



Universidade do Minho

Universidade do Minho

Licenciatura em Ciências da Computação

POO - Trabalho Prático

Grupo nº5

Simão Pedro Batista Caridade Quintela
(A97444)

Pedro Alexandre Silva Gomes
(A91647)

Gonçalo da Silva
(A95696)

18 de maio de 2022



Conteúdo

1	Introdução	3
2	Classes	4
2.1	App	4
2.2	CasaInteligente	4
2.2.1	CasaInteligente	4
2.2.2	CasaInteligenteTest	4
2.2.3	SmartDevices	4
2.3	ComercializadoresEnergia	5
2.3.1	Comercializador	5
2.3.2	Fatura	6
2.4	Comunidade	6
2.5	Controller	6
2.6	Parser	6
2.7	SaveProgramText	6
2.8	SimulParser	6
2.9	View	6
2.10	Diagrama de classes do <i>blueJ</i>	7
3	Estrutura do projeto	9
4	Diagrama do <i>IntelliJ</i>	10
5	Conclusão	11

Capítulo 1

Introdução

Este relatório descreve o desenvolvimento do projeto prático da Unidade Curricular de Programação Orientada aos Objetos, inserida no 2ºano da Licenciatura em Ciências da Computação da Universidade do Minho.

Este trabalho consistia no desenvolver de um sistema que monitorize e registre a informação sobre o consumo energético das habitações de uma comunidade. Em cada casa existem um conjunto muito alargado de dispositivos que são todos controlados a partir deste programa, os *SmartDevice*. Para que a realização do referido fosse possível, utilizamos os conhecimentos da linguagem *Java*, adquiridos ao longo do semestre.

Capítulo 2

Classes

Estas são as classes que constituem o nosso programa:

2.1 App

```
static Scanner scan = new Scanner(System.in);  
static Comunidade comunidade = new Comunidade("Jackson");  
static Controller controller = new Controller(comunidade);
```

2.2 CasaInteligente

2.2.1 CasaInteligente

```
private String proprietario;  
//private int numeroDePorta;  
private int NIF;  
//private String morada;  
private String fornecedor;  
private Map<String, SmartDevice> devices; // identificador ->  
SmartDevice  
private Map<String, List<String>> locations; // Espaço -> Lista  
codigo dos devices  
private List<Fatura> faturas; // lista de faturas que foram geradas e  
associadas casa
```

2.2.2 CasaInteligenteTest

2.2.3 SmartDevices

- SmartDevice

```

private String id;
private boolean on;
private LocalDate time;
private float consumption;
private float consumptionPerDay;
private int custoInstalacao;

```

- SmartDeviceTest

- SmartBulb

```

private static final int WARM = 80;
private static final int NEUTRAL = 60;
private static final int COLD = 40;
private int tone;
private int dimensions;

```

- SmartBulbTest

- SmartSpeaker

```

private static final int MAX = 100; //volume maximo
private int volume;
private String channel;
private String brand;

```

- SmartCamera

```

private int xRes;
private int yRes;
private int fileSize;
private float custoInstalacao;

```

2.3 ComercializadoresEnergia

2.3.1 Comercializador

```

private String nomeEmpresa;
private int numeroDispositivos;
private int valorBase;
private int imposto;
private Map<String, List<Fatura>> faturas; // Propriet rio -> Lista
de Faturas
//private double precoDiaPorDispositivo = numeroDispositivos > 10?(
    valorBase * consumoDispositivo * (1 + imposto)) * 0.9 : (
    valorBase * consumoDispositivo * (1 + imposto)) * 0.75;

```

2.3.2 Fatura

```
private int codigo;  
private int nif;  
LocalDate dataEmissao;  
private Map<String, Float> consumoDevice; //id -> consumo  
private String empresa;  
private String cliente;
```

2.4 Comunidade

```
private String nomeDaComunidade;  
private Map<String, CasaInteligente> casas;  
private Map<String, Comercializador> mercado;
```

2.5 Controller

```
private Comunidade comunidade;  
private int idFatura;  
private LocalDate timeNow;
```

2.6 Parser

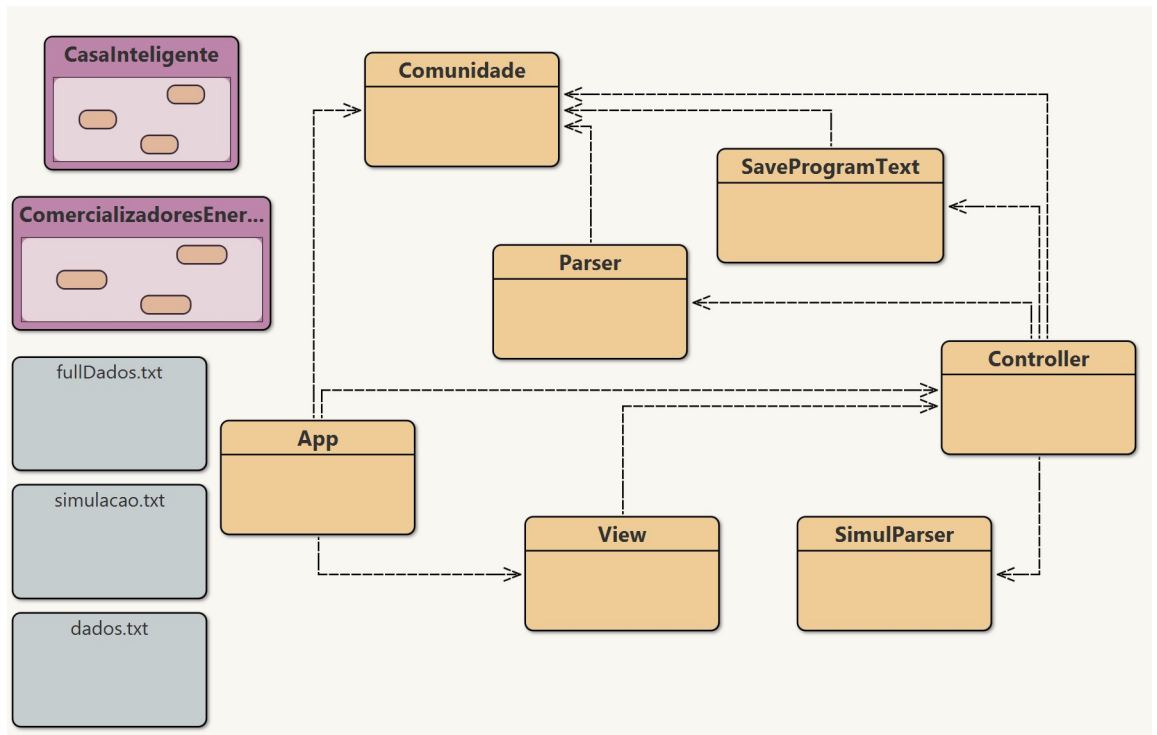
2.7 SaveProgramText

2.8 SimulParser

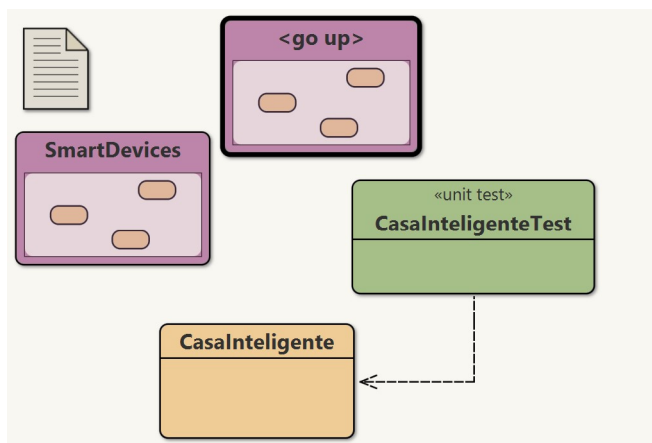
2.9 View

```
private Controller controller;  
private Scanner scan;
```

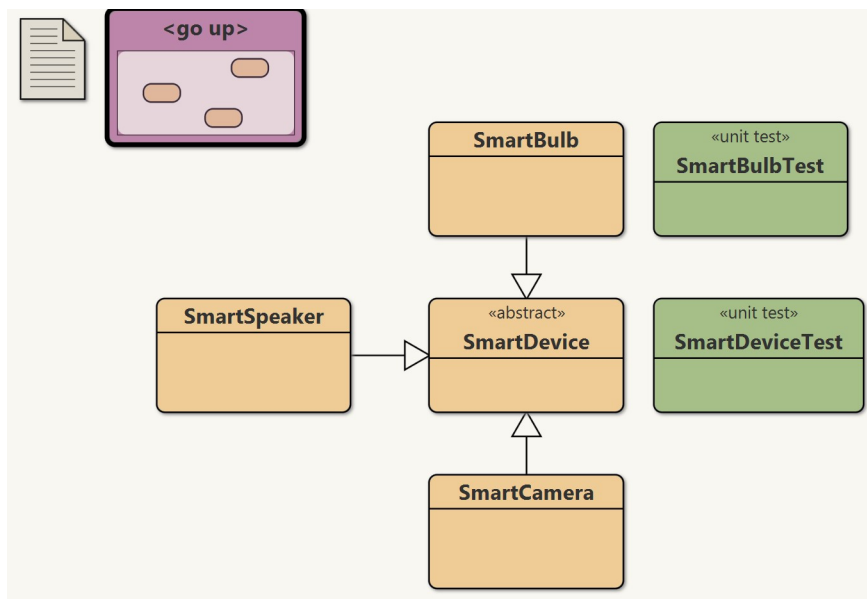
2.10 Diagrama de classes do *blueJ*



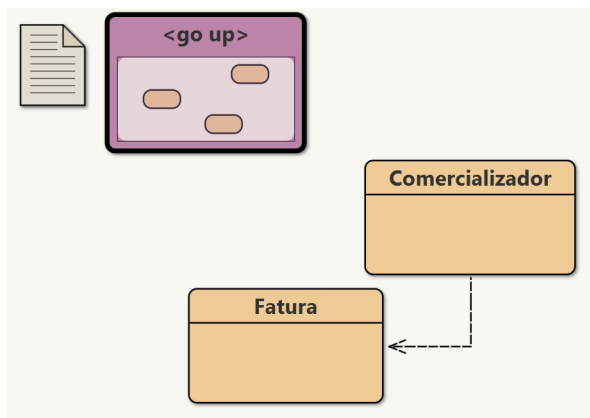
1. Diagrama principal com as super classes, interfaces de simulação e ficheiros .txt



2. Diagrama da classe CasaInteligente



3. Diagrama da classe SmartDevices



4. Diagrama da classe ComercializadoresEnergia

Capítulo 3

Estrutura do projeto

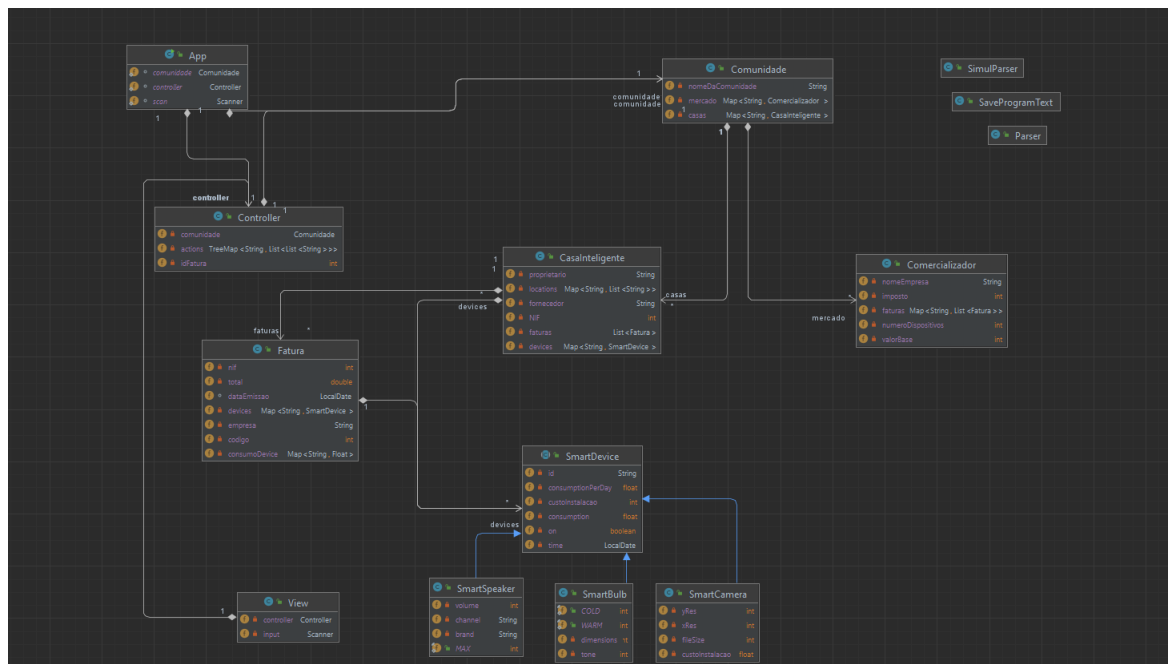
O nosso projeto segue o modelo de estrutura **Model View Controller (MVC)**, que consiste na organização deste em três camadas:

- A camada de dados (**o modelo**) é composta pelas classes CasaInteligente, SmartDevice, SmartBulb, SmartSpeaker, SmartCamera, Comercializador, Fatura, Comunidade e a pelas interfaces de Simulação.
- A camada de interação com o utilizador (**a vista, ou apresentação**) é composta unicamente pela classe View.
- A camada de controlo do fluxo do programa (**o controlador**) é composta pela classe Controller.

Em todo o projeto respeitamos a ideia de encapsulamento, como pedido.

Capítulo 4

Diagrama do *Intellij*



Capítulo 5

Conclusão

Em conclusão, a nível geral, e tendo em conta o panorama apresentado nos capítulos anteriores e os objetivos pedidos para esta realização, como grupo, achamos que todos os objetivos foram cumpridos, conseguindo superar com sucesso, todas as dificuldades que fomos encontrando, sempre com um olhar crítico e empenhados. Acreditamos que aprendemos os conteúdos e objetivos desta Unidade Curricular e em particular deste projeto.