

Underlying

**PROJETO 2022.01 – UNDERLYING
RELATÓRIO FINAL – RF**

Versão 0.5

Equipe de Projeto Underlying:

Bruno Brandão Borges - 2018014331

Ivan Leoni Vilas Boas - 2018009073

Leonardo Rodrigo de Sousa - 2018015965

Lucas Tiense Blazzi - 2018003310

Thiago Marcelo Passos - 2018002850

Wesley Alexandre de Almeida Gomes - 2018005806



IMC - Instituto de Matemática e Computação

Av. BPS, 1303 - Caixa postal 50 - 37500-903

Itajubá - MG - Brasil Telefone: 35-3629-1135

E-mail: imc@unifei.edu.br

Revisões do Documento

Revisões são melhoramentos na estrutura do documento e também no seu conteúdo. O objetivo primário desta tabela é a fácil identificação da versão do documento. Toda modificação no documento deve constar nesta tabela.

Data	Versão	Descrição	Autor
15/04/2022	0.0	Descrição do Sistema	Ivan
15/04/2022	0.0	Elaboração dos Requisitos de Sistemas (RFS)	Ivan
22/04/2022	0.1	Elaboração dos Requisitos de Não Funcionais (RNF) organizacionais e externos	Ivan
22/04/2022	0.1	Persona e histórias de usuário	Ivan
22/04/2022	0.1	Descrição da equipe	Ivan
22/04/2022	0.1	Backlog do Produto	Ivan
22/04/2022	0.2	Backlog do Sprint 1	Ivan
26/04/2022	0.2	Inserção dos Mockups do Sistema a ser desenvolvido	Ivan
26/04/2022	0.2	Inserção de RNF de Produto	Ivan
30/04/2022	0.3	Inserção de RNF de Produto	Ivan
30/04/2022	0.3	Inserção dos Mockups do Sistema a ser desenvolvido	Ivan
30/05/2022	0.4	Inserção de RNF Adicionais	Ivan
04/06/2022	0.4	Alterações solicitada pela Lina	Ivan
05/06/2022	0.4	Sprint 1 - Conclusões	Lucas
17/07/2022	0.5	Sprint 2 - Conclusões	Ivan/Lucas
Referências			
04/04/2022	0.0	Entrevista	Lucas
19/04/2022	0.1	Cronograma Sprint 1	Lucas
26/04/2022	0.2	Stack Tecnológico	Lucas
26/04/2022	0.2	RNF de Produto – Atributos de Qualidade	Ivan
04/06/2022	0.4	Cronograma Sprint 2	Ivan
Documentações em anexo			
16/07/2022	0.5	Anexo 1 - Cronograma Sprint 3	Ivan
10/06/2022	0.4	Anexo 2 – Documento da Arquitetura	Lucas/ Ivan
17/06/2022	0.5	Anexo 3 – Código fonte do sistema	Lucas e Leonardo
27/06/2022	0.5	Anexo 4.1 – Testes Estrutural do sistema	Ivan / Leonardo / Lucas / Bruno

			/ Thiago / Wesley
02/07/2022	0.5	Anexo 4.2 – Testes Funcionais do sistema	Ivan / Leonardo / Lucas / Bruno / Thiago / Wesley
05/07/2022	0.5	Anexo 5 - Código dos Testes Unitários e de Integração do Sistema	Lucas
12/07/2022	0.5	Anexo 6.1 - Planejamento dos Testes de Aceitação com Usuário Final	Lucas/ Leonardo/ Ivan
12/07/2022	0.5	Anexo 6.2 - Execução dos Testes de Aceitação com Usuário Final	Lucas/ Leonardo/ Ivan
16/07/2022	0.5	Anexo 7 – Arquivos dos Testes não funcionais	Lucas
16/07/2022	0.5	Anexo 8 – Relatório de pendencias e conclusões	Ivan
17/07/2022	0.5	Anexo 9 – Slides da sprint 2 do projeto Underlying	Ivan / Leonardo / Lucas / Bruno / Thiago / Wesley

Auditorias do Documento

Auditorias são inspeções conduzidas o SEPG – Software Engineer Process Group (Grupo de Engenharia de Processo de Software), e tem por objetivo garantir uma qualidade mínima dos artefatos gerados durante o processo de desenvolvimento. Essa tabela pode ser utilizada também pelo GN – Gerente da Área de Negócio com o objetivo de documentar a viabilidade do mesmo.

Data	Versão	Descrição	Autor
16/04/2022	0.0	Inspeção dos RFS	Lucas
22/04/2022	0.1	Inspeção dos RNF	Lucas
22/04/2022	0.1	Backlog do Produto	Lucas
23/04/2022	0.2	Backlog do Sprint 1	Lucas
30/04/2022	0.3	Inspeção dos documentos em Anexos	Ivan
30/05/2022	0.3	Revisão completa	Lina
05/06/2022	0.4	Inspeção completa	Lucas
17/07/2022	0.5	Inspeção completa	Lucas/Ivan

Sumário

1. INTRODUÇÃO.....	9
1.1 CONVENÇÕES, TERMOS E ABREVIACÕES.....	9
1.1.1 Identificação das histórias de usuário e dos Requisitos	10
1.1.2 Prioridades dos Requisitos.....	10
2. EQUIPE DO PROJETO.....	10
3. VISÃO GERAL DO PRODUTO/SERVIÇO.....	11
3.1 ABRANGÊNCIA E SISTEMAS RELACIONADOS.....	12
3.2 DESCRIÇÃO DO CLIENTE.....	12
3.3 DESCRIÇÃO DOS USUÁRIOS	13
4. PERSONA E HISTÓRIAS DE USUÁRIO	13
4.1 PERSONA ÚNICA DO SISTEMA	13
4.1.1 Principais Objetivos da Persona Pedro:.....	14
4.2 HISTÓRIAS DE USUÁRIO	14
5. REQUISITOS FUNCIONAIS.....	15
5.1 RF01 - MANTER OPÇÕES.....	15
[RFS01] CONSULTAR OPÇÕES	15
[RFS02] VISUALIZAR OPÇÕES	16
[RFS03] CRAWLER DE DADOS DE REGISTRO DE OPÇÕES	17
5.2 RF02 - MANTER HISTÓRICO DIÁRIO DOS PREÇOS DAS OPÇÕES.....	18
[RFS04] REALIZAR CRAWLER DE DADOS DE TRANSAÇÕES	18
[RFS05] GERAR RELATÓRIO DE COTAÇÃO DE OPÇÃO	18
5.3 RF03 - MANTER USUÁRIO	19
[RFS06] INSERIR USUÁRIO.....	19
[RFS07] CONSULTAR USUÁRIO	20
[RFS08] EDITAR USUÁRIO	21
[RFS09] REMOVER USUÁRIO	21
5.4 RF04 - MANTER MÉTRICAS DE INVESTIMENTO DAS OPÇÕES	21
[RFS10] CALCULAR E APRESENTAR O VALOR INTRÍNSECO.....	22
[RFS11] CALCULAR E APRESENTAR O VALOR NO TEMPO	22
[RFS12] CALCULAR E APRESENTAR O DELTA.....	23
[RFS13] CALCULAR E APRESENTAR O THETA.....	23
[RFS14] CALCULAR E APRESENTAR O RHO	24
[RFS15] CALCULAR E APRESENTAR O VEGA	24
[RFS16] CALCULAR E APRESENTAR O GAMA.....	25
[RFS17] CALCULAR E APRESENTAR O ÍNDICE RATIO.....	26
5.5 RF05 - MANTER ESTRUTURA DE OPERAÇÃO	26
[RFS18] INSERIR ESTRUTURA DE OPERAÇÃO	26
[RFS19] CONSULTAR ESTRUTURA DE OPERAÇÕES.....	27
[RFS20] EDITAR ESTRUTURA DE OPERAÇÃO	27
[RFS21] REMOVER ESTRUTURA DE OPERAÇÃO	28
5.6 RF06 - COMPARTILHAR DA ESTRUTURA DE OPERAÇÃO.....	28
[RFS22] COMPARTILHAR ESTRUTURA	28



Underlying



[RFS23] CONSULTAR ESTRUTURA COMPARTILHADA.....	29
5.7 RF07 – GERAR PAYOFFS DE OPERAÇÃO.....	30
[RFS24] CALCULAR E APRESENTAR PAYOFF.....	30
[RFS25] - GERAR GRÁFICO PAYOFF.....	30
5.8 RF08 – MANTER AÇÕES.....	31
[RFS26] CONSULTAR AÇÕES.....	31
[RFS27] SELECIONAR AÇÕES.....	32
5.9 RF09 – MANTER GRADE DE OPÇÕES	32
[RFS28] GERAR GRADE DE OPÇÕES	32
5.10 RF10 – MANTER OPÇÕES FICTÍCIAS.....	33
[RFS29] ADICIONAR OPÇÃO FICTÍCIA NA ESTRUTURA.....	34
[RFS30] EDITAR OPÇÃO FICTÍCIA	34
[RFS31] REMOVER OPÇÃO FICTÍCIA	35
6. REQUISITOS NÃO FUNCIONAIS.....	35
6.1 RNF DO PRODUTO.....	35
6.1.1. USABILIDADE.....	36
[RNF01] - PADRONIZAR INTERFACES	36
[RNF02] - LEGENDAR GRÁFICOS.....	37
[RNF03] - VALIDAR DADOS DE ENTRADA.....	37
[RNF04] – ABRIR NOVAS ABAS PARA LINKS EXTERNOS	37
[RNF05] – UTILIZAR RESPONSIVIDADE.....	37
[RNF06] – INCLUIR OPÇÕES DE ALTERAÇÃO VISUAL.....	38
[RNF07] – INCLUIR OPÇÃO DE ALTERAÇÃO DE LÍNGUA.....	38
6.1.2. SEGURANÇA.....	38
[RNF08] - CRIPTOGRAFIAR SENHAS E DADOS DOS USUÁRIOS.....	38
[RNF09] – VERIFICAR AUTENTICIDADE	39
[RNF10] – REALIZAR LOGOUT	39
[RNF11] – PERMITIR APENAS SENHAS FORTES	39
[RNF12] – AUTENTICAR USUÁRIO	40
6.1.3. CONFIABILIDADE.....	40
[RNF13] – DESFAZER AÇÃO DE EXCLUSÃO.....	40
[RNF14] – DESFAZER AÇÃO DE COMPARTILHAMENTO	40
[RNF15] – INTERROMPER SOLICITAÇÃO	41
[RNF16] – APRESENTAR DISPONIBILIDADE (99%).....	41
[RNF17] – APRESENTAR INFORMAÇÕES REAIS, ATUAIS E CONSISTENTES COM O MERCADO DE OPÇÕES.....	41
[RNF18] – RECUPERAR DE FALHAS	42
6.1.4. DESEMPENHO / EFICIÊNCIA.....	42
[RNF19] – APRESENTAR TEMPO DE RESPOSTA BAIXO (2S)	42
[RNF20] – APRESENTAR TEMPO DE THROUGHPUT BAIXO (1S)	42
[RNF21] – REALIZAR ESCALABILIDADE	43
6.1.5. COMPATIBILIDADE.....	43
[RNF22] – ACESSAR EM DIVERSOS NAVEGADORES.....	43
6.2 RNF ADICIONAIS	43
[RNF23] – ADICIONAR GLOSSÁRIO [USABILIDADE].....	44
6.3 RNF ORGANIZACIONAIS.....	44
[RNF24] – MÉTODO ORGANIZACIONAL	44
[RNF25] – REPOSITÓRIO.....	44
[RNF26] – TECNOLOGIAS A SEREM UTILIZADAS.....	45
[RNF27] – TEMPO DE ENTREGA DA TAREFA.....	45



Underlying



[RNF28] – ENTREGA DE ARTEFATOS	45
[RNF29] – ARMAZENAMENTO DE DADOS	46
[RNF30] – ALTERAÇÃO NOS REQUISITOS	46
6.4 RNF EXTERNOS	46
[RNF31] – LEI GERAL DE PROTEÇÃO DE DADOS PESSOAIS	47
[RNF32] – INTEGRAÇÃO COM SISTEMAS EXTERNOS:	47
[RNF33] – ÉTICO	47
7. BACKLOG DO PRODUTO	47
8. BACKLOG DA SPRINT 01	53
8.1 VELOCIDADE DA EQUIPE:	55
8.2 RETROSPECTIVA DA EQUIPE DO SPRINT 1	56
8.2.1 O que funcionou bem?	56
8.2.2 O que deu errado?	57
8.2.3 Ações para melhorar	58
9. BACKLOG DA SPRINT 02	59
9.1 VELOCIDADE DA EQUIPE:	62
9.2 IMPACTO DA ARQUITETURA NA SPRINT 2	64
9.3 RETROSPECTIVA DA EQUIPE DO SPRINT 2	64
9.3.1 O que funcionou bem?	64
9.3.2 O que deu errado?	65
9.3.3 Ações para melhorar	65
10. BACKLOG DA SPRINT 03	66
11. MOCKUPS DO SISTEMA:	68
PROTÓTIPO 01 – HOME (PAGINA INICIAL)	69
PROTÓTIPO 02 – LOGIN DE USUÁRIO	70
PROTÓTIPO 03 – CADASTRO DE USUÁRIO (REGISTRO)	71
PROTÓTIPO 04 - GRADE DE OPÇÕES	72
PROTÓTIPO 05 – VISUALIZAR OPÇÕES	73
PROTÓTIPO 06 – CRIAÇÃO DE OPÇÃO FICTÍCIA	74
PROTÓTIPO 07 - BUSCAR OPÇÃO	75
PROTÓTIPO 08 – CRIAR ESTRUTURA DE OPERAÇÃO	76
PROTÓTIPO 09 - ESTRUTURA DE OPERAÇÃO COMPARTILHADA	77
12. REFERÊNCIAS E ANEXOS:	77



Underlying



Sumario de tabelas

TABELA 1 - HISTÓRIAS DE USUÁRIO	15
TABELA 2 - DADOS DO USUÁRIO	20
TABELA 3 - DADOS DA ESTRUTURA DE OPERAÇÕES.....	26
TABELA 4 - FILTRO DA ESTRUTURA DE OPERAÇÕES.....	27
TABELA 5 - FILTROS DO RELATÓRIO DE GRADE DE OPÇÕES.....	33
TABELA 6 - DADOS DA OPÇÃO FICTÍCIA	34
TABELA 7 - BACKLOG DO PRODUTO	53
TABELA 8 - BACKLOG DA SPRINT 01	55
TABELA 9 - RETROSPECTIVA DA SPRINT 01	59
TABELA 10 - BACKLOG DA SPRINT 02	62
TABELA 11 - RETROSPECTIVA DA SPRINT 02	66



Underlying



Sumário de Figuras

FIGURA 1 - EQUIPE SCRUM	11
FIGURA 2 - VISUALIZAR OPÇÕES	17
FIGURA 3 - PREÇO DA OPÇÃO.....	17
FIGURA 4 - HISTÓRICO DE PREÇOS	19
FIGURA 5 - ESTRUTURA COMPARTILHADA.....	29
FIGURA 6 - PAYOFF : PREÇO X LUCRO	31
FIGURA 7 - GRADE DE OPÇÕES	33
FIGURA 8 - PADRONIZAÇÃO DE INTERFACES.....	36
FIGURA 9 - GRÁFICO DE CONTROLE SPRINT 01.....	56
FIGURA 10 - DESEMPENHO SPRINT 02.....	63
FIGURA 11 - PROTÓTIPO 01 – HOME (PAGINA INICIAL)	69
FIGURA 12 - PROTÓTIPO 02 – LOGIN DE USUÁRIO.....	70
FIGURA 13 - PROTÓTIPO 03 – CADASTRO DE USUÁRIO (REGISTRO).....	71
FIGURA 14 - PROTÓTIPO 04 - GRADE DE OPÇÕES	72
FIGURA 15 - PROTÓTIPO 05 – VISUALIZAR OPÇÕES.....	73
FIGURA 16 - PROTÓTIPO 06 – CRIAÇÃO DE OPÇÃO FICTÍCIA	74
FIGURA 17 - PROTÓTIPO 07 - BUSCAR OPÇÃO	75
FIGURA 18 - PROTÓTIPO 08 – CRIAR ESTRUTURA DE OPERAÇÃO	76
FIGURA 19 - PROTÓTIPO 09 - ESTRUTURA DE OPERAÇÃO COMPARTILHADA	77

1. INTRODUÇÃO

Este documento especifica os requisitos do sistema UNDERLYING fornecendo aos desenvolvedores as informações necessárias para a execução de seu projeto e implementação, assim como para a realização dos testes e homologação.

Esta introdução fornece as informações necessárias para fazer um bom uso deste documento, explicitando seus objetivos e as convenções que foram adotadas no texto. As demais seções apresentam a especificação do UNDERLYING e estão organizadas como descrito abaixo:

Seção 2 – Equipe de projeto: Apresenta a equipe e a sua responsabilidade com o projeto desenvolvido em SCRUM.

Seção 3 - Descrição geral do produto/serviço: apresenta uma visão geral do produto/serviço, caracterizando qual é o seu escopo e descrevendo seus usuários.

Seção 4 - Persona e Histórias de Usuário: apresenta a Persona do Pedro um do mercado de Opções e as histórias de usuário para a geração dos requisitos

Seção 5 - Requisitos funcionais: lista e descreve os requisitos funcionais do produto/serviço, especificando seus objetivos, funcionalidades, atores e prioridades.

Seção 6 - Requisitos não funcionais: especifica todos os requisitos não funcionais do produto/serviço, divididos em requisitos de usabilidade, confiabilidade, desempenho, segurança, distribuição, adequação a padrões e requisitos de hardware e software.

Seção 7 - Backlog do produto: Apresenta o Backlog do produto em uma planilha contendo a tarefa, esforço e sprint.

Seção 8 - Backlog da Sprint 1: Apresenta o Backlog da sprint 1 com a velocidade da equipe, a retrospectiva da equipe contendo os acertos, erros e melhorias.

Seção 9 - Backlog da Sprint 2: Apresenta o Backlog da sprint 2 com a velocidade da equipe, a retrospectiva da equipe contendo os acertos, erros e melhorias.

Seção 10 - Backlog da Sprint 3: Apresenta o Backlog da sprint 3 com a estimativa em uma planilha contendo a tarefa, esforço, responsável e data de entrega

Seção 11 – Mockups do sistema a ser desenvolvido: Apresenta prints das telas do sistema que foram idealizadas com uma breve descrição dos requisitos funcionais do sistema que se pretendem abordar em cada tela.

Seção 12 - Referências e Anexos: são citados os documentos em anexos que colaboram para o desenvolvimento deste projeto.

1.1 Convenções, termos e abreviações

A correta interpretação deste documento exige o conhecimento de algumas convenções e termos específicos, que são descritos a seguir.

1.1.1 Identificação das histórias de usuário e dos Requisitos

Por convenção, a referência a requisitos ou a história é feita através do identificador, de acordo com o esquema abaixo:

[identificador de tipo + identificador Numérico]

O identificador de tipo de requisito pode ser:

- HU – História de usuário
- RF – Requisito funcional
- RFS – Requisito funcional de sistema
- RNF – Requisito não-funcional

Identificador é um número, criado sequencialmente, que determina que aquele requisito/história é único para um determinado tipo.

Exemplos: HU01, HU02, RF01, RF02, RFS01, RFS02, RNF01, RNF02.

1.1.2 Prioridades dos Requisitos

Para estabelecer a prioridade dos requisitos foram adotadas as denominações “essencial”, “importante” e “desejável”.



Essencial é o requisito sem o qual o sistema não entra em funcionamento. Requisitos essenciais são requisitos imprescindíveis, que têm que ser implementados impreterivelmente.

Importante é o requisito sem o qual o sistema entra em funcionamento, mas de forma não satisfatória. Requisitos importantes devem ser implementados, mas, se não forem, o sistema poderá ser implantado e usado mesmo assim.

Desejável é o requisito que não compromete as funcionalidades básicas do sistema, isto é, o sistema pode funcionar de forma satisfatória sem ele. Requisitos desejáveis são requisitos que podem ser deixados para versões posteriores do sistema, caso não haja tempo hábil para implementá-los na versão que está sendo especificada.

2. EQUIPE DO PROJETO

A equipe do projeto Underlyng é formada por 6 integrantes que irão realizar o desenvolvimento por SCRUM. Os integrantes com as respectivas funções e as responsabilidades são descritas as seguir:

-  Bruno – (Desenvolvimento) - Back
-  Ivan - (Scrum Master) - Documentação / Teste

- ✚ Leonardo - (Desenvolvimento) - Frontend / UX
- ✚ Lucas - (Product Owner) - Back / Arquitetura
- ✚ Thiago - (Desenvolvimento) - Frontend / UX / Teste
- ✚ Wesley - (Desenvolvimento) - Frontend / UX

A função de cada integrante no projeto Underlying pode ser mais bem observada no organograma SCRUM a seguir:

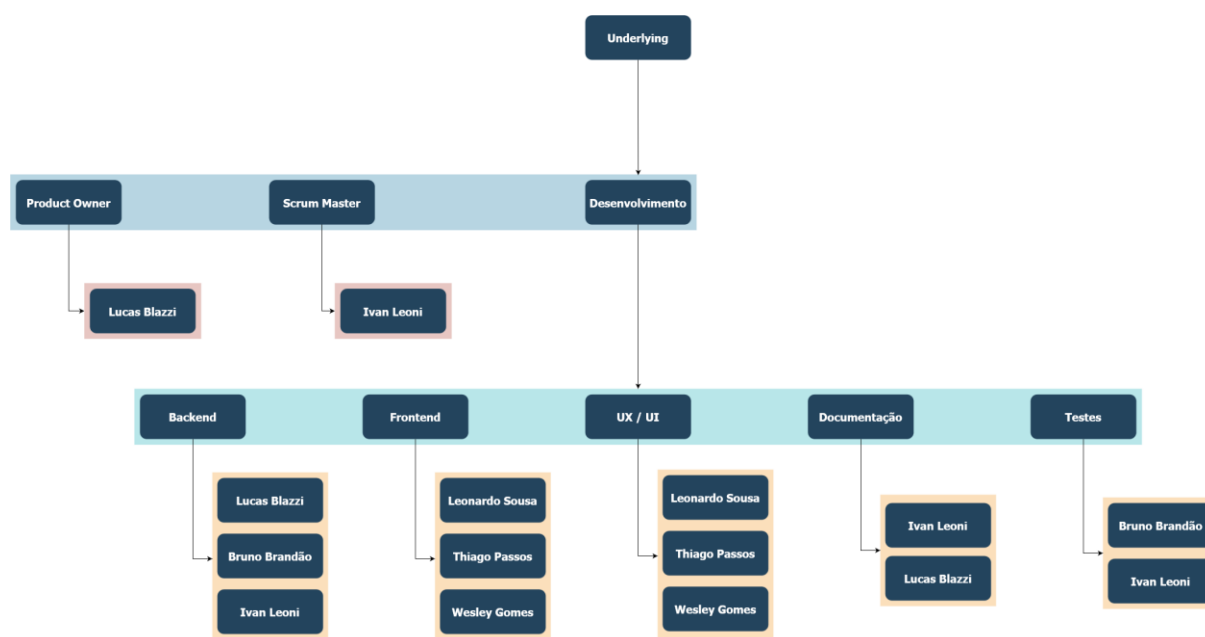


Figura 1 - Equipe Scrum

Cada integrante deverá atentar-se para os requisitos deste documento e utilizar rotineiramente a ferramenta Jira para executar as suas tarefas com a máxima qualidade possível. As tarefas podem ser vistas no Jira. Abaixo segue o link do cronograma do projeto a ser realizado:

<https://underlying.atlassian.net/jira/software/c/projects/UN/boards/1>

3. VISÃO GERAL DO PRODUTO/SERVIÇO

A plataforma web Underlying funcionará via web e será de grande valia ao mercado financeiro, pois terá como foco o produto de investimento **opções**. O usuário poderá acessar o registro histórico de opções negociadas na bolsa. Além de dados cadastrais, o



Underlying



serviço fornecerá variáveis de estratégia (greeks) previamente calculadas para trazer informações mais avançadas a respeito das opções, que poderão auxiliar o usuário em sua busca. Será possível navegar por uma grade de opções a partir da ação objeto em um determinado período, tendo assim uma visão geral de todo o escopo selecionado. A plataforma contará com a possibilidade de criação de opções fictícia, além das reais, para realização de testes de composição de operação, visualização gerada a partir de um gráfico de payoff. Ao acessar o registro de uma opção específica, o usuário contará com uma visão detalhada e gráfica das negociações do determinado papel desde seu início. A partir do registro do usuário será possível que ele salve suas opções fictícias criadas e compartilhe sua operação estruturada com os demais usuários da plataforma. As informações do sistema serão provenientes do banco dados oriundos da B3.

3.1 Abrangência e sistemas relacionados

O Sistema web UNDERLYING agregará valor aos usuários uma vez que permitirá consultar dados referentes a Opções expiradas e disponíveis para gerar cenários de possíveis operações, auxiliando assim no processo de tomada de decisão.

Entre as funcionalidades do sistema teremos as seguintes *Features*:

- Visualização de registro de opções
- Gráfico de diário de transações
- Criação de operações estruturadas
- Geração de payoffs
- Mecanismo de busca avançada por filtros
- Grade das opções de uma ação
- Perfil específico para cada usuário / salvar operações
- Cálculo de métricas de mercado (greeks + payoff) por opção
- Compartilhamento das estruturas de operações
- Criação de opções fictícias para teste de operação

Para suas funcionalidades o sistema deverá ser capaz de realizar diariamente de forma automática a busca por registros de Opções em plataformas de cadastro para popular o banco de dados.

3.2 Descrição do cliente

Visto a necessidade do mercado financeiro e de seus usuários, e, a fim de auxiliar os investidores no processo de tomada de decisão através de dados apurados e gráficos o projeto está sendo desenvolvido para qualquer usuário investidor, seja um aprendiz que queira adentrar ao mundo dos investimentos ou um experiente investidor que queira melhorar sua experiência como usuário da plataforma, a fim de tomarem as melhores decisões de mercado. O sistema é indicado principalmente para aqueles investidores que



Underlying



que desejam obter informações de produtos do mercado de opções e testar estruturas de operações.

3.3 Descrição dos usuários

A aplicação possuirá apenas um tipo de usuário. Este por sua vez será capaz de realizar todas as operações que o sistema ofereça, desde que seja realizado o cadastro e login na plataforma.

4. PERSONA E HISTÓRIAS DE USUÁRIO

Nesta sessão ser apresentado a Persona Pedro, um investido, que representará o principal usuário do sistema e logo a seguir as histórias de Usuário.

4.1 Persona Única do Sistema



Pedro Costa Damasceno é um investidor do mercado de Opções. Ele opera mercado de opções a 3 anos. Se formou em Administração da universidade de Santa Catarina em 2021. Possui 26 anos e utiliza diariamente seu notebook para realizar pesquisas na B3 e demais sites para se informar melhor sobre o mercado de Opções. Atualmente ele utiliza um sistema pago para obter as informações que precisa para tomar as melhores decisões acerca do mercado de Opções, porém não somente o design do sistema o

incomoda como os gráficos não lhe proporcionam uma boa experiência de usuário, falta a disponibilização de alguns dados que são vistos como importantes para análises mais complexas, o processo ainda é muito manual para o acesso aos dados e para a montagem de cenários fictícios e, ainda apresenta uma limitação visual na geração dos cenários, e, somado a isso, o custo de acesso a tais plataformas atuais são caras.

Pedro esperava obter um sistema web que seja intuitivo e que ele consiga achar as informações de maneira objetiva e com gráficos fáceis de serem interpretados. Ele espera que com o auxílio de recursos visuais e com a possibilidade de busca através de filtros avançados, ele possa obter os dados de Opções, mesmo que expiradas, de forma organizada para sua melhor análise e que ao selecionar múltiplas Opções para compor uma operação resulte em gráfico de Payoff de fácil entendimento para sua tomada de decisão.

Pedro tem a necessidade de visualizar mais informações sobre as Opções além do básico cadastral, como os cálculos das greeks, fazendo assim cada usuário do sistema criar e manter suas próprias estratégias no próprio sistema. Além disso, Pedro gostaria que o sistema tivesse a possibilidade adicional de salvar as estruturas de operações montadas e de exibir a visão macro geral do cenário de cada uma das opções individualmente.

Pedro acredita que este novo sistema traria mais usuários e seria um grande sucesso entre os investidores se caso além de criar Opções, estas pudessem ser criadas de forma fictícia para realizar uma análise mais interessante, também se as estruturas de operação pudessem ser compartilhadas e vistas por seus amigos e demais usuários do sistema.

4.1.1 Principais Objetivos da Persona Pedro:

- ✓ Ter acesso ao sistema;
- ✓ Utilizar um sistema de forma intuitiva com boa experiência de usuário;
- ✓ Facilidade em obter informações e analisar gráficos;
- ✓ Utilização de filtros avançados e específicos;
- ✓ Obter todos os dados (expirados ou não) para análise,
- ✓ Visualizar Gráfico de payoff de cada Opção;
- ✓ Obter métricas de mercado (greeks);
- ✓ Cada usuário com um perfil de cadastro onde suas estratégias ficarão registradas;
- ✓ Criar estrutura de Opções;
- ✓ Criar Opções Fictícias;
- ✓ Fazer compartilhamento das estruturas de operação;
- ✓ Ser capaz de realizar melhores escolhas possíveis tomando melhores decisões.

4.2 Histórias de Usuário

A seguir apresentaremos as histórias de usuário (HU) provenientes da entrevista e da persona Pedro, principal usuário deste sistema.

Identificador	História de usuário
HU01	Como usuário, Pedro quer buscar por opções de ações para visualizar dados cadastrais e de cotação histórica da Opção selecionada.
HU02	Como usuário, Pedro quer visualizar métricas de desempenho e <i>greeks</i> de uma Opção buscada na plataforma.
HU03	Como usuário, Pedro quer visualizar o gráfico de <i>Payoff</i> de uma Opção específica.
HU04	Como usuário, Pedro quer possuir um perfil próprio no sistema (acessado por login e senha), onde será possível que eu visualize seus recursos criados e a sua estratégia de análises.
HU05	Como usuário, Pedro quer ter a possibilidade de buscar Opções utilizando filtros avançados de busca que permitam limitar o escopo.
HU06	Como usuário, Pedro quer elaborar uma estrutura de operação de Opções e poder visualizar o gráfico de <i>Payoff</i> referente a essa estrutura
HU07	Como usuário, Pedro quer ter a possibilidade de criar Opções fictícias que possam ser utilizadas na composição de uma estrutura de operação de Opções



HU08	Como usuário, Pedro quer visualizar a grade de Opções de uma ação a partir da seleção de uma ação através do seu ticker
HU09	Como usuário, Pedro quer visualizar métricas de mercado e greeks na grade de Opção
HU10	Como usuário, Pedro quer ter a opção de compartilhar a suas estruturas de operações para que todos os demais usuários da plataforma consigam visualizá-las.
HU11	Como usuário, Pedro quer visualizar os termos comuns utilizados no mercado de opções na forma de um glossário, para consultar quando possuir alguma dúvida

Tabela 1 - Histórias de Usuário

5. REQUISITOS FUNCIONAIS

Nesta sessão estão listados os requisitos funcionais (RF) e os requisitos funcionais de sistema (RFS), que compreendem as funcionalidades e serviços que definem o sistema. Sem a correta implementação desses requisitos a funcionalidade do sistema ficará comprometida.

5.1 RF01 - Manter Opções

As funcionalidades apresentadas nesta seção são os requisitos funcionais de sistema que permitirão ao sistema consultar as Opções existentes.

Os RF aqui declarados são provenientes do HU01 e HU05

O sistema web terá integração direta com o banco de dados. Todos as Opções já estarão cadastradas no banco, podendo apenas serem consultados, não apresentado assim cadastro, alteração ou a sua exclusão.

[RFS01] Consultar Opções

Ator: Usuário

Este requisito começa quando o ator desejar consultar as Opções disponíveis no sistema. Será possível consultar as Opções através dos filtros **ativo base (underlying)**, **mês de execução**, **preço de strike**. Por default caso o filtro não seja preenchido (nulo) as opções serão exibidas a partir do nome buscado.

Será retornado então uma lista de opções que atendam os requisitos estabelecidos previamente no filtro.



A lista de opções será acompanhada dos atributos: **nome, ativo base, data de expiração, preço strike e valor atual da cota.**

<<Opção01>>

<<Ticker01>><<Ativo_Base01>><<Expiração01>><<Preço_Strike01>><<Cota01>

<<Opção02>>

<<Ticker02>><<Ativo_Base02>><<Expiração02>><<Preço_Strike02>><<Cota02>

<<Opção03>>

<<Ticker03>><<Ativo_Base03>><<Expiração03>><<Preço_Strike03>><<Cota03>>

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

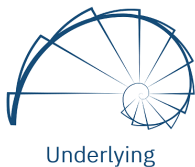
[RFS02] Visualizar Opções

Ator: Usuário

Este requisito começa após o usuário selecionar uma Opção específica depois de passar pelo processo de consulta (**RFS01**). A seleção da Opção retornará todos os dados de registro da Opção acompanhados de séries de preço e informações de desempenho previamente calculadas. Assim, o usuário visualizará os atributos **tipo, ativo base, preço de exercício, data de execução, empresa alvo, greeks e um relatório** no formato de gráfico de linhas (**RFS05**) referente série de **preço da Opção**.

A Opção será apresentada da seguinte forma:

☰	⚙
BBASE468 - CALL DE BBAS3 - STRIKE R\$ 45,90 - VENCIMENTO 20/05/2022	



BBASE468							
BBASE468 - Cotação não ajustada							
	Min	Pri	Med	Ult	Max	Negócios	Vol. Fin.
22/04/2022	0,01	0,01	0,01	0,15	0,15	2	66
22/04/2022	0,01	0,12	0,10	0,01	0,16	25	55
Min	0,01	0,01	0,01	0,01	0,15	2	98
Max	0,01	0,12	0,10	0,15	0,16	25	27

BBASE468 - Volatilidade implícita						BBASE468 - Cotação não ajustada				
	Min	Pri	Med	Ult	Max	Min	Abe	Med	Ult	Max
22/04/2022	40,00	40,00	48,23	59,89	59,89	34,15	34,73	34,43	34,43	34,77
22/04/2022	37,00	54,19	53,58	37,00	57,37	34,81	35,24	34,99	35,16	35,27
Min	37,00	40,00	48,23	37,00	57,37	34,15	34,73	34,43	34,43	34,77
Max	40,00	54,19	53,58	59,89	59,89	34,81	35,24	34,99	35,16	35,27

Figura 2 - Visualizar Opções

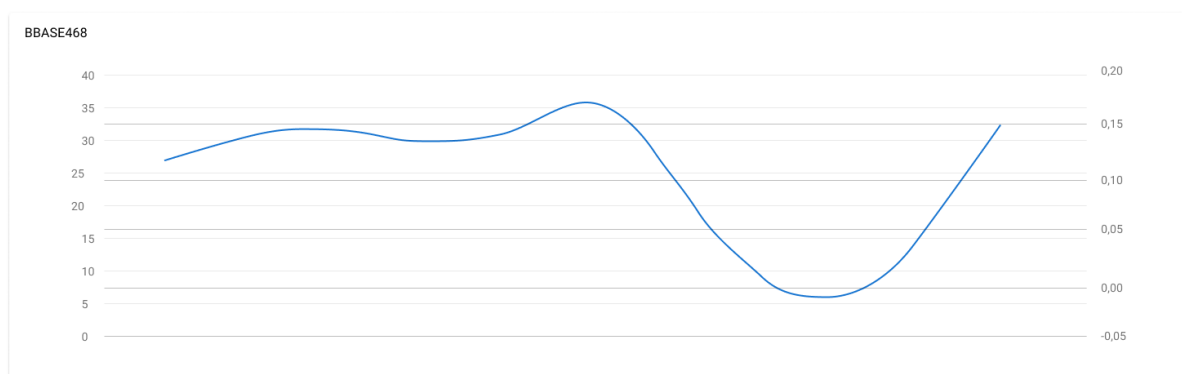


Figura 3 - Preço da Opção

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RFS03] Crawler de dados de registro de opções

Ator: Sistema

Este requisito começa quando o sistema diariamente de forma automática busca por registros de opções em **plataformas de cadastro na B3** para popular o banco de dados. As informações crawleadas serão referentes a **dados cadastrais**.



Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

5.2 RF02 - Manter histórico diário dos preços das Opções

As funcionalidades apresentadas nesta seção são os requisitos funcionais de sistema que permitirão ao sistema apresentar ao usuário o histórico dos preços das opções através de gráfico de linhas.

Os RF aqui declarados são provenientes do HU01.

O sistema web terá integração direta com o banco de dados. Os valores dos preços já estarão cadastrados no banco, podendo apenas serem consultados, não apresentado assim cadastro, alteração ou a sua exclusão.

[RFS04] Realizar Crawler de dados de transações

Ator: Sistema

Este requisito começa quando o sistema diariamente de forma automática busca por registros de Opções em **plataformas de cadastro na B3** para popular o banco de dados. As informações crawleadas serão referentes a **dados de séries históricas de negociações de opções na bolsa de valores**.

Prioridade: ☐ Essencial ☐ Importante ☒ Desejável

[RFS05] Gerar Relatório de cotação de opção

Ator: Usuário

O sistema permitirá a emissão de um relatório do histórico referente a cotação da Opção. Para isso será utilizado a opção selecionada a partir dos filtros do mecanismo de busca avançada RFS01.

O retorno desse relatório de cotação diária da Opção será um gráfico de linha referente ao histórico completo de preços dessa Opção. O eixo X do gráfico será a data da cotação, enquanto o eixo Y representará o valor da cota. Assim como mostrado abaixo:

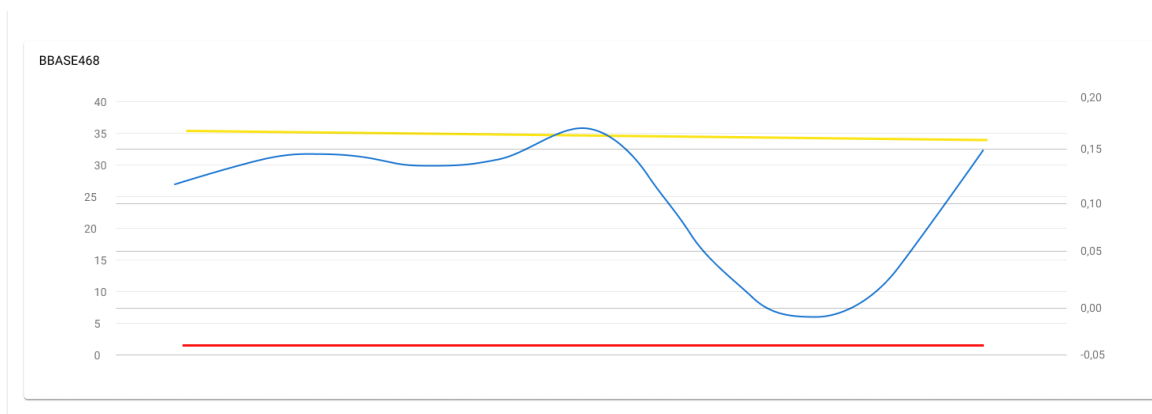
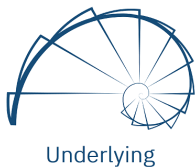


Figura 4 - Histórico de preços

Prioridade: ☐ Essencial ☐ Importante ☒ Desejável

5.3 RF03 - Manter Usuário

As funcionalidades apresentadas nesta seção são os requisitos funcionais de sistema que permitirão manter o usuário no sistema web e também realizar o seu login para ter acesso a todas as funcionalidades que o sistema proporciona.

Os RF aqui declarados são provenientes do HU04.

[RFS06] Inserir Usuário

Ator: Usuário

Este requisito funcional de sistema começa quando o Ator deseja fazer uso da plataforma e se cadastrar no sistema. Para inserir um Cliente o usuário deverá entrar com as informações presentes na tabela a seguir:

Nome do Campo	Descrição
*Nome	Nome completo
*CPF	Campo aberto
*E-mail	Validar Email

*Telefone/Celular	Formato (XX)XXXXX-XXXX
*Cidade	Campo de escolha fechada para selecionar a cidade
*Estado	Campo de escolha fechada para selecionar o Estado
*Senha	No mínimo 6 Caracteres, obrigatório número, caractere especial, letra maiúscula e minúscula. No Formato: XXXXXX

Tabela 2 - Dados do Usuário

(*) Todo atributo marcado com o símbolo de (*) é de preenchimento obrigatório

Importante: Para a realização do login no sistema será utilizado o atributo E-mail e Senha, ambos deveram ser verificáveis e únicos.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RFS07] Consultar Usuário

Ator: Usuário

Este requisito Funcional de sistema começa quando o ator deseja consultar o seu cadastro. Será possível navegar no cadastro próprio apenas.

Será retornado ao Usuário todas as informações referentes ao seu cadastro conforme tabela 02. Assim sendo serão listados o Nome do usuário, seguido do CPF, Telefone, Código, o CEP, Login e senha assim como no exemplo a seguir:

<<Nome do Usuário>>

<<CPF>>

<<Telefone>>

<<Código>

<<CEP1>>

<<Login>>

<<Senha>>

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável



[RFS08] Editar Usuário

Ator: Usuário

Este requisito funcional de sistema começa quando o ator deseja alterar alguns dos seus dados pessoais ou de login cadastrado no sistema. O requisito será acessado através da consulta ao usuário assim como propõe o [RFS07]. Na aba de configurações o usuário poderá alterar todos os atributos referentes aos seus dados que estão presentes na Tabela 01.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RFS09] Remover Usuário

Ator: Usuário

Este requisito funcional de sistema começa quando o usuário desejar se remover do sistema. A remoção do usuário deve ser feita a partir da consulta como propõe o [RFS07] na aba de configurações. A partir disso, será possível visualizar os dados do usuário e assim, remover.

A função de remoção do Cliente não deverá excluir integralmente do sistema, mas sim alterar o status de atividade dele, de ativo para inativo. Assim, o sistema poderá manter um registro histórico das informações.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

5.4 RF04 - Manter Métricas de investimento das opções

As funcionalidades apresentadas nesta seção são os requisitos funcionais de sistema que permitirão ao sistema apresentar as métricas filtradas pelos usuários.

Entre as métricas utilizadas citamos: Delta, Theta, Rho, Vega, Gama, Hedge Ratio caracterizadas abaixo:

Delta de uma Opção representa a mudança no preço da Opção em relação à alteração no preço da ação-objeto.

Gama é a taxa da variação do Delta em referência ao deslocamento do preço da Ação.

Theta mede a taxa de desvalorização de uma Opção ao longo do tempo, à medida que ela se aproxima da data de exercício.



Vega de uma Opção é a taxa da mudança do valor da Opção em relação a uma mudança na volatilidade.

Rho representa a variação no preço da Opção se houver uma mudança na taxa de juros.

O **Hedge Ratio** ou índice de hedge é resultante da quantidade do item protegido que a entidade efetivamente protege e a quantidade do instrumento de hedge que a entidade efetivamente utilizada para proteger essa quantidade do item protegido.

A seguir serão apresentados cada um dos requisitos funcionais de sistemas referente as métricas que são essenciais para a qualidade do sistema.

Os RF aqui declarados são provenientes do HU02.

[RFS10] Calcular e Apresentar o Valor Intrínseco

Ator: Usuário

Este requisito começa quando o ator desejar consultar o valor intrínseco. Será possível obter o valor intrínseco quando o ator selecionar uma Opção a partir do mecanismo de busca avançada descrito pelo RFS01

Após ser calculado pelo sistema será retornado ao Usuário o valor intrínseco da seguinte forma:

Valor Intrínseco:

<<Valor Numérico>>

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RFS11] Calcular e Apresentar o Valor no Tempo

Ator: Usuário

Este requisito começa quando o ator desejar consultar o valor no tempo. Será possível obter o valor no tempo quando o ator selecionar uma Opção a partir do mecanismo de busca avançada descrito pelo RFS01.

Após ser calculado pelo sistema será retornado ao Usuário o valor no tempo da seguinte forma:

Valor no Tempo:

<<valor numérico>>



Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RFS12] Calcular e Apresentar o Delta

Ator: Usuário

Este requisito começa quando o ator desejar consultar o Delta. Será possível obter o Delta quando o ator selecionar uma opção a partir do mecanismo de busca avançada descrito pelo RFS01.

O valor de Delta será calculado pelas seguintes fórmulas:

Definition

$$\text{delta} = \frac{\partial}{\partial F_0}$$

Call

$$e^{-rT} N(d_1) > 0$$

Put

$$-e^{-rT} N(-d_1) < 0$$

Após **ser calculado pelo sistema** será retornado ao Usuário o Delta da seguinte forma:

Delta:

<<valor numérico>>

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RFS13] Calcular e Apresentar o Theta

Ator: Usuário

Este requisito começa quando o ator desejar consultar o Theta. Será possível obter o Theta quando o ator selecionar uma Opção a partir do mecanismo de busca avançada descrito pelo RFS01.

O valor de Theta será calculado pelas seguintes fórmulas:

Definition

$$\text{theta} = \frac{\partial}{\partial (-T)}$$

Call

$$-\frac{e^{-rT} F_0 \sigma}{2\sqrt{2\pi T}} \exp(-d_1^2/2) + re^{-rT} [F_0 N(d_1) - KN(d_2)]$$

Put

$$-\frac{e^{-rT} F_0 \sigma}{2\sqrt{2\pi T}} \exp(-d_1^2/2) + re^{-rT} [KN(-d_2) - F_0 N(-d_1)]$$

Após ser calculado pelo sistema será retornado ao Usuário o Theta da seguinte forma:



Underlying



Theta:

<<valor numérico>>

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RFS14] Calcular e Apresentar o Rho

Ator: Usuário

Este requisito começa quando o ator deseja consultar o Rho. Será possível obter o Rho quando o ator selecionar uma Opção a partir do mecanismo de busca avançada descrito pelo RFS01.

O valor de Rho será calculado pelas seguintes fórmulas:

Definition	Call	Put
$\rho = \frac{\partial}{\partial r}$	$-Te^{-rT}[\mathcal{F}_0 N(d_1) - KN(d_2)] < 0$	$-Te^{-rT}[KN(-d_2) - \mathcal{F}_0 N(-d_1)] < 0,$

Após ser calculado pelo sistema será retornado ao Usuário o Rho da seguinte forma:

Rho:

<<valor numérico>>

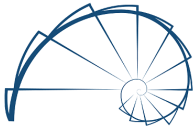
Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RFS15] Calcular e Apresentar o Vega

Ator: Usuário

Este requisito começa quando o ator deseja consultar o Vega. Será possível obter o Vega quando o ator selecionar uma Opção a partir do mecanismo de busca avançada descrito pelo RFS01.

O valor de Vega será calculado pelas seguintes fórmulas:



Underlying

Definition

$$\text{vega} = \frac{\partial}{\partial \sigma}$$

Call

$$\frac{e^{-rT} F_0 \sqrt{T}}{\sqrt{2\pi}} \exp(-d_1^2/2) > 0$$

Put

$$\frac{e^{-rT} F_0 \sqrt{T}}{\sqrt{2\pi}} \exp(-d_1^2/2) > 0$$

Após ser calculado pelo sistema será retornado ao Usuário o Vega da seguinte forma:

Vega:

<<valor numérico>>

Prioridade:



Essencial



Importante



Desejável

[RFS16] Calcular e Apresentar o Gama

Ator: Usuário

Este requisito começa quando o ator desejar consultar o Gama. Será possível obter o Gama quando o ator selecionar uma opção a partir do mecanismo de busca avançada descrito pelo RFS01.

O valor de Gama será calculado pelas seguintes fórmulas:

Definition

$$\text{gamma} = \frac{\partial^2}{\partial F_0^2}$$

Call

$$\frac{e^{-rT}}{F_0 \sigma \sqrt{2\pi T}} \exp(-d_1^2/2) > 0$$

Put

$$\frac{e^{-rT}}{F_0 \sigma \sqrt{2\pi T}} \exp(-d_1^2/2) > 0$$

Após ser calculado pelo sistema será retornado ao Usuário o Gama da seguinte forma:

Gama

<<valor numérico>>

Prioridade:



Essencial



Importante



Desejável



[RFS17] Calcular e Apresentar o índice Ratio

Ator: Usuário

Este requisito começa quando o ator deseja consultar o *Hedge Ratio*. Será possível obter o *Hedge Ratio* quando o ator selecionar uma opção a partir do mecanismo de busca avançada descrito pelo RFS01.

Após ser calculado pelo sistema será retornado ao Usuário o *Hedge Ratio* da seguinte forma:

Hedge Ratio

<<Valor_Indice>>

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

5.5 RF05 - Manter Estrutura de operação

As funcionalidades apresentadas nesta seção são os requisitos funcionais de sistema que permitirão ao sistema manter a estrutura de operação, onde o usuário poderá escolher uma ou mais Opções para compor a estrutura de operação.

Os RF aqui declarados são provenientes do HU06.

[RFS18] Inserir Estrutura de operação

Ator: Usuário

Este requisito funcional de sistema começa quando o Ator deseja adicionar uma nova estrutura de operação no sistema. Para inserir uma nova estrutura deverá entrar com as informações presentes na tabela a seguir:

Nome do Campo	Descrição
*Nome da estrutura	Nome de identificação estrutura
*ID	Campo de id único gerado por um uuid de identificação designado automaticamente.
*Opções	Conjunto de Opções reais ou fictícias mantidas pelo sistema

Tabela 3 - Dados da estrutura de operações



(*) Todo atributo marcado com o símbolo de (*) é de preenchimento obrigatório

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RFS19] Consultar Estrutura de operações

Ator: Usuário

Este requisito Funcional de sistema começa quando um dos atores deseja consultar por uma Estrutura de Operações. Será possível pesquisar por uma estrutura através do filtro dado pela Tabela 04 a seguir:

Filtro	Descrição
Nome da estrutura	Nome de identificação da estrutura

Tabela 4 - Filtro da Estrutura de Operações

A estrutura de operação poderá então ser consultada pelo filtro Nome e deverá ser retornado ao usuário todas as informações referentes a estrutura. Assim sendo serão listados o Nome da estrutura, seguido do número e a seguir será listado todas as opções pertencentes a estrutura buscada, assim como no exemplo a seguir:

<<Nome da Estrutura >> <<Numero>>

<Opção 01>

<Opção 02>

<Opção 03>

<Opção 04>

<Opção 05>

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RFS20] Editar Estrutura de Operação

Ator: Usuários



Underlying



Este requisito funcional de sistema começa quando o ator deseja alterar a estrutura de operação no sistema. O requisito será acessado através da consulta a estrutura assim como propõe o [RFS19]. A partir disso, será possível visualizar a estrutura e assim, alterá-la.

Todos os atributos da Tabela 03 – Dados da estrutura de operações podem ser alterados, **exceto** o número de identificação.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RFS21] Remover Estrutura de Operação

Ator: Usuário

Este requisito funcional de sistema começa quando o Ator deseja remover a Estrutura de operações do sistema. A remoção da Estrutura deve ser feita a partir da consulta como propõe o [RFS19]. A partir disso, será possível visualizar a estrutura e assim, removê-la.

A função de remoção da Estrutura não deverá excluir integralmente do sistema, mas sim alterar o status de atividade, de ativo para inativa. Assim, o sistema poderá manter um registro histórico das informações.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

5.6 RF06 - Compartilhar da Estrutura de operação

As funcionalidades apresentadas nesta seção são os requisitos funcionais de sistema que permitirão ao sistema realizar o compartilhamento da estrutura de operações com os demais usuários do sistema.

Os RF aqui declarados são provenientes do HU10.

[RFS22] Compartilhar Estrutura

Ator: Usuário

Este requisito funcional de sistema começa quando o Ator deseja compartilhar a Estrutura de operações do sistema. O Compartilhamento da Estrutura deve ser feita a partir da consulta como propõe o [RFS19]. A partir disso, será possível visualizar a estrutura e assim, compartilhá-la.

Prioridade: ☐ Essencial ☐ Importante ☒ Desejável

[RFS23] Consultar Estrutura Compartilhada

Ator: Usuário

Este requisito Funcional de sistema começa quando um dos atores deseja consultar por alguma Estrutura compartilhadas.

Essa consulta não possui filtros, todas as estruturas de operações compartilhadas serão visíveis em uma página única e específica e serão ordenadas por data de publicação decrescente.

A página deverá apresentar ao usuário todas as informações referentes a estrutura. Assim sendo serão listados o Nome da estrutura, seguido do número e a seguir será listado todas as opções pertencentes a estrutura buscada, assim como apresentado no exemplo a seguir:



Figura 5 - Estrutura Compartilhada

Prioridade: ☐ Essencial ☐ Importante ☒ Desejável



Underlying



5.7 RF07 – Gerar Payoffs de Operação

As funcionalidades apresentadas nesta seção são os requisitos funcionais de sistema que permitirão ao sistema apresentar de forma gráfica a estrutura de operação.

Os RF aqui declarados são provenientes do HU03 e HU06.

[RFS24] Calcular e apresentar Payoff

Ator: Usuário

Este requisito funcional de sistema começa quando o Ator deseja visualizar a Estrutura de operações do sistema. Será possível obter o *Payoff* durante a criação das estruturas de Opções, sendo esse atributo atualizado toda vez que o usuário altera, adiciona ou remove uma ou mais opções de sua estrutura.

Após ser calculado pelo sistema será retornado ao Usuário o *Payoff* representado pelo RFS25 na forma de gráfico de linhas.

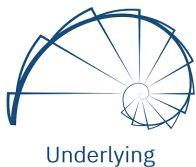
Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RFS25] - Gerar gráfico Payoff

Ator: Usuário

O sistema permitirá a emissão de um relatório do Payoff da estrutura de operação. Para isso será necessário o preenchimento de algum dos atributos referentes a opção, um gráfico de payoff pode ser gerado a partir de uma ou mais opções.

O retorno desse relatório de Payoff da estrutura de operação será um gráfico de linhas. O eixo X do gráfico representa o preço do ativo base na expiração, enquanto o eixo Y representará o lucro referente à operação. Assim como mostrado abaixo:



Underlying



BBDC4 - Estratégia 3 - 25/04/2022

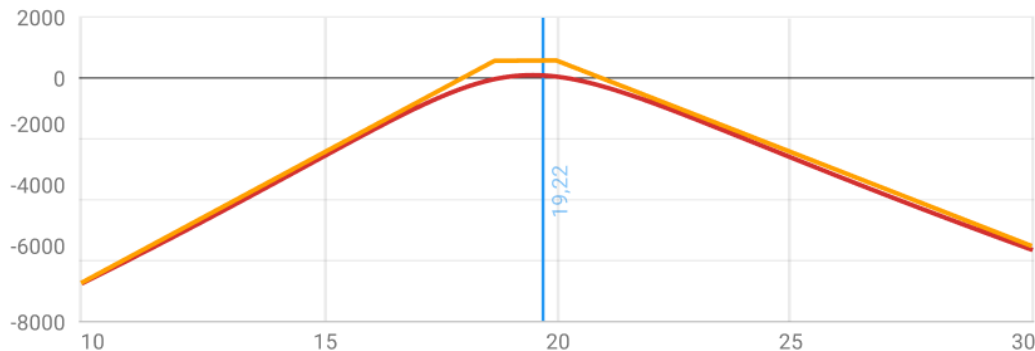


Figura 6 - Payoff : Preço x Lucro

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

5.8 RF08 – Manter Ações

As funcionalidades apresentadas nesta seção são os requisitos funcionais de sistema que permitirão ao sistema consultar as possíveis Ações existentes.

O sistema web terá integração direta com o banco de dados. Todos as ações já estarão cadastrados no banco, podendo apenas serem consultados, não apresentado assim cadastro, alteração ou a sua exclusão.

Os RF aqui declarados são provenientes do HU08.

[RFS26] Consultar Ações

Ator: Usuário

Este requisito começa quando um o ator desejar consultar as ações do banco de dados. Será possível pesquisar pelas ações, através do filtro ticker.

Será retornado então apenas a ação juntamente com seu valor.

<<Nome da Ação >> <<Valor>>



Por default caso o filtro não seja preenchido (nulo) serão então listados todos os Ativos em ordem Alfabética Crescente conforme Nome do Ação seguindo do valor.

<<Nome da Ação1>><<valor1>>

<<Nome da Ação2>><<valor2>>

<<Nome da Ação3>><<valor3>>

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

[RFS27] Selecionar Ações

Ator: Usuário

Este requisito funcional de sistema começa quando o ator deseja Selecionar a Ação. O requisito será acessado através da consulta a Ação assim como propõe o [RFS25]. A partir disso, será possível visualizar a ação e assim, seleciona-la.

<<Nome da Ação>> <<Valor>> << [x] Selecionar >>

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

5.9 RF09 – Manter Grade de Opções

As funcionalidades apresentadas nesta seção são os requisitos funcionais de sistema que permitirão ao sistema, a partir da seleção de uma Ação pelo usuário, visualizar a grade de Opções da Ação em um determinado período.

Os RF aqui declarados são provenientes do HU08.

[RFS28] Gerar Grade de Opções

Ator: Usuário

O sistema permitirá a emissão de um relatório de grade de Opções. Para isso será necessário o preenchimento de algum dos filtros destacados na tabela 05.



Filtros	Descrição
Ativo Base	Ticker do ativo base que a opção se refere
Tipo	Campo fechado que pode obter os valores de CALL, PUT ou Todas
Data Inicial	Data Inicial na qual o relatório se baseará para coletar as informações no formato DD/MM/AAAA
Data Final	Data Final na qual o relatório se baseará para coletar as informações no formato DD/MM/AAAA

Tabela 5 - Filtros do relatório de grade de Opções.

Caso nenhum dos filtros seja preenchido, o sistema utilizará por default o período de **um ano a partir do dia atual**.

O retorno desse relatório de grade de Opções listará todas a Opções referentes ao ativo base no período filtrado assim pode ser observado a seguir:

Grade de Opções - BPAC11								
Ticker	Tipo	Strike	Cota	Vol.	Delta	Gamma	Theta	Vencimento
BPAC288	CALL	0,01	0,01	0,15	0,15	2	3,4	20/02/2023
BPACQ246	PUT	0,12	0,10	0,01	0,16	25	2,4	12/07/2023
BPACA241	PUT	0,01	0,01	0,01	0,15	2	6,4	31/03/2023
BPACC245	CALL	0,12	0,10	0,15	0,16	25	7,4	03/09/2023
BPAC288	CALL	0,01	0,01	0,15	0,15	2	3,4	20/02/2023
BPACQ246	PUT	0,12	0,10	0,01	0,16	25	2,4	12/07/2023
BPACA241	PUT	0,01	0,01	0,01	0,15	2	6,4	31/03/2023
BPACC245	CALL	0,12	0,10	0,15	0,16	25	7,4	03/09/2023
BPACQ246	PUT	0,12	0,10	0,01	0,16	25	2,4	12/07/2023
BPACA241	PUT	0,01	0,01	0,01	0,15	2	6,4	31/03/2023
BPACC245	CALL	0,12	0,10	0,15	0,16	25	7,4	03/09/2023

Figura 7 - Grade de Opções

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

5.10 RF10 – Manter Opções Fictícias



As funcionalidades apresentadas nesta seção são os requisitos funcionais de sistema que permitirão ao sistema criar Opções Fictícias a partir de parâmetro base. Que poderão futuramente compor uma estrutura de operação. Os parâmetros utilizados para a criação de uma opção fictícia serão: nome, tipo da opção, tipo da transação, número de contratos, preço do ativo base, preço da Opção e preço de strike.

Os RF aqui declarados são provenientes do HU07 e HU09.

[RFS29] Adicionar Opção Fictícia na Estrutura

Ator: Usuário

Este requisito funcional de sistema começa quando o Ator deseja adicionar uma Opção Fictícia na estrutura de operação. Para inserir uma nova Opção deverá entrar com as informações presentes na tabela 06 a seguir.

Filtro	Descrição
*Nome da Opção	Nome de identificação da Opção
*Tipo da Opção	Multivalorado fechado: <ul style="list-style-type: none">• PUT• CALL• TODAS
Tipo da transação	Campo fechado. Valor Long ou Short
Número de contratos	Valor inteiro
Preço do ativo base	Valor ponto flutuante
Preço da Opção	Valor ponto flutuante
Preço de Strike	Valor ponto flutuante

Tabela 6 - Dados da Opção Fictícia

(*) Todo atributo marcado com o símbolo de (*) é de preenchimento obrigatório

Prioridade: ☐ Essencial ☐ Importante ☒ Desejável

[RFS30] Editar Opção Fictícia

Ator: Usuário



Este requisito funcional de sistema começa quando o ator deseja alterar a Opção Fictícia já adicionada na estrutura de operação. O requisito será acessado através da página de criação e edição de estrutura de operação propõe os [RFS18 e RFS19].

Todos os atributos da Tabela 06 – Dados da Opção Fictícia podem ser alterados.

Prioridade: ☐ Essencial ☐ Importante ☒ Desejável

[RFS31] Remover Opção Fictícia

Ator: Usuário

Este requisito funcional de sistema começa quando o Ator deseja remover uma Opção Fictícia registrada na estrutura de operação. A remoção da Opção deve ser feita a partir da criação e edição da estrutura de operação como propõe os [RFS18 e RFS19]. A partir disso, será possível visualizar a Opção Fictícia e assim removê-la.

A função de remoção desta Opção deverá excluir integralmente a opção da estrutura, não mantendo registro de opções excluídas da operação.

Prioridade: ☐ Essencial ☐ Importante ☒ Desejável

6. REQUISITOS NÃO FUNCIONAIS

Esta seção apresenta os requisitos não funcionais do sistema. Os requisitos não-funcionais são os requisitos relacionados ao uso da aplicação em termos de desempenho, usabilidade, confiabilidade, segurança, disponibilidade, manutenibilidade e tecnologias envolvidas. OS RNF permitem avaliar aspectos INTERNOS da qualidade do software que normalmente só são percebidos pela equipe técnica (ex. facilidade de manutenção, teste, evolução do software) e aspectos EXTERNOS da qualidade do software que podem ser avaliados pela equipe de desenvolvimento do ponto de vista do usuário (ex. eficiência). Abaixo Serão apresentados os RNF deste projeto subdivididos em Produto, organizacionais e Externos.

6.1 RNF do Produto

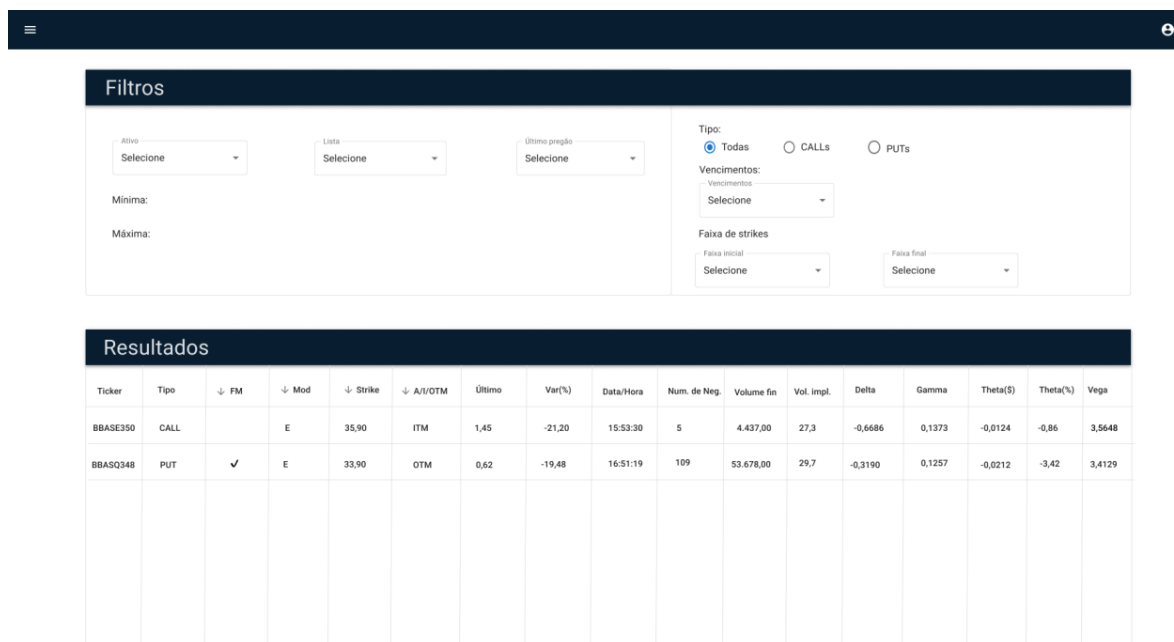
Esta seção apresenta os requisitos não funcionais do sistema Requisitos que especificam o comportamento do produto como Usabilidade, Confiabilidade, Desempenho, Segurança, entre outros. A seguir serão apresentados o RNF agrupados por características/atributos da qualidade.

6.1.1. Usabilidade

Para a facilidade de compreensão e utilização do sistema é preciso oferecer características às atividades desempenhadas pelo sistema. Esta seção descreve os requisitos não funcionais associados à facilidade de uso da interface com o usuário.

[RNF01] - Padronizar Interfaces

O sistema deve apresentar um layout com uma interface de fácil aprendizado e navegabilidade com ícones e menus intuitivos. Todas as telas/páginas do sistema seguirá uma interface padrão na cor azul escura e branca, assim como pode ser observado abaixo:



The screenshot displays a web application interface for financial data. It features a dark blue header with a menu icon on the left and a user profile icon on the right. Below the header, there is a 'Filtros' (Filters) section with several dropdown menus for 'Ativo', 'Lista', and 'Último preço'. To the right of these are radio buttons for 'Tipo' (Todas, CALLs, PUTs) and a 'Vencimentos' dropdown. Below these are input fields for 'Faixa de strikes' (Faixa inicial and Faixa final). The 'Resultados' section below contains a table with 16 columns: Ticker, Tipo, ↓ FM, ↓ Mod, ↓ Strike, ↓ A/I/OTM, Último, Var(%), Data/Hora, Num. de Neg., Volume fin, Vol. Impl., Delta, Gamma, Theta(\$), Theta(%), and Vega. Two rows of data are visible, corresponding to tickers BBASE350 and BBASQ348.

Ticker	Tipo	↓ FM	↓ Mod	↓ Strike	↓ A/I/OTM	Último	Var(%)	Data/Hora	Num. de Neg.	Volume fin	Vol. Impl.	Delta	Gamma	Theta(\$)	Theta(%)	Vega
BBASE350	CALL		E	35,90	ITM	1,45	-21,20	15:53:30	5	4.437,00	27,3	-0,6686	0,1373	-0,0124	-0,86	3,5648
BBASQ348	PUT	✓	E	33,90	OTM	0,62	-19,48	16:51:19	109	53.678,00	29,7	-0,3190	0,1257	-0,0212	-3,42	3,4129

Figura 8 - Padronização de Interfaces



Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RNF02] - Legendar Gráficos

Todos os mapas e gráficos devem apresentar legendas para os elementos para que seja possível identificá-los de maneira rápida e fácil.

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

[RNF03] - Validar Dados de Entrada

Todos os campos de input de usuário devem conter validação e/ou máscara de input, para evitar que o usuário insira dados inconsistentes.

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

[RNF04] – Abrir novas abas para links externos

Todos links existentes no sistema que sejam de um domínio diferente deveram ser abertos em uma nova aba.

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

[RNF05] – Utilizar Responsividade

O sistema deve ser responsivo e para isso dever ser apresentado em sua **largura** em pixels, com o mínimo de pixels pro layout de notebook e monitores de 992px e no mínimo do layout de celular para 388px.



Prioridade: ☐ Essencial ☒ Importante ☐ Desejável

[RNF06] – Incluir Opções de Alteração Visual

O sistema deve ser adaptável para os usuários que sofrem algum grau de deficiência visual com a opção de algumas mudanças como: Contraste, Cor de background, tamanho de fonte e leitura de tela.

Prioridade: ☐ Essencial ☐ Importante ☒ Desejável

[RNF07] – Incluir Opção de Alteração de Língua

O sistema deverá permitir a alteração de língua portuguesa para inglesa e vice-versa

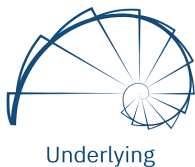
Prioridade: ☐ Essencial ☐ Importante ☒ Desejável

6.1.2. Segurança

Os requisitos de segurança correspondem às definições sobre as regras de segurança para criação de usuários, os procedimentos exigidos para a utilização de senhas, a necessidade de criptografia e demais questões relacionadas para garantir a proteção dos dados. Esta seção descreve os requisitos não funcionais associados à integridade, privacidade e autenticidade dos dados do produto.

[RNF08] - Criptografar Senhas e dados dos usuários

O sistema deverá criptografar as **senhas** do banco de dados e **todos os dados que permitam identificar um usuário do sistema** utilizando a criptografia padrão do serviço de gerenciamento de usuários da AWS (Cognito). Assim, os dados dos usuários estarão mais seguros mesmo que subtraídos.



Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RNF09] – Verificar Autenticidade

O sistema deverá identificar a fonte das informações quando solicitado uma alteração nos dados do usuário. Quando a fonte for desconhecida uma mensagem de erro deve ser fornecida para a alteração solicitada.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RNF10] – Realizar Logout

O sistema deverá armazenar as informações de login do usuário por até 3 horas, após isso deve realizar a limpeza das informações do usuário do navegador (cookies) e assim sair do site automaticamente.

Prioridade: ☐ Essencial ☐ Importante ☒ Desejável

[RNF11] – Permitir Apenas Senhas Fortes

O sistema deverá apresentar uma política de senhas onde deverá avaliar durante o cadastro de usuários as senhas determinadas pelo usuário, negando senhas que não atendam as condições mínimas de segurança (baseado no NIST 800-23) ou que pertençam a uma lista de senhas padrão (weak password check).

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável



[RNF12] – Autenticar Usuário

O sistema deverá utilizar o sistema AWS Amazon Cognito para gerenciamento de usuário, uma vez que todo o sistema e suas APIs deverão ser acessadas com usuário autenticado, salvo Endpoints de autenticação e cadastro.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

6.1.3. Confiabilidade

Os requisitos de confiabilidade se referem às definições sobre o que fazer para garantir a consistência das informações caso ocorra a indisponibilidade do sistema por queda de energia e outras semelhantes. Esta seção descreve os requisitos não funcionais de produto associados à severidade de falhas, habilidade de recuperação das mesmas, as ações que podem ser desfeitas, bem como à corretude do produto.

[RNF13] – Desfazer Ação de Exclusão

O sistema deverá dar a opção de cancelar alguma exclusão caso seja requerido pelo usuário.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RNF14] – Desfazer Ação de Compartilhamento

O sistema deverá dar a opção de desfazer o compartilhamento caso seja requerido pelo usuário.

Prioridade: ☐ Essencial ☒ Importante ☐ Desejável



[RNF15] – Interromper Solicitação

O sistema deverá desfazer qualquer ação iniciada pelo usuário quando houver persistência da ausência de rede (Internet) ou quando a energia cair bruscamente, e realizar assim o logout do usuário.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RNF16] – Apresentar Disponibilidade (99%)

O sistema deverá dar acesso aos seus usuários independente do horário de requisições e apresentar disponibilidade 99% com a utilização dos servidores EC2 da AWS Amazon, indicando assim que o sistema não deve exceder 52,56 minutos de indisponibilidade em 365 dias.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RNF17] – Apresentar Informações Reais, Atuais e Consistentes com o mercado de Opções

O sistema deverá apresentar dados reais, atuais e consistentes oriundos diretamente da B3. Uma vez que o mesmo deve ser um espelho confiável das opções de mercado em tempo real. Para isso o sistema deverá realizar Crawler diretamente da B3 para obter os valores, índices e realizar os cálculos necessários que foram solicitados pelo usuário.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável



[RNF18] – Recuperar de Falhas

O sistema deverá mediante a oscilação de rede/Internet não realizar o logout. Deverá continuar operante e realiza o processamento das ações solicitadas pelo usuário.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

6.1.4.Desempenho / Eficiência

Os requisitos de desempenho são voltados às necessidades de infraestrutura para garantir que o sistema funcione sem lentidão, sem problemas por falta de espaço em disco ou com outras ocorrências que impactem na qualidade de uso do sistema. Esta seção descreve os requisitos não funcionais associados à eficiência, uso de recursos e tempo de resposta do produto.

[RNF19] – Apresentar Tempo de Resposta Baixo (2S)

O tempo de resposta que o sistema deve responder a uma consulta/demanda do usuário não deverá ser maior que 2 segundos.

Prioridade: ☐ Essencial ☐ Importante ☒ Desejável

[RNF20] – Apresentar Tempo de Throughput Baixo (1S)

O tempo de processamento do sistema não deverá ser maior que 1 segundo.

Prioridade: ☐ Essencial ☐ Importante ☒ Desejável



[RNF21] – Realizar Escalabilidade

O sistema deverá se auto escalar utilizando a AWS Auto Scaling que ao receber milhares de solicitações de usuários manter a sua disponibilidade dos serviços.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

6.1.5. Compatibilidade

Nesse requisito, são especificadas quais as compatibilidades necessárias para a execução do sistema. Logo, podem fazer parte dessa solicitação a compatibilidade com navegadores, em quais versões do sistema operacional o sistema é capaz de rodar.

[RNF22] – Acessar em Diversos Navegadores

O sistema deverá ser compatível com a maioria dos navegadores modernos, que suportam aplicações em React, a fim de atingir o maior número de usuários possível como o Edge, Chrome, Firefox, Opera, Safari.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

6.2 RNF Adicionais

Os requisitos Adicionais referem-se aos requisitos solicitados pelo cliente do sistema após o início do projeto, ou seja, aqueles requisitos solicitados fora o escopo e durante a validação das entregas de cada etapa do projeto.



[RNF23] – Adicionar Glossário [Usabilidade]

O sistema deve apresentar um glossário contendo palavras, termos e expressões que são geralmente pouco conhecidas pelos usuários iniciantes, principalmente por representarem conceitos técnicos e complexos do mercado de opções.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

6.3 RNF Organizacionais

O sistema deve ser desenvolvido de acordo com as políticas e definições da empresa para garantir que o produto gerado esteja em conformidade com as normas empresariais. Esta seção apresenta os requisitos não funcionais do sistema que são decorrentes de políticas e procedimentos corporativos para garantir a qualidade de desenvolvimento do projeto.

[RNF24] – Método Organizacional

O desenvolvimento do sistema será realizado utilizando o método ágil SCRUM. Serão realizadas sprints mensais com reuniões de acompanhamento semanalmente.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RNF25] – Repositório

Para o controle de versão todas os documentos (Artefatos) produzidos e alterados nos processos referentes a este projeto deverão ser armazenados no repositório do Git em:

<https://github.com/lucasblazzi/underlying-project>



Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RNF26] – Tecnologias a serem utilizadas

Para o controle de atividade, definição de sprints e o gerenciamento do projeto será utilizada a ferramenta Jira. Para o desenvolvimento deverá utilizar as ferramentas da Amazon Web Services. Assim para melhor escalabilidade e manutenibilidade deverá ser utilizado o CloudFormation. Para o armazenamento de dados volumosos deve-se utilizar o Storage (S3) e o NoSQL para armazenar os dados dos clientes. As linguagens a serem utilizadas são Python e NodeJS. O frontend deve utilizar o React e fazer uso da biblioteca MaterialUI.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RNF27] – Tempo de Entrega da Tarefa

Todas as tarefas referentes ao sistema deverão ser entregues pela equipe de desenvolvimento no prazo previamente estipulado (Jira). Caso não ocorra, o responsável deve reportar a equipe com antecedência de 5 dias.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RNF28] – Entrega de Artefatos

As entregas dos artefatos atualizadas do sistema devem ser realizadas no final de cada sprint.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável



[RNF29] – Armazenamento de Dados

Com a volatilidade do mercado, e grande variação dos valores do mercado de Opções é necessária a constante atualização e busca de dados na rede, sendo essencial o espaço em disco o suficiente para armazenar dados diversas vezes ao dia. Deverá ser utilizado o Sistema AWS que apresenta uma alta capacidade de armazenamento.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RNF30] – Alteração nos Requisitos

Sempre que seja necessária a introdução de alterações em relação aos requisitos descritos neste documento ou a inclusão de novos requisitos, os seguintes itens devem ser seguidos:

1. Toda solicitação de mudança de requisitos deverá ser documentada pelo requisitante e enviada via e-mail aos integrantes da equipe
2. A equipe toda deverá avaliar o impacto da mudança no cronograma e no custo do serviço e submeter a votação para a aprovação. Uma vez que a mudança ou inserção de mais requisitos implicará na alteração subsequente de todos os documentos, além da codificação prorrogando consequentemente o prazo e custo.
3. Após aprovação da alteração e inserção a alteração pela maioria da equipe. Os responsáveis por cada artefato de software deverão realizar imediatamente a nova alteração/inserção.
4. Os integrantes da equipe deverão fazer inspeção destas novas alterações/inserções. Caso aja falhas deverá então montar um relatório de falhas e repassar ao responsável do artefato em questão, para que haja correção do mesmo

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

6.4 RNF Externos

Os requisitos externos são aqueles que estão relacionados a qualquer tipo de agente externo ao software. Ou seja, qualquer aspecto não relacionado diretamente com o produto,



mas que pode impactar no seu funcionamento deve ser definido. Esta seção apresenta os requisitos não funcionais do sistema que são decorrentes de fatores externos ao sistema e ao processo de desenvolvimento.

[RNF31] – Lei Geral de Proteção de Dados Pessoais

O sistema deve implementar as disposições de uso dos dados e privacidade dos usuários, tal como estabelecido na lei 13.709 ou LGPD (Lei Geral de Proteção de Dados Pessoais).

Prioridade: ☐ Essencial ☐ Importante ☒ Desejável

[RNF32] – Integração com Sistemas Externos:

O sistema diariamente de forma automática busca por registros de opções em plataformas de cadastro da B3 para popular o banco de dados.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

[RNF33] – Ético

O sistema não apresentará aos usuários quaisquer dados de cunho privativo.

Prioridade: ☒ Essencial ☐ Importante ☐ Desejável

7. BACKLOG DO PRODUTO

A seguir será apresentado em planilha o Backlog do produto realizado no Jira. Para mais detalhes verificar o documento do Cronograma com Jira.



Underlying

ID HU	ID RF	ID RFS	ID RNF	Tarefa	Esforço (story points)	Sprint Planejada
HU01	RF01	RFS03	RNF16 ao RNF20 RNF29 RNF32	Crawler de registro de Opções	10	Sprint 01
HU01	RF01	RFS04	RNF16 ao RNF20 RNF29 RNF32	Crawler de preços de Opções	10	Sprint 01
HU02	RF04	RFS10 a RFS17	RNF17 RNF19 RNF20	Cálculo de greeks de opções	8	Sprint 01
HU03	RF01 e RF02	RFS01, RFS02 e RFS05	RNF01 RNF03 RNF05 RNF16 RNF18 RNF21 RNF22	Página de registro de Opção	10	Sprint 01
HU03	RF07	RFS25	RNF02 RNF19 RNF20	Gráfico de payoff	6	Sprint 01
HU03	RF07	RFS24	RNF17 RNF19 RNF20	Cálculo de payoff de Opções	6	Sprint 01



Underlying



HU02	RF04	RFS10 a RFS17	RNF01 RNF05 RNF16 ao RNF22	Tabela de greeks	3	Sprint 01
HU02	RF04	RFS10 a RFS17	RNF17 RNF19 RNF20	Cálculo de métricas de desempenho de Opções	4	Sprint 01
HU02	RF04	RFS10 a RFS17	RNF01 RNF05 RNF16 ao RNF22	Tabela de métricas	3	Sprint 01
HU04	RF03	RFS06 a RFS09	RNF03 RNF09 RNF10 RNF12 RNF22	Autenticação de usuário	3	Sprint 02
HU04	RF03	RFS06 a RFS09	RNF03 RNF08 RNF09 RNF11 RNF29 RNF31	Registro de usuário	3	Sprint 02
HU04	RF03	RFS06 a RFS09	RNF01 RNF05 RNF16 RNF18 RNF21	Tela de autenticação	5	Sprint 02



Underlying



			RNF22			
HU04	RF03	RFS06 a RFS09	RNF01 RNF05 RNF16 RNF18 RNF21 RNF22	Tela de registro de usuário	4	Sprint 02
HU04	RF03	RFS06 a RFS09	RNF01 RNF05 RNF06 RNF07 RNF16 RNF18 RNF21 RNF22	Tela de perfil de usuário	6	Sprint 02
HU05	RF01	RFS01 RFS02	RNF03 RNF15 RNF17 RNF19 RNF20 RNF22	Busca avançada de Opções por filtros	7	Sprint 03
HU05	RF01	RFS01 RFS02	RNF01 RNF03 RNF05 RNF16 RNF21 RNF 22	Formulário de filtros para busca	6	Sprint 03
HU06	RF05	RFS18 a	RNF01 RNF03	Tela de estrutura de operação	6	Sprint 02



Underlying



		RFS22	RNF05 RNF16 RNF18 RNF21 RNF22			
HU06	RF05	RFS18 a RFS22	RNF03 RNF15 RNF19 RNF20 RNF29	Criação de estrutura de operação	3	Sprint 02
HU06	RF07	RFS25	RNF01 RNF02 RNF05 RNF16 ao RNF22	Payoff de estrutura de operação	4	Sprint 02
HU07	RF10	RFS29 a RFS31	RNF03 RNF15 RNF19 RNF20 RNF29	Criação de Opção Fictícia	8	Sprint 02
HU07	RF10	RFS29 a RFS31	RNF01 RNF03 RNF05 RNF16 RNF 22	Formulário para criação de Opção Fictícia	5	Sprint 02
HU08	RF08	RFS26	RNF16 ao RNF20	Crawler de registro de ações	10	Sprint 03



Underlying



			RNF29 RNF32			
HU08	RF08	RFS27	RNF03 RNF15 RNF17 RNF19 RNF20 RNF22	Busca de ação por ticker	4	Sprint 03
HU09	RF09	RFS28	RNF03 RNF15 RNF19 RNF20 RNF29	Geração de grade de Opções por ticker	8	Sprint 03
HU09	RF09	RFS28	RNF01 RNF05 RNF16 RNF18 RNF21 RNF22	Tela de grade de Opções	9	Sprint 03
HU09	RF04	RFS10 a RFS17	RNF01 RNF05 RNF16 ao RNF22	Tabela de métricas de mercado e greeks	6	Sprint 03
HU08	RF08	RFS26	RNF01 RNF05 RNF16 ao RNF 22	Exibição de dados de ações	4	Sprint 03

HU10	RF06	RFS22	RNF13 RNF14 RNF15	Compartilhamento de estruturas de operações	6	Sprint 02
HU10	RF06	RFS23	RNF01 RNF05 RNF16 ao RNF 22	Listagem de estruturas de operações compartilhadas	6	Sprint 02
HU10	RF06	RFS23	RNF01 RNF04 RNF05 RNF16 ao RNF 22 RNF 23	Tela de estrutura de operações compartilhadas	9	Sprint 02

Tabela 7 - BACKLOG DO PRODUTO

8. BACKLOG DA SPRINT 01

A seguir será apresentado em planilha o Backlog da Sprint 01 realizado no Jira. Para mais detalhes verificar o documento do Cronograma com Jira.

ID HU	ID RF	ID RFS	ID RNF	Tarefa	Esforço	Responsável	Entrega	Esforço Real
HU01	RF01	RFS03	RNF16 ao RNF20 RNF29 RNF32	Crawler de registro de opções	7	Lucas Tiense Blazzi	17-05-2022	10



Underlying

HU01	RF01	RFS04	RNF16 ao RNF20 RNF29 RNF32	Crawler de registro de opções	10	Lucas Tiense Blazzi	17-05-2022	8
HU02	RF04	RFS10 a RFS17	RNF17 RNF19 RNF20	Cálculo de greek de opções	8	IVAN LEONI VILAS BOAS	20-05-2022	6
HU03	RF01 e RF02	RFS01, RFS02 e RFS05	RNF01 RNF03 RNF05 RNF16 RNF18 RNF21 RNF22	Página de registro de opção	10	Leo Sousa	27-05-2022	11
HU03	RF07	RFS25	RNF02 RNF19 RNF20	Gráfico de payoff	6	Wesley Alexandre	23-05-2022	4
HU03	RF07	RFS24	RNF17 RNF19 RNF20	Cálculo de payoff de opções	6	Thiago Marcelo Passos	24-05-2022	7
HU02	RF04	RFS10 a RFS17	RNF01 RNF05 RNF16 ao RNF22	Tabela de greek	3	Thiago Marcelo Passos	25-05-2022	3
HU02	RF04	RFS10 a RFS17	RNF17 RNF19 RNF20	Cálculo de métricas de desempen ho de	4	Lucas Tiense Blazzi	23-05-2022	4

				opções				
HU02	RF04	RFS10 a RFS17	RNF01 RNF05 RNF16 ao RNF22	Tabela de métricas	3	Wesley Alexandre	26-05-2022	2

Tabela 8 - BACKLOG DA SPRINT 01

8.1 Velocidade da Equipe:

Durante o desenvolvimento e finalização da sprint 1 foram percebidos alguns problemas que impactaram na finalização de todos os objetivos propostos durante o planejamento.

No planejamento a Sprint 1 possuía um total de 60 story points incluindo os processos de testes e deployment de todas as funcionalidades propostas. Ao fim da sprint o total de story points reais mapeados foram **62, sem a inclusão dos testes unitários** da aplicação. Sendo assim, alguns dados de velocidade levantados foram:

- Velocidade planejada: 60 story points (com testes de implementação)
- Velocidade real: 62 story points (sem testes de implementação)
- Tempo médio de uma tarefa: 5 dias e 14 horas
- Tempo mediano de uma tarefa: 5 dias
- Maior tempo de uma tarefa: 2 semanas, 6 dias e 20 horas

Sendo o controle da velocidade podendo ser acompanhado no gráfico de controle abaixo:

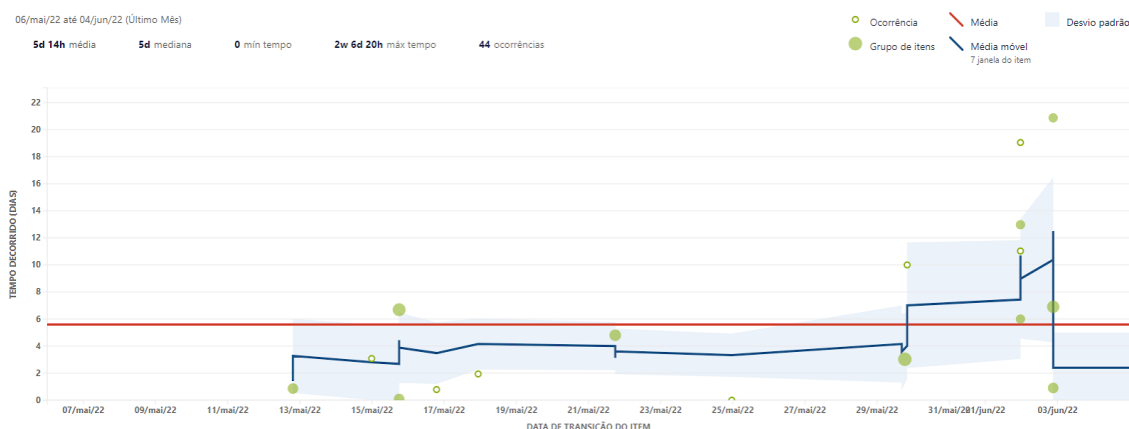


Figura 9 - Gráfico de controle Sprint 01

A partir dos resultados obtidos foi possível concluir que **ocorreu um erro de estimativa ocasionados pela dificuldade no processo de realização de testes e por mudança em recurso arquitetura para melhora de performance.**

No caso dos testes, a dificuldade ocorreu dado a maior complexidade de implementação de testes unitários voltados para ambientes nuvem como a AWS, necessitando de um conhecimento de ferramentas mais específicas para realizar o mock dos recursos da nuvem e testá-los de modo que não afete os recursos reais. Além disso, a falta de conhecimento do framework proposto para os testes do frontend impactou a entrega da atividade pela necessidade de estudo da ferramenta. No caso da arquitetura, foi planejado a implementação de queries SQL utilizando o Athena para a realização de buscas de opções, no entanto, após o desenvolvimento da api de busca e do pipeline necessário para o Athena, foi constatado que a velocidade de resposta das queries não foi satisfatória, necessitando uma revisão do recurso de busca. Com pesquisa encontrou-se a alta performance das Search Engines, que são ferramentas específicas para buscas em objetos complexo, assim, foi adotado o Elasticsearch para esse objetivo, sendo que o não conhecimento da ferramenta aumentou o tempo gasto na atividade.

8.2 Retrospectiva da Equipe do Sprint 1

A seguir será apresentado o que deu certo na execução da sprint 01, na sequência o que não deu certo, e, por fim, quais ações deverão ser tomadas para melhorar a realização das próximas Sprints.

8.2.1 O que funcionou bem?



Durante o planejamento da sprint o maior problema mapeado foi a implementação dos requisitos estabelecidos na modelagem da arquitetura. Por isso, foi considerado que a **implementação arquitetural** do projeto foi um dos pontos de destaque, já que foi implementado com sucesso.

No serviço do frontend todas **as features** foram desenvolvidas de acordo com o planejado e dentro do prazo, necessitando apenas de alguns ajustes visuais após a validação da entrega com os stakeholders.

No backend em geral, o **serviço de opções** (proposto na sprint 1) também foi desenvolvido com sucesso e dentro do prazo, sendo que os requisitos de implementação serverless e orientado a microserviços foram seguidos, assim como o desenvolvimento dos recursos a partir do cloudformation, garantindo a fácil manutenção do projeto.

O requisito de **serviço de deployment** automatizado também foi implementado com sucesso, sendo o GitHub actions desenvolvido como ferramenta de deploy associado com o AWS CLI, garantindo assim, a integração contínua.

O processo de **comunicação para integração** entre os integrantes do **frontend e backend** também foi considerado sucesso durante o processo, sendo o processo de mocks alinhado para facilitar tanto a agilidade do desenvolvimento quanto o consumo dos endpoints pelo frontend.

8.2.2 O que deu errado?

Durante o desenvolvimento do projeto foram encontrados problemas relacionados aos **testes de código, comunicação da equipe, realocação de tarefas e validação da entrega com o usuário final**.

Quanto aos testes de implementação, a arquitetura modelada dificultou a realização dos testes dado a diferença de complexidade associada a criação de testes automatizados na AWS, que necessita de um conhecimento específico para a criação de recursos fictícios de testes. Além disso, não existia conhecimento prévio por nenhum dos integrantes da equipe em relação ao framework de testes do frontend escolhido (Jest) e nenhum outro framework. Esses pontos impossibilitaram o desenvolvimento das tarefas, que foram repriorizadas para sprint 2.

Quanto a comunicação, existiram casos de falta de comunicação entre os integrantes que não se situaram em relação as tarefas explicitadas no backlog, não levantando dúvidas em relação aos pontos discutidos e não aderindo aos requisitos estabelecidos nos documentos devido à ausência nas reuniões semanais. Essa característica dificultou o cumprimento dos prazos devido a necessidade de retrabalho em algumas tarefas e até mesmo sua realocação.



Underlying



Por fim, foi observado também o problema da validação da entrega com o usuário final, onde não foi possível obter uma agenda que fosse compatível com a data de entrega da release 1 com os horários disponíveis dos entrevistados, impossibilitando a validação imediata das entregas pelos clientes.

8.2.3 Ações para melhorar

Dado o mapeamento dos problemas encontrados na sprint a equipe levantou algumas ações a serem realizadas pelos membros do time, sendo elas:

- ✓ Alocação de pessoas específicas para implementação de testes unitários, dado a complexidade das tarefas no cenário do projeto (Bruno – Backend, Wesley - Frontend)
- ✓ Definição e garantia de participação das reuniões semanais com uma semana de antecedência
- ✓ Adaptação do cronograma da sprint 2 para realização de tarefa específica realizada a criação do ambiente fictício para implementação de testes unitários
- ✓ Atualização diária (pelos responsáveis de cada tarefa) dos status das tarefas no Jira para melhor controle das atividades pelo Scrum Master
- ✓ Padronização do sistema de commit e merge por todos os membros da equipe através da utilização do GitFlow para evitar conflitos e sobrescrição de código.
- ✓ Melhoria da proatividade e qualidade da entrega já nos primeiros passos de desenvolvimento

Com todos os pontos levantados foi criado a tabela abaixo para que a equipe acompanhe e revise os aspectos citados durante o desenvolvimento da sprint 2.



Underlying



O que deu certo

Features do frontend
Features do backend
Elaboração da arquitetura
Documentação proposta
Adaptação da arquitetura para ganho de performance
Deploy da arquitetura serverless na nuvem
Pipeline de deployment automatizado

O que deu errado

Testes de implementação
Comunicação
Realocação de tarefas
Validação da entrega com o usuário (tempo / disponibilidade)

Pontos de atenção

Teste unitário AWS
Teste unitário Jest
Validação de desempenho de recurso antes da implementação para evitar retrabalho
Controle das tarefas no Jira

Ações para melhorar

Alocar pessoas especificamente para a realização dos testes
Definir horário de reunião semanal com uma semana de antecedência
Estudar de modo aprofundado as tecnologias que envolvem testes unitários
Criar recursos da aws fictícios para realização de testes
Solicitar atualização diária das atividades no Jira
Padronizar ações no GitHub - GitFlow

Tabela 9 – Retrospectiva da Sprint 01

9. BACKLOG DA SPRINT 02

A seguir será apresentado em planilha o Backlog da Sprint 02 realizado no Jira com o esforço real para o desenvolvimento das tarefas pelos seus responsáveis correspondentes:

ID HU	ID RF	ID RFS	ID RNF	Tarefa	Esforço Estimado	Responsável	Entrega	Esforço Real
HU04	RF03	RFS06a RFS09	RNF03 RNF09 RNF10 RNF12 RNF22	Autenticação de usuário	3	Lucas	23/06/2022	5
HU04	RF03	RFS06a RFS09	RNF03 RNF08 RNF09 RNF11 RNF29 RNF31	Registro de usuário	3	Ivan	23/06/2022	3
HU04	RF03	RFS06a	RNF01 RNF05	Tela de autenticação	5	Wesley	23/06/2022	1

		RFS09	RNF16 RNF18 RNF21 RNF22					
HU04	RF03	RFS06 a RFS09	RNF01 RNF05 RNF16 RNF18 RNF21 RNF22	Tela de registro de usuário	4	Thiago	23/06/2022	1
HU04	RF03	RFS06 a RFS09	RNF01 RNF05 RNF06 RNF07 RNF16 RNF18 RNF21 RNF22	Tela de perfil de usuário	6	Thiago	01/07/2022	4
HU06	RF05	RFS18 a RFS22	RNF01 RNF03 RNF05 RNF16 RNF18 RNF21 RNF22	Tela de estrutura de operação	6	Wesley	03/07/2022	6
HU06	RF05	RFS18 a RFS22	RNF03 RNF15 RNF19 RNF20	Criação de estrutura de operação	3	Lucas	01/07/2022	8

			RNF29					
HU06	RF07	RFS25	RNF01 RNF02 RNF05 RNF16 ao RNF22	Payoff de estrutura de operação	4	Ivan	28/06/2022	4
HU07	RF10	RFS29 a RFS31	RNF03 RNF15 RNF19 RNF20 RNF29	Criação de Opção Fictícia	8	Bruno	28/06/2022	3
HU07	RF10	RFS29 a RFS31	RNF01 RNF03 RNF05 RNF16 RNF 22	Formulário para criação de Opção Fictícia	5	Leonardo	28/06/2022	5
Todos, exceto os específicos da sprint 03			-	Testes frontend	10	Wesley	18/07/2022	9
			-	Testes backend	10	Bruno	18/07/2022	10
HU11	-	-	RNF23	Página de glossário	3	Ivan	06/07/2022	3
HU10	RF06	RFS22	RNF13 RNF14 RNF15	Compartilhamento de estruturas de operações	6	Lucas	26/08/2022	6
HU10	RF06	RFS23	RNF01 RNF05 RNF16	Listagem de estruturas de operações compartilhadas	6	Leonardo	26/08/2022	6

			ao RNF 22					
HU10	RF06	RFS23	RNF01 RNF04 RNF05 RNF16 ao RNF 22 RNF 23	Tela de estrutura de operações compartilhadas	9	Leonardo	26/08/2022	9

Tabela 10 - Backlog da Sprint 02

9.1 Velocidade da Equipe:

O desenvolvimento e finalização da sprint 2 impactaram positivamente na finalização de todos os objetivos propostos durante o planejamento, permitindo o adiantamento de tarefas da sprint 3.

No planejamento a Sprint 2 possuía um total de **63** story points incluindo os processos de testes e deployment de todas as funcionalidades propostas. Ao fim da sprint 2 o total de story points reais mapeados foram **67**. O controle da velocidade da equipe do projeto pode ser acompanhado no gráfico de controle abaixo:



Underlying

Total de Story Points planejado: 63 | Total de Story Points real: 67 | Tempo médio de atividade: 5 dias



Figura 10 - Desempenho Sprint 02

A segunda sprint começou dia 05/06 e pelo gráfico vemos que o início da mesma começou de forma mais rápida com menos de 5 dias para a realização das tarefas, mas a partir do dia 26/06 as tarefas começaram a demorar mais de 5 dias para ser concluídas devido ser o fim de semestre e a equipe estar envolvida nos demais projetos da unifei. O gráfico apresenta os seguintes dados de velocidade equipe desde o início do projeto:

- **Velocidade planejada: 63 story points**
- **Velocidade real: 67 story points**
- **Tempo médio de uma tarefa: 6 dias**
- **Tempo mediano de uma tarefa: 5 dias**
- **Maior tempo de uma tarefa: 3 semanas e 7 horas**

Considerando que a velocidade real da sprint 1 foi de 62 Story Points e da sprint 2 de 67 a velocidade média atual da equipe corresponde a aproximadamente 65 Story Points por sprint.

Em relação a sprint 02 foi planejado 63 Story Points, porém a equipe concluiu as atividades programadas antes do tempo previsto e então foi adicionado ao cronograma 21 Story Points, indo de 63 planejados para 84, porém deste novo escopo foram realizados um total de 67 Story Points, realizando então apenas 4 Story Points a mais do que foi planejado inicialmente, e assim adiantou o projeto.

Tem-se dois motivos principais pelo adiantamento do projeto. O 1º é que Os Valores do planejamento das tarefas foram superestimados e o segundo que também



Underlying



justifica o fato deles terem sido subestimado foi em relação ao tempo gasto com a arquitetura na 1ª Sprint que foi superior ao tempo gasto na 2ª Sprint.

9.2 Impacto da arquitetura na sprint 2

Na 1ª sprint foi preciso realizar todo o planejamento e a criação da arquitetura da aws, voltou se então para a execução das tarefas de infraestrutura da api com o cloudformation, lambdas, hospedagem frontend, storages e engine. E estas tarefas consumiram um total de 20 Story Points dos 62 realizados na Sprint 1 o que correspondeu a 32,3% dos Story Points realizados.

Na 2ª sprint o tempo com a arquitetura foi apenas gasto com o Banco de dados não relacional: o dynamodb que consumiu somente 3 Story Points dos 67 realizados na sprint 2, ou seja, consumiu apenas 4,47% do Story Points realizados.

Então os benefícios e conclusões que tiramos foram

- Que a preocupação, dedicação e tempo inicial gasto com arquitetura na primeira sprint foi crucial para dar a execução e vida ao projeto, principalmente agora na segunda sprint onde sua correta implementação fez com que não precisássemos retrabalhar e nem alterar a arquitetura inicial e assim permitiu a equipe a adiantar o projeto já que o tempo que foi gasto anteriormente com a arquitetura pode ser utilizado para dar continuidade em outras e novas tarefas do projeto.
- Outra afirmativa é que a Sprint 2 realizou uma maior entrega de funcionalidades ao cliente ao implementar mais funções requeridas pelo usuário que estavam previstas apenas para serem implementadas na Sprint 3 como, por exemplo, o compartilhamento das estratégias pelo usuário

9.3 Retrospectiva da Equipe do Sprint 2

A seguir será apresentado o que deu certo na execução da sprint 02, na sequência o que não deu certo, e, por fim, quais ações deverão ser tomadas para melhorar a realização da Sprint 03.

9.3.1 O que funcionou bem?

Com a devida arquitetura já bem definida e estruturada possibilitou o adiantamento do projeto com aumento do escopo da sprint 2 com atividades extras que seriam apenas realizadas na sprint 3.

A sprint 2 conseguiu prover maior valor e resultado ao usuário final trazendo maior funcionalidades do sistema.



Underlying



Possibilitou a execução dos testes em atrasos e das atuais tarefas, em especial os testes funcionais, não funcionais e o teste de validação com o usuário final das features desenvolvidas.

9.3.2 O que deu errado?

Durante o desenvolvimento do projeto ainda foram encontrados problemas relacionados aos **desinteresses dos integrantes da equipe, prazos em cima da hora, realocação de tarefas, falta de tempo e foco, a não atualização da ferramenta jira.**

- Mesmo sendo comunicado e definidas as tarefas não foram todos os integrantes que se envolveram diretamente no desenvolvimento final do projeto. Não houve falta de comunicação, e o próprio documento e ferramentas assimilam as responsabilidades de cada integrante.
- Para o cumprimento dos prazos as tarefas tiveram de ser realocadas várias vezes e demandando mais esforços de alguns integrantes do que de outros, alguns deixaram a entrega em cima da última hora, prejudicando o desenvolvimento de toda a equipe comprometida.
- Por mais que se tenha conseguido adiantar o projeto com o comprometimento de alguns a ausência de tempo foi e, é sempre crucial, uma vez que, os envolvidos estão realizando outros projetos dividindo assim o foco e os objetivos deste para com os demais.
- Os responsáveis pelas tarefas na ferramenta jira não têm atualizado constantemente as tarefas conforme as finalizam, dificultando assim a análise e o uso de relatórios do scrum master.

9.3.3 Ações para melhorar

Dado o mapeamento dos problemas encontrados na sprint 2 a equipe levantou algumas ações a serem realizadas pelos membros do time, sendo elas:

- ✓ Definição e garantia de participação das reuniões duas vezes por semana.
- ✓ Atualização pelos responsáveis de cada tarefa no Jira para melhorar o controle e a análise das atividades pelo Scrum Master
- ✓ Participação nas reuniões, na execução e mais proatividade na entrega de todas as atividades requeridas e estabelecidas.

Com todos os pontos levantados foi criado a tabela abaixo para que a equipe acompanhe e revise os aspectos citados durante o desenvolvimento da sprint 2.



Underlying



O que deu certo	O que deu errado	Pontos de atenção	Ações para melhorar
<p>Features do frontend</p> <p>Features do backend</p> <p>Redução de esforço com arquitetura</p> <p>Maior valor e resultado ao usuário</p> <p>Entrega além do planejado (cronograma adiantado)</p> <p>Ambiente de testes AWS</p> <p>Teste de aceitação do usuário final</p> <p>Testes funcionais e não funcionais</p>	<p>Atualização das atividades finalizadas no Jira para análise e controle</p> <p>Fluxo GitHub - GitFlow (evitar conflitos de código)</p> <p>A distribuição de tarefas: mesmo atingindo o cronograma uns se envolveram mais que outros no projeto.</p>	<p>Foco e planejamento de tempo</p> <p>Proatividade e interesse dos integrantes</p> <p>Controle das tarefas no Jira</p> <p>Não realizar entregas em cima da hora</p>	<p>Solicitar atualização diária das atividades no Jira</p> <p>Reuniões semanais</p> <p>Mais participação, colaboração e empatia dos envolvidos</p>

Tabela 11 - Retrospectiva da Sprint 02

10. BACKLOG DA SPRINT 03

A seguir será apresentado em planilha o Backlog da Sprint 03 realizado no Jira. Para mais detalhes verificar o documento do Cronograma com Jira em anexo.

ID HU	ID RF	ID RFS	ID RNF	Tarefa	Esforço (story points)	Responsável	Entrega
HU08	RF08	RFS26	RNF16 ao RNF20 RNF29 RNF32	Crawler de registro de ações	10	Lucas	06/08/2022
HU08	RF08	RFS27	RNF03 RNF15 RNF17 RNF19 RNF20	Busca de ação por ticker	4	Bruno	06/08/2022



Underlying



			RNF22				
HU09	RF09	RFS28	RNF03 RNF15 RNF19 RNF20 RNF29	Geração de grade de Opções por ticker	8	Lucas	06/08/2022
HU09	RF09	RFS28	RNF01 RNF05 RNF16 RNF18 RNF21 RNF22	Tela de grade de Opções	9	Leonardo	06/08/2022
HU09	RF04	RFS10 a RFS17	RNF01 RNF05 RNF16 ao RNF22	Tabela de métricas de mercado e greeks	6	Leonardo	18/08/2022
HU08	RF08	RFS26	RNF01 RNF05 RNF16 ao RNF 22	Exibição de dados de ações	4	Thiago	18/08/2022
HU05	RF01	RFS01 RFS02	RNF03 RNF15 RNF17 RNF19 RNF20 RNF22	Busca avançada de Opções por filtros	7	Lucas	18/08/2022
HU05	RF01	RFS01	RNF01	Formulário de	6	Wesley	18/08/2022

		RFS02	RNF03	filtros para busca			
			RNF05				
			RNF16				
			RNF21				
			RNF 22				

Tabela 12 - Backlog da Sprint 03

11. MOCKUPS DO SISTEMA:

Esta seção apresenta os protótipos das principais telas e funcionalidades que foram idealizadas para o sistema. Após ser apresentada a tela será feita uma breve descrição dos requisitos funcionais do sistema que se pretendem abordar em cada tela. As Principais telas/páginas apresentadas são:

- Página inicial
- Login de usuário
- Registro de usuário
- Grade de opções
- Visualizar Opções
- Criação de Opção Fictícia
- Buscar Opção
- Criar Estrutura de Operação
- Estrutura de Operação Compartilhada

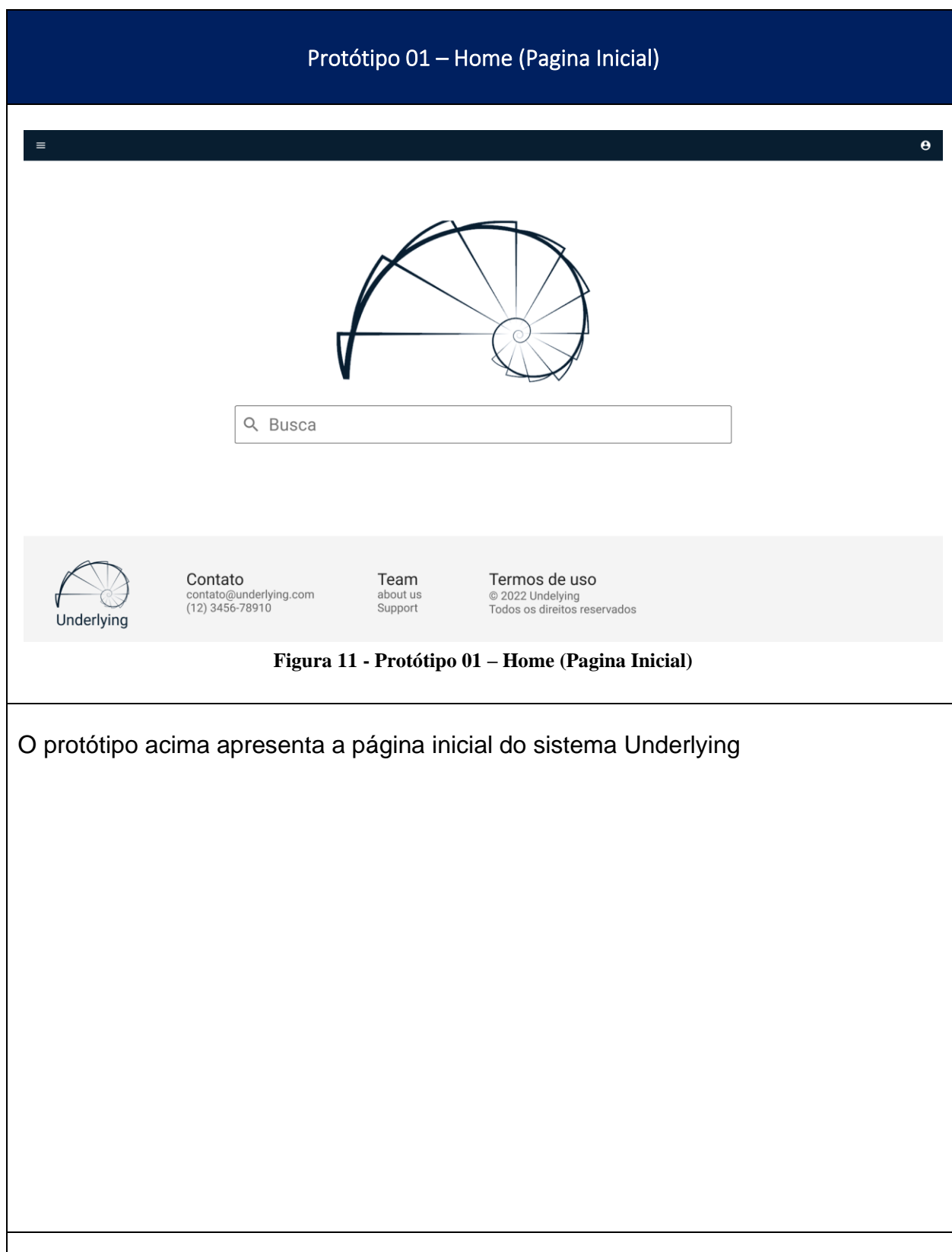
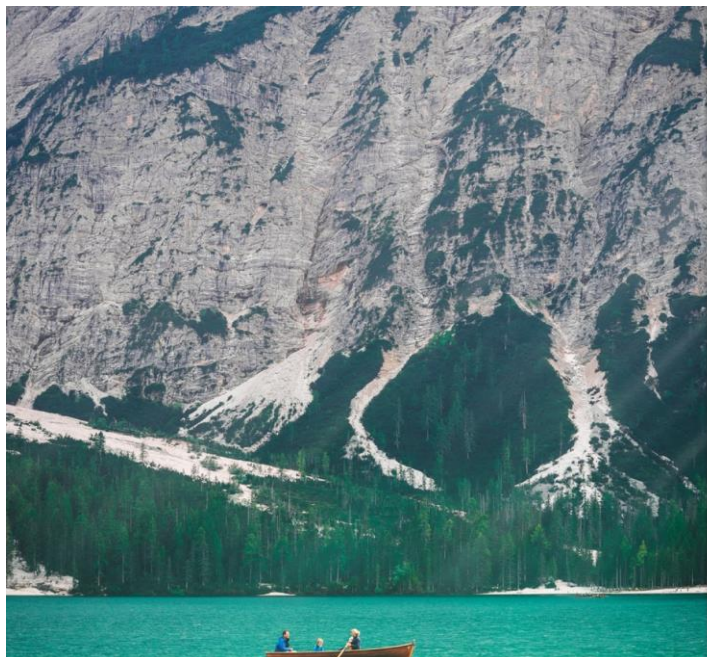



Figura 11 - Protótipo 01 – Home (Pagina Inicial)

O protótipo acima apresenta a página inicial do sistema Underlying

Protótipo 02 – Login de Usuário





Entrar

☐ Lembre-me neste dispositivo

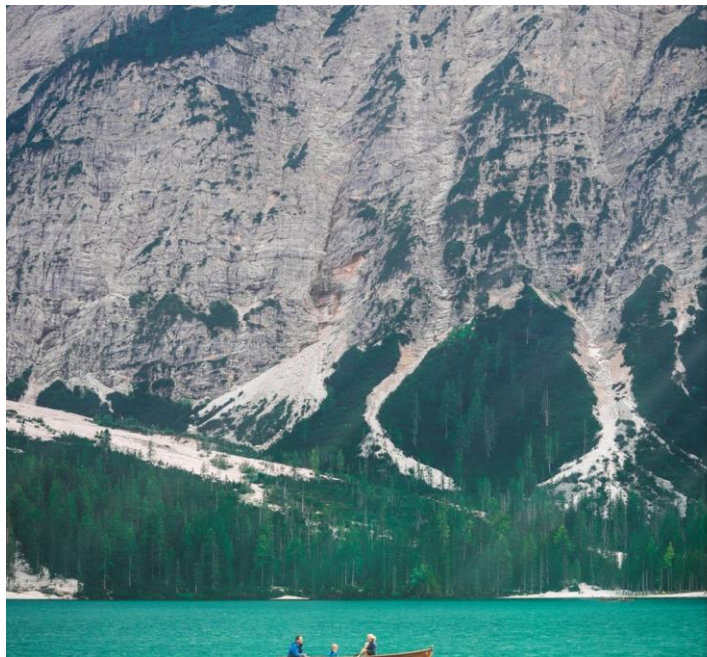
ENTRAR


[Criar conta](#) [Esqueceu a senha?](#)

Figura 12 - Protótipo 02 – Login de Usuário

O protótipo acima representa a HU04 com o RF03 onde será permitido realizar o login de usuário para acesso ao sistema.

Protótipo 03 – Cadastro de Usuário (Registro)





Registrar

☐ Lembre-me neste dispositivo

ENTRAR

Figura 13 - Protótipo 03 – Cadastro de Usuário (Registro)

O protótipo acima representa a HU04 com o RF03 e o RFS06 onde sistema permitirá o usuário realizar o cadastro no sistema

Protótipo 04 - Grade de Opções

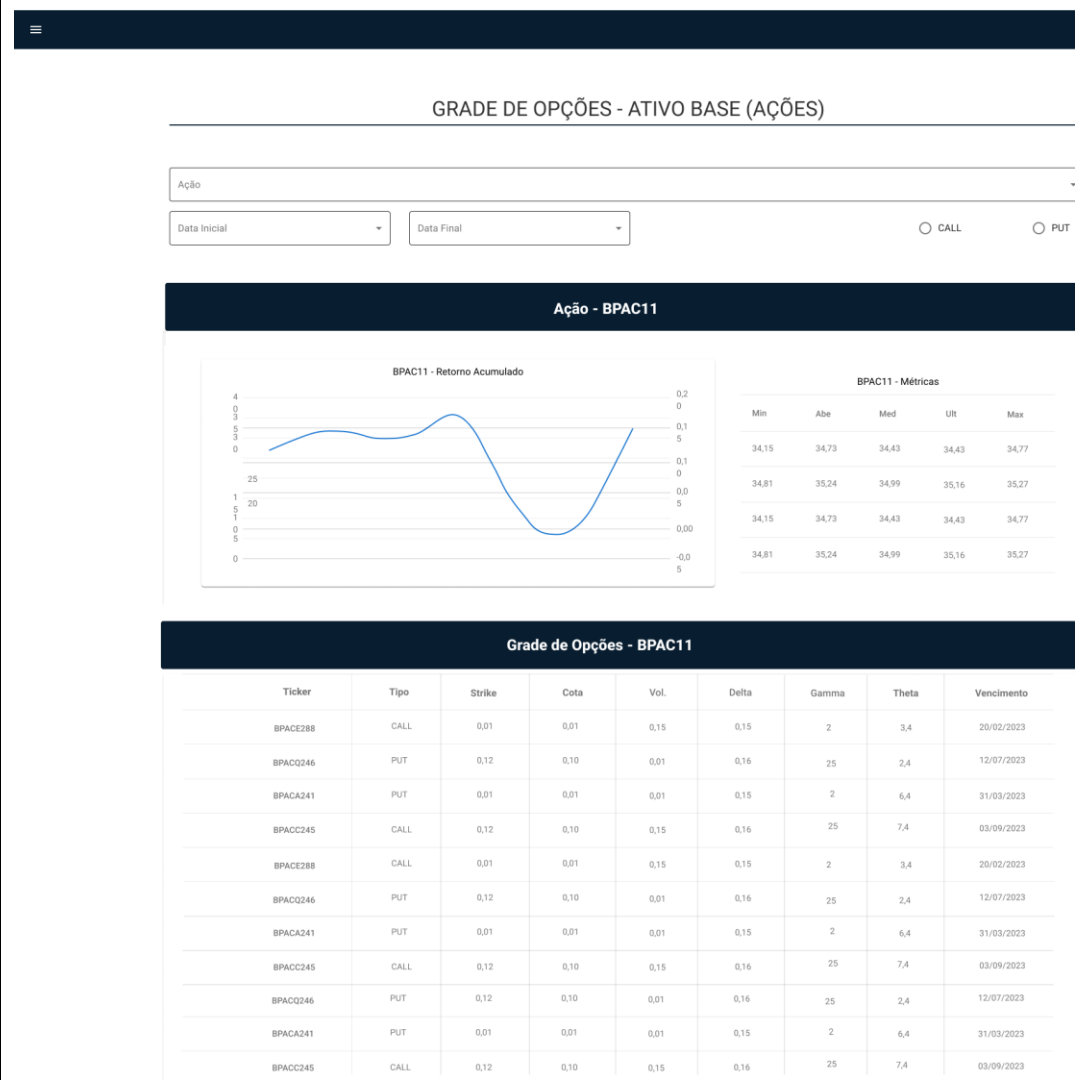


Figura 14 - Protótipo 04 - Grade de Opções

O protótipo acima representa as HU09 com o RF09 e o RFS28 onde sistema permitirá a emissão do relatório de grade de Opções contendo informações relevantes a partir da seleção da Ação e do período.

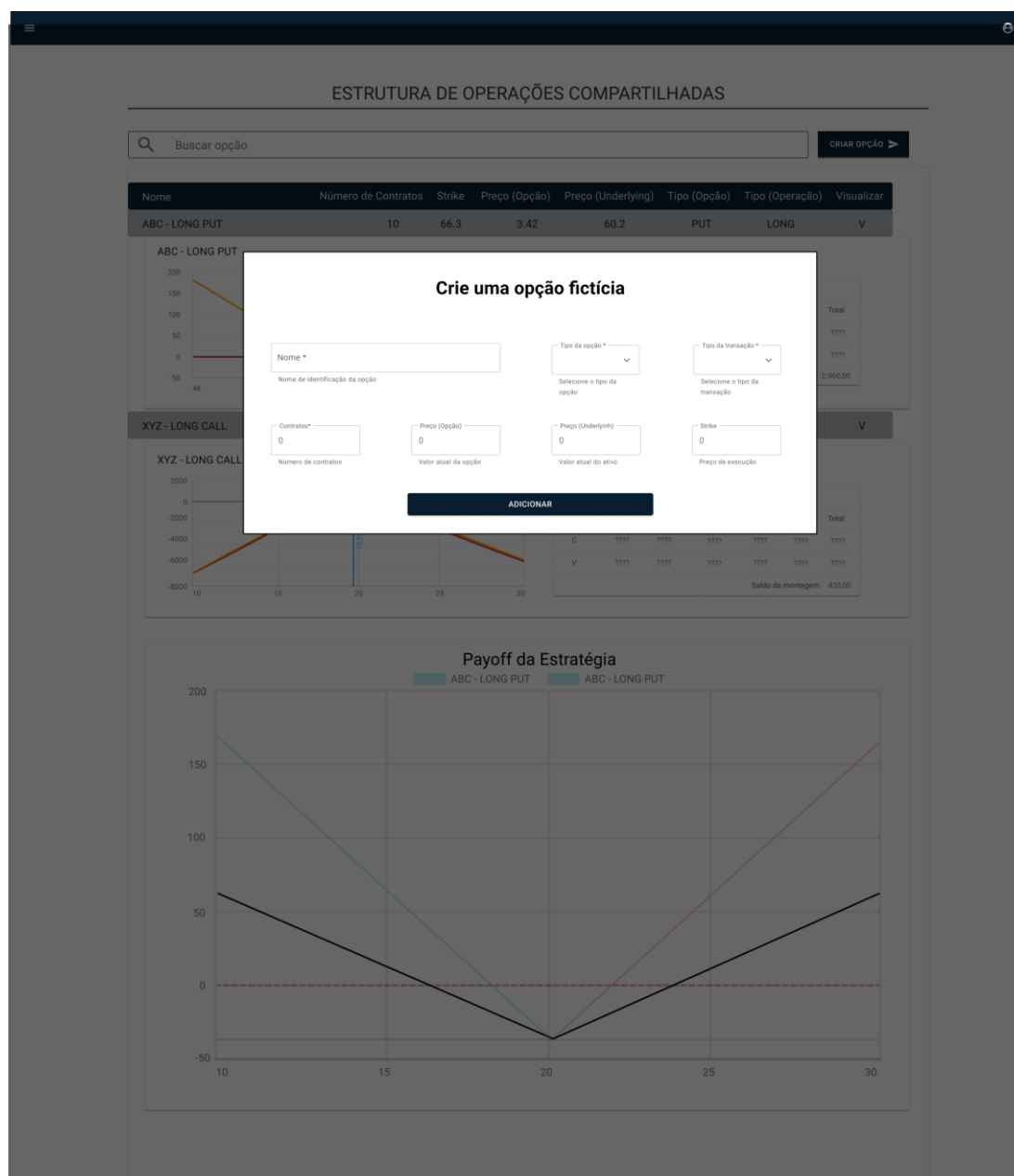
Protótipo 05 – Visualizar Opções



Figura 15 - Protótipo 05 – Visualizar Opções

O protótipo acima representa as HU03 com o RF01 e o RFS02 onde sistema permitirá que ao selecionar uma Opção seja retornado todos os dados de registro da Opção acompanhados de séries de preço e informações de desempenho previamente calculadas.

Protótipo 06 – Criação de Opção Fictícia



ESTRUTURA DE OPERAÇÕES COMPARTILHADAS

Buscar opção CRIAR OPÇÃO ➤

Nome	Número de Contratos	Strike	Preço (Opção)	Preço (Underlying)	Tipo (Opção)	Tipo (Operação)	Visualizar
ABC - LONG PUT	10	66.3	3.42	60.2	PUT	LONG	V

Crie uma opção fictícia

Nome *

Nome de identificação da opção

Tipo da opção *

Selecione o tipo da opção

Tipo da transação *

Selecione o tipo da transação

Contratos*

Número de contratos

Preço (Opção)

Valor atual da opção

Preço (Underlying)

Valor atual do ativo

Strike

Preço de execução

ADICIONAR

Payoff da Estratégia

ABC - LONG PUT ABC - LONG PUT

200
150
100
50
0
-50

10 15 20 25 30

Figura 16 - Protótipo 06 – Criação de Opção Fictícia

O protótipo acima representa as HU07 com os RF10 e o RFS29 onde sistema permitirá a criação de uma opção fictícia com os seguintes dados: nome, tipo da opção, tipo da transação, número de contratos, preço do ativo base, preço da Opção e preço de strike.



Protótipo 07 - Buscar Opção

[illegible]

Figura 17 - Protótipo 07 - Buscar Opção

O protótipo acima representa as HU01, HU02 e HU05 com os RF01 e RF04 e ainda os RFS01 a RFS04 Que permitiram o sistema realizar através de filtros avançados a busca pela Opções selecionada e, assim visualizar as informações, as métricas de desempenho e geeks.

Protótipo 08 – Criar Estrutura de Operação

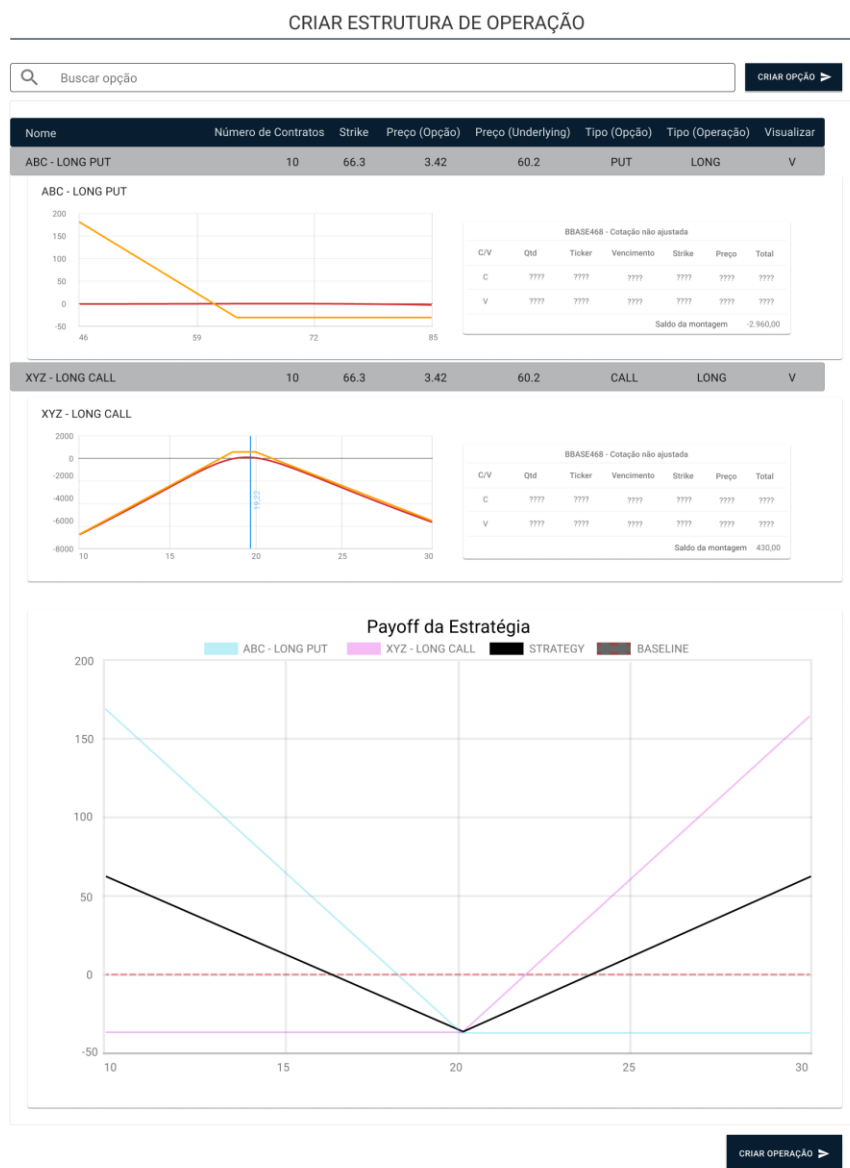


Figura 18 - Protótipo 08 – Criar Estrutura de Operação

O protótipo acima representa a HU06 e a HU03 com os RF05 e os RFS18, RFS19, RFS 24 e RFS25. Onde o sistema permitirá a criação de uma estrutura de Operação de Opções, e a sua visualização com dados e o gráfico de Payoff.

Protótipo 09 - Estrutura de Operação compartilhada



Figura 19 - Protótipo 09 - Estrutura de Operação compartilhada

O protótipo acima representa a HU10 com o RF06 e o RFS23 onde o sistema permite realizar a visualização das estruturas compartilhadas pelos usuários do sistema Underlying.

12. REFERÊNCIAS E ANEXOS:

A seguir são apresentados os seguintes artefatos que estão anexos a este documento e que são de grande importância para o desenvolvimento deste projeto.

- **Anexo 1: Cronograma do Sprint 3.pdf**
- **Anexo 2: Documento da Arquitetura do Sistema.pdf**
- **Anexo 3: Código Fonte.pdf**
- **Anexo 4.1: TestesEstrutural_DTE_Caixa_Branca_VF02.pdf**
- **Anexo 4.2: Testes_Funcionais_DTF_Caixa_Preta_VF02.pdf**
- **Anexo 5 Código dos Testes Unitários e de Integração.pdf**
- **Anexo 6.1: Planejamento do Teste de Aceitação.pdf**



Underlying



- **Anexo 6.2: Execução do Teste de Aceitação.xlsx**
- **Anexo 7: Pasta contendo os Testes não funcionais**
- **Anexo 8: Relatório Sprint - Relatório de pendencias e conclusoes.pdf**
- **Anexo 9: Slides- Underlying.pdf**

Plataforma em produção: <http://underlying.com.br>