

App Install & Go

Roberto Filipe Manso Barreto
(nrº 21123, regime diurno)

Orientação de
Luís Gonzaga Martins Ferreira

LICENCIATURA EM ENGENHARIA EM SISTEMAS INFORMÁTICOS
ESCOLA SUPERIOR DE TECNOLOGIA
INSTITUTO POLITÉCNICO DO CÁVADO E DO AVE

Identificação do aluno

Roberto Filipe Manso Barreto
Aluno número 21123, regime diurno
Licenciatura em Engenharia em Sistemas Informáticos

Orientação

Luís Gonzaga Martins Ferreira

Informação sobre o Estágio

Motorline Eletrocelos S.A
Travessa do Sobreiro, 29 Rio Côvo (Sta. Eugénia) 4755-474 Barcelos
Eng. Helder Remelhe

Resumo

Este documento trata o processo de análise, especificação e implementação da solução Install&Go. Esta vem resolver o problema de comunicação entre a empresa Motorline e os seus técnicos, dado que, na eventualidade de um problema estes deverão telefonar para a empresa, o que gera sobrecarga do sistema.

Para a resolução deste problema foi desenvolvido uma plataforma de fórum, onde as empresas podem registar os seus técnicos. Aqui, podem expor questões, onde técnicos externos de outras empresas, e/ou técnicos Motorline poderão prestar auxílio. Se um técnico possuir um problema já resolvido, este poderá pesquisar pela solução dada no fórum.

O desenvolvimento desta solução proveu a aquisição de novas capacidades tecnológicas como o desenvolvimento *cross-platform* e as suas *frameworks*, sendo que destas foi explorado o *flutter*. Através da elaboração desta aplicação foi possível, para além das competências mencionadas anteriormente, assimilar capacidades como análise, especificação de projetos e comunicação com clientes. Por fim, foi desenvolvida completamente a solução conforme as necessidades e expectativas do cliente.

Abstract

This document describes the analysis, specification and development process of the Install&Go solution. This solution solves the communication problem between Motorline and their professionals, since that, in the event of a problem, they must call the company, which generates an overload.

To solve this problem a forum platforma was developed, where companies can register their professionals. Here they can submit questions, so that other professionals from other companies and/or Motorline can provide assistance. If a professional has a problem that has already been solved, he can search for the solution in the forum

The development of this solution provided the acquisition of new technological skills such as cross-platform development and its frameworks, of which flutter was explored. Through the elaboration of this application it was possible, besides the competences previously mentioned, to assimilate skills such as analysis, project specification and communication with clients. At the end, the solution was completely developed according to the client's needs and expectations.

Agradecimentos

Em primeiro lugar, gostaria de agradecer à minha família, com destaque à minha mãe, ao meu padrinho, aos meus avós e à minha namorada, que com todo o carinho orientaram-me nesta caminhada.

Também, gostaria de salientar um caloroso agradecimento à empresa Motorline, uma vez que, fizeram-me sempre sentir um membro da equipa, com destaque ao supervisor Helder Remelhe que sempre esteve disponível para eventuais dúvidas que surgissem no desenvolvimento do projeto.

Por fim, mas não menos importante, gostaria de enfatizar toda a orientação, disponibilidade, conversa e ensinamentos proporcionados pelo professor doutor Luís Ferreira e também aos meus colegas de curso Henrique Cartucho e João Castro que mais que colegas, são amigos para a vida.

Conteúdo

1	Introdução	1
1.1	Objetivos	2
1.2	Contexto	2
1.3	Planificação do trabalho	3
1.4	Estrutura do documento	4
2	Estado da arte	5
2.1	Ferramentas de trabalho utilizadas	5
2.2	Plataforma Tecnológica	6
2.2.1	Web Scraper	6
2.2.1.1	Selenium	6
2.2.2	Serviços Backend	7
2.2.2.1	Serviços RestFull e SOAP	7
2.2.2.2	NodeJS	8
2.2.2.3	Typescript	8
2.2.2.4	PostgreSQL	9
2.2.2.5	Logs e Logging	9
2.2.2.6	Morgan	9
2.2.2.7	Gestão de <i>emails</i>	10
2.2.2.8	Agendamento de tarefas	10
2.2.2.9	Encriptação de <i>passwords</i>	10
2.2.2.10	Encriptação de configurações do servidor	11
2.2.2.11	Firebase	12
2.2.2.12	Axios	12
2.2.3	<i>Frontend</i>	13
2.2.3.1	Desenvolvimento cross-platform	13
2.2.3.2	Flutter	14
2.2.3.3	Dart	14

2.2.3.4	Links de aplicações	15
2.2.4	Qualidade de código	16
2.2.4.1	Design patterns	16
2.2.4.2	Documentação	16
2.2.4.3	Typedoc	16
2.2.4.4	Swagger	17
2.2.4.5	Testes de código	17
3	Análise e especificação	19
3.1	Modelo de negócio	19
3.2	Objetivos de negócio	19
3.3	Estudo de Soluções existentes	20
3.4	Domínio de aplicação do sistema	20
3.5	Operações a realizar no sistema	21
3.6	Descrição dos intervenientes	24
3.7	Partes Interessadas	24
3.8	Condições Específicas	24
3.9	Esquematização do conteúdo das páginas	25
3.9.1	Autenticação e página Inicial	26
3.9.2	Fórum	27
3.9.3	Criar nova tópico	27
3.9.4	Detalhes de tópicos	28
3.9.5	Pesquisa de tópicos	28
3.9.6	Notificações	29
3.9.7	Perfil	29
3.9.8	Gestão de recursos humanos	30
3.10	<i>User Stories</i>	31
3.11	Casos de uso	34
3.11.1	Especificação de casos de uso	36
3.11.1.1	Especificação de caso de uso de listagem de tópicos	36
3.11.1.2	Especificação de caso de uso de criar novo tópico	37
3.11.1.3	Especificação de caso de uso de pesquisar tópicos por escrito	38
3.11.1.4	Especificação de caso de uso de ver finalizar tópico	38
3.11.1.5	Especificação de caso de uso de selecionar melhor resposta	39
3.11.1.6	Especificação de caso de uso de eliminar tópico	40

3.11.1.7	Especificação de caso de uso de alterar visibilidade de um tópico	40
3.11.1.8	Especificação de caso de uso gostar de um tópico	41
3.11.1.9	Especificação de caso de uso gostar de um comentário	41
3.11.1.10	Especificação de caso de uso de comentar tópico	42
3.11.1.11	Especificação de caso de uso ativar conta	43
3.11.1.12	Especificação de caso de uso configurar notificações	44
3.11.1.13	Especificação de caso de uso registar técnico	44
3.11.2	Diagramas de casos de uso	45
3.11.2.1	Casos de uso Fórum	45
3.11.2.2	Casos de uso de pesquisar tópicos	45
3.11.2.3	Casos de uso ver detalhes de tópico	46
3.11.2.4	Casos de uso ver comentários	46
3.11.2.5	Casos de uso ativação de conta	47
3.11.2.6	Casos de uso perfil	47
3.11.2.7	Casos de uso notificações	48
3.11.2.8	Casos de uso gestão de recursos humanos	48
3.12	Diagrama Entidade Relação	49
3.12.1	Escolha de diagrama de entidade relação	50
3.12.2	Dicionário de termos	51
3.13	Diagrama de Classes	55
3.14	Mockups	56
3.14.1	Página Inicial	56
3.14.2	Autenticação - Login e Registo	57
3.14.3	Autenticação - Ativação e Confirmação de conta	58
3.14.4	Página inicial fórum	59
3.14.5	Página de detalhes de um tópico	60
3.14.6	Página de criação de um tópico	61
3.14.7	Página de notificações	62
3.14.8	Página de perfil de utilizador	63
3.14.9	Página de gestão de recursos humanos	64
3.14.10	Página de perfil de técnico registado	65
3.14.11	Página de registo de novo técnico	66
3.15	Diagramas de atividades	67
3.15.1	Diagrama de atividades página inicial	67
3.15.2	Diagrama de atividades página de perfil	67

3.15.3	Diagrama de atividades página inicial do fórum	68
3.15.4	Diagrama de atividades página de criação de tópico	69
3.15.5	Diagrama de atividades página de detalhes do tópico	70
3.15.6	Diagrama de atividades páginas de autenticação	71
3.15.7	Diagrama de atividades registrar técnico	72
3.15.8	Diagrama de atividades confirmar conta	72
3.16	Diagramas de estados	73
3.16.1	Diagrama de estados criação de tópico	73
3.16.2	Diagrama de estados responder a tópico	74
3.16.3	Diagrama de estados autenticação e validação de conta	75
3.17	Diagrama de sequência	76
3.17.1	Diagrama de sequência Login e ativação de conta	76
3.17.2	Diagrama de sequência Registo e ativação de conta	77
3.17.3	Diagrama de sequência registo de técnicos	78
3.18	Arquitetura de sistema	79
3.18.1	Arquitetura de funcional	79
3.18.2	Arquitetura de componentes	80
3.18.3	Tabela de endpoints	81

Lista de Figuras

1.1	Planificação de <i>sprints</i>	3
2.1	Funcionamento dos dynamic links em <i>Android</i> (?)	15
3.1	Diagrama de contexto da aplicação	20
3.2	Esquema de organização de páginas do <i>software</i>	25
3.3	Esquema de organização das páginas de autenticação e página inicial . . .	26
3.4	Esquema de organização da página de fórum	27
3.5	Esquema de organização da página de criação de tópicos	27
3.6	Esquema de organização da página de detalhes de tópico	28
3.7	Esquema de organização da página de pesquisa de tópicos	28
3.8	Esquema de organização da página de notificações	29
3.9	Esquema de organização da página de perfil	29
3.10	Esquema de organização da página de gestão de recursos humanos	30
3.11	Diagrama de casos de uso de fórum	45
3.12	Diagrama de casos de uso de pesquisa de tópicos	45
3.13	Diagrama de casos de uso de detalhes de tópico	46
3.14	Diagrama de casos de uso de ver comentários	46
3.15	Diagrama de casos de uso de ativação de conta	47
3.16	Diagrama de casos de uso de perfil	47
3.17	Diagrama de casos de notificações	48
3.18	Diagrama de casos de uso de recursos humanos	48
3.19	Diagrama Entidade Relação base de dados Install&Go	49
3.20	Diagrama Entidade Relação alternativo	50
3.21	Diagrama de classes Install&Go	55
3.22	Página inicial do fórum	56
3.23	Autenticação - Login e Registo	57
3.24	Autenticação - Ativação e Confirmação de conta	58
3.25	Página inicial do fórum	59

3.26	Página de detalhes de tópico do software	60
3.27	Página de criação de tópico	61
3.28	Página de notificações	62
3.29	Página de perfil de utilizador	63
3.30	Página de gestão de recursos humanos	64
3.31	Página de perfil de técnico registado	65
3.32	Página de registo de novo técnico	66
3.33	Diagrama de atividades de página inicial da aplicação	67
3.34	Diagrama de atividades de página de perfil	67
3.35	Diagrama de atividades de página inicial do fórum	68
3.36	Diagrama de atividades de página de criação de tópico	69
3.37	Diagrama de atividades de página de detalhes do tópico	70
3.38	Diagrama de atividades de página de validação de conta	71
3.39	Diagrama de atividades de página de registar técnico	72
3.40	Diagrama de atividades de página de confirmar conta de técnico	72
3.41	Diagrama de estados de criar tópico	73
3.42	Diagrama de estados de criar tópico	74
3.43	Diagrama de estados de autenticação e validação de conta	75
3.44	Diagrama de sequência de login e ativação de conta	76
3.45	Diagrama de sequência de registo e validação de conta	77
3.46	Diagrama de sequência de registo de técnicos	78
3.47	Arquitetura do sistema	79
3.48	Arquitetura do funcional	79
3.49	Arquitetura de componentes	80
3.50	Listagem de serviços documentados	82

Lista de Tabelas

3.1	Tabela de requisitos funcionais	21
3.2	Tabela de requisitos não funcionais	24
3.3	Tabela de <i>user stories</i>	31
3.4	Tabela de casos de uso	34
3.5	Tabela de especificação de caso de uso de listagem de tópicos do técnico . .	36
3.6	Tabela de especificação de caso de uso login	37
3.7	Tabela de especificação de caso de uso de pesquisa por escrito	38
3.8	Tabela de especificação de caso de uso de finalizar tópico	38
3.9	Tabela de especificação de caso de uso de selecionar melhor resposta	39
3.10	Tabela de especificação do caso de uso de eliminar tópico	40
3.11	Tabela de especificação de caso de uso de alteração de visibilidade de um tópico	40
3.12	Tabela de especificação de caso de uso de gostar de um tópico	41
3.13	Tabela de especificação de caso de uso de gostar de comentário	41
3.14	Tabela de especificação de caso de uso de comentar um tópico	42
3.15	Tabela de especificação de caso de uso ativação de conta	43
3.16	Tabela de especificação de caso de uso de configuração de notificações . . .	44
3.17	Tabela de especificação de caso de uso de registrar técnico	44
3.18	Dicionário de termos da base de dados	51
3.19	Tabela de endpoints	81

Siglas & Acrónimos

API Application Programming Interface. 7, 79

JSON JavaScript Object Notation. 7, 17

MVC Model View Controller. 16

UC Use Cases. 34

US User Stories. 31

1. Introdução

Este projeto intitulado *Install & Go*, refere-se a uma aplicação para *smartphone* direcionada a todos os clientes e técnicos Motorline, com o propósito de agilizar todo o processo de resolução de problemas e de acesso aos produtos. Para alcançar estes objetivos, a aplicação conta com um catálogo para todos os utilizadores e também com um fórum para os clientes e técnicos Motorline.

No início de estágio foi indicado pelo supervisor da empresa a subdivisão do projeto em duas grandes componentes. Também este projeto foi desenvolvido com uma colega que ficou encarregue de desenvolver o *frontend* da componente do catálogo. Já o desenvolvimento completo do *backend*, aquisição do catálogo de produtos do *website* da empresa e o *frontend* para o fórum, gestão de utilizadores e autenticação ficaram a cargo do autor deste documento.

O fórum da aplicação é uma plataforma que permite aos clientes e técnicos criar tópicos para realizar questões e/ou expor problemas para a comunidade. Estes tópicos podem ser comentados, onde também é possível realizar uma comunicação.

O suporte *backend* da aplicação foi realizado através de um conjunto de serviço, sendo inicialmente apenas necessário para fórum. Contudo, foi posteriormente acrescentado suporte para o catálogo de produtos.

1.1 Objetivos

A plataforma do fórum da aplicação deverá permitir que a comunidade partilhe os seus tópicos. Para isso, o utilizador necessitará conseguir criar estes com a indicação do problema e uma descrição do mesmo, com a possibilidade de anexar imagens, assim como referenciar o produto, para facilitar a identificação e resolução do problema.

A comunidade deverá também ter a possibilidade de comentar os tópicos e comunicar nesta secção. O responsável pelo tópico necessitará ter a possibilidade de o colocar em privado caso tenha como objetivo que apenas técnicos Motorline respondam ao mesmo. Este também deverá ter a seu dispor a possibilidade de indicar quando o tópico estiver finalizado e qual a melhor resposta que obteve. Os utilizadores deverão também conseguir gostar de tópicos e comentários para destacar os mesmos perante a restante comunidade.

A comunidade deverá conseguir ver os tópicos em destaque, os mais recentes, os seus tópicos, os que não estão finalizados e os técnicos Motorline necessitarão de conseguir ver os tópicos privados existentes. A comunidade deverá ter a possibilidade de pesquisar por tópicos com assuntos específicos, pesquisa por nome e por produto. Para filtragem de pesquisa a comunidade deverá também conseguir seleccionar o tipo de tópico.

1.2 Contexto

O projeto foi desenvolvido na empresa Motorline Eletrocelos S.A daqui em diante designada Motorline, esta empresa é especializada na produção e comercialização de automatismos para portas e portões, sistemas de controlo de acessos, sistemas de segurança, entre outros produtos relacionados com o setor da automação.

Este projeto vem por este meio resolver o problema de comunicação com o cliente, que atualmente para solucionar as suas questões tem de contactar a assistência técnica que por vezes pode estar sobrecarregada, pelo que deverão preencher um formulário para expor a sua questão.

Com esta solução os clientes e técnicos Motorline poderão procurar ou expor os seus problemas com a comunidade onde têm a possibilidade de ser respondidos, o que torna o processo de resolução de problemas mais ágil.

1.3 Planificação do trabalho

Para obter uma visão geral do projeto e uma previsão de finalização foi realizada uma planificação expectável de tarefas.



Figura 1.1: Planificação de *sprints*

1.4 Estrutura do documento

Este documento contém a descrição de todo o processo de engenharia de *software*, desenvolvimento e pensamento sobre o problema em mãos, sendo estes pontos divididos sobre diversos tópicos:

1. Estado da Arte, onde é abordado as ferramentas utilizadas e tecnologias exploradas.
2. Análise e especificação, onde é abordado o estado da arte, descrito o modelo de negócio e apresentada toda a engenharia de *software*
3. Trabalho desenvolvido, onde é descrito todo o processo de desenvolvimento do projeto.
4. Análise de resultados, onde é discutido os resultados obtidos.
5. Conclusão e trabalho futuro, onde é abordada a conclusão sobre o projeto e também futuras implementações que poderiam ser realizadas.

2. Estado da arte

Para a organização de todo o trabalho a desenvolver, visto que este é dividido com mais uma colega, foi utilizada a técnica de desenvolvimento ágil, através da qual foi possível organizar todas as tarefas entre os elementos de desenvolvimento do projeto.

2.1 Ferramentas de trabalho utilizadas

A organização de todas as tarefas, foi realizada na ferramenta *Github Projects*. Esta dispõe de funções que permitem ligar um projeto a um repositório de *Github*, onde se consegue personalizar completamente todo o projeto e parâmetros das tarefas que resulta numa organização minuciosa.

O *Microsoft Excel* foi utilizado para a engenharia de *software* onde foram descritos os requisitos do projeto, *user stories* e também a especificação de casos de uso. Esta ferramenta também foi utilizada para a organização de reuniões com o cliente e redação de tópicos a abordar.

Para o desenvolvimento do *design* do *software* foi utilizada a ferramenta *figma*, que permite o *design* de todas as componentes tendo em conta as reais dimensões de um dispositivo. Esta ferramenta dispõe de funções para criar apresentações interativas que conseguem demonstrar o comportamento da aplicação como resultado final, dando também suporte à implementação.

O *draw.io* foi utilizado para os desenhos das arquiteturas do projeto tendo revelado grande auxílio, uma vez que, permite uma grande liberdade ilustrativa. Esta ferramenta permite conectar com o *github* o que proporciona a facilidade de guardar projetos e ter acesso a partir de qualquer dispositivo.

A engenharia de *software* foi realizada através da utilização *Visual Studio Paradigm*, esta é uma ferramenta muito completa que contém modelos e regras para a engenharia de *software*. Esta tornou-se um grande recurso no desenvolvimento da base de dados, dado que é possível desenhar o modelo e exportar para um ficheiro de criação de base de dados.

2.2 Plataforma Tecnológica

2.2.1 Web Scraper

Web scraping é terminologia dada à "(...) *construction of an agent to download, parse, and organize data from the web in an automated manner*(...)"(?). O grande problema com *web scraping* é que poderá ser considerado ilegal e é facilmente detetado. Tendo em conta este problema surgiram duas grandes formas de realizar *web scraping*. A forma mais comum de *web scraping* é realizar um pedido para obter uma página web e ler esta, sendo assim um processo rápido e simples.

A segunda forma de realizar *web scraping* é através da simulação da ação humana com da abertura do navegador programaticamente, pesquisa pela página desejada, descarregar e daí ler os dados. Este torna-se um processo lento e complexo.

A grande diferença entre estas duas formas é a velocidade, visto que a segunda forma tem de esperar que o navegador inicie, de seguida terá de esperar que a página carregue e apenas após este processo se poderá ler os dados.

2.2.1.1 Selenium

O *Selenium* é uma ferramenta "(...) *for a range of tools and libraries that enable and support the automation of web browsers*(...)"(?), esta provém "(...) *extensions to emulate user interaction with browsers*(...)"(?). Na sua base esta "(...) *is WebDriver, an interface to write instruction sets that can be run interchangeably in many browsers*(...)"(?). Esta ferramenta provém também a possibilidade de escalar com *multi threading*, o que permite abrir diversas janelas do navegador simultaneamente e obter dados destas o que diminui drasticamente o tempo de execução, esta funcionalidade não foi explorada devido a limitações de *hardware*, mas seria uma importante implementação futura.

2.2.2 Serviços Backend

Para a realização da integração entre a aplicação *frontend* e os dados, foi necessário desenvolver uma *Application Programming Interface (API)* para dar suporte a todos os serviços necessários para a aplicação. Uma *API* "(...)exposes a set of data and functions to facilitate interactions between computer programs and allow them to exchange information.(...)" (?). Estas ferramentas apesar de serem "(...)designed to work with other programs, they're mostly intended to be understood and used by humans writing those other programs(...)"(?).

2.2.2.1 Serviços RestFull e SOAP

Os serviços *RestFull* "(...)expose data as resources and use standard HTTP methods to represent Create, Read, Update, and Delete (CRUD) transactions against these resources(...)"(?), sendo que a resposta é no formato *JavaScript Object Notation (JSON)* "(...)due to its simplicity and ease of use with JavaScript, JSON has become the standard for modern APIs(...)"(?).

Já *SOAP* é "(...)XML-based envelope for the information being transferred, and a set of rules for translating application and platform-specific data types into XML representations(...)"(?) sendo que a resposta é realizada em *XML* leva a que "(...)It relies heavily on XML standards like XML Schema and XML Namespaces for its definition and function(...)"(?). A utilização de *XML* permite que "(...)two applications, regardless of operating system, programming language, or any other technical implementation detail, may openly share information using nothing more than a simple message encoded in a way that both applications understand(...)"(?).

Por fim foi decidido utilizar *RestFull* devido a ser *much lighter compared to SOAP*. *It does not require formats like headers to be included in the message, like it is required in SOAP architecture(...)"(?).* Outra maior valia é que "(...)it parses JSON, a human readable language designed to allow data exchange and making it easier to parse and use by the computer. It is estimated to be at around one hundred times faster than XML(...)"(?).

2.2.2.2 NodeJS

Para o desenvolvimento do projeto *backend* foi escolhido *NodeJS*, este surgiu quando "(...)the original developers took JavaScript, something you could usually only run inside the browser, and they let it run on your machine as a standalone process(...)"(?), isto significa que é possível correr código "(...)JavaScript outside the context of the browser(...)"(?).

Para correr *JavaScript* a nível de servidor *web* e a nível de computador pessoal é utilizado o mesmo motor, "(...)it's called the V8 JavaScript runtime engine(...)"(?), este é "(...)an opensource engine that takes JavaScript code and compiles it into much faster machine code. And that's a big part of what makes Node.js so fast(...)"(?).

NodeJs permite a utilização de bibliotecas externas "(...)by providing the Node Package Manager(...)"(?), este provém ao programador "(...)to easily install, manage, and even provide your own modules for a rapidly grown and well-maintained open source repository(...)"(?).

NodeJS foi escolhido para o *backend*, uma vez que, permite a utilização de *Typescript* para o desenvolvimento e por ser vastamente utilizado, o que dá acesso a diversas fontes de informação para a resolução de problemas, assim como, para o auxílio ao desenvolvimento.

2.2.2.3 Typescript

O *Typescript* "(...)is a bit unusual as a language in that it neither runs in an interpreter (as Python and Ruby do) nor compiles down to a lower-level language (as Java and C do).(...)"(?) isto porque este "(...)compiles to another high-level language, JavaScript(...)"(?), isto faz com que o *Typescript* seja visto como "(...)a superset of JavaScript in a syntactic sense(...)"(?).

Todos os programas *JavaScript* "(...)are TypeScript programs, but the converse is not true(...)"(?), "(...)This is because TypeScript adds additional syntax for specifying types(...)"(?). O sistema de tipagens do *TypeScript* tem como objetivo "(...)to detect code that will throw an exception at runtime, without having to run your code(...)"(?). "(...)The type checker cannot always spot code that will throw exceptions, but it will try(...)"(?).

Esta linguagem foi escolhida para o *backend*, visto que, assegura as tipagens, o que proporciona um maior nível de segurança quando se trabalha com dados recebidos, assim como, a agilização do processo de programação devido à capacidade de prever a maioria dos erros de código.

2.2.2.4 PostgreSQL

PostgreSQL "(...)is an open source object relational database management system(...)"(?). Esta "(...)emphasizes extensibility(...)"(?) o que permite a utilização de extensões desenvolvidas pela comunidade, mas "(...)Also, there are several extensions to access, manage, and monitor PostgreSQL clusters, such as pgAdmin III(...)"(?). Esta ferramenta é de aprendizagem simples devido ao facto de "(...)it complies with ANSI SQL standards(...)"(?), o que leva a que, qualquer indivíduo com conhecimentos prévios em *SQL* consiga facilmente aprender esta tecnologia.

PostgreSQL foi escolhido para a base de dados, dado que, a empresa já utiliza vastamente esta tecnologia, mas também porque é *open-source* e não existem custos associados para este tipo de utilização. A funcionalidade de extensões foi utilizada para implementar *id's*, que utilizam a estrutura *uuid*, o que dificulta o ataque aos dados, uma vez que, é difícil de prever os valores de *id's*.

2.2.2.5 Logs e Logging

Logs "(...)are a very useful source of information for computer system resource management (printers, disk systems, battery backup systems, operating systems, etc.), user and application management (login and logout, application access, etc.), and security(...)"(?). Um Log é "(...)what a computer system, device, software, etc. generates in response to some sort of stimuli. What exactly the stimuli are greatly depends on the source of the log message(...)"(?), ou seja, perante um estímulo desejado, um *log* poderá ser gerado. Os dados dos *logs* são "(...)the intrinsic meaning that a log message has(...)"(?), o que significa que estes contêm apenas dados relevantes ao objetivo do *log*. *Logging* é o nome que se dá ao processo de geração de *logs*.

Neste contexto *logging* poderá ser utilizado para realizar a monitorização de pedidos e erros. Estas informações poderão até auxiliar na toma de decisões sobre o *software* e em quais funcionalidades colocar mais atenção.

2.2.2.6 Morgan

Morgan é uma biblioteca que permite extrair dados de um pedido, assim como a criação de *logs*. Este atua como um *middleware* do servidor, que recebe qual o tipo de *log* a ser escrito, estes estão definidos na biblioteca. Os principais dados obtidos por este são, a data e hora do pedido, o tipo, o serviço, os dados recebidos, a resposta devolvida e também a descrição do sistema utilizado. Com estes dados é possível saber que plataforma é mais utilizada no *software*, quais as horas de maior utilização e quais os serviços mais executados. Estes dados permitem direcionar mais recursos para uma indicada plataforma e/ou serviço, assim como escolher os melhores horários de manutenção dos servidores.

2.2.2.7 Gestão de *emails*

O envio de *emails* para os utilizadores, foi desenvolvido através da biblioteca *node-mailer*, que permite a utilização de um servidor de *SMTP*. Esta ferramenta foi escolhida devido a ser uma das mais utilizadas para este tipo de necessidade, o que permite que exista mais informação sobre a mesma que auxilia a resolução e identificação de erros.

Para desenvolver o conteúdo dos *emails* foi utilizada a ferramenta *Tabular Email*, esta permite realizar o *design* do conteúdo de um *email*, sendo possível exportar para *html*. A maior dificuldade desta ferramenta é que não permite a utilização de acentuação, e visto que o *html* é gerado por uma máquina este torna-se complicado de navegar e traduzir.

2.2.2.8 Agendamento de tarefas

Um requisito deste projeto foi o envio diário de *emails* com o relatório de notificações ao final do dia. Primeiramente, para realizar o agendamento de tarefas, foi feita uma análise das ferramentas existentes para a realização deste tipo de ações. Deste modo, foram encontradas o *cronetab* e o *node-cron*. A grande diferença entre estas duas ferramentas é que o *cronetab* funciona a nível de servidor, sendo que, o funcionamento tem por base "(...)run this command at this time on this date(...)"(?), este comando poderá por exemplo executar um código para enviar *emails*. Já o *node-cron* trata-se de uma biblioteca de *NodeJs* "(...)in pure JavaScript for node.js based on GNU crontab(...)"(?), este permite o fácil agendamento de tarefas de forma programática, assim como a indicação do código a ser executado sem necessidade de criar comandos.

A hora de execução do código de envio de *emails* poderá variar e necessitar de reprogramação, pelo que, foi optada a utilização do *node-cron*, uma vez que, facilita a utilização e agiliza o processo de reprogramação de horas de envio do relatório de notificações.

2.2.2.9 Encriptação de *passwords*

Para garantir a segurança das *passwords* dos utilizadores é necessário encriptar estas, a encriptação poderá ser realizada manualmente ou com o auxílio de ferramentas, a grande diferença é que manualmente poderá não se obter uma encriptação tão segura como com o auxílio de uma ferramenta. Sendo assim, foi decidido utilizar uma ferramenta para encriptação de *passwords*, a ferramenta escolhida foi *bcrypt*. Esta foi escolhida devido a ser vastamente utilizada e tem por base a *hash bcrypt*, esta "(...)uses a variant of the Blowfish encryption algorithm's keying schedule, and introduces a work factor, which allows you to determine how expensive the hash function will be(...)"(?) isto permite que esta acompanhe a lei de Moore, pois "(...)As computers get faster you can increase the work factor and the hash will get slower(...)"(?). Esta ferramenta oferece um conjunto de métodos dos quais foram utilizados os de *hash* e de *compare*. O método de *hash* permite através de um valor, chamado *salt*, que não indica a complexidade a aplicar sobre a encriptação, sendo de seguida devolvida a *password* encriptada. O método *compare* permite comparar uma *password* encriptada com uma *password* não encriptada, e devolve verdadeiro ou falso conforme as *passwords* sejam iguais ou não.

2.2.2.10 Encriptação de configurações do servidor

Com o objetivo de garantir um nível de segurança maior foram realizadas pesquisas sobre as principais falhas de segurança no *Node.js*. Nestas, foi descoberto que as principais formas de ataque são as bibliotecas de *malware* e o ataque direto com o objetivo de obter dados de acesso a servidores que se encontram nos ficheiros de configuração.

Por norma, nas metodologias mais recentes de desenvolvimento de *software*, é sugerido que se coloque todas as configurações de servidores num ficheiro à parte, devido à praticidade de gerir estes dados, mas esta leva a um nível de segurança mais baixo, uma vez que, se alguém conseguir acesso a este ficheiro, consegue obter todos os dados de configuração de servidores. Neste projeto foi utilizado o ficheiro *env* para este fim, este no momento de iniciar o servidor é utilizado para carregar todas as variáveis para o ambiente do mesmo. Sendo assim qualquer um com acesso ao ficheiro ou às variáveis de ambiente poderá ver todas as configurações do servidor.

A solução mais indicada para este problema é a encriptação do ficheiro *env* e das variáveis de ambiente. A biblioteca mais utilizada para este objetivo é a *secur-env*, visto que, esta permite realizar a encriptação de um ficheiro com a indicação de uma *password*. A *password* deverá ser indicada no processo de inicialização do servidor de forma a ser possível ao mesmo descriptar o ficheiro, sendo que a gestão das variáveis cifradas passa então a estar encarregue desta biblioteca.

Embora exista esta solução, continuam a haver possibilidades de ataque, uma vez que, é possível ver o histórico do terminal do servidor, o que permite obter a *password* escrita. Para resolver este problema é indicada a biblioteca *readline*, pois esta possui o modo de *password* que apaga o histórico do terminal sempre que utilizado. Esta contém a vertente assíncrona, *readline* e a vertente síncrona, *readline-sync*. Neste projeto, foi utilizada a versão síncrona da biblioteca visto que o objetivo é o servidor apenas inicie após a indicação da *password* sem nenhum serviço a correr em simultâneo.

2.2.2.11 Firebase

Firebase é uma solução que foi comprada pela *Google* em 2014. O seu objetivo é "(...)to provide the tools and infrastructure that you need to build great apps(...)"(?), esta alcança este objetivo através da oferta de serviços pré configurados, sendo que "(...)many of the technologies are available at no cost(...)"(?).

Firestore também conhecida como *cloud storage*, é um serviço que dispõe "(...)a simple API that is backed up by Google Cloud Storage(...)"(?), que permite guardar e transmitir até um gigabyte de ficheiros de forma gratuita.

Cloud messaging é também um serviço do *Firebase* que permite "(...)to reliably deliver messages at no cost(...)"(?). Este garante que "Over 98% of connected devices receive these messages in less than 500ms(...)"(?). *Cloud messaging* permite a utilização de diversas formas de envio de notificações como "(...)driven by analytics to pick audiences, or using topics or other methods(...)"(?).

Dynamic links é um serviço do *Firebase* que permite a criação de "(...)links to an app that contain context about what you want the end user to see in the app(...)"(?).

Esta ferramenta foi escolhida devido à sua capacidade de fornecer serviços pré configurados de forma gratuita, o que evita o gasto monetário e a alocação de tempo para a configuração de servidores durante o desenvolvimento.

2.2.2.12 Axios

Para ser possível realizar pedidos a outros serviços externos como *Firebase*, é necessário utilizar uma biblioteca capaz do mesmo. Para isso foi optado por utilizar *Axios*. Esta é "(...)a promise-based HTTP Client for node.js(...)"(?) que "(...)uses the native node.js http module(...)"(?). Esta disponibiliza um conjunto de métodos para a realização de pedidos a serviços externos, assim como a configuração total dos mesmos.

2.2.3 Frontend

Um dos requisitos do projeto é o desenvolvimento do *frontend* com a utilização de *Flutter*, visto que a empresa no seu trabalho diário já utiliza esta ferramenta. Deste modo, foi fulcral a aprendizagem desta ferramenta e da sua linguagem de programação o *dart*.

2.2.3.1 Desenvolvimento cross-platform

O desenvolvimento de aplicações *cross-platform* ou multi-plataforma, consiste no desenvolvimento de uma aplicação para diversas plataformas e este pode ser realizado de diversas formas, mas as principais formas conhecidas são *WebView*, nativo e outras abordagens.

As *frameworks* nativas são "(...)the most stable choice for mobile application development(...)"(?) e dispõem de um grande comunidade e leque de aplicações desenvolvidas. O que torna estas *frameworks* estáveis é o facto de "(...)the app in this framework talks directly to the system(...)"(?). Todo o desenho no ecrã é realizado através de o que é chamado de *OEM components* que são disponibilizados pela *framework* mas não permitem customização total. A grande desvantagem desta abordagem é o facto de se o objetivo do projeto é o desenvolvimento para *iOS* e *Android*, então "(...)you need to learn two different languages(...)"(?), porque estas são utilizadas para "(...)write two different apps with the same functionalities(...)"(?) o que significa que "(...)every modification must be duplicated on both platforms(...)"(?).

Uma outra abordagem para o desenvolvimento para diversas plataformas através de uma única base de código é o *WebView*. "(...)Cordova-, Ionic-, PhoneGap-, and WebView-based frameworks in general are good examples of cross-platform frameworks(...)"(?), mas o grande problema desta abordagem é a "(...)lack in performance(...)"(?) pois esta é composta por um processo intermédio chamado *WebView* que renderiza código *HTML*, isto significa que "(...)the app is basically a website(...)"(?). Esta abordagem acrescenta também o componente de ponte que realiza o "(...)switch between JavaScript to the native system(...)"(?) para obter acesso aos serviços nativos.

Um concorrente à tecnologia mencionada na secção seguinte(2.2.3.2) é o *React Native*, este assim como as *frameworks* nativas "(...)heavily relies on OEM components(...)"(?) e "(...)expands the bridge concept in the WebView systems, and uses it not only for services, but also to build widgets(...)"(?), isto leva a grandes problemas em termos de performance devido a que "(...)a component may be built hundreds of times during an animation, but due to the expanded concept of the bridge, this component may slow down to a great extent(...)"(?).

2.2.3.2 Flutter

Flutter é uma *framework* desenvolvida pela *Google*, de início "(...)was an experiment, as the developers at Google were trying to remove a few compatibility supports from Chrome, to try to make it run smoother(...)"(?), por fim, acabaram por descobrir que "(...)they had something that rendered 20 times faster than Chrome did and saw that it had the potential to be something great(...)"(?). Em suma, *Google* desenvolveu "(...)a layered framework that communicated directly with the CPU and the GPU in order to allow the developer to customize the applications as much as possible(...)"(?).

Para o *Flutter* tudo é um *widget*, "(...)Orientation, layout, opacity, animation... everything is just a widget(...)"(?), isto permite que os utilizadores "(...)choose composition over inheritance, making the construction of an app as simple as building a Lego tower(...)"(?). Todos estes *widgets* oficiais estão identificados no catálogo de *widgets* do *Flutter*. Como tudo no *Flutter* é composto por *widgets* "(...)the more you learn how to use, create, and compose them, the better and faster you become at using Flutter(...)"(?).

A abordagem ao *cross-platform* realizada pelo *Flutter* é baseada em "(...)AOT (Ahead Of Time) instead of JIT (Just In Time) like the JavaScript solutions(...)"(?) mostradas anteriormente. Esta também permite a conversação direta com o *cpu* sem necessidade de ponte e "(...)does not rely on the OEM platform(...)"(?). Esta faculta que "(...)custom components to use all the pixels in the screen(...)"(?), o que significa que "(...)the app displays the same on every version of Android and iOS(...)"(?). Esta também utiliza "(...)Platform Channels to use the services(...)"(?), o que leva a que "(...)if you need to use a specific Android or iOS feature, you can do it easily(...)"(?).

2.2.3.3 Dart

Dart é a linguagem de programação utilizada pela *framework Flutter*, esta é "(...)a general purpose programming language(...)"(?) que foi desenhada para ser "(...)familiar to the vast majority of programmers(...)"(?). Esta linguagem é "(...)purely object-oriented(...)"o que significa que "(...)all values a Dart program manipulates at run time are objects(...)"(?), até tipos básicos como números e booleanos, esta é também "(...)class-based, optionally typed(...)"(?). Esta é opcionalmente tipada, o que significa que a decisão de utilizar tipagens cai sobre o programador, mas no caso de *Flutter*, na sua versão mais recente é recomendado a utilização de tipagens de variáveis. Por fim, esta "(...)supports mixin-based inheritance and actor-style concurrency(...)"(?).

2.2.3.4 Links de aplicações

Existem diferentes tipos de *links* sendo que para *mobile* é utilizado os *app links*, *deep links* e os *dynamic links*.

Os *app links* são "(...)web links that use the HTTP and HTTPS schemes(...)"(?), estes possuem também um atributo extra chamado *autoVerify*. Este atributo permite a uma aplicação "(...)to designate itself as the default handler of a given type of link(...)"(?), isto permite que "(...)app opens immediately if it's installed(...)"(?). O grande problema é que estes *links* não permitem o redirecionamento do utilizador para uma parte específica da aplicação e é necessário dispor de um domínio próprio.

Os *deep links* são "(...)URIs of any scheme that take users directly to a specific part of your app(...)"(?), o grande problema deste tipo de *links* é que se os utilizadores não dispuserem da aplicação instalada no dispositivo, este irá falhar e não permite a customização de comportamento.

Já os *dynamic links*, desenvolvidos pela *Firebase*, assim como os *deep links* "(...)if a user opens a Dynamic Link on iOS or Android, they can be taken directly to the linked content in your native app(...)"(?), mas para além disto, este permite que "(...)if a user opens the same Dynamic Link in a desktop browser, they can be taken to the equivalent content on your website(...)"(?), ou seja, este permite a customização de comportamento de *links* para diversas situações e em caso do utilizador não dispor da aplicação instalada, este permite que "(...)the user can be prompted to install it; then, after installation, your app starts and can access the link(...)"(?). Visto que este é o comportamento desejado pelo cliente da aplicação, então foi decidido utilizar esta abordagem.

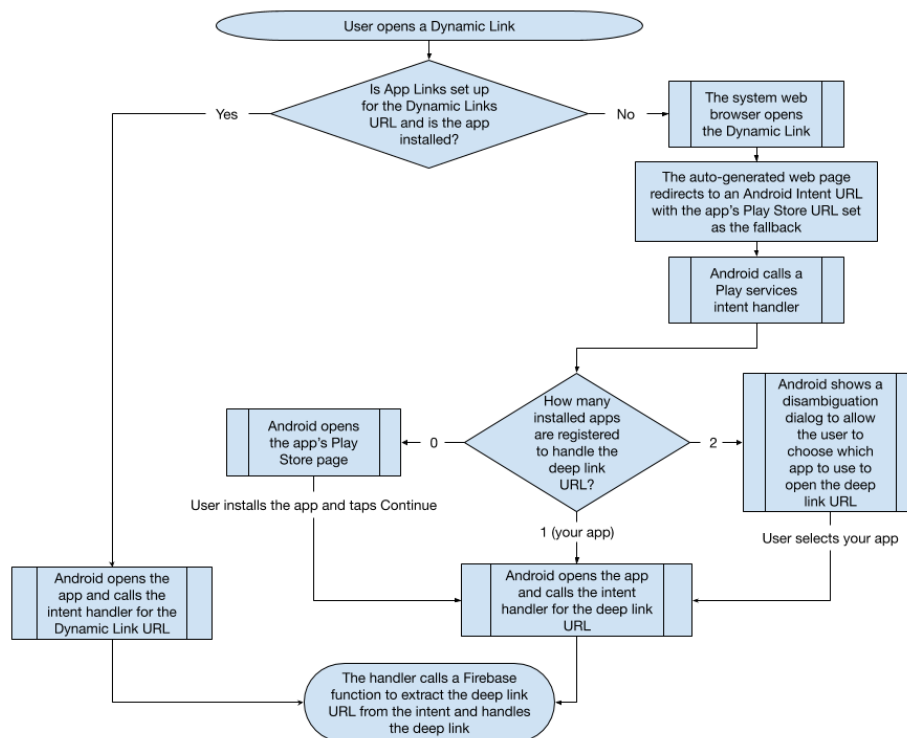


Figura 2.1: Funcionamento dos dynamic links em *Android* (?)

2.2.4 Qualidade de código

A qualidade do código desenvolvido é de extrema importância para possibilitar e facilitar a manutenção da solução desenvolvida, assim como a melhoria da segurança da mesma. Esta permite a percepção de objetivo de código e a estruturação do mesmo de acordo com normas estabelecidas perante a comunidade. A qualidade de código tem como objetivo diminuir a complexidade, pois nem sempre documentação e estruturação é o suficiente para o código ser de qualidade, sendo que, este deverá ser simplificado para ser interpretável. Todos estes pontos poderão ser implementados através do uso de *design patterns*, documentação e testes de código.

2.2.4.1 Design patterns

Os *design patterns* "(...)names, abstracts, and identifies the key aspects of a common design structure that make it useful for creating a reusable object-oriented design(...)"(?). Estes identificam "(...)the participating classes and instances, their roles and collaborations, and the distribution of responsibilities(...)"(?).

O *Model View Controller (MVC)* consiste em "(...)three kinds of objects(...)"(?), o *Model* que é "(...)the application object(...)"(?), a *View* que é a "(...)screen presentation(...)"(?) do *Model* e por fim o *Controller* que "(...)defines the way the user interface reacts to user input(...)"(?). Antes do *MVC* "(...)user interface designs tended to lump these objects together(...)"(?), o *MVC* "(...)decouples them to increase flexibility and reuse(...)"(?), através de "(...)establishing a subscribe/notify protocol between them(...)"(?).

2.2.4.2 Documentação

Para ser possível manter todo o projeto desenvolvido, foi criada documentação a diferentes níveis, sendo esta desenvolvida a nível de serviços com a explicação do objetivo do serviço e que dados este recebe, como também a nível de código sendo explicado o código desenvolvido em cada *script* existente. Para estes níveis de documentação foram utilizadas diferentes ferramentas, para documentação de serviços, foi utilizada a ferramenta *swagger* e para a documentação de código foi utilizado o *typedoc*.

2.2.4.3 Typedoc

Typedoc é um "(...)documentation generator for TypeScript(...)"(?), uma ferramenta "(...)which reads your TypeScript source files, parses comments contained within them, and generates a static site containing documentation(...)"(?). Esta utiliza chaves específicas para detetar as informações e criar categorias para melhor organizar toda a documentação. Esta documentação possibilita a interligação entre si mesma, o que permite ao visualizador seguir todo o processo.

2.2.4.4 Swagger

Swagger é uma ferramenta "(...)built around the OpenAPI Specification(...)"(?) que ajudam com "(...)design, build, document and consume REST APIs(...)"(?), o *OpenAPI* "(...)is an extensible API description format for REST APIs(...)"(?) que poderá ser "(...)written in YAML or JSON(...)"(?). Esta permite gerar documentação a nível de serviços que é acessível a partir do mesmo servidor com indicação de uma rota, o que evita outro servidor para hospedar a documentação. Esta documentação poderá ser gerada a partir de comentários a nível de código ou a partir de um ficheiro em formato *JSON* ou *yaml*, como mencionado anteriormente. Este ficheiro poderá ser mantido manualmente ou automaticamente. Neste projeto foi decidido manter este manualmente em *JSON*.

2.2.4.5 Testes de código

Aquando o fim do desenvolvimento de cada serviço é necessário testar este para verificar se a funcionalidade encontra-se de acordo com o desejado e/ou existem erros de código. Para realizar estes testes poderão ser utilizadas ferramentas de auxílio ou então poderão ser realizados manualmente. O grande problema dos testes manuais é a exaustividade, uma vez que, são longos e propensos a erros, pelo que, são realizados em menor escala. Para os testes deste projeto foram utilizadas ferramentas de auxílio, sendo as ferramentas escolhidas *mocha* e *chai*.

Mocha é uma *framework* de testes *javascript* que permite teste assíncrono. "(...)Mocha tests run serially, allowing for flexible and accurate reporting, while mapping uncaught exceptions to the correct test cases(...)"(?). Esta ferramenta foi escolhida devido à sua simplicidade e à capacidade de testar código *typescript* para além de *javascript*.

Chai é "(...)a BDD / TDD assertion library for node and the browser that can be delightfully paired with any javascript testing framework(...)"(?), neste caso esta foi utilizada em conjunto com *Mocha*. A utilização de *BDD* permite a utilização de uma "(...)expressive language & readable style(...)"(?), já *TDD* é "(...)more classical feel(...)"(?). Para tornar os testes de código mais interpretáveis foi explorado o estilo de testes *BDD*.

A realização de testes de código foi muito importante, visto que, com esta ferramenta foi possível encontrar erros de lógica de negócio, bem como, erros de código tanto a nível da formulação de respostas como a nível de código.

3. Análise e especificação

Na análise e especificação está contida a descrição do modelo de negócio, onde serão identificados os principais problemas do modelo atual e será indicada a necessidade de implementação do *software*. Nesta análise estarão identificados os objetivos de negócio e o principal impacto deste *software* neste modelo de negócio. Nesta análise estará também contida a engenharia de *software* com toda a especificação do projeto em mãos.

3.1 Modelo de negócio

Este projeto surgiu com a necessidade da Motorline melhorar a comunicação e a experiência das empresas clientes e técnicos. Atualmente sempre que um técnico possui uma questão este deverá contactar o suporte técnico o que pode levar a sobrelotação do sistema, ou então este deverá se juntar ao grupo do Facebook de técnicos e colocar neste a sua questão. Sempre que é necessário um manual de produto ou sempre que o utilizador pretender ver o catálogo este deverá deslocar-se ao *site* da Motorline e procurar o produto no catálogo onde não existe um acesso rápido. Como este processo acaba por não ser prático para o utilizador, foi decidido suprimir o problema com o projeto *Install & Go*. Com estas questões em mente surgiu então a ideia de implementar um fórum de resolução de problemas.

3.2 Objetivos de negócio

Este *software* visa minimizar o problema acima descrito, visto que, são efetuadas muitas questões por parte dos técnicos que são comuns entre si. Deste modo, surgiu a ideia de implementar um fórum onde o técnico poderá pesquisar por questões semelhantes à sua, e encontrar a solução sem necessidade de contactar o suporte técnico. O técnico poderá também expor a sua questão com o anexo de imagens, o que facilita o processo de resolução da questão. Para existir partilha de conhecimento este poderá responder a tópicos, pois caso encontre alguma questão que já tenha resolvido ajudará outro técnico. O técnico também poderá alterar a visibilidade da sua tópico caso deseje que apenas técnicos Motorline visualizem a sua questão.

A empresa poderá realizar as mesmas ações que o técnico, e também criar contas para os seus técnicos e realizar a gestão destas sendo importante permitir apagar e impedir acesso à conta em caso de necessidade.

3.3 Estudo de Soluções existentes

Para ser possível entender todo o negócio onde o projeto se encaixa foi realizado um estudo que engloba outras soluções do mesmo ramo. Estas soluções investigadas foram indicações do supervisor de estágio, pois este já tinha realizado um estudo da situação do negócio previamente. As soluções investigadas foram, *FixAndGo*, *My VDS* e *Ultimate Cardin*. A solução *FixAndGo* foi indicada devido à capacidade de calcular a possibilidade de instalar um motor com base em medidas indicadas pelo utilizador. Já a solução *My VDS* foi mencionada devido ao facto de dispor de documentação sobre os seus produtos como manuais, vídeos e uma forma de comunicação com a equipa de suporte. Por fim, a solução *Ultimate Cardin* foi utilizada como exemplo de *design* a não seguir, devido a ser confuso, esta foi também utilizada como exemplo de organização de design do ecrã de detalhes de produto.

3.4 Domínio de aplicação do sistema

Com o diagrama abaixo representado (Figura 3.1) é possível visualizar todos os atores do *software* e as suas ações, assim como os sistemas envolvidos na aplicação e as funções. Destes é possível identificar que este *software* contém três atores principais, o Utilizador que é um utilizador sem sessão iniciada, o Técnico, um utilizador com sessão iniciada, já a Empresa é uma empresa cliente da Motorline. Também é possível visualizar os diferentes sistemas integrados no projeto, como Servidor Motorline onde serão obtidas informações do catálogo de produtos, Servidor *Install & Go* onde estão todas as funções de suporte ao *software*, o Servidor de Imagens onde serão guardadas todas as imagens da aplicação e por fim o Servidor de *Email* que enviará *email* com o código de validação de conta para os clientes assim que se registarem no *software*.

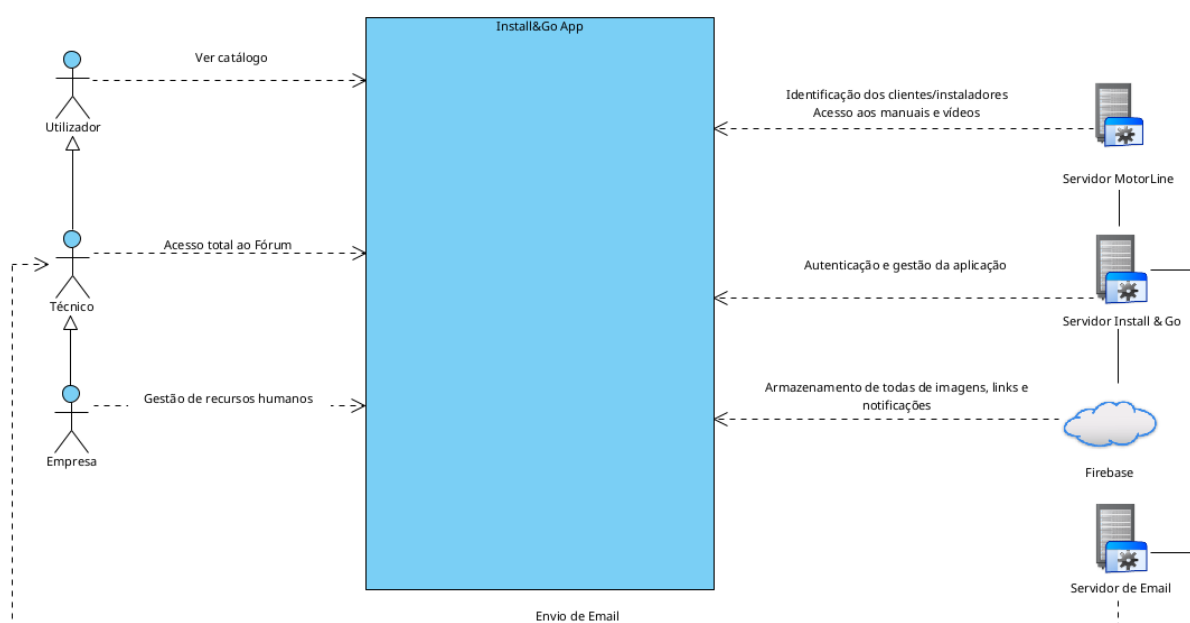


Figura 3.1: Diagrama de contexto da aplicação

3.5 Operações a realizar no sistema

A primeira tarefa a realizar no desenvolvimento desta etapa do projeto foi o levantamento dos Requisitos Funcionais (RF) (Tabela 3.1) e Requisitos Não Funcionais (RNF), tendo sido posteriormente validados com cliente. Este levantamento de requisitos funcionais, foi realizado num conjunto de reuniões realizadas com os clientes do projeto.

Tabela 3.1: Tabela de requisitos funcionais

#	Descrição	Fonte	Data
Autenticação			
RF01	O Utilizador deverá conseguir visualizar o catálogo	Helder Remelhe	2/13/2023
RF02	A Empresa deverá conseguir realizar o registo na aplicação	Helder Remelhe	2/13/2023
RF03	O Técnico deverá conseguir realizar o login na aplicação	Helder Remelhe	2/13/2023
RF04	O login deverá ser realizado utilizando número de contribuinte e <i>password</i>	Helder Remelhe	2/13/2023
RF05	Assim que o registo é realizado, a motor-line deverá validar a conta sendo posteriormente enviado um <i>email</i> para a empresa ativar e utilizar a conta	Rafael Viana	2/13/2023
RF06	O Técnico deverá conseguir pedir reenvio de código de verificação de conta	Rafael Viana	2/27/2023
RF07	Os Técnicos deverão ser identificados como técnicos certificados ou técnicos oficiais	Helder Remelhe	2/27/2023
Fórum			
RF08	O Técnico deverá conseguir aceder ao fórum e realizar operações	Helder Remelhe	2/13/2023
RF09	O Técnico deverá conseguir visualizar os tópicos mais recentes	Brainstorming	2/14/2023
RF10	O Técnico deverá conseguir visualizar os tópicos em destaque	Brainstorming	2/14/2023
RF11	O Técnico deverá conseguir visualizar os tópicos por responder	Brainstorming	2/14/2023
RF12	O Técnico deverá conseguir acessar aos tópicos privados do fórum	Helder Remelhe	2/13/2023
RF13	O Técnico deverá conseguir pesquisar por tópicos referentes a um assunto desejado	Brainstorming	2/14/2023

Continued on next page

Tabela 3.1: Tabela de requisitos funcionais (Continued)

RF14	O Técnico deverá conseguir pesquisar por tópicos referentes a um produto desejado	Brainstorming	2/14/2023
RF15	O Técnico deverá conseguir pesquisar por tópicos referentes a um produto por código QR	Brainstorming	2/14/2023
RF16	O Técnico deverá conseguir visualizar os seus tópicos	Brainstorming	2/14/2023
	Criar Tópico		
RF17	O Técnico deverá conseguir criar tópicos para expor a sua questão	Helder Remelhe	2/13/2023
RF18	O Técnico deverá conseguir colocar o seu tópico público ou privado para assim apenas outras empresas conseguirem ver	Helder Remelhe	2/13/2023
RF19	O Técnico deverá conseguir indicar tipo de tópico para agilizar a identificação do mesmo	Helder Remelhe	2/14/2023
RF20	O Técnico deverá conseguir indicar o produto referente ao tópico para agilizar a identificação da sua questão	Brainstorming	2/14/2023
RF21	O Técnico deverá conseguir anexar imagens ou vídeos ao tópico em questão para agilizar a comunicação e identificação do seu problema	Helder Remelhe	2/13/2023
	Gestão de tópico		
RF22	O Técnico deverá conseguir indicar qual a melhor resposta ao seu tópico para assim facilitar a pesquisa por resposta a outras empresas que possuam o mesmo problema	Helder Remelhe	2/14/2023
RF23	O Técnico deverá conseguir indicar que o tópico se encontra finalizado quando o seu problema se encontrar resolvido	Helder Remelhe	2/13/2023
RF24	O Técnico deverá conseguir remover o seu tópico	Brainstorming	2/14/2023
RF25	O Técnico deverá conseguir alterar a visibilidade do seu tópico	Helder Remelhe	2/13/2023
	Tópicos		
RF26	O Técnico deverá conseguir ver todas as respostas ao tópico	Helder Remelhe	2/13/2023
RF27	O Técnico deverá conseguir gostar do tópico para dar destaque ao mesmo	Brainstorming	2/14/2023

Continued on next page

Tabela 3.1: Tabela de requisitos funcionais (Continued)

RF28	O Técnico deverá conseguir remover uma resposta que colocou em um tópico	Brainstorming	2/14/2023
	Respostas a Tópicos		
RF29	O Técnico deverá conseguir comentar tópicos	Helder Remelhe	2/13/2023
RF30	O Técnico deverá conseguir responder a comentários de tópicos	Helder Remelhe	2/13/2023
RF31	O Técnico deverá conseguir anexar imagens e videos à sua resposta	Helder Remelhe	2/13/2023
RF32	O Técnico deverá conseguir gostar de alguma resposta para dar destaque a esta resposta	Helder Remelhe	2/13/2023
	Perfil		
RF33	O Técnico deverá conseguir alterar o seu <i>email</i>	Helder Remelhe	2/27/2023
RF34	O Técnico deverá conseguir alterar imagem de perfil	Brainstorming	2/27/2023
RF35	O Técnico deverá conseguir alterar nome	Brainstorming	2/27/2023
RF36	O Técnico deverá ser identificado como Técnico oficial ou certificado	Helder Remelhe	2/28/2023
	Notificações		
RF37	O Técnico deverá conseguir receber notificações por <i>email</i> e/ou push	Rafael Viana	2/27/2023
RF38	O Técnico deverá conseguir alterar o tipo de notificação ente relatório diário e notificação direta para ambos os tipos	Rafael Viana	2/27/2023
RF39	O Técnico deverá conseguir ver as suas notificações	Helder Remelhe	2/27/2023
RF40	O Técnico deverá conseguir apagar as suas notificações	Helder Remelhe	2/27/2023
	Gestão de recursos humanos		
RF41	A Empresa deverá conseguir criar contas para os seus técnicos	Brainstorming	2/13/2023
RF42	Assim que a conta de técnico for criada, este deverá receber um <i>email</i> para registar as restantes informações e ativar a conta	Helder Remelhe	2/27/2023
RF43	A Empresa deverá conseguir impedir acesso a uma conta de técnico	Helder Remelhe	2/27/2023

Continued on next page

Tabela 3.1: Tabela de requisitos funcionais (Continued)

RF44	A Empresa deverá conseguir remover uma conta de técnico da empresa	Helder Remelhe	2/27/2023
------	--------------------------------------------------------------------	----------------	-----------

3.6 Descrição dos intervenientes

O projeto envolve um conjunto de intervenientes, sendo estes, o utilizador, a empresa e o técnico. Estes desempenham um papel fundamental e podem realizar diferentes ações.

O utilizador, sem sessão iniciada terá apenas acesso ao catálogo de produtos.

O técnico conseguirá realizar as mesmas ações que o utilizador, mas este ator conseguirá também ter acesso total ao fórum e se for técnico oficial tem também acesso a questões privadas. O fórum permite expor questões, com auxílio de imagens, a ligação da questão a uma categoria e um produto para facilitar a sua resolução. As questões poderão ser públicas, ou então privadas, para apenas técnicos oficiais conseguirem ver. Assim que o técnico estiver satisfeito com a questão poderá indicar a melhor resposta obtida ao problema sendo esta destacada e a tópico finalizada.

O técnico pode realizar pesquisas por questões, onde evita um telefonema ou o preenchimento de um formulário para contactar um técnico oficial. Com estas pesquisas poderá responder a outras questões. As respostas podem conter imagens anexadas e responder a outras respostas para manter comunicação. O técnico poderá destacar tópicos e respostas de tópicos com um *like*.

A empresa pode realizar a gestão de contas de técnicos, onde cria, impede acesso e elimina a conta em caso de necessidade.

3.7 Partes Interessadas

Este projeto foi proposto pelo supervisor de estágio, na empresa Motorline representada por Rafael Viana e André Viana, as partes interessadas.

3.8 Condições Específicas

Ao longo de um conjunto de reuniões foram levantados os requisitos não funcionais do projeto representados na Tabela 3.2.

Tabela 3.2: Tabela de requisitos não funcionais

#	Tipo	Descrição
RNF01	Cultural	O <i>software</i> deverá suportar vários idiomas (prioritariamente, português e inglês)
RNF02	Configuração	A falha dos servidores, implica a inutilidade total do fórum da aplicação
RNF03	Conexão	Necessário o uso de dados móveis ou WIFI
RNF04	Segurança	O <i>software</i> tem de ser seguro para proteger os dados do consumidor
RNF05	Desenvolvimento	A aplicação deverá ser desenvolvida com a utilização da tecnologia <i>Flutter</i>

3.9 Esquematização do conteúdo das páginas

Para ser possibilitar a percepção dos dados necessários para alimentar o *software*, o que apresentar em cada página e também como navegar entre os ecrãs da aplicação foi então desenhado um esquema (Figura 3.2).

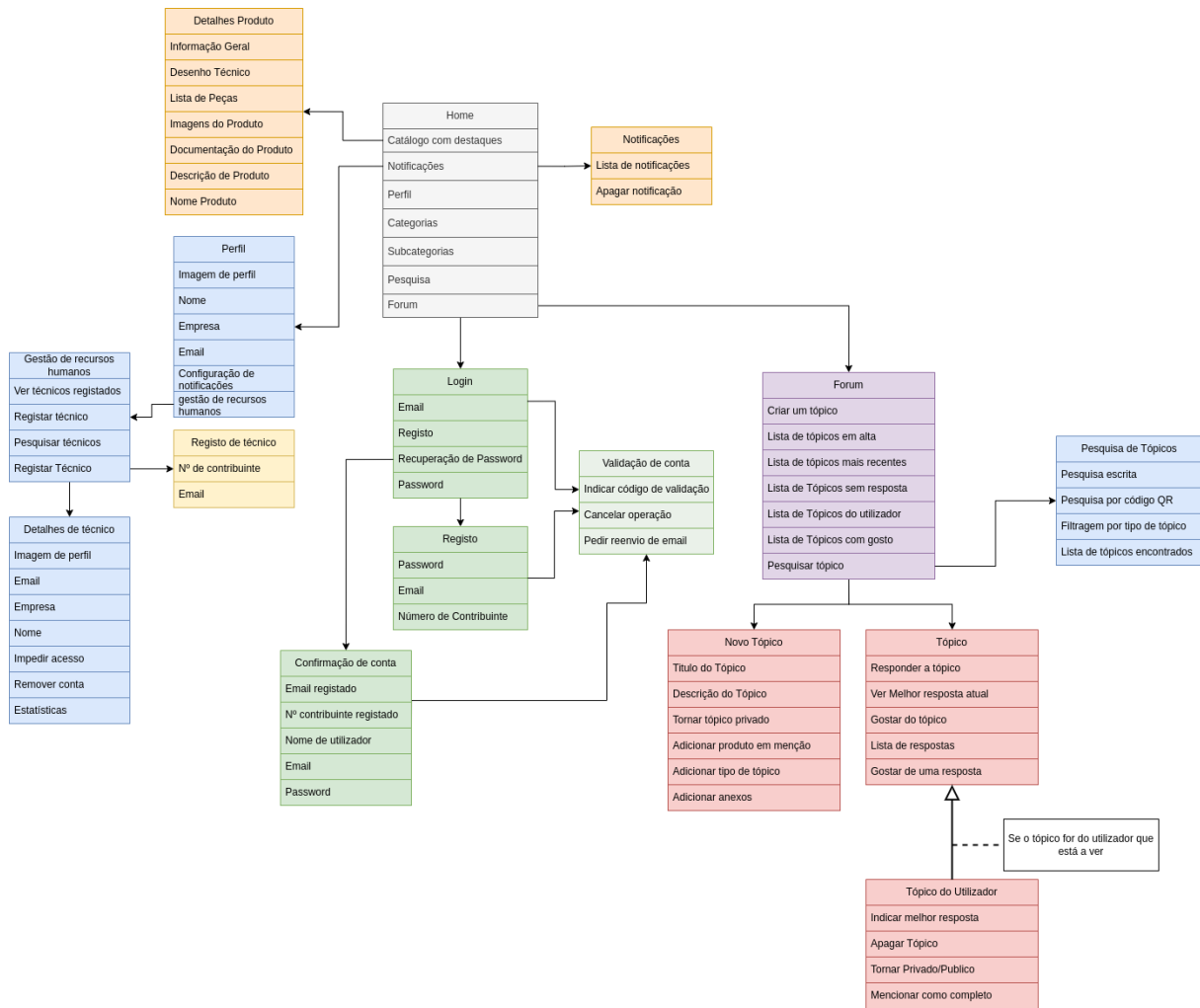


Figura 3.2: Esquema de organização de páginas do *software*

3.9.1 Autenticação e página Inicial

Através deste esquema é possível perceber que do ecrã principal, o utilizador tem acesso ao catálogo de produtos e ao fórum, neste pode também realizar o *login* e o *logout* que redirecionam para os respetivos ecrãs.

No ecrã de *login* é necessário o utilizador indicar o número de contribuinte e a *password*, neste ele pode também pedir recuperação de *password* e/ou redirecionar para o registo onde necessitará de número de contribuinte, *password* e *email* para o realizar.

Em caso de o utilizador não possuir a conta ativa, este será encaminhado para o ecrã de validação conta em que deverá indicar o código de validação, cancelar a operação e pedir o reenvio do código de validação.

Em caso de se tratar de um técnico que necessita de confirmar a sua conta, este conseguirá ver as informações registadas, introduzir o seu nome, alterar o seu *email* e *password*.

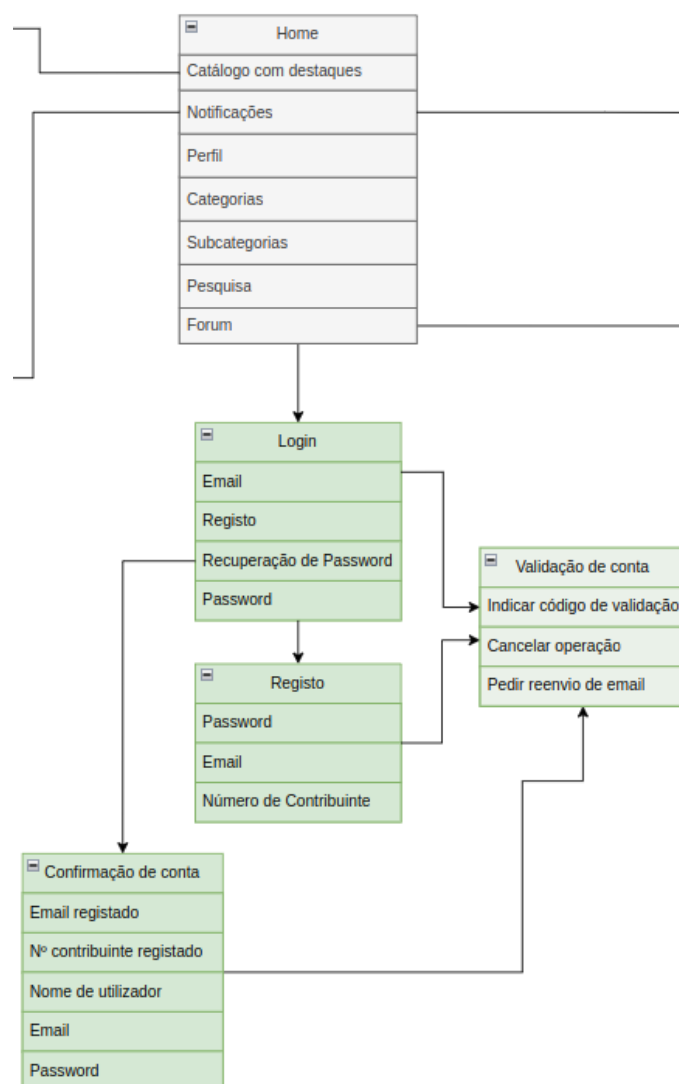


Figura 3.3: Esquema de organização das páginas de autenticação e página inicial

3.9.2 Fórum

Através do ecrã inicial o utilizador pode direccionar-se para o ecrã do fórum. Neste ecrã, poderá pesquisar por tópicos, ou então aceder a tópicos em alta, mais recentes ou sem resposta. O técnico consegue também criar e ver as seus tópicos.

3.9.3 Criar nova tópico

Quando o técnico decide criar uma tópico, este tem de indicar o título e a descrição, de seguida poderá indicar se é privado ou não, o tipo de tópico, o produto referente e adicionar anexos.

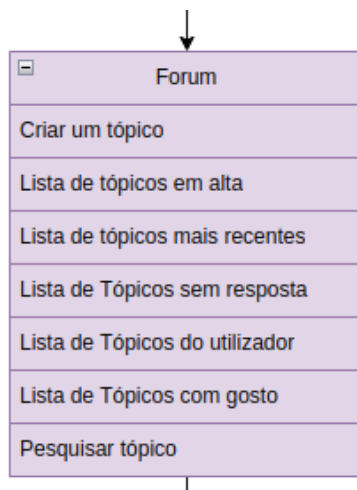


Figura 3.4: Esquema de organização da página de fórum

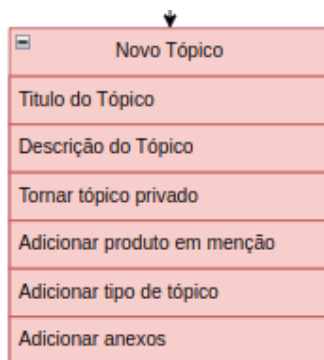


Figura 3.5: Esquema de organização da página de criação de tópicos

3.9.4 Detalhes de tópicos

O técnico pode também ver os detalhes da tópico, responder, gostar e gostar de uma resposta. Caso esta seja do mesmo, este pode indicar a melhor resposta, apagar a tópico, tornar pública ou privada e indicar como completa.

3.9.5 Pesquisa de tópicos

A página de pesquisa permite ao técnico procurar por tópicos específicos tanto por nome como por produto. Para além da pesquisa o utilizador pode também realizar a filtragem dos tópicos por tipo e categoria.

3.9.6 Notificações

A página de notificações permite ao técnico visualizar as suas notificações, assim como também apagar.

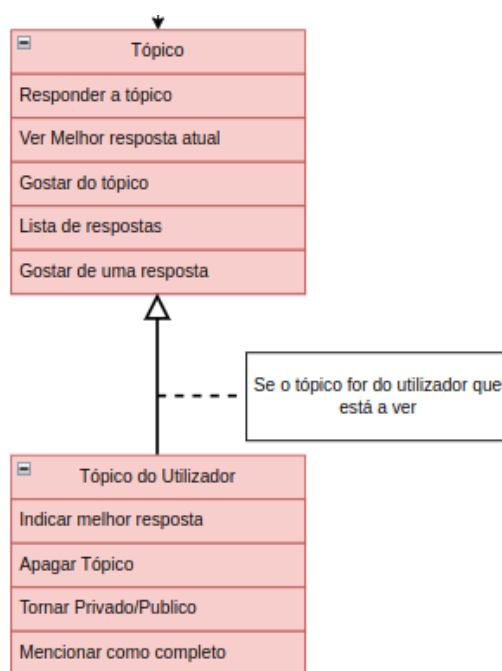


Figura 3.6: Esquema de organização da página de detalhes de tópico

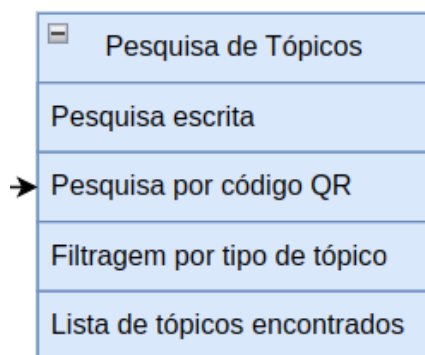


Figura 3.7: Esquema de organização da página de pesquisa de tópicos

3.9.7 Perfil

A página de perfil de técnico permite visualizar as suas informações, assim como alterar o seu *email* e configurar as notificações. Caso se trate de uma empresa a visualizar o seu perfil, esta poderá ter acesso à gestão de recursos humanos, para gerir os seus técnicos.

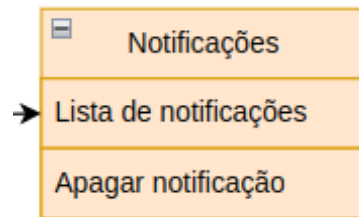


Figura 3.8: Esquema de organização da página de notificações

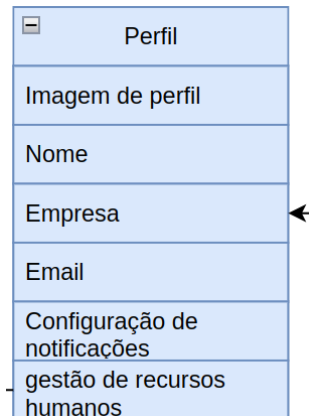


Figura 3.9: Esquema de organização da página de perfil

3.9.8 Gestão de recursos humanos

A página de gestão de recursos humanos permite à empresa gerir todos os seus técnicos registados e criar contas. Assim que a empresa seleciona um técnico, esta vê o seu perfil, com estatísticas.

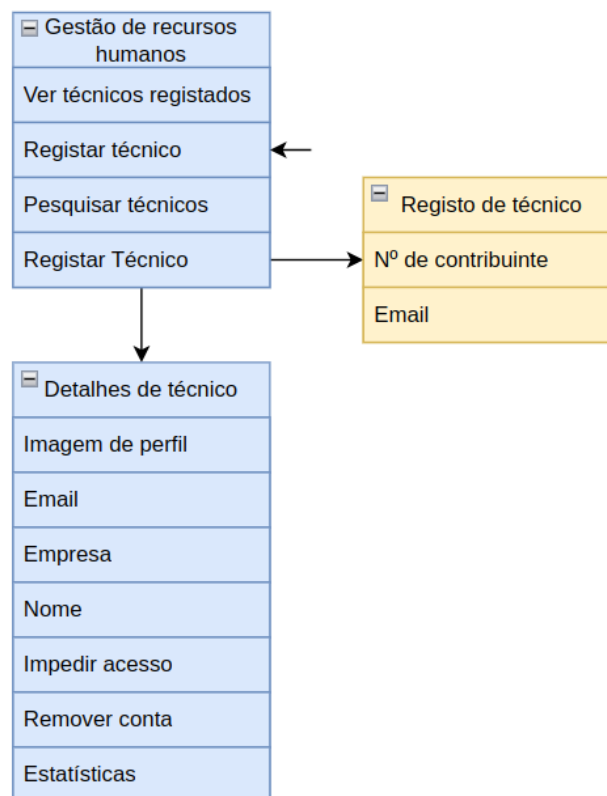


Figura 3.10: Esquema de organização da página de gestão de recursos humanos

3.10 User Stories

Antes de desenvolver os casos de uso foram criadas *User Stories (US)* para ser possível descrever os objetivos ao realizar uma ação.

Tabela 3.3: Tabela de *user stories*

#	Ator	Descrição
Autenticação		
US01	Utilizador	Eu como Utilizador, quero conseguir utilizar a aplicação sem realizar o login
US02	Empresa	Eu como Empresa, quero conseguir realizar o registo na aplicação
US03	Técnico	Eu como Técnico, quero conseguir realizar o login na aplicação utilizando o número de contribuinte e <i>password</i>
US04	Técnico	Eu como Técnico, quero conseguir pedir reenvio de código de ativação de conta caso eu não receba o código
US05	Técnico	Eu como Técnico, quero conseguir ser identificado como tal na aplicação

Continued on next page

Tabela 3.3: Tabela de *user stories* (Continued)

	Fórum	
US06	Técnico	Eu como Técnico, quero conseguir acessar ao fórum
US07	Técnico	Eu como Técnico, quero conseguir visualizar os tópicos mais recentes, de forma a conseguir ver os mais falados no dia atual
US08	Técnico	Eu como Técnico, quero conseguir visualizar os tópicos em destaque, de forma a ver quais são mais falados
US09	Técnico	Eu como Técnico, quero conseguir ver os meus tópicos de forma a conseguir aceder a estes facilmente
US10	Técnico	Eu como Técnico, quero conseguir visualizar os tópicos por responder, de forma a conseguir ajudar alguém com maior facilidade
US11	Técnico	Eu como Técnico, quero conseguir visualizar os tópicos privados
US12	Técnico	Eu como Técnico, quero conseguir realizar filtragem de tópicos por tipo
US13	Técnico	Eu como Técnico, quero conseguir pesquisar por um tópico relativo a um assunto de forma a obter a solução
US14	Técnico	Eu como Técnico, quero conseguir pesquisar por um tópico relativo a um produto de forma a encontrar questões comuns a este
US15	Técnico	Eu como Técnico, quero conseguir pesquisar por código QR de um produto de forma a encontrar tópicos referentes ao mesmo mais facilmente
	Criar Tópico	
US16	Técnico	Eu como Técnico, quero conseguir criar tópicos de forma a conseguir expor questões
US17	Técnico	Eu como Técnico, quero conseguir indicar se o meu tópico é público ou privado, de forma a conseguir respostas de qualquer cliente, ou apenas de técnicos
US18	Técnico	Eu como Técnico, quero conseguir indicar o tipo de tópico em que o este se enquadra de forma a facilitar a sua identificação
US19	Técnico	Eu como Técnico, quero conseguir indicar o produto referente ao tópico para facilitar a identificação do mesmo
US20	Técnico	Eu como Técnico, quero conseguir anexar imagens ao tópico de forma a facilitar a comunicação e identificação do problema
	Gestão de Tópico	
US21	Técnico	Eu como Técnico quero conseguir indicar qual a melhor resposta ao meu tópico de forma a facilitar o encontro da solução do problema a outros clientes ou técnicos com o mesmo problema
US22	Técnico	Eu como Técnico quero conseguir indicar que o tópico se encontra finalizado quando o problema está resolvido

Continued on next page

Tabela 3.3: Tabela de *user stories* (Continued)

US23	Técnico	Eu como Técnico quero conseguir remover o meu tópico
US24	Técnico	Eu como Técnico quero conseguir alterar a visibilidade do meu tópico
	Tópicos	
US25	Técnico	Eu como Técnico, quero conseguir ver todas as respostas a um tópico
US26	Técnico	Eu como Técnico, quero conseguir gostar de um tópico caso o ache relevante
US27	Técnico	Eu como Técnico, quero conseguir apagar uma resposta minha
	Respostas a Tópicos	
US28	Técnico	Eu como Técnico, quero conseguir comentar um tópico de forma a dar a minha resposta
US29	Técnico	Eu como Técnico, quero conseguir responder a um comentário de forma a comunicar
US30	Técnico	Eu como Técnico, quero conseguir anexar imagens ao meu comentário
US31	Técnico	Eu como Técnico, quero conseguir gostar de um comentário caso ache este relevante
	Perfil	
US32	Técnico	Eu como Técnico quero conseguir alterar o meu <i>email</i>
US33	Técnico	Eu como Técnico quero conseguir alterar a minha imagem de perfil
US34	Técnico	Eu como Técnico quero conseguir alterar o meu nome
	Notificações	
US35	Técnico	Eu como Técnico quero conseguir receber notificações por <i>email</i> e/ou push de forma a manter-me atualizado das minhas questões
US36	Técnico	Eu como Técnico quero conseguir alterar o tipo de notificação que recebo entre relatório diário e tempo real
US37	Técnico	Eu como Técnico quero conseguir apagar as minhas notificações de forma a evitar aglomeração
	Gestão de recursos humanos	
US38	Empresa	Eu como Empresa quero conseguir criar conta para os meus técnicos utilizarem o fórum
US39	Empresa	Eu como Empresa quero conseguir impedir acesso a uma conta de técnico
US40	Empresa	Eu como Empresa quero conseguir remover uma conta de técnico em caso de este já não pertencer à empresa

3.11 Casos de uso

Para transformar as *user stories* em ações e especificar todas as reações do sistema com o qual o ator interage foram desenvolvidos casos de uso (*Use Cases (UC)*).

Tabela 3.4: Tabela de casos de uso

#	User Story	Ator	Nome	Descrição
Criar tópico				
UC 1.0	US15	Técnico	Criar novo tópico	Criação de um novo tópico no fórum
Pesquisa de tópicos				
UC 1.1	US11	Técnico	Pesquisar tópicos específicos	Pesquisar por tópicos no fórum
UC 1.1.1	US12 e US13	Técnico	Pesquisa escrita	Pesquisar tópicos por assunto
UC 1.1.2	US14	Técnico	Pesquisa por código QR	Pesquisar tópicos referentes a um produto
Listagens de tópicos				
UC 1.2	US05	Técnico	Ver tópicos	Ver listagens de tópicos do fórum
Detalhes de tópico				
UC 1.3	US08	Técnico	Selecionar tópico	Ver detalhes de um tópico selecionado
UC 1.3.1	US21	Técnico	Finalizar tópico	Finalizar um tópico para indicar que está respondido
UC 1.3.2	US20	Técnico	Selecionar melhor resposta	Selecionar a melhor resposta do tópico
UC 1.3.3	US22	Técnico	Eliminar tópico	Eliminar um tópico do fórum
UC 1.3.4	US23	Técnico	Alterar visibilidade do tópico	Alterar a visibilidade de um tópico entre publico e privado
UC 1.3.5	US28	Técnico	Comentar o tópico	Comentar um tópico
UC 1.3.6	US25	Técnico	Gostar de tópico	Gostar de um tópico
Comentários				
UC 1.3.7	US24	Técnico	Ver comentários	Ver comentários do tópico
UC 1.3.7.1	US27	Técnico	Apagar comentário	Apagar comentário de um tópico
UC 1.3.7.2	US29	Técnico	Responder a comentário	Responder a um comentário de um tópico

Continued on next page

Tabela 3.4: Tabela de casos de uso (Continued)

UC 1.3.7.3	US31	Técnico	Gostar de comentá- rio	Gostar de um comentário
Ativação de conta				
UC 1.4	-	Técnico	Ativação de conta	Ativar conta de cliente
UC 1.4.1	US04	Técnico	Pedir reenvio de có- digo de ativação	Pedir reenvio de <i>email</i> de có- digo de ativação
Perfil				
UC 1.5	US31 - US32 - US33	Técnico	Ver Perfil	Ver perfil de utilizador
Notificações				
UC1.6	US34 e US36	Técnico	Ver notificações	Ver todas as notificações
UC1.7	US34 e US35	Técnico	Configuração de notificações	Configurar o modo e tipo de notificações a receber
Gestão de recursos humanos				
UC1.8	US37	Empresa	Registar Técnico	Registar conta de técnico da empresa
UC1.9	US38	Empresa	Impedir acesso a técnico	Registar conta de técnico da empresa
UC1.10	US39	Empresa	Remover conta de técnico	Registar conta de técnico da empresa

3.11.1 Especificação de casos de uso

Para demonstrar todas as interações entre os atores e o sistema, assim como todas as ações e fluxos possíveis, foram realizadas especificações de casos de uso.

3.11.1.1 Especificação de caso de uso de listagem de tópicos

Através da listagem de tópicos é possível visualizar todas as listagens que o utilizador poderá visualizar, sendo que, o técnico oficial consegue para além destas listagens, ver os seus tópicos e os tópicos privados.

Tabela 3.5: Tabela de especificação de caso de uso de listagem de tópicos do técnico

Caso de Uso	Ver listagem de tópicos	
Descrição	Ver a listagem de tópicos existentes no fórum por diversas categorias	
Ator	Técnico	
Pré-condição	-	
Pós-condição	-	
	Ator	Sistema
Fluxo Principal	1-Ver tópicos populares	
		2-Lista de tópicos populares
Fluxo Alternativo(A1)	1-Ver tópicos mais recentes	
		2-Lista de tópicos mais recentes
Fluxo Alternativo(A2)	1-Ver meus tópicos	
		2-Lista de tópicos do técnico
Fluxo Alternativo(A3)	1-Ver tópicos por responder	
		2-Lista de tópicos por responder

3.11.1.2 Especificação de caso de uso de criar novo tópico

Aquando a criação de um tópico, um técnico poderá realizar diversas ações sendo que obrigatoriamente terá de indicar o título, descrição e tipo de tópico, para além desta informação o técnico poderá anexar imagens, referenciar um produto e indicar a visibilidade.

Tabela 3.6: Tabela de especificação de caso de uso login

Caso de Uso	Criar novo tópico	
Descrição	Criação de um novo tópico no fórum	
Ator	Técnico	
Pré-condição	Clicar em adicionar novo tópico	
Pós-condição	-	
	Ator	Sistema
Fluxo Principal	1-Indicar o título do tópico	
	2-Indicar a descrição do tópico	
	3-Indicar se o tópico é privado	
	4-Indicar o tipo do tópico	
	5-Indicar o produto referido no tópico	
	6-Adicionar imagens de anexo	
	7-Confirmar a criação do tópico	
		8-Verificar se titulo está inserido
		9-Verificar se descrição está inserida
		10-Inserir novo tópico no fórum
Fluxo Alternativo(A1)	1-Cancelar a criação do tópico	
Fluxo Alternativo(A2)	1-Indicar o título do tópico	
	2-Indicar se o tópico é privado	
	3-Confirmar a criação do tópico	
		4-Verificar se titulo está inserido
		5-Verificar se descrição está inserida
		6-Erro descrição em falta

3.11.1.3 Especificação de caso de uso de pesquisar tópicos por escrito

Assim que um técnico deseje pesquisar por um assunto específico de tópico este poderá realizar uma pesquisa escrita onde conseguirá realizar filtragem por tipo e categoria de tópico.

Tabela 3.7: Tabela de especificação de caso de uso de pesquisa por escrito

Caso de Uso	Pesquisa por escrita	
Descrição	Pesquisar por tópicos no fórum	
Ator	Utilizador	
Pré-condição	Selecionar pesquisa de fórum	
Pós-condição	-	
	Ator	Sistema
Fluxo Principal	1-Pesquisar assunto	
		2-Lista de tópicos do assunto
	3-Filtrar por tipo	
		3-Filtragem de tópicos do assunto por tipo
Fluxo Alternativo(A1)	1-Pesquisar assunto	
		2-Lista de tópicos do assunto

3.11.1.4 Especificação de caso de uso de ver finalizar tópico

Quando um técnico encontra-se satisfeito com a solução do problema este poderá indicar que o tópico está finalizado, o que é sinalizado para outros técnicos.

Tabela 3.8: Tabela de especificação de caso de uso de finalizar tópico

Caso de Uso	Finalizar tópico	
Descrição	Finalizar um tópico para indicar que está respondido	
Ator	Técnico	
Pré-condição	Clicar no tópico desejado	
Pós-condição	Alterar tópico para finalizado	
	Ator	Sistema
Fluxo Principal	1-Clicar em finalizar tópico	
		2-Alterar tópico para finalizado
Fluxo Alternativo(A1)	-	-

3.11.1.5 Especificação de caso de uso de selecionar melhor resposta

Sempre que o técnico encontrar uma resposta no seu tópico que se destaca na solução da sua questão, este poderá colocar esta resposta como a melhor resposta do tópico. Caso já exista uma melhor resposta, esta automaticamente é removida e a nova é colocada como melhor resposta.

Tabela 3.9: Tabela de especificação de caso de uso de selecionar melhor resposta

Caso de Uso	Selecionar melhor resposta	
Descrição	Selecionar a melhor resposta do tópico	
Ator	Técnico	
Pré-condição	Clicar no tópico desejado	
Pós-condição	Alterar a resposta para melhor resposta	
	Ator	Sistema
Fluxo Principal	1-Clicar em melhor resposta	
		2-Verificar se já existe uma melhor resposta - Não
		3- Colocar a resposta como melhor resposta do tópico
Fluxo Alternativo(A1)	1-Clicar em melhor resposta	
		2-Verificar se já existe uma melhor resposta - Sim
		3- Verificar se a resposta existente é a mesma selecionada - Não
		4-Alterar melhor resposta
Fluxo Alternativo(A2)	1-Clicar em melhor resposta	
		2-Verificar se já existe uma melhor resposta - Sim
		3- Verificar se a resposta existente é a mesma selecionada - Sim
		4-Remover melhor resposta

3.11.1.6 Especificação de caso de uso de eliminar tópico

O técnico sempre que desejar poderá eliminar o tópico, o que permite remover do fórum e não volta a ser apresentado.

Tabela 3.10: Tabela de especificação do caso de uso de eliminar tópico

Caso de Uso	Eliminar tópico	
Descrição	Eliminar um tópico do fórum	
Ator	Técnico	
Pré-condição	Clicar no tópico desejado	
Pós-condição	Remoção do tópico	
	Ator	Sistema
Fluxo Principal	1-Clicar em remover tópico	
		3-Remover o tópico
Fluxo Alternativo(A1)	-	-

3.11.1.7 Especificação de caso de uso de alterar visibilidade de um tópico

Quando um técnico publica um tópico este pode desejar alterar a sua visibilidade para apenas técnicos oficiais ou todos os técnicos.

Tabela 3.11: Tabela de especificação de caso de uso de alteração de visibilidade de um tópico

Caso de Uso	Alterar visibilidade do tópico	
Descrição	Alterar a visibilidade de um tópico entre público e privado	
Ator	Técnico	
Pré-condição	Clicar no tópico desejado	
Pós-condição	Alterar visibilidade do tópico	
	Ator	Sistema
Fluxo Principal	1-Clicar em alterar visibilidade	
		2-Inverter visibilidade do tópico
Fluxo Alternativo(A1)	-	-

3.11.1.8 Especificação de caso de uso gostar de um tópico

O técnico sempre que encontra um tópico que identifica como útil, este poderá colocar *like* o que gera destaque.

Tabela 3.12: Tabela de especificação de caso de uso de gostar de um tópico

Caso de Uso	Gostar do tópico	
Descrição	Gostar de um tópico	
Ator	Técnico	
Pré-condição	Clicar no tópico desejado	
Pós-condição	Alterar gostos do tópico	
	Ator	Sistema
Fluxo Principal	1-Clicar em gosto	
		2-Verificar se o gosto já existe - Não
		3-Acrescentar gosto ao tópico
Fluxo Alternativo(A1)	1-Clicar em gosto	
		2-Verificar se o gosto já existe - Sim
		3-Remover gosto do tópico

3.11.1.9 Especificação de caso de uso gostar de um comentário

Sempre que um técnico identificar um comentário como útil este poderá colocar *like* o que gera destaque.

Tabela 3.13: Tabela de especificação de caso de uso de gostar de comentário

Caso de Uso	Gostar de comentário	
Descrição	Gostar de um comentário	
Ator	Técnico	
Pré-condição	Clicar no tópico desejado	
Pós-condição	-	
	Ator	Sistema
Fluxo Principal	1-Clicar em gosto	
		2-Verificar se o gosto já existe - Não
		3-Acrescentar gosto ao comentário
Fluxo Alternativo(A1)	1-Clicar em gosto	
		2-Verificar se o gosto já existe - Sim
		3-Remover gosto do tópico

3.11.1.10 Especificação de caso de uso de comentar tópico

Sempre que um técnico encontra um tópico sobre uma questão que poderá ajudar, este consegue responder através de um comentário, onde este também poderá adicionar imagens.

Tabela 3.14: Tabela de especificação de caso de uso de comentar um tópico

Caso de Uso	Comentar o tópico	
Descrição	Comentar um tópico	
Ator	Técnico	
Pré-condição	Clicar no tópico desejado	
Pós-condição	Inserir a resposta no tópico	
	Ator	Sistema
Fluxo Principal	1-Indicar a descrição da resposta	
	2-Anexar Imagem	
	3-Confirmar a resposta	
		4-Inserir novo comentário no tópico
Fluxo Alternativo(A1)	1-Indicar a descrição da resposta	
	2-Confirmar a resposta	
		3-Inserir novo comentário no tópico
Fluxo Alternativo(A2)	1-Cancelar a criação do tópico	

3.11.1.11 Especificação de caso de uso ativar conta

O técnico para ativar a sua conta deverá introduzir o código de ativação correto, caso contrário não será possível ativar.

Tabela 3.15: Tabela de especificação de caso de uso ativação de conta

Caso de Uso	Ativar conta	
Descrição	Ativar conta de aplicação	
Ator	Técnico	
Pré-condição	-	
Pós-condição	-	
	Ator	Sistema
Fluxo Principal	1-Inserir código de validação	
	2-Validar conta	
		3-Verificar se o código está correto-Sim
		4-Validar conta
Fluxo Alternativo(A1)	1-Cancelar Ativação de conta	
Fluxo Alternativo(A2)	1-Inserir código de validação	
	2-Validar conta	
		3-Verificar se o código está correto-Não
		4-Código de validação incorreto

3.11.1.12 Especificação de caso de uso configurar notificações

As notificações da aplicação poderão ser personalizadas, o que possibilita escolher entre *email*, *push* e ambos, para além destas configurações, é também possível personalizar o tipo de notificação para cada método, seja relatório diário de todas as notificações ou notificações em tempo real.

Tabela 3.16: Tabela de especificação de caso de uso de configuração de notificações

Caso de Uso	Configuração de notificações	
Descrição	Configuração de notificações do técnico	
Ator	Técnico	
Pré-condição	-	
Pós-condição	-	
	Ator	Sistema
Fluxo Principal	1-Indicar preferência de receção de notificações	
	2-Indicar tipo de receção de notificações	
Fluxo Alternativo(A1)	1-Ver notificações	

3.11.1.13 Especificação de caso de uso registar técnico

Sempre que uma empresa deseja realizar o registo de técnicos em seu nome, esta poderá indicar o nº contribuinte e *email*, com isto, este receberá um *email* para confirmar o registo de conta.

Tabela 3.17: Tabela de especificação de caso de uso de registar técnico

Caso de Uso	Registar técnico	
Descrição	Registar conta de técnico da empresa	
Ator	Empresa	
Pré-condição	-	
Pós-condição	-	
	Ator	Sistema
Fluxo Principal	1-Indicar o nºcontribuinte	
	2-Indicar <i>email</i>	
		3-Registar técnico
Fluxo Alternativo(A1)	-	-

3.11.2 Diagramas de casos de uso

Para ser possível visualizar graficamente todas as ações que os atores conseguem realizar e para melhorar a comunicação com as partes interessadas do projeto, foram desenvolvidos diagramas de casos de uso.

3.11.2.1 Casos de uso Fórum

Na Figura 3.11, é possível visualizar o diagrama de casos de uso para o fórum. Neste, o técnico poderá ver as listagens de tópicos em destaque e tópicos mais recentes. Caso seja um técnico oficial terá acesso aos tópicos privados e os seus tópicos. O técnico poderá também pesquisar por tópicos, dos quais ele terá a possibilidade de selecionar um para visualizar. O técnico conseguirá, além disso, criar um novo tópico, onde se dirigirá para a criação de tópicos. Aqui, o técnico será obrigado a inserir um título, descrição e tipo do tópico para o criar, mas poderá inserir imagens e indicar produto referente. Para finalizar a criação o técnico conseguirá confirmar ou cancelar o processo.

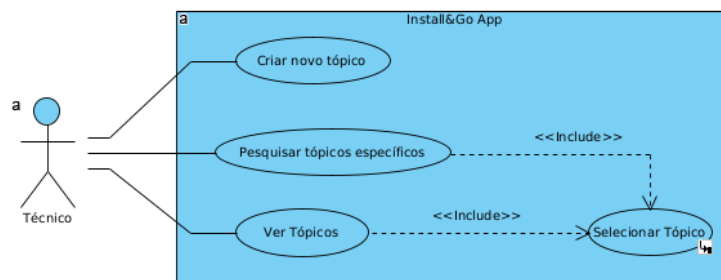


Figura 3.11: Diagrama de casos de uso de fórum

3.11.2.2 Casos de uso de pesquisar tópicos

O técnico poderá realizar a pesquisa por tópicos específicos, esta será ser realizada por escrito onde indica o assunto a pesquisar e poderá ser filtrada.

O técnico terá também a possibilidade de pesquisar por código QR de produto, uma vez que, o servidor da Motorline esteja desenvolvido para tal.

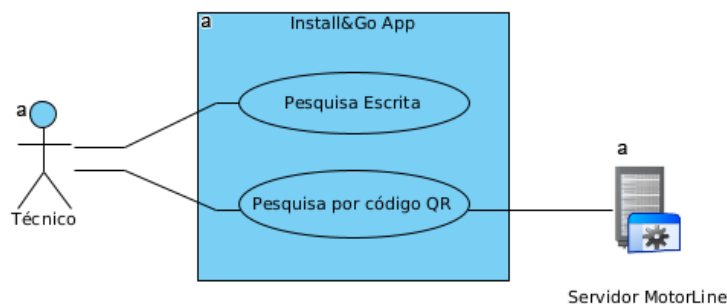


Figura 3.12: Diagrama de casos de uso de pesquisa de tópicos

3.11.2.3 Casos de uso ver detalhes de tópico

Assim que um técnico seleciona um tópico, é movido para os detalhes, onde consegue visualizar os detalhes, responder e, caso seja o seu tópico, consegue finalizar, selecionar a melhor resposta, remover a melhor resposta, eliminar e alterar a visibilidade do tópico.

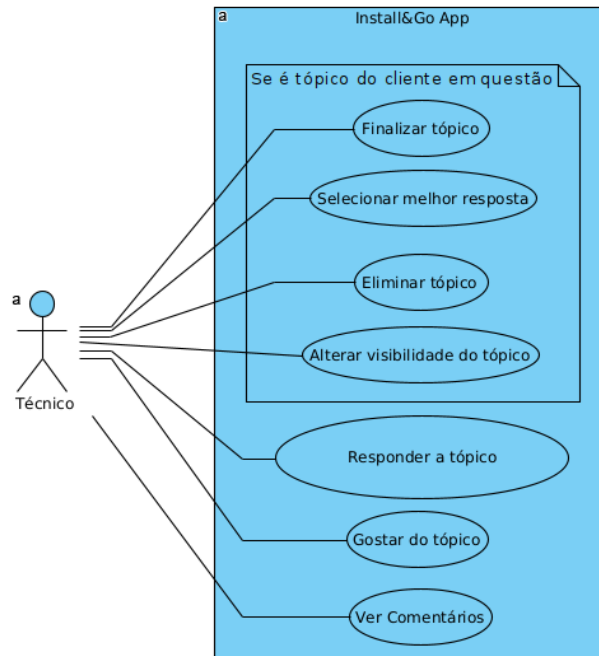


Figura 3.13: Diagrama de casos de uso de detalhes de tópico

3.11.2.4 Casos de uso ver comentários

O técnico quando decide visualizar os comentários consegue responder e gostar de uma resposta ou comentário, caso este seja seu ainda o consegue apagar.

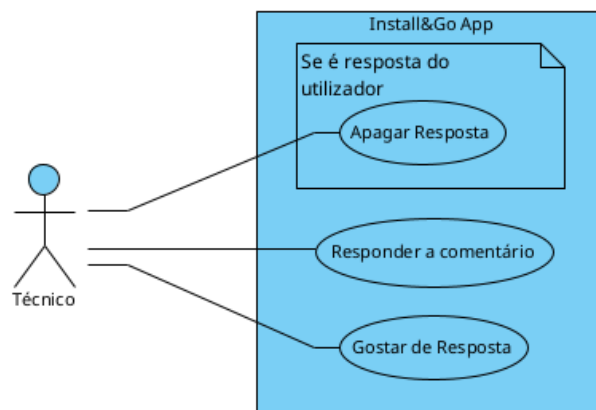


Figura 3.14: Diagrama de casos de uso de ver comentários

3.11.2.5 Casos de uso ativação de conta

Assim que uma conta é confirmada, um *email* de ativação é enviado para técnico e esta deverá ser ativada. Para isto, o código deverá ser indicado pelo técnico para se proceder à ativação da conta. Este, poderá em caso de necessidade, pedir o reenvio do código de ativação, o qual será gerado novamente e reenviado.

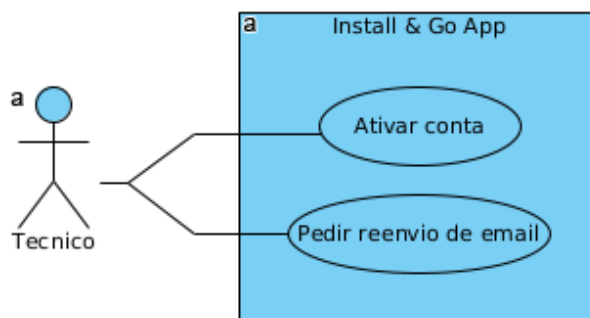


Figura 3.15: Diagrama de casos de uso de ativação de conta

3.11.2.6 Casos de uso perfil

Sempre que o técnico desejar alterar alguma informação, este poderá alterar o seu nome, *email* e imagem de perfil.

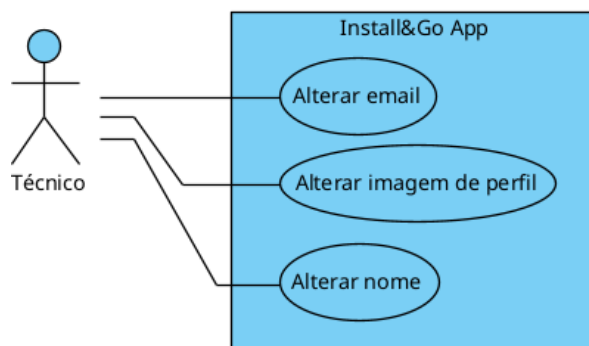


Figura 3.16: Diagrama de casos de uso de perfil

3.11.2.7 Casos de uso notificações

Sempre que o técnico desejar ver as suas notificações, poderá seleciona-las, também dispõe da possibilidade de alterar a configuração das notificações, para apenas as receber por *email* ou push, ou então, ambas. Terá também a possibilidade de personalizar cada método, para receber um relatório diário de notificações ou então, notificações em tempo real.

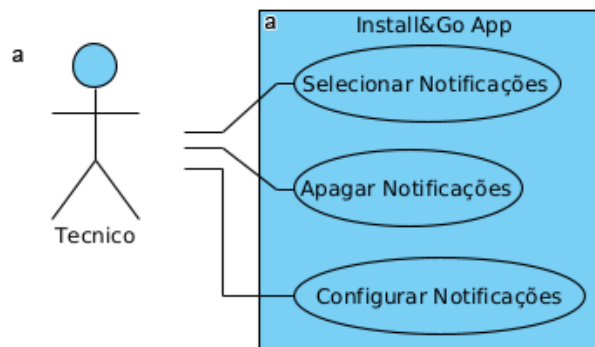


Figura 3.17: Diagrama de casos de notificações

3.11.2.8 Casos de uso gestão de recursos humanos

Uma empresa poderá registrar contas para os seus técnicos no seu nome, com a indicação do *email* e nº contribuinte. Esta poderá também impedir acesso a estas contas ou remover completamente a conta da aplicação.

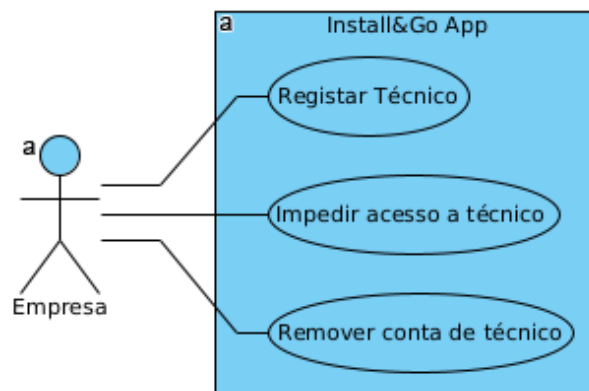


Figura 3.18: Diagrama de casos de uso de recursos humanos

3.12 Diagrama Entidade Relação

O software Install&Go é suportado por uma base de dados relacional esquematizada de acordo com as necessidades do projeto.

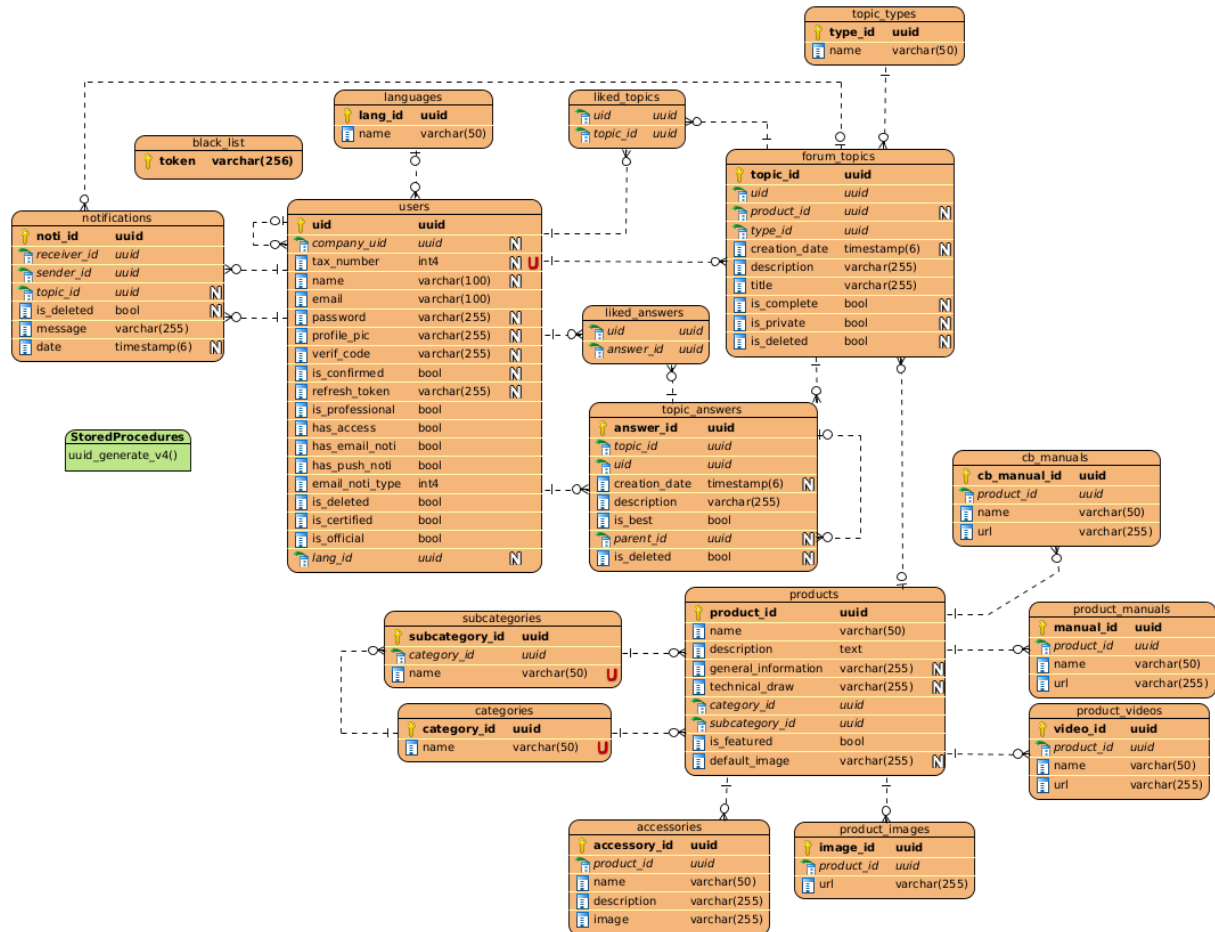


Figura 3.19: Diagrama Entidade Relação base de dados Install&Go

3.12.1 Escolha de diagrama de entidade relação

Durante o desenvolvimento do diagrama de entidade relação, surgiu a opção de separar as empresas dos seus técnicos em duas tabelas como exemplificado na Figura 3.20. Neste sempre que se deseja, por exemplo, obter o utilizador que criou um tópico é necessário verificar se o *uid* contido é de uma empresa ou de um técnico e apenas de seguida se obter o utilizador que criou o tópico, sendo este um exemplo entre os demais do mesmo tipo. Tendo em conta este problema foi decidido optar pelo diagrama da Figura 3.19.

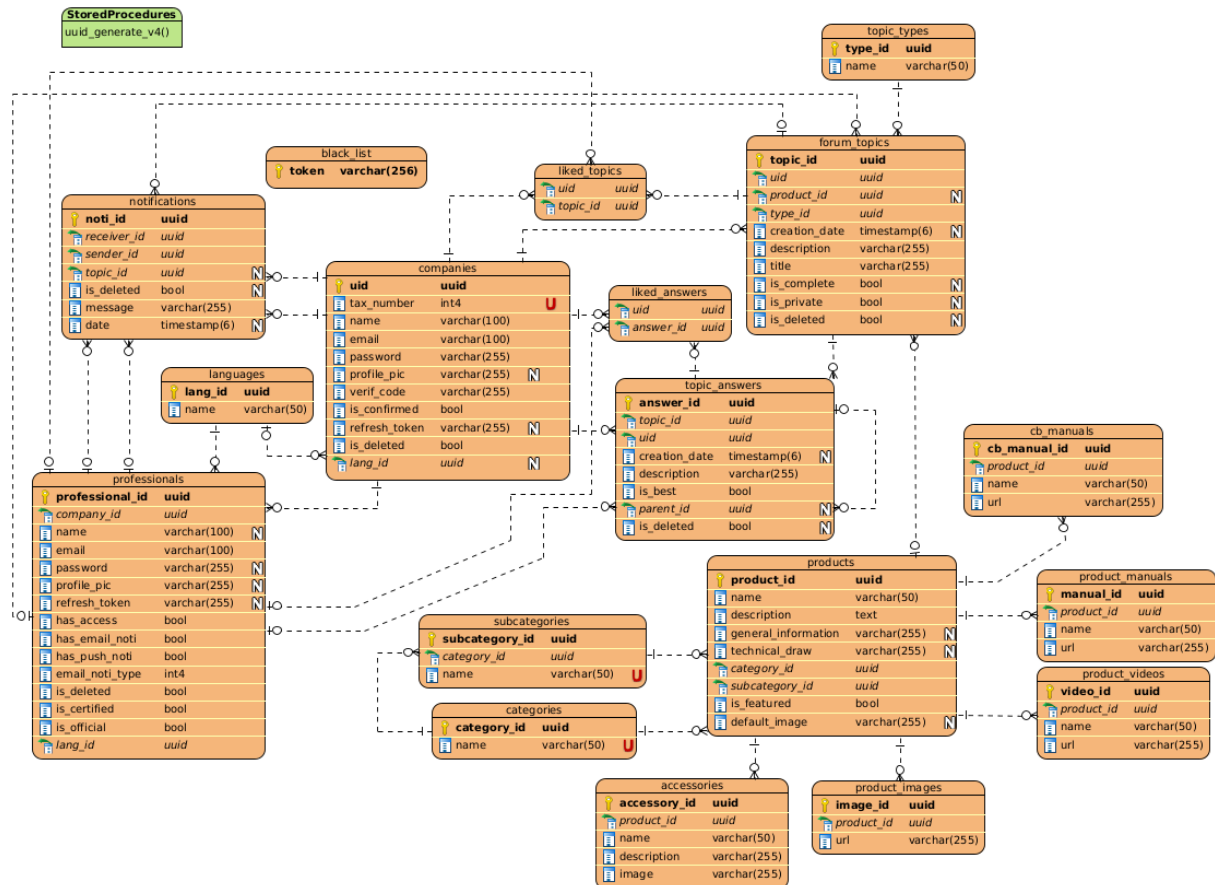


Figura 3.20: Diagrama Entidade Relação alternativo

3.12.2 Dicionário de termos

Com intuito de alcançar o propósito de cada tabela e atributo foi criado um dicionário de termos para a base de dados(Tabela 3.18).

Tabela 3.18: Dicionário de termos da base de dados

Tabela	Descrição	Atributos	Descrição
users	Tabela encarregue de guardar todos os dados referentes aos utilizadores da aplicação	uid	<i>id</i> do utilizador
		company_id	<i>id</i> da empresa referente ao técnico
		tax_number	Número de contribuinte do utilizador
		name	Nome do utilizador
		<i>email</i>	Email do utilizador
		<i>password</i>	Password do utilizador
		profile_pic	Imagem de perfil do utilizador
		verif_code	Código de verificação do utilizador
		is_confirmed	Verificação de se o código está confirmado
		refresh_token	Token de refresh
		is_professional	Verificação se é profissional
		has_access	Verificação se tem acesso à conta
		has_email_noti	Verificação se ativou notificações de <i>email</i>
		has_push_noti	Verificação se ativou notificações push
		<i>email_noti_type</i>	Tipo de notificação de <i>email</i>
		push_noti_type	Tipo de notificação push
		is_deleted	Verificação se a conta se encontra apagada
		is_certified	Verificação se é um técnico certificado
		is_official	Verificação se é um técnico oficial

Continued on next page

Tabela 3.18: Dicionário de termos da base de dados (Continued)

		lang_id	<i>id</i> da linguagem preferencial do utilizador
black_list	Tabela que guarda os tokens a bloquear	token	Token a bloquear
liked_topics	Tabela encarregue de guardar todos os tópicos que foram gostados pelo utilizador	uid	<i>id</i> do utilizador
		topic_id	<i>id</i> do tópico
liked_answers	Tabela encarregue de guardar todas as respostas que receberam gosto do utilizador	uid	<i>id</i> do utilizador
		answer_id	<i>id</i> da resposta
topic_types	Tabela encarregue de guardar os tipos de tópico existentes	type_id	<i>id</i> do tipo de tópico
		name	nome do tipo de tópico
notifications	Tabela encarregue de guardar todas as notificações do técnico	noti_id	<i>id</i> da notificação
		receiver_id	Recetor da notificação
		sender_id	Emissor da notificação
		topic_id	<i>id</i> do tópico em caso de estar referente a um tópico
		is_deleted	Verificação se a notificação está apagada
		message	Mensagem da notificação
		date	Data de emissão da notificação

Continued on next page

Tabela 3.18: Dicionário de termos da base de dados (Continued)

forum_topics	Tabela encarregue de guardar todos os tópicos existentes na aplicação	topic_id	<i>id</i> do tópico
		uid	<i>id</i> do dono do tópico
		product_id	Produto referente ao tópico
		tiype_id	<i>id</i> do tipo referente ao tópico
		creation_date	Data de criação do tópico
		description	Descrição do tópico
		title	Título do tópico
		is_complete	Verificação se o tópico está finalizado
		is_private	Verificação se o tópico é privado
		is_deleted	Verificação se o tópico está apagado
topic_answers	Tabela encarregue de guardar todas as respostas a um tópico	answer_id	<i>id</i> da resposta
		topic_id	<i>id</i> do tópico
		uid	<i>id</i> do dono da resposta
		creation_date	Data de criação da resposta
		description	Descrição da resposta
		is_best	Verificação se é a melhor resposta
		parent_id	<i>id</i> da resposta pai
		is_deleted	Verificação se o topico se encontra apagado
categories	Tabela encarregue de guardar todas as categorias de produtos existentes	category_id	<i>id</i> da categoria
		name	Nome da categoria
subcategories	Tabela encarregue de guardar as subcategorias de produtos existentes	subcategory_id	<i>id</i> da subcategoria
		category_id	<i>id</i> da categoria
		name	Nome da subcategoria

Continued on next page

Tabela 3.18: Dicionário de termos da base de dados (Continued)

cb_manuals	Tabela encarregue de guardar os manuais de utilização das placas de controlo	cb_manual_id	<i>id</i> do manual
		product_id	<i>id</i> do produto
		name	Nome da placa de controlo
		url	Url do manual
products	Tabela encarregue de guardar as informações dos produtos do catálogo da empresa	product_id	<i>id</i> do produto
		name	Nome do produto
		description	Descrição do produto
		general_information	Url da informação geral do produto
		technical_draw	Url do desenho técnico do produto
		category_id	<i>id</i> da categoria de produto
		subcategory_id	<i>id</i> da subcategoria de produto
		is_featured	Verificação se o produto é um destaque
product_manuals	Tabela encarregue de guardar os manuais de utilização dos produtos	cb_manual_id	<i>id</i> do manual
		product_id	<i>id</i> do produto
		name	Nome do manual
		url	Url do manual
product_videos	Tabela encarregue de guardar os videos de cada produto	cb_manual_id	<i>id</i> do video
		product_id	<i>id</i> do produto
		name	Nome do video
		url	Url do video
languages	Tabela encarregue de guardar as linguagens suportadas pela aplicação	lang_id	<i>id</i> da linguagem
		name	Nome da linguagem

3.13 Diagrama de Classes

A fim de prever e organizar o *software* foi desenvolvido um diagrama de classes(Figura 3.21) que permite visualizar cada classe que se espera conter no *software*, assim como também os seus atributos e métodos.

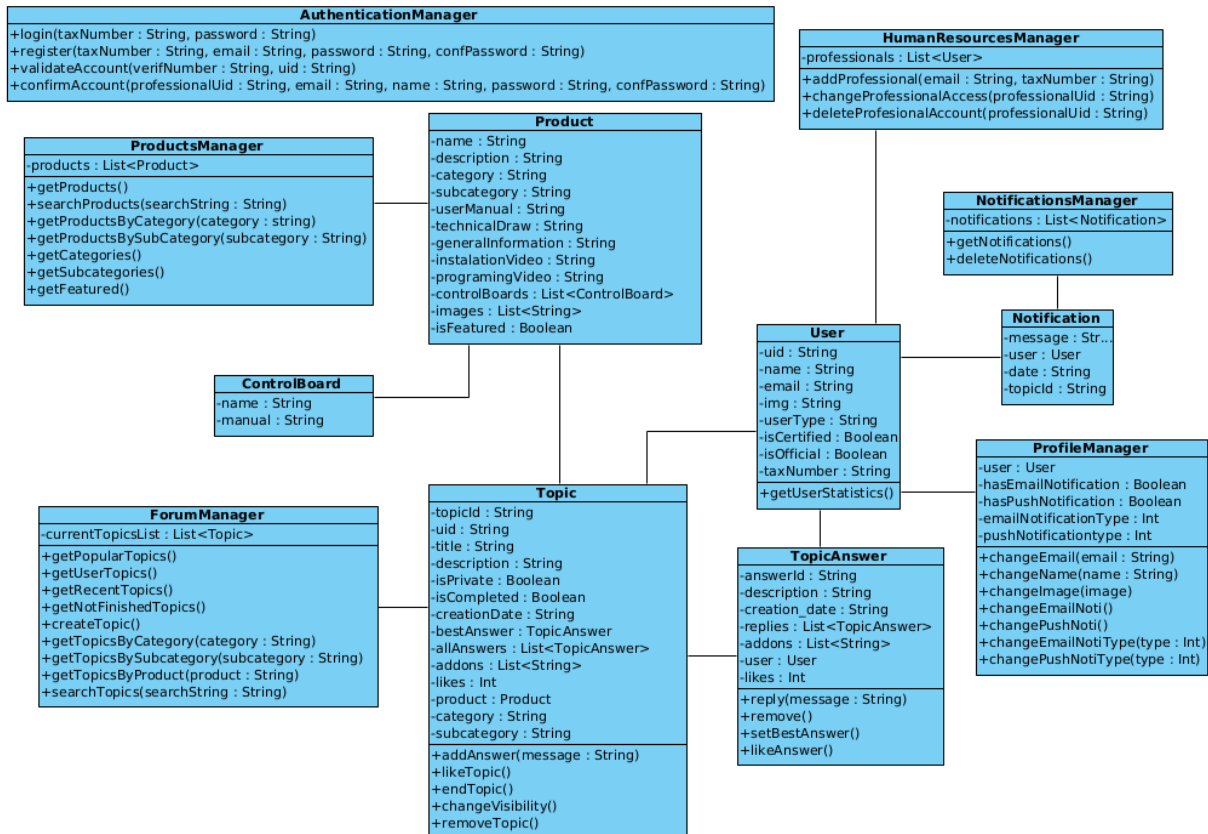


Figura 3.21: Diagrama de classes Install&Go

3.14 Mockups

Com o propósito de criar um *design* para seguir e apresentar às partes interessadas antes de iniciar a fase de desenvolvimento, então foram realizadas *mockups* do *design* da aplicação. Este *design* foi iterativamente revisto pelo cliente e ajustado até alcançar o estado final.

3.14.1 Página Inicial

A página inicial da aplicação, dá ao utilizador a possibilidade de navegar pelos produtos do catálogo, filtrar por categorias e subcategorias, assim como realizar uma pesquisa rápida e por fim navegar para o fórum. Caso um técnico esteja com sessão iniciada este poderá visualizar o *icon* de notificações e a sua imagem de perfil.

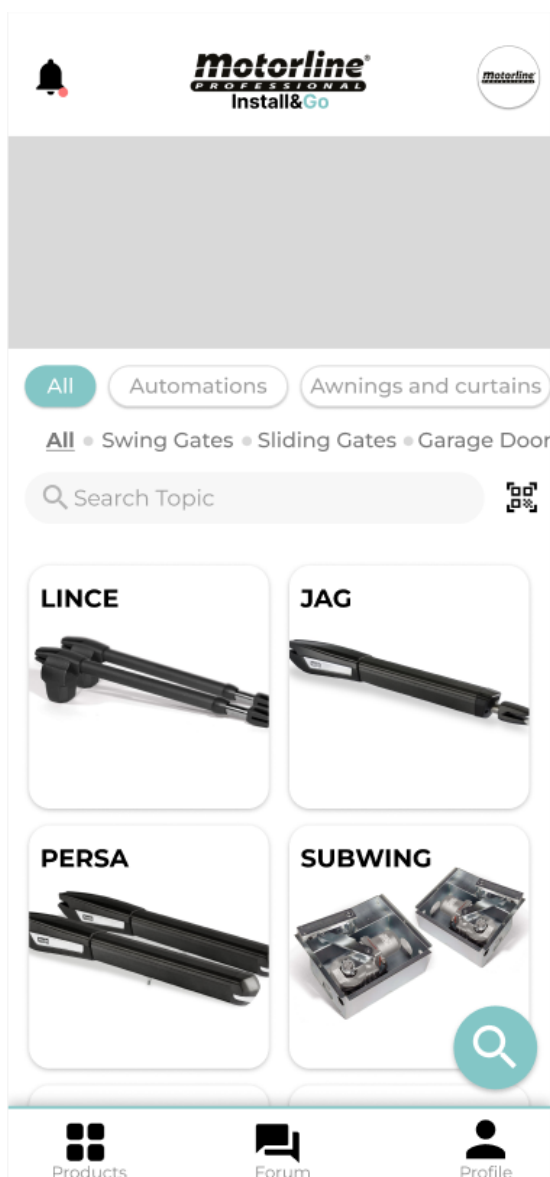


Figura 3.22: Página inicial do fórum

3.14.2 Autenticação - Login e Registo

Na autenticação, é possível iniciar sessão e registo, mas a empresa é a única entidade poderá realizar o registo no *software*.

The figure displays two side-by-side mockups of a web application's authentication interface. Both mockups feature a light gray background and a white central card.

(a) Página de login: The card has the title "Install & Go" at the top. Below it are two input fields: "Tax number" (with placeholder "Tax Number") and "Password" (with placeholder "Password"). A teal link "Forgot Password?" is positioned to the right of the password field. Below the inputs are two buttons: a teal "Login" button and a white "Continue without session" button with a gray border. At the bottom, the text "Don't have an account yet?" is followed by a teal "Sign Up" link.

(b) Página de registo: The card has the title "Install&Go" at the top. Below it are four input fields: "Tax number" (placeholder "Tax Number"), "Company Name" (placeholder "Company name"), "Email" (placeholder "Email"), and "Password" (placeholder "Password"). Below the password field is a "Password Confirmation" field (placeholder "Password Confirmation"). A teal "Register" button is located below the confirmation field. At the bottom, the text "Already have an account?" is followed by a teal "Login" link.

(a) Página de login

(b) Página de registo

Figura 3.23: Autenticação - Login e Registo

3.14.3 Autenticação - Ativação e Confirmação de conta

Na autenticação também existe a página de confirmação de conta. Um técnico que tem a conta recentemente adicionada poderá confirmar o registo, indicar as informações finais e por fim será direcionado para a página de ativação onde terá de colocar o código de ativação enviado para o *email*, esta página também será aberta caso um técnico realize o *login* com uma conta que não foi ativada ou sempre que um registo é finalizado.

The figure consists of two side-by-side screenshots of a web application interface for 'Install&Go'.

Screen (a) is the 'Registered Details' page. It features the 'Install&Go' logo at the top. Below it, the text 'Registered Details:' is followed by 'Email: joapereira@motorline.com' and 'Tax Number: 999999999'. There are four input fields: 'Name', 'Email' (pre-filled with 'joapereira@motorline.com'), 'Password', and 'Password Confirmation'. A teal 'Register' button is at the bottom. A link 'Already have an account? Login' is at the very bottom.

Screen (b) is the 'Account Activation' page. It features the 'Install&Go' logo at the top. Below it, the text 'Account Activation' is followed by an 'Activation code' input field. Below the field is a link 'Didn't receive code? Resend code'. At the bottom are two buttons: 'Cancel' and 'Verify'.

(a) Página de confirmação de conta

(b) Página de ativação de conta

Figura 3.24: Autenticação - Ativação e Confirmação de conta

3.14.4 Página inicial fórum

O técnico assim que se dirige ao fórum entrará na página inicial. Esta página permite navegar entre as diferentes listagens de tópicos acessíveis ao técnico, pesquisar, filtrar por tipo e criar um novo tópico.

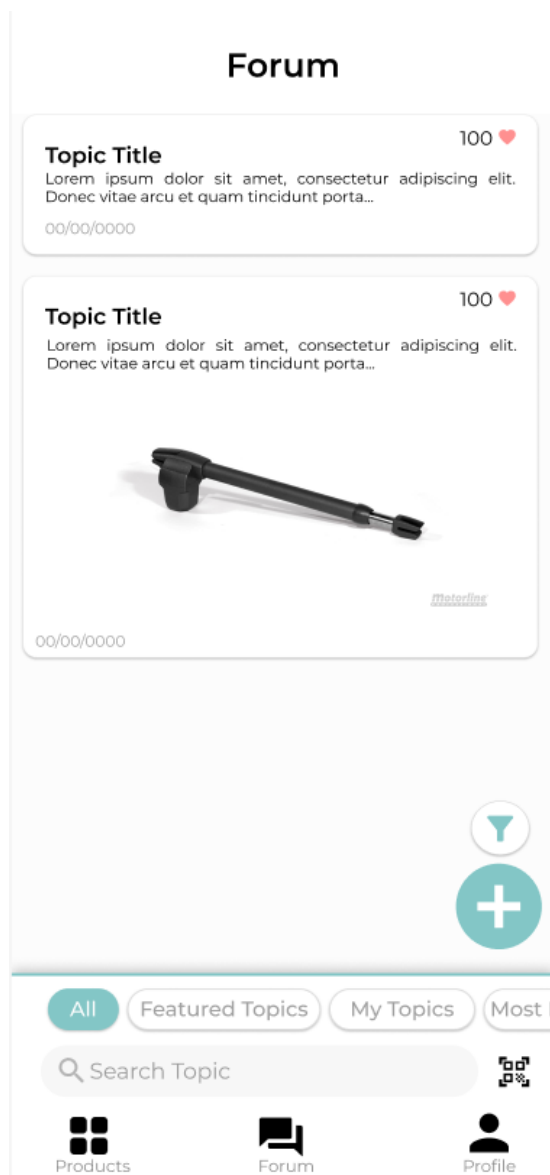
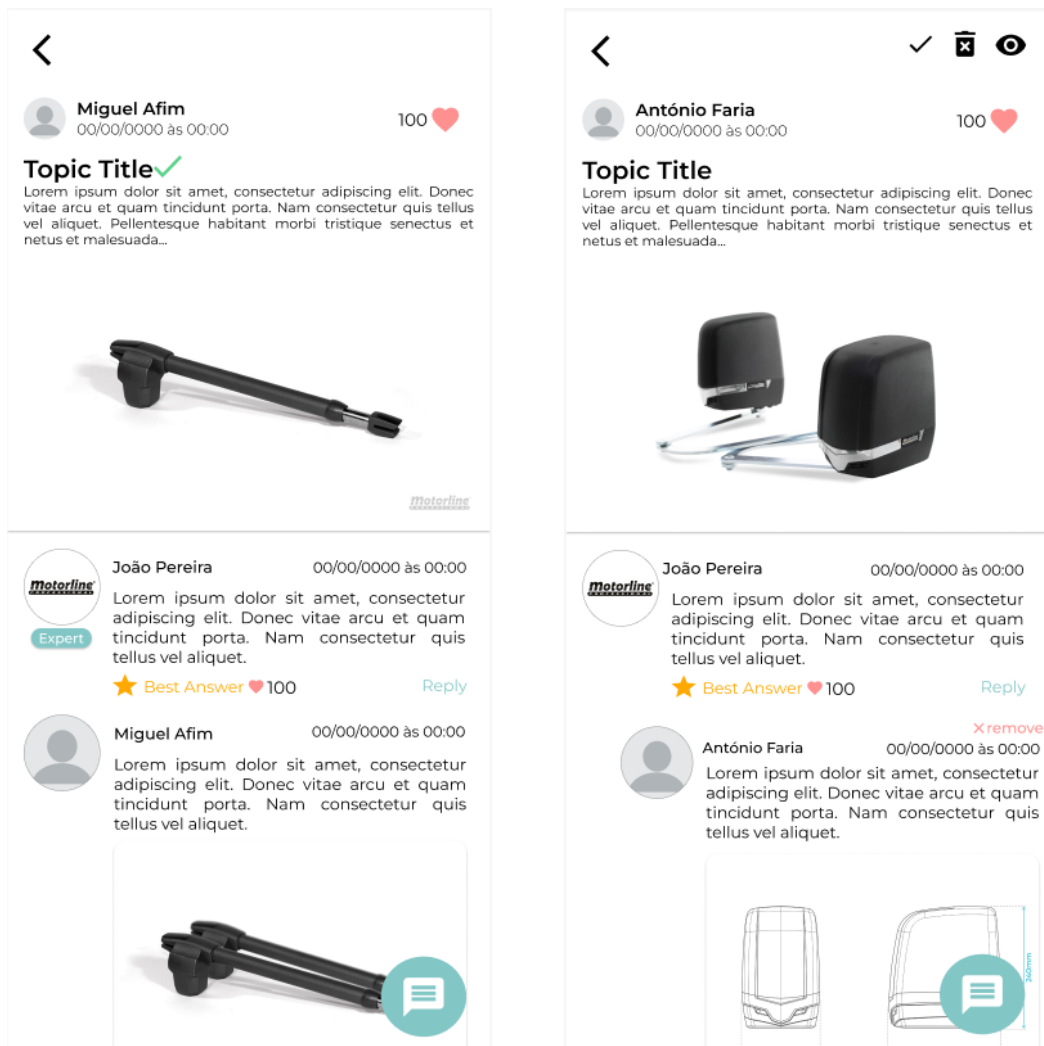


Figura 3.25: Página inicial do fórum

3.14.5 Página de detalhes de um tópico

Assim que o técnico seleciona um tópico, será encaminhado para a página de detalhes, onde é indicado o nome do proprietário, a imagem de perfil, a hora de criação, a quantidade de gostos, o título, a descrição, as imagens e os comentários. É possível gostar do tópico, gostar de comentários, comentar o tópico e outros comentários.

Se o tópico for do técnico que está a visualizar, poderá também concluir, eliminar e alterar a sua visibilidade.



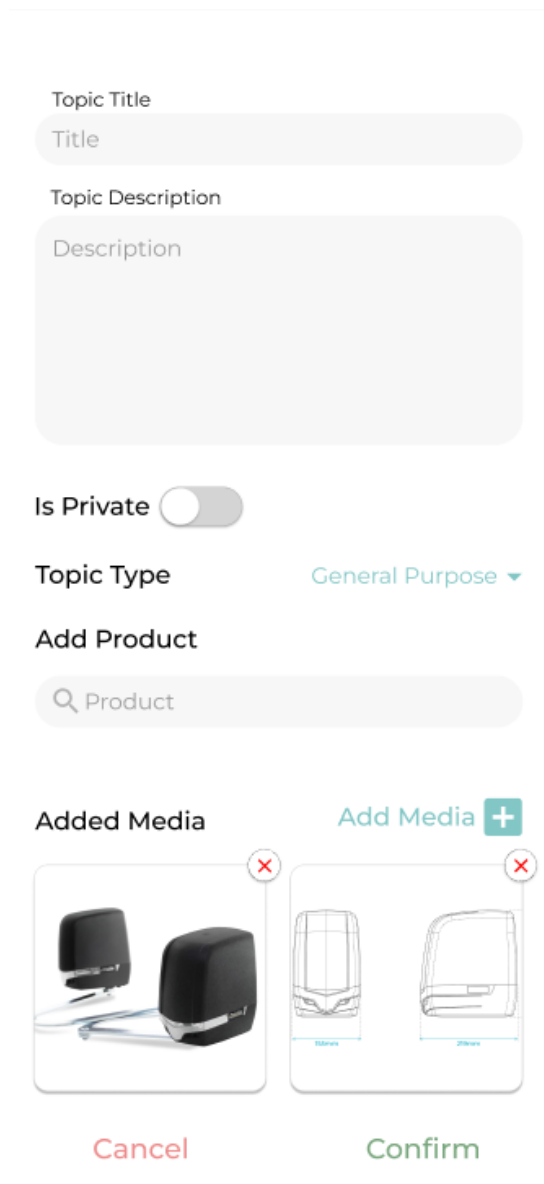
(a) Página de detalhes de um tópico

(b) Página de detalhes de um tópico do técnico

Figura 3.26: Página de detalhes de tópico do software

3.14.6 Página de criação de um tópico

Quando um técnico inicia a criação de um tópico, é obrigado a inserir o título e a descrição. Já a indicação da visibilidade, do tipo de tópico, do produto referente e de imagens é opcional. A qualquer momento poderá cancelar ou confirmar a ação.



The mockup shows a form for creating a topic. It includes a 'Topic Title' field with the placeholder 'Title', a 'Topic Description' field with the placeholder 'Description', an 'Is Private' toggle switch, a 'Topic Type' dropdown menu set to 'General Purpose', an 'Add Product' section with a search bar containing 'Product', and an 'Added Media' section. The 'Added Media' section contains two image thumbnails: one of a car stereo and one of a car's front view. Each thumbnail has a red 'X' icon in the top right corner. Below the thumbnails are two buttons: 'Cancel' in red and 'Confirm' in green. An 'Add Media +' button is located to the right of the 'Added Media' section.

Figura 3.27: Página de criação de tópico

3.14.7 Página de notificações

Um técnico sempre que desejar tem como opção visualizar as suas notificações. Neste ecrã, é possível ver todas com a identificação de quem enviou, qual a descrição e a data de receção. O técnico, também tem como escolha apagar se assim desejar.

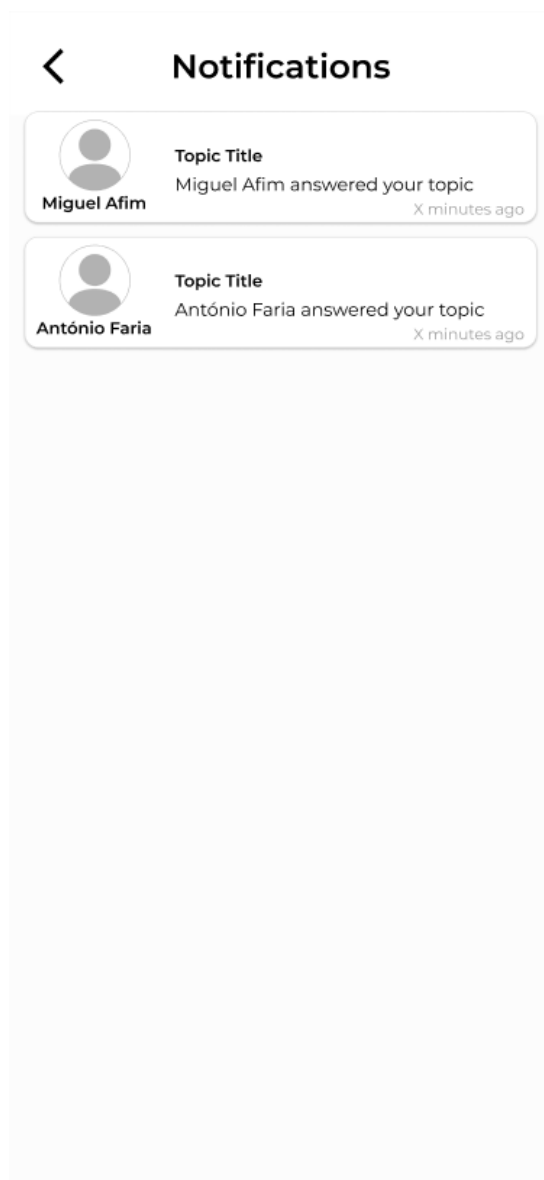


Figura 3.28: Página de notificações

3.14.8 Página de perfil de utilizador

O técnico, sempre que desejar alterar as suas informações, tem a possibilidade de modificar o *email*, a imagem de perfil e a configuração das notificações com indicação dos métodos e tipo a receber.

Caso uma empresa entre no perfil, esta visualizará um botão para aceder à gestão de recursos humanos.

The mockup displays a user profile for João Pereira, identified as an 'Official Expert' for Motorline. The profile includes an email field set to 'joapereira@motorline.com'. Under the 'Notifications' section, both 'Push Notifications' and 'Email Notifications' are checked. For 'Push Notification Type', 'All notifications' is selected over 'Daily report'. For 'Email Notification Type', 'Daily report' is selected over 'All notifications'. A 'Resources Management' button is located below the notification settings. At the bottom of the form are 'Cancel' and 'Confirm' buttons. The bottom navigation bar contains three icons: 'Products' (a 2x2 grid), 'Forum' (a speech bubble), and 'Profile' (a person icon).

Figura 3.29: Página de perfil de utilizador

3.14.9 Página de gestão de recursos humanos

Na página de gestão de recursos humanos, apenas acessível a empresas, é possível registrar novos técnicos, pesquisar e gerir os que já se encontram registados através dos seus perfis.

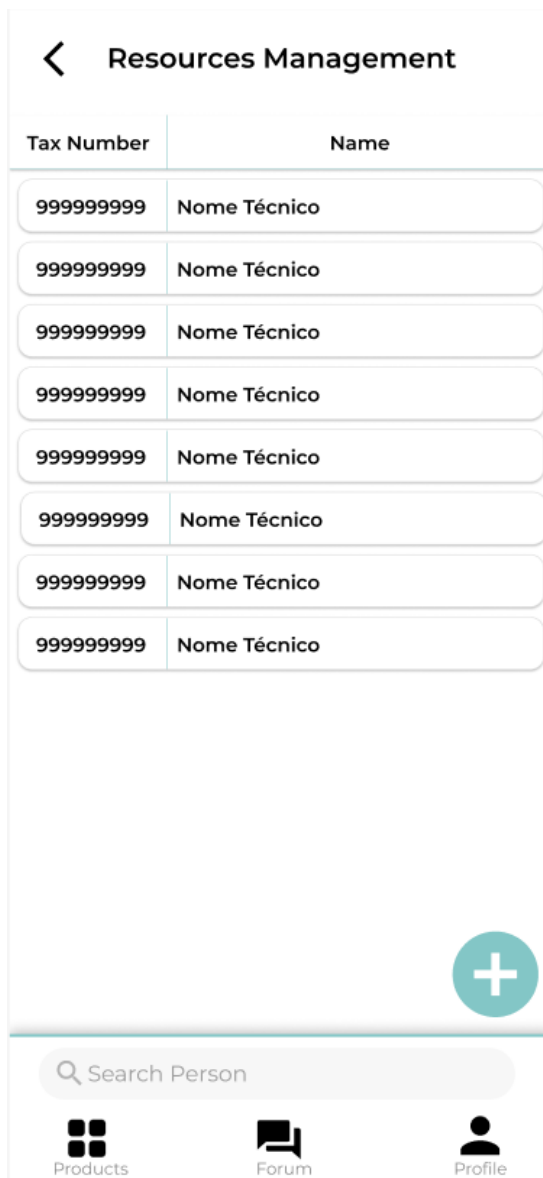


Figura 3.30: Página de gestão de recursos humanos

3.14.10 Página de perfil de técnico registrado

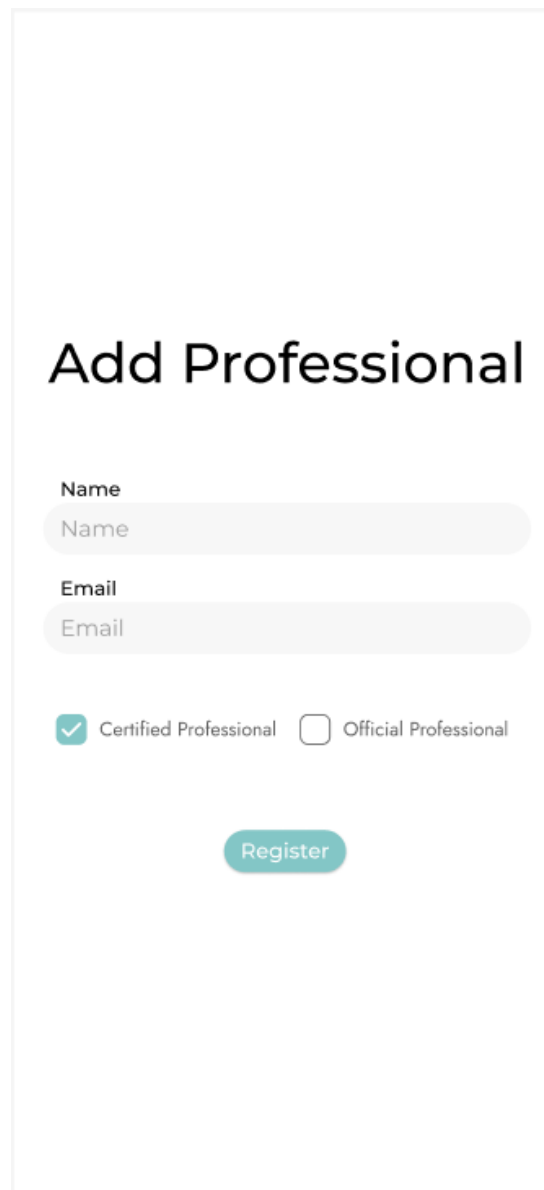
O perfil de técnico, apenas acessível para empresas, permite visualizar as estatísticas, as informações e permite impedir acesso à conta ou então remover da plataforma.



Figura 3.31: Página de perfil de técnico registrado

3.14.11 Página de registo de novo técnico

Sempre que uma empresa deseja registar um novo técnico, esta deverá indicar o nome, o *email* e tipo de técnico.



The image shows a web form titled "Add Professional". It contains two text input fields: "Name" and "Email", each with a light gray placeholder text of the same name. Below these fields are two radio button options: "Certified Professional" (which is selected, indicated by a teal checkmark) and "Official Professional" (which is unselected). At the bottom of the form is a teal "Register" button.

Figura 3.32: Página de registo de novo técnico

3.15 Diagramas de atividades

O detalhe de forma simples das ações do ator nos diferentes ecrãs foi realizado em diagramas de atividades.

3.15.1 Diagrama de atividades página inicial

Da página inicial da aplicação é possível deslocar para o fórum, para as notificações, para o perfil e realizar operações do catálogo.

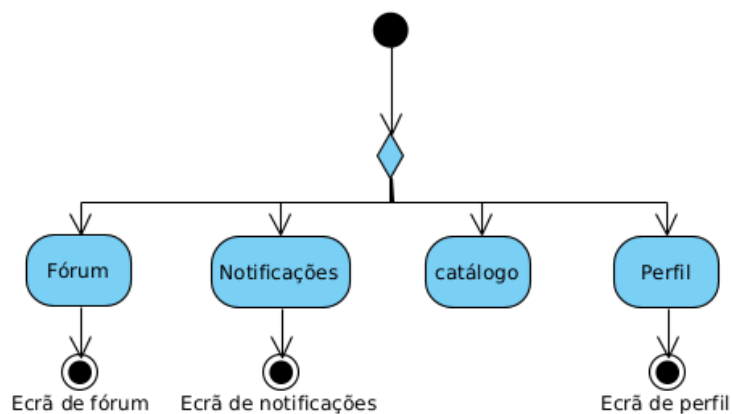


Figura 3.33: Diagrama de atividades de página inicial da aplicação

3.15.2 Diagrama de atividades página de perfil

Na página de perfil, é possível alterar a imagem, o nome, o *email*, selecionar os métodos e os tipos de notificação a receber. Caso uma empresa veja o perfil esta poderá, além das operações acima mencionadas, gerir os recursos humanos onde é encaminhada para o ecrã de gestão de recursos humanos.

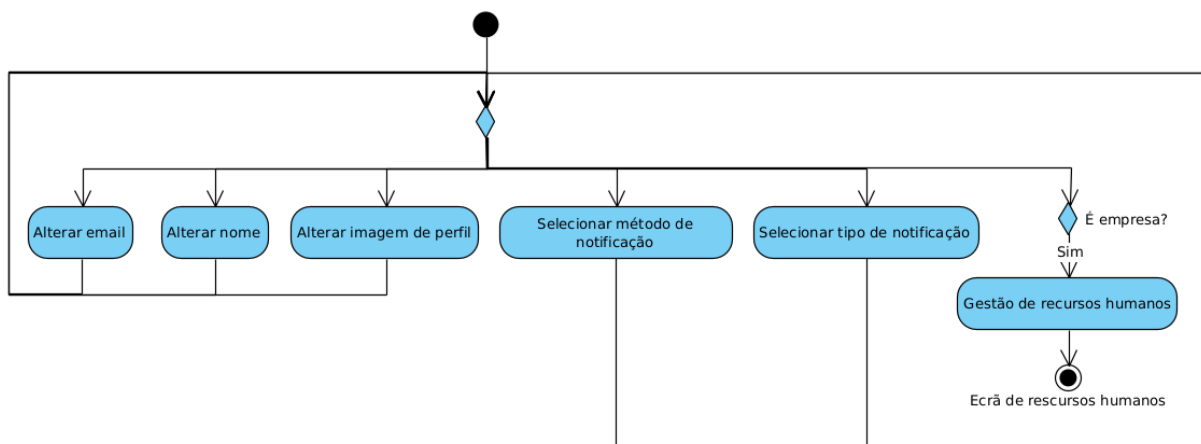


Figura 3.34: Diagrama de atividades de página de perfil

3.15.3 Diagrama de atividades página inicial do fórum

Na página inicial do fórum o técnico poderá selecionar um dos tipos de pesquisa, escrita ou código QR, filtrar por tipo, ver as listagens de tópicos em destaque, mais recentes, por responder, os seus tópicos e criar um novo tópico. Estas listagens poderão ser filtradas por tipo e sobre as mesmas tem a possibilidade de selecionar um tópico o que o redirecionará para o ecrã de detalhes.

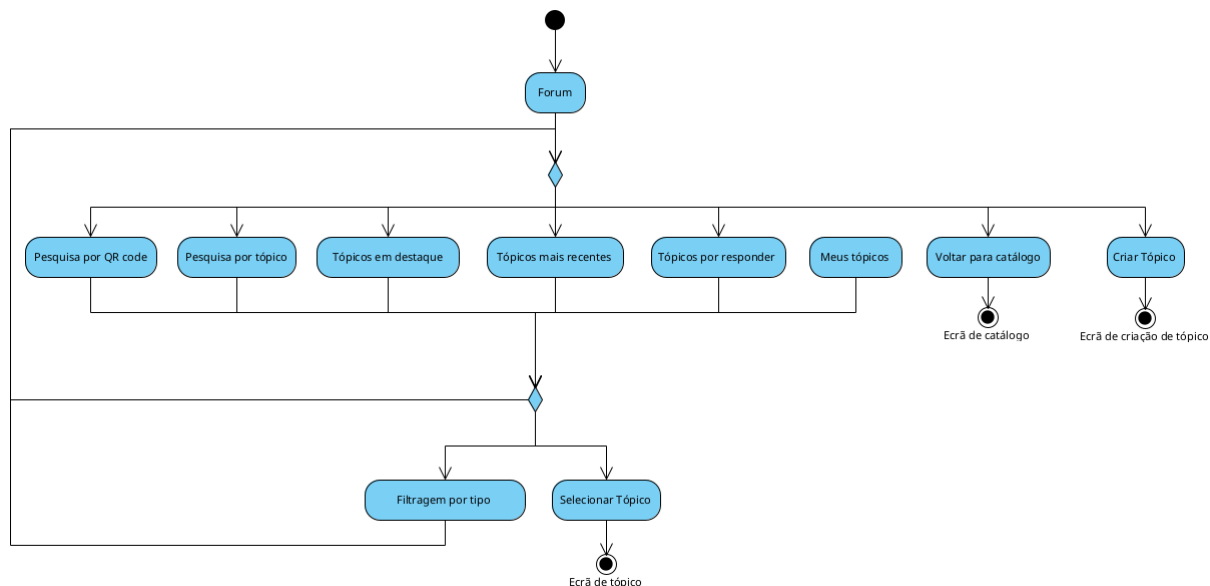


Figura 3.35: Diagrama de atividades de página inicial do fórum

3.15.4 Diagrama de atividades página de criação de tópico

Quando o técnico decide criar um tópico, obrigatoriamente tem de indicar o título, a descrição e o tipo do tópico. Por predefinição a visibilidade deste é pública, mas o técnico poderá alterar. Facultativamente o técnico poderá indicar o produto referente ao tópico, assim como anexar e remover imagens. A qualquer momento, o técnico poderá confirmar a criação do tópico, quando esta ação inicia, é realizada uma verificação do título e da descrição para concluir se estão preenchidos. Caso estes dados não estejam preenchidos é indicado ao técnico que as informações estão em falta, caso contrário este volta para o ecrã anterior.

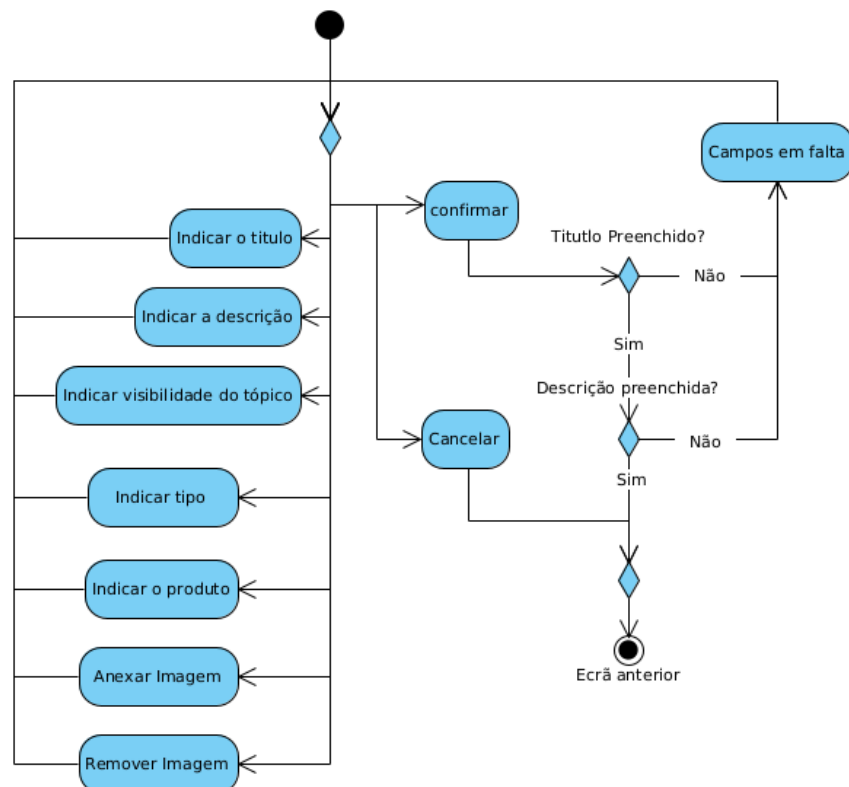


Figura 3.36: Diagrama de atividades de página de criação de tópico

3.15.6 Diagrama de atividades páginas de autenticação

Para realizar a ativação da conta do técnico, assim que este realiza o registo, confirmação da conta ou o login com uma conta que não se encontra ativa, este é encaminhado para o ecrã de ativação da conta. Neste ecrã poderá cancelar e indicar o código de ativação. Se o código estiver errado, o técnico deverá inserir-lo novamente. Por outro lado, se inserir um código correto a conta será validada e o técnico ficará autenticado. Também, em caso de necessidade, o proprietário da conta terá como opção pedir o envio de um novo código de ativação.

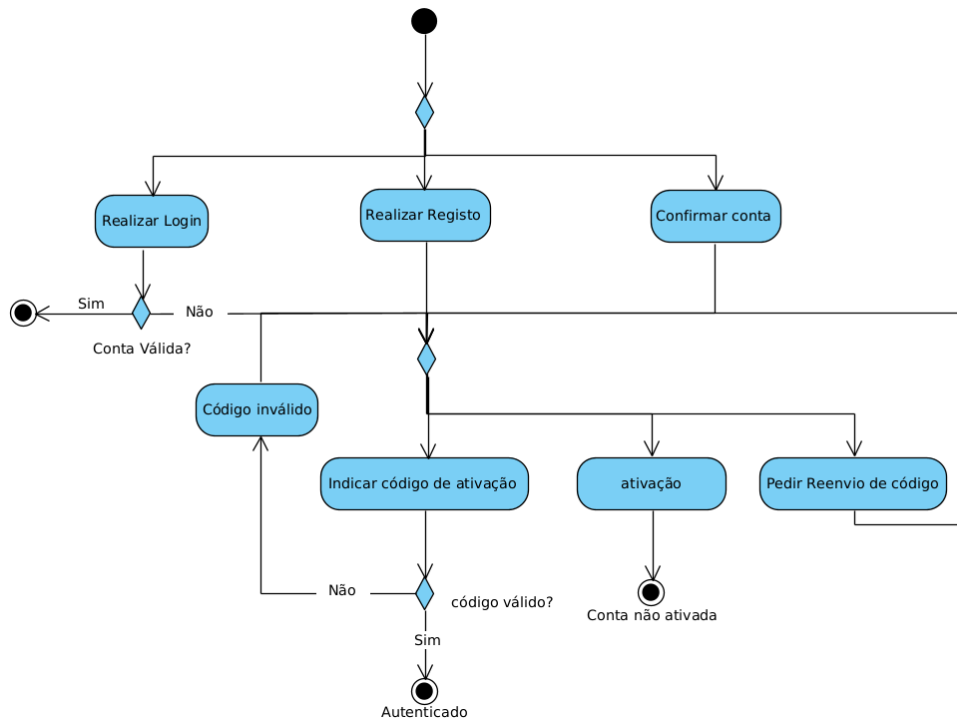


Figura 3.38: Diagrama de atividades de página de validação de conta

3.15.7 Diagrama de atividades registrar técnico

Assim que uma empresa inicia o registo de um técnico, esta é redirecionada para a página de registo do técnico. Nesta página, terá de indicar o nome, o *email* e o tipo de técnico. Por fim será capaz de confirmar o registo da conta e automaticamente a empresa é movida para a página anterior.

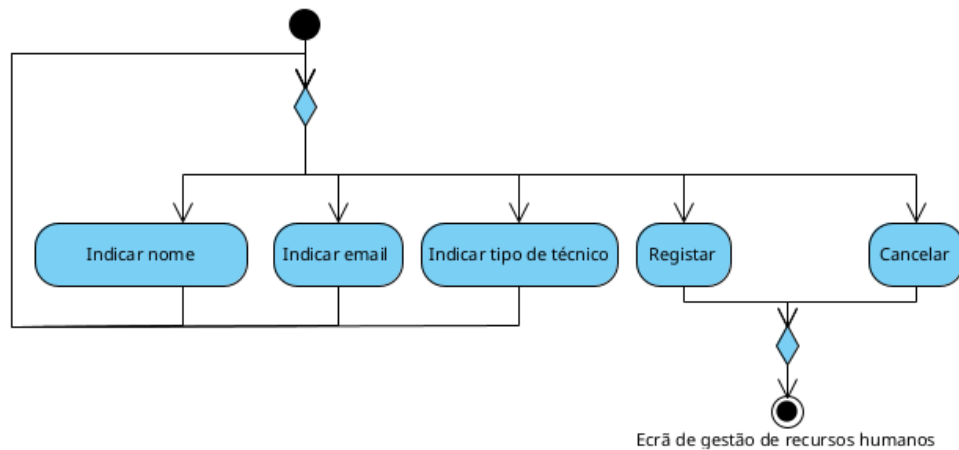


Figura 3.39: Diagrama de atividades de página de registrar técnico

3.15.8 Diagrama de atividades confirmar conta

Quando uma conta de técnico é registrada, um *email* de confirmação é enviado para o técnico. Assim que este o recebe, deverá clicar em confirmar a conta. A partir desta ação, este move-se para a página de confirmação da conta. Nesta página, o técnico poderá alterar o seu *email*, indicar o nome, a *password* e a confirmação da *password*. Todo o processo termina quando o botão de registrar for pressionado.

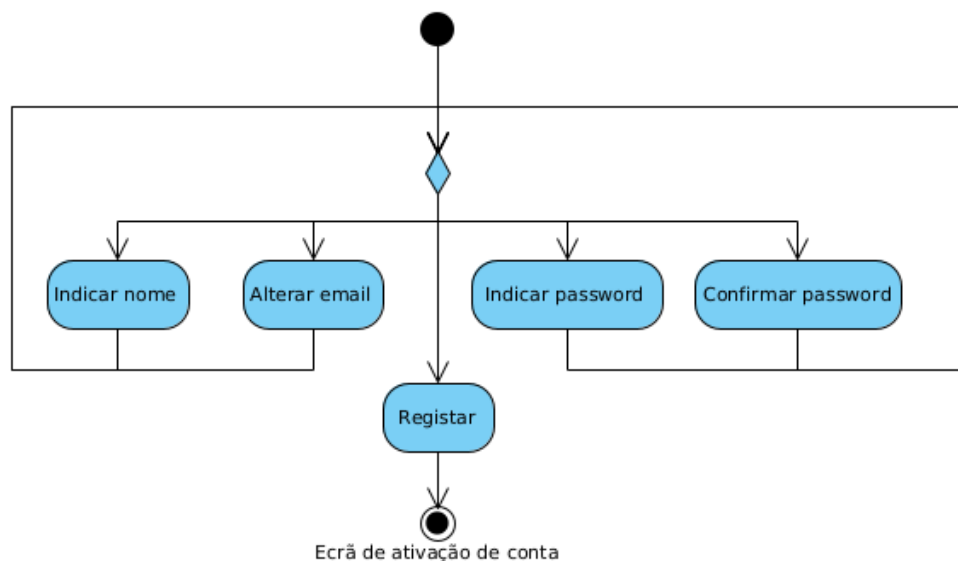


Figura 3.40: Diagrama de atividades de página de confirmar conta de técnico

3.16 Diagramas de estados

Para especificar os principais processos do projeto foram desenvolvidos diagramas de estados, com o objetivo de demonstrar o processo de criação de um tópico do fórum por parte de um técnico, o processo de aceder e responder a um tópico e o login com ativação de conta, visto que, estas interações são as de maior significância e regradas no *software*.

3.16.1 Diagrama de estados criação de tópico

Com o diagrama de estados de criação de tópico é pretendido demonstrar o processo por parte de um técnico. Assim sendo, primeiramente terá de estar autenticado, caso não esteja, será encaminhado para autenticação. De seguida criará um tópico. Após preencher os campos desejados este poderá confirmar. Caso confirme, é verificado se o tópico possui título. Caso não possua, é determinado como inválido o que levará o técnico a preencher os dados em falta. Contudo, se o título estiver preenchido, é verificado se possui descrição e tipo. Na condição de não possuir descrição ou tipo, é seguido o mesmo fluxo que o caso anterior. Caso contrário é criado um novo tópico. Por fim, na hipótese de o técnico não desejar confirmar o tópico, este poderá cancelá-lo, o que o determina como cancelado.

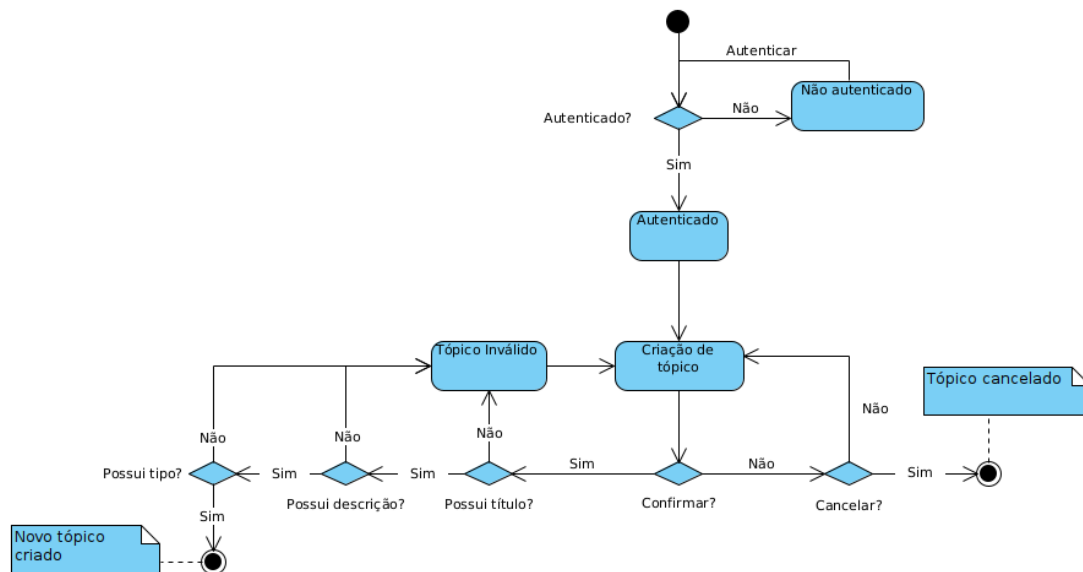


Figura 3.41: Diagrama de estados de criar tópico

3.16.2 Diagrama de estados responder a tópico

Com o diagrama de estados de responder a tópico é pretendido demonstrar o processo de seleção e de responder a um tópico por parte de um técnico. Assim sendo, primeiramente terá de estar autenticado, caso não esteja, será encaminhado para autenticação. Após a autenticação, o técnico irá por predefinição ver os tópicos em destaque. Nesta listagem selecionará um tópico o que permite responder. Depois da criação do comentário, este poderá confirmar e caso confirme ficará criado, caso contrário ficará cancelado.

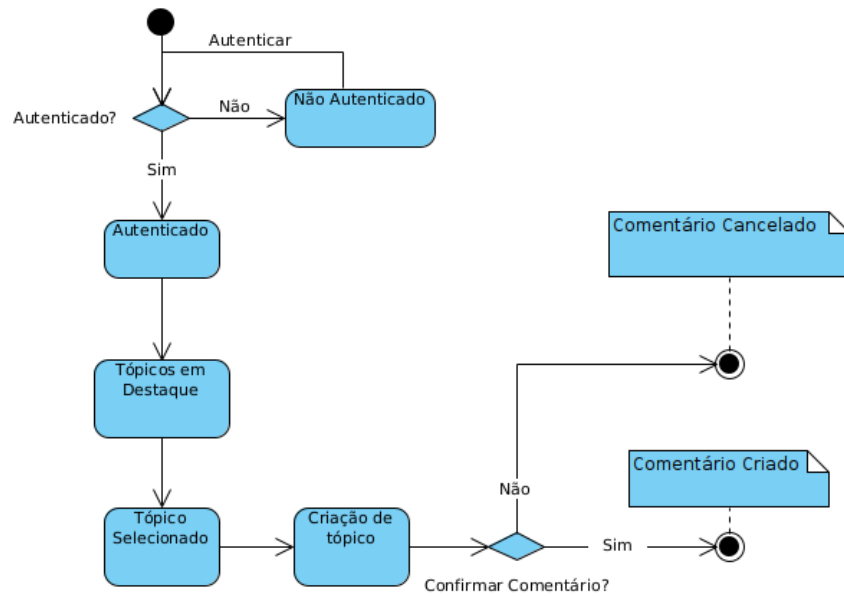


Figura 3.42: Diagrama de estados de criar tópico

3.16.3 Diagrama de estados autenticação e validação de conta

Aquando a realização do login, o técnico indicará as suas credenciais. Todavia, se não estiverem corretas, a autenticação será determinada como incorreta. Caso as credenciais estejam corretas e a conta válida, o técnico ficará automaticamente autenticado. Contudo, na condição de não conter uma conta válida, esta deverá ser validada e para isso, este terá de inserir o código de validação. No caso de o código estar correto, a conta será validada e o técnico ficará autenticado. Caso contrário o código será inválido e este necessitará indicar o código de validação novamente.

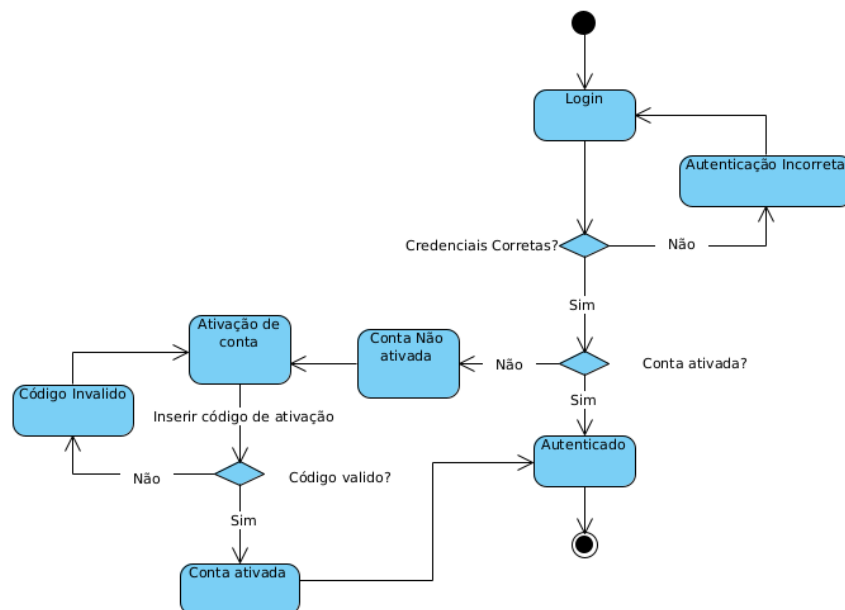


Figura 3.43: Diagrama de estados de autenticação e validação de conta

3.17 Diagrama de sequência

A realização da autenticação, ativação e confirmação da conta requerem procedimentos extras e seguem determinadas regras. Deste modo, foi necessário elaborar diagramas de sequência para especificar as interações com o sistema.

3.17.1 Diagrama de sequência Login e ativação de conta

O objetivo do diagrama da figura 3.44 consiste em demonstrar, que assim que o técnico realiza o login, é necessário verificar as credenciais. Se estas estiverem incorretas, este receberá uma mensagem de erro. Contudo, caso estas estejam válidas e a conta estiver ativada o técnico ficará autenticado.

Se porventura o técnico colocar as credenciais corretas, mas a conta não estiver ativada, este irá realizar a ativação da conta, onde poderá enviar o código de ativação. Se eventualmente estiver correto, a conta será ativada, caso contrário, este receberá uma mensagem de erro. Porém, o técnico também terá como hipótese cancelar a ativação da conta e/ou pedir um novo *email* de ativação, onde será pedido um novo código ao servidor e este automaticamente será gerado e enviado.

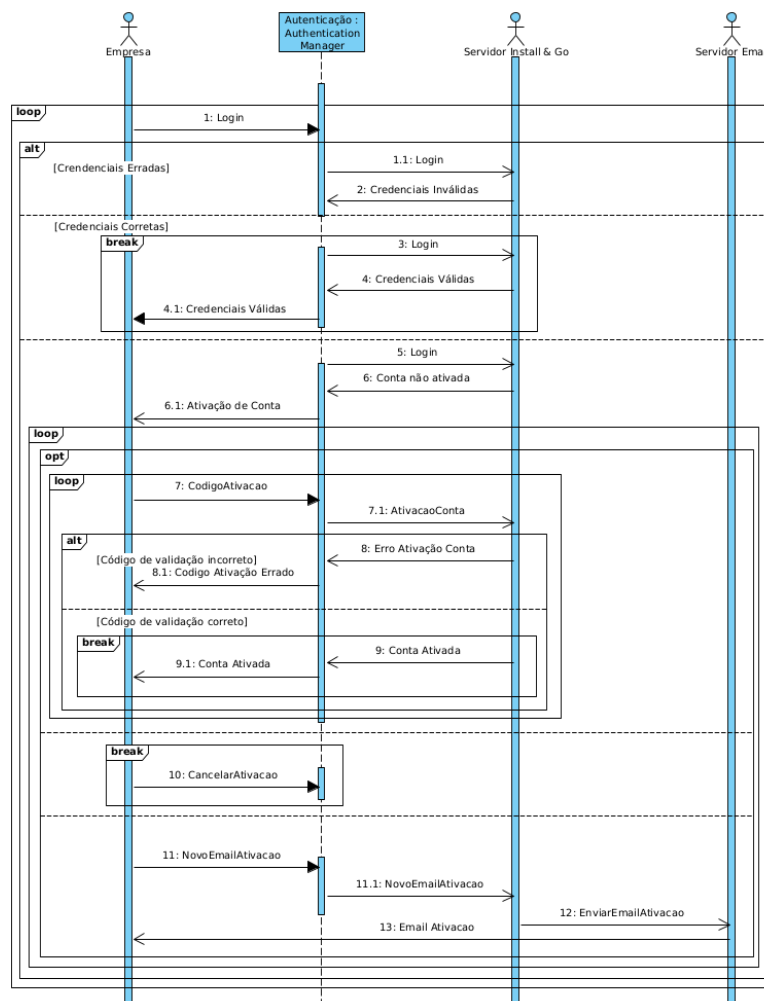


Figura 3.44: Diagrama de sequência de login e ativação de conta

3.17.2 Diagrama de sequência Registo e ativação de conta

Através do diagrama da figura 3.45 é possível perceber que quando uma empresa realiza o registo, este será enviado para o servidor, o qual registrará a empresa com uma conta não ativada. Esta conta será validada pela Motorline, de seguida é gerado um código de ativação e por fim é enviado para o *email* de registo. Após o registo ou o clique em ativar a conta no *email* recebido, a empresa será encaminhada para a validação de conta e toda a validação ocorre conforme o processo mencionado no anteriormente.

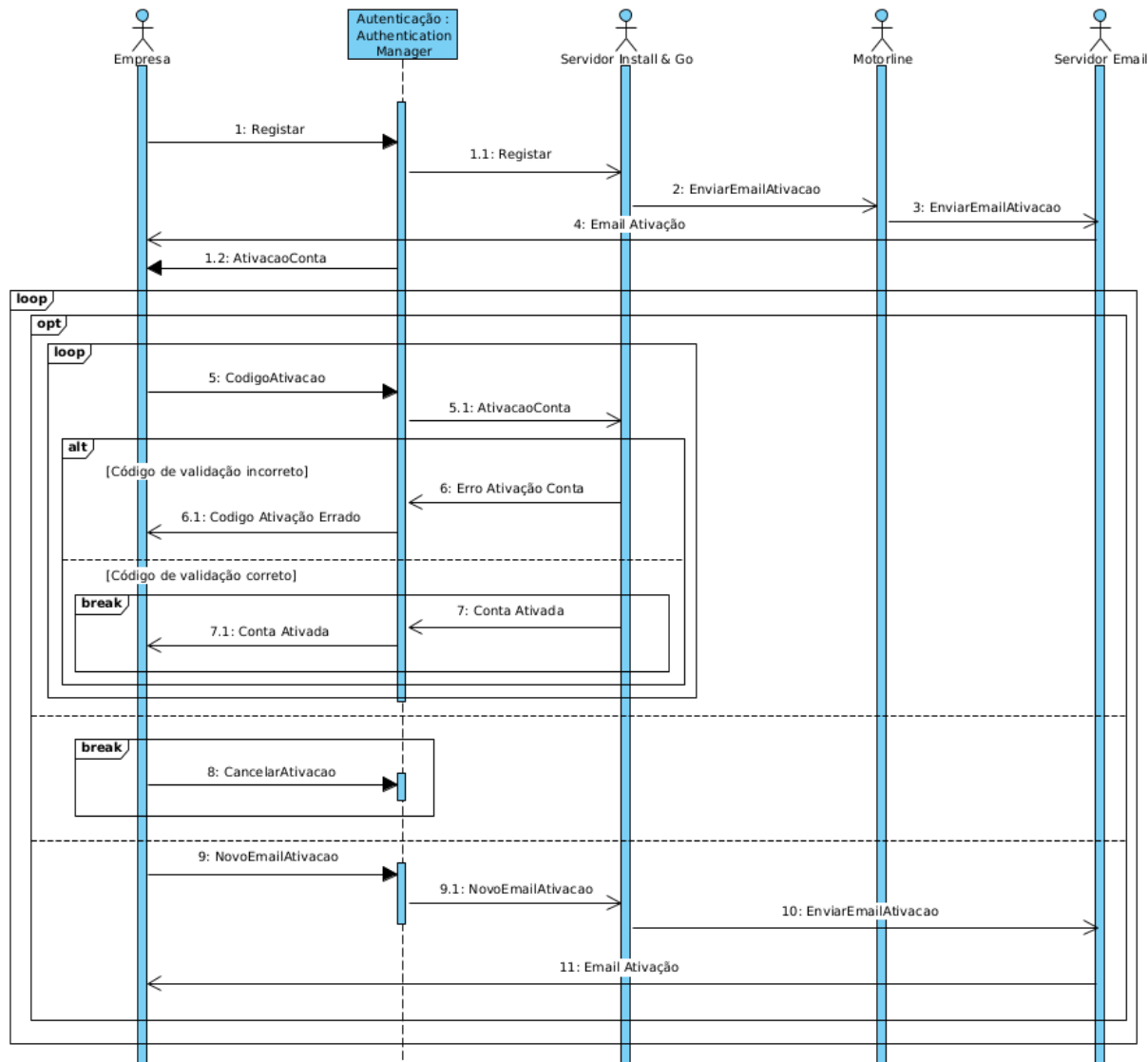


Figura 3.45: Diagrama de sequência de registo e validação de conta

3.17.3 Diagrama de sequência registro de técnicos

O diagrama da figura 3.46 tem como propósito dar a perceber que quando uma empresa deseja registrar um técnico esta introduzirá os seus dados no sistema, o que leva estes a serem enviados e a conta criada. Posteriormente, um código de ativação é gerado e enviado para o técnico poder ativar a sua conta, este processo segue a mesma orientação mencionada anteriormente.

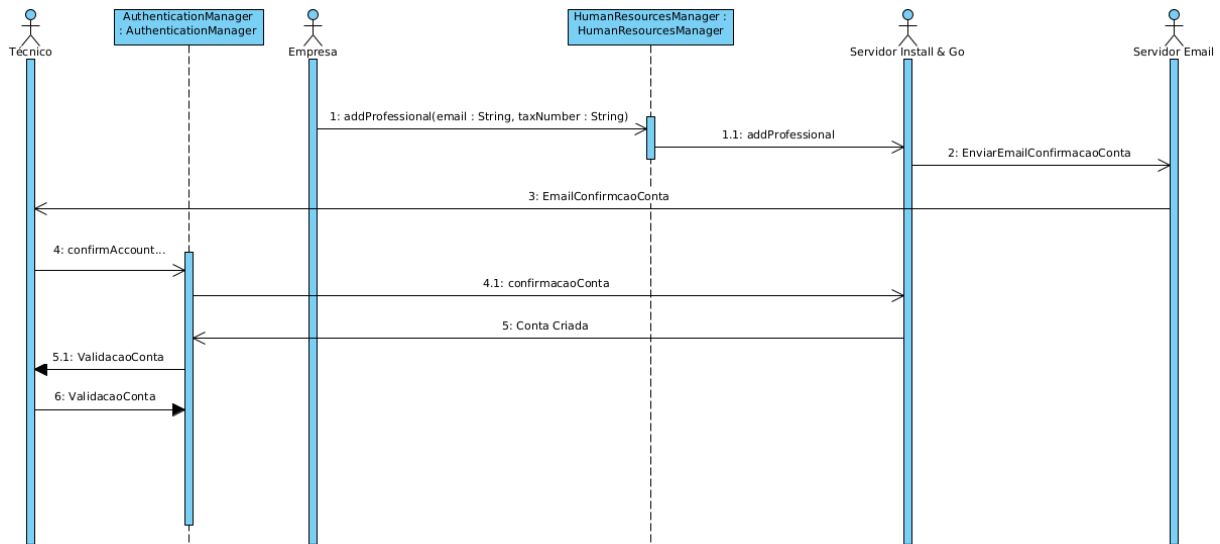


Figura 3.46: Diagrama de sequência de registo de técnicos

3.18 Arquitetura de sistema

A figura 3.47 expõe a arquitetura do sistema que indica as principais componentes. Entre estas, é possível visualizar a aplicação *frontend*, que realiza pedidos a uma aplicação *backend* e espera respostas. O *backend* é composto por uma *api rest* que recebe e responde aos pedidos e por uma base de dados a qual recebe *queries* e devolve dados para a *api rest*.

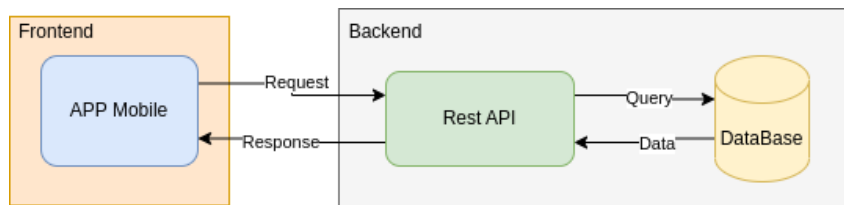


Figura 3.47: Arquitetura do sistema

3.18.1 Arquitetura de funcional

A especificação da implementação da *api rest* foi realizada através de uma arquitetura de *backend* (Figura 3.48). Aqui, é possível visualizar que sempre que a *API* recebe um pedido, este é redirecionado primeiramente para o *router*. O *router* tem como função identificar a rota referente ao pedido e deslocar para os respectivos *middlewares*.

Os *middlewares* têm como objetivo realizar todas as ações necessárias antes de proceder à execução do código de rota. Os *middlewares* existentes são o *SessionTokenValidator*, valida a sessão do utilizador a realizar o pedido, de forma similar, o *middleware RefreshTokenValidator*, valida a sessão principal do utilizador e por fim, o *middleware RoleValidator*, valida se o utilizador que efetua o pedido tem cargos suficientes para tal. Na eventualidade de não existir nenhum impedimento, o pedido é direcionado para o *controller*.

O *controller* extrai os dados do pedido, verifica se os dados obrigatórios existem e cumprem as regras de negócio e encaminha o pedido para o serviço. O serviço, em caso de necessidade, irá proceder à interação com a base de dados, esta realiza diversas ações como, obter, atualizar, apagar e inserir dados. Por fim, após todo o código de serviço ser executado, a resposta é formulada e devolvida para o utilizador.

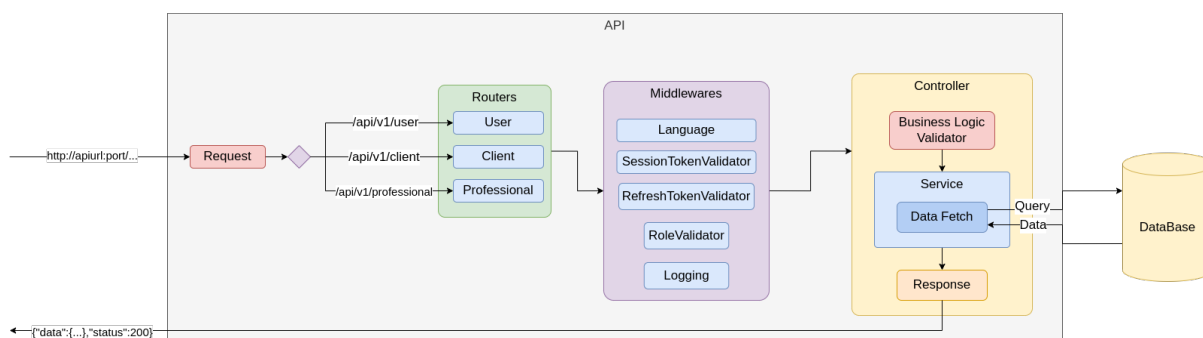


Figura 3.48: Arquitetura do funcional

3.18.2 Arquitetura de componentes

A arquitetura de componentes contém todos os serviços que deverão ser implementados na *api frontend*, com a identificação dos atores que poderão realizar estes pedidos. Esta foi desenvolvida após uma análise de todos os dados necessários para suporte do *frontend*.

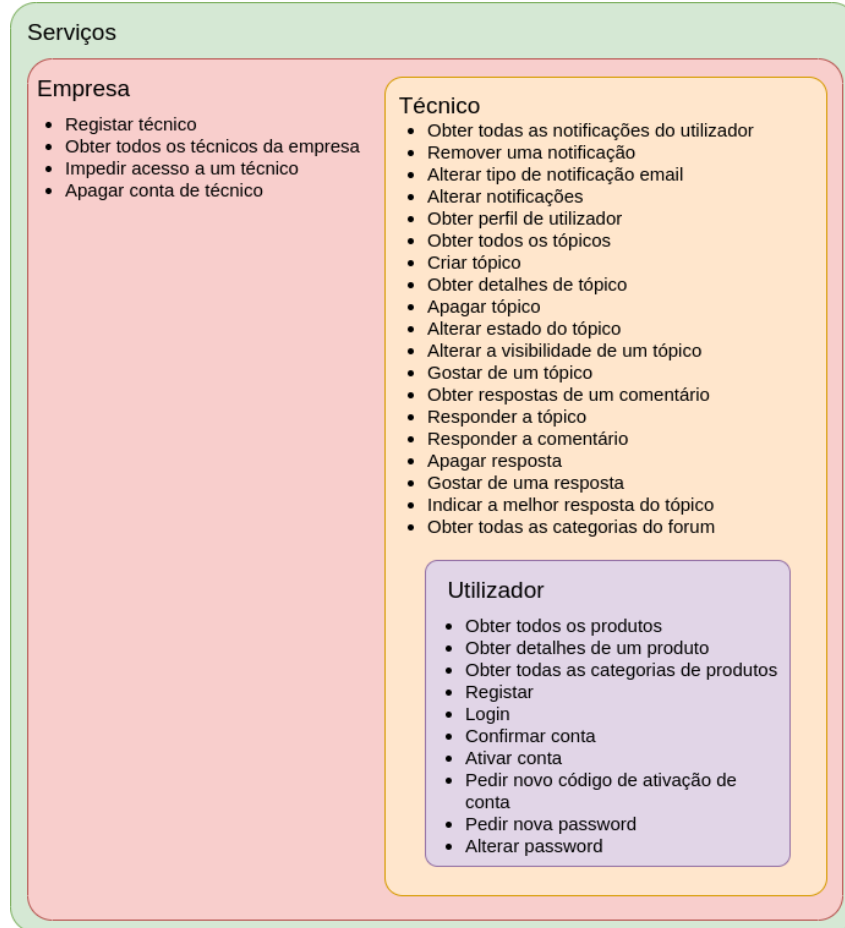


Figura 3.49: Arquitetura de componentes

3.18.3 Tabela de endpoints

Com o propósito de evitar colisões de *endpoints* durante a implementação, foi desenvolvida uma tabela de *endpoints*. Esta possui dados semelhantes à arquitetura de componentes (Figura 3.49), mas para cada serviço é indicada a rota e o método a utilizar.

Tabela 3.19: Tabela de endpoints

Serviço	Ator	Rota	Método
Obter informações do utilizador	Cliente	baseurl/client/:uid	GET
Realizar login	Utilizador	baseurl/login	POST
Realizar registo	Utilizador	baseurl/register	POST
Esquecimento de <i>password</i>	Utilizador	baseurl/forgot- <i>password</i>	GET
Ativação de conta	Cliente	baseurl/client/:uid/activate	POST
Reenvio de código de ativação de conta	Cliente	baseurl/client/:uid/new-code	GET
Obter tópicos em destaque	Cliente	baseurl/client/topics/featured	GET
Obter tópicos mais recentes	Cliente	baseurl/client/topics/latest	GET
Obter tópicos por responder	Cliente	baseurl/client/topics/to-answer	GET
Obter tópicos do utilizador	Cliente	baseurl/client/topics	GET
Obter tópicos privados	Técnico	baseurl/professional/topics/private	GET
Gostar de um tópico	Cliente	baseurl/client/topics/:topicId/like	PUT
Adicionar resposta a tópico	Cliente	baseurl/client/topics/:topicId/answer	POST
Adicionar resposta a outra resposta	Cliente	baseurl/client/answers/:answerId/	POST
Gostar de uma resposta	Cliente	baseurl/client/answers/:answerId/like	PUT
Marcar tópico como completo	Cliente	baseurl/client/topics/:topicId/completed	PUT
Remover Tópico	Cliente	baseurl/client/topics/:topicId/	DELETE
Alterar visibilidade do tópico	Cliente	baseurl/client/topics/:topicId/visibility	PUT

Continued on next page

Tabela 3.19: Tabela de endpoints (Continued)

Adicionar melhor resposta do tópico	Cliente	baseurl/client/topics/:topicId/answers/:answerId/best-answer	PUT
Adicionar novo tópico	Cliente	baseurl/client/topics/	POST

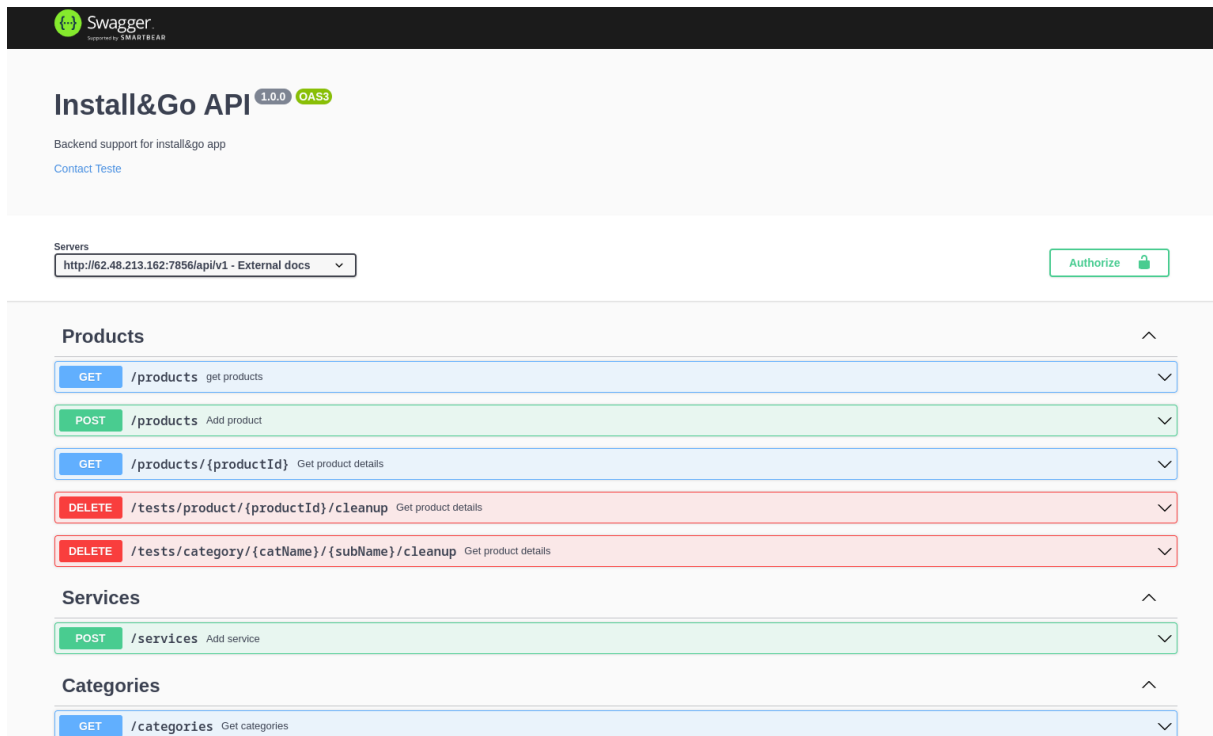


Figura 3.50: Listagem de serviços documentados