



Universidade do Minho
Escola de Engenharia
Mestrado Integrado em Engenharia Informática

Unidade Curricular de Laboratórios de Informática IV

Ano Letivo de 2020/2021

SOL:AR - Soluções Energéticas Conscientes

Júlio Miguel de Sá Lima Magalhães Alves (A89468)
Eduardo Benjamim Lopes Coelho (A89616)
Carolina da Torre Vila Chã (A89495)
Henrique Gabriel dos Santos Neto (A89618)

30 de abril de 2022

L | 4

Data de Recepção	
Responsável	
Avaliação	
Observações	

SOL:AR - Soluções Energéticas Conscientes

Júlio Miguel de Sá Lima Magalhães Alves (A89468)

Eduardo Benjamim Lopes Coelho (A89616)

Carolina da Torre Vila Chã (A89495)

Henrique Gabriel dos Santos Neto (A89618)

30 de abril de 2022

Resumo

Este projeto, realizado no âmbito da unidade curricular de Laboratórios de Informática IV, consiste no desenvolvimento de um sistema de monitorização de eventos. Com este trabalho, pretende-se aproximar ao máximo o processo de desenvolvimento de uma aplicação ao domínio da Engenharia de Software, na medida em que nos proporciona um contacto mais "profissional" e real com o mundo do desenvolvimento de software para sistemas reais.

Com este objetivo em mente, decidimos criar um sistema de monitorização de painéis solares, a SOL:AR (Soluções Energéticas Conscientes). Com o aumento da aderência a energias renováveis tais como a energia solar, acreditamos ser um tema bastante relevante, tanto no presente, como no futuro.

Este projeto está dividido em três fases distintas:

- **Fundamentação** - Fundamentar, projetar e gerir o desenvolvimento de um sistema de software;
- **Especificação** - Analisar e especificar de forma completa todos os requisitos operacionais e funcionais de um sistema de software;
- **Construção** - Desenvolver, validar, documentar e instalar sistemas de software.

Área de Aplicação: Engenharia de Software, Desenvolvimento de Aplicações, Energia Solar, Aplicações Web

Palavras-Chave: Engenharia, Software, Energias Renováveis, Metodologia, Fundamentação, Especificação, Construção

Índice

1	Introdução	1
1.1	Contextualização	1
1.2	Apresentação do Caso de Estudo	1
1.3	Motivação e Objectivos	2
1.4	Estrutura do Relatório	2
2	Fundamentação e Projeção	4
2.1	Justificação do sistema	4
2.2	Utilidade do sistema	5
2.3	Estabelecimento da identidade do projeto	6
2.4	Identificação dos recursos necessários	6
2.5	Maqueta do sistema	7
2.6	Definição de um conjunto de medidas de sucesso	9
2.7	Plano de desenvolvimento (diagrama GANTT)	10
3	Levantamento e Análise de Requisitos	11
3.1	Tabela de Requisitos Funcionais	11
3.2	Requisitos não funcionais	13
4	Especificação em <i>UML</i>	14
4.1	Modelo de domínio	14
4.2	Diagrama de casos de uso	16
4.3	Especificação dos Casos de Uso	17
4.3.1	Registrar nova conta	17
4.3.2	Efetuar Login	17
4.3.3	Aceder aos consumos energéticos	18
4.3.4	Aceder à produção energética	18
4.3.5	Ver todos os eventos	18
4.3.6	Ativar alerta para um evento	19
4.3.7	Ver alertas ativados	19
4.3.8	Ver histórico de reparações	20
4.3.9	Ver informações pessoais	20
4.3.10	Editar informações pessoais	20
4.3.11	Eliminar alerta	21
4.3.12	Ver alertas de avarias	21
4.3.13	Ver estado dos painéis solares	21
4.3.14	Marcar horário de manutenção	22

4.3.15 Ver distribuição dos painéis num mapa	22
4.3.16 Ligar/Desligar um painel solar	23
4.3.17 Comunicar Avaria	23
4.3.18 Comunicar produção energética	24
4.4 Arquitetura da Aplicação	25
4.4.1 Diagrama de componentes	25
4.4.2 Diagrama de classes	26
4.5 Diagramas de sequência	27
4.5.1 Registar Nova Conta	27
4.5.2 Efetuar Login	28
4.5.3 Aceder aos consumos energéticos	28
4.5.4 Aceder à produção energética	29
4.5.5 Ver todos os eventos	29
4.5.6 Ativar lembrete para um evento	30
4.5.7 Ver alertas ativados	30
4.5.8 Ver informações pessoais	31
4.5.9 Editar informações do perfil	32
4.5.10 Ver histórico de reparações	32
4.5.11 Eliminar lembrete	33
4.5.12 Marcar horário de manutenção	33
4.5.13 Ver distribuição dos painéis num mapa	34
4.5.14 Ver estado dos painéis solares	34
4.5.15 Ligar/desligar um painel solar	35
4.5.16 Comunicar avaria	35
5 Base de dados	36
5.1 Modelo Lógico	36
5.2 Dicionário de Dados	37
5.3 Espaço ocupado	39
6 Definição da interface do sistema	41
6.1 Máquina de estado da interface do sistema	41
6.2 Mockups	42
6.3 Diagramas de Atividade	48
6.3.1 Diagrama de atividade da página inicial	48
6.3.2 Diagrama de atividade da dashboard	49
6.3.3 Diagrama de atividade do calendário	50
6.3.4 Diagrama de atividade da dashboard do funcionário	51
6.3.5 Diagrama de atividade do utilizador	52
6.3.6 Diagrama de atividade do utilizador	52
6.4 Ponto de situação - Diagrama de GANTT atualizado	53
7 Metodologia da Implementação	54
8 Ferramentas utilizadas na Implementação	55

9 Desenvolvimento do Projeto	56
9.1 Conexão à base de dados	56
9.2 Visualização dos painéis no mapa	56
9.3 Calendarização de eventos	56
10 Produto Final	57
10.1 Área do Utilizador	57
10.2 Área do Funcionário	58
10.2.1 Marcação de Eventos	59
11 Conclusões e Trabalho Futuro	61
Lista de Siglas e Acrónimos	64
Anexos	65
Anexo 1	65

Lista de Figuras

2.1	Evolução do uso de energia solar entre 2000 e 2019	4
2.2	Evolução dos custos de energia solar entre 2000 e 2015	5
2.3	Maqueta do sistema	7
2.4	Mockup da <i>landing page</i> da aplicação	9
2.5	Diagrama de GANTT do plano de desenvolvimento	10
4.1	Modelo de Domínio	14
4.2	Diagrama de use cases	16
4.3	Diagrama de componentes	25
4.4	Diagrama de classes	26
5.1	Dicionário de Dados	36
6.1	Máquina de estado da interface do sistema	41
6.2	Página inicial da aplicação (sem sessão iniciada)	42
6.3	Dashboard do utilizador	43
6.4	Calendário do utilizador	44
6.5	Dashboard do funcionário	45
6.6	Secção das informações pessoais (Utilizador)	46
6.7	Secção do estado dos painéis (Utilizador)	47
6.8	Diagrama de atividade da página inicial	48
6.9	Diagrama de atividade da dashboard	49
6.10	Diagrama de atividade do calendário	50
6.11	Diagrama de atividade do mapa	51
6.12	Diagrama de atividade do perfil do utilizador	52
6.13	Diagrama de atividade do estado dos painéis	52
6.14	Diagrama de GANTT atualizado	53
10.1	Página Inicial	57
10.2	Visualização da Dashboard	58
10.3	Visualização dos Painéis no mapa	58
10.4	Visualização dos eventos	59
10.5	Visualização dos Eventos no calendário com vista diária	59
10.6	Visualização dos Eventos no calendário com vista semanal	60
10.7	Visualização dos Eventos no calendário com vista mensal	60

11.1 Consumo de energia elétrica per capita: total e por tipo de consumo kWh (quilowatt-hora)/ hab. - Rácio	65
--	----

Lista de Tabelas

2.1	Identidade do projeto	6
3.1	Tabela de Requisitos	11
5.1	Dicionário de Dados	37

1 Introdução

1.1 Contextualização

Num mundo em que as necessidades energéticas de cada habitação aumentam exponencialmente(11), as centrais de produção de electricidade a partir de fontes não renováveis têm um impacto cada vez mais negativo sobre o ambiente.

Lentamente, as maiores companhias do planeta têm estado a efectuar uma mudança para fontes de energia renováveis, e um futuro 100% renovável é algo que muitos especialistas consideram longínquo, mas exequível, se existir um esforço conjunto por parte da população do planeta. Torna-se, então, responsabilidade de todos, desde as grandes companhias mundiais, ao cidadão comum, contribuir para um futuro mais estável e saudável.

Portugal é um país pioneiro nesta área. Já em maio de 2016 o país utilizou apenas energia proveniente de fontes renováveis durante quatro dias consecutivos(1).

Com o crescente número de casos de COVID-19 e consequente confinamento da população mundial, falhas no serviço têm um impacto muito maior na vida da população, pelo que muitas pessoas têm estado à procura de soluções para mitigar este impacto negativo, e assegurar que qualquer falha total não implique uma completa falta energética nas suas casas. A escolha mais popular é a da instalação de painéis solares.

É neste cenário que a SOL:AR nasce, com o objectivo de fornecer uma aplicação capaz de gerir um sistema de painéis solares, tanto para os funcionários da empresa responsável pela sua instalação/manutenção, como aos habitantes. Com este propósito em mente, a SOL:AR tornará o acesso a esta fonte de energia menos intimidante e mais acessível para o utilizador comum e aumentará a eficiência da manutenção e gestão dos painéis solares por parte das empresas responsáveis pela instalação das mesmas.

1.2 Apresentação do Caso de Estudo

A SOL:AR - Soluções Energéticas Conscientes é um software de monitorização da produção de energia solar desenvolvido em parceria com uma empresa multinacional do ramo das energias. Sendo que essa empresa possui uma imagem de respeito e uma grande reputação, exige que todos os produtos que disponibiliza à sua clientela sejam produtos que correspondam aos mais

exigentes padrões de qualidade, o que faz com que tudo aquilo que faz seja feito com o mais elevado profissionalismo e rigor.

Desta forma, este projeto foi elaborado com base em variadíssimos requisitos que vão desde a monitorização dos níveis de energia que uma habitação será capaz de produzir num dia até à notificação ao usuário que não irá conseguir produzir a energia estipulada pelo mesmo, sendo que este terá a opção de personalizar o que deseja que aconteça nos mais variados cenários.

1.3 Motivação e Objectivos

Com a necessidade de marcar a diferença no mundo das energias renováveis, a nossa empresa-mãe ALSET decidiu que a melhor forma de o fazer seria fazendo mais que oferecer tarifários extremamente competitivos aos seus clientes. Então, após um processo de brainstorming de ideias, chegou-se à conclusão de que para se distanciar da concorrência teria de se oferecer ao cliente a liberdade de personalizar o que pode fazer com a energia que os seus painéis solares produzem.

A criação deste software parte exatamente desta ideia, em que o objetivo é dar a possibilidade ao usuário de decidir como controlar a energia que produz. Tendo em mente que o usuário poderá não estar sempre conectado à rede, também terá a opção de automatizar este processo, podendo escolher de antemão o que fazer com base num conjunto de cenários pré-definidos.

1.4 Estrutura do Relatório

O nosso relatório está dividido em 3 capítulos: A Introdução, que já foi detalhada anteriormente, a "Fundamentação e Projeção" e, por último, "Conclusões e Trabalho Futuro", onde iremos concluir tudo que o foi abordado nesta fase e começar a projetar o trabalho das próximas fases.

O segundo capítulo "Fundamentação e Projeção" está dividido em 7 subcapítulos:

- Justificação do Sistema - Aqui iremos abordar o porquê de termos escolhido este sistema para o nosso projeto.
- Utilidade do Sistema - Neste subcapítulo detalharemos as funcionalidades do nosso sistema e as suas vantagens.
- Estabelecimento da identidade do projeto - Definição da identidade do nosso sistema.
- Identificação dos recursos necessários - Abordagem de todos os recursos necessários ao desenvolvimento do nosso sistema.

- Maqueta do sistema - Definição de uma maquete do sistema.
- Definição de um conjunto de medidas de sucesso - Métricas que estabelecem o quanto bem sucedida foi a nossa aplicação.
- Plano de desenvolvimento (diagrama Gantt) - Para melhor organizarmos todas as tarefas de todas as fases do projeto, iremos apresentar um diagrama de Gantt onde estão representadas todas as etapas, tais como os seus períodos de duração e datas, os seus responsáveis, progresso e outras informações que sejam relevantes. Nas próximas fases do trabalho iremos reapresentar este diagrama para confrontar o esperado com a realidade, tal como atualizar o progresso de cada etapa.

2 Fundamentação e Projeção

2.1 Justificação do sistema

Ao longo das últimas duas décadas, as energias renováveis têm sido um dos principais focos do mundo, como resposta às alterações climáticas, escassez e preço das combustíveis fósseis, o que levou a um aumento exponencial do investimento por parte das empresas e da consequente adesão por parte das pessoas. A figura 2.1 representa o crescimento do consumo de energia solar, que das energias renováveis mais limpas e promissoras para o futuro.

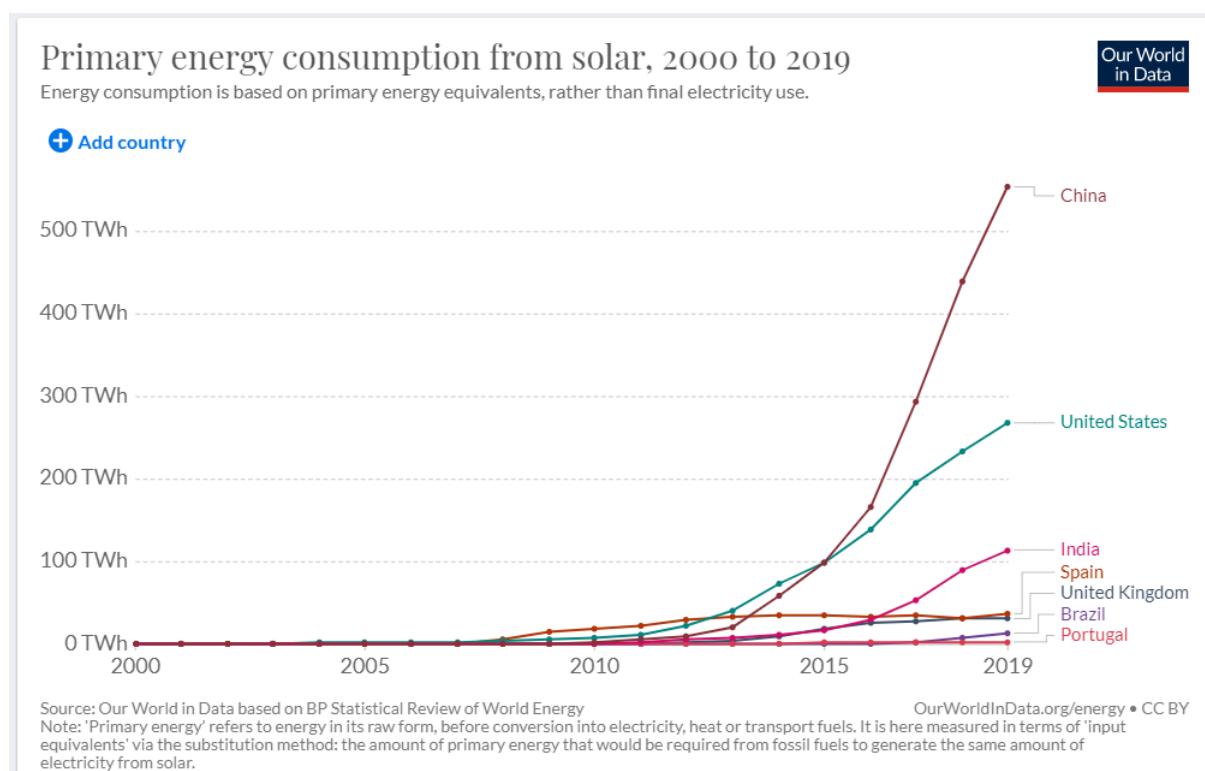


Figura 2.1: Evolução do uso de energia solar entre 2000 e 2019

Como referimos anteriormente, outro catalisador desta forte adesão é o preço por watt da energia solar, que para além de, em vários contextos, ser mais baixo que as alternativas resultantes da combustão de combustíveis fósseis, tem mostrado uma tendência a descer com o progressivo aumento da tecnologia e consequente facilidade de acessibilidade ao mesmo. Para

além disso, o custo de um sistema de painéis solares tem vindo a descer consideravelmente, o que também aumenta a atratividade desta solução energética (figura 2.2).

Tendo em conta este crescimento, a SOL:AR visa tornar a manutenção e gestão dos painéis solares mais simples e eficiente, tanto para as empresas que fazem a instalação como os seus clientes.

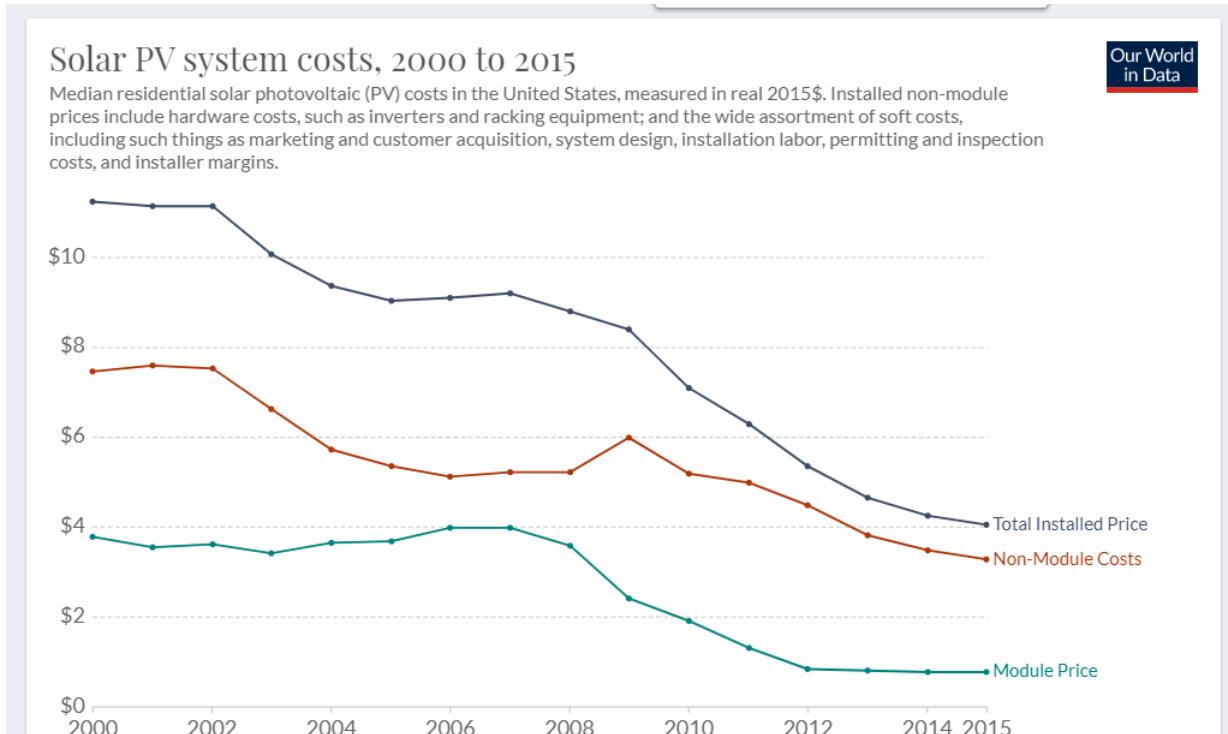


Figura 2.2: Evolução dos custos de energia solar entre 2000 e 2015

2.2 Utilidade do sistema

Como foi referido anteriormente, o nosso sistema terá dois tipos de utilizadores distintos: funcionários e clientes. Os funcionários poderão receber alertas de avarias e/ou necessidade de manutenção por parte dos painéis solares, tal como marcar um horário de manutenção, que será visível pelos clientes. Poderão também verificar a distribuição dos painéis solares num mapa. Por outro lado, os clientes poderão ver todos estes eventos organizados numa dashboard, verificar informações como o consumo e produção energética ao longo do tempo, definir alertas (por exemplo, de baixa produção de energia devido a mau tempo), receber sugestões de medidas a tomar.

Nome	SOL:AR - Soluções Energéticas Conscientes
Slogan	Solar Energy, Made simple
Categoria	Gestão Energética
Idiomas	Português, Inglês
Características	Gestão da produção de Energia Solar Gestão de consumo de uma Habitação Gestão de períodos de manutenção técnica Previsão de quebras na produção elétrica
Faixa Etária	18+

Tabela 2.1: Identidade do projeto

2.3 Estabelecimento da identidade do projeto

2.4 Identificação dos recursos necessários

Como é esperado, uma aplicação desta dimensão necessitará de vários recursos e financiamentos para a sua conclusão. Desta forma, para diminuir os eventuais custos de desenvolvimento é necessário pré-alocar os vários recursos intervenientes no desenvolvimento da aplicação. No nosso caso, tendo em conta que a SOL:AR se trata de um serviço de software relacionado com a energia fotovoltaica, para além dos recursos inerentes ao software, serão adicionalmente imprescindíveis recursos relativos às energias renováveis solares.

Primeiramente será necessário atribuir uma equipa de engenheiros de *software* com os conhecimentos e capacidades necessárias para desenvolver a aplicação em capaz. Consequentemente, existirá uma entidade de controlo responsável por garantir a qualidade e os prazos estabelecidos pelo cliente da aplicação.

Adicionalmente serão contactados vários indivíduos externos ao projeto, ou seja alguns possíveis futuros utilizadores, de maneira a se perceber quais seriam as interações que estes teriam com a aplicação e sucessivamente estabelecer qual das possíveis abordagens é mais útil e prática para um utente.

Por fim, serão estabelecidas as ferramentas e equipamentos necessários para o desenvolvimento da aplicação *web*. No nosso caso iremos utilizar a ferramenta *Visual Paradigm* como recurso para modelar e especificar a aplicação de software. Posteriormente iremos usar a plataforma *ASP.NET* desenvolvida pela *Microsoft* como meio para o desenvolvimento *HTML*, *CSS*, *Javascript* e *C#* da aplicação. Adicionalmente faremos uso ao *Microsoft SQL Server* para a integração e o gerenciamento das base de dados necessárias.

2.5 Maqueta do sistema

Esta aplicação foi concebida a pensar oferecer ao utilizador uma acessibilidade simples e que seja possível controlar a sua casa quer se encontre nela ou no outro lado do mundo.

Desta forma, e apesar de o utilizador poder estabelecer um conjunto de ações para o caso de não ter acesso à rede, é aconselhável que o utilizador esteja conectado à internet de forma a receber as notificações em tempo real e para que possa alterar as ações a realizar conforme o que lhe dê mais jeito no momento.

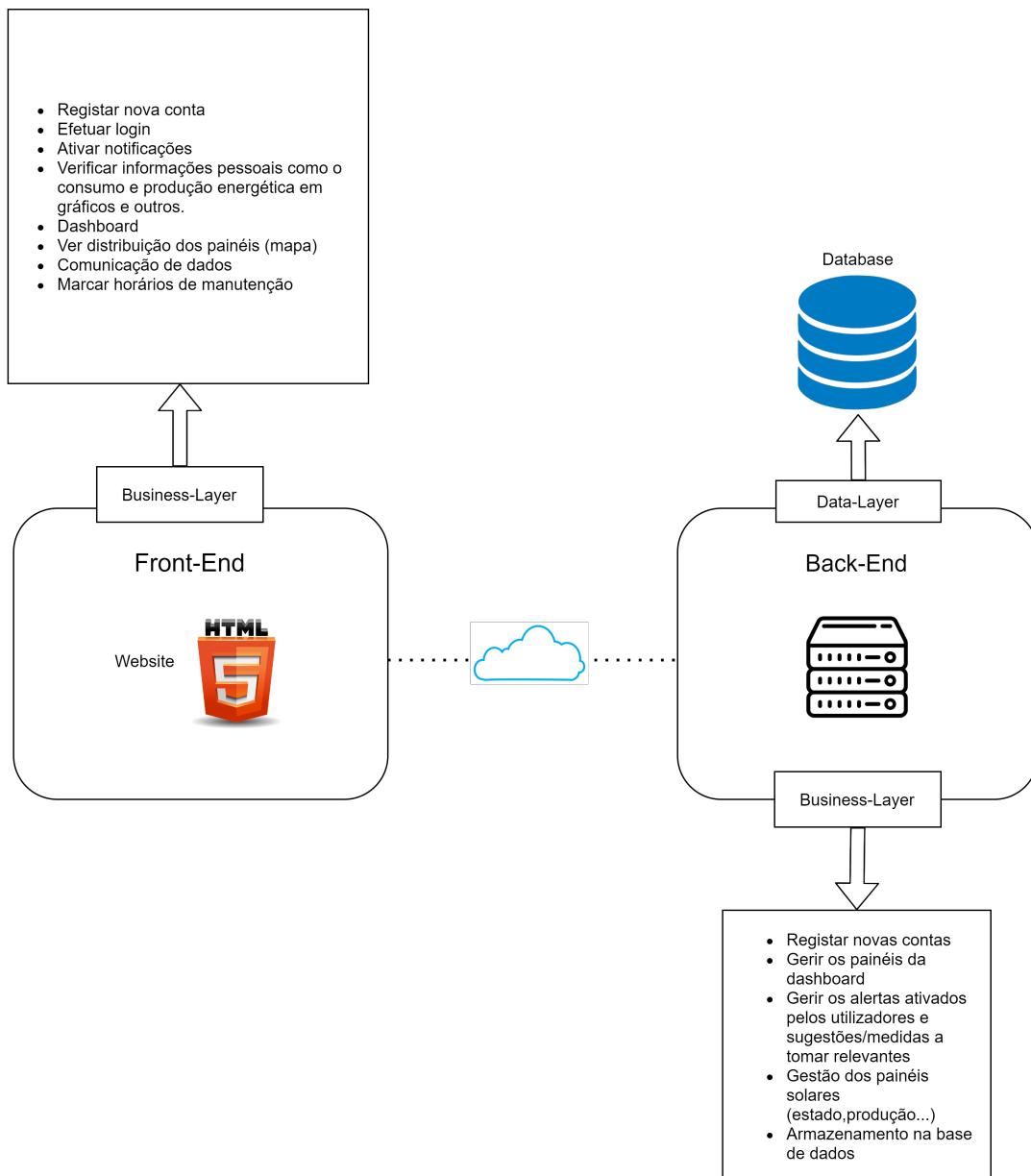


Figura 2.3: Maqueta do sistema

A SOL:AR será desenvolvida para plataformas web de modo a ter o máximo de compatibilidade entre diversos dispositivos, tais como computadores, telemóveis, tablets, entre outros.

A parte *Front-end* da SOL:AR consiste nomeadamente no *website* que está dividido em *landing page*, e a página pessoal de cada conta (seja utilizador normal, ou entidade com privilégios de administrador). A partir desta o utilizador pode consultar informações sobre a sua conta, tais como: notificações, o seu consumo, produção energética, entre outros. De forma semelhante, as entidades responsáveis pela gestão a aplicação (por exemplo, a nossa companhia-mãe AL-SET) podem consultar informações gerais tais como ver a distribuição de painéis no mapa, registar novas contas, gerir contas existentes, enviar e gerir os alertas, entre outros.

A parte *Back-end* da SOL:AR consiste no código responsável pela interação entre a *Front-end* (e a informação enviada ao sistema através dela) e a Base de Dados. A informação inserida no sistema é comunicada a esta secção da aplicação que a processa e realiza as ações necessárias.

Quando um utilizador se regista, os dados introduzidos por ele são enviados da aplicação para o servidor que, por sua vez, trata de validar e, caso sejam válidos, regista esse novo utilizador na base de dados. Qualquer alteração feita por um utilizador às informações da sua conta é, também, enviada para o servidor, que as processa e atualiza a base de dados.

Para termos uma dashboard funcional, quando um utilizador acede ao website, o servidor tem de enviar as informações relativas a todos os eventos relevantes para o website. Se um utilizador ativar notificações para um dado evento, como, por exemplo, um alerta de baixa produção de energia, então a aplicação avisa o servidor que, por sua vez, regista esta ação na base de dados e estará pronto a notificar o devido cliente (o que implicará uma nova transferência de dados para o website). Quando o alerta for enviado, o cliente poderá verificar essa mesma notificação na aplicação, tal como um conjunto de sugestões/medidas a tomar como resposta.

Para além disto, o servidor tem de manter as informações relativas a cada cliente atualizadas e armazenadas na base de dados. Deste modo, quando um utilizador aceder à página do seu perfil, o servidor poderá ir buscar essas mesmas informações e enviá-las para aplicação que, por sua vez, as apresentará ao cliente na forma adequada (por exemplo, um gráfico que mostra o consumo elétrico ao longo do tempo).

Outro aspecto importante da nossa aplicação, é a gestão dos painéis solares para os utilizadores que são responsáveis da manutenção dos mesmos, ou seja, os funcionários das empresas que realizam a instalação destes equipamentos. Assim, se na aplicação um destes utilizadores indicar que um dado painel solar necessita de manutenção, esta informação será enviada para o servidor, que terá de atualizar o estado deste painel solar na base de dados. Esta gestão é importante para garantir que a informação relativa ao estado dos painéis solares apresentada na aplicação está sempre atualizada de acordo com a realidade (por exemplo na distribuição de painéis no mapa do serviço).

Todas estas funcionalidades implicam que haja uma ligação constante entre o servidor e o website da aplicação, uma vez que todas as informações apresentadas ao utilizador têm de ser confiáveis e em tempo real.

Por exemplo, no calendário em que o utilizador pode verificar a distribuição de eventos, é importante que o servidor mantenha a aplicação devidamente atualizada para garantir que os clientes tenham sempre ao seu dispor informação credível e correta. Este aspeto é também bastante importante no sentido contrário, ou seja, queremos garantir que o fluxo de dados da aplicação para o servidor seja a mais rápida possível, de modo a que a latência entre as ações do utilizador seja mínima (isto verifica-se em ações, tais como login, registo, ativação de alertas, etc).

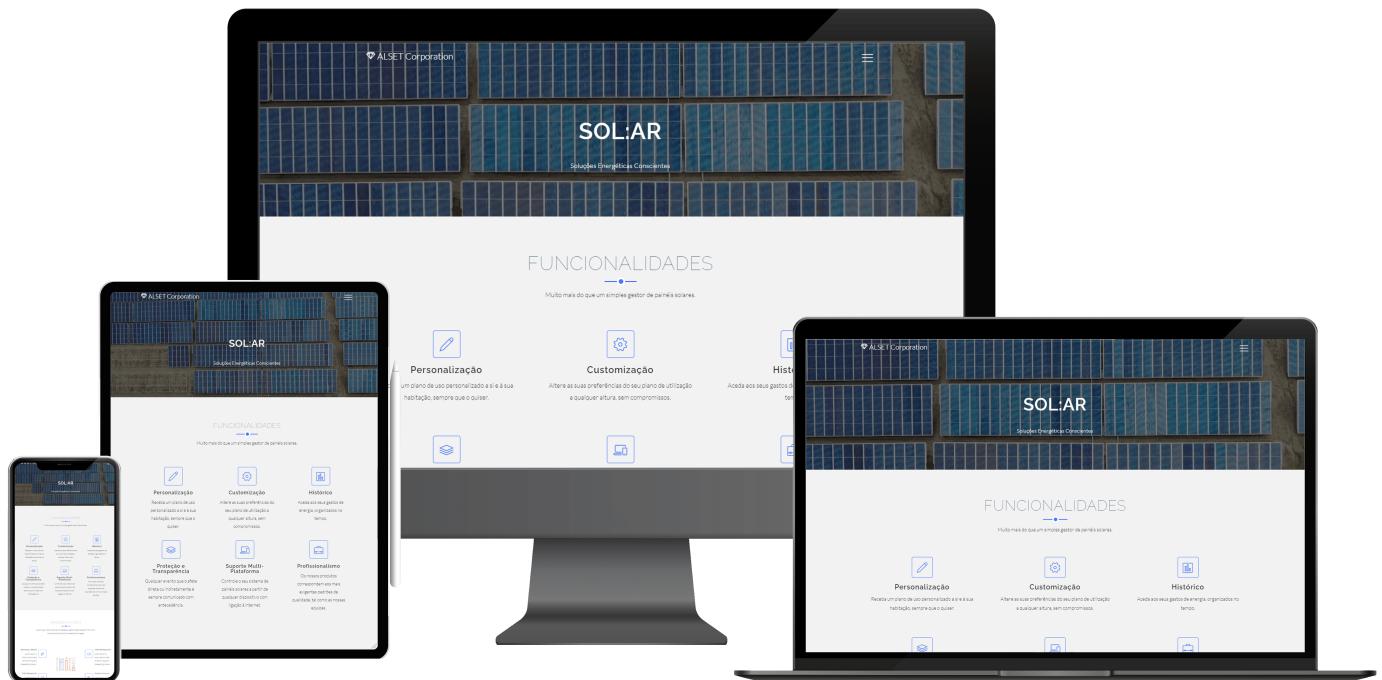


Figura 2.4: Mockup da *landing page* da aplicação

2.6 Definição de um conjunto de medidas de sucesso

Dado que o nosso software apenas irá existir enquanto site, não poderemos usar, por exemplo, o número de downloads como métrica de sucesso, no entanto poderemos utilizar o número de acessos diárias para avaliar se a *SOL:AR* foi bem recebida pelo público.

Outra métrica que também poderemos usar é o número de contas ativas, isto é, contas que foram acedidas nas últimas 72 horas, por exemplo.

Queremos também criar uma aplicação que seja simples e fácil de interagir, seja o utilizador alguém familiarizado com as novas tecnologias ou não. É também fundamental que a nossa aplicação tenha o mínimo número de bugs possível, sendo que idealmente nem os tenha. Outro aspeto em que não poderemos facilitar é na eficiência e na velocidade da aplicação, dado que esse é um fator cada vez mais crucial.

2.7 Plano de desenvolvimento (diagrama GANTT)

SO:LAR

Laboratórios de Informática IV
Uminho

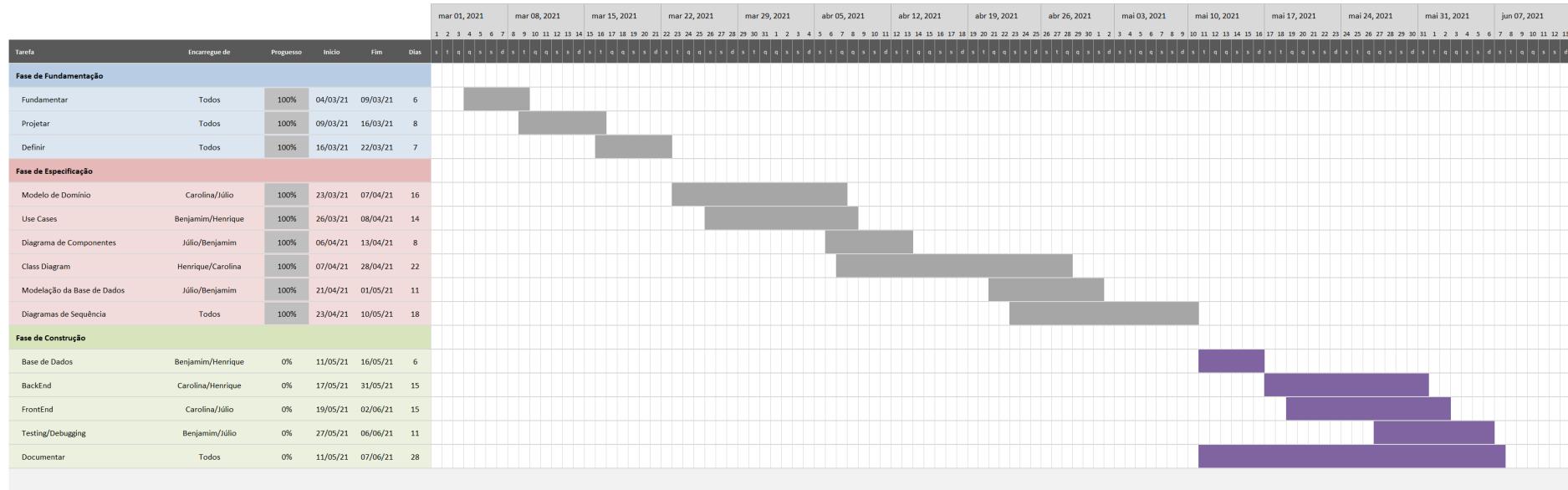


Figura 2.5: Diagrama de GANTT do plano de desenvolvimento

3 Levantamento e Análise de Requisitos

3.1 Tabela de Requisitos Funcionais

Requisitos Funcionais	Requisitos de utilizador	Requisitos do Sistema
Fazer registo de uma nova conta	O funcionário deve registar o utilizador na aplicação para este poder aceder às suas funcionalidades	O sistema deve pedir os dados do utilizador (email, nome, palavra passe, número de telemóvel, morada). O sistema não pode permitir o registo de contas com o mesmo email. O sistema deve armazenar os dados do utilizador na base de dados.
Aceder à dashboard	O utilizador deve estar autenticado. O utilizador tem de ser capaz de visualizar todas as informações relativas ao seu sistema de painéis solares	O sistema deve atualizar os consumos e produções energéticas diariamente e ser capaz de organizar essa informação de modo a mostrar a evolução temporal em gráficos. O sistema deverá manter o estado energético da habitação atualizado em tempo real (consumo, produção e bateria).
Ver todos os eventos	O utilizador deve estar autenticado. O utilizador tem de ser capaz de visualizar todos os eventos relevantes	O sistema deverá ser capaz de organizar todos os eventos relevantes para o dado utilizador
Ativar alerta para um dado evento	O utilizador deve estar autenticado e ficar com um lembrete ativo para o dado evento	O sistema deverá ser capaz de guardar os alertas ativados por cada utilizador e enviar notificações na hora pretendida
Ver os lembretes ativados	O utilizador deve estar autenticado e ser capaz de visualizar todos os lembretes previamente ativados	O sistema deverá guardar eficientemente todos os lembretes ativados para esse utilizador

Requisitos Funcionais	Requisitos de utilizador	Requisitos do Sistema
Fazer login	O utilizador tem de inserir o nome de utilizador e palavra passe para se autenticar	O sistema deve verificar se os dados inseridos correspondem a algum utilizador na base de dados
Aceder às informações pessoais	O utilizador deve estar autenticado e ser capaz de visualizar todas as informações pessoais	O sistema deverá ser capaz de organizar as informações pessoais do utilizador
Editar informações pessoais	O utilizador deve estar autenticado e ser capaz de editar todas as informações pessoais	O sistema deverá ser capaz de organizar as informações pessoais do utilizador
Consultar o histórico de reparações	O utilizador deve estar autenticado e ser um funcionário. O funcionário deve ser capaz de visualizar o histórico de reparações que realizou	O sistema deverá guardar o histórico de reparações e manutenções realizadas por cada funcionário
Eliminar alertas	O utilizador deve estar autenticado e ser capaz de eliminar os lembretes que ativou previamente	O sistema deve permitir alterações aos alertas já definidos pelo utilizador
Ver alertas de avarias	O utilizador deve estar autenticado e ser um funcionário. O funcionário deve ser capaz de visualizar todas as avarias que foram reportadas ao sistema	O sistema deve permitir ao Funcionário consultar a listagem de avarias que ainda não foram reparadas
Marcar horário de manutenção	O utilizador deve estar autenticado e ser um funcionário. O funcionário deve ser capaz de agendar manutenções	O sistema deve permitir ao Funcionário marcar o horário de manutenção para cada avaria
Ver distribuição dos painéis num mapa	O utilizador deve ser um funcionário e estar autenticado. O funcionário deverá ser capaz de visualizar a distribuição dos painéis solares num mapa interativo	O sistema tem de guardar toda a informação necessária para termos a localização exata de todos os painéis solares
Ver estado dos painéis solares	O utilizador deve estar autenticado e ser capaz de visualizar o estado atual de todos os painéis solares da sua habitação	O sistema deverá manter atualizado o estado de cada painel solar.
Ligar/desligar painel solar	O utilizador deve estar autenticado e ser capaz de ligar/desligar qualquer painel solar da sua habitação	O sistema deverá ser capaz de alterar o estado de cada painel
Comunicar avaria		O sistema deverá ser capaz de gerir as avarias reportadas pelos painéis solares
Comunicar Produção energética		O sistema deverá ser capaz de organizar a informação recebida dos painéis solares

3.2 Requisitos não funcionais

- O sistema deve ser simples de utilizar, com uma interface intuitiva
- O serviço deve estar online 24/7
- O sistema deve manter as informações sempre atualizadas
- A aplicação deve ser suportada por todos os browsers

4 Especificação em UML

4.1 Modelo de domínio

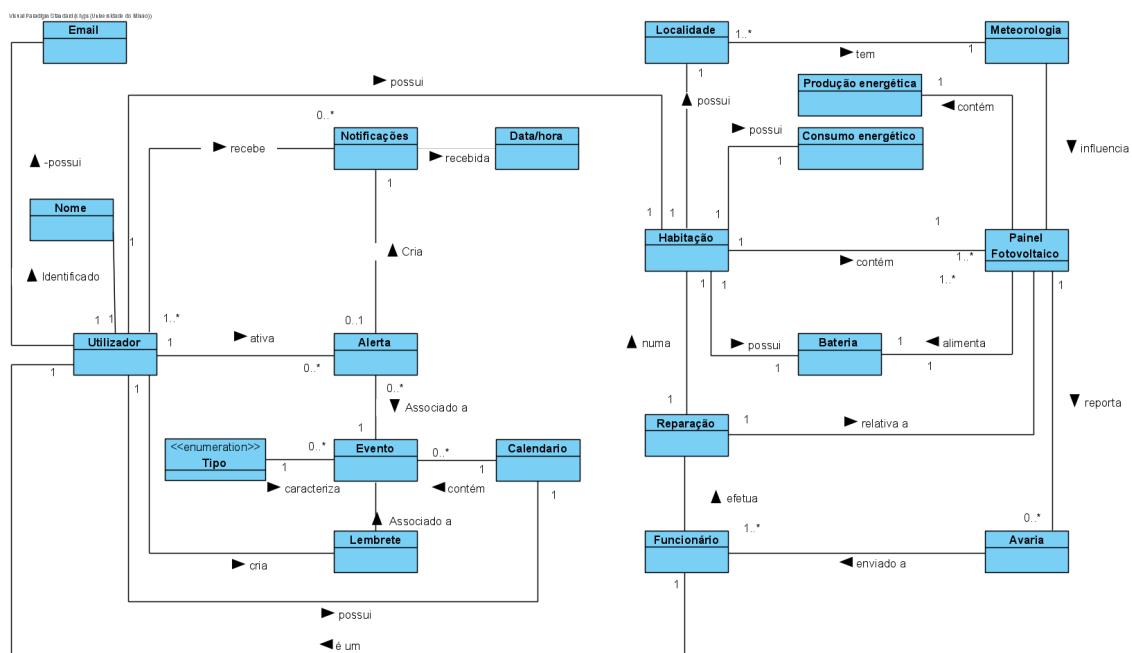


Figura 4.1: Modelo de Domínio

O modelo de domínio representa todo o sistema, nomeadamente as suas entidades, e as relações entre estas. Este esquema foi elaborado tendo em consideração todos os requisitos anteriormente estabelecidos.

Sendo um gestor de eventos, uma das entidades mais importantes desta aplicação é o Evento, que pode ter vários tipos, como por exemplo um evento meteorológico, uma avaria de um painel, ou um workshop da empresa.

O utilizador desta aplicação é também, evidentemente, uma entidade, e pode ser identificado pelo seu nome. O Utilizador interage com o sistema nomeadamente no modo em que pode activar alertas, que estão associados a eventos, para que seja alertado quando estes eventos acontecerem.

Uma variante do Utilizador é o Funcionário, cujas funções não só englobam as do Utilizador, como também lhe é permitido ver a distribuição num mapa dos painéis abrangidos pela aplicação, e também marcar manutenções para eventuais avarias.

Cada Utilizador possui também uma Habitação. Esta Habitação tem também a sua localização própria, o que significa que os painéis nela instalados serão afetados por eventos meteorológicos que atinjam a área.

Um painel solar é também uma entidade neste sistema. Possui funções limitadas, na medida em que apenas tem um estado atual (ligado/desligado) e uma taxa de produção energética como atributos, podendo interagir com o resto do sistema apenas através da comunicação de uma avaria.

Os Alertas anteriormente mencionados, quando ativados, criam Notificações, sendo estas as entidades que avisarão o Utilizador de eventos que se aproximam.

4.2 Diagrama de casos de uso

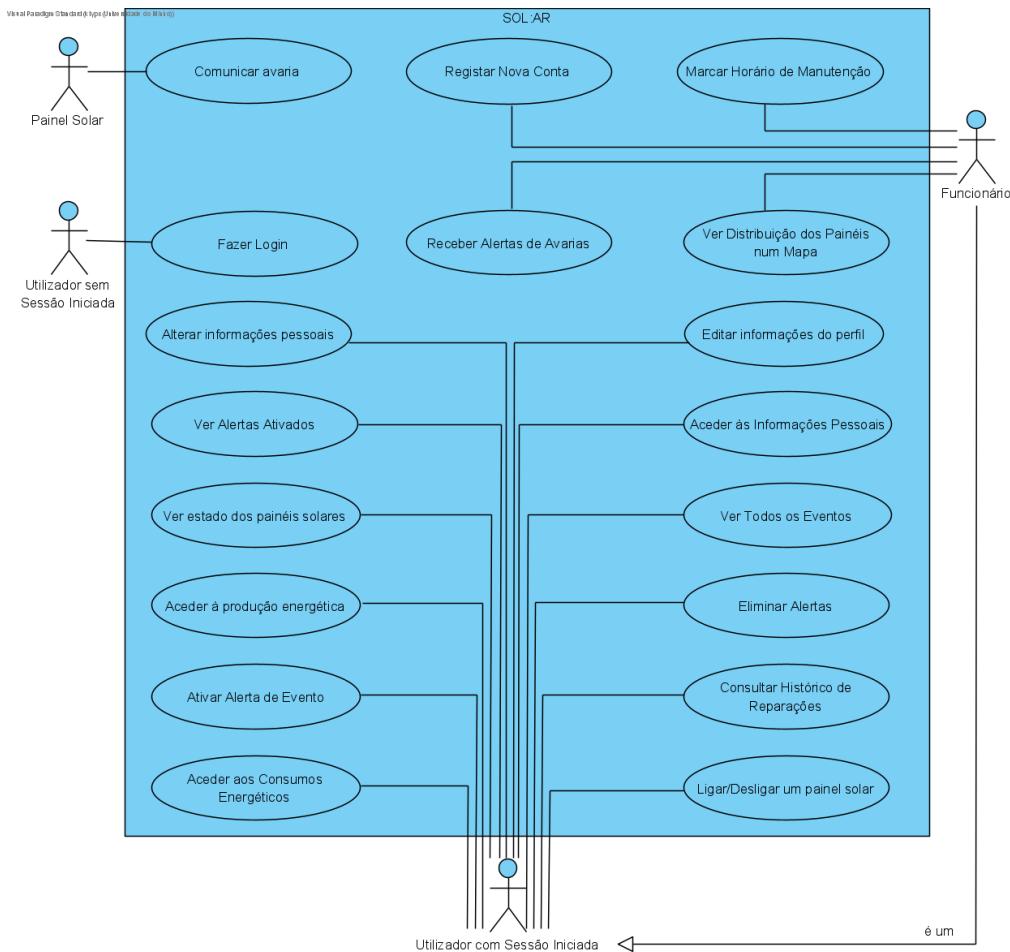


Figura 4.2: Diagrama de use cases

Os vários atores referenciados no diagrama permitem especificar os vários casos de uso associados às diversas funcionalidades da aplicação.

Estão especificados 4 atores fundamentais que interagem com o sistema. O primeiro é o Utilizador sem sessão iniciada, que corresponderia a uma pessoa que estivesse simplesmente a consultar as informações do Website ou a fazer autenticação. De seguida, temos o Utilizador com sessão iniciada, que efetua a maior parte dos casos de uso, quer estes de gestão energética, de eventos ou de informações pessoas. O ator funcionário é por sua vez uma extensão ao Utilizador, e consegue adicionalmente planear eventos de manutenção e gerir quer as contas de outros utilizadores, quer as habitações geridas pelo sistema. Por fim, o ator Painel Solar simplesmente limita-se a comunicar possíveis acontecimentos repentinos.

Indicamos também, no diagrama de Use Case, todas os casos de uso de cada ator. Todos estes casos de uso serão explicados em mais detalhe na secção seguinte.

4.3 Especificação dos Casos de Uso

4.3.1 Registar nova conta

Atores envolvidos - Funcionário
<p>Pré-condição: Funcionário recebe pedido de nova conta Pós-condição: O sistema fica com um registo da nova conta Descrição: Funcionário regista novo utilizador</p>
<p>Fluxo normal: 1- Funcionário insere no sistema as informações para criar a nova conta 2- Sistema regista nova conta</p>
<p>Fluxo de Exceção 1 [Utilizador cancela criação de conta] (passo 2) 2.1- Sistema informa o funcionario que a operação falhou</p>
<p>Fluxo de Exceção 2 [Utilizador já tem conta no Sistema] (passo 2) 2.1- Sistema informa o funcionario que a conta já existe</p>

4.3.2 Efetuar Login

Atores envolvidos: Utilizador não autenticado
<p>Pré-condição: true Pós-condição: O Utilizador fica autenticado no sistema Descrição: Um Utilizador inicia sessão no Sistema</p>
<p>Fluxo Normal: 1-Utilizador insere o nome de utilizador e password 2-Sistema valida os dados inseridos 3- Utilizador é redirecionado para a página inicial com sessão iniciada</p>
<p>Fluxo Exceção [Dados inválidos] (passo 2): 2.1-Sistema informa que os dados não são válidos</p>

4.3.3 Aceder aos consumos energéticos

Atores envolvidos: Utilizador autenticado
<p>Pré condição: Utilizador estar autenticado e haver registos sobre o consumo energético Pós-condição: Utilizador tem acesso aos consumos energéticos em gráfico Descrição: Visualizar consumos energéticos em gráfico</p>
<p>Fluxo Normal:</p> <ul style="list-style-type: none">1- Utilizador acede à dashboard2- Sistema apresenta os consumos

4.3.4 Aceder à produção energética

Atores envolvidos: Utilizador autenticado
<p>Pré-condição: Utilizador tem de estar autenticado Pós-condição: Utilizador tem acesso à produção energética da sua habitação Descrição: Mostrar num gráfico a produção energética da habitação do utilizador</p>
<p>Fluxo Normal:</p> <ul style="list-style-type: none">1- Utilizador acede à sua dashboard2- Sistema apresenta a produção

4.3.5 Ver todos os eventos

Atores envolvidos: Utilizador autenticado
<p>Pré-condição: utilizador estar autenticado Pós-condição: utilizador tem acesso a todos os eventos Descrição: Visualizar todos os eventos</p>
<p>Fluxo Normal:</p> <ul style="list-style-type: none">1- Utilizador acede à página de eventos2- Sistema apresenta todos os eventos relevantes ao dado utilizador

4.3.6 Ativar alerta para um evento

Atores envolvidos: Utilizador autenticado
<p>Pré-condição: Utilizador está autenticado e existe pelo menos um evento Pós-condição: O sistema fica com o alerta desse evento ativado para esse utilizador Descrição: Utilizador ativa um alerta para um dado evento</p>
<p>Fluxo Normal:</p> <p>1- Utilizador ativa o alerta para um evento 2 - O sistema regista o alerta</p>
<p>Fluxo Exceção [Esse alerta já foi ativado] (passo 1):</p> <p>1.1- Sistema indica ao utilizador que ele já ativou esse alerta</p>

4.3.7 Ver alertas ativados

Atores envolvidos: Utilizador autenticado
<p>Pré-condição: Utilizador está autenticado Pós-condição: Utilizador consultou os alertas Descrição: Utilizador consulta os seus alertas</p>
<p>Fluxo Normal:</p> <p>1-Utilizador acede à sua página pessoal 2- Sistema apresenta todos os alertas ao Utilizador</p>

4.3.8 Ver histórico de reparações

Atores envolvidos: Utilizador autenticado
<p>Pré- condição: Utilizador estar autenticado Pós-condição: Utilizador consultou histórico de reparações Descrição: Utilizador consulta histórico de reparações aos seus painéis</p>
<p>Fluxo Normal:</p> <p>1- Utilizador acede à página de reparações 2- Sistema apresenta histórico de reparações aos painéis pertencentes ao utilizador</p>

4.3.9 Ver informações pessoais

Atores envolvidos: Utilizador autenticado
<p>Pré-condição: Utilizador está autenticado Pós-condição: Utilizador consultou as suas informações pessoais Descrição: Utilizador consulta as suas informações pessoais</p>
<p>Fluxo Normal:</p> <p>1-Utilizador acede à sua página pessoal 2- Sistema apresenta as informações pessoais ao Utilizador</p>

4.3.10 Editar informações pessoais

Atores envolvidos: Utilizador autenticado
<p>Pré-condição: Utilizador está autenticado Pós-condição: Utilizador editou as suas informações pessoais Descrição: Utilizador edita as suas informações pessoais</p>
<p>Fluxo Normal:</p> <p>1-Utilizador acede à sua página pessoal 2- Sistema apresenta as informações pessoais ao Utilizador 3- Utilizador edita as informações pessoais que necessita de editar</p>

4.3.11 Eliminar alerta

Atores envolvidos: Utilizador autenticado
<p>Pré-condição: Consultou os seus alertas Pós-condição: Alerta foi eliminado Descrição: Utilizador um dos seus alertas</p>
<p>Fluxo normal:</p> <p>1- Utilizador seleciona a opção de eliminação no alerta pretendido 2- Sistema elimina o alerta</p>

4.3.12 Ver alertas de avarias

Atores envolvidos: Funcionário
<p>Pré-condição: Painel solar reportou uma avaria Pós-condição: Funcionário recebe notificação Descrição: Funcionário recebe alerta de avaria</p>
<p>Fluxo Normal:</p> <p>1- Sistema envia o alerta aos funcionários 2- Sistema desliga o Painel Solar</p>

4.3.13 Ver estado dos painéis solares

Atores envolvidos: Funcionário e utilizador autenticado
<p>Pré-Condição: Utilizador está autenticado Pós-Condição: Utilizador sabe o estado atual de funcionamento do painel Descrição: Visualizar o estado atual de funcionamento do painel</p>
<p>Fluxo Normal:</p> <p>1- Utilizador acede à dashboard 2- Sistema apresenta ao utilizador o estado atual de funcionamento dos painéis</p>

4.3.14 Marcar horário de manutenção

Atores envolvidos: Funcionário
<p>Pré-condição: Funcionário consultou a lista de avarias Pós-condição: Horário de manutenção foi marcado Descrição: Funcionário marca horário de manutenção para uma avaria</p>
<p>Fuxo Normal:</p> <ul style="list-style-type: none">1- Funcionário seleciona avaria a tratar2- Funcionário indica o horário de manutenção3- Sistema regista o horário de manutenção
<p>Fluxo de Exceção [O funcionário já tem marcações naquele horário]:</p> <ul style="list-style-type: none">3.1 - Sistema informa o funcinário que já existe uma marcação naquele horário

4.3.15 Ver distribuição dos painéis num mapa

Atores envolvidos: Funcionário
<p>Pré-condição: Funcionário estar autenticado Pós-condição: Funcionário tem acesso à distribuição de painéis num mapa Descrição: O funcionário visualiza a distribuição de painéis no mapa</p>
<p>Fuxo Normal:</p> <ul style="list-style-type: none">1- Funcionário acede à area do mapa2- Funcionário visualiza o mapa com todos os painéis sinalizados no mesmo
<p>Fluxo de Exceção [O funcionário já tem marcações naquele horário]:</p> <ul style="list-style-type: none">3.1 - Sistema informa o funcinário que já existe uma marcação naquele horário

4.3.16 Ligar/Desligar um painel solar

Atores envolvidos: Utilizador autenticado
<p>Pré-condição: Utilizador acedeu às suas informações pessoais Pós-condição: O estado do Painel Solar foi alterado Descrição: O Utilizador altera o estado do painel solar</p>
<p>Fluxo Normal:</p> <p>1- Utilizador seleciona a opção de alterar o estado do painel solar 2- Sistema regista alteração do estado do painel</p>
<p>Fluxo de Exceção [Painel Solar está avariado] (passo 1):</p> <p>2.1- Sistema informa o ator que o painel está avariado</p>

4.3.17 Comunicar Avaria

Atores envolvidos: Painel Solar
<p>Pré-condição: true Pós-condição: Sistema fica com um registo da avaria no painel solar Descrição: Painel solar deteta avaria e comunica ao sistema</p>
<p>Fluxo Normal:</p> <p>1- Painel Solar comunica avaria ao Sistema 2- Sistema regista avaria 3- Sistema desliga o Painel Solar</p>

4.3.18 Comunicar produção energética

Atores envolvidos: Painel Solar

Pré-condição: true

Pós-condição: Sistema fica com o registo da produção energética desse dia

Descrição: Painel solar envia informação da produção energética diária ao sistema

Fluxo Normal:

1- Painel Solar comunica a produção energética diária ao sistema

2- Sistema regista a produção energética desse dia

4.4 Arquitetura da Aplicação

4.4.1 Diagrama de componentes

Para garantir uma boa implementação e escalabilidade no desenvolvimento futuro da aplicação, o sistema foi separado em vários componentes distintos como é visível na figura 4.3.

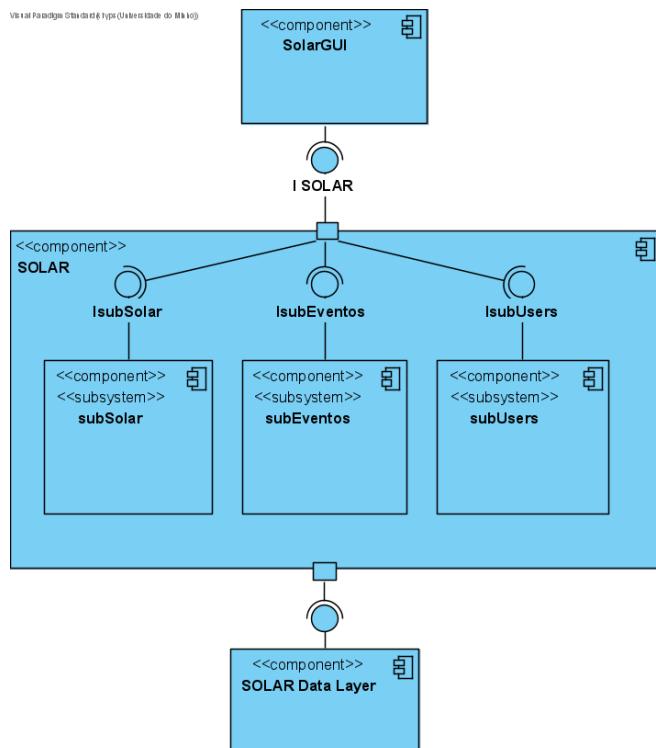


Figura 4.3: Diagrama de componentes

O componente mais superficial na hierarquia (*SolarGUI*) corresponde à interface gráfica de utilizador e é responsável pela interação com o utilizador e desta forma lê e apresenta informação ao mesmo.

De seguida o componente *SOLAR* que implementa a interface *ISOLAR* é responsável pela lógica de negócio do programa sendo constituída por três subsistemas. O subsistema *subSolar*, tem o dever de gerir as várias habitações e respetivos painéis fotovoltaicos. De seguida, o subsistema *subEventos* responsabiliza-se pelos eventos, lembretes e alertas existentes numa aplicação como esta. Por fim, o subsistema *subUsers* implementa as funcionalidades exigidas para a gestão dos clientes e funcionários. Estes subsistemas implementam respetivamente as interfaces *IsubSolar*, *IsubEventos* e *IsubUsers* permitindo assim um melhor controlo das suas tarefas.

Finalmente, o ultimo componente na hierarquia (*Solar Data Layer*) é responsável pela permanência e gestão dos dados.

4.4.2 Diagrama de classes

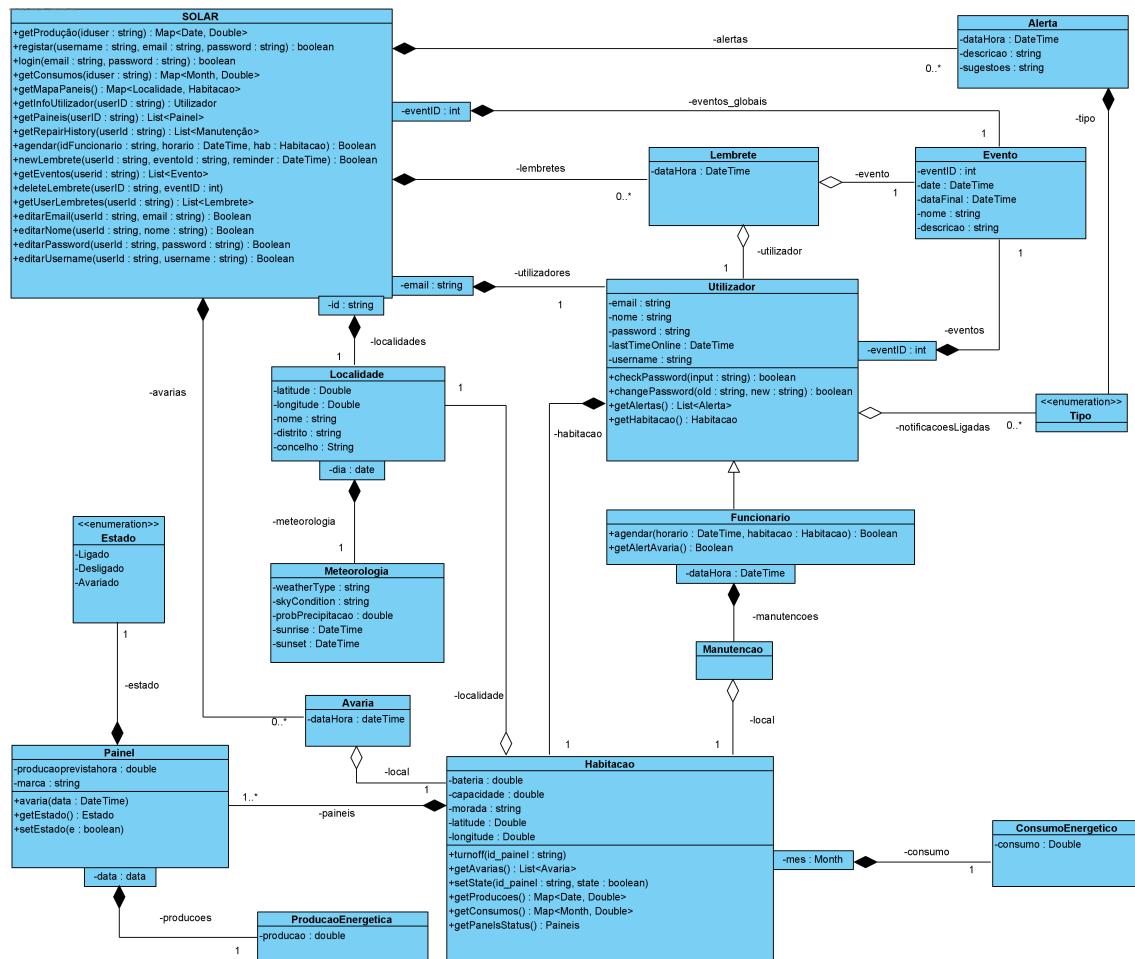
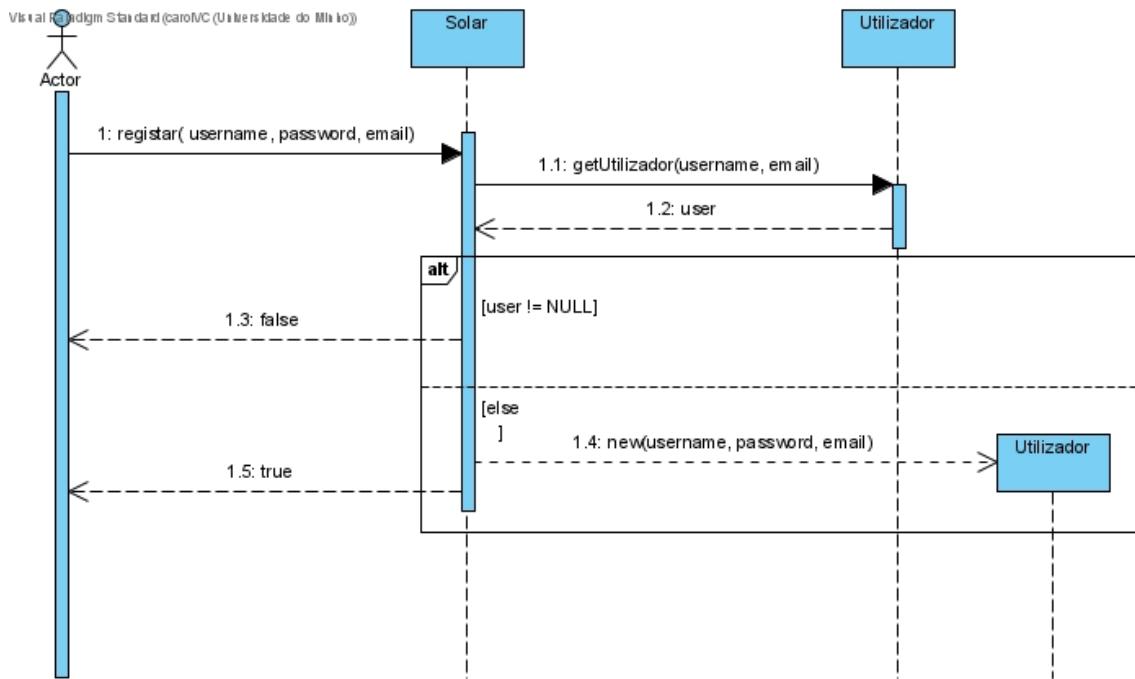


Figura 4.4: Diagrama de classes

Como classe principal do nosso programa, temos a classe *Solar* que é responsável por gerir todo o programa. Depois decidimos implementar as classes necessárias para o *Utilizador*, *Habitação*, *Painel*, *Meteorologia*, *Localidade*, *Evento*, *Alerta*, *ConsumoEnergetico*, *ProducaoEnergetica*, *Avaria*, *Funcionario* e *Lembrete*. Todas estas classes são responsáveis pelas entidades com o mesmo nome. Neste diagrama também podemos ver que existem *maps*, como por exemplo na classe *Habitacao* que possui um *map* de *ConsumoEnergetico* com a chave *Month* para representar os consumos energéticos por mês. Também, para além dos *maps*, temos *lists* como no caso da classe *Habitacao* que possui uma lista de objetos *Painel*.

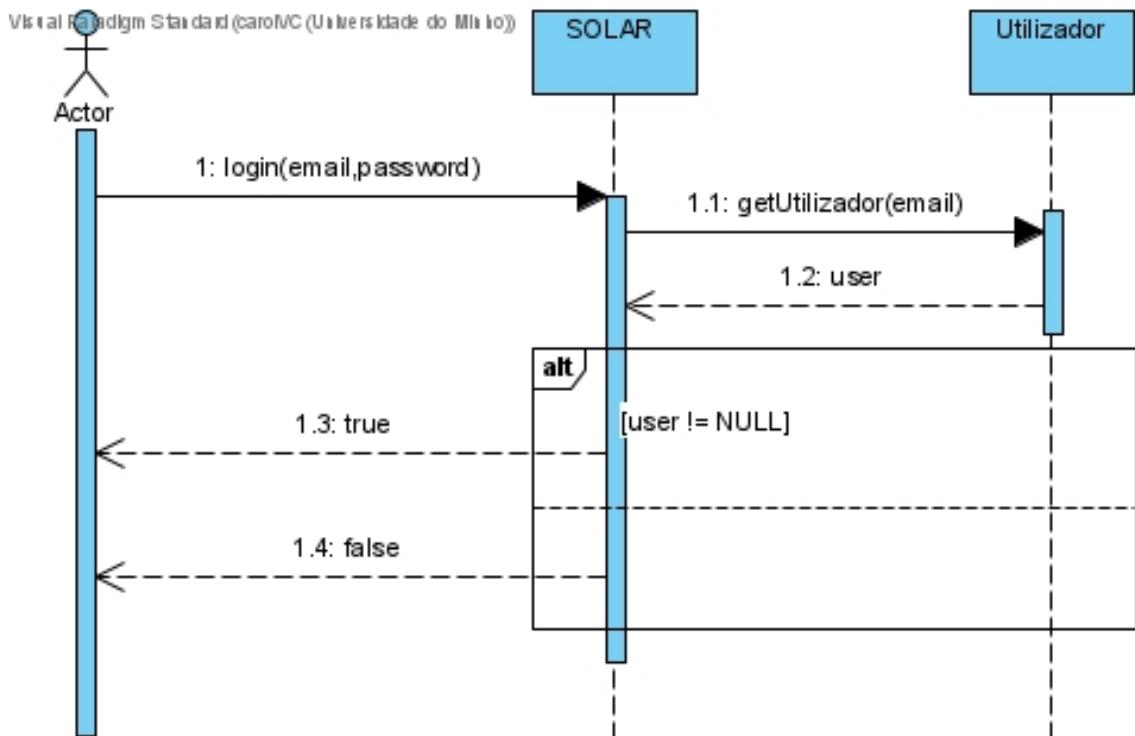
4.5 Diagramas de sequência

4.5.1 Registar Nova Conta



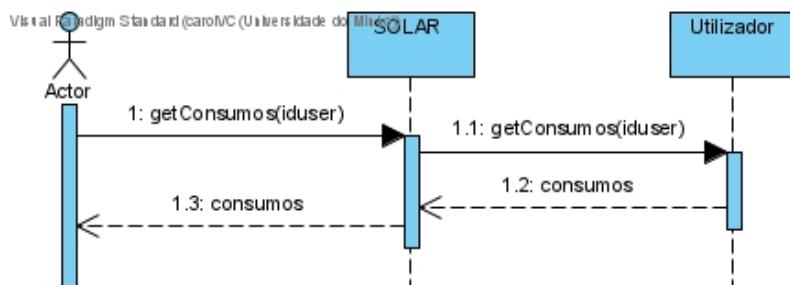
Para registar uma nova conta, utiliza-se o método *registrar(username,password,email)* na *Solar*. De seguida, a *SOLAR* encarrega-se de ir a todos os *Utilizador* e verificar se não existe um *Utilizador* que não possua um username ou um email iguais aos fornecidos pelo utilizador. Caso exista, o método retorna uma mensagem de erro e caso não exista, utiliza-se o método *new(username,password,email)* para criar um novo *Utilizador*.

4.5.2 Efetuar Login



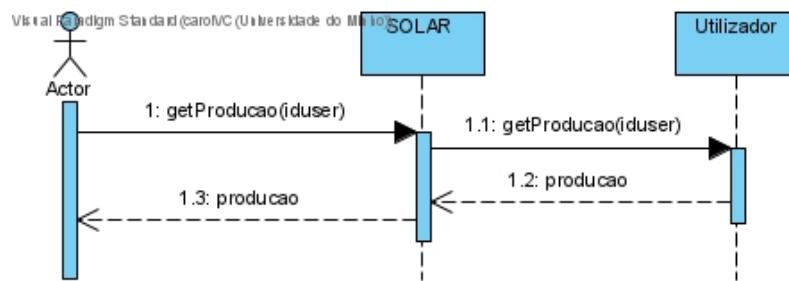
Para efetuar login, utiliza-se o método *login(email,password)* que é chamado pela *SOLAR*. Esta irá buscar o *Utilizador* que possui esse email. Caso encontre o *Utilizador*, devolve o *Utilizador* pretendido e caso não encontre retornará uma mensagem de erro.

4.5.3 Aceder aos consumos energéticos



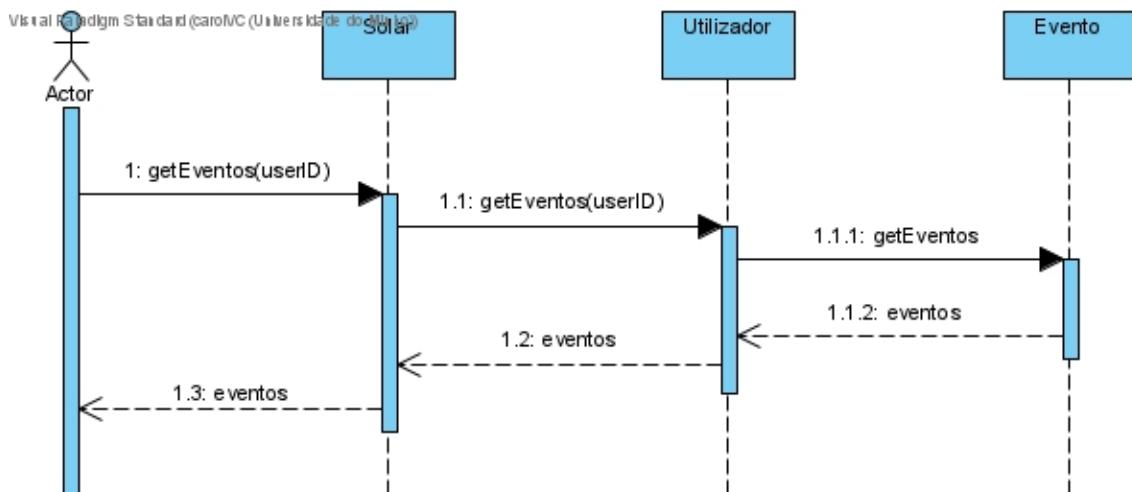
Para aceder aos consumos energéticos, recorre-se ao método *getConsumos(iduser)*, com o id do utilizador cujos consumos pretendemos saber como argumento. Esse argumento é chamado pela *SOLAR* e depois é novamente chamado pelo *Utilizador* que devolverá os consumos do utilizador indicado.

4.5.4 Aceder à produção energética



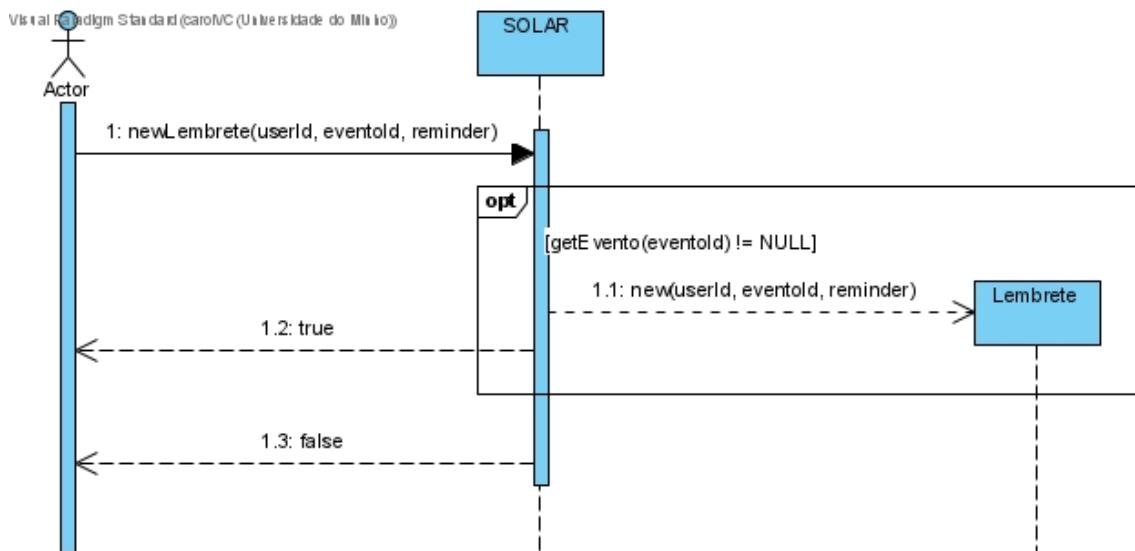
Para aceder à produção energética, recorre-se ao método *getProducao(iduser)*, com o id do utilizador cuja produção pretendemos saber como argumento. Esse argumento é chamado pela *SOLAR* e depois é novamente chamado pelo *Utilizador* que devolverá a produção do utilizador indicado.

4.5.5 Ver todos os eventos



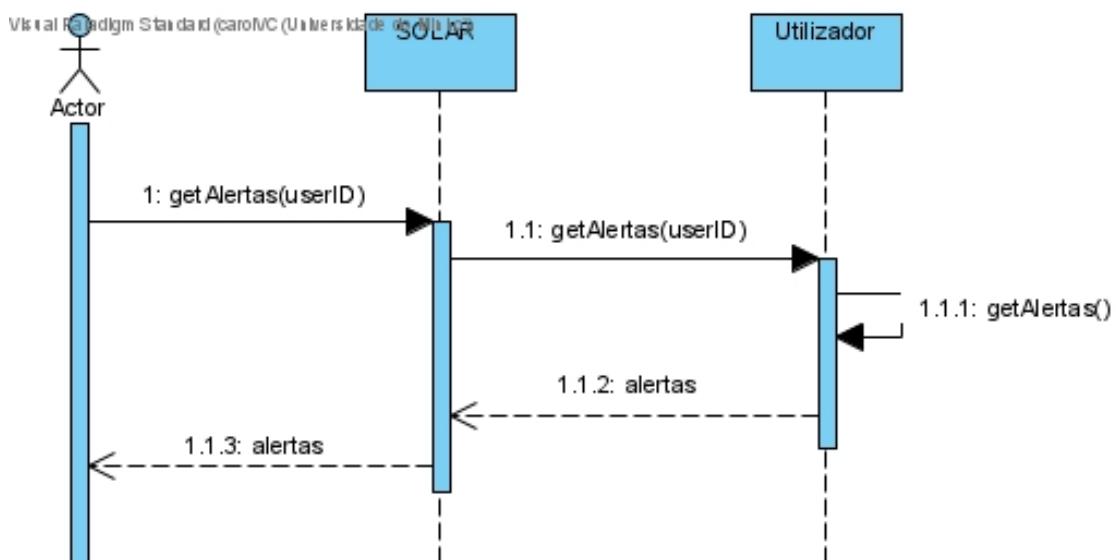
Para consultar todos os eventos existentes no sistema, recorre-se ao método *getEventos(userID)*, que é chamado pelo *SOLAR*, que passará pelo *Utilizador* pretendido, que também chamará o método *getEventos* que acederá ao *map* com todos os eventos que esse *Utilizador* possui e devolverá ao sistema todos os eventos existentes.

4.5.6 Ativar lembrete para um evento



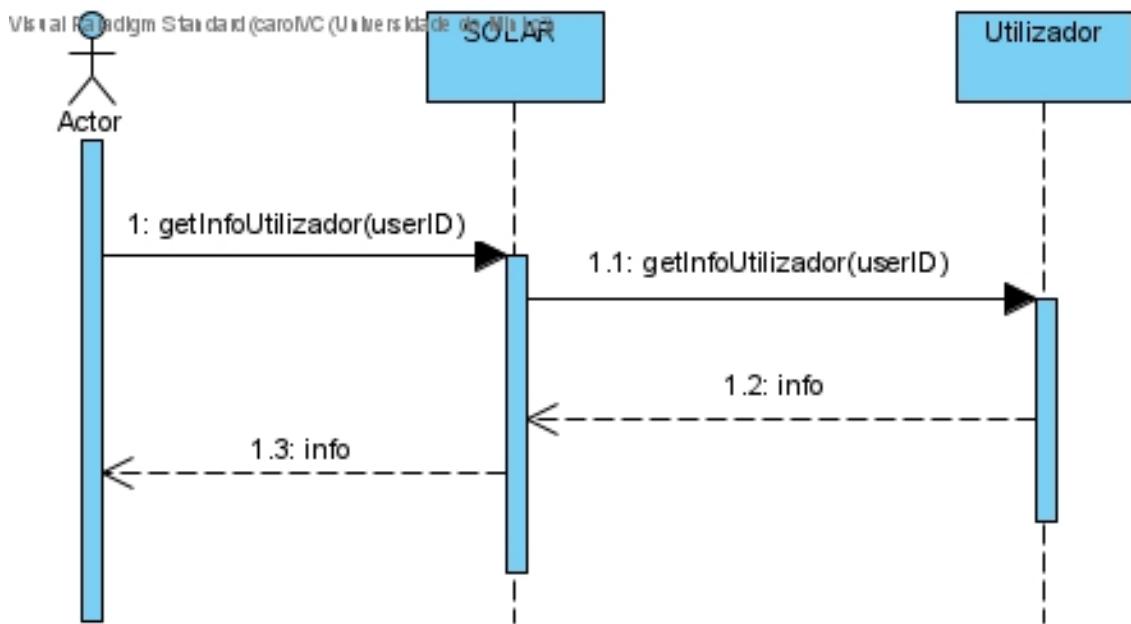
Se o utilizador pretender ativar um lembrete para um evento, o sistema chamará o método *newLembrete(userID,eventoID,reminder)*. Se existir um evento com o id igual a eventoID, então, através do método *new(userID,eventoID,reminder)*, é criado um lembrete para o utilizador.

4.5.7 Ver alertas ativados



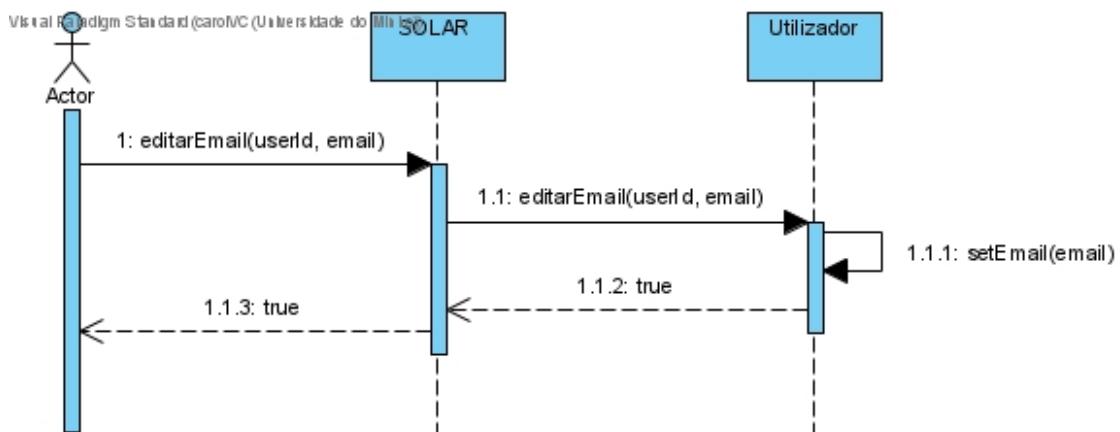
Para consultar os alertas que um utilizador possui, utiliza-se o método `getAlertas(userid)`, que será chamado pelo *Solar* e o *Utilizador* utilizará o método `getAlertas()` que devolverá todos os alertas que determinado utilizador possui.

4.5.8 Ver informações pessoais



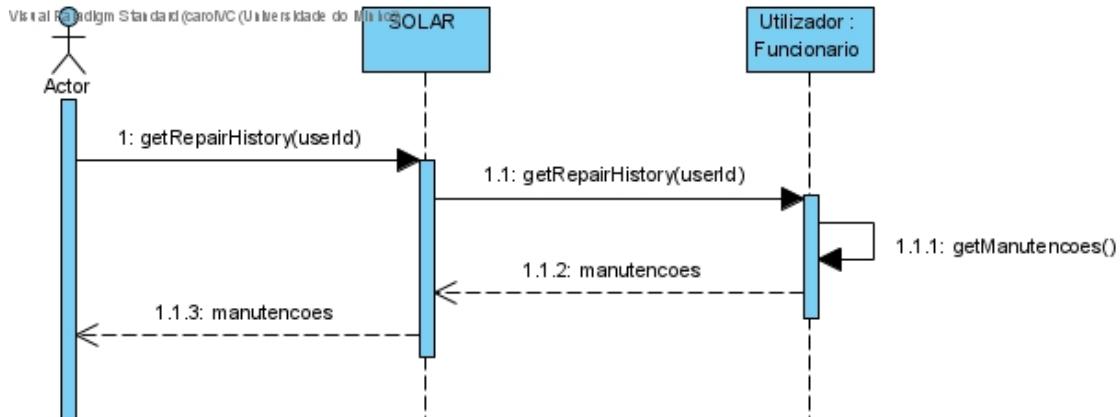
Para poder ver as informações pessoais de um utilizador, utiliza-se o método `getInfoUtilizador(userID)`, que será chamado pela *SOLAR* e de seguida pelo *Utilizador* que devolverá a informação do *Utilizador* em questão.

4.5.9 Editar informações do perfil



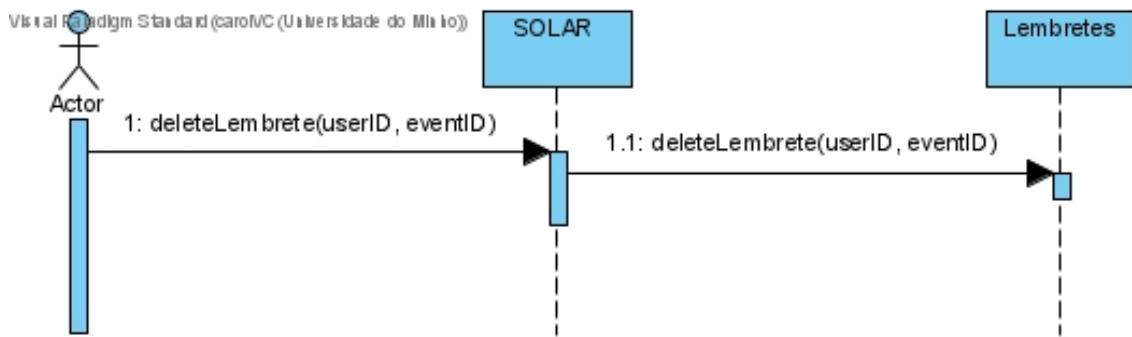
A edição das outras informações do Utilizador (username, password, nome, e morada) serão feitas de modo equivalente.

4.5.10 Ver histórico de reparações



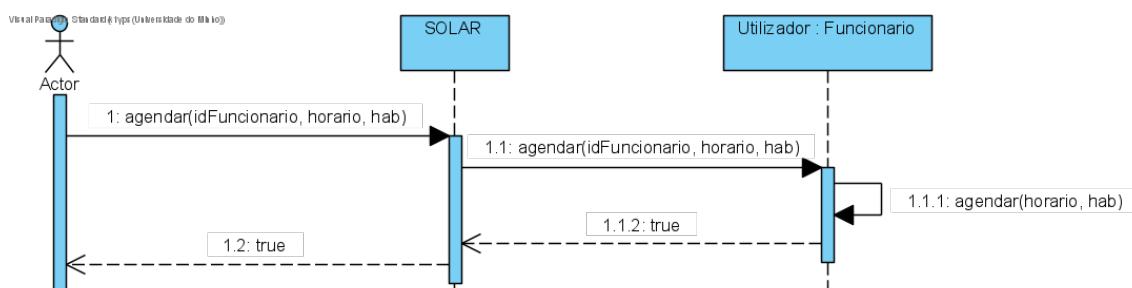
Para consultar o histórico de reparações, utiliza-se o método `getRepairHistory(userID)`. De seguida, para se obter o utilizador em questão, recorre-se ao método `getUser(userID)`, que devolverá o utilizador pretendido

4.5.11 Eliminar lembrete



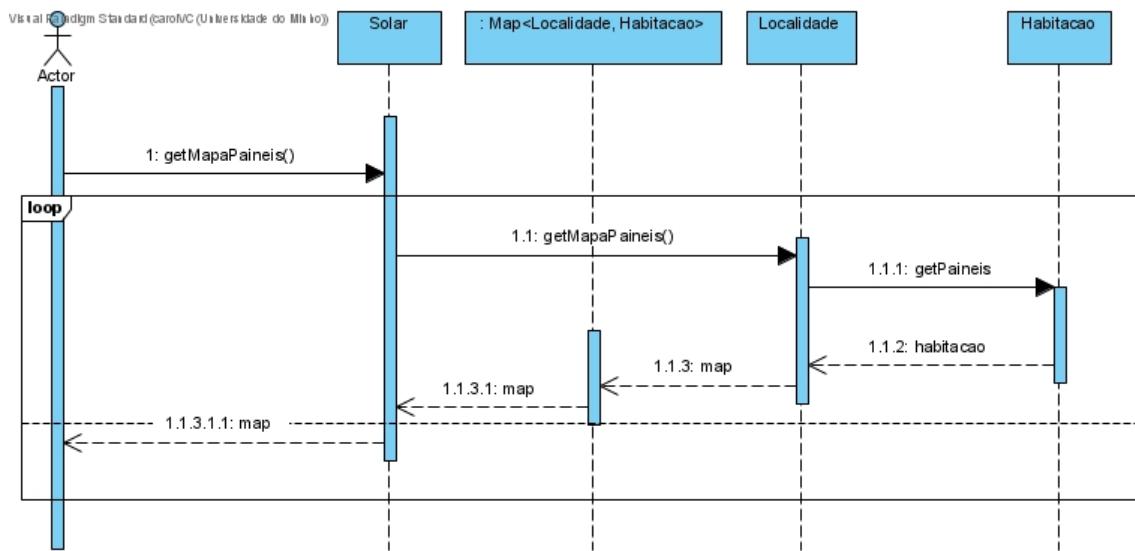
Para o utilizador eliminar um lembrete, utiliza-se o método *deleteLembrete(userID, eventID)*, que será chamado pela *SOLAR* e de seguida por *Lembrete* que se encarregará de eliminar o lembrete em questão.

4.5.12 Marcar horário de manutenção



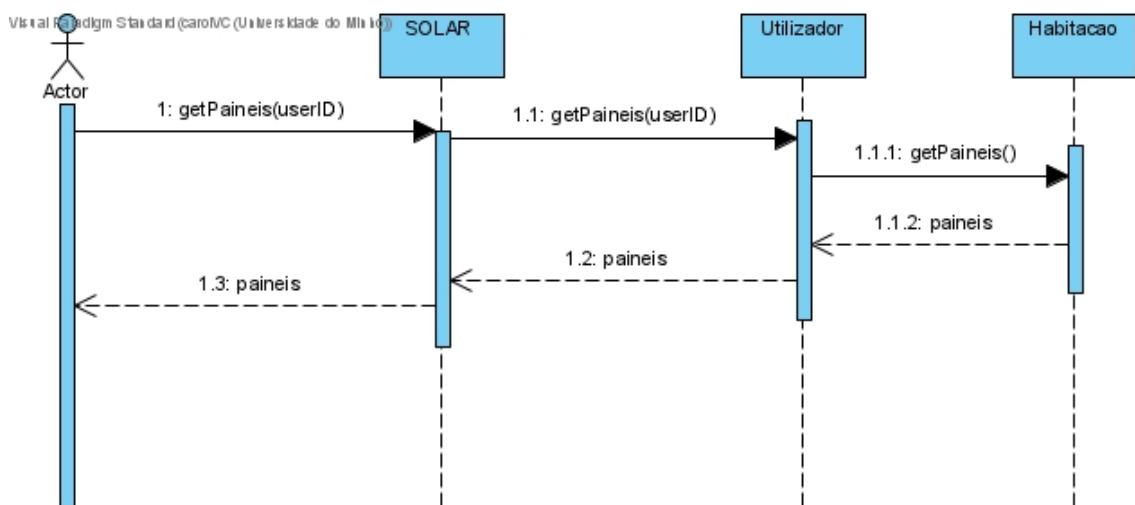
Para o funcionário marcar um horário de manutenção, recorre-se ao método *agendar(idFuncionario, horario, hab)*, que será chamado pela *SOLAR* e em seguida pelo *Utilizador:Funcionario*. Este, depois de obter o funcionário em questão, utiliza o método *agendar(horario, hab)* para, finalmente, marcar um horário de manutenção.

4.5.13 Ver distribuição dos painéis num mapa



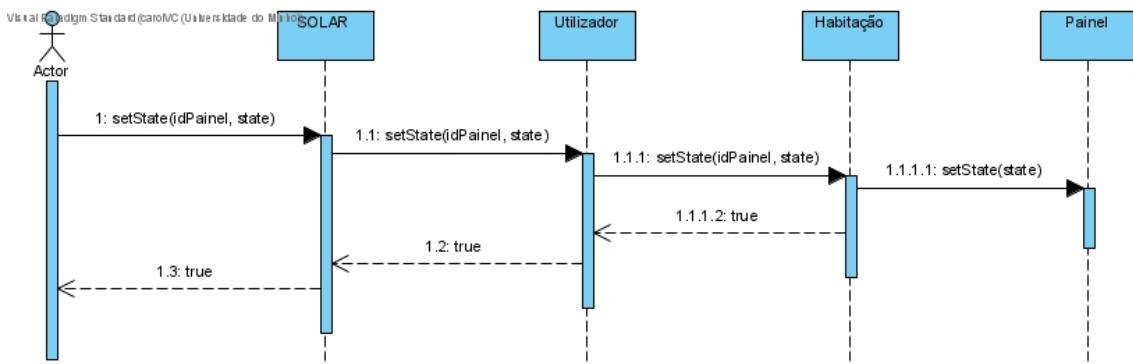
De forma a se obter a distribuição de painéis num mapa, utiliza-se o método *getMapaPaineis()*, que será chamado pela *SOLAR*. Aqui, entrará num loop de forma a percorrer todas as localidades e habitações. Após a chamada de *getMapaPaineis()* na *SOLAR*, este método é novamente chamado por *Localidade*, que irá a cada *Habitacao*, onde se irá chamar o método *getPaineis()*, que devolverá todas as habitações com painéis.

4.5.14 Ver estado dos painéis solares



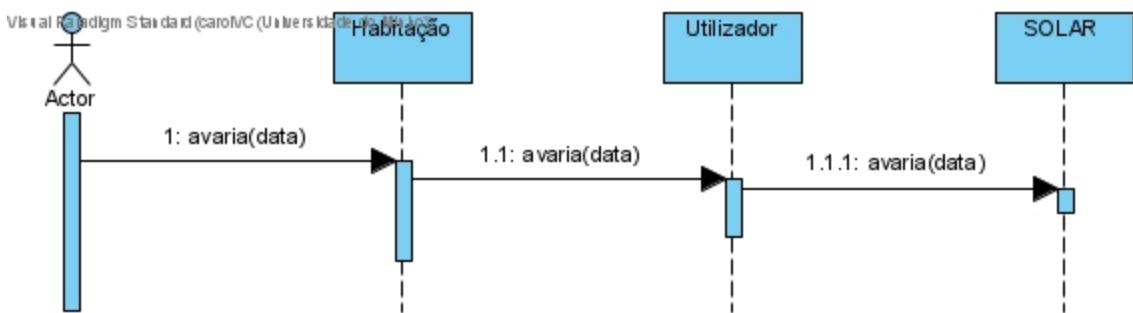
Para o utilizador saber o estado dos seus painéis solares, utilizamos o método `getPaineis(userID)`, que será chamado consecutivamente pela *SOLAR* e pelo *Utilizador* indicado. De seguida, *Habitacao* chamará o método `getPaineis()` que devolverá todos os painéis que a *Habitacao* possui e consequentemente poderemos saber o estado dos mesmos.

4.5.15 Ligar/desligar um painel solar



Quando se pretender ligar ou desligar um painel solar, utiliza-se o método `setState(idPainel, state)`. Este será chamado pela *SOLAR* e em seguida pelo *Utilizador* e pela *Habitacao*. Por fim, quando chegar ao *Painel* que se pretende, utiliza-se o método `setState(state)` que por fim liga ou desliga o painel.

4.5.16 Comunicar avaria

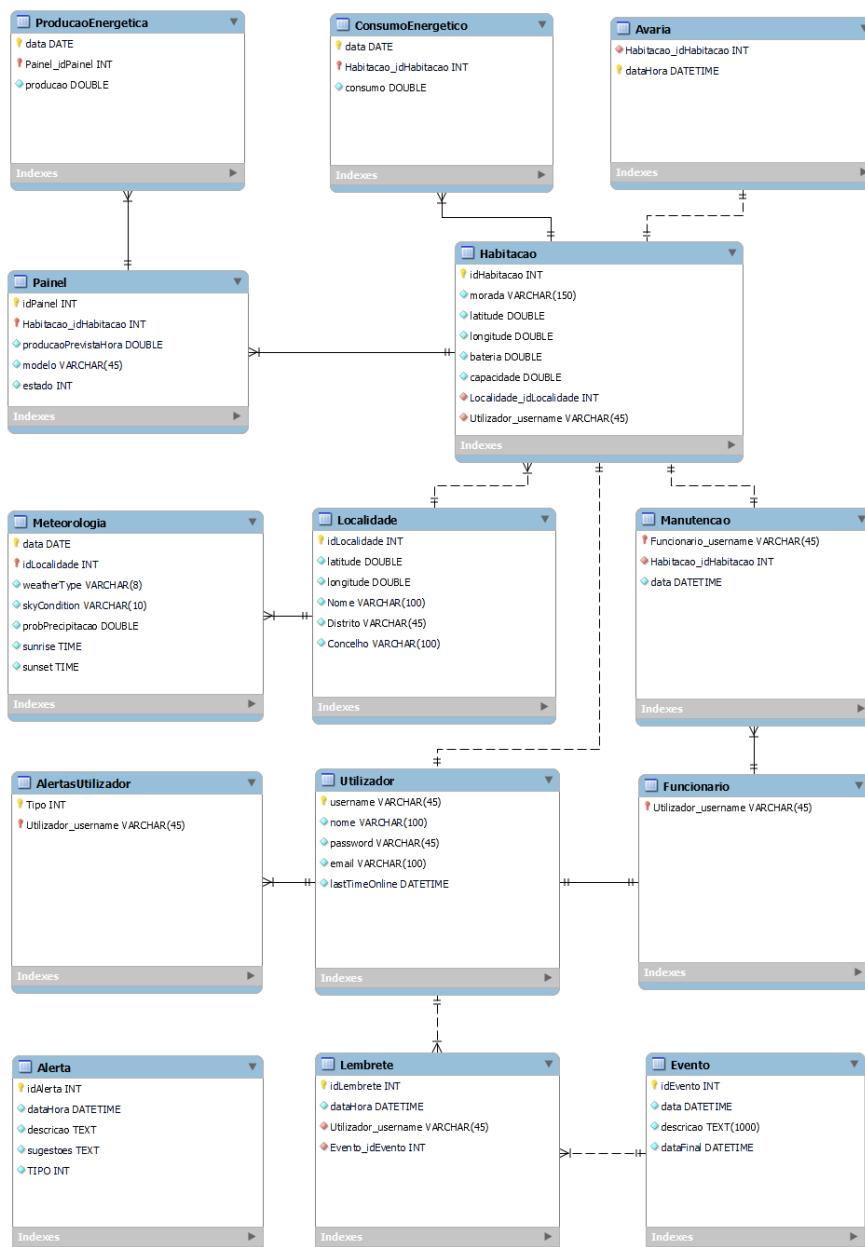


Quando um painel se avaria, este comunica para *Habitacao*, *Utilizador* e *SOLAR*, através de múltiplas chamadas de `avaria(data)` que comunicará a avaria e indicará a data em que a mesma foi detetada.

5 Base de dados

5.1 Modelo Lógico

A base de dados foi modelada com base no diagrama de classes da lógica de negócios (figura 4.4) e permite assim armazenar a informação gerida sem redundâncias e erros.



5.2 Dicionário de Dados

Entidade	Atributos	Tipo de Dados	Descrição	Exemplo
Painel	idPainel idHabitacao producaoPrevista modelo estado	INT INT DOUBLE VARCHAR(45) INT	Chave primária Chave estrangeira de Habitacao Produção energética prevista Modelo do painel Estado atual do painel	1 1 200.5 SOLAR:1 1
ProducaoEnergetica	data idPainel producao	DATE INT DOUBLE	Chave primária Chave estrangeira de Painel Produção Energética desse dia	12/06/2021 1 40.23
ConsumoEnergetico	data idHabitacao consumo	DATE INT DOUBLE	Chave primária Chave estrangeira de Habitacao Consumo energético desse dia	15/08/2021 3 301.15
Habitacao	idHabitacao morada latitude longitude bateria capacidade idLocalidade Utilizador_username	INT VARCHAR(150) DOUBLE DOUBLE DOUBLE DOUBLE INT VARCHAR(45)	Chave Primária Morada da habitação Coordenada latitude Coordenada longitude Energia armazenada Capacidade de armazenamento Chave estrangeira de Localidade Chave estrangeira do proprietário	10 Rua Feliz 120.0123 1124.5221 2540.21 5000 18 jose011
Evento	idEvento data descricao	INT DATETIME TEXT(1000)	Chave primária Data do evento Descrição do evento	1 12/06/2021 Tempestade aproxima-se
Manutencao	Funcionario_username idHabitacao data	VARCHAR(45) INT DATE	Chave primária Chave estrangeira de Habitacao Data agendada	joao1720 23 02/04/2021
Avaria	idHabitacao dataHora	INT DATETIME	Id da Habitação com painel avariado Data e hora da avaria	2 20-02-2021 12:00:00

Localidade	idLocalidade latitude longitude Nome Distrito Concelho	INT DOUBLE DOUBLE VARCHAR(100) VARCHAR(45) VARCHAR(100)	Chave Primária Coordenada latitude Coordenada longitude Nome da localidade Nome do distrito Nome do concelho	33 591.24 5291.2300 Ferreira Porto Paços de Ferreira
Metereologia	data idlocalidade weatherType skyCondition probPrecipitacao sunrise sunset	DATE INT VARCHAR(8) VARCHAR(10) DOUBLE TIME TIME	Data do dia Chave estrangeira da localidade Tipo da meteorologia (METAR) Condição do céu (METAR) Probabilidade de precipitação Hora do amanhecer Hora do pôr do sol	20-02-2021 4590 DS (dust storm) CB (Cumulonimbus) 0.34 07:00:00 19:00:00
Utilizador	username nome password email lastTimeOnline	VARCHAR(45) VARCHAR(100) VARCHAR(45) VARCHAR(100) DATETIME	Nome de utilizador Nome do proprietário da conta Password do Utilizador Email do utilizador Data da última sessão do utilizador	jose011 José Almeida passwordSegura101 jose@gmail.com 01-01-2021 12:00:00
Alerta	idAlerta dataHora descricao sugestoes TIPO	INT DATETIME TEXT TEXT INT	Chave primária do alerta Data e hora do alerta Descrição do alerta Sugestões de ações para o utilizador Tipo do alerta	1 20-02-2021 12:00:00 Alerta mau tempo Desligar painéis solares 2
Lembrete	idLembrete dataHora Utilizador_username Evento_idEvento	INT DATETIME VARCHAR(45) INT	Chave primária Data e hora do lembrete Nome de utilizador Id do Evento	2 20-02-2021 12:00:00 jose011 3
Funcionario	Utilizador_username	VARCHAR(45)	Chave primária	joao1720
AlertasUtilizador	Tipo Utilizador_username	INT VARCHAR(45)	Chave Primária Chave estrangeira de Utilizador	Geral joaoze23

5.3 Espaço ocupado

Os valores foram calculados somando o número de bytes necessários para cada atributo.

Entidade	Tamanho (Bytes)
Painel	65
ProducaoEnergetica	15
ConsumoEnergetico	15
Habitacao	190
Evento	$20 + t$
Manutencao	57
Avaria	12
Localidade	265
Meteorologia	39
Utilizador	302
Alerta	$16 + 2*t$
Lembrete	61
Funcionario	45
AlertasUtilizador	49

Nota: A incógnita t representa o tamanho do atributo de tipo TEXT, cujo espaço ocupado depende do seu comprimento. Por exemplo, admitindo um tamanho de 100 caracteres, o Evento ocuparia 120 bytes, e o Alerta 216.

Assumindo que:

- Serão criados, anualmente , 365 registos dos consumos e produções energéticas diárias para cada utilizador.
- Anualmente, realizam-se em média 500 manutenções, 700 eventos, 30 lembretes por utilizador, 100 alertas e 365 registos para meteorologia.
- Existem, em média, 4 painéis solares, 1 habitação, 3 AlertasUtilizador e 1 localidade por utilizador.

Então, o espaço ocupado pelo base de dados, no final do primeiro ano da aplicação será:

- Funcionario e Utilizador : $100*302 + 10*45 = 30650$ bytes
- ProducaoEnergetica e ConsumoEnergetico : $(365*15 + 365*15) * 100 = 1095000$ bytes
- Alerta, Evento, Meteorologia, Lembrete e Manutencao: $500*57 + 700 * (20 + t) + 30 * 61 + 100 * 100 * (16 + 2*t) + 365 * 39 = 218565 + 2700t$ bytes
- Painel : $4*100*65=26000$ bytes

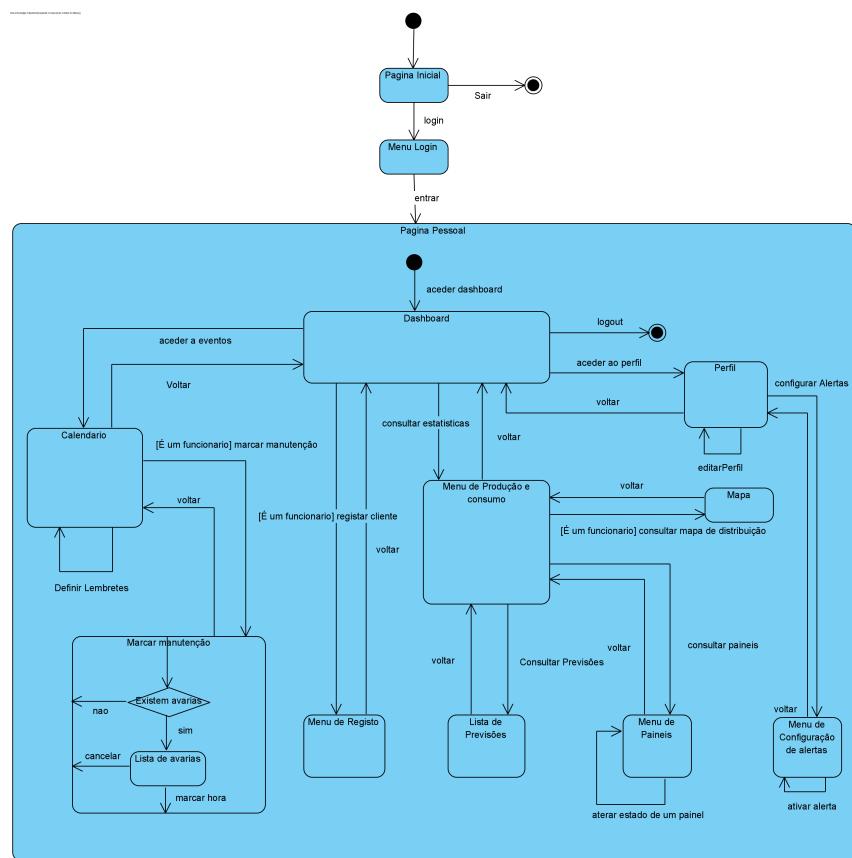
- Localidade e Habitacao: $100*190 + 100*265 = 45500$ bytes
- AlertasUtilizador: $3*100*49 = 14745$ bytes
- Total Anual: $1095000 + 218565 + 2700t + 26000 + 45500 + 30650 + 14745 = 1430460 + 2700t$ bytes

Assumindo que nos primeiros anos da aplicação adirão ao serviço 1000 utilizadores por ano, então o crescimento anual do espaço ocupado pela nossa base de dados será de $(1430460 + 2700t) * 10 = 14304600 + 27000t$ bytes.

Se considerarmos 500 como o valor médio de t, então: $14304600 + 27000 * 500 = 27\ 804\ 600$ bytes = 27,8 MBytes.

6 Definição da interface do sistema

6.1 Máquina de estado da interface do sistema



Para uma melhor compreensão da interface a ser desenvolvida, inicialmente foi desenvolvido uma máquina de estados que permitisse estabelecer os vários momentos e pontos que nos quais a vista oscilaria.

Desta forma, após uma página inicial e um menu de autenticação, o utilizador seria enviado para a sua pagina pessoal onde inicialmente se apresenta na *Dashboard* respetiva ao mesmo. A partir daqui, este poderia aceder ás varias paginas (estados) correspondentes ás tarefas que pretida fazer, como por exemplo aceder ao Calendário e respetivos eventos ou ás suas produções e consumos.

6.2 Mockups

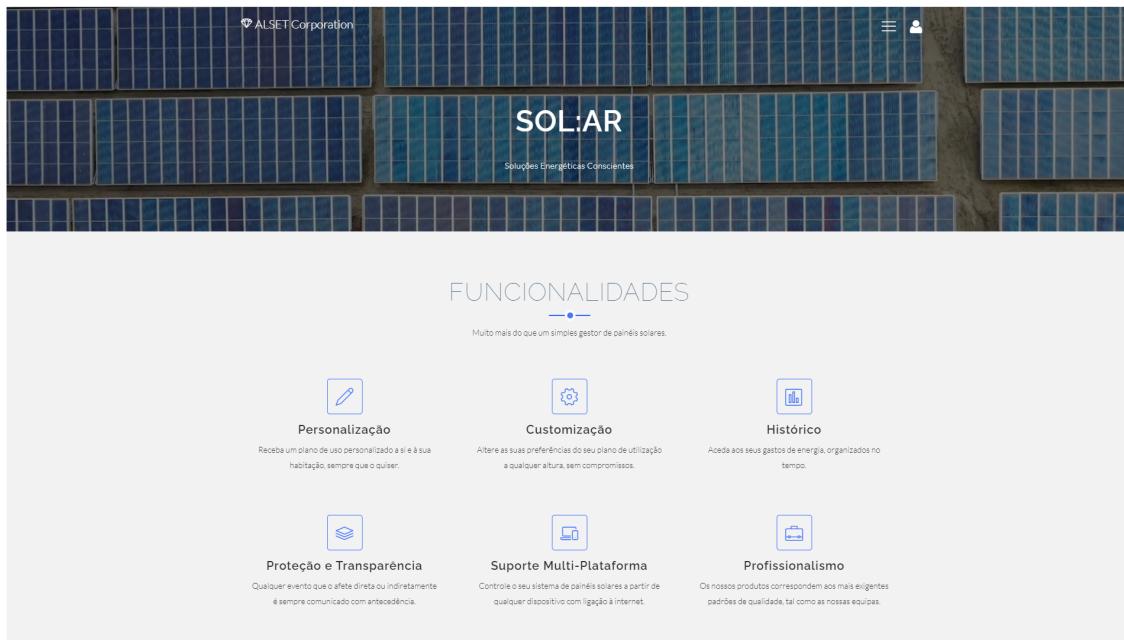


Figura 6.2: Página inicial da aplicação (sem sessão iniciada)

Para a página inicial, decidimos colocar as funcionalidades da aplicação, de forma a captar a atenção do potencial cliente e para o mesmo saber, rapidamente e sem esforço, aquilo que a nossa aplicação oferece.

Pode consultar o diagrama de atividade desta página na Figura 6.8.

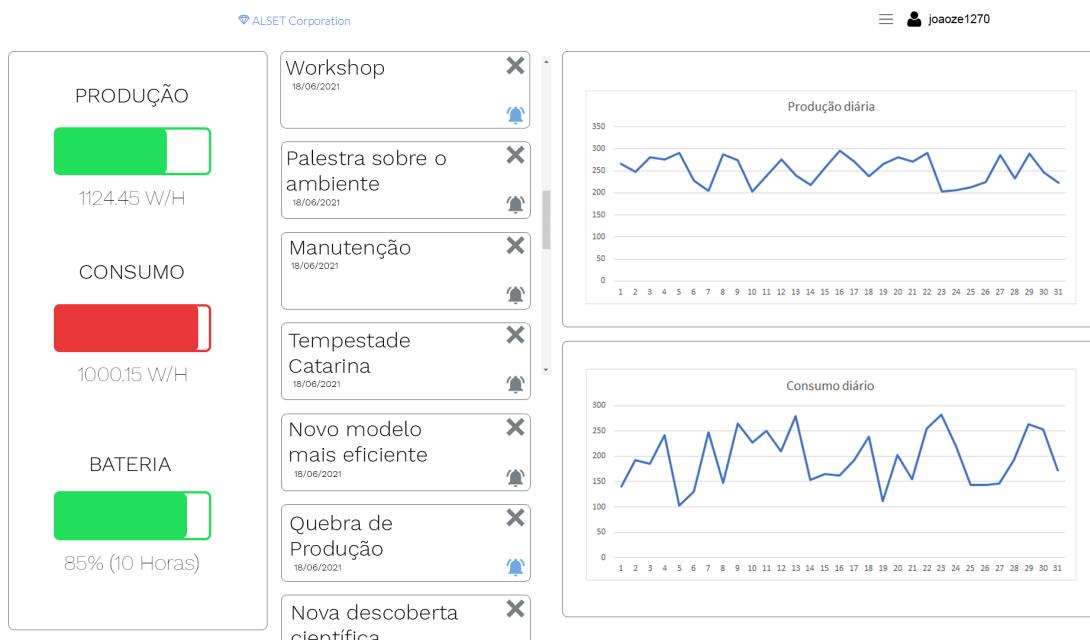


Figura 6.3: Dashboard do utilizador

Para a dashboard, decidimos apresentar toda a informação ao utilizador em painéis. À esquerda pode ver as informações atuais sobre o seu sistema de painéis solares, tais como a produção energética, consumo energético, e bateria restante. À direita é-lhe apresentada a evolução temporal dos consumos e produções energéticas diárias ao longo do último mês.

No centro da página são apresentados os eventos que o sistema determinar que são relevantes para o dado utilizador. Nestes painéis mais pequenos, o utilizador pode clicar no símbolo de ativar lembrete (um sino) para receber uma notificação pouco tempo antes desse evento se realizar. Para além disso, pode clicar no botão X para dispensar eventos em que não tenha interesse, e pode clicar no painel em si para lhe serem apresentados mais detalhes sobre o evento selecionado.

Pode consultar o diagrama de atividade desta página na Figura 6.9.

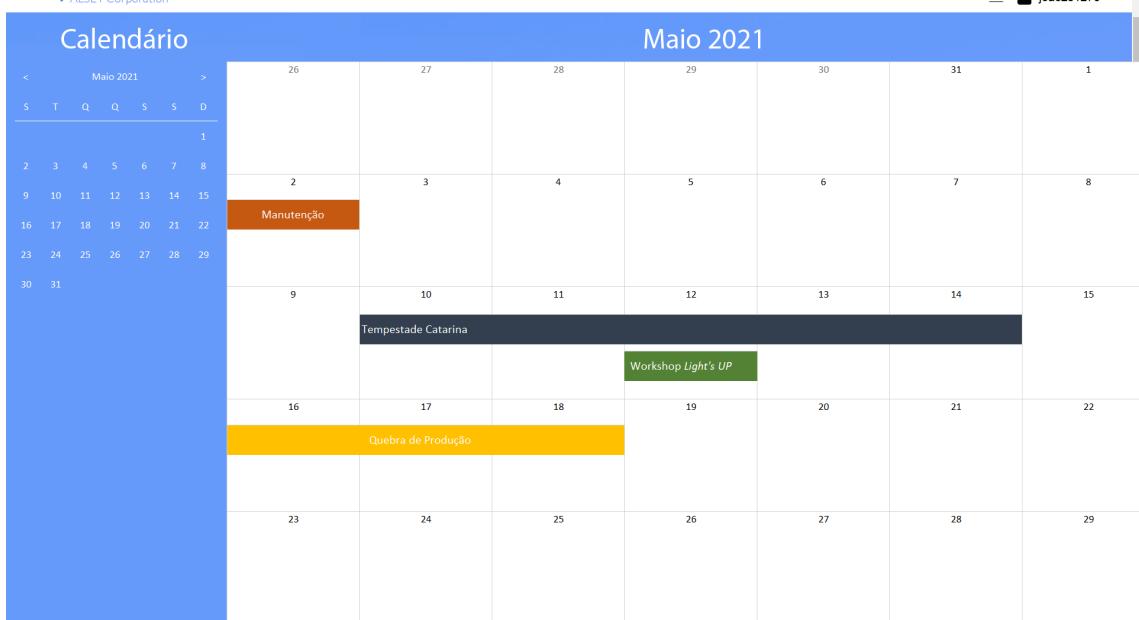


Figura 6.4: Calendário do utilizador

Como pretendemos que o calendário seja algo agradável à vista e de fácil interação para o utilizador, decidimos implementar um calendário onde seja percetível quando e qual evento irá decorrer. Apresentamos também do lado um calendário mais pequeno onde o utilizador poderá mudar o mês que pretende visualizar.

Pode consultar o diagrama de atividade desta página na Figura 6.10.

Distribuição dos Painéis Solares

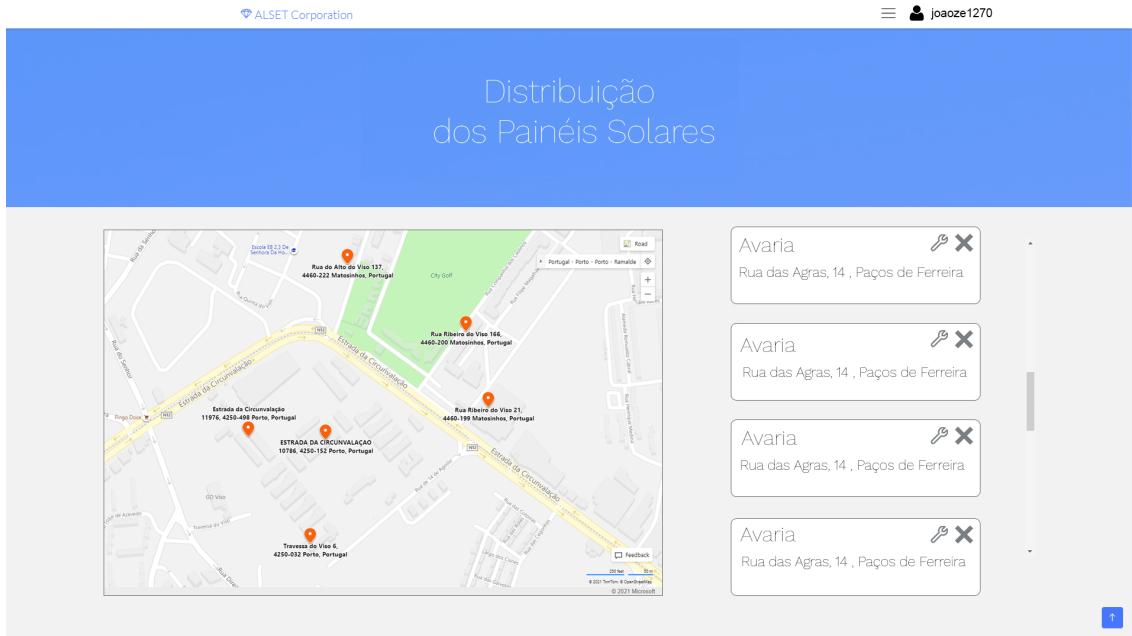


Figura 6.5: Dashboard do funcionário

De forma a que o funcionário consiga visualizar rapidamente as suas tarefas, decidimos colocar na sua dashboard um mapa onde o mesmo poderá ver a localização de todos os painéis que estão inseridos na rede da **SOL:AR**. Assim, o mesmo conseguirá sempre saber onde se encontra a casa do cliente a quem terá de prestar apoio.

Temos também, ao lado do mapa, uma lista de eventos aos quais o funcionário terá de ir dar suporte. Em cada caixa de evento, temos a indicação do tipo do evento, a localização do mesmo, um botão com o símbolo de uma ferramenta (onde o funcionário poderá clicar para marcar manutenção do painel) e um botão com formato de cruz para apagar o evento.

Pode consultar o diagrama de atividade desta página na Figura 6.11.

The screenshot shows the 'Informações Pessoais' (Personal Information) section of a user profile. At the top, there's a blue header bar with the text 'O meu perfil' (My Profile). The main content area has a light gray background. On the left, there's a sidebar with links: 'Informação Pessoal' (Personal Information), 'Definições de Notificações' (Notification Settings), 'Dashboard', and 'Os meus painéis' (My Dashboards). The right side contains three input fields: 'Nome' (Name) with the value 'João Zé do Pipo', 'Endereço de email' (Email Address) with the value 'joaoze1720@gmail.com', and 'Morada' (Address) with the value 'Rua das Ferreiras nº 17, Porto'. Below this, under 'Definições de Notificações' (Notification Settings), there are three toggle switches: 'Receber alertas meteorológicos' (Receive weather alerts), 'Receber alertas de quebra de produção' (Receive production break alerts), and 'Receber alertas de novos produtos SOL:AR' (Receive new products SOL:AR alerts), all of which are turned on (blue).

Figura 6.6: Secção das informações pessoais (Utilizador)

Na página pessoal, na secção das informações pessoais, o utilizador pode rapidamente visualizar todas as informações da sua conta, tal como alterar as mesmas, alterando os respetivos campos. À esquerda tem um menu para facilitar a navegação entre as diferentes secções da página pessoal. Ao clicar numa das opções a tela do utilizador é redirecionada para a zona correspondente da página. Também é visível parte da configuração das notificações que é possível fazer, de modo a o utilizador ser capaz de controlar os alertas que recebe do sistema.

Pode consultar o diagrama de atividade desta página na Figura 6.12.

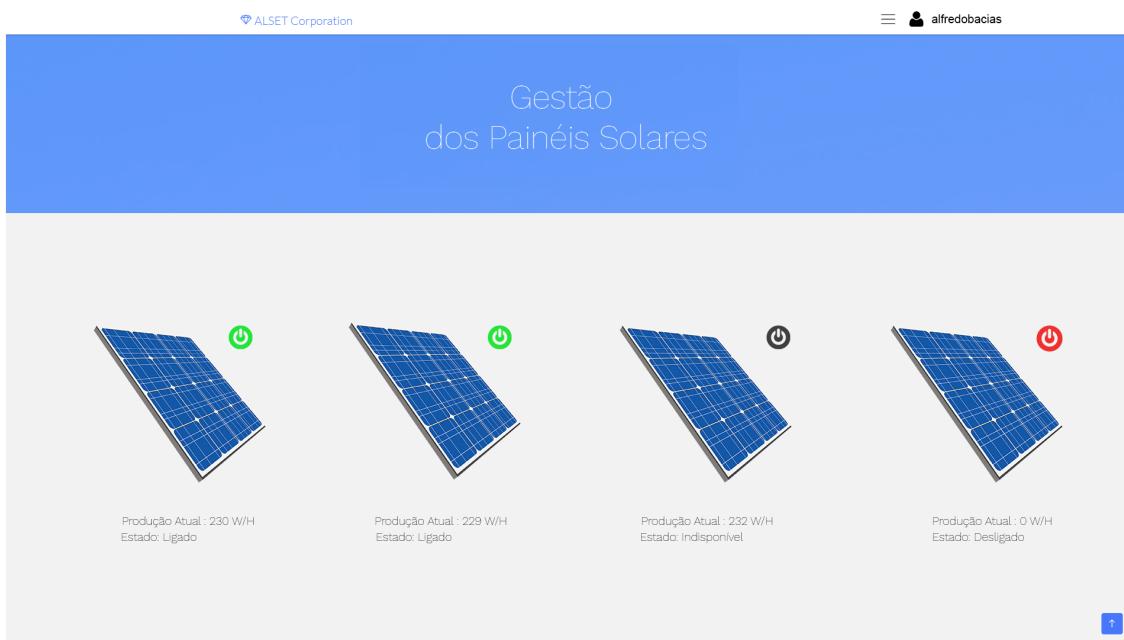


Figura 6.7: Secção do estado dos painéis (Utilizador)

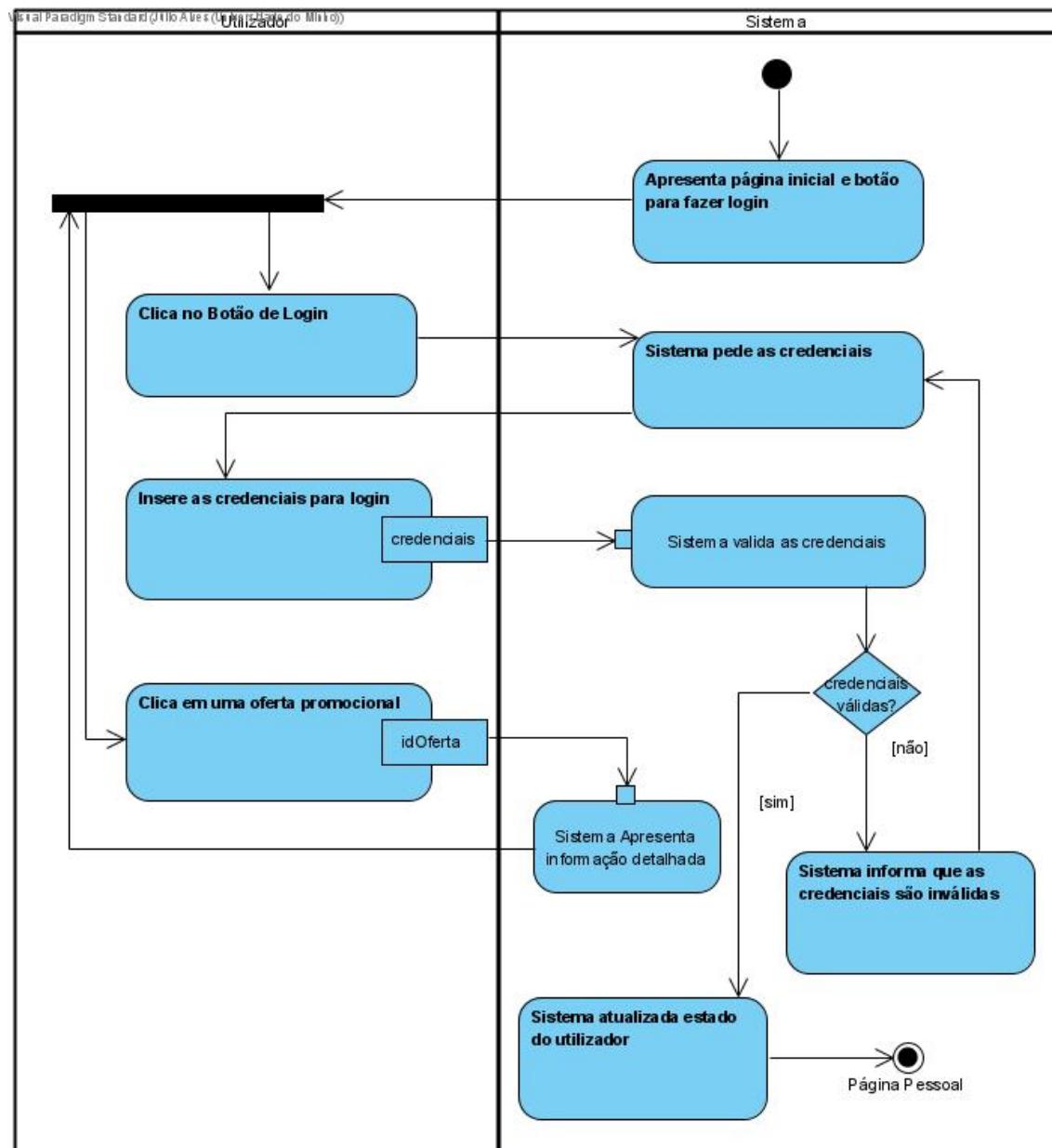
Nesta secção da sua página pessoal, o utilizador pode visualizar o estado de cada painel solar instalado na sua habitação e a respetiva produção energética atual. Para além disso pode também alterar o estado do painel clicando no botão correspondente. Não poderá alterar o estado do painel caso este se encontre avariado (ilustrado na cor cinzenta do botão de alteração de estado).

Pode consultar o diagrama de atividade desta página na Figura 6.13.

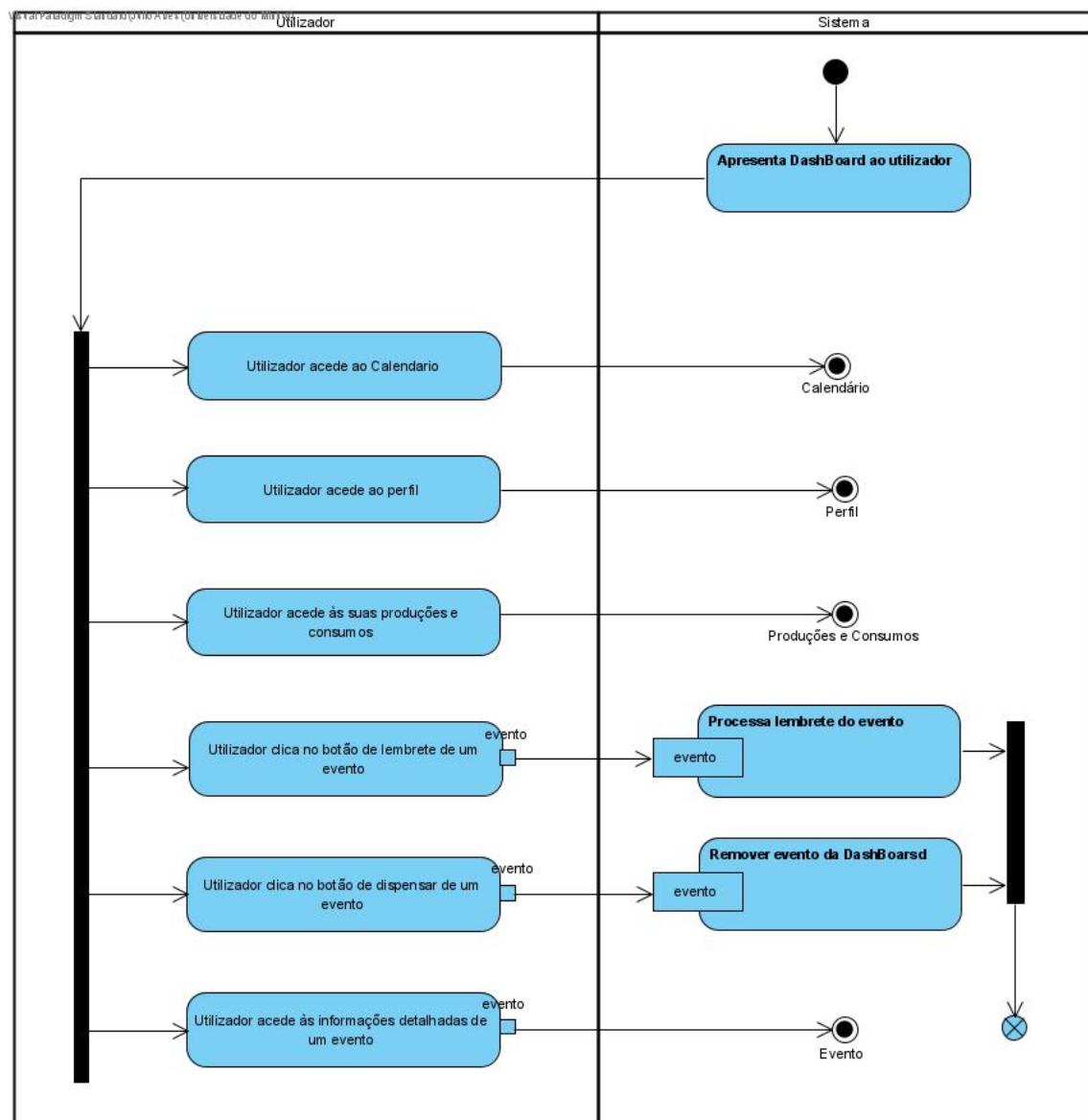
6.3 Diagramas de Atividade

Para cada *mockup* foi desenvolvido um diagrama de atividade que demonstra o fluxo de ações que existem em cada conceito demonstrado.

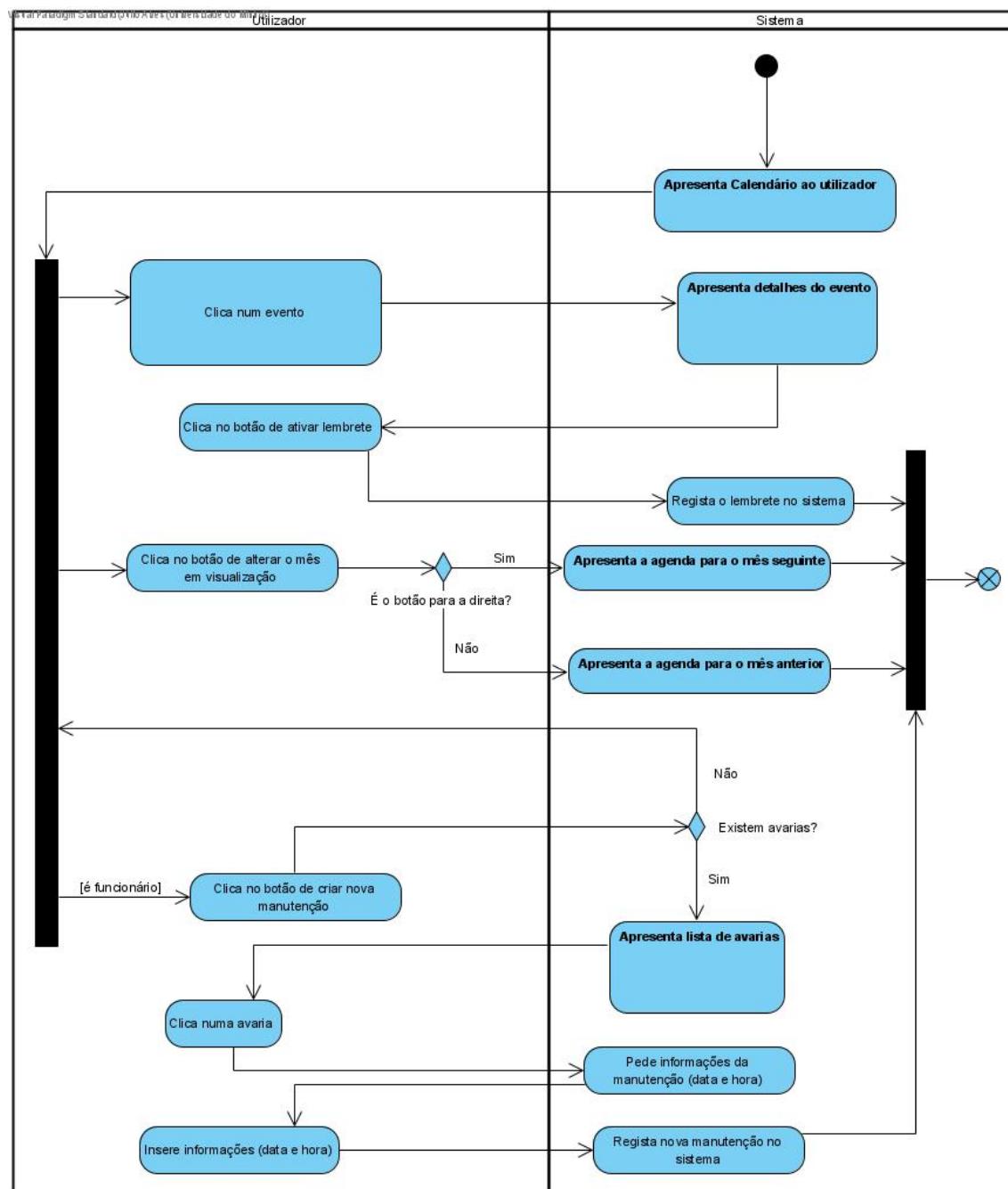
6.3.1 Diagrama de atividade da página inicial



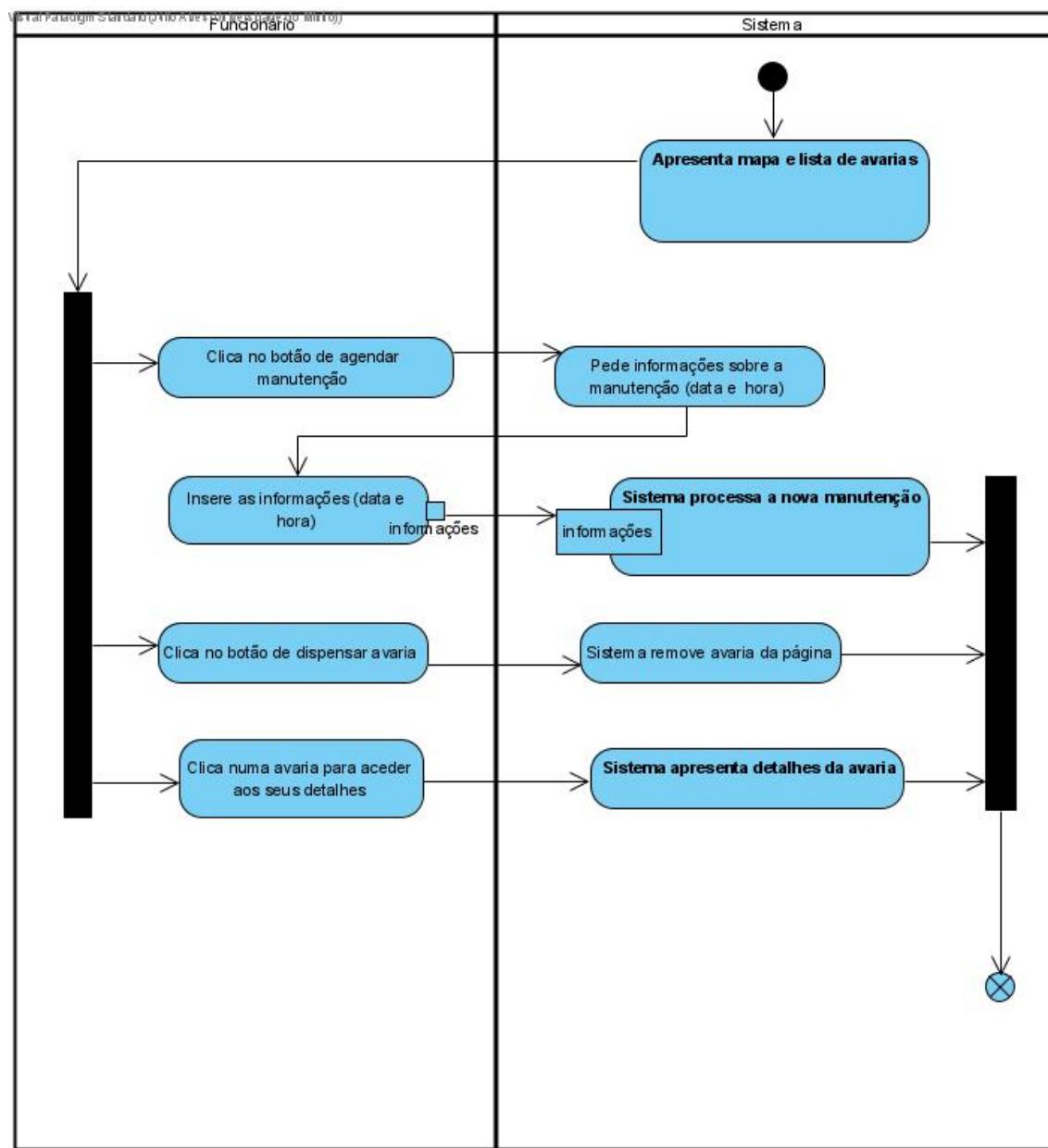
6.3.2 Diagrama de atividade da dashboard



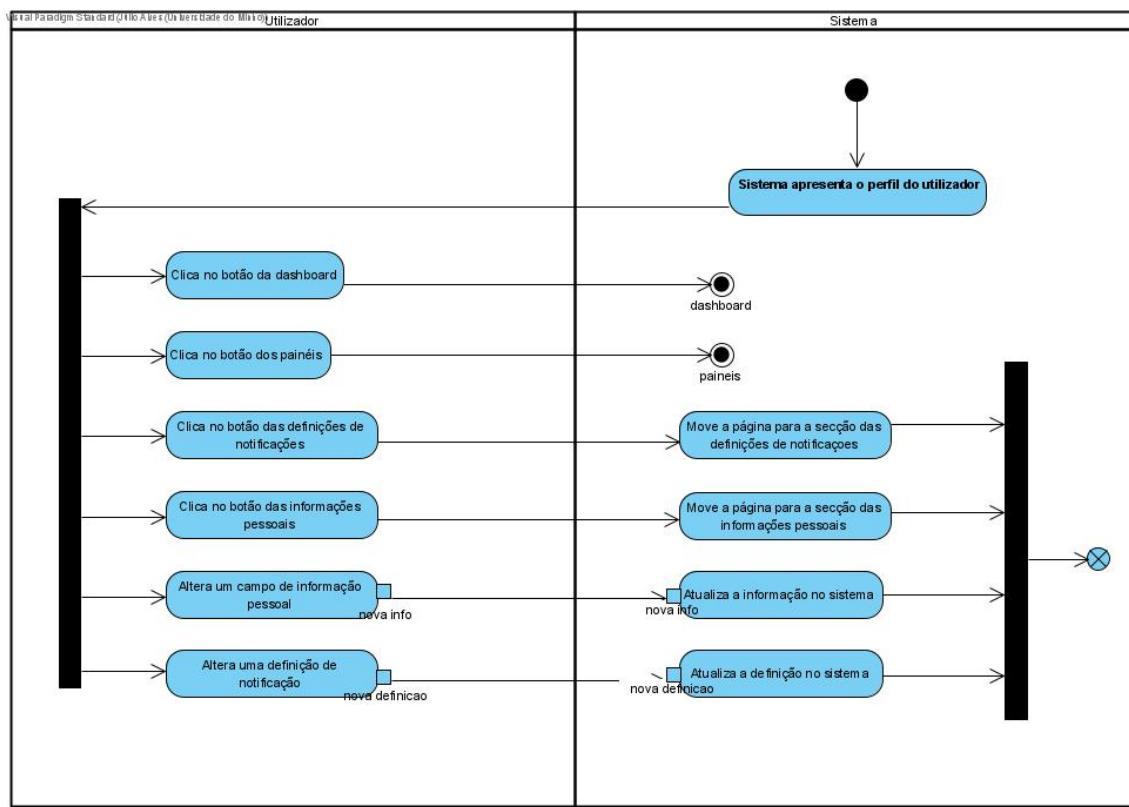
6.3.3 Diagrama de atividade do calendário



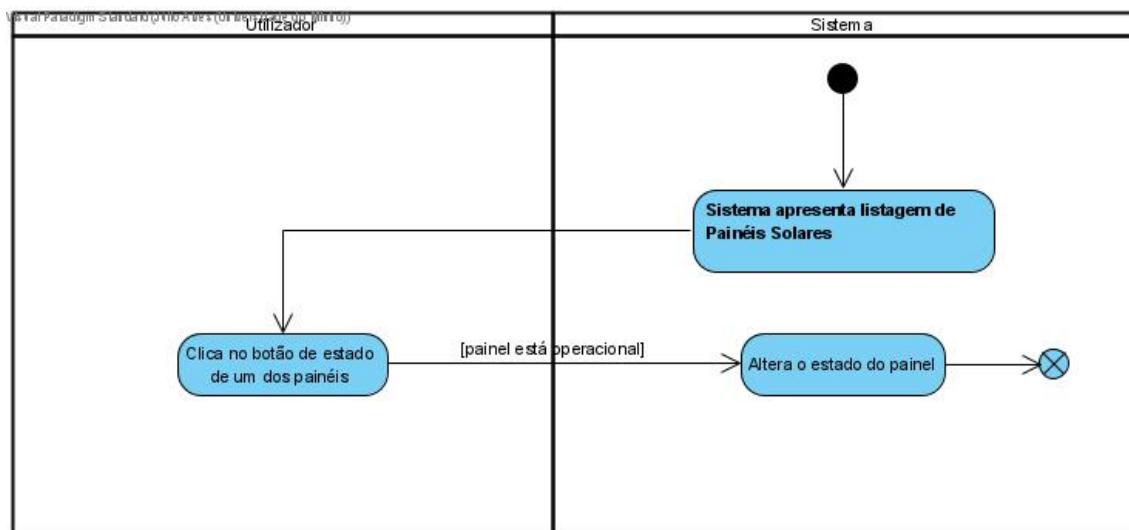
6.3.4 Diagrama de atividade da dashboard do funcionário



6.3.5 Diagrama de atividade do utilizador



6.3.6 Diagrama de atividade do utilizador



6.4 Ponto de situação - Diagrama de GANTT atualizado

SO:LAR

Laboratórios de Informática IV
Uminho

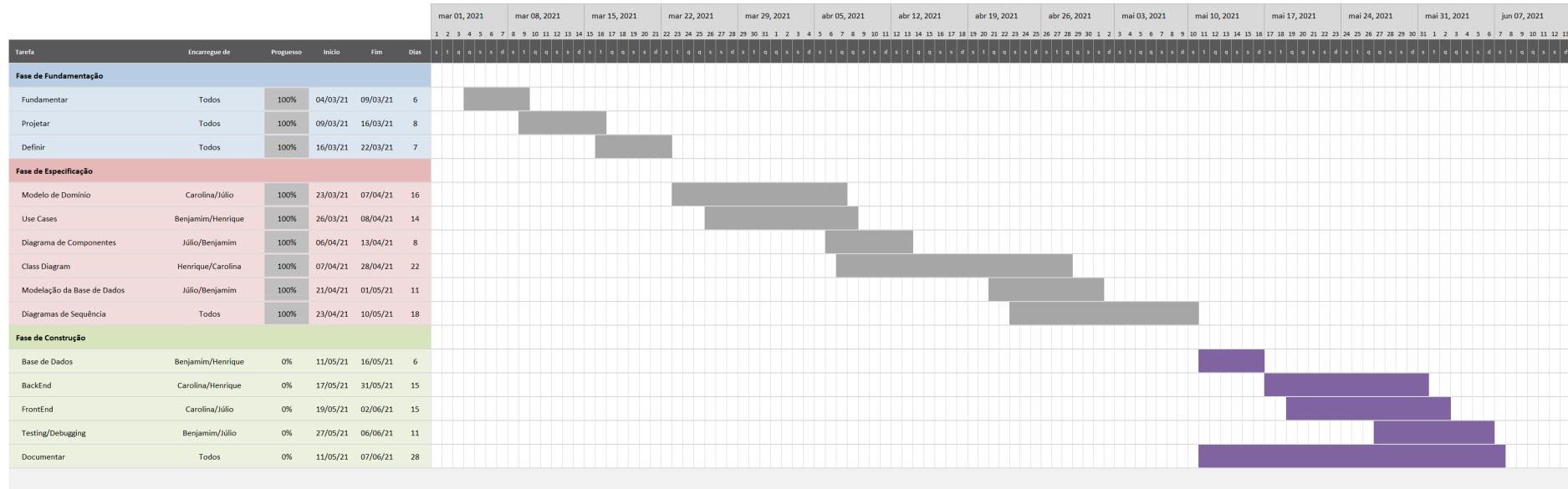


Figura 6.14: Diagrama de GANTT atualizado

7 Metodologia da Implementação

O padrão arquitectural escolhido foi o Model-View-Controller (MVC), uma vez que permite a separação da aplicação em três componentes modulares: Modelo, Vista e Controlador. Esta separação tem a vantagem de tornar a estrutura interna dos dados e procedimentos independente da interface do utilizador, simplificando assim a realização de testes e a implementação multi-plataforma.

O modelo foi dividido numa vista, lógica de negócio e numa camada de dados suportada por uma base de dados, com o objetivo de proporcionar a integridade, consistência, persistência e disponibilidade dos dados de toda a aplicação.

8 Ferramentas utilizadas na Implementação

Como estabelecido nas outras frase, a principal ferramenta utilizada na implementação foi o *ASP.NET Core* em conjunto com a base de objetos *Entity Framework*.

O gestor de base de dados escolhido foi o SQL Server. Este usa o modelo relacional e integra-se transparentemente com a hierarquia de ferramentas e funcionalidades do *ASP.NET core*.

9 Desenvolvimento do Projeto

9.1 Conexão à base de dados

A criação e conexão à base de dados é efetuada pelo *ASP.NET Core*. De acordo com a *framework*, o ficheiro *Startup.cs* é chamado aquando da inicialização do programa. De seguida é chamada uma função que inicializa a base de dados, que por sua vez chama o processo responsável pela sua população com valores de teste, o *DblInitializer.cs*.

9.2 Visualização dos painéis no mapa

Como é fundamental para o funcionário ter acesso à localização dos painéis solares que estão sobre o seu encargo, foi necessário pensar numa resposta para esse problema. A resposta passou pela utilização do Bing Maps API Service. Com recurso a esta ferramenta, temos acesso ao mapa mundo e o funcionário consegue saber a localização dos painéis através de pins.

9.3 Calendarização de eventos

De forma a ter acesso permanente à calendarização, recorremos à API *dhtmlxScheduler*. Esta permite ao utilizador marcar eventos de uma forma extremamente simples e intuitiva. Este calendário permite visualizar e criar novos eventos em três modos diferentes: diário, semanal e mensal. Apesar de ser possível efetuar estas operações sobre o calendário, estas alterações não estão a ser aplicadas com sucesso na base de dados. Isto constitui um problema a resolver no futuro da aplicação.

10 Produto Final

10.1 Área do Utilizador

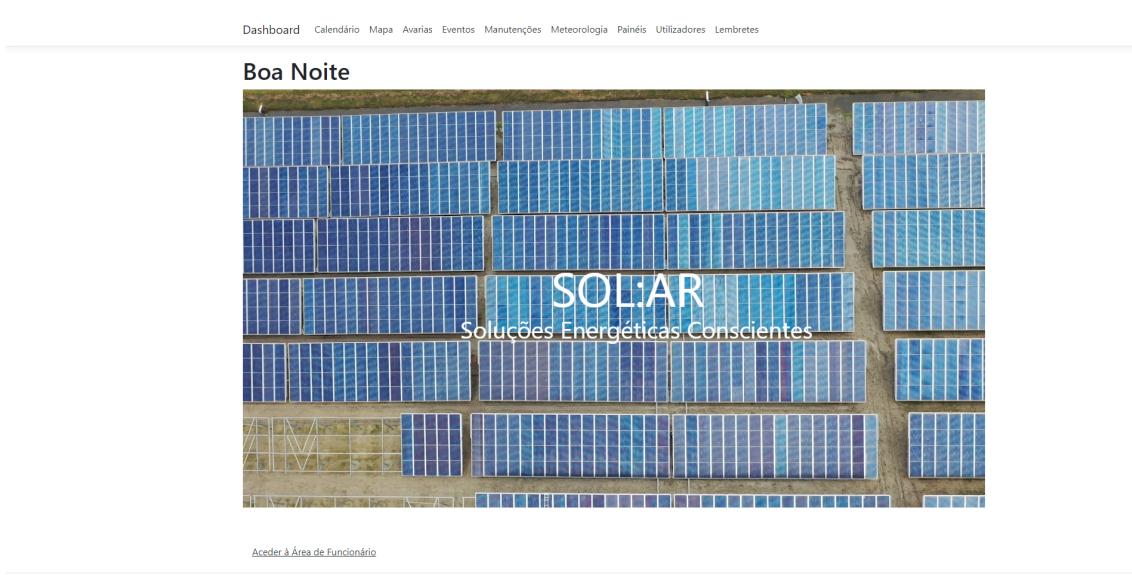


Figura 10.1: Página Inicial

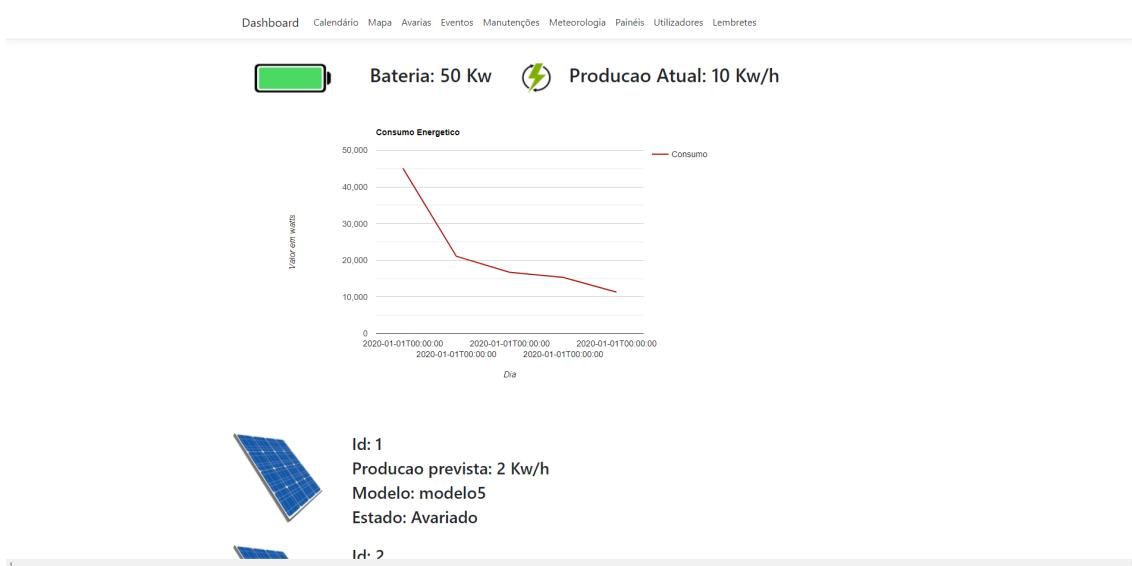


Figura 10.2: Visualização da Dashboard

10.2 Área do Funcionário

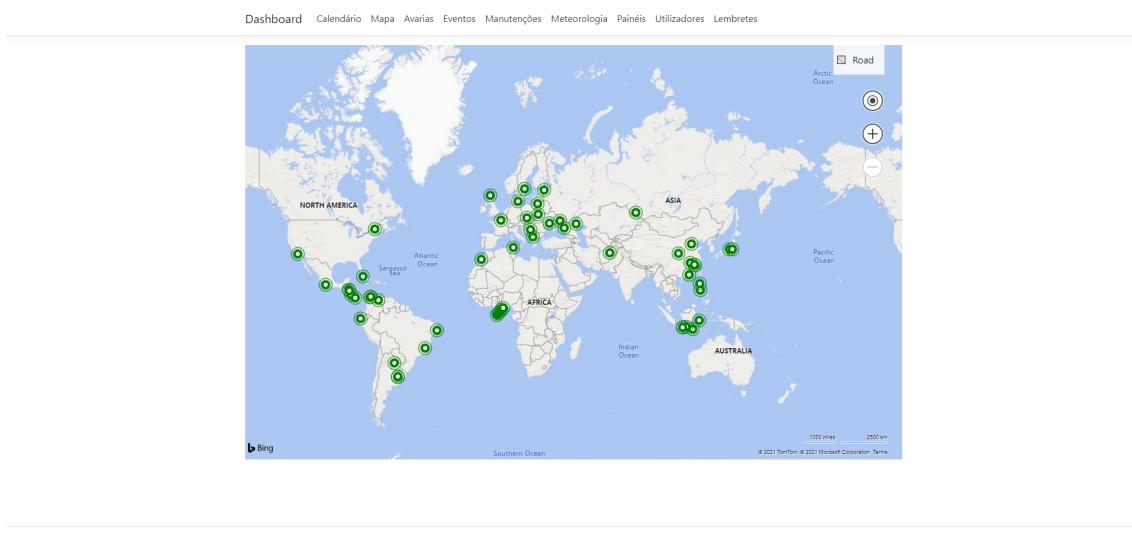


Figura 10.3: Visualização dos Painéis no mapa

10.2.1 Marcação de Eventos

Dashboard Calendário Mapa Avarias Eventos Manutenções Meteorologia Painéis Utilizadores Lembretes

Eventos

Criar Novo

Pesquisar por nome: Procurar | Voltar

Nome	Data	DataFinal	Descrição	Ações
evento1	18/04/2021	21/04/2021	Evento1	Editar Detalhes Apagar
evento2	19/04/2021	21/04/2021	Evento2	Editar Detalhes Apagar
evento3	20/04/2021	21/04/2021	Evento3	Editar Detalhes Apagar
evento4	21/04/2021	21/04/2021	Evento4	Editar Detalhes Apagar
evento5	22/04/2021	22/04/2021	Evento5	Editar Detalhes Apagar

© 2021 - SOLAR - [Dashboard](#)

Figura 10.4: Visualização dos eventos

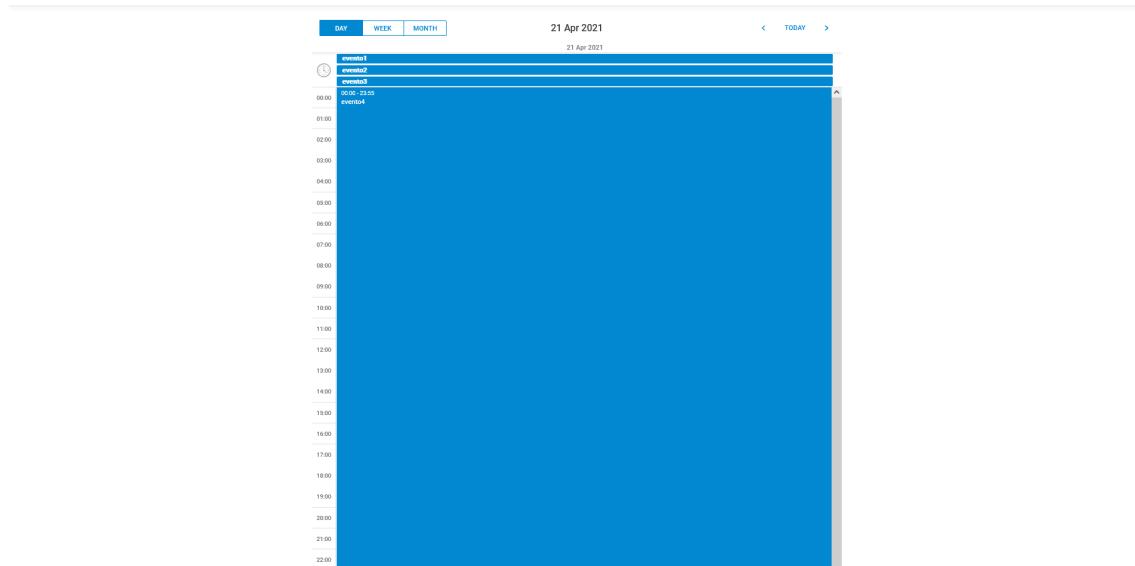


Figura 10.5: Visualização dos Eventos no calendário com vista diária

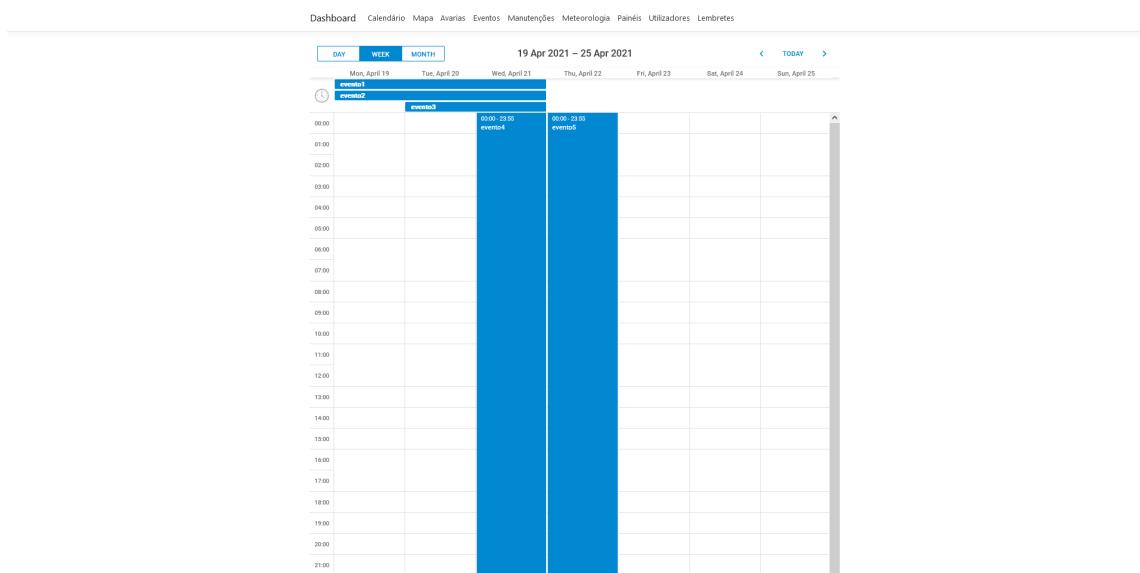


Figura 10.6: Visualização dos Eventos no calendário com vista semanal

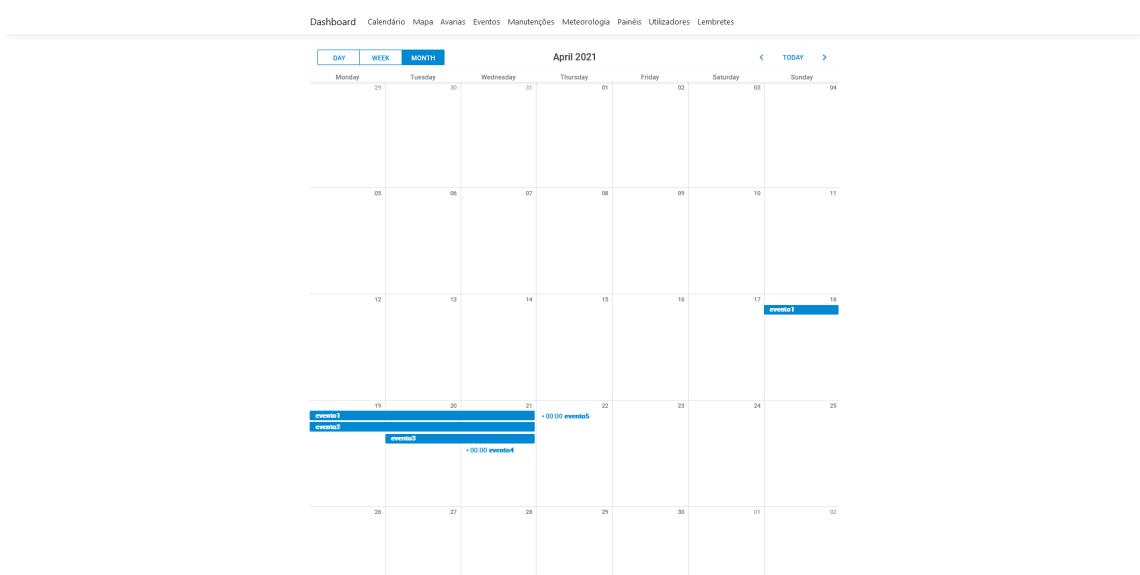


Figura 10.7: Visualização dos Eventos no calendário com vista mensal

11 Conclusões e Trabalho Futuro

Quando a nossa equipa foi abordada para a realização deste projeto, encaramos o mesmo com otimismo e com confiança que o conseguíramos concluir no prazo estabelecido.

Na fase da **Fundamentação** começamos por perceber o contexto da nossa aplicação, a sua utilidade, quem a iria utilizar e quais serão as funcionalidades que esta deveria oferecer aos seus utilizadores.

Ficamos a entender que havia um espaço no mercado para esta aplicação pois as energias renováveis são um tema que está em voga na atualidade e, se pretendemos um mundo sustentável, o caminho a seguir é este. Após dialogar com a nossa empresa mãe e com potenciais clientes, vimos também quais deveriam ser os focos desta aplicação, devendo esta ser uma aplicação user friendly e que oferecesse ao utilizador personalizar a mesma com base nos seus desejos.

Na fase da **Especificação**, passamos, então, a definitivamente estabelecer quais os requisitos que iríamos implementar, a arquitetura da aplicação e o seu funcionamento.

Para definir quais os requisitos que írimos efetivamente implementar, tivemos de fazer uma avaliação sobre quais os requisitos que conseguiríamos implementar, quais os que seriam realmente interessantes e que acrescentariam algo ao projeto e quais os requisitos mais importantes para os nossos clientes.

Relativamente à arquitetura da aplicação, após termos definido os requisitos a fazer, tivemos de avaliar qual estrutura seria a melhor para assegurar o bom funcionamento do programa, que permitiria ao mesmo ser escalável e que exigiria menos trabalho de manutenção futura.

Chegando à fase da **Implementação**, terceira e última do projeto, o nosso grupo finalmente encontrou adversidades que puseram em causa a concretização da aplicação.

Nesta fase, o nosso grupo finalmente começou a criar a aplicação em si, no entanto, acreditamos agora ter sido demasiado ambiciosos na fase de planeamento do projeto, pois tivemos de utilizar ferramentas com as quais nunca tínhamos trabalhado e numa área completamente desconhecida para todos nós. Concluímos então que nos comprometemos a realizar o projeto num intervalo de tempo demasiado pequeno para as nossas capacidades e para um projeto que acarretava este nível de complexidade.

Posto isto, não conseguimos concretizar algumas das tarefas às quais nos tínhamos comprometido, tais como ativar alerta para um dado evento, fazer login, aceder e editar às informações

pessoais, consultar o histórico de reparações, ver alertas de avarias e comunicar avaria.

Chegando à data de entrega do projeto, percebemos que não conseguimos implementar o programa a que nos comprometemos. Isto deveu-se, principalmente, à nossa inexperiência em desenvolvimento web e ao tempo que destinámos a esta fase que se revelou demasiado curto.

Referências

- [1] Arthur Neslen, 2016. Portugal runs for four days straight on renewable energy alone. [online] (19 May and 21 December 2016) Available at: <https://www.theguardian.com/environment/2016/may/18/portugal-runs-for-four-days-straight-on-renewable-energy-alone> * [Accessed 20 March 2021].

Lista de Siglas e Acrónimos

HTML HyperText Markup Language

CSS Cascading Style Sheets

Anexos

Anexo 1

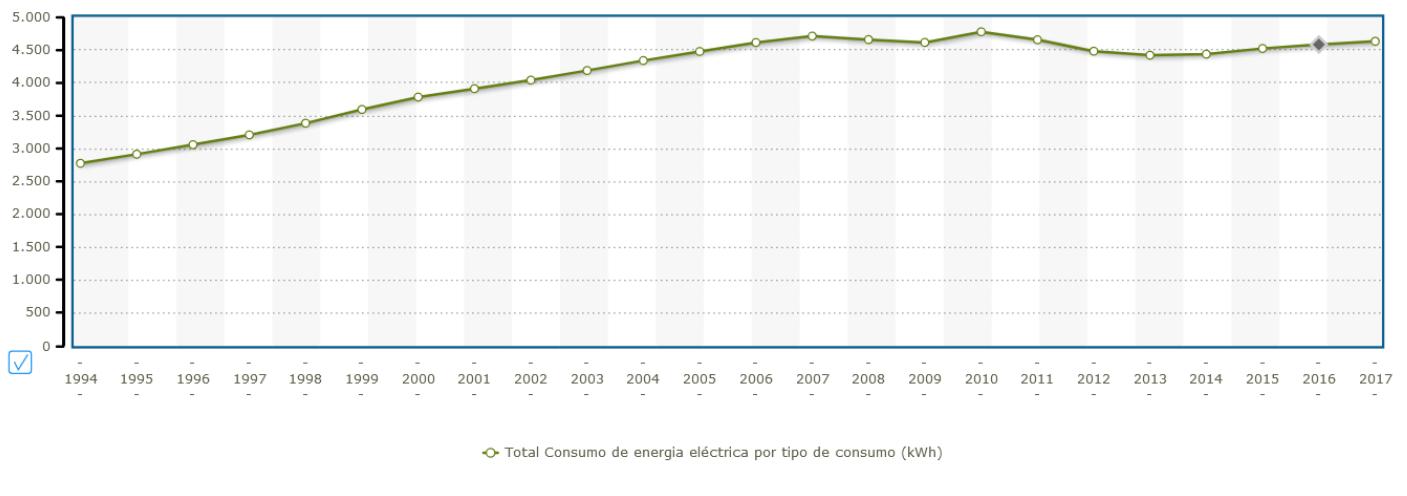


Figura 11.1: Consumo de energia elétrica per capita: total e por tipo de consumo
kWh (quilowatt-hora)/ hab. - Rácio