto que permite fazer uma análise ao código escrito e divulgar de imediato erros e soluções para o problema encontrado.

Capítulo

6

Conclusões e Trabalho Futuro

6.1 Conclusões Principais

Através do desenvolvimento deste projeto podemos observar que com simples aparelhos podemos criar uma "casa inteligente" com todas as funcionalidades que o utilizador pretende. Para o caso de monitorização podemos tornar a vida de pessoas mais debilitadas numa vida melhor e com assistência assim que necessário. Com o "controlo" e monitorização de certas atividades, é possível ter acesso a informação relevante e interagir de alguma forma quando necessário para evitar males maiores. Este exemplo como tantos outros permite melhorar o estilo de vida das pessoas conforme as suas necessidades e receber ajuda se necessário. Tendo em conta os milhares de idosos que vivem nas suas casas isoladamente, na minha opinião considero uma mais valia este tipo de tecnologia presente na vida dos mesmos.

6.2 Trabalho Futuro

Uma das tarefas que poderia implementar seria o controlo remoto a partir da *Web*. Dependendo do interesse do utilizador existem milhares de dispositivos que poderiam ser controlados remotamente, ou seja, existe um leque de opções para adicionar ao projeto. Um dos objetivos que gostaria de ter feito seria a integração de um altifalante para por exemplo lembrar as pessoas de certas tarefas. Outras funcionalidades interessantes seriam a implementação de comandos por voz e a utilização de dispositivos *wearable* para se tornar mais rápido e prático efetuar as operações. Como foi referido, este tema tem vários rumos por onde seguir, dependendo do contexto, do ambiente, do desejo de cada utilizador adaptar nas suas vidas. Este ramo tem estado em constante

crescimento e cada vez mais conseguimos automatizar as nossas tarefas para benefício próprio.

Apêndice

A

Funções e Excertos de Código

```
#include "FirebaseESP8266.h"
#include <ESP8266WiFi.h>
#include "ACS712.h"

#define FIREBASE_HOST "casain-2020.firebaseio.com"
#define FIREBASE_AUTH "zBSt0VZEfZgYMmhXotEoAaRr67ohO6r4Yt2UfEWY"
#define WIFI_SSID "NOS-CASA"
#define WIFI_PASSWORD "redecasa"
#define Relay1 D1

FirebaseData firebaseData;
const int sensorIn = A0;
int mVperAmp = 185;
double Voltage=0;
double VRMS=0;
double AmpsRMS=0;
int sumWatt;
float preco;
int Wattage;
int rel;

void setup()
{
    pinMode(Relay1, OUTPUT);
    digitalWrite(Relay1, HIGH);
    WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
    //Serial.print("Connecting to Wi-Fi");
    while (WiFi.status() != WL_CONNECTED)
    {
        //Serial.print(".");
        delay(300);
    }
}
```

```
}

Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
Firebase.setInt(firebaseData, "CASA/ELETRICIDADE/STATUS",0);
Firebase.reconnectWiFi(true);
pinMode(A0, INPUT);
Serial.begin(115200);
delay(10);
Firebase.setFloatDigits(2);
}

void loop()
{
    Voltage = getVPP();
    VRMS = (Voltage / 2.0) * 0.707;
    AmpsRMS = (VRMS * 1000)/mVperAmp;
    Wattage = (220*AmpsRMS);
    sumWatt = sumWatt + Wattage;
    //preco = (sumWatt/1000)*0.14*1.23;
    Firebase.setInt(firebaseData, "CASA/ELETRICIDADE/WATTS",Wattage);
    Firebase.setInt(firebaseData, "CASA/ELETRICIDADE/TOTAL_ENERGIA",
                    sumWatt);

    if(Firebase.getInt(firebaseData, "CASA/ELETRICIDADE/STATUS")){
        rel = firebaseData.intData();

        Serial.println(rel);
        if(rel==1)
        {
            digitalWrite(Relay1,HIGH);
            Serial.println("Relay 1 ON");
        }
        if(rel==0)
        {
            digitalWrite(Relay1,LOW);
            Serial.println("Relay 1 OFF");
        }
    }
}

float getVPP(){

    float result;
    int readValue;
    int maxValue = 0;
    int minValue = 1024;

    uint32_t start_time = millis();

    while((millis()-start_time) < 1000){
```

```
readValue = analogRead(sensorIn);

if (readValue > maxValue)
{
    maxValue = readValue;
}
if (readValue < minValue)
{
    minValue = readValue;
}

// Subtract min from max
result = ((maxValue - minValue) * 5) / 1024.0;
return result;
}
```

Excerto de Código A.1: Função ConsumoEletricidade.ino

Apêndice

B

Anexos

HC-SR501 PIR MOTION DETECTOR

Product Description

HC-SR501 is based on infrared technology, automatic control module, using Germany imported LHI778 probe design, high sensitivity, high reliability, ultra-low voltage operating mode, widely used in various auto-sensing electrical equipment, especially for battery-powered automatic controlled products.

Specification:

- Voltage: 5V ~ 20V
- Power Consumption: 65mA
- TTL output: 3.3V, 0V
- Delay time: Adjustable (3.3~>5min)
- Lock time: 0.2 sec
- Trigger methods: L – disable repeat trigger, H enable repeat trigger
- Sensing range: less than 120 degree, within 7 meters
- Temperature: -15 ~ +70
- Dimension: 32*24 mm, distance between screw 28mm, M2, Lens dimension in diameter: 23mm

Application:

Automatically sensing light for Floor, bathroom, basement, porch, warehouse, Garage, etc, ventilator, alarm, etc.

Features:

- Automatic induction: to enter the sensing range of the output is high, the person leaves the sensing range of the automatic delay off high, output low.
- Photosensitive control (optional, not factory-set) can be set photosensitive control, day or light intensity without induction.
- Temperature compensation (optional, factory reset): In the summer when the ambient temperature rises to 30 °C to 32 °C, the detection distance is slightly shorter, temperature compensation can be used for performance compensation.
- Triggered in two ways: (jumper selectable)
 - non-repeatable trigger: the sensor output high, the delay time is over, the output is automatically changed from high level to low level;
 - repeatable trigger: the sensor output high, the delay period, if there is human activity in its sensing range, the output will always remain high until the people left after the delay will be high level goes low (sensor module detects a time delay period will be automatically extended every human activity, and the starting point for the delay time to the last event of the time).
- With induction blocking time (the default setting: 2.5s blocked time): sensor module after each sensor output (high into low), followed by a blockade set period of time, during this time period sensor does not accept any sensor signal. This feature can be achieved sensor output time "and" blocking time "interval between the work can be applied to interval detection products; This function can inhibit a variety of interference in the process of load switching. (This time can be set at zero seconds ~ a few tens of seconds).
- Wide operating voltage range: default voltage DC4.5V-20V.
- Micropower consumption: static current <50 microamps, particularly suitable for battery-powered automatic control products.
- Output high signal: easy to achieve docking with the various types of circuit.

Adjustment:

- Adjust the distance potentiometer clockwise rotation, increased sensing distance (about 7 meters), on the contrary, the sensing distance decreases (about 3 meters).
- Adjust the delay potentiometer clockwise rotation sensor the delay lengthened (300S), on the contrary, shorten the induction delay (5S).

Instructions for use:

- Sensor module is powered up after a minute, in this initialization time intervals during this module will output 0-3 times, a minute later enters the standby state.
- Should try to avoid the lights and other sources of interference close direct module surface of the lens, in order to avoid the introduction of interference signal malfunction; environment should avoid the wind flow, the wind will cause interference on the sensor.
- Sensor module with dual probe, the probe window is rectangular, dual (A/B) in both ends of the longitudinal direction
 - so when the human body from left to right or right to left through the infrared spectrum to reach dual time, distance difference, the greater the difference, the more sensitive the sensor,
 - when the human body from the front to the probe or from top to bottom or from bottom to top on the direction traveled, double detects changes in the distance of less than infrared spectroscopy, no difference value the sensor insensitive or does not work;
- The dual direction of sensor should be installed parallel as far as possible in line with human movement. In order to increase the sensor angle range, the module using a circular lens also makes the probe surrounded induction, but the left and right sides still up and down in both directions sensing range, sensitivity, still need to try to install the above requirements.



ATmega328P

8-bit AVR Microcontroller with 32K Bytes In-System Programmable Flash

DATASHEET

Features

- High performance, low power AVR® 8-bit microcontroller
- Advanced RISC architecture
 - 131 powerful instructions – most single clock cycle execution
 - 32 × 8 general purpose working registers
 - Fully static operation
 - Up to 16MIPS throughput at 16MHz
 - On-chip 2-cycle multiplier
- High endurance non-volatile memory segments
 - 32K bytes of in-system self-programmable flash program memory
 - 1Kbytes EEPROM
 - 2Kbytes internal SRAM
 - Write/erase cycles: 10,000 flash/100,000 EEPROM
 - Optional boot code section with independent lock bits
 - In-system programming by on-chip boot program
 - True read-while-write operation
 - Programming lock for software security
- Peripheral features
 - Two 8-bit Timer/Counters with separate prescaler and compare mode
 - One 16-bit Timer/Counter with separate prescaler, compare mode, and capture mode
 - Real time counter with separate oscillator
 - Six PWM channels
 - 8-channel 10-bit ADC in TQFP and QFN/MLF package
 - Temperature measurement
 - Programmable serial USART
 - Master/slave SPI serial interface
 - Byte-oriented 2-wire serial interface (Phillips I²C compatible)
 - Programmable watchdog timer with separate on-chip oscillator
 - On-chip analog comparator
 - Interrupt and wake-up on pin change
- Special microcontroller features
 - Power-on reset and programmable brown-out detection
 - Internal calibrated oscillator
 - External and internal interrupt sources
 - Six sleep modes: Idle, ADC noise reduction, power-save, power-down, standby, and extended standby

78100-AVR-01/15

FL-3FF **PC Board Relay**



E332369

19.2×15.5×15.5



- 10A switching capability
- Small footprint
- Sealed type available
- Class B/F available
- Conform to RoHS,ELV directive

COIL DATA				
at 20°C				
Coil Power 0.36W				
Nominal Voltage (VDC)	Rated Current (mA)	Max Operate Voltage (VDC)	Min Release Voltage (VDC)	Coil Resistance ($\Omega \pm 10\%$)
3	120	2.25	0.15	25
5	71.4	3.75	0.25	69
6	60	4.5	0.3	100
9	40	6.75	0.45	225
12	30	9	0.6	400
18	20	13.5	0.9	900
24	15	18	1.2	1600
48	7.5	36	2.4	6400

APPROVED STANDARDS	
UL & cUL	IH:10A 240VAC 12A 120VAC IZ: 7A 240VAC 10A 120VAC

ORDERING CODE					
FL-3FF— S — H F (TBF-1) / 12VDC					
1	2	3	4	5	6

1. Relay Model 2. Sealed 3. Contact Form H: Form A Z: Form C 4. Temperature Range F: UL 105°C Nil: UL 85°C 5. Insulated Level TBF-1: Class F Nil: Standard 6. Coil Nominal Voltage 3,5,6,9,12,18,24,48VDC



FLOURISHING RELAY
NINGBO FLOURISHING PRECISION ELECTRON CO.,LTD

2012.03.A

ITEM:YF-S201C**Product Features:**

1. The appearance of this product is light and flexible, small size, easy to install.
2. Impeller with stainless steel beads inside, always wear.
3. Seal using the upper and lower force structure will never leak.
4. Hall elements imported from Germany, and encapsulated with potting, To prevent water, never aging.
5. All raw materials are in line with ROHS testing standards

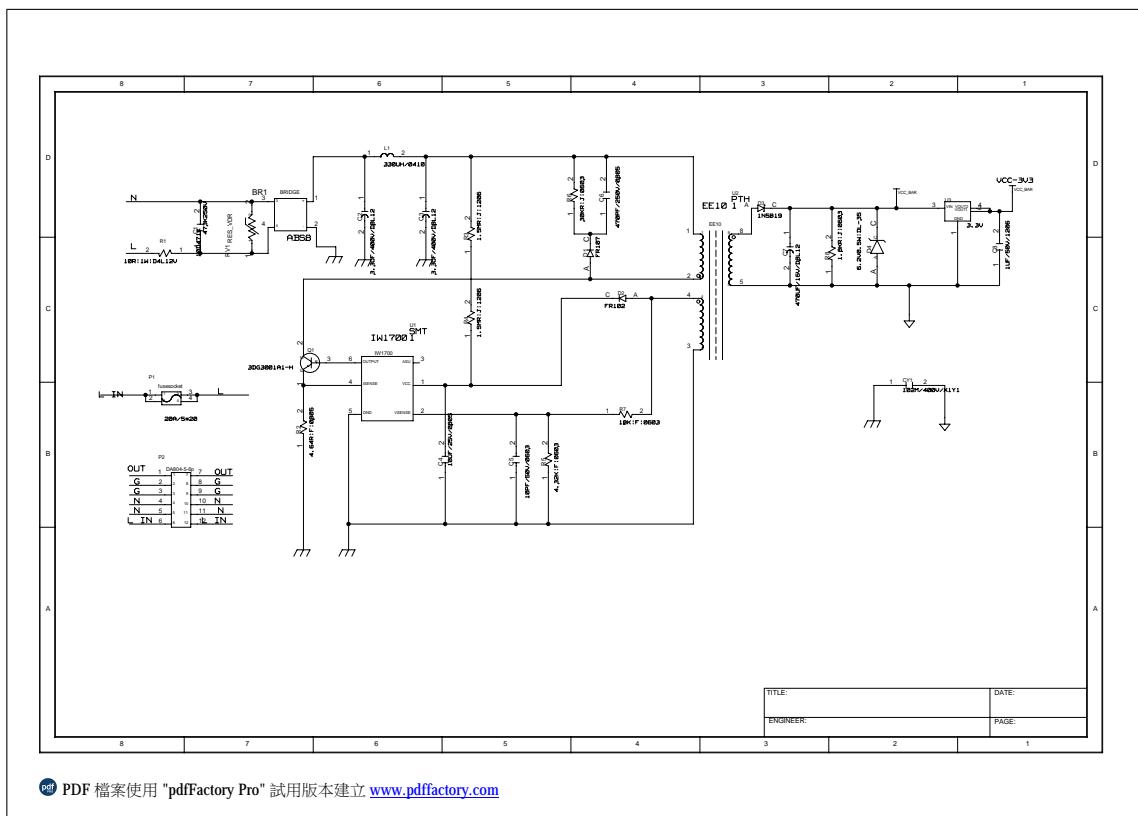
Product Description:

Water flow sensor is mainly composed of plastic body, water rotor components and Hall sensors.

It is installed in the water heater inlet, used to detect water flow, when the water flow through the rotor assembly,

The magnetic rotor rotates and the rotational speed changes with the flow rate change, the Hall sensor outputs the corresponding pulse

Signal, feedback to the controller, the controller to determine the size of water flow, regulation.



DHT 11 Humidity & Temperature Sensor

1. Introduction

DHT11 Temperature & Humidity Sensor features a temperature & humidity sensor complex with a calibrated digital signal output. By using the exclusive digital-signal-acquisition technique and temperature & humidity sensing technology, it ensures high reliability and excellent long-term stability. This sensor includes a resistive-type humidity measurement component and an NTC temperature measurement component, and connects to a high-performance 8-bit microcontroller, offering excellent quality, fast response, anti-interference ability and cost-effectiveness.



1. Overview

1.2. Specifications

Table 1-1. Specifications

Categories	Items	Parameters
Wi-Fi	Certification	Wi-Fi Alliance
	Protocols	802.11 b/g/n (HT20)
	Frequency Range	2.4 GHz ~ 2.5 GHz (2400 MHz ~ 2483.5 MHz)
	TX Power	802.11 b: +20 dBm 802.11 g: +17 dBm 802.11 n: +14 dBm
	Rx Sensitivity	802.11 b: -91 dbm (11 Mbps) 802.11 g: -75 dbm (54 Mbps) 802.11 n: -72 dbm (MCS7)
	Antenna	PCB Trace, External, IPEX Connector, Ceramic Chip
	CPU	Tensilica L106 32-bit processor
Hardware	Peripheral Interface	UART/SDIO/SPI/I2C/I2S/IR Remote Control GPIO/ADC/PWM/LED Light & Button
	Operating Voltage	2.5 V ~ 3.6 V
	Operating Current	Average value: 80 mA
	Operating Temperature Range	-40 °C ~ 125 °C
	Package Size	QFN32-pin (5 mm x 5 mm)
	External Interface	-
Software	Wi-Fi Mode	Station/SoftAP/SoftAP+Station
	Security	WPA/WPA2
	Encryption	WEP/TKIP/AES
	Firmware Upgrade	UART Download / OTA (via network)
	Software Development	Supports Cloud Server Development / Firmware and SDK for fast on-chip programming
	Network Protocols	IPv4, TCP/UDP/HTTP
	User Configuration	AT Instruction Set, Cloud Server, Android/iOS App

Note:

The TX power can be configured based on the actual user scenarios.

Bibliografia

- [1] C. Humphreys. Can smart home technology support older people. *British journal of community nursing*, 23(9):435–438, 2018.
- [2] Lexis M. A. Joore M. Dirksen C. D. De Witte L. P. Peetoom, K. K. Literature review on monitoring technologies and their outcomes in independently living elderly people. *Disability and Rehabilitation: Assistive Technology*, 10(4):271–294, 2015.
- [3] Cotea C. Pullman S. Pietrzak, E. Does smart home technology prevent falls in community-dwelling older adults: a literature review. *Journal of Innovation in Health Informatics*, 21(3):105–112, 2014.
- [4] McKinstry B. Stewart, L. S. Fear of falling and the use of telecare by older people. *British Journal of Occupational Therapy*, 75(7):304–312, 2012.
- [5] Stroulia E. Nikolaidis I. Miguel-Cruz A. Rincon A. R. Liu, L. Smart homes and home health monitoring technologies for older adults: A systematic review. *International journal of medical informatics*, 91:44–59, 2016.
- [6] Sitar-Taut D. A. Cramariuc O. Negrean V. Sampelean D. Rusu L. ... Cozma A. Sitar-Taut, A. V. Smart homes for older people involved in rehabilitation activities-reality or dream, acceptance or rejection. *Balneo Res J*, 9(3), 2018.
- [7] Google. Android Studio, 2013. [Online] <https://developer.android.com/studio>. Último acesso a 10 de Fevereiro de 2021.
- [8] JetBrains. IntelliJ IDEA, 2001. [Online] <https://www.jetbrains.com/idea/>. Último acesso a 15 de Janeiro de 2021.
- [9] Google. Firebase, 2005. [Online] <https://firebase.google.com/>. Último acesso a 14 de Fevereiro de 2021.
- [10] Tom Igoe Gianluca Martino e David Mellis Massimo Banzi, David Cuartielles. Arduino, 2005. [Online] <https://www.arduino.cc/>. Último acesso a 21 de Janeiro de 2021.

- [11] Microsoft. Visual Studio Code, 2015. [Online] <https://code.visualstudio.com/>. Último acesso a 20 de Fevereiro de 2021.
- [12] ITEAD Intelligent Systems CO. ITEAD Wiki, 2011. [Online] <https://www.itead.cc/wiki/>. Último acesso a 17 de Maio de 2021.