



UNIVERSIDADE  
**LUSÓFONA**  
DO PORTO

**YOURSPOT**

---

*Gabriel Faria de Souza Lima - 21800381*

*Wendel da Silva Guimarães - 21802021*

*Docente: José Vasconcelos*

## Índice

Versão .....	7
1. Descrição do projeto de software.....	8
1.1. Propósito do projeto. ....	9
1.2. Calendarização do projeto.....	9
1.2.1. Enumeração dos marcos a atingir (Milestones).....	9
1.3. Mockups e Protótipos. ....	10
2. Requisitos de software.....	11
2.1. Atores do sistema e casos de uso. ....	11
2.1.1 Principais Exclusões.....	13
2.2 Diagrama de caso de uso.....	14
2.2.1. Casos de uso: Pacote Gestão Pessoal. ....	15
2.2.2. Casos de uso: Pacote Área Acadêmica .....	17
2.3. Requisitos funcionais e domínio. ....	20
2.4. Requisitos não funcionais.....	29
3. Software Design.....	31
3.1. Diagrama de Classes .....	31
3.2. Diagrama de Componentes.....	31
3.3. Mind Map .....	32
3.4 Modelo Entidade-associação YourSpot .....	33
3.4.1 Modelo Relacional YourSpot.....	34
3.5 User Interface Design.....	35
3.6. Padrão de Arquitetura .....	40
4. Software Construction .....	40
4.1. - Descrição do Sistema Desenvolvido.....	40
4.2. - Abordagem a nível de programação.....	42
5. Implementação .....	44
5.1. Fases de Desenvolvimento.....	44
5.2. Ferramentas e bibliotecas de software.....	46
6. Avaliação da Solução.....	49
6.1. Ambiente e Metodologia de Teste.....	49
6.2. Testes funcionais .....	50
6.3. Testes não funcionais.....	56
7. Conclusões e Trabalho Futuro .....	57
7.1. Conclusões .....	57
7.2. Trabalho Futuro.....	58

## Índice de Figuras

<b>Figura 1</b> - Diagrama de pacotes do sistema YourSpot .....	12
<b>Figura 2</b> - Diagrama de caso de uso completo do sistema YourSpot .....	14
<b>Figura 3</b> - Mind Map do sistema YourSpot .....	32
<b>Figura 4</b> - Modelo Relacional YourSpot.....	34
<b>Figura 5</b> - Persona 1 YourSpot.....	36
<b>Figura 6</b> - Persona 2 YourSpot.....	36
<b>Figura 7</b> - User Scenario Persona 1 .....	37
<b>Figura 8</b> - User Scenario Persona 2 .....	38
<b>Figura 9</b> - Diagrama de Fluxo de Processos YourSpot.....	39
<b>Figura 10</b> - Processo de Comunicação Simplificado YourSpot.....	43
<b>Figura 11</b> - Trabalho futuro, Chat de mensagens YourSpot .....	58
<b>Figura 12</b> - Trabalho futuro, Configurações do YourSpot .....	59

## Índice de Tabelas

<b>Tabela 1</b> – Atores do sistema YourSpot .....	12
<b>Tabela 2</b> – Pacotes de casos de uso do sistema YourSpot .....	13
<b>Tabela 3</b> – Processos não suportados no sistema YourSpot .....	13
<b>Tabela 4</b> – Caso de uso “Criar Tabela” .....	15
<b>Tabela 5</b> – Descrição de fluxo “Criar Tabela” .....	15
<b>Tabela 6</b> – Caso de uso “Adicionar Tabela” .....	16
<b>Tabela 7</b> – Descrição de fluxo “Adicionar Tabela” .....	16
<b>Tabela 8</b> – Caso de uso “Registrar Exercícios” .....	17
<b>Tabela 9</b> – Descrição de fluxo “Registrar Exercícios” .....	17
<b>Tabela 10</b> – “Descrição de fluxos de eventos alternativo do Caso de Uso “Fazer upload do arquivo” .....	18
<b>Tabela 11</b> – Caso de uso “Consultar Exercícios” .....	19
<b>Tabela 12</b> – Descrição de fluxo “Consultar Exercícios” .....	19
<b>Tabela 13</b> – Descrição das prioridades de um requisito .....	20
<b>Tabela 14</b> – Descrição Requisito funcional “Adicionar uma nova página” .....	20
<b>Tabela 15</b> – Descrição Requisito funcional “Adicionar uma nova tabela” .....	21
<b>Tabela 16</b> – Descrição Requisito funcional “Adicionar uma nova lista” .....	21
<b>Tabela 17</b> – Descrição Requisito funcional “Adicionar um calendário” .....	22
<b>Tabela 18</b> – Descrição Requisito funcional “Adicionar várias listas em uma página” .....	22
<b>Tabela 19</b> – Descrição Requisito funcional “Adicionar título em uma página” .....	23
<b>Tabela 20</b> – Descrição Requisito funcional “Adicionar divisorias em uma página” .....	23
<b>Tabela 21</b> – Descrição Requisito funcional “Adicionar um link de redirecionamento em uma página” .....	23
<b>Tabela 22</b> – Descrição Requisito funcional “Adicionar imagens em uma página” .....	24
<b>Tabela 23</b> – Descrição Requisito funcional “ficheiros em uma página” .....	24
<b>Tabela 24</b> – Descrição Requisito funcional “Adicionar vídeos em uma página” .....	25
<b>Tabela 25</b> – Descrição Requisito funcional “Consultar exercícios” .....	25
<b>Tabela 26</b> – Descrição Requisito funcional “Consultar textos” .....	25
<b>Tabela 27</b> – Descrição Requisito funcional “Consultar livros” .....	26
<b>Tabela 28</b> – Descrição Requisito funcional “Guardar exercícios” .....	26
<b>Tabela 29</b> – Descrição Requisito funcional “Guardar textos” .....	27
<b>Tabela 30</b> – Descrição Requisito funcional “Guardar livros” .....	27
<b>Tabela 31</b> – Descrição Requisito funcional “Registrar exercícios” .....	27
<b>Tabela 32</b> – Descrição Requisito funcional “Registrar textos” .....	28
<b>Tabela 33</b> – Descrição Requisito funcional “Registrar livros” .....	28
<b>Tabela 34</b> – Descrição Requisito não funcional “Iniciar sessão no sistema” .....	29
<b>Tabela 35</b> – Descrição Requisito não funcional “Encerrar sessão no sistema” .....	29
<b>Tabela 36</b> – Descrição Requisito não funcional “Interoperabilidade” .....	30
<b>Tabela 37</b> – Descrição Requisito não funcional “Componentes” .....	30
<b>Tabela 38</b> - Testes Funcionais YourSpot .....	50

## Abstract

O ambiente acadêmico é um dos campos mais importantes de nossa sociedade, é por meio dela que um indivíduo consegue dar asa aos seus pensamentos que posteriormente o levam aos seus sonhos. É notável como o conhecimento evoluiu e com isso, atividades acadêmicas se tornaram atividades muito mais recompensadoras porém complexas e em um mundo que não para e pessoas muitas vezes são confrontadas com inúmeros desafios a mais do que apenas sua educação, este documento procura descrever uma proposta para a melhoria da gestão pessoal e a motivação, integração e acompanhamento de todo o âmbito acadêmico que possa rodear um indivíduo da atualidade. Um dos maiores desafios enfrentado pela parte interessada na obtenção de conhecimento, surge quando são submetidos a inúmeros deveres, afazeres e datas o que gera um ambiente desorganizado e propício ao não cumprimento dos mesmos, o YourSpot surge como uma solução aos problemas do dia a dia sendo implementado para ser uma ferramenta de criação de ambiente pessoal aonde um indivíduo pode armazenar todos seus apontamentos pessoais e adquirir todo conhecimento disponível pela própria plataforma como é abordado ao decorrer deste documento, tudo com o incentivo da evolução pessoal de cada um dos utilizadores do sistema.

**Keywords:** Academia, Gestão pessoal

## **Agradecimentos**

Primeiramente, gostaria de agradecer a minha família, que sempre apoiou-me no percurso acadêmico.

A minha namorada que vem dando-me grande suporte não só desde o início deste projeto, mas também do curso.

Aos professores e amigos que dadas as suas devidas competências auxiliaram na construção deste documento e da aplicação.

E ao meu companheiro de projeto, Gabriel, que juntos conseguimos realizar este trabalho da melhor e da mais completa forma possível.

## Versão

Versão	Data	Autores	Comentário
01	2020-11-27	Gabriel Faria & Wendel Guimarães	Arquitetura do sistema

Versão	Data	Autores	Comentário
02	2021-01-08	Gabriel Faria & Wendel Guimarães	<p>Nova documentação com foco especial no design do projeto, adicionado um capítulo referente aos novos diagramas do projeto.</p> <p>Atualização grande nos MockUps do projeto, para a melhor representação do design do app possível!</p>

Versão	Data	Autores	Comentário
03	2021-01-13	Gabriel Faria & Wendel Guimarães	Adição dos modelos entidade associação e modelo relacional referente a base de dados do sistema YourSpot

Versão	Data	Autores	Comentário
04	2021-03-31	Gabriel Faria & Wendel Guimarães	Adição do tópico 3.5 relativo ao “User Interface Design” e o tópico 4 que trata das questões do “Software Construction”.

Versão	Data	Autores	Comentário
05	2021-05-10	Gabriel Faria & Wendel Guimarães	Correção na escrita geral do documento, reestruturação do documento, adição de capítulos 5, 6 e 7

## 1. Descrição do projeto de software.

---

Trata-se de uma ferramenta de gestão de informação pessoal para o âmbito acadêmico, fornecendo um espaço virtual com possibilidade de organizar e manter qualquer tipo de informação pertinente ao utilizador, podendo separar cada uma dessas informações em uma categoria específica e de acordo com o gosto do utilizador, como por exemplo uma área de rastreamento das tarefas relacionadas a cada uma das suas disciplinas acadêmicas, podendo acompanhar o estado delas e utilizá-las quando necessárias. Ou uma área onde pode encontrar indicações feitas pelos professores relacionadas a algum assunto específico, salvar exercícios, resoluções ou até mesmo links que direcionam para uma reunião online de alguma unidade curricular. O projeto surge como forma de automatizar operações não necessariamente administrativas, porém pessoal do dia a dia de um estudante ou professor no âmbito acadêmico e além disso proporcionar uma gestão própria e personalizada, utilizando-se de todas as funcionalidades do projeto para qualquer outro tipo de finalidade que possa auxiliar o utilizador.

De modo a visar a organização pessoal tanto do aluno como do professor, o projeto além de automatizar as ações citadas acima, visa também melhorar as condições de comunicação entre ambos os utilizadores e facilitar os aspectos acadêmicos desde alterações no calendário pessoal acadêmico publicado até mesmo atividades e leituras referentes às respectivas matérias desejadas, tal como dispor de materiais que possam vir a ser importantes nesse âmbito. De momento também tem sua utilidade destacada devido a pandemia gerada do vírus Covid-19, dessa forma, com tantas mudanças não previstas, nosso projeto teria papel constante no dia a dia de modo a deixar essas mudanças um pouco mais confortáveis para o utilizador se adequar e organizar.



## **1.1. Propósito do projeto.**

A intenção e o desejo final para o projeto é a adoção efetiva do aplicativo pela parte interessada, ou seja o projeto conseguir prover todos os objetivos e garantir uma ferramenta confiável, acessível, personalizável e diferenciada. Podendo salvar horas acumuladas de trabalho desnecessário, sendo uma ajuda que efetivamente pode melhorar níveis acadêmicos, sociais, organizacionais e até mesmo emocionais do utilizador com a melhor gestão de suas atividades, direitos e deveres através da customização pessoal.

Em épocas incertas como a atual de uma pandemia, o aplicativo pode ser uma grande ferramenta para professores e alunos que necessitam de uma organização online em outro patamar. O comprometimento e propósito deste projeto consiste no sucesso do utilizador em fazer uso de todas as ferramentas que serão desenvolvidas e aplicá-las no seu dia a dia, dessa forma obtendo melhores desempenhos e auxiliando na sua organização, além de pôr em discussão possíveis melhores formas de se comunicar e transmitir informações entre aluno e professor.

## **1.2. Calendarização do projeto.**

A calendarização do projeto feito em um diagrama de Gantt se encontra [neste link](#).

### **1.2.1. Enumeração dos marcos a atingir (Milestones).**

O projeto tem 5 metas como principais marcos a se atingir.

- **Levantar os requisitos**

Como de costume todo projeto se inicia com a descoberta dos requisitos, através das fontes de requisitos como os interessados no projeto.



- **Especificação dos requisitos**

Após a descoberta dos requisitos e após a fase de análise e negociação movemos para especificação dos requisitos onde cada requisito passará por uma rigorosa descrição para se entender o que se espera de cada requisito.

- **Bases de dados**

Criação da base de dados com todas as tabelas a serem analisadas para o melhor cumprimento de suas funcionalidades e integração com o sistema.

- **Aplicação (código)**

Utilização do conhecimento em programação para o uso efetivo dos dados da base de dados com uma interface apropriada para o mesmo, todo o trabalho de interação com o utilizador, design gráfico e funcional será atribuída a essa milestone.

- **Testes**

Etapas de testar o sistema como um todo e verificar se é satisfatório e cumpre com todos os requisitos apresentados e suas respectivas funcionalidades.

### **1.3. Mockups e Protótipos.**

Prototipar é um método comum de validação no processo de interpretação do software, também pode ser usada para o descobrimento de novos requisitos e tem papel fundamental no desenvolvimento de software. Existem inúmeras técnicas de prototipagem que poderiam ser implementadas para estes objetivos, o método escolhido foi criar em uma plataforma um mockup virtual que poderia facilitar no processo de interpretação de como os casos de usos e os requisitos são implementados em um contexto e exibidos a parte interessada no projeto, e podendo trazer mais feedbacks com a “utilização” de um protótipo responsivo e dinâmico. O protótipo também é uma ótima forma de antecipar requisitos falhos ou ambíguos e poupar o processo de uma alteração posterior muito dispendiosa por tantas alterações que seriam feitas.

Todos os mockups implementados na plataforma podem e serão modificados com inserção de todas as funcionalidades pretendidas e um polimento do produto final extremamente maior em questão visual e funcional, o mockup apenas oferece uma ideia vaga de como seria alguma das interfaces gráficas do sistema, não possuindo todas as interfaces que podem ser adicionadas no resultado final. A interface gráfica do sistema tem como principal objetivo combater a frustração dos utilizadores,

tentando promover uma interface de fácil aprendizagem, eficiente e com alta memorabilidade ou seja a proficiência após período sem utilização, o programa deve ser intuitivo e auto explicativo para os utilizadores.

O mockup para o projeto YourSpot encontra-se [neste link](#).

## 2. Requisitos de software.

---

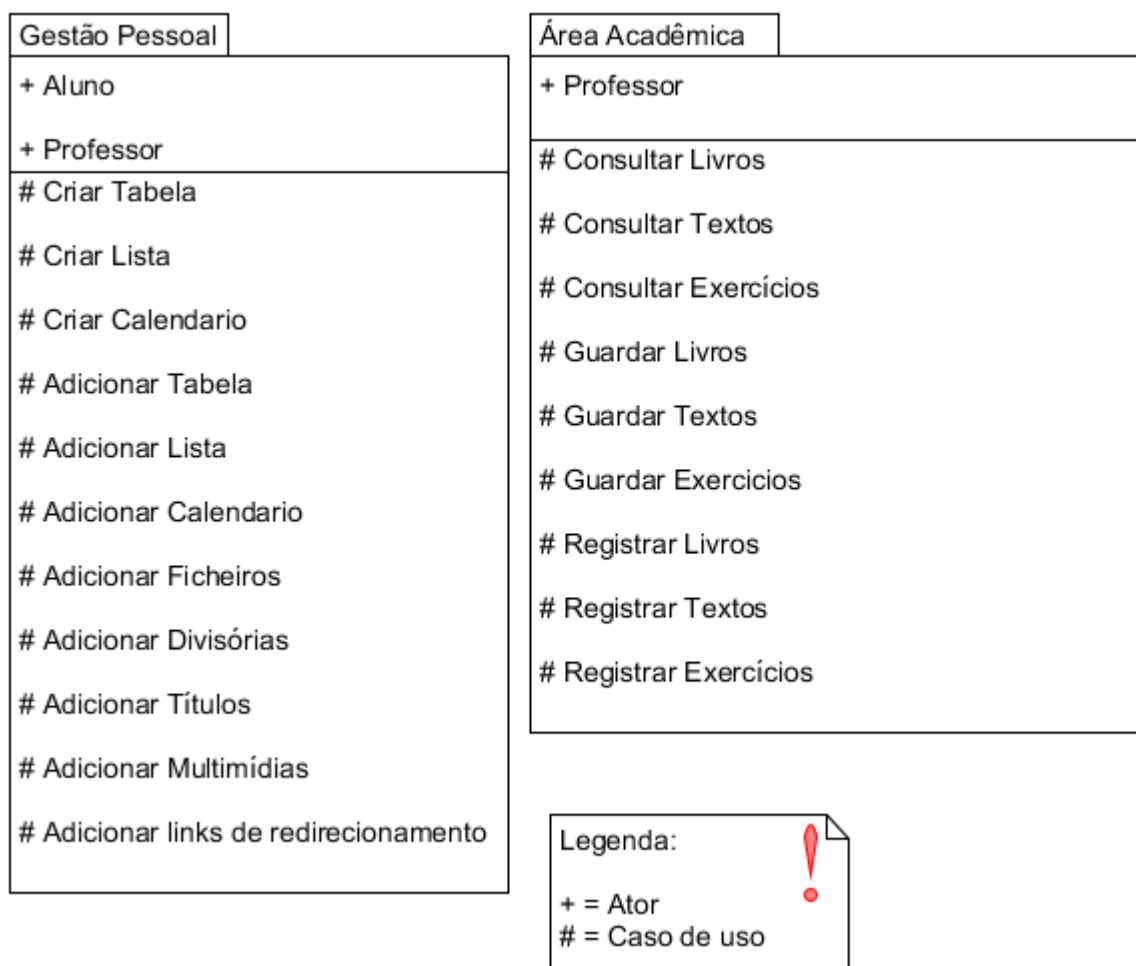
Este capítulo se preocupa em elicitar, analisar e especificar os requisitos do software do projeto, oferecendo uma análise a fundo de como os atores interagem com o sistema, representado através de casos de uso com suas devidas explicações de fluxograma em busca de munir o documento com documentação de apoio ao desenvolvimento do projeto.

Os requisitos de software expressam as necessidades e restrições impostas a um produto de software que contribuem para a solução de algum problema do mundo real.

A informação organizada neste capítulo é resultado de um processo de desenvolvimento de requisitos. Nesse processo, foi recolhida informação referente ao domínio do projeto e posteriormente analisada e organizada sob a forma de requisitos que descrevem o sistema a desenvolver.

### 2.1. Atores do sistema e casos de uso.

Na figura 1 representa-se o modelo genérico de casos de uso do sistema **YourSpot** sob a forma de um diagrama de pacotes. Cada pacote agrega uma ou mais partes do sistema que se destinam a suportar processos da organização e/ou a reunir um conjunto de funcionalidades. Em cada pacote incluem-se alguns exemplos de Atores e casos de uso desenvolvidos para o sistema. São representadas igualmente as relações de dependência entre os pacotes. Os atores são somente incluídos nos pacotes em que surgem pela primeira vez. O maior objetivo desse diagrama é a representação do relacionamento dos atores com as funcionalidades previstas no nosso sistema, exibindo de forma clara e de fácil compreensão o que é esperado que cada ator consiga fazer em nosso sistema.



**Figura 1 - Diagrama de pacotes do sistema YourSpot**

Na tabela 1 apresenta-se uma descrição de cada Ator do sistema.

**Tabela 1 – Atores do sistema YourSpot**

Ator	Descrição resumida
Aluno	O aluno interage com o sistema e utiliza de todas as funcionalidades previstas na área de “gestão pessoal” para a eventual criação de sua própria área personalizada e também pode acessar a área acadêmica para visualização ou armazenamento de qualquer conteúdo disponibilizado na plataforma.
Professor	Também pode acessar todas as funcionalidades de um aluno no sistema, tendo sua diferenciação na “Área Acadêmica” onde o mesmo pode participar efetivamente na criação e publicação de novos conteúdos para a plataforma.

Na tabela 2 descreve-se sumariamente cada um dos pacotes do sistema **Your Spot**. Os capítulos seguintes deste documento destinam-se à descrição em detalhe de cada pacote, dos casos de uso que incluem e da forma como cada Ator interage com o sistema.

***Tabela 2 – Pacotes de casos de uso do sistema YourSpot***

<b><i>Pacote</i></b>	<b><i>Descrição resumida</i></b>
<b>1. Gestão Pessoal</b>	Pacote responsável por toda a área de armazenamento e exibição de informação, os atores que possuem acesso a esse pacote podem então organizar, adicionar, criar e personalizar qualquer tipo de data da forma que o pretenderem.
<b>2. Área Académica</b>	Pacote associado aos diversos conteúdos académicos disponibilizados pela plataforma para engrandecimento do utilizador, neste pacote serão armazenadas e exibidas todo tipo de informação pertinente à vida académica de um utilizador do sistema.

### **2.1.1 Principais Exclusões.**

Na tabela 3 indicam-se os processos que não são apoiados pelo sistema **YourSpot**. A justificação para as exclusões relaciona-se com o facto da falta de recursos.

***Tabela 3 – Processos não suportados no sistema YourSpot***

<b><i>Processo</i></b>	<b><i>Descrição resumida</i></b>
<b>1. Integração de universidades</b>	O YourSpot não suporta uma Integração dinâmica das universidades para cada um dos casos dos alunos ou professores, ou seja o utilizador do sistema poderia se beneficiar das funcionalidades do sistema para armazenamento de todo tipo de compromissos de sua universidade pessoal, porém o sistema não conseguiria fazer uma vinculação completa da mesma e não conseguiria de forma automática exibir informações( aulas, provas, horários, comunicados) referentes a universidade específica de cada um dos utilizadores.

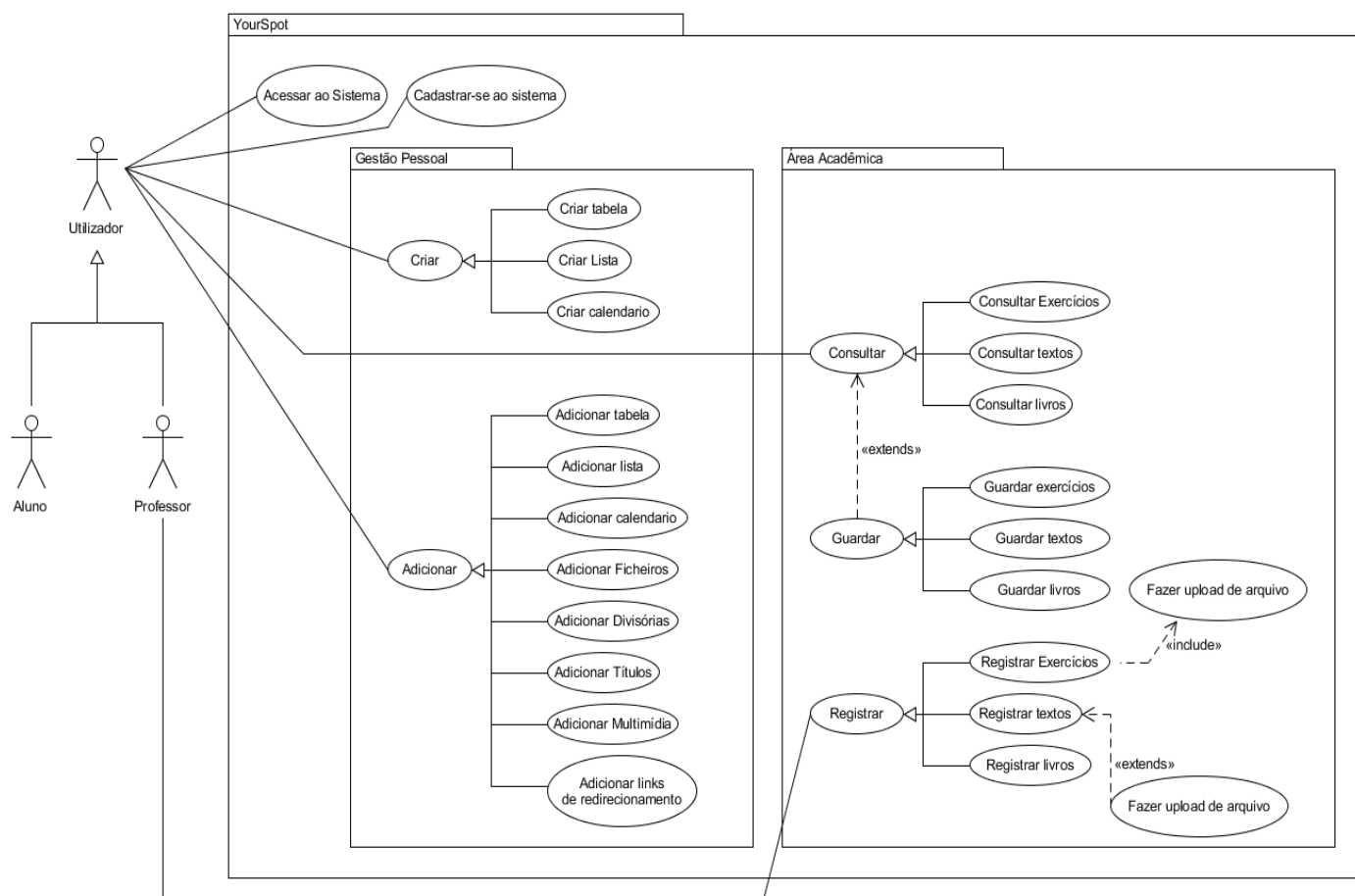


## 2. Diagramas de Gantt

O YourSpot ainda não consegue exibir dados de forma dinâmica com um diagrama de Gantt, com todas suas dependências e características principais.

## 2.2 Diagrama de caso de uso

Na figura 2 apresenta-se o diagrama de casos de uso referente ao sistema completo do YourSpot, com dois pacotes principais já mencionados previamente.



**Figura 2 - Diagrama de caso de uso completo do sistema YourSpot**

Nas secções seguintes apresenta-se uma descrição de alguns ( tendo em vista que muitos partilham de uma linha de fluxo semelhante) dos casos de usos, com indicação dos estado de descrição do caso de uso, respetivos fluxos de eventos e requisitos necessários para a implementação do mesmo no sistema.

## 2.2.1. Casos de uso: Pacote Gestão Pessoal.

### 2.2.1.1 Criar Tabela.

***Tabela 4 – Caso de uso “Criar Tabela”***

GHM.CSU.001	Criar Tabela
<b>Objetivo</b>	Criação de uma nova página em formato de tabela.
<b>Descrição sumária</b>	O utilizador é apresentado com uma série de opções no caso de uma criação de nova página, o mesmo pode escolher a criação de uma tabela, a partir daí o utilizador tem total liberdade de armazenar a sua informação e personalizar a estrutura para o seu gosto pessoal.
<b>Pré-condições</b>	“Acessar ao Sistema”.
<b>Pós-condições</b>	----
<b>Estado</b>	Descreve o cenário principal sob ponto de vista externo com enfoque deliberado na usabilidade.

***Tabela 5 – Descrição de fluxo “Criar Tabela”***

Cenário Principal		
Passos	Descrição	Pontos de extensão
1	O utilizador escolhe criar uma nova página na sua área de gestão pessoal.	
2	O utilizador escolhe a opção de criar a página como sendo uma tabela.	
3	O utilizador pode agora armazenar informação ou personalizar a tabela de acordo com a sua preferência pessoal.	

### 2.2.1.2 Adicionar Tabela

***Tabela 6 – Caso de uso “Adicionar Tabela”***

GHM.CSU.001	Adicionar Tabela
<b>Objetivo</b>	Adição de uma nova tabela dentro de uma página já existente.
<b>Descrição sumária</b>	O utilizador escolhe é apresentado com uma série de opções no caso de uma criação de nova página, o mesmo pode escolher a criação de uma tabela, a partir daí o utilizador tem total liberdade de armazenar a sua informação e personalizar a estrutura da tabela para o seu gosto pessoal.
<b>Pré-condições</b>	“Acessar ao Sistema”.
<b>Pós-condições</b>	----
<b>Estado</b>	Descreve o cenário principal sob ponto de vista externo com enfoque deliberado na usabilidade.

***Tabela 7 – Descrição de fluxo “Adicionar Tabela”***

Cenário Principal		
Passos	Descrição	Pontos de extensão
1	O utilizador escolhe criar uma nova página na sua área de gestão pessoal.	
2	O utilizador escolhe a opção de criar a página como sendo uma tabela.	
3	O utilizador pode agora armazenar informação ou personalizar a tabela de acordo com a sua preferência pessoal.	



## 2.2.2. Casos de uso: Pacote Área Acadêmica

### 2.2.2.1 Registrar Exercício

***Tabela 8 – Caso de uso “Registrar Exercícios”***

<b>GHM.CSU.001</b>	<b>Registrar Exercícios</b>
<b>Objetivo</b>	Professor registra um exercício pessoal com suas devidas características, disponibilizando-o para os utilizadores da plataforma.
<b>Descrição sumária</b>	O professor estando na área acadêmica e explorando a opção de exercícios, pode criar um novo exercício, aplicando as filtragens de acordo com o nível do exercício, a unidade disciplinar do exercício e o curso a qual o professor entende que se designa esse exercício criado.
<b>Pré-condições</b>	Acessar ao Sistema, autenticar-se como um professor.
<b>Pós-condições</b>	Fazer Upload do arquivo.
<b>Estado</b>	Descreve o cenário principal sob ponto de vista externo com enfoque deliberado na usabilidade.

***Tabela 9 – Descrição de fluxo “Registrar Exercícios”***

<b>Cenário Principal</b>		
<b>Passos</b>	<b>Descrição</b>	<b>Pontos de extensão→</b>
1	O utilizador autentica-se ao sistema como sendo um professor	
2	O utilizador escolhe explorar os exercícios na área acadêmica.	



3	O professor escolhe compartilhar um exercício com a plataforma	
4	O professor deve compartilhar todas as características de filtragem do exercício.	
5	O professor completa o registro do exercício.	Inclusão → upload do arquivo
6	O sistema disponibiliza o exercício para os utilizadores do YourSpot.	

***Tabela 10 – “Descrição de fluxos de eventos alternativo do Caso de Uso “Fazer upload do arquivo”***

Cenário alternativo “Fazer upload do arquivo”		
Passo	Descrição	Ponto de Inclusão
1	O sistema solicita o upload do arquivo ao professor que pretende registrá-lo como um exercício.	
2	O sistema vincula todas as características de filtragem ao exercício.	
3	Retorna o fluxo básico em “Registrar Exercício” Passo 5.	

## 2.2.2.2 Consultar Exercícios

***Tabela 11 – Caso de uso “Consultar Exercícios”***

GHM.CSU.001	Consultar Exercícios
<b>Objetivo</b>	Consultar exercícios disponibilizados na plataforma
<b>Descrição sumária</b>	O utilizador quando acessa a “Área Acadêmica” tem a opção de consultar os diversos tipos de exercícios disponibilizados na plataforma, o mesmo pode filtrar os exercícios pertinentes a seu gosto pessoal, escolhendo matéria, nível de dificuldade e o curso relacionado ao exercício pretendido.
<b>Pré-condições</b>	“Acessar ao Sistema”
<b>Pós-condições</b>	----
<b>Estado</b>	Descreve o cenário principal sob ponto de vista externo com enfoque deliberado na usabilidade.

***Tabela 12 – Descrição de fluxo “Consultar Exercícios”***

Cenário Principal		
Passos	Descrição	Pontos de extensão
1	O utilizador escolhe explorar os exercícios na área acadêmica.	
2	O utilizador pode agora filtrar os exercícios de acordo com seu gosto pessoal, escolhendo entre matéria, nível de dificuldade e o curso relacionado ao exercício.	

## 2.3. Requisitos funcionais e domínio.

De modo que no escopo do projeto será necessário uma definição de prioridades a serem dadas aos requisitos baseadas no seu nível de importância dentro do funcionamento do sistema e na sua implementação, foram adotados os níveis “Essencial”, “Importante” e “Desejável”.

***Tabela 13 – Descrição das prioridades de um requisito***

<b><i>Prioridade</i></b>	<b><i>Descrição</i></b>
<b>1. Essencial</b>	Tipo de requisito o qual o sistema não entrará em funcionamento sem o mesmo, são requisitos considerados imprescindíveis para que o sistema decorra de forma correta.
<b>2. Importante</b>	Requisitos importantes são requisitos extremamente necessários, o sistema conseguirá funcionar sem o mesmo porém não de forma satisfatória. Estes requisitos devem ser implementados logo que possível.
<b>3. Desejável</b>	São requisitos que não conflitam com as funcionalidades básicas do sistema, é possível ter um sistema em funcionamento sem eles. Estes tipos de requisitos podem ser deixados para serem implementados por último e sem que o sistema seja comprometido.

***Tabela 14 – Descrição Requisito funcional “Adicionar uma nova página”***

<b>YS.REQ.0100</b>	<b>Adicionar uma nova página</b>
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	Após o utilizador ter acesso a sua área de gestão pessoal, é dada a ele a possibilidade de criar páginas(agrupamento de informações). Cada página criada pode ser preenchida com funcionalidades já pré-definidas pela aplicação. Caso o utilizador não escolha nenhuma destas, a página se “comporta” como uma folha de texto, entretanto, a qualquer momento este pode fazer uso das funções antes citadas.
<b>Motivação</b>	O ambiente gráfico da área de gestão pessoal trabalha em volta de um conceito de “páginas” e “subpáginas”, nessas páginas serão armazenadas toda e qualquer informação pertinente ao utilizador, tendo isso em conta é de extrema necessidade a implementação de um requisito que possa efetuar tal funcionalidade

***Tabela 15 – Descrição Requisito funcional “Adicionar uma nova tabela”***

<b>YS.REQ.0101</b>	<b>Adicionar uma nova tabela</b>
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	<p>Ao estar dentro da sua área de gestão pessoal um utilizador tem duas formas de adicionar uma tabela, são elas:</p> <ol style="list-style-type: none"><li>1. Dentro de uma página o utilizador pode adicionar uma tabela.</li><li>2. Quando no espaço de gestão pessoal, podemos criar uma página como uma tabela.</li></ol> <p>A tabela funcionará como uma forma de representação de informação, o utilizador poderá criar linhas e colunas de forma personalizada, e para cada célula da tabela o utilizador poderá preencher com a informação desejada(texto), essa informação pode ser formatada da maneira que bem entender.</p>
<b>Motivação</b>	<p>Uma tabela é um dos métodos mais utilizados em armazenamento de informação, se dá então a necessidade da funcionalidade de adição de uma, sendo possível criar uma página personalizada de tabela ou adicionando uma tabela já na página criada.</p>

***Tabela 16 – Descrição Requisito funcional “Adicionar uma nova lista”***

<b>YS.REQ.0102</b>	<b>Adicionar uma nova lista</b>
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	<p>Ao estar dentro da sua área de gestão pessoal um utilizador tem duas formas de adicionar uma lista, são elas:</p> <ol style="list-style-type: none"><li>1. Dentro de uma página o utilizador pode adicionar uma lista.</li><li>2. Quando no espaço de gestão pessoal, podemos criar uma página como uma lista.</li></ol> <p>A lista funcionará como um agrupamento de subpáginas, o utilizador poderá ordená-la da forma como bem entender. Dentro da lista o utilizador poderá inserir novos elementos e deletar os que já lá se encontram.</p>
<b>Motivação</b>	<p>Uma lista é um dos métodos mais utilizados em armazenamento de informação, se dá então a necessidade da funcionalidade de adição de uma, sendo possível criar uma página personalizada de lista e suas subpáginas ou adicionando uma lista já na página criada.</p>

***Tabela 17 – Descrição Requisito funcional “Adicionar um calendário”***

YS.REQ.0103	Adicionar um calendário
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	<p>Ao estar dentro da sua área de gestão pessoal um utilizador tem duas formas de adicionar um calendário, são elas:</p> <ol style="list-style-type: none"><li>1. Dentro de uma página o utilizador pode adicionar uma folha de calendário.</li><li>2. Quando no espaço de gestão pessoal, podemos criar uma página como um calendário.</li></ol> <p>O calendário funcionará como uma representação visual dos dias de um determinado mês, sendo assim o utilizador é capaz de observar eventos passados e futuros definidos por si. Poderá também inserir novos eventos, adicionando informações como nome, data e uma descrição. Todos esses eventos terão o formato de subpágina.</p>
<b>Motivação</b>	<p>Uma calendarização é um dos métodos mais utilizados em armazenamento de informação, se dá então a necessidade da funcionalidade de adição de uma, sendo possível criar uma página personalizada de calendarização ou adicionando um calendário já na página criada.</p>

***Tabela 18 – Descrição Requisito funcional “Adicionar várias listas em uma página”***

YS.REQ.0104	Adicionar Variadas Listas em uma página
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	<p>O utilizador estando dentro de uma página pode adicionar uma lista de três formas, são elas:</p> <ol style="list-style-type: none"><li>1. Lista de afazeres: Esta lista permite ao utilizador adicionar, editar, excluir ou organizar qualquer informação em um formato que permita o mesmo dar uma informação como completa, como por um exemplo um afazer que já foi feito.</li><li>2. Lista “Bulleted”: Esta lista é uma lista simples não ordenada apenas para a exibição de informação.</li><li>3. Lista Enumerada: Como o nome já diz, esta lista funciona como lista que enumera as informações de forma crescente do utilizador</li><li>4. Lista “Toggle”: Esta lista também permite ao utilizador armazenar e modificar suas informações, porém ela agrupa todas essas informações e alterna entre dois estados, o estado de exibir as informações ou o estado de escondê-las mostrando apenas a primeira informação da lista.</li></ol>
<b>Motivação</b>	<p>Um utilizador pode optar por não querer uma lista de subpáginas por qualquer motivo que seja, se dá então a necessidade de adicionar apenas em páginas já existentes, as mais famosas e recorrentes versões de uma lista para o armazenamento adequado de uma informação.</p>



***Tabela 19 – Descrição Requisito funcional “Adicionar título em uma página”***

YS.REQ.0105	Adicionar Título em uma página
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	O utilizador estando dentro de uma página pode adicionar um título para a mesma, este título funciona apenas como demarcador pessoal, ou processador de alguma informação que se relacione com o mesmo, o utilizador de acordo com sua preferência pessoal pode também definir o tamanho e a cor desse título.
<b>Motivação</b>	Um utilizador sempre irá precisar adicionar novos títulos em suas páginas, seja por organização ou melhor identificação de uma informação, se dá então a necessidade de implementar esses títulos em qualquer área que convenha o utilizador.

***Tabela 20 – Descrição Requisito funcional “Adicionar divisorias em uma página”***

YS.REQ.0106	Adicionar divisorias em uma página
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	O utilizador estando dentro de uma página pode adicionar divisórias na mesma, essas divisórias funcionam apenas como um método de separação dos conteúdos e melhor visualização da página atual, essa divisória pode ter sua cor alterada de acordo com as preferências do utilizador.
<b>Motivação</b>	Um utilizador a fim de melhor identificar, orientar e organizar suas informações sempre irá precisar adicionar novas divisórias as suas páginas,, se dá então a necessidade de implementar essas divisórias em qualquer área que convenha o utilizador.

***Tabela 21 – Descrição Requisito funcional “Adicionar um link de redirecionamento em uma página”***

YS.REQ.0107	Adicionar um link de redirecionamento em uma página
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	O utilizador estando dentro de uma página pode adicionar um link que o fará ser redirecionado para outro local no seu espaço de gestão pessoal, esse redirecionamento será realizado para uma página já existente diferente da atual e esta será definida pelo mesmo.



<b>Motivação</b>	Um utilizador pode querer fazer uma menção ou um link de acesso rápido para alguma de outra das suas páginas pessoais já estando dentro de uma página, se dá então a necessidade de implementar um link que possa fazer esse redirecionamento possível.
------------------	---

***Tabela 22 – Descrição Requisito funcional “Adicionar imagens em uma página”***

<b>YS.REQ.0108</b>	<b>Adicionar imagens em uma página</b>
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	O utilizador estando dentro de uma página pode adicionar imagens para a mesma, essas imagens podem vir diretamente de um upload do utilizador, ou podem ser referenciadas através do link da imagem, funcionando para qualquer imagem da web.
<b>Motivação</b>	Uma das formas mais importantes de melhor interpretar uma necessidade ou até mesmo melhor organizar qualquer tipo de informação se dá através de métodos visuais, a implementação de uma funcionalidade de adicionar imagens é essencial para melhor atender as necessidades de todos os utilizadores do YourSpot.

***Tabela 23 – Descrição Requisito funcional “ficheiros em uma página”***

<b>YS.REQ.0109</b>	<b>Adicionar ficheiros em uma página</b>
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	O utilizador estando dentro de uma página pode adicionar algum ficheiro à sua escolha, esse ficheiro pode ser referenciado através de um caminho no seu disco de armazenamento, este será representado por um ícone na página, onde poderá ser feito download do mesmo sempre que requisitado.
<b>Motivação</b>	Uma das formas mais importantes de melhor interpretar uma necessidade ou até mesmo melhor organizar qualquer tipo de informação se dá através de métodos visuais, a implementação de uma funcionalidade de adicionar imagens é essencial para melhor atender as necessidades de todos os utilizadores do YourSpot.



***Tabela 24 – Descrição Requisito funcional “Adicionar vídeos em uma página”***

YS.REQ.0110	Adicionar vídeos em uma página
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	O utilizador estando dentro de uma página pode adicionar vídeos para a mesma, esses vídeos podem ser referenciados através do link da imagem, funcionando para qualquer vídeo da plataforma “youtube”.
<b>Motivação</b>	Uma das formas mais importantes de melhor interpretar uma necessidade ou até mesmo melhor organizar qualquer tipo de informação se dá através de métodos visuais, a implementação de uma funcionalidade de adicionar vídeos essencial para melhor atender as necessidades de todos os utilizadores do YourSpot.

***Tabela 25 – Descrição Requisito funcional “Consultar exercícios”***

YS.REQ.0200	Consultar exercícios
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	Quando o utilizador aceder ao espaço acadêmico este poderá então ser capaz de consultar os diversos exercícios que foram fornecidos por professores e colocados na biblioteca virtual da aplicação.
<b>Motivação</b>	O utilizador estando na área acadêmica da plataforma precisa ser capaz de consultar todos os tipos de exercícios disponibilizados pelo aplicativo, a fim de procurar aqueles que possam o ajudar em sua vida acadêmica, e portanto se dá a necessidade de implementar a funcionalidade de navegação entre esses exercícios do sistema.

***Tabela 26 – Descrição Requisito funcional “Consultar textos”***

YS.REQ.0201	Consultar textos
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	Quando o utilizador aceder ao espaço acadêmico este poderá então ser capaz de consultar os diversos textos que foram fornecidos por professores e colocados na biblioteca virtual da aplicação.
<b>Motivação</b>	O utilizador estando na área acadêmica da plataforma precisa ser capaz de consultar todos os tipos de textos disponibilizados pelo aplicativo, sejam eles artigos científicos, textos



	peçoais e brainstorming ou qualquer outro documento de leitura que possa o ajudar em sua vida acadêmica, e portanto se dá a necessidade de implementar a funcionalidade de navegação entre esses textos do sistema.
--	---

***Tabela 27 – Descrição Requisito funcional “Consultar livros”***

YS.REQ.0202	Consultar livros
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	Quando o utilizador aceder ao espaço académico este poderá então ser capaz de consultar os diversos livros que foram fornecidos por professores e colocados na biblioteca virtual da aplicação.
<b>Motivação</b>	O utilizador estando na área académica da plataforma precisa ser capaz de consultar todos os tipos de livros disponibilizados pelo aplicativo, e deve poder navegar entre os diversos tipos para encontrar o que melhor auxilie em seu processo de conhecimento, e portanto se dá a necessidade de implementar a funcionalidade de navegação entre esses livros do sistema.

***Tabela 28 – Descrição Requisito funcional “Guardar exercícios”***

YS.REQ.0203	Guardar exercícios
<b>Prioridade</b>	<b>Importante</b>
<b>Descrição</b>	Quando no o espaço académico do utilizador este poderá ao aceder os exercícios que foram colocados na biblioteca virtual da aplicação.(YS.REQ.0200) A partir disso tem consigo a opção de guardar o exercício que fora atualmente consultado no seu espaço de armazenamento local.
<b>Motivação</b>	O utilizador pode encontrar um exercício que seja de grande ajuda e desejar um método de fácil acesso ao mesmo, a funcionalidade de guardar exercícios surge como forma de oferecer uma área própria agrupando todos os exercícios salvos por escolha do utilizador.

***Tabela 29 – Descrição Requisito funcional “Guardar textos”***

YS.REQ.0204	Guardar textos
<b>Prioridade</b>	<b>Importante</b>
<b>Descrição</b>	Quando no o espaço académico do utilizador este poderá ao aceder os textos que foram colocados na biblioteca virtual da aplicação.(YS.REQ.0201) A partir disso tem consigo a opção de guardar o texto que fora atualmente consultado no seu espaço de armazenamento local.
<b>Motivação</b>	O utilizador pode encontrar um texto que seja de grande ajuda e desejar um método de fácil acesso ao mesmo, a funcionalidade de guardar textos surge como forma de oferecer uma área própria agrupando todos os textos salvos por escolha do utilizador.

***Tabela 30 – Descrição Requisito funcional “Guardar livros”***

YS.REQ.0205	Guardar livros
<b>Prioridade</b>	<b>Importante</b>
<b>Descrição</b>	Quando no o espaço académico do utilizador este poderá ao aceder os livros que foram colocados na biblioteca virtual da aplicação.(YS.REQ.0202) A partir disso tem consigo a opção de guardar o livro que fora atualmente consultado no seu espaço de armazenamento local.
<b>Motivação</b>	O utilizador pode encontrar um livros que seja de grande ajuda e desejar um método de fácil acesso ao mesmo, a funcionalidade de guardar livros surge como forma de oferecer uma área própria agrupando todos os livros salvos por escolha do utilizador.

***Tabela 31 – Descrição Requisito funcional “Registar exercícios”***

YS.REQ.0206	Registar exercícios
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	Quando no espaço académico do utilizador, caso este tenha acesso de “professor” terá assim a possibilidade de colocar novos exercícios na biblioteca virtual, os mesmos ficarão pendentes a aprovação, logo que aprovados, estes serão armazenados na biblioteca virtual e ficarão disponíveis para serem consultados.



<b>Motivação</b>	Com a necessidade do uso de exercícios como objetos de pesquisa e aprendizado no âmbito acadêmico o professor deve ser capaz de transmitir estes de forma acessível para os estudantes, e o YourSpot oferece isso ao permitir que o professor faça upload de exercícios por meio da plataforma, livros esses que ficarão disponíveis para que os estudantes possam consultá-los.
------------------	--

***Tabela 32 – Descrição Requisito funcional “Registrar textos”***

<b>YS.REQ.0207</b>	<b>Registrar textos</b>
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	Quando no espaço acadêmico do utilizador, caso este tenha acesso de “professor” terá assim a possibilidade de colocar novos textos na biblioteca virtual, os mesmos ficarão pendentes a aprovação, logo que aprovados, estes serão armazenados na biblioteca virtual e ficarão disponíveis para serem consultados.
<b>Motivação</b>	Com a necessidade do uso de textos como objetos de pesquisa e aprendizado no âmbito acadêmico o professor deve ser capaz de transmitir estes de forma acessível para os estudantes, e o YourSpot oferece isso ao permitir que o professor faça upload de textos por meio da plataforma, livros esses que ficarão disponíveis para que os estudantes possam consultá-los.

***Tabela 33 – Descrição Requisito funcional “Registrar livros”***

<b>YS.REQ.0208</b>	<b>Registrar livros</b>
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	Quando no espaço acadêmico do utilizador, caso este tenha acesso de “professor” terá assim a possibilidade de colocar novos livros na biblioteca virtual, os mesmos ficarão pendentes a aprovação, logo que aprovados, estes serão armazenados na biblioteca virtual e ficarão disponíveis para serem consultados.
<b>Motivação</b>	Com a necessidade do uso de livros como objetos de pesquisa e aprendizado no âmbito acadêmico o professor deve ser capaz de transmitir estes de forma acessível para os estudantes, e o YourSpot oferece isso ao permitir que o professor faça upload de livros por meio da plataforma, livros esses que ficarão disponíveis para que os estudantes possam consultá-los.

## 2.4. Requisitos não funcionais

**Tabela 34 – Descrição Requisito não funcional “Iniciar sessão no sistema”**

YS.REQ.0001	Iniciar sessão no sistema
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	<p>Um utilizador tem de efectuar autenticação antes de poder ter acesso a qualquer funcionalidade do sistema. A autenticação deve ser efectuada introduzindo o par “nome de utilizador” seguido de “palavra-chave”. A partir desse momento, o utilizador inicia uma sessão no sistema. O utilizador tem dois tipos de login.</p> <p><b>Aluno:</b> Destina-se ao utilizador primário da plataforma, todos os recém cadastrados são tido como alunos a fim de buscar conhecimento e uma melhor organização pessoal, o aluno por sua vez pode acessar todas as funções da área de gestão pessoal como discutido nos requisitos funcionais e também pode acessar a consulta e armazenamento de qualquer exercício, texto ou livros disponibilizados na biblioteca virtual e no feed de conhecimento da plataforma.</p> <p><b>Professor:</b> Um utilizador que possui todas as funções que um aluno já teria, porém pode contribuir com a plataforma registrando novos exercícios, textos ou livros na área académica.</p>
<b>Motivação</b>	Para realizar qualquer tipo de função, é preciso controlar a acessibilidade em certas funções do sistema e esse requisito surge exatamente para isso.

**Tabela 35 – Descrição Requisito não funcional “Encerrar sessão no sistema”**

YS.REQ.0002	Encerrar sessão no sistema
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	Um utilizador deve poder encerrar a sua sessão no sistema. A partir desse momento, deixa de ter acesso às funcionalidades e regressa ao estado em que estava antes de efectuar o início de sessão.
<b>Motivação</b>	Para realizar qualquer tipo de função, é preciso controlar a acessibilidade em certas funções do sistema e esse requisito surge exatamente para isso.

***Tabela 36 – Descrição Requisito não funcional “Interoperabilidade”***

YS.REQ.0003	Interoperabilidade
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	A interoperabilidade deste será a capacidade do mesmo de ser executado em qualquer sistema operativo móvel, permitindo assim a liberdade do utilizador em utilizar o sistema operativo móvel que mais lhe agrada e ainda assim ser capaz de fazer uso do YourSpot.
<b>Motivação</b>	Facilidade na utilização por qualquer dispositivo móvel, de modo a não inviabilizar sua execução.

***Tabela 37 – Descrição Requisito não funcional “Componentes”***

YS.REQ.0004	Componentes
<b>Prioridade</b>	<b>Essencial</b>
<b>Descrição</b>	Para o servidor é necessário a utilização de um sistema de base de dados não SQL, no projeto utilizamos da FireBase, desenvolvida pela própria Google no auxílio de criar aplicações moveis de forma a armazenar informações e para a plataforma espera-se uso de kotlin para sua construção.
<b>Motivação</b>	Para se tornar possível a construção da aplicação e esta deter de um armazenamento próprio para dados e informação que esteja em comunicação com a internet.

## 3. Software Design

---

### 3.1. Diagrama de Classes

O diagrama de classes do projeto se encontra na primeira página [deste link](#).

O diagrama atual refere-se a uma representação visual de como as classes irão se relacionar umas com as outras baseadas na estrutura da aplicação “YourSpot”. São representados neste diagrama as classes que servirão como modelo para os objetos no desenvolvimento da aplicação, tal como seus respectivos atributos e métodos incluindo a tipagem dos mesmos e seu encapsulamento no sistema.

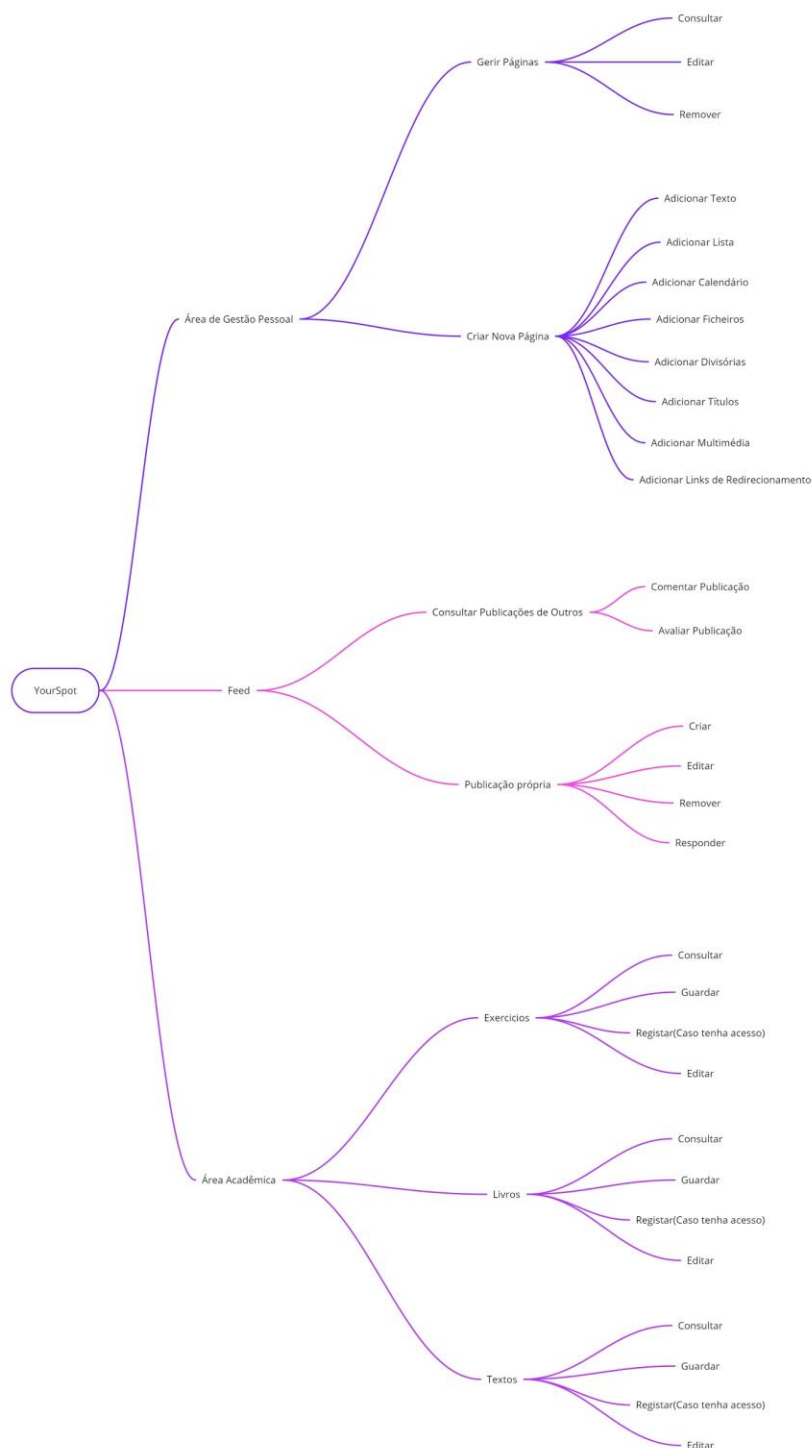
### 3.2. Diagrama de Componentes

O diagrama de componentes do projeto se encontra na segunda página [deste link](#).

Procurando a melhor definição e visualização do nosso sistema, foi também sugerido a criação de um diagrama de componentes. Tal diagrama tem papel fundamental neste documento por demonstrar como a arquitetura do sistema dividida em componentes relacionam-se entre si, esses componentes são unidades do nosso sistema que providenciam uma interface que pode posteriormente ser utilizada por outro componente, assim podemos saber como cada unidade do sistema interage com outra e temos como resultado final uma ótima demonstração do projeto YourSpot conseguindo desempenhar sua função se beneficiando das suas interfaces. É importante ressaltar que quase todos os componentes de UI estão diretamente ligado aos componentes responsáveis por armazenar informações e desempenhar funções relacionada a sua área específica, e por sua vez esses componentes requerem a interface do nosso sistema de base de dados, sistema esse que seguindo as boas práticas do desenvolvimento em kotlin é dividido em 4 partes, o “Your Spot ViewModel” que será responsável por guardar a viewmodel de todo o projeto e servir como ponte para a ligação de um fragment para a base de dados, se comunica com o repository onde são guardada todas as nossas funções que por sua vez se comunica com o nosso Dao, ou, Data access control que possui o nome sugestivo de sua função, se comunicando ultimamente com a base de dados e efetivamente executando nossa query.



### 3.3. Mind Map



**Figura 3 - Mind Map do sistema YourSpot**

De forma a perceber melhor como será o funcionamento e abstraindo-se de âmbitos de processo, foi realizado um “mind map” com sua estrutura dada em árvore onde é possível perceber as principais funções da aplicação e a divisão estrutural que será vista na aplicação “YourSpot”.



### 3.4 Modelo Entidade-associação YourSpot

O modelo de entidade-associação do projeto se encontra na segunda página [deste link](#).

#### Contextualização

Legenda: <Nome de uma associação>; [Nome de uma entidade].

A figura demonstra uma entidade principal, [utilizador] como previsto antes da criação do modelo entidade-associação, nosso modelo de base de dados gira em torno de um utilizador, deste utilizador podemos dividi-lo em duas generalizações [Aluno] e [Professor], utilizadores efetivos do YourSpot e portanto pretende-se o armazenamento de informações acerca dos mesmos, os atributos ligados a cada uma das entidades nos mostra o que pretendemos armazenar dentro de nosso sistema de base de dados, esses dados posteriormente serão utilizados para exibição dos dados e funcionamento correto da aplicação no que diz o controle de acesso, o [utilizador] está efetivamente associado a todas as outras entidades do sistema, temos a área acadêmica, com 3 entidades, sendo elas [Texto], [Exercicio] e [Livro] com todos seus atributos respectivos e a área de gestão pessoal sendo constituída apenas de uma entidade [Pagina Pessoal].

As associações com destino a área acadêmica sempre passam por duas formas associativas, <publicar> ou <pesquisar>, essas associações surgem para podermos implementar um mecanismo de histórico e favoritos do exercício enquanto ainda permitem o maior armazenamento de informação necessário no momento de publicar uma informação na biblioteca virtual do YourSpot.

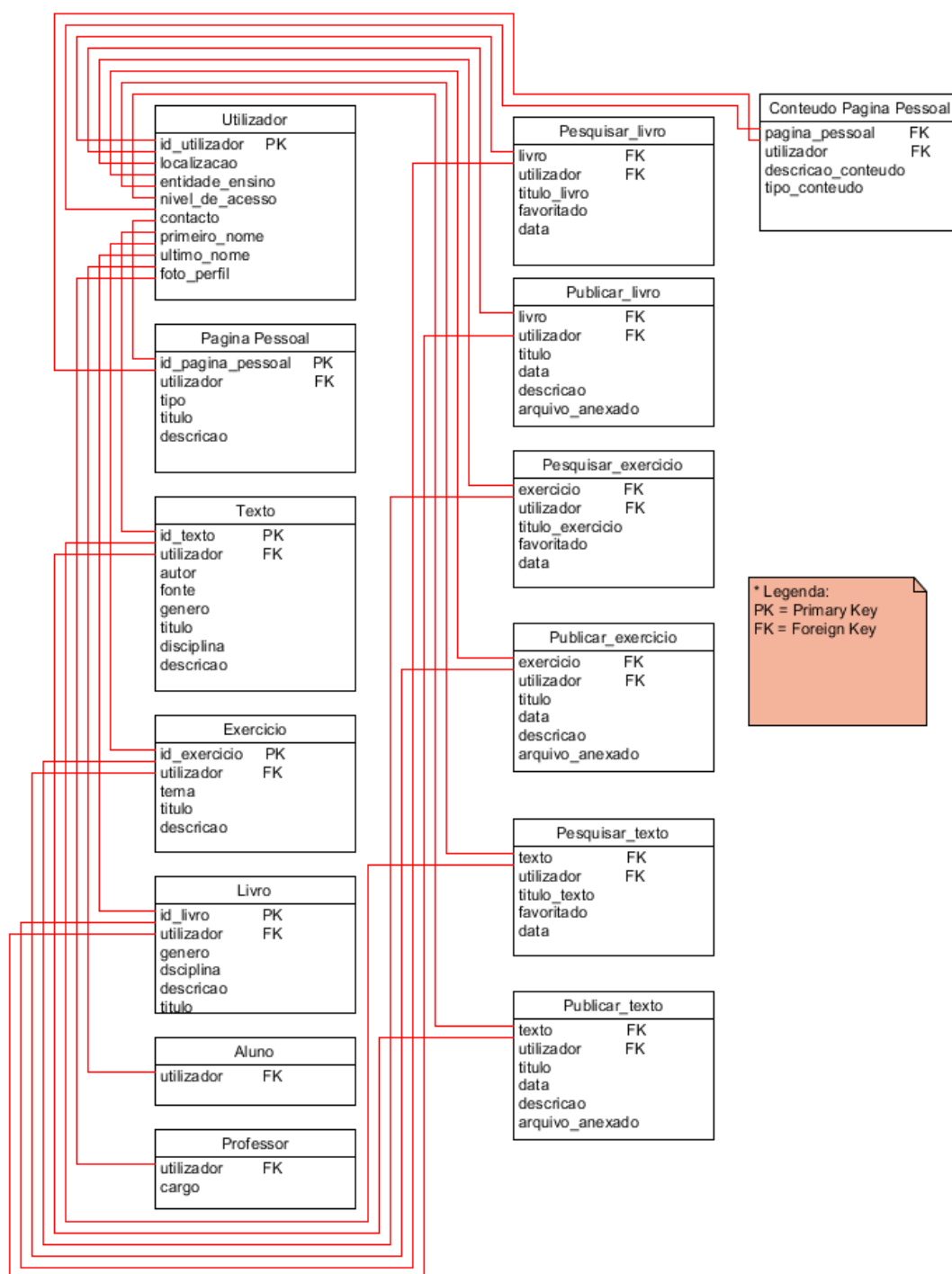
Enquanto a associação com destino a área de gestão pessoal, buscar armazenar a informação sobre o que o [utilizador] guarda em sua [pagina pessoal], anotações, tabelas, calendários ou qualquer <Conteúdo Pagina Pessoal> conveniente ao mesmo.

Também são apresentadas as cardinalidades de cada uma das associações, fazendo referência a qual o grau de relação entre as duas entidades (tabelas) associadas, lê-se como:

- 1:n → [Professor] pode <Publicar> **n (muitos)** [Livro(s)] , enquanto [Livro] só pode ser <Publica(**do**)> por apenas **1** [Professor]
- n:m → [Utilizador] pode <Pesquisar> **n (muitos)** [Livro(s)] , enquanto [Livro] pode ser <Pesquisa(**do**)> por **n (muitos)** [Utilizador(es)]



### 3.4.1 Modelo Relacional YourSpot



**Figura 4 - Modelo Relacional YourSpot**

Depois de levantarmos o modelo entidade associação, é criado o modelo relacional, que tem como objetivo demonstrar de uma forma mais natural todas as tabelas geradas pelo modelo entidade associação, como todas as associações possuíam atributos próprios, as mesmas geraram tabelas e também são referenciadas nesse modelo, demonstrando os seus atributos e relacionamentos entre cada tabela e demonstrando como as chaves (valores da coluna PK) são referenciadas (FK)



### 3.5 User Interface Design

A considerar que o Yourspot será acedido pelas mais variadas pessoas, com diferentes ideias e pensamentos, é parte importante para o nosso objetivo de desenvolvimento construir uma plataforma que seja acessível a todos, e parte deste caminho se refere a termos interfaces que sejam de fácil utilização e que passem de certa forma, uma experiência de uso que seja agradável para o utilizador final.

Além disso, como nossa aplicação detém dois tipos de públicos, que são os alunos e os professores, é de enorme importância que ambos possam trabalhar e utilizar da plataforma da melhor maneira possível.

A pensar no desafio de conseguir passar tal experiência para todo o tipo de pessoas foram desenvolvidas duas personas, personas essas que tinham como objetivo simular duas pessoas distintas, uma de cada nicho do nosso público alvo, ou seja, um aluno e um professor.

Cada uma dessas personas tem suas devidas informações, o tipo de pessoa que é, seus objetivos, sua história, suas motivações e suas frustrações. E é baseada nessas características que vamos moldar a sua utilização na aplicação.

Baseado no que foi dito acima, as duas personas desenvolvidas para esse efeito são:

- **Caio Manuel da Silva:** Um aluno de 21 anos, do tipo “Idealista”.
- **Luiza Pereira d’Albuquerque:** Uma professora de 49 anos, do tipo “Racional”

Para fins explicativos referentes aos tipos de personas:


- **Idealista** - *Desejam situações e esperam que a mesma seja conveniente para os demais, querem agir bem e ter a consciência limpa, sempre tentam alcançar seus objetivos sem comprometer o código de ética dos demais.*
- **Racional** - *Se preocupa com quais problemas podem afetá-lo e pensa em como solucioná-los, age com maior eficiência possível para conseguir seus objetivos, algumas vezes ignorando arbitrariamente as regras e condições impostas sobre o mesmo.*



**YOURSPOT**

PROJECT: untitled PERSONA: Caio Manuel da Silva

NAME	MARKET SIZE	TYPE
Caio Manuel da Silva	60 %	Idealist



**Goals**

- Se organizar melhor;
- Agrupar os diversos tipos de informações;
- Lembrar de suas responsabilidades escolares e pessoais;
- Sanar suas dificuldades acadêmicas;
- Se conectar com outras pessoas em âmbito acadêmico.

**Background**

Caio é um estudante e se encontra no meio de sua licenciatura, após um tempo de experiência notou certa dificuldade em organizar-se perante tamanha quantidade de informações das suas muitas unidades curriculares onde encontra-se inscrito. Devido a isso, Caio decidiu procurar maneiras de tornar-se mais produtivo.

**Demographic**

Male 21 years

Brasil

Single

Estudante




**Motivations**

- Tornar-se mais produtivo;
- Melhorar seu desempenho acadêmico;
- Aproveitar melhor seu tempo;
- Compartilhar possíveis soluções de exercícios.

**Frustrations**

- Não conseguir organizar-se de maneira efetiva;
- Aplicação não irencontro dos seus objetivos;
- Usabilidade fraca;

**Technology**


- 
- 
- 

***Figura 5 - Persona 1 YourSpot***

Um link com a imagem ampliada [se encontra aqui](#)

PROJECT: untitled PERSONA: Luiza Pereira d'Albuquerque

NAME	MARKET SIZE	TYPE
Luiza Pereira d'Albuquerque	40 %	Rational



**Goals**

- Manter-se organizada na sua profissão;
- Melhorar seus planos de aulas;
- Lecionar com maior dinamismo;
- Se conectar com seus alunos;
- Demonstrar seu trabalho;
- Promover um fonte de conhecimento adicional para seus alunos.

**Background**

Luiza é professora do 1o ciclo de Engenharia Aeroespacial, leciona a mais de 15 anos, e sempre procurou formas de manter suas aulas sempre dinâmicas e passar o melhor conhecimento possível para seus alunos, com o passar dos tempos e o avançar da tecnologia, Luiza que sempre preferiu uma forma mais tradicional de ensino, resolveu dar uma chance e modernizar seus métodos de transmitir conhecimento.

**Demographic**

Female 49 years

Brasil

Married

Professora



**Motivations**

- Modernizar seus conceitos de aula;
- Incentivar a busca de aprendizagem dos seus alunos;
- Ter um ambiente extra de aulas mais interativo.

**Frustrations**

- Não conseguir adaptar-se a aplicação;
- Não obter o feedback pretendido;
- Sentir-se limitada pela aplicação;
- Não obter retorno favorável dos seus alunos.

**Technology**

- 
- 

***Figura 6 - Persona 2 YourSpot***

Um link com a imagem ampliada [se encontra aqui](#)

***Maio de 2021***



Após a confecção das personas, foram atribuídos a estas “User Scenarios”, que se referem aos passos do utilizador desde antes mesmo do uso da aplicação. Os User Scenarios vão retratar como e por que o utilizador fez uso daquela aplicação, baseada em uma “User Story Release”.

**User Persona:** Nome da pessoa

**User Activities:** Ação que foi realizada dentro da aplicação a fim de seguir o que foi colocado no "User steps".

**User Steps:** Passos realizados com base na "Storie Release" da persona de modo a solucionar seu possível "problema".

**Storie Release:** Contexto sobre o qual a persona se encontra para realizar determinada ação.



**Figura 7 - User Scenario Persona 1**

Um link com a imagem ampliada [se encontra aqui](#)

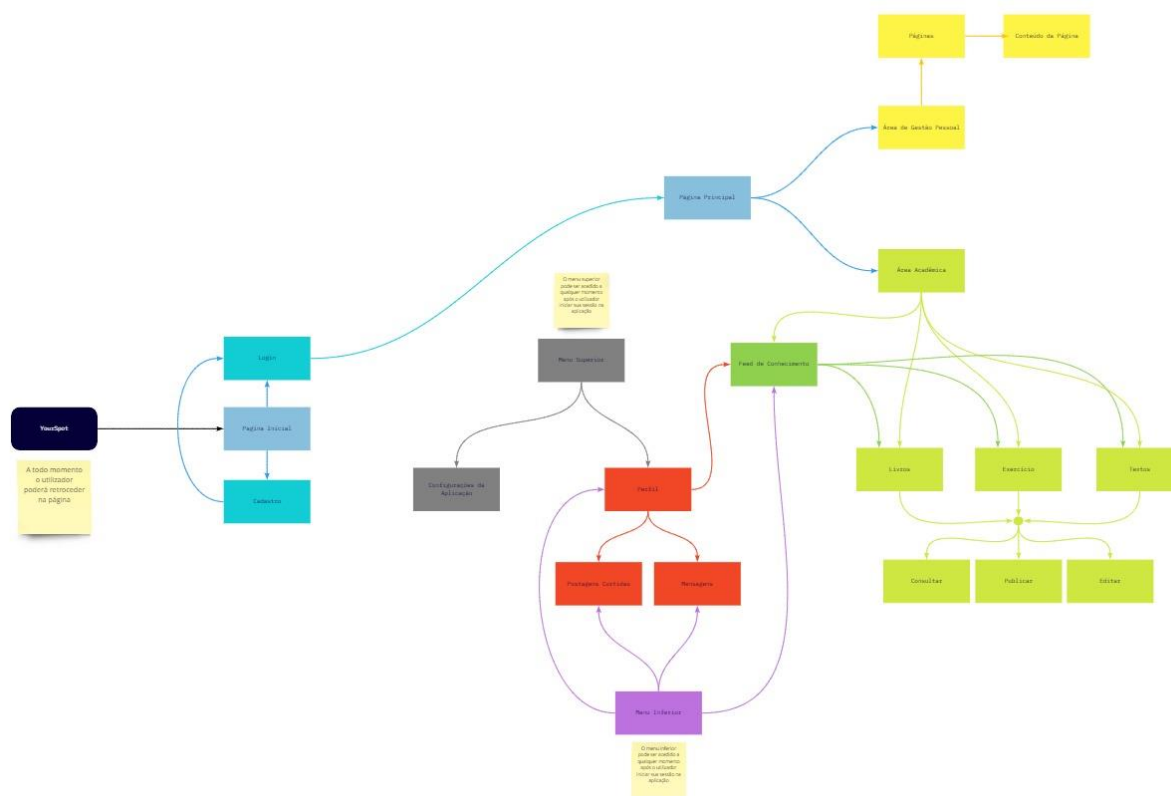


***Figura 8 - User Scenario Persona 2***

Um link com a imagem ampliada [se encontra aqui](#)



Após todas as construções de utilização das personas citadas, foi elaborado um Diagrama de fluxo de processos para que se pudesse perceber melhor o funcionamento dentro da plataforma, esse diagrama demonstra visualmente como seria o passo a passo do utilizador na aplicação.



***Figura 9 - Diagrama de Fluxo de Processos YourSpot***

Um link com a imagem ampliada [se encontra aqui](#)

Com base nas informações recolhidas das personas e os cenários das suas possíveis utilizações, podemos elaborar interfaces que se adaptam melhor a cada tipo de público, assim como deixar a experiência de utilizar a aplicação mais satisfatória no geral.



### 3.6. Padrão de Arquitetura

O padrão de arquitetura surge como uma solução reutilizável para algum tipo de problema dentro da arquitetura de software promovendo maior disponibilidade e segurança para o sistema.

Dentro do YourSpot utilizamos de um padrão de arquitetura denominado “MVVM”, ou Model-View-ViewModel que tem como maior objetivo a minimização da complexidade no processo de separação dos componentes que a parte interessada interage com o sistema. Esse padrão é o responsável em converter diversos tipos de dados de um modelo de uma forma que esses mesmos dados podem ser facilmente apresentados e manuseados.

## 4. Software Construction

---

### 4.1. - Descrição do Sistema Desenvolvido

O YourSpot sempre foi idealizado tendo por base a [tendência de número de pessoas com acesso ao telemóvel](#), mais da metade da população se conecta através dessa plataforma e tendo em conta essa informação, o sistema do aplicativo foi desenvolvido com a utilização de dois grandes componentes, são eles:

#### **IDE, Android Studio**

Android Studio é a IDE oficial da Google e foi construída especificamente para a construção de aplicativos móveis, atualmente se beneficia de linguagem em Java, ou uma adaptação mais recente de java denominada Kotlin que será a linguagem utilizada em YourSpot.

Essa IDE garante ao aplicativo ferramentas de muita importância ao decorrer do desenvolvimento do aplicativo e também garante um dinamismo correto com as várias funcionalidades de um dispositivo móvel já que o mesmo possui um sistema de emulação para fins de teste.



## Linguagem, Kotlin

Em 2019 Kotlin passou a ser a linguagem preferencial da Google no desenvolvimento de aplicativos móveis, sendo suportada em múltiplas plataformas computacionais. Kotlin foi desenvolvido para interpretar por completo a linguagem Java. Tendo a vantagem de possuir Funcionalidades e sintaxes mais concisas.

Toda a criação do sistema YourSpot se baseia nos princípios fundamentais da construção de software criado pela cooperação de diversos profissionais e membros da indústria, denominado [SWEBOK](#).

Tendo isso em conta, no Âmbitos do desenvolvimento desse aplicativo buscamos incorporar os seguintes tópicos:

- **Minimizar a complexidade:**

A necessidade de reduzir a complexidade na hora da construção do software se aplica a quase todas as partes do projeto, promovendo uma clara compreensão tanto na documentação quanto a nível de programação, através da criação de código concisa com apoio de padrões que facilitam até mesmo outros pontos como a reutilização do código.

- **Antecipação de Alterações:**

Todo projeto passa por alterações, com o objetivo da criação de um software duradouro, antecipar alterações promove diretamente a melhoria contínua do aplicativo e é garantida no nosso sistema pelas técnicas utilizadas na hora da criação efetiva do sistema.

- **Reutilização:**

A reutilização se refere ao uso inteligente de componentes operacionais para resolver diversos problemas, esses componentes operacionais podem ser trechos de códigos reutilizáveis para promover um ambiente de antecipação de alteração ( Análise de risco ) mais eficiente assim como também criar um padrão para o desenvolvimento da aplicação promovendo uma redução de complexidade geral dado que a parte interessada só precisa estudar um certo componente para entender várias utilizações do mesmo.

## 4.2. - Abordagem a nível de programação

### **Yourspot Fragments Repository**

Dentro do nosso projeto nós temos um Repositório, este trata-se de uma classe estática que não precisa ser instanciada, que foi criada especificamente para que pudéssemos armazenar as operações comuns feitas pela nossa aplicação, sendo assim, podemos manter o código mais organizado e legível na ótica de quem tentar percebê-lo. E além deste fato evita-se a repetição de código e torna mais fácil a utilização de funções.

Basicamente nosso repositório concentra todas as principais funções com maior uso, como já foi mencionado acima, estas funções estão divididas nas subclasses criadas dentro da classe principal. Estas subclasses se dividem em categorizações de funções dentro da aplicação, onde assim pode-se perceber facilmente qual o tipo de função e do que ela se trata. Consoante a ação desejada, essas funções podem ser preenchidas com parâmetros fornecidos na sua chamada.

Sua utilização dentro da aplicação consiste em fazer uma chamada a função citando o repositório e sua subclasse respectiva, quando necessário também são fornecidos parâmetros. Quando terminada a função é executada a ação esperada da mesma.

### **Navegação do yoursport**

A navegação do aplicativo diz respeito a todo o fluxo do utilizador quando acessa cada uma das diferentes "páginas", a seguir segue a demonstração de como seria tal navegação.

### **Yourspot ViewModel**

A designação “ViewModel” se dá a uma classe projetada para armazenar e gerenciar dados relacionados à unidade de interface.

É essa a classe que garante que os dados sempre continuem atualizados frente a qualquer mudança imposta ao aplicativo, como por um exemplo uma rotação de tela do telemóvel ou alguma atualização feita na base de dados.

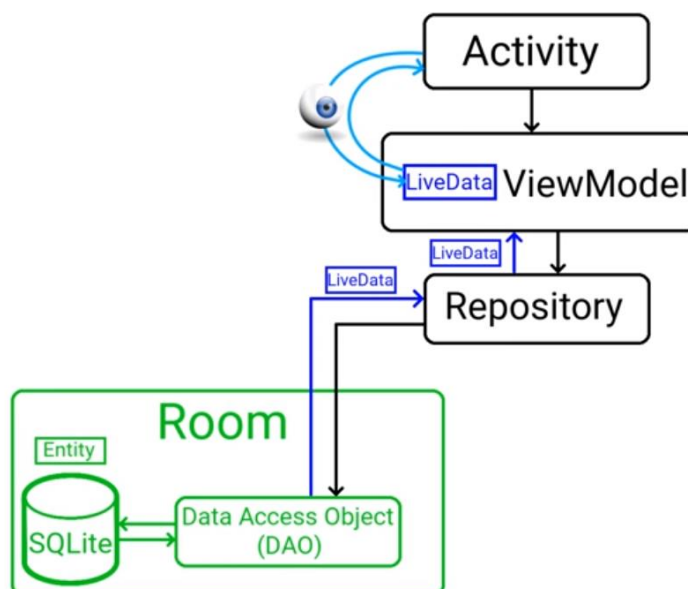
Em resumo uma classe ViewModel tem como principal objetivo exibir dados da interface e reagir às ações do usuário enquanto se comunica com o sistema operacional, pensar que uma classe já responsável por tudo isso também lidaria com todas as chamadas a base de dados, tornaria uma classe completamente pesada em questão de alocamento de processamento, nesse âmbito surge o Repository do YourSpot, responsável por fazer a chamada ao DAO (Data Access Object) que em última instância irá fazer a chamada para a base de dados.

Garantindo um ambiente mais fácil e eficiente separando a propriedade de base de dados e a propriedade de exibição dos dados através do controle da unidade de interface

## Yourspot DataBase Repository

Assim como o fragments repository cuida de todas as operações comuns feitas pela nossa aplicação, o repository em âmbito DataBase cuida de todas as solicitações comuns feitas a nossa base de dados, todas elas com seus devidos parâmetros que tornam uma função em algo dinâmico que pode ser então reutilizado inúmeras vezes ao decorrer do desenvolvimento de toda a aplicação, dividir nosso sistema em várias classes com um nível de importância tão grande como esse, é o que garante um código com minimização de complexidade, promovendo um código inteligente e de fácil percepção quando estudado. Esse Repository é o responsável por fazer a chamada ao Data Access Object do nosso aplicativo que como o nome já sugere é essa classe que fica responsável pela efetiva chamada da Base de Dados e concluir um pedido de solicitação de um usuário a mesma.

Já foi demonstrado como o aplicativo se comunica com as suas páginas se utilizando desse sistema explicado acima, para focarmos apenas nesse tipo específico de comunicação, é demonstrado a seguinte imagem que explica o processo feito por essas classes importantes do nosso aplicativo .



SUBS  
CRIBE

***Figura 10 - Processo de Comunicação Simplificado YourSpot***

## 5. Implementação

---

### 5.1. Fases de Desenvolvimento

- Fase Conceitual

Nosso projeto deu-se início primeiramente com uma fase conceitual, onde tínhamos como objetivo primário pensar como as coisas iriam funcionar. A partir de uma ideia inicial sobre qual tema seria o abordado, tivemos de pensar não só no problema como também na solução que nosso projeto traria consigo e juntamente disso a forma como essa solução seria dada através dele.

- Fase de Seleção

Percebida a ideia inicial, já tendo em mente a identificação do problema e uma possível solução para o mesmo, passamos então para uma fase de seleção, onde nesta, como o próprio nome já sugere, fizemos uma seleção de quais ferramentas, plataformas, linguagens de programação, sistemas de gerenciamento de base de dados, entre outros fatores necessários na construção da nossa solução. Para tal, fizemos um apanhado do que nos seria possível utilizar, se era viável nesse cenário e se éramos capazes de implementar nossa solução de maneira integral fazendo uso do que viria a ser selecionado.

Para esta fase também foram realizadas pesquisas sobre qual seria a melhor plataforma a ser utilizada de modo a visar o público que tínhamos em mente, e se isso se encaixaria dentro do cenário de utilização.

Dessa forma foram escolhidas:

- **Android Studio**
- **Kotlin**
- **Firebase**

Com o que fora selecionado, conseguiríamos desenvolver uma aplicação móvel para o sistema Android de forma eficiente e responsiva, juntamente com o suporte de armazenamento de dados fornecidos pela Firebase.

- Fase de Documentação Inicial

Após ambas as fases anteriores iniciamos um processo de documentação do nosso desenvolvimento, esse processo deu-se início com a formulação de uma descrição da forma a qual víamos esse projeto de software, elaborando assim qual seria seu propósito e seus objetivos a serem cumpridos.



Após isto foi elaborado uma calendarização consoante ao tempo que nos foi disponibilizado, com as tarefas a serem realizadas e quem iria realiza-las. Essa calendarização foi dividida em duas etapas, onde na primeira etapa seria elaborado toda a parte teórica deste projeto e na segunda foi designada para implementação e desenvolvimento prático do mesmo.

Ainda na fase de documentação inicial elaboramos algumas “mockups” que servirão como um rascunho do modo como nossa aplicação será vista pelos utilizadores e demonstrará uma possível disposição de menus e itens do mesmo, além da forma de navegação que será aplicada.

- **Fase de Levantamento de Requisitos**

Definidos os objetivos da nossa solução, passamos a fase do levantamento de requisitos funcionais e não funcionais, onde estes são especificados com uma descrição e um grau de prioridade. Estes são baseados nas funcionalidades que nossa aplicação terá, com isso ao levantarmos estes requisitos tivemos sempre que manter o foco em cumprir os objetivos que antes foram decididos.

- **Fase de Construção dos casos de uso**

Nesta fase elaboramos os nossos casos de uso que já forma citados neste documento, os mesmos foram desenvolvidos com base nos requisitos levantados na fase anterior, onde expressam o fluxo de utilização dentro da nossa aplicação.

- **Fase de Elaboração do Software Design**

Já tendo como base os requisitos que foram levantados, o processo de utilização teórico que teríamos dentro da nossa aplicação, começamos a pensar nos quesitos de software design, como nossa aplicação funcionaria em cenário prático, como os componentes iriam interagir entre si, quais seriam nossas entidades e como elas iriam se relacionar umas com as outras. Dessa forma essa fase foi uma fase de construção do nosso Software Design, para isso foram elaborados os Diagramas de Classe, Diagramas de componentes, Mapas conceituais, Modelo Entidade-Associação e o Modelo Relacional.

Foi também elaborado o “User Interface Design” que se trata da interface de utilização, para tal, foram confeccionadas personas, cenários de uso e diagramas de fluxos de processos, visando entender melhor a interação entre os utilizadores e nossa aplicação.

- **Fase de Descrição do “Software Construction”**

A última fase da parte de documentação foi a descrição do nosso “Software Construction” onde seria descrito tudo que fora utilizado da nossa parte e com qual fundamento, é também elaborada uma descrição da arquitetura de funcionamento e a forma a qual o sistema foi desenvolvido.

- Fase de Implementação e Desenvolvimento da Aplicação

Dando início a fase prática temos esta fase, a qual temos como objetivo implementar e desenvolver nossa aplicação, como o próprio nome sugere, fundamentado em tudo aquilo que fora documentado nas fases teóricas.

- Fase de Ligação a Base de dados

Nesta fase realizamos a ligação do nosso sistema de gerenciamento de base de dados a nossa aplicação e assim torna-se possível o uso e armazenamento de dados.

- Fase de prototipagem

Chegando ao fim do desenvolvimento passamos as fases de prototipagem, onde começam a surgir as primeiras versões da aplicação que está a ser desenvolvida, dessa forma nos é possível observar a sua utilização em um cenário de uso real e avaliar seu comportamento.

- Fase de testes

Fase a qual nossa aplicação é submetida a diversos testes no cenário de utilização e são também realizados testes de desempenho, funcionais e não funcionais.

## 5.2. Ferramentas e bibliotecas de software

Entre as ferramentas e bibliotecas utilizadas no projeto do Yourspot, nós temos:

- Miro
- Figma
- Android Studio
- Kotlin
- Firebase
- Biblioteca AndroidX
- Bibliotecas de Suporte v7

### Miro



O Miro é uma plataforma online que vai ajudar você na construção de mapas mentais, diagramas e quadros com notas. Tudo em tempo real e em colaboração com quem você quiser.



Toda a parte de mapas conceituais da nossa aplicação foi elaborada por meio desta ferramenta.

### **Figma**



Figma é um editor gráfico de vetor e prototipagem de projetos de design baseado principalmente no navegador web, com ferramentas offline adicionais para aplicações desktop para GNU/Linux, macOS e Windows. O aplicativo Figma Mirror é um sistema de prototipagem que espelha o que está sendo feito no computador para o smartphone Android e/ou iOS, permitindo a simulação do vetor criado no computador como um aplicativo ou página da web. O Figma é um software focado no desenvolvimento de sistemas de design gráfico, prototipagem de interface gráfica de usuário e desenvolvimento de UI/UX (user interface experience ou experiência da interface com o usuário), permitindo também o desenvolvimento colaborativo em tempo real com outros usuários remotamente.

No Figma foi possível desenvolvermos nossas mockups iniciais, protótipos e os designs que foram utilizados na nossa aplicação.

### **Android Studio**



O Android Studio é o ambiente de desenvolvimento integrado (IDE, na sigla em inglês) oficial para o desenvolvimento de apps Android e é baseado no IntelliJ IDEA . Além do editor de código e das ferramentas de desenvolvedor avançadas do IntelliJ, o Android Studio oferece ainda mais recursos para aumentar sua produtividade na criação de apps Android, como:

- Um sistema de compilação flexível baseado em Gradle
- Um emulador rápido com inúmeros recursos
- Um ambiente unificado que possibilita o desenvolvimento para todos os dispositivos Android
- A aplicação de mudanças para enviar mudanças de código e recursos ao aplicativo em execução sem reiniciar o app



- Modelos de código e integração com GitHub para ajudar a criar recursos comuns de apps e importar exemplos de código
- Frameworks e ferramentas de teste cheios de possibilidades
- Ferramentas de lint para detectar problemas de desempenho, usabilidade, compatibilidade com versões, entre outros
- Compatibilidade com C++ e NDK
- Compatibilidade integrada com o Google Cloud Platform, facilitando a integração do Google Cloud Messaging e do App Engine.



Kotlin é uma Linguagem de programação multiplataforma, orientada a objetos e funcional, concisa e estaticamente tipada (variáveis com tipos específicos), desenvolvida pela JetBrains em 2011,[4] que compila para a Máquina virtual Java e que também pode ser traduzida para a linguagem JavaScript e compilada para código nativo (via LLVM).

Toda nossa aplicação faz uso da linguagem Kotlin e da sua aplicação dentro da IDE do Android Studio.

### **Firestore**



Se pudermos rotular o Firestore, diremos que ele é Backend-as-a-Service (Baas).

Neste sentido de servidor, o Firestore fornece aos desenvolvedores uma variedade de ferramentas e serviços para ajudá-los a desenvolver aplicativos de qualidade, aumentar sua base de usuários e ser mais lucrativo.

Toda sua base é construída na infraestrutura do Google, sendo categorizado como um programa de banco de dados NoSQL, que armazena dados em documentos do tipo JSON.

O Firestore conta um grande conjunto de ferramentas de desenvolvimento. Destas, o Realtime Database e o Cloud Firestore podem armazenar dados estruturados em documentos e sincronizar os aplicativos correspondentes em milissegundos sempre que ocorre uma transformação de dados.

### **Biblioteca AndroidX**





O AndroidX é uma grande melhoria para a Biblioteca de Suporte original do Android, que não é mais mantida. Os pacotes do androidx substituem totalmente a Biblioteca de Suporte, fornecendo paridade de recursos e novas bibliotecas.

Além disso, o AndroidX inclui os seguintes recursos:

- Todos os pacotes do AndroidX estão disponíveis em um namespace consistente, começando com a string `androidx`. Os pacotes da Biblioteca de Suporte foram mapeados nos pacotes `androidx.*` correspondentes. Para mapear totalmente todas as classes e artefatos de compilação antigos para os novos, consulte a página de Refatoração de pacotes.
- Ao contrário da Biblioteca de Suporte, os pacotes do androidx são mantidos e atualizados separadamente. Os pacotes do androidx usam o controle de versões semântico restrito a partir da versão 1.0.0. Você pode atualizar as bibliotecas do AndroidX no seu projeto de forma independente.
- A versão 28.0.0 é a versão mais nova da Biblioteca de Suporte. Não haverá mais versões da biblioteca `android.support`. Todo o desenvolvimento de novos recursos estará no namespace `androidx`.

### **Biblioteca de Suporte v7**

Estas bibliotecas oferecem conjuntos de recursos específicos e podem ser incluídas no aplicativo de forma independente umas das outras.

Foi bastante utilizada para implementação da barra de ação padrão da interface do utilizador fazendo uso da classe “ActionBar”.

## **6. Avaliação da Solução**

---

### **6.1. Ambiente e Metodologia de Teste**

De modo que possamos avaliar da melhor forma a nossa aplicação que foi desenvolvida, seguiremos então um padrão de testes, nele realizaremos alguns testes funcionais e não



funcionais respectivamente, sendo assim conseguiremos realizar o processo de validação e medição da qualidade do produto desenvolvido.

## 6.2. Testes funcionais

Para os testes funcionais, realizamos tarefas de modo a se ter uma entrada por parte do utilizador e uma saída suposta, dessa forma são testados se estas realmente estão a gerar as saídas previamente definidas ou se obtemos um resultado diferente disso.

***Tabela 38 - Testes Funcionais YourSpot***

MÓDULO	TAREFA	DESCRIÇÃO	ENTRADA	SAÍDA ESPERADA	RESULTADO EFETIVO
Login	Autenticar-se com dados corretos	Utilizador, preenche 2 campos (e-mail, senha) e os dados preenchidos correspondem aos armazenados na base de dados	Dados de autenticação presentes na base de dados	Atribuição de ID do utilizador para a navegação dentro do aplicativo	Correta atribuição de ID do utilizador para a navegação dentro do aplicativo
	Autenticar-se com dados incorretos	Utilizador, preenche 2 campos (email, senha) e os dados preenchidos não correspondem a nenhum dos dados armazenados na base de dados	Dados de autenticação inexistentes	Mensagem de erro, indicando qual dos campos não foi devidamente preenchido e a navegação permanece na área de login	Correta mensagem de erro, indicando qual dos campos não foi devidamente preenchido e a navegação permanece corretamente na área de login
	Autenticar-se como aluno	Utilizador, através de seu ID concedido pela autenticação tem suas permissões geridas para funcionalidades destinadas apenas a "Alunos"	Dados de autenticação de um aluno	Gestão (Exibição ou omissão) de todos os elementos associados a permissão "Aluno"	Correta gestão de todos os elementos associados a permissão "Aluno"
	Autenticar-se como professor	Utilizador, através de seu ID concedido pela autenticação tem suas permissões geridas para funcionalidades destinadas apenas a "Professor"	Dados de autenticação de um professor	Gestão (Exibição ou omissão) de todos os elementos associados a permissão "Professor"	Correta gestão de todos os elementos associados a permissão "Professor"



	Registrar uma nova conta de utilizador	Utilizador, preenche 4 campos (primeiro nome, ultimo nome, email, senha) os dados preenchidos são então armazenados na base de dados para autentica-lo posteriormente	Preenchimento de formulário e clique no botão de criar conta	Armazenamento de todos os campos preenchidos e vinculação (atribuição) de um ID único para o utilizador recém registrado	Correto Armazenamento de todos os campos preenchidos e vinculação (atribuição) de um ID único para o utilizador em questão
Menu lateral	Navegar sem autenticação	Navegar pelo menu lateral sem qualquer autenticação	Clique em um botão de navegação	Ao selecionar o botão de navegação nada ocorre	Ocorre o esperado, pois um utilizador não autenticado não deve conseguir navegar antes de autenticar-se  Ainda é necessária a implementação de uma mensagem de alerta
	Navegar com autenticação	Navegar pelo menu lateral estando autenticado previamente	Clique em um botão de navegação	Ao selecionar o botão de navegação é direcionado a página selecionada	Ocorre o esperado pois consegue navegar para a página pretendida
Gestão Pessoal	Criar uma nova página em branco	Utilizador dentro de sua área pessoal prossegue na criação de uma nova página e escolhe a opção de uma página em branco para ser posteriormente editada.	Clique no botão “+”	Criação de uma nova página em branco, vinculada ao ID único do utilizador e com possibilidade de expansão com novos dados a serem introduzidos na base de dados	Ainda não implementando
	Editar uma página em branco	Utilizador dentro de uma página anteriormente criada pode edita-la, podendo adicionar uma tabela ou uma lista ou um calendário	Clique no botão de edição	Dados preenchidos são substituídos ou adicionados na base de dados mediante as edições escolhidas pelo utilizador.	Ainda não implementando
	Remover uma página em branco	Utilizador dentro de uma página criada pode decidir por exclui-la, removendo todos	Clique no botão de remoção	Dados vinculados a página em questão e consequentemente vinculados ao ID único do utilizador	Ainda não implementando



		os dados vinculados a mesma armazenados na base de dados		são removidos da base de dados.	
Área Acadêmica	Publicar um Livro	Professor preenche 4 campos obrigatórios (Título, gênero, disciplina, descrição) e pode preencher também 2 campos opcionais (Edição, Autor), após isso o utilizador deve obrigatoriamente anexar 2 arquivos (PDF, Imagem)	Clique no botão de publicação e preenchimento dos dados	Dados preenchidos são armazenados na base de dados e vinculados ao ID único do professor a publica-los, assim como também é criado um ID único para a publicação em questão.	Correto armazenamento dos dados preenchidos para a base de dados com a vinculação correta do ID único e criação efetiva do ID da publicação.
	Publicar um texto	Professor preenche 6 campos obrigatórios (Título, gênero, disciplina, descrição, autor e fonte, após isso o utilizador deve obrigatoriamente anexar 2 arquivos (PDF, Imagem)	Clique no botão de publicação e preenchimento dos dados	Dados preenchidos são armazenados na base de dados e vinculados ao ID único do professor a publica-los, assim como também é criado um ID único para a publicação em questão.	Correto armazenamento dos dados preenchidos para a base de dados com a vinculação correta do ID único e criação efetiva do ID da publicação.
	Publicar um exercício	Professor preenche 3 campos obrigatórios (Título, disciplina, descrição), após isso o utilizador deve obrigatoriamente anexar 2 arquivos (PDF, Imagem)	Clique no botão de publicação e preenchimento dos dados	Dados preenchidos são armazenados na base de dados e vinculados ao ID único do professor a publica-los, assim como também é criado um ID único para a publicação em questão.	Correto armazenamento dos dados preenchidos para a base de dados com a vinculação correta do ID único e criação efetiva do ID da publicação.
	Editar um livro	Professor quando visualiza uma publicação na área de livro e é confirmado como o autor da publicação	Clique no botão de edição de uma das publicações	Dados preenchidos são atualizados na base de dados mediante as edições escolhidas pelo utilizador.	Correta substituição dos dados preenchidos para a base de dados mediante ao ID da publicação



		(Através do ID) pode prosseguir para a edição de todos os campos uma vez preenchidos para a publicação do livro (Título, gênero, disciplina, descrição, Edição, Autor, PDF e Imagem)			
	Editar um texto	Professor quando visualiza uma publicação na área de textos e é confirmado como o autor da publicação (Através do ID) pode prosseguir para a edição de todos os campos uma vez preenchidos para a publicação do texto (Título, gênero, disciplina, descrição, fonte, Autor, PDF e Imagem)	Clique no botão de edição de uma das publicações	Dados preenchidos são atualizados na base de dados mediante as edições escolhidas pelo utilizador.	Correta substituição dos dados preenchidos para a base de dados mediante ao ID da publicação
	Editar um exercício	Professor quando visualiza uma publicação na área de exercícios e é confirmado como o autor da publicação (Através do ID) pode prosseguir para a edição de todos os campos uma vez preenchidos para a publicação dos exercícios (Título, disciplina, descrição, PDF e Imagem)	Clique no botão de edição de uma das publicações	Dados preenchidos são atualizados na base de dados mediante as edições escolhidas pelo utilizador.	Correta substituição dos dados preenchidos para a base de dados mediante ao ID da publicação
	Remover um livro	Professor quando visualiza uma publicação na	Clique no botão de remoção de	Dados vinculados a publicação em questão e	Correta eliminação dos dados vinculados ao ID da



		área de livros e é confirmado como autor da publicação (Através do ID) pode prosseguir e deletar a mesma, com a limpeza de todos os dados vinculados a base de dados	uma das publicações	consequentemente vinculados ao ID único do utilizador são removidos da base de dados.	publicação dentro da base de dados.
	Remover um texto	Professor quando visualiza uma publicação na área de textos e é confirmado como autor da publicação (Através do ID) pode prosseguir e deletar a mesma, com a limpeza de todos os dados vinculados a base de dados	Clique no botão de remoção de uma das publicações	Dados vinculados a publicação em questão e consequentemente vinculados ao ID único do utilizador são removidos da base de dados.	Correta eliminação dos dados vinculados ao ID da publicação dentro da base de dados.
	Remover um exercício	Professor quando visualiza uma publicação na área de exercícios e é confirmado como autor da publicação (Através do ID) pode prosseguir e deletar a mesma, com a limpeza de todos os dados vinculados a base de dados	Clique no botão de remoção de uma das publicações	Dados vinculados a publicação em questão e consequentemente vinculados ao ID único do utilizador são removidos da base de dados.	Correta eliminação dos dados vinculados ao ID da publicação dentro da base de dados.
	Visualizar um livro	Utilizador dentro da área de livros ou dentro do “Feed” de conhecimento pode clicar em uma publicação, abrindo assim uma visualização expandida da mesma, porém com todas as informações	Clique em uma das publicações	Exibição de todos os dados inseridos no momento da publicação do livro em questão, com a gestão dos elementos exibidos na visualização mediante ao ID do utilizador visitante e o ID utilizador	Correta exibição de todos os dados inseridos no momento da publicação do livro em questão, com a gestão dos elementos exibidos na visualização mediante ao ID do utilizador visitante e o ID utilizador



		vinculadas ao livro selecionado.		autor da publicação	autor da publicação.
	Visualizar um texto	Utilizador dentro da área de textos ou dentro do “Feed” de conhecimento pode clicar em uma publicação, abrindo assim uma visualização expandida da mesma, porém com todas as informações vinculadas ao texto selecionado.	Clique em uma das publicações	Exibição de todos os dados inseridos no momento da publicação do texto em questão, com a gestão dos elementos exibidos na visualização mediante ao ID do utilizador visitante e o ID utilizador autor da publicação	Correta exibição de todos os dados inseridos no momento da publicação do texto em questão, com a gestão dos elementos exibidos na visualização mediante ao ID do utilizador visitante e o ID utilizador autor da publicação.
	Visualizar um exercício	Utilizador dentro da área de exercícios ou dentro do “Feed” de conhecimento pode clicar em uma publicação, abrindo assim uma visualização expandida da mesma, porém com todas as informações vinculadas ao exercício selecionado.	Clique em uma das publicações	Exibição de todos os dados inseridos no momento da publicação do exercício em questão, com a gestão dos elementos exibidos na visualização mediante ao ID do utilizador visitante e o ID utilizador autor da publicação	Correta exibição de todos os dados inseridos no momento da publicação do exercício em questão, com a gestão dos elementos exibidos na visualização mediante ao ID do utilizador visitante e o ID utilizador autor da publicação.
	Anexar ficheiro em uma publicação	Professor no momento de publicação é solicitado a escolher um documento utilizando-se de qualquer aplicativo satisfatório em seu dispositivo móvel.	Clique no botão de anexo em uma publicação	Mensagem confirmando a recepção do ficheiro, assim como a confirmação do upload efetivo para a base de dados.	Mensagem sendo corretamente enviada tanto na recepção quanto no upload efetivo do anexo.
	Selecionar um gênero em uma publicação	Utilizador dentro do “Feed” de conhecimento pode optar por filtrar as publicações, preenchendo o campo de “Gênero”	Seleção de um gênero por parte do utilizador	Visualização de todas as publicações nas quais seus gêneros correspondem ao filtrado pelo utilizador	Visualização correta de todas as publicações nas quais seus gêneros correspondem ao filtrado pelo utilizador



	Selecionar uma disciplina em uma publicação	Utilizador dentro do “Feed” de conhecimento pode optar por filtrar as publicações, preenchendo o campo de “Disciplina”	Seleção de uma disciplina por parte do utilizador	Visualização de todas as publicações nas quais suas disciplinas correspondem ao filtrado pelo utilizador	Visualização correta de todas as publicações nas quais suas disciplinas correspondem ao filtrado pelo utilizador
Suporte	Enviar uma mensagem ao suporte	O utilizador é capaz de enviar mensagens de suporte devido a problemas que possam ocorrer na aplicação	Clique no botão de envio de mensagem	Abertura de um cliente de envios de email pelo utilizador, com conteúdo já preenchido e pronto para envio	Funcionamento correto de acordo com o esperado para envio de mensagem ao suporte
	Selecionar um cliente para envio da mensagem	Abertura de uma aplicação em segundo plano para que possa ser possível enviar um email ao suporte do “Yourspot”	Seleção de um cliente de e-mail por parte do utilizador	Abertura de uma aplicação secundária de serviços de email	Abertura de aplicação de serviço de email como esperado em segundo plano

### 6.3. Testes não funcionais

Para testes não funcionais, foram realizados alguns métodos de avaliação que geraram alguns resultados, são eles:

- **Testes de confiabilidade e Integridade dos dados:** Em todos os testes realizados os dados armazenados pela aplicação são somente acessados pelos próprios detentores destes, ou pelos gestores da base de dados da aplicação. De mesmo modo, dados como a palavra-passe são encriptados na própria base dados, dessa forma, não permitindo a sua leitura indevida.
- **Testes de eficiência:** Nos testes feitos tanto em ambiente simulado e prática, observamos que a aplicação consegue se manter eficiente mesmo com a execução simultânea de varias aparelhos. Entretanto em uma larga escala será necessário um servidor de mesma proporção para que possa manter a aplicação com o mesmo desempenho.
- **Testes de usabilidade:** Realizados os devidos testes, todos estes obtiveram um resultado satisfatório no quesito usabilidade, os menus e botões são de fácil acesso e estão visíveis para o utilizador, a navegação é simples e se mantem de fácil aprendizado, o design é responsivo mediante a sua utilização em um dispositivo móvel diferente, foram testados diversos modelos de telemóveis e todos adaptaram-se bem a visualização da interface. Vale ressaltar que a aplicação emite mensagens de erros





específicas caso algo aconteça, assim o utilizador poderá encontrar seu erro facilmente e corrigi-lo.

- **Testes de performance:** Obtivemos respostas da aplicação em um tempo satisfatório mostrando assim uma boa performance no cenário prático quando executados em telemóveis. No cenário virtual isso era variante mediante ao computador que executava a máquina virtual para teste.

## 7. Conclusões e Trabalho Futuro

---

### 7.1. Conclusões

De modo a concluir tanto o que foi visto neste documento quanto na sua forma prática em um cenário de utilização como forma de aplicação móvel, abordamos aqui um tema ambicioso, porém de bastante utilidade e necessidade nos dias atuais, principalmente no cenário atual ao qual todos nos encontramos, relativo à pandemia causada pelo vírus do Covid-19.

Nosso tema tratava-se da Gestão pessoal em âmbito acadêmico, sabemos o quão importante é a organização tanto para alunos quanto para professores no dia a dia e isso nos motivou a desenvolver esta aplicação, manter-se organizado sem o auxílio de ferramentas que proporcionem isso, é bastante complicado, e para um aluno e um professor isso é essencial no seu percurso e desenvolvimento acadêmico, mas não só isso, queríamos que nossa aplicação fosse além de um gestor pessoal, queríamos que ela também servisse de fonte de informação compartilhada, por isso decidimos dividi-la em duas vertentes, uma seria voltada para o aspecto de gestão pessoal e a outra seria voltada para a fornecimento e troca de informação acadêmica.

Falando da outra vertente, achamos que compartilhar informação é o que realmente move o mundo hoje, isso é visto em todos os lugares, como a própria internet e redes sociais, e porque não ter um ambiente assim voltado para o cenário acadêmico? Assim surgiu nossa proposta, na nossa aplicação é possível haver essa troca entre o professor e seus alunos, dessa forma o professor pode fornecer materiais como textos, livros e exercícios para seus alunos e estes podem consultar os mesmos e compartilhar ideias entre si, possíveis resoluções de um exercício, entre outros pontos.

E foi assim que o “Yourspot” surgiu, com a ideia de tornar esse cenário mais agradável e conectar professores e alunos de modo que todos possam desenvolver-se academicamente.

## 7.2. Trabalho Futuro

Diante do tempo e recursos que nos foi dado, tivemos de priorizar alguns pontos da nossa aplicação, por isso, não nos foi possível termina-la ao todo, existiam algumas funcionalidades as quais não foram possíveis de serem implementadas que tínhamos em mente e estas serão dadas como trabalho futuro a ser realizado. São elas:

- **Postagens Curtidas:** Temos como funcionalidade para trabalho futuro algo que é bastante importante no ponto de vista de quem está a utilizar o “Yourspot” diariamente, que é o fator de “favoritar” e ser capaz de visualizar em forma de lista as postagens que foram “curtidas”, desse modo o utilizador poderia ter a informação das postagens que são importantes para si e um fácil acesso das mesmas.
- **Chat de mensagens instantâneas entre utilizadores:** Como nossa aplicação trata de conectar pessoas, algo que tivemos de colocar como trabalho futuro foi a realização de um chat de mensagens instantâneas para que os utilizadores se comuniquem entre si. Devido à complexidade e o curto período de tempo tornou-se complicada a sua implementação em tempo hábil.



**Figura 11 - Trabalho futuro, Chat de mensagens YourSpot**



- **Configurações da aplicação:** Uma funcionalidade que gostaríamos de implementar, mas que também não houve tempo hábil para tal e ficará para trabalho futuro é a elaboração de configuração personalizada para cada utilizador, onde o mesmo poderia gerir algumas configurações relevantes do uso da aplicação.



***Figura 12 - Trabalho futuro, Configurações do YourSpot***