

IFSP - Instituto Federal de Educação, Ciência e Tecnologia
Câmpus São Paulo

ANAÍ VILLCA ROJAS	SP3029085
JAMILI VITÓRIA GIOIELLI	SP3027473
JOSÉ ROBERTO CLAUDINO FERREIRA	SP3024369
JULIA ROMUALDO PEREIRA	SP3023061
KAIKY MATSUMOTO SILVA	SP185075X

IFriends: uma comunidade virtual

São Paulo - SP - Brasil

2022

**IFSP - Instituto Federal de Educação, Ciência e Tecnologia
Câmpus São Paulo**

ANAÍ VILLCA ROJAS	SP3029085
JAMILI VITÓRIA GIOIELLI	SP3027473
JOSÉ ROBERTO CLAUDINO FERREIRA	SP3024369
JULIA ROMUALDO PEREIRA	SP3023061
KAIKY MATSUMOTO SILVA	SP185075X

IFriends: uma comunidade virtual

Projeto de sistema IFriends apresentado, conforme as normas ABNT, à disciplina de Prática para Desenvolvimento de Sistemas.

Professor: Johnata Souza Santicioli

Professor: Carlos Henrique Verissimo Pereira

IFSP - Instituto Federal de Educação, Ciência e Tecnologia
Câmpus São Paulo

Curso Técnico em Informática Integrado ao Ensino Médio

PDS - Prática para Desenvolvimento de Sistemas

São Paulo - SP - Brasil

2022

Listas de ilustrações

Figura 1 – Link da aplicação	15
Figura 2 – Link do <i>blog</i> do projeto	15
Figura 3 – Link do canal no <i>Youtube</i>	15
Figura 4 – Link do repositório	16
Figura 5 – Arquitetura REST API	18
Figura 6 – Cronograma de Desenvolvimento Semanal	23
Figura 7 – Histórias no Notion	26
Figura 8 – Diagrama de Casos de Uso	33
Figura 9 – Diagrama de Classes	34
Figura 10 – Diagrama de Classes	35
Figura 11 – Diagrama de Entidade e Relacionamento	36
Figura 12 – Diagrama de Tabelas Relacionais	37
Figura 13 – Logo do projeto	38
Figura 14 – Modelo Colmeia	39
Figura 15 – Mapa mental	39
Figura 16 – Swagger UI da API	41
Figura 17 – Exemplo Swagger na API	42
Figura 18 – Análise de segurança na API	43
Figura 19 – Análise de segurança no Front-end	43
Figura 20 – Análise de respostas HTTP na API	44
Figura 21 – Teste na API	46
Figura 22 – Cobertura de testes	47
Figura 23 – Página inicial sem <i>login</i>	70
Figura 24 – Página da pergunta sem <i>login</i>	71
Figura 25 – Página de cadastro de perguntas sem <i>login</i>	72
Figura 26 – Página de <i>login</i>	72
Figura 27 – Página de cadastro	73
Figura 28 – Página inicial	74
Figura 29 – Pergunta e respostas	76
Figura 30 – Cadastro de perguntas	77
Figura 31 – Logo da equipe Bunka Bytes	84
Figura 32 – Logo Yamillint	92
Figura 33 – Diretório	96
Figura 34 – Erro - <i>Trailling spaces</i>	96
Figura 35 – Erro - <i>no new line at the end of file</i>	96
Figura 36 – Cuidado	97

Figura 37 – Exemplo de como abrir o terminal	99
Figura 38 – Arquivo de caption	100

Lista de quadros

Quadro 1 – Análise de Concorrência	14
Quadro 2 – Distribuição de tarefas	22
Quadro 3 – Métricas de Organização das Histórias	26
Quadro 4 – Resumo: Gestão de perguntas	28
Quadro 5 – Resumo: Gestão de respostas	28
Quadro 6 – Resumo: Gestão de eventos	29
Quadro 7 – Resumo: Gestão de usuários	29
Quadro 8 – Resumo: Usabilidade	29
Quadro 9 – Regras de Negócio	30
Quadro 10 – Partes interessadas	31
Quadro 11 – Usuários	31
Quadro 12 – História: Manter uma pergunta	58
Quadro 13 – História: Filtrar perguntas	58
Quadro 14 – História: Adicionar assuntos	59
Quadro 15 – História: Curtir uma pergunta	59
Quadro 16 – História: Buscar perguntas	59
Quadro 17 – História: Manter uma resposta	60
Quadro 18 – História: Curtir uma resposta	60
Quadro 19 – História: Manter evento	61
Quadro 20 – História: Favoritar evento	61
Quadro 21 – História: Autenticação do usuário	61
Quadro 22 – História: Internacionalização	62
Quadro 23 – Usuário	64
Quadro 24 – Usuário_Pergunta	64
Quadro 25 – Usuário_Resposta	65
Quadro 26 – Usuário_Título	65
Quadro 27 – Título	65
Quadro 28 – Curso	66
Quadro 29 – Resposta	66
Quadro 30 – Tag	66
Quadro 31 – Pergunta	67
Quadro 32 – Tag_Pergunta	67
Quadro 33 – Tag_Evento	68
Quadro 34 – Evento	68
Quadro 35 – Categoria	68

Lista de abreviaturas e siglas

ACME	<i>Advanced Counter-Measures Environment</i> - Ambiente avançado de contra-medidas - Citado em 42
API	<i>Application Programming Interface</i> - Interface de programação de aplicações - Citado em 17, 34, 40, 41, 43, 46, 61, 82, 91, 92, 102, 103, 104, 105, 106
cmd	<i>Command Prompt</i> - Prompt de Comando - Citado em 95
CORS	<i>Cross-Origin Resource Sharing</i> - Compartilhamento de recursos entre origens - Citado em 104
CSS	<i>Cascading Style Sheets</i> - Folha de Estilo em Cascatas - Citado em 104
DD	Dicionário de Dados - Citado em 37, 38, 64, 106
DER	Diagrama Entidade Relacionamento - Citado em 35, 105, 106
DTR	Diagrama de Tabelas Relacionais - Citado em 36, 105, 106
EAD	Ensino a Distância - Citado em 90
HSTS	<i>HTTP Strict Transport Security</i> - Segurança restrita de transporte HTTP - Citado em 42
HTTP	<i>Hypertext Transfer Protocol</i> - Protocolo de Transferência de Hiper-texto - Citado em 17, 40, 41, 43
IDE	<i>Integrated Development Environment</i> - Ambiente de Desenvolvimento Integrado - Citado em 19, 46
IFSP	Instituto Federal de Educação, Ciência e Tecnologia de São Paulo - Citado em 10, 11, 12, 13, 38, 79, 80, 81, 82, 85, 87, 88, 90
JaCoCo	<i>Java Code Coverage</i> - Cobertura de Código Java - Citado em 46, 104, 105
JWT	JSON Web Token - Citado em 41, 61, 102
MVC	<i>Model-View-Controller</i> - Modelo-Visão-Controle - Citado em 17
PDS	Prática de Desenvolvimento de Sistemas - Citado em 10, 23, 24, 49, 79, 84, 85, 92, 98
PJI	Projeto Integrador - Citado em 91
POC	<i>Proof of Concept</i> - Prova de Conceito - Citado em 48, 80, 81, 82, 83, 89, 90, 91, 92, 102, 103, 105, 106
Q/A	<i>Questions and Answers</i> - Perguntas e Respostas - Citado em 85
RN	Regra de Negócio - Citado em 30
SGBD	<i>Data Base Management System</i> - Sistema de Gerenciamento de Banco de Dados - Citado em 18, 44

- SO Sistema Operacional - Citado em [94](#)
- SQL *Structured Query Language* - Linguagem de Consulta Estruturada
- Citado em [18](#)
- TDS Teoria e Desenvolvimento de Sistemas - Citado em [84](#)
- TLS *Transport Layer Security* - Segurança da camada de transporte -
Citado em [42](#), [44](#)
- UML *Unified Modeling Language* - Linguagem de Modelagem Unificada
- Citado em [32](#)
- URI *Uniform Resource Identifier* - Identificador Uniforme de Recurso -
Citado em [102](#)

Sumário

1	INTRODUÇÃO	10
1.1	Origem do Projeto	10
1.2	Pesquisa de viabilidade	11
1.2.1	Resultados	11
1.3	Objetivo	12
1.4	Justificativa	12
1.5	Análise de Concorrência	13
2	DESENVOLVIMENTO	15
2.1	Materiais e Métodos	16
2.1.1	Tecnologias e ferramentas de desenvolvimento	16
2.1.2	Ferramentas de apoio	18
2.2	Métodos de gestão e desenvolvimento	19
2.2.1	Metodologia ágil: Scrum	19
2.2.1.1	O quadro de kanban	20
2.3	Gestão da equipe	21
2.3.1	Desenvolvimento: artefatos e eventos	23
2.3.2	Métricas de Desenvolvimento	25
2.4	Análise de Requisitos	27
2.4.1	Histórias de Usuário	27
2.4.2	Regras de Negócio	30
2.4.3	Partes interessadas e usuários	30
2.5	Modelagem	31
2.5.1	Diagrama de Casos de Uso	32
2.5.2	Diagrama de Classes	33
2.5.3	Diagrama de Entidade e Relacionamento	35
2.5.4	Diagrama de Tabelas Relacionais	36
2.5.5	Dicionário de Dados	37
2.6	Design do projeto	38
2.6.1	Prototipagem	39
2.6.1.1	Protótipos de alta fidelidade	40
2.7	Swagger na API	40
2.8	Análise de segurança	42
2.9	Plano de testes	44
2.9.1	Testes unitários	45

2.9.2	Testes de usabilidade	47
3	HISTÓRICO DE DESENVOLVIMENTO	48
3.1	Descartes	49
 REFERÊNCIAS		50
GLOSSÁRIO		51
 APÊNDICES		56
APÊNDICE A – PERGUNTAS DA PESQUISA DE VIABILIDADE		57
APÊNDICE B – HISTÓRIAS DE USUÁRIO		58
B.1	Épico - Gestão de perguntas	58
B.2	Épico - Gestão de respostas	60
B.3	Épico - Gestão de eventos	60
B.4	Épico - Gestão de usuários	61
B.5	Épico - Usabilidade	62
APÊNDICE C – BOAS PRÁTICAS PARA UMA PERGUNTA		63
APÊNDICE D – DICIONÁRIO DE DADOS		64
APÊNDICE E – PROTOTIPAÇÃO DAS TELAS		69
E.1	Friend sem <i>login</i>	69
E.2	Friend com <i>login</i>	73
APÊNDICE F – ATAS DAS REUNIÕES		78
F.1	1ºbimestre	78
F.1.1	Planejamento - 15/03/2022	78
F.1.2	Planejamento/Alinhamento - 17/03/2022	78
F.1.3	Planejamento/Alinhamento - 18/03/2022	78
F.1.4	Alinhamento - 21/03/2022	79
F.1.5	Alinhamento - 25/03/2022	79
F.1.6	Planejamento/Alinhamento - 28/03/2022	79
F.1.7	Planejamento/Retrospectiva - 02/04/2022	79
F.1.8	Planejamento - 04/04/2022	80
F.1.9	Alinhamento - 09/04/2022	80
F.1.10	Alinhamento - 10/04/2022	80

F.1.11	Retrospectiva - 12/04/2022	80
F.1.12	Planejamento/Alinhamento - 17/04/2022	80
F.1.13	Planejamento - 18/04/2022	81
F.1.14	Alinhamento - 21/04/2022	81
F.1.15	Alinhamento - 25/04/2022	81
F.1.16	Alinhamento - 01/05/2022	81
F.1.17	Alinhamento - 03/05/2022	82
F.2	2ºbimestre	82
F.2.1	Retrospectiva - 16/05/2022	82
F.2.2	Planejamento - 29/05/2022	82
F.2.3	Planejamento - 06/06/2022	82
F.2.4	Planejamento 13/06/2022	83
	APÊNDICE G – PUBLICAÇÕES NO BLOG DA EQUIPE	84
G.1	1ª Semana - 14/03 à 20/03	84
G.2	2ª Semana - 21/03 à 27/03	85
G.3	3ª Semana - 28/03 à 03/04	86
G.4	4ª Semana - 04/04 à 10/04	87
G.5	5ª Semana - 11/04 à 17/04	87
G.6	6ª Semana - 18/04 à 24/04	88
G.7	7ª Semana - 25/04 à 01/05	89
G.8	8ª Semana - 02/05 à 08/05	90
G.9	9ª Semana - 09/05 à 15/05	90
G.10	10ª Semana - 16/05 à 22/05	91
G.11	Como utilizar o Yamllint: validador de arquivos .yaml	92
G.12	Gerando o Vídeo do Gource	98
G.13	11ª Semana - 23/05 à 29/05	102
G.14	12ª Semana - 30/05 à 05/06	103
G.15	13ª Semana - 06/06 à 12/06	104
G.16	14ª Semana - 13/06 à 19/06	105
	APÊNDICE H – DOCUMENTOS ENTREGUES	107
	APÊNDICE I – PROVA DE CONCEITO	125

1 Introdução

O presente documento é resultado da proposta de um projeto cujo objetivo é sustentado no planejamento e na execução de um sistema para a Web através do aprendizado obtido nas matérias técnicas do Curso Técnico de Informática, realizado no [IFSP](#), e como forma de trabalho de conclusão de curso.

Tendo isto em vista e diante dos desafios presentes na lista de requisitos e orientações propostas para a iniciação deste projeto, os integrantes da equipe Bunka Bytes reuniram-se em busca de encontrar a solução que melhor se encaixasse em seus objetivos e nos da disciplina de [PDS](#).

Portanto, este documento apresenta o processo de desenvolvimento do [IFFriends](#), desde a origem do projeto, até sua execução final, documentando o acompanhamento de todas as atividades executadas pelos membros da equipe para sua elaboração.

1.1 Origem do Projeto

E foi pensando em soluções viáveis que a equipe voltou seu olhar para sistemas que contribuem para a criação de comunidades colaborativas na área de desenvolvimento de sistemas, que, segundo [Rosa \(2008\)](#), foi fortemente difundida pelas comunidades de Código Aberto (do inglês, *Open Source*), sendo primeiramente criada pela cultura *hacker*, na qual afirma que a paixão e o interesse dos *hackers* nas soluções foi uma das principais propulsoras do espírito colaborativo.

Além deles, foi trazido ao debate as possibilidades de aliar as principais dificuldades que os integrantes observaram durante sua vida acadêmica no [IFSP](#), a um sistema que pudesse suprir determinadas necessidades dos alunos, como os questionamentos que começam a surgir com mais frequência conforme o início dos estudos é dado, sendo eles em âmbitos diversos como: sobre a instituição de ensino, matérias e assuntos tratados no ensino médio, dúvidas sobre os conteúdos técnicos ou até mesmo a busca por um apoio educacional - como ocorrem nas monitoriais.

O Instituto Federal de São Paulo ([IFSP](#)), especificamente no campus São Paulo, cenário onde os integrantes da equipe estão inseridos e também público-alvo que a comunidade pretende atingir, é uma instituição de ensino que fornece ensino médio, técnicos profissionalizantes e graduações. Por ser um ambiente novo e repleto de informações simultâneas, muitos alunos ingressantes ficam perdidos no momento de lidar com coisas cotidianas, como: funcionamento da instituição, quantidade de disciplinas, volume de atividades, novas pessoas para se conviver, liberdade e responsabilidade que é concedida ao

aluno, entre muitas outras. Isso torna o IFSP um ambiente muito desafiador que pode gerar inúmeras dúvidas entre os novos alunos.

O IFriends surge nesse cenário, no qual a criação de uma comunidade de estudantes que colaborassem entre si, pudesse instigar o interesse dos alunos em ajudarem uns aos outros de maneira acessível e prática, onde uma dúvida estivesse a um palmo de distância.

1.2 Pesquisa de viabilidade

Foi realizada pela equipe uma pesquisa de viabilidade da aplicação, visando verificar se o público-alvo realmente estava sendo atingido, ou seja, se os alunos da instituição possuem interesse na aplicação e pretendem utilizar a mesma como ferramenta cotidiana para auxilia-los durante os estudos.

Para isto, houve a elaboração de um formulário por meio da ferramenta Google Forms, onde solicitamos que os alunos da instituição respondessem, com sinceridade, dez questões (Apêndice A) sobre a proposta. Sua divulgação ocorreu por meio do aplicativo de conversas WhatsApp, onde os integrantes da equipe ficaram responsáveis por enviar o endereço de compartilhamento do formulário nos grupos de alunos conhecidos na instituição.

A realização desta pesquisa é de grande importância para a equipe avaliar a proposta que está sendo apresentada e para analisar sua viabilidade por meio dos resultados.

1.2.1 Resultados

A partir deste formulário, foram recebidas **quarenta e cinco** respostas acumuladas dentro de um período de **cinco** dias, e ao final, foi possível constatar as seguintes características na maioria das respostas:

Algumas características do maior grupo que responderam ao questionário são:

- 86,7% dos estudantes são do ensino médio integrado ao técnico;
- 66,7% dos que sentiram dificuldade ao ingressar no IFSP, compartilhou em forma de resposta curta suas experiências (como, adaptação com as atividades, dificuldades em matérias específicas, falta de transparência nas informações institucionais);
- 60% nunca frequentaram ou vão raramente às monitorias;
- 85,5% usariam o sistema e acreditam que o mesmo o ajudaria academicamente;

Com base nesses resultados, a equipe entende que a proposta cumpre seu propósito de atingir os alunos do IFSP com o desenvolvimento de uma comunidade de apoio aos assuntos enfrentados durante sua formação.

1.3 Objetivo

Dadas as informações citadas anteriormente, o objetivo deste projeto é tentar instigar o interesse dos estudantes que compreendem o IFSP para poderem criar espaços colaborativos, por um sistema no qual os usuários interajam entre perguntas e respostas, fornecendo caminhos para o esclarecimento de suas dúvidas sobre a instituição de ensino, as áreas e disciplinas que a ela pertence.

Dessa forma, o objetivo será aplicado através da construção de uma plataforma de perguntas, respostas e mentorias para a Web, em que a comunidade interna poderá submeter uma pergunta para ser respondida pelos outros membros da comunidade; além de possibilitar que estudantes possam escolher se tornar mentores sobre determinados assuntos, disponibilizando recursos para a criação de anúncios de eventos de monitorias (cuja localidade a eles deve competir) dentro de seus perfis de usuário.

Tendo isso em vista e pensando numa melhor interatividade entre os membros da comunidade, o sistema deve passar por um processo de [gamificação](#) em algumas de suas funcionalidades, como as votações para respostas e perguntas mais relevantes e os atributos dos usuários mais ativos - assim como outros exemplos que devem ser adicionados durante o planejamento do projeto.

1.4 Justificativa

A reflexão com relação às formas complementares de aprendizagem é importante para a ampliação dos conteúdos interessados tanto aos alunos, quanto aos seus professores, pois permite que enxerguem, juntos, o ensino como um meio que evite a passagem de aprendizados de forma restrita e hierarquizada.

Por isso, [Fernandes \(2011\)](#) traz em sua pesquisa que o desafio da construção de sociedades de aprendizagem parte do pressuposto de que os recursos tecnológicos disponibilizados atualmente permitem aos estudantes aprenderem dentro e fora da escola e das mais variadas formas. Assim, para ele, a melhor forma se dá “construindo comunidades sustentadas pelo uso de tecnologias Web”.

O autor dá continuidade na exposição desse fenômeno ao atribuir o sucesso da potencialização da aprendizagem complementar e das relações sociais à “Web 2.0”. Isto, pois, de acordo [Fernandes \(2011\)](#), a mesma permitiu novas formas e possibilidades de criação de conteúdos e possibilitou o enfoque a uma aprendizagem motivada pelos interesses do

aluno, em que ele deve assumir um papel exploratório nessa experiência, da qual poderá colher ensinamentos significativos, explica [Fernandes \(2011\)](#).

Visando atrair atenção para o tema, o projeto tem como principal missão, permitir que os estudantes possam usufruir de uma ferramenta gratuita que proporcione a suavização do seu processo de aprendizagem, quando seus próprios colegas contribuirão com suas experiências passadas, além de deixarem um histórico para possibilitar um caminho menos árduo aos estudantes que virão. Por isso, espera-se que, com este projeto, a instituição de ensino também seja um agente na construção de uma comunidade propícia para estudantes, onde poderão unir-se em razão de dúvidas comuns, e assim incentivarem a disseminação de uma cultura colaborativa dentro de seus espaços.

1.5 Análise de Concorrência

Além disso, levou-se em consideração uma análise de concorrência com plataformas que apresentem características semelhantes ou até mesmo parecidas com as que gostaríamos de oferecer. Dessa forma, pretendemos demonstrar melhor como o nosso projeto se destaca em relação à concorrência e no que ele pode agregar.

Como primeiro concorrente pensamos no [Moodle](#) já que ele faz parte da vida de muitos alunos do [IFSP](#), pois ele é usado como ferramenta complementar de estudos. Ele fornece, nas disciplinas criadas pelos professores, a possibilidade de criar um fórum e também dispõem uma versão mobile.

Como segundo concorrente pensou-se no [Scoold](#) devido a ser uma inspiração para o nosso projeto, já que o mesmo possui quase todas as funcionalidades pensadas para o [IFriends](#), mas seu foco é em fornecer um espaço de perguntas e respostas.

Dessa forma, o [Quadro 1](#) apresenta a comparação dessas duas plataformas com o projeto de sistemas [IFriends](#):

Quadro 1 – Análise de Concorrência

Funcionalidades	IFriends	Moodle	Scoold
Comunicação pública (fórum) entre as partes envolvidas	●		●
Publicação de eventos	●		
Publicação de perguntas	●	●	●
Filtro de pesquisa no fórum de perguntas	●		●
Filtro de pesquisa nos eventos	●		
Gamificação	●		●
Acesso à plataforma por aplicativo móvel		●	

Fonte: Os autores

Devido ao curto tempo de desenvolvimento, uma questão que ficou em aberto, em relação às funcionalidades propostas no [Quadro 1](#) foi essa questão da versão *mobile*. Mas isso não foi considerado uma questão muito prejudicial para o desenvolvimento do projeto, visto que, o que levou a tal escolha foi a priorização em desenvolver melhor as outras funcionalidades, visto que a nossa missão visa a criação de uma comunidade que facilite e quebre essa barreira de interação entre as partes envolvidas.

Além disso, boas práticas de programação serão tomadas na elaboração do projeto, logo, a versão *mobile* poderá ser trabalhada futuramente sem muitos impedimentos.

2 Desenvolvimento

Baseando-se na conceituação de engenharia de software dada por Sommerville (2019), neste capítulo é descrito o desenvolvimento do sistema, cujas etapas estão apoiadas em técnicas que vão desde sua especificação até sua evolução.

Além do presente documento, o desenvolvimento do projeto ainda pode ser acompanhado através dos outros canais de registro que fazem parte dos requisitos da disciplina, sendo eles:

- **Aplicação:** A aplicação do projeto já hospedada na plataforma Heroku.

Figura 1 – Link da aplicação



Fonte: Os autores

- **Blog do projeto:** Pelo menos uma vez por semana, o blog do projeto recebe uma postagem a respeito do que foi desenvolvido naquele período.

Figura 2 – Link do blog do projeto



Fonte: Os autores

- **Canal no YouTube:** Através do canal no YouTube é possível conferir as apresentações realizadas pela equipe.

Figura 3 – Link do canal no Youtube



Fonte: Os autores

- **Repositório no SVN:** É o repositório principal do projeto, nele se encontram todos os arquivos do projeto, desde a documentação até os arquivos de desenvolvimento.

Figura 4 – Link do repositório



Fonte: Os autores

2.1 Materiais e Métodos

Segundo Júnior (2010), engenharia de software é “um conjunto integrado de métodos e ferramentas utilizadas para especificar, projetar, implementar e manter um sistema”, e que, para tanto, reúne em si metodologias, métodos e ferramentas para que o projeto seja bem definido em todas as suas etapas, variando desde sua problemática inicial e indo até sua entrega enquanto um produto de software. É a partir disto, que Júnior (2010) distingue um método de uma ferramenta:

Um método é uma prescrição explícita de como chegar a uma atividade requerida por um modelo de ciclo de vida, visando otimizar a execução das atividades que foram especificadas. Já as ferramentas proporcionam apoio automatizado ou semi-automatizado aos métodos (JÚNIOR, 2010).

Ademais, Júnior (2010) ainda separa os métodos de desenvolvimento de sistema em três categoriais que permitem visualizar e solucionar o problema de diferentes maneiras para sua modelagem. Para a composição do presente projeto, no entanto, será utilizado como método principal a metodologia escolhida para a gestão do projeto e as ferramentas poderão ou não ser relacionadas a mesma, como há de ser especificado nas próximas seções.

2.1.1 Tecnologias e ferramentas de desenvolvimento

Pensando no objetivo e alcance do projeto, foi decidida criação de uma aplicação Web, a qual pode ser acessada de diferentes dispositivos de forma fácil, apenas sendo tem como requisito básico o acesso à Internet. Outros fatores considerados são a experiência da equipe com o formato, os recursos computacionais exigidos para tal desenvolvimento (sendo menores do que para desenvolvimento de aplicações móveis) e a maior disponibilidade de conteúdos gratuitos e de fácil acesso sobre desenvolvimento Web.

De modo a desenvolver essa aplicação, considerou-se o uso do **Framework** JavaScript e ReactJs como base para o **Front-end**, já as bibliotecas estilísticas que servirão de auxílio serão o **React-Bootstrap** e o **Ant**.

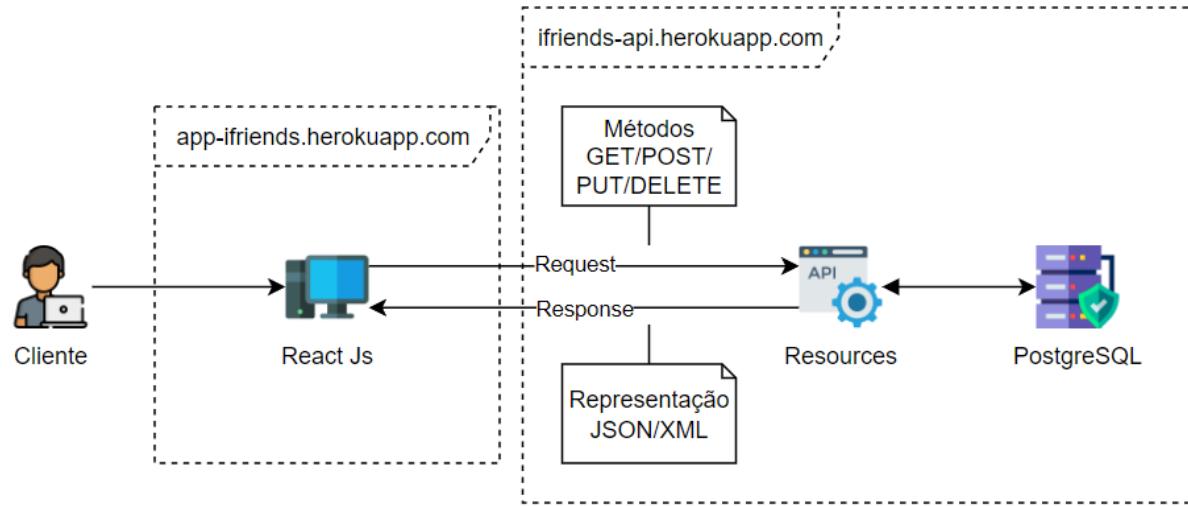
Além disso, para a internacionalização, faz-se uso da biblioteca de uso livre, a **i18n**, escrita em JavaScript e disponível como uma dependência no ReactJs. Essa biblioteca possui muitas funcionalidades e possibilidades de uso, porém a utilizamos para internacionalizar o site, com ela é possível traduzir texto para idiomas diversos, escolher a linguagem padrão do navegador do usuário e várias outras possibilidades.

Quanto ao banco de imagens da aplicação, utilizamos o **imgBB**, o qual é uma plataforma de hospedagem de imagens grátis, com ela é possível enviar uma requisição a **API** do **imgBB**, que junto de uma chave própria do usuário é possível fazer a inserção, edição, visualização e exclusão das fotos. Ele também nos fornece os links das imagens, o que permite que salvemos no nosso banco de dados apenas os links das fotos enviadas pelo usuário.

Para o **Back-end** foi considerada a linguagem Java, orientada a objetos; sua escolha foi feita pelo fato da equipe já possuir experiência com a linguagem, além de alguns já trabalharem com a mesma, ela também possui uma boa maturidade, contendo muitas informações disponíveis para ajudar no desenvolvimento. Em conjunto, será utilizado **Spring Boot**: **Framework** Java de código aberto, e será juntamente usado o **Maven**: **Framework** de automação e gerenciamento de dependências no projeto. Já como padrão de desenvolvimento utilizado no **Back-end**, será aplicado o padrão **MVC**, que separa o projeto em três principais camadas, *model*, *view* e *controller*.

No entanto, como arquitetura de sistema (**Figura 5**), foi escolhido utilizar a arquitetura **REST API** na qual é possível separar o código **Back-end** do **Front-end** de maneira que as duas aplicações possam determinar a troca informações entre elas com requisições via protocolo **HTTP**. Esta solução também é útil para escalar a plataforma, já que a utilização de uma **API** pode ser feita tanto por aplicações móveis, como para a Web.

Figura 5 – Arquitetura REST API



Fonte: Os autores.

Quanto ao banco de dados, foi utilizado o [PostgreSQL](#): um [SGBD](#) popular de [SQL](#), que serve para executar comandos no banco, como consultas, e alterações nos dados ou na estrutura.

Para hospedar a aplicação, esta se usando o [Heroku](#), o qual é uma plataforma de nuvem que suporta diversas linguagens, que permite a implantação, escalonamento e gerenciamento do sistema.

2.1.2 Ferramentas de apoio

Nesta seção serão especificados as ferramentas a serem utilizadas para a estruturação do sistema Web.

- **Notion:** Foi escolhido como plataforma de organização de projetos, pois este possui métodos de gerenciamento de equipe, fornecendo uma interface para vários desenvolvedores trabalharem utilizando o quadro de kanban, assim como o compartilhamento de arquivos, vídeos e artigos; permitindo também notificações via e-mail sobre atualizações do projeto e de reuniões marcadas, por exemplo.
- **Discord:** Plataforma de comunicação online escolhida pela equipe, por ser a mais utilizada e conhecida entre os mesmos. Nela é possível criar servidores privados para compartilhar informações e realizar reuniões sobre o projeto.
- **Overleaf:** Para a preparação do documento de visão, será usado o [LATEX](#), visto que este possui comando de padronização para documentos acadêmicos, facilitando a sua construção, juntamente, com editor Overleaf que oferece um ambiente compartilhado entre os membros da equipe.

- **Figma:** Na parte de modelagem do sistema e elaboração ideias de soluções de problemas, escolheu-se o Figma, visto que é uma ferramenta gratuita que possui diversas opções de edição.
- **Visual Studio Code:** A [IDE](#) escolhida para realizar a edição de código-fonte na parte *Front-end* do sistema, por razões de ser um recurso da atualidade com possibilidades de várias adaptações de ambiente, principalmente as nossas tecnologias escolhidas.
- **Eclipse:** A [IDE](#) escolhida para realizar a edição de código-fonte na parte *Back-end*, por motivo de facilidade que o ambiente de desenvolvimento proporciona para rodar aplicações com Java e, também, seus recursos que fazem a integração com o banco de dados.

2.2 Métodos de gestão e desenvolvimento

Tendo em vista a organização do projeto [IFriends](#), a equipe escolheu utilizar os princípios e valores do Manifesto Ágil como norteadores nos seus processos de planejamento, modelagem e desenvolvimento do projeto. Deste modo, elencou-se o [Framework Scrum](#) como metodologia ágil que irá servir como referência para o trabalho da Bunka Bytes.

2.2.1 Metodologia ágil: Scrum

Segundo [Ambler \(2004\)](#), o termo metodologias ágeis surgiu em fevereiro de 2001, quando especialistas em processos de engenharia de *software* se reuniram e estabeleceram princípios comuns entre todas as metodologias, criando a Aliança Ágil e o estabelecimento do “Manifesto Ágil” (*Agile Manifesto*).

O termo metodologia ágil consiste na otimização do tempo para a realização de determinado projeto, visando a rapidez na entrega e na qualidade do mesmo, surgindo assim, como uma resposta mais leve, mais assertiva e menos custosa em relação aos métodos pesados que eram utilizados para a construção de sistemas. Nas metodologias ágeis, os processos, as ferramentas, as documentações, as negociações e os planejamentos, possuem prioridade secundária, pois os indivíduos e suas interações são considerados essências e indispensáveis ([SGANDERLA, 2016](#)).

Para isso, o manifesto determina quatro valores principais, sendo eles: o enfoque nos indivíduos e nas interações, e não nos processos ou algoritmos; a adaptação e maior flexibilidade a novos fatores decorrentes do desenvolvimento do projeto; o foco na funcionalidade do sistema e na documentação mais simples e objetiva, e por último, a preferência

por um ambiente de trabalho mais colaborativo e menos burocrático ([SGANDERLA, 2016](#)).

Segundo [Cruz \(2018\)](#), o Scrum pode ser definido como “um *Framework* para desenvolver e manter produtos complexos que também pode ser utilizado no gerenciamento ágil de projetos que se destinam também à criação de produtos”. Neste caso, sabida a escolha da utilização de uma metodologia ágil para o gerenciamento do presente projeto, foi decidido aplicá-la com base no Scrum e suas especificações, porém, como a finalidade do trabalho é acadêmica, resolveu-se adaptar algumas de suas características para que a metodologia pudesse ser implementada como uma referência na organização e gestão do projeto. Estas, por conseguinte, serão explicitadas mais a frente, conforme apresentadas as particularidades do *Framework*.

Dentre as principais características do Scrum, está a divisão do desenvolvimento em ciclos repetitivos e curtos, permitindo modificações, adaptações e correções no produto de forma iterativa e incremental, o que, segundo [Cruz \(2018\)](#) permite encontrar desvios mais rápido e com menos impacto. [Cruz \(2018\)](#) explica que para que os processos sejam otimizados de tal forma, o Scrum possui ainda três pilares de sustentação: transparência, a inspeção e a adaptação.

Além dos pilares da metodologia, há cinco valores importantes para sua construção e prática durante um projeto: coragem, foco, comprometimento, respeito e abertura. De acordo com [Cruz \(2018\)](#) os valores do Scrum são responsáveis por reforçar os princípios do manifesto ágil, principalmente considerando o comportamento e as pessoas maiores do que os processos e ferramentas.

2.2.1.1 O quadro de kanban

Segundo [Peinado e Aguiar \(2007\)](#), o nome Kanban vem do japonês “cartão” e a sua origem deu-se pela seguinte razão:

Este nome surgiu em razão do sistema de controle visual dos estoques de materiais, pois são frequentemente utilizados cartões para representar os contentores cheios ou vazios, estes cartões são retirados ou colocados em um quadro à medida que o material é utilizado, ou reposto ([PEINADO; AGUIAR, 2007](#)).

E através da implantação realizada por [Silva, Santos e Neto \(2012\)](#), devido aos problemas e necessidades encontrados ao longo dos *sprints*, houve a criação de um processo ágil, baseado nas práticas do Scrum com características do Kanban:

As tarefas ou itens de trabalho foram representadas através de cartões (do inglês *post-its*) fixados em um quadro (do inglês *cardwall*). Esse quadro, por sua vez, era dividido em colunas que representavam as fases do fluxo de trabalho (do inglês *workflow*) da equipe. As tarefas eram

distribuídas sequencialmente nas colunas à medida que avançavam no fluxo de trabalho ([SILVA, 2012](#)).

No projeto a implementação do kanban foi similar, porém todo o esquema presencial foi adaptado ao quadro remoto. Para conciliar a participação continua da equipe, a ferramenta *Notion* é utilizada como auxiliar aos processos de organização, onde também foi possível criar o quadro de kanban do projeto.

O quadro de kanban usado pela equipe Bunka Bytes, responsável pelo projeto apresenta 5 colunas, nas quais são representadas visualmente em qual estagio a tarefa se encontra, as colunas se classificam como “para fazer”, “planejamento”, “em andamento”, “para revisar” e “feito”. E por “cartões virtuais” são colocadas as tarefas no quadro.

Foi decidido utilizar uma classificação em ordem de prioridade para cada tarefa, podendo ser “alta”, representada pela cor vermelha, “média”, representada pela cor laranja, e “baixa” representada pela cor verde, tais prioridades são atribuídas às tarefas assim que a *sprint* é definida.

De maneira geral, o uso do quadro de kanban é benéfico ao projeto, já que através de sua aplicação a equipe consegue conciliar de maneira clara e precisa as tarefas compartilhadas. A utilização do quadro remotamente, faz com que a equipe possa ainda acessá-lo de qualquer lugar, facilitando o processo de transparência dos itens e faz com que todos tenham em suas mãos as atividades a serem feitas, podendo ajustá-las no momento em que preferirem.

2.3 Gestão da equipe

A equipe Scrum é composta por três papéis: o *Scrum Master*, o *Product Owner* e o Time de Desenvolvimento:

O primeiro, chamado *Scrum Master*, é considerado o responsável por garantir que o Scrum seja entendido e aplicado, para que o Time Scrum esteja aderindo os valores, as práticas e as regras do Scrum, e, portanto, trabalha como um líder ou técnico da equipe. Já o *Product Owner*, é o principal responsável pelo gerenciamento do *backlog* do produto, por garantir o valor do trabalho realizado pelo Time, e pela satisfação e atendimento das necessidades do cliente. O Time de Desenvolvimento, por outro lado, é responsável por executar o desenvolvimento e transformar o *backlog* do produto em incrementos de funcionalidades, criando um sistema pronto que possa ser entregue ao cliente ([CRUZ, 2018](#)).

Portanto, para fins de organização, o Time de Desenvolvimento poderá ser dividido em subequipes, sendo elas: desenvolvimento ([Front-end](#) e [Back-end](#)), design e documentação, e mídias. Nelas, estão inclusas as funções de programadores [Front-end](#) e [Back-end](#), divididas entre os integrantes Jamilli Vitória Gioielli, José Roberto Claudinho Ferreira

e Kaiky Matsumoto Silva; sendo a Jamilli responsável por supervisionar o *Front-end* da aplicação, e o José (que também auxiliará no *Front-end*) juntamente com o Kaiky pelo *Back-end* - que também inclui a administração do banco de dados.

Além dessas, as outras duas subequipes foram divididas entre as integrantes Anaí Villca Rojas, responsável por supervisionar o design de experiência/interface de usuário; e Julia Romualdo Pereira, responsável pelo gerenciamento da documentação e das mídias do projeto (como o canal no [YouTube](#) e as postagens no Blog).

De todo modo, a documentação e as mídias do projeto deverão ser revisadas, obrigatoriamente, por todo o time, de modo a manter um conhecimento sobre a aplicação melhormente distribuída. Entretanto, precisa-se compreender que a modelagem de dados e a elicitação de requisitos da aplicação serão feitas com o auxílio de todos os integrantes, visto que são partes de extrema importância para que a compreensão de todos sobre o cenário corresponda com o objetivo principal pretendido. No [Quadro 2](#), é possível observar uma relação das supervisões dos integrantes em cada área do projeto.

Quadro 2 – Distribuição de tarefas

Responsável	Front-end	Back-end	Documentação e Mídias	Design
Anaí			●	●
Jamilli	●			●
José	●	●		
Julia			●	
Kaiky		●		

Fonte: Os autores.

Entretanto, é importante salientar que as funções atribuídas a cada um não são exclusivas e podem variar conforme a necessidade de entrega do projeto. As subequipes são de responsabilidade de todos os integrantes e apenas foram divididas assim para fins de organização das tarefas do projeto. Levando isso em consideração, as funções de *Scrum Master* ou *Iteration Manager* e *Product Owner* foram adaptadas, ainda que não seja o ideal na metodologia ágil, para um único papel, representado pela integrante Jamilli Vitória Gioielli. Porém, vale ressaltar ainda que, nesse modelo escolhido, toda a equipe é responsável pela inspeção dos processos do projeto, e não somente o *Scrum Master* ou *Iteration Manager*:

O Time deve ser multidisciplinar e multifuncional, possuindo todo o conhecimento necessário para criar um incremento no trabalho. [...] Não

há títulos no Time, e não há exceção a esta regra. Não deve existir distinção de cargos ou funções, títulos ou senioridades, e muito menos áreas determinadas ou específicas de atuação. No Scrum todos os integrantes do Time são conhecidos como desenvolvedores.

Individualmente os integrantes do Time de Desenvolvimento podem ter habilidades específicas, mas, independentemente disso, a responsabilidade a respeito de uma entrega continua sendo do Time de Desenvolvimento como um todo (CRUZ, 2018).

Para organizar a produtividade continua de desenvolvimento do projeto, buscando seguir uma das vertentes propostas na metodologia ágil de entregas semanais, a equipe criou um cronograma de entregas por aula baseado a partir do plano de ensino da disciplina disponibilizado pelos orientadores, ou seja, a equipe busca realizar durante a semana a atividade proposta para a aula seguinte para aproveitar o tempo em sala de aula validando o que foi realizado com os orientadores.

O cronograma de entrega semanal utilizado para o desenvolvimento do projeto IFriends pode ser observado na [Figura 6](#).

Figura 6 – Cronograma de Desenvolvimento Semanal

Conteúdo Planejado	Data / Semana	<input checked="" type="checkbox"/> Post no Bloq?	# Qtd de Reuniões	Resumo das entregas/atividades
Apresentação e definição dos procedimentos durante o ano letivo. Divisão das equipes.	14/03/2022	<input checked="" type="checkbox"/>	3	- Brainstorm e conversas iniciais sobre as ideias
Análise de temas de trabalhos	21/03/2022	<input checked="" type="checkbox"/>	3	- Acesso.txt - Documento sobre temas
Estudo de viabilidade	28/03/2022	<input checked="" type="checkbox"/>	2	- Inscrição do blog no feedburner - Criação da pesquisa de viabilidade
Análise de tecnologias	02/04/2022	<input checked="" type="checkbox"/>	2	- Resumo das tecnologias a serem utilizadas
Proposição de temas de trabalho	04/04/2022	<input checked="" type="checkbox"/>	3	- Divulgação da pesquisa - Análise dos trabalhos anteriores - Criação do documento para proposta
Apresentação das propostas iniciais	11/04/2022	<input checked="" type="checkbox"/>	2	- Apresentação da proposta - Envio do documento da proposta e da análise dos outros projetos - Envio dos Slides
Técnicas de gerenciamento de projeto	18/04/2022	<input checked="" type="checkbox"/>	2	- Entrega da análise sobre as outras equipes - Entrega do vídeo da proposta - Apresentação do rascunho do backlog do produto
Requisitos de negócio	25/04/2022	<input checked="" type="checkbox"/>	2	- Dúvidas sobre as histórias de usuário - Revisão da modelagem e das regras de negócio - Início da prototipagem
Modelagem de dados	02/05/2022	<input checked="" type="checkbox"/>	1	- Mostramos o protótipo e o fluxo do sistema - Mostramos a versão do banco de dados para a PoC

Fonte: Os autores.

2.3.1 Desenvolvimento: artefatos e eventos

Tendo em vista a finalidade acadêmica do projeto, foi estipulado um valor mínimo de duas reuniões semanais dedicadas na construção do mesmo, considerando a organização individual de cada membro da equipe, além da priorização para que a tomada de decisões e planejamentos sejam feitas no dia das aulas da disciplina de [PDS](#). Sobre os artefatos

e eventos do Scrum, lista-se a seguir suas funcionalidades originais, descritos por [Cruz \(2018\)](#), e como serão adaptados:

- *Backlog*: é descrito como uma lista de todas as características, funções, tecnologias, melhorias e correções que constituem o produto a ser entregue. É também subdividido em “do produto”, “da versão de entrega” e “da Sprint”;
- *Sprint*: originalmente pensada para durar um mês ou menos e possuir uma meta estabelecida com um objetivo claro, foi pensada para durar até duas semanas, já que as orientações da disciplina exigem que existam entregas frequentes, isto é, todas as semanas. Portanto, o ideal é que se dê início ao trabalho a ser entregue pelo menos uma semana antes de sua entrega;
- *Time-boxed*: é esperado que o tempo estipulado para executar uma tarefa seja cumprido e que o trabalho proposto, seja realizado. Tendo em vista os curtos prazos, também poderá ser adaptado conforme a entrega;
- Planejamento da *Sprint*: ainda será utilizado para definir “o que será feito” e “como”, porém, ao contrário das oito horas mínimas estipuladas, a equipe definiu um mínimo de duas horas para tal reunião, que poderá acontecer nas segundas-feiras, durante a aula de [PDS](#) ou aos finais de semana, logo após a finalização dos entregáveis;
- Reuniões Diárias: não serão mantidas de maneira síncrona, visto que cada membro da equipe possui horários de disponibilidade diferentes. Porém, poderão ocorrer de forma assíncrona, de modo a compreender se existem impedimentos durante a execução das tarefas planejadas para a semana;
- Revisão da *Sprint*: seu maior objetivo é a revisão do *Product Owner*, ou do cliente, em todos os itens concluídos pelo Time. Porém, não terá *Time-boxed* de quatro horas, visto que a aprovação dos professores orientadores (o cliente) pode ser mais rápida e acontecer durante a execução das tarefas. No entanto, o Time a realizará em todo final de entrega, para que os retornos dos professores levem em consideração o trabalho final;
- Retrospectiva da *Sprint*: possui originalmente *Time-boxed* de até três horas e é feita para identificação de medidas de melhoria no processo do time que serão aplicadas na próxima *Sprint*. Foi escolhido adaptá-la para acontecer às terças-feiras, antes do início das aulas, para que as entregas feitas às segundas-feiras estejam frescas e o processo realizado pelo time possa ser melhor avaliado para a melhoria contínua.

Para a definição do *backlog* do produto, o Scrum conta com um recurso chamado histórias de usuário, que pode ser definido como:

História é uma descrição resumida, porém clara e objetiva, de alguma funcionalidade que deverá ser fornecida pelo produto a ser entregue, sempre do ponto de vista do usuário final. Uma história não é uma especificação completa da funcionalidade, mas uma promessa de discutir uma funcionalidade, ou, simplesmente, um lembrete de que a discussão já aconteceu. Um modelo simples de como escrever uma história seria: “Como um <tipo de usuário>, eu quero <um objetivo> para que <atenda a uma necessidade>” (CRUZ, 2018).

Entretanto, vale ressaltar que as histórias de usuário do presente projeto serão montadas somente na parte de desenvolvimento, isto é, quando o produto começar a ser construído pelo Time Scrum e após os principais itens de *backlog* estarem descritos. Como ainda se encontra na fase de planejamento e modelagem, tal recurso ainda não foi acionado no projeto.

Outra ressalva diz respeito a utilização do *Scrum Poker* ou *Planning Poker Card*, que, segundo Cruz (2018), é definido como “uma técnica que auxilia na estimativa de histórias e tarefas com base no consenso de todo o Time”. Para tanto, o Time utiliza um conjunto de cartas com valores representando os pontos ou horas por história. Sobre o funcionamento do *Planning Poker Card*:

O seu uso é simples: o *Product Owner* ou um membro do Time apresenta a história, ou tarefa. Após uma breve discussão, cada um escolhe uma carta e a coloca virada sobre uma mesa, de forma que um não constate o valor da carta que o outro escolheu. Quando todos colocarem suas cartas, elas são desviradas para que todos vejam os valores.

Caso não haja consenso entre as cartas escolhidas, as diferenças são discutidas brevemente e uma nova rodada acontece, até que haja convergência e consenso (CRUZ, 2018).

Para o presente projeto, no entanto, o *Planning Poker Card* será utilizado durante o processo de desenvolvimento do produto e após as histórias de usuário da *sprint* estarem bem descritas. Vale ressaltar que este processo é apenas para estimar o trabalho da equipe e pode variar conforme o andamento do projeto.

2.3.2 Métricas de Desenvolvimento

De modo a fazer nosso processo de desenvolvimento o mais produtivo possível, e entregar o máximo do *backlog* do produto, foram definidas as seguintes pontuações (Quadro 3) para definir o grau de complexidade das tarefas de modo a priorizar o desenvolvimento daquelas que exijam um trabalho maior. Essas pontuações foram baseadas no princípio do *Planning Poker Card*, no qual a sequência das cartas usadas para a estimativa seguem a sequência de *Fibonacci*.

Quadro 3 – Métricas de Organização das Histórias

Pontuação	Tamanho	Descrição
< 5	Pequeno	Pontuação usada para classificar tarefas que são de nível fácil e que podem ser realizadas em um curto período de tempo.
5 e 8	Médio	Pontuação usada para classificar tarefas de nível médio e exigem um desempenho maior comparada com as de nível fácil, mas não chegam a ser muito complexas.
≥ 13	Grande	Pontuação usada para classificar tarefas que são de nível difícil e exigem um grande desempenho, além de precisarem de mais tempo para serem desenvolvidas.

Fonte: Os autores

Esses critérios de pontuação foram aplicados as histórias de usuário ([Apêndice B](#)), portanto, com o uso dessas métricas é possível priorizar tarefas mais complexas (com maior pontuação) das mais simples (com menor pontuação) além de ajudar na execução das [Sprints](#). A aplicação dessas métricas pode ser constada na figura [Figura 7](#).

Figura 7 – Histórias no Notion

Name	Description	Acceptance Criteria	Epic	Priority	Points	Size
Manter uma pergunta	Como aluno, eu gostaria de manter uma pergunta na comunidade para retirar uma dúvida	<ul style="list-style-type: none"> Mostrar "como fazer uma boa pergunta". O usuário deve conseguir somente criar e remover uma pergunta da visualização. O usuário deve conseguir fechar o espaço de respostas para a pergunta. 	Gestão de perguntas	Alta	13	Grande
Adicionar assuntos	Como aluno, eu gostaria de adicionar assuntos nas minhas perguntas para encontrá-las mais facilmente	<ul style="list-style-type: none"> Devem existir tipos de tags (exemplo: o tipo é informática, que possui várias tags); As tags serão divididas em uma lista pré-definida de acordo com seu tipo e um campo "outras" para tags que não existem. 	Gestão de perguntas	Alta	3	Pequena
Curtir uma pergunta	Como aluno, eu gostaria de votar em uma pergunta para	<ul style="list-style-type: none"> Um usuário só poderá votar numa única vez. 	Gestão de perguntas	Alta	2	Pequena

Fonte: Os autores.

A sua utilização se deu através de uma reunião realizada por todos os membros da equipe e como o auxílio do [Scrum Poker Online](#) que nos forneceu as ferramentas necessárias para realizar a estimativa de pontuação.

2.4 Análise de Requisitos

Segundo Machado (2018), ao definir as características de um requisito, é preciso salientar que não são dependentes da tecnologia empregada, visto que suas especificações estão contidas no campo do cumprimento das necessidades do usuário. Dessa forma, Machado (2018) define os requisitos como “objetivos ou restrições estabelecidas por clientes e usuários do sistema que definem suas diversas propriedades”.

Assim, tanto Machado (2018) como Sommerville (2019) concordam que a fase de definição de requisitos, a chamada engenharia de requisitos, é essencial para a tomada de decisões sobre os passos para adquirir ou desenvolver o sistema. Por outro lado, Sommerville ainda acrescenta sobre a necessidade de mudança durante o desenvolvimento:

Naturalmente, são feitas mudanças subsequentes nos requisitos de usuário, que podem ser ampliados para requisitos de sistema mais detalhados. Às vezes, pode-se utilizar uma abordagem ágil para eliciar simultaneamente os requisitos à medida que o sistema é desenvolvido, a fim de acrescentar detalhes e refinar os requisitos de usuário (SOMMERVILLE, 2019).

Tendo tais definições em vista, as próximas seções visam apresentar os requisitos funcionais e não funcionais, e as regras do negócio característicos do projeto tratado neste documento.

2.4.1 Histórias de Usuário

Segundo Cruz (2018) as histórias de usuário se caracterizam como uma descrição resumida, clara e objetiva de uma funcionalidade fornecida pelo produto a ser entregue, visando o ponto de vista final do usuário. Ainda segundo o autor para uma história ser tida como completa, ela deve possuir uma descrição objetiva e critérios de aceitação, esses critérios representam o que ela precisa fazer para ser considerada válida.

No projeto, a equipe aproveitou as histórias de usuário para representar os requisitos funcionais e não funcionais, dessa forma os requisitos funcionais podem ser identificados a partir do nome das histórias, e o não funcionais por meio dos critérios de aceitação definidos para tais.

Todas as histórias se encontram no Apêndice B e apresentam sete componentes: o seu nome, a sua descrição, seus critérios de aceitação, o épico a qual pertence, a pontuação que ela recebeu no *Planning Poker Card*, a estimativa de tamanho conforme a sua pontuação e a sua prioridade conforme o seu tamanho e pontuação. Ainda com o fim de organizá-las foram separadas por meio dos seguintes épicos:

- **Gestão de Perguntas:** Para este épico foram destinadas às histórias de usuário do [Quadro 4](#). Logo, os devidos detalhes de cada história de usuário se encontra na [seção B.1](#) do apêndice.

Quadro 4 – Resumo: Gestão de perguntas

História	Pontuação	Tamanho	Prioridade
Manter uma pergunta	13	Grande	ALTA
Filtrar perguntas	8	Médio	MÉDIA
Adicionar assuntos	3	Pequeno	ALTA
Curtir uma pergunta	2	Pequeno	ALTA
Buscar perguntas	2	Pequeno	MÉDIA

Fonte: Os autores

Como a primeira história a ser desenvolvida pela equipe se encontrava nesse épico, era necessário considerar também a configuração de ambiente, dessa forma, ele acabou recebendo uma estimativa global de esforço maior, em relação aos outros, totalizando 28 pontos.

- **Gestão de Respostas:** Para este épico foram destinadas às histórias de usuário do [Quadro 5](#). Logo, os devidos detalhes de cada história de usuário se encontra na [seção B.2](#) do apêndice.

Quadro 5 – Resumo: Gestão de respostas

História	Pontuação	Tamanho	Prioridade
Manter uma resposta	5	Médio	ALTA
Curtir uma resposta	1	Pequeno	ALTA

Fonte: Os autores

Se tratando do segundo épico a ser trabalhado, a sua estimativa global de esforço foi menor em relação ao primeiro, mas nele ainda foi mantida a prioridade alta, dessa forma, o épico totalizou 6 pontos.

- **Gestão de Eventos:** Para este épico foram destinadas às histórias de usuário do [Quadro 6](#). Logo, os devidos detalhes de cada história de usuário se encontra na [seção B.3](#) do apêndice.

Quadro 6 – Resumo: Gestão de eventos

História	Pontuação	Tamanho	Prioridade
Manter evento	3	Pequeno	BAIXA
Favoritar evento	2	Pequeno	BAIXA

Fonte: Os autores

O épico totalizou 5 pontos, e suas histórias foram classificadas com tamanho pequeno devido à semelhança no desenvolvimento das histórias anteriores, dessa forma, foram denotadas como histórias de usuário nível fácil.

- **Gestão de Usuários:** Para este épico foi destinada à história de usuário do [Quadro 7](#). Logo, seus devidos detalhes se encontram na [seção B.4](#) do apêndice.

Quadro 7 – Resumo: Gestão de usuários

História	Pontuação	Tamanho	Prioridade
Autenticação do usuário	5	Médio	ALTA

Fonte: Os autores

O épico é responsável pelo quesito de autenticação do sistema, no entanto, por não ser uma tarefa muito complexa totalizou 5 pontos como estimativa de esforço.

- **Usabilidade:** Para este épico foi destinada à história de usuário do [Quadro 8](#). Logo, seus devidos detalhes se encontram na [seção B.5](#) do apêndice.

Quadro 8 – Resumo: Usabilidade

História	Pontuação	Tamanho	Prioridade
Internacionalização	5	Médio	ALTA

Fonte: Os autores

O épico é responsável para classificar entregas relacionadas a usabilidade do sistema, sendo a única mapeada até o momento, criada para a entrega do valor de disponibilidade de tradução dos conteúdos em português e inglês. Logo, o épico totalizou 5 pontos como estimativa de esforço.

2.4.2 Regras de Negócio

Regras de negócio são requisitos de domínio de aplicação tratado no desenvolvimento de um sistema, isso significa, as declarações sobre como determinada empresa faz negócio. É a partir dessas regras que se define como o negócio funciona e quais são suas especificações, além da sua importância para o desenvolvimento de um sistema, pois, auxiliam no controle dos processos, ajudam na tomada de decisões além de afetarem diretamente os requisitos funcionais, como descrevem [Dallavalle e Cazarini \(2000\)](#). Dessa forma, o [Quadro 9](#) lista as regras de negócio levantadas para o projeto [IFriends](#).

Quadro 9 – Regras de Negócio

ID	Descrição
RN01	Não serão permitidos usuários com os mesmos dados, ou seja, o sistema só permitirá a criação de uma conta por usuário
RN02	A fim de garantir a segurança de nossos usuários, o sistema deverá fazer uso de algumas diretrizes da Lei Marco Civil da Internet, lei nº12.965/2014, que tem como objetivo estabelecer princípios, garantias, direitos e deveres para o uso da internet no Brasil.
RN03	Dentro do direito a exclusão, ao excluir seu perfil, o usuário deve ter ciência de que suas perguntas e respostas serão mantidas na comunidade, porém como parte de um usuário anônimo (exemplo: user12345678).
RN04	O usuário deve levar em consideração as boas prática para o desenvolvimento de uma pergunta (Apêndice C).

Fonte: os autores

2.4.3 Partes interessadas e usuários

Segundo [Pontotel \(2021\)](#) por *stakeholders* ou partes interessadas se entende todas as organizações ou grupo de pessoas que possam apresentar interesse ou serem afetadas pelas ações de uma determinada empresa. Nas partes interessadas podem se classificar desde colaboradores, tidos como *stakeholders* internos, até fornecedores, investidores, clientes e comunidades, tidos como externos.

Dessa forma, no [Quadro 10](#) foram representados os *stakeholders* considerados para o desenvolvimento do projeto [IFriends](#).

Quadro 10 – Partes interessadas

Nome	Nível de influência	Interesses e Expectativas
Discentes	Alto	Participar do fórum de perguntas e respostas, assim como gerenciar os eventos
Docentes	Médio	Acompanhamento extracurricular das dificuldades relacionadas a respeito do conteúdo lecionado.
Coordenação e Direção	Médio	Regulação no uso adequado do sistema, ou seja, se ele está cumprindo o que foi planejado inicialmente.
Pais e responsáveis	Médio	Estar a par dos meios extracurriculares aderidos pela instituição.

Fonte: Os autores

Já por usuário, segundo [blog \(2017\)](#), são caracterizados aqueles que usam um determinado sistema, não sendo necessário um conhecimento técnico completo para compreender e executá-lo. Dessa forma, o [Quadro 11](#) apresenta todos os possíveis usuários que possam ter acesso ao projeto de sistemas [IFriends](#).

Quadro 11 – Usuários

Nome	Descrição	Ações
Friends	Dicentes que tenham interesse no fórum e nos eventos.	Este tipo de usuário pode visualizar, responder, curtir, denunciar, cadastrar perguntas e respostas no fórum além de poder realizar as mesmas ações na seção de eventos.
Docentes	Professores que tenham interesse em realizar acompanhamento de seus alunos por meio do IFriends .	Este tipo de usuário possui as mesmas ações que o tipo de usuário Friends .
Usuários externos	Pessoas que desejem conhecer a comunidade e que não façam parte da instituição.	Este tipo de usuário terá apenas a possibilidade de visualizar páginas públicas.

Fonte: Os autores

O termo [Friends](#) surge devido a um apelido destinado aos usuários do sistema [IFriends](#), além de ele nos ajudar a demonstrar melhor as funcionalidades do sistema, o mesmo também nos permite demonstrar a importância e o carinho desempenhado pela equipe na produção deste projeto.

2.5 Modelagem

Segundo [Sommerville \(2019\)](#), a modelagem de sistemas é definida como “um processo de desenvolvimento de modelos abstratos de um sistema, cada um apresentando

uma visão diferente do mesmo”. Para isso, descreve também que são geralmente usadas notações gráficas baseadas nos tipos de diagrama em UML durante o processo de engenharia de requisitos “para ajudar a derivar os requisitos detalhados de um sistema; durante o processo [...]”; e depois da implementação, para documentar a estrutura e operação do sistema” (SOMMERVILLE, 2019).

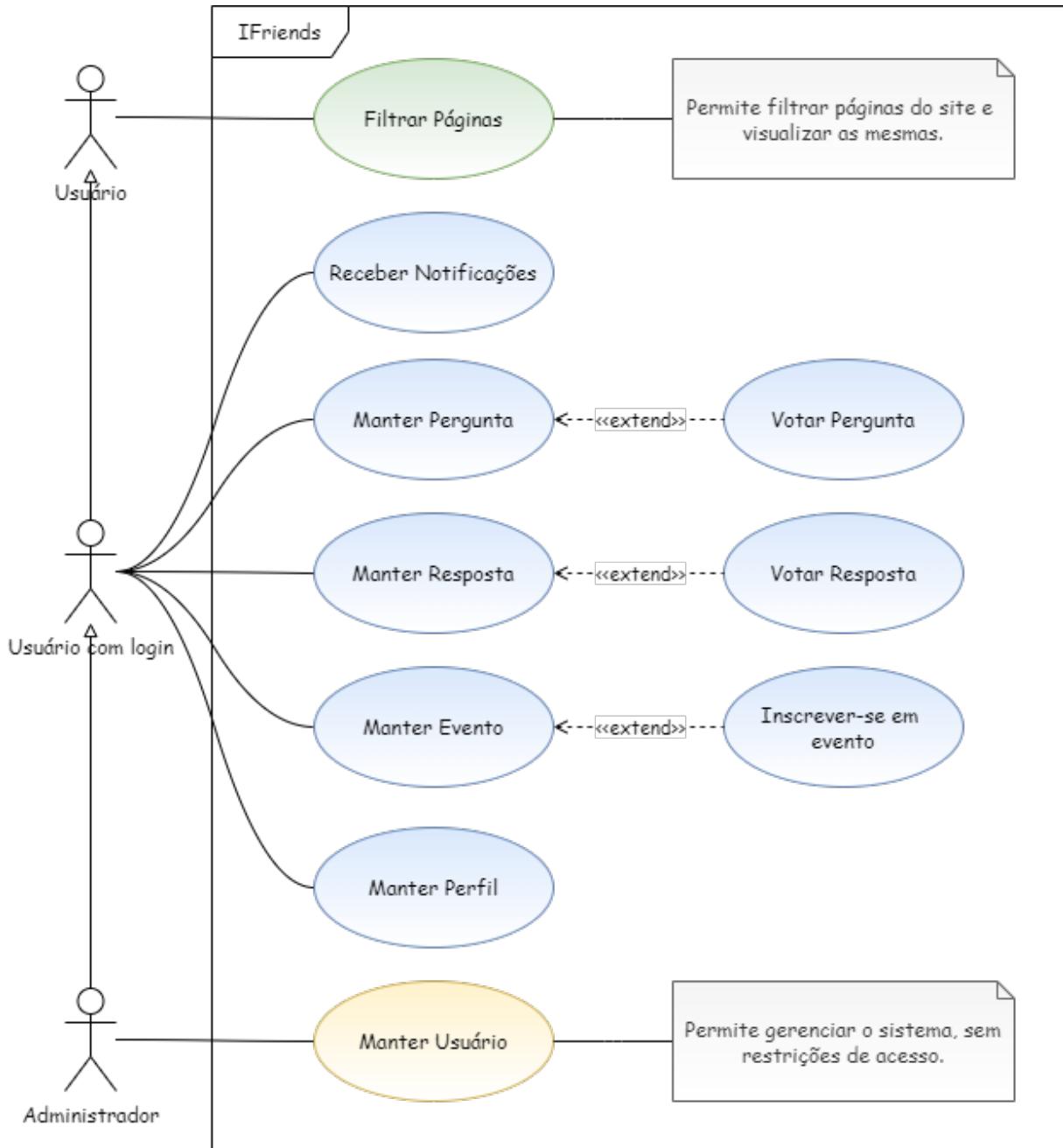
2.5.1 Diagrama de Casos de Uso

De acordo com Guedes (2011), o diagrama de casos de uso UML é descrito como:

O diagrama de casos de uso [...] tem por objetivo apresentar uma visão externa geral das funcionalidades que o sistema deverá oferecer aos usuários, sem se preocupar com a questão de como tais funcionalidades serão implementadas. [...] É de grande auxílio para a identificação e compreensão dos requisitos do sistema, ajudando a especificar, visualizar e documentar as características, funções e serviços do sistema desejados pelo usuário (GUEDES, 2011).

Logo, a Figura 8 representa os casos de uso do projeto de sistema IFriends.

Figura 8 – Diagrama de Casos de Uso



Fonte: Os autores.

2.5.2 Diagrama de Classes

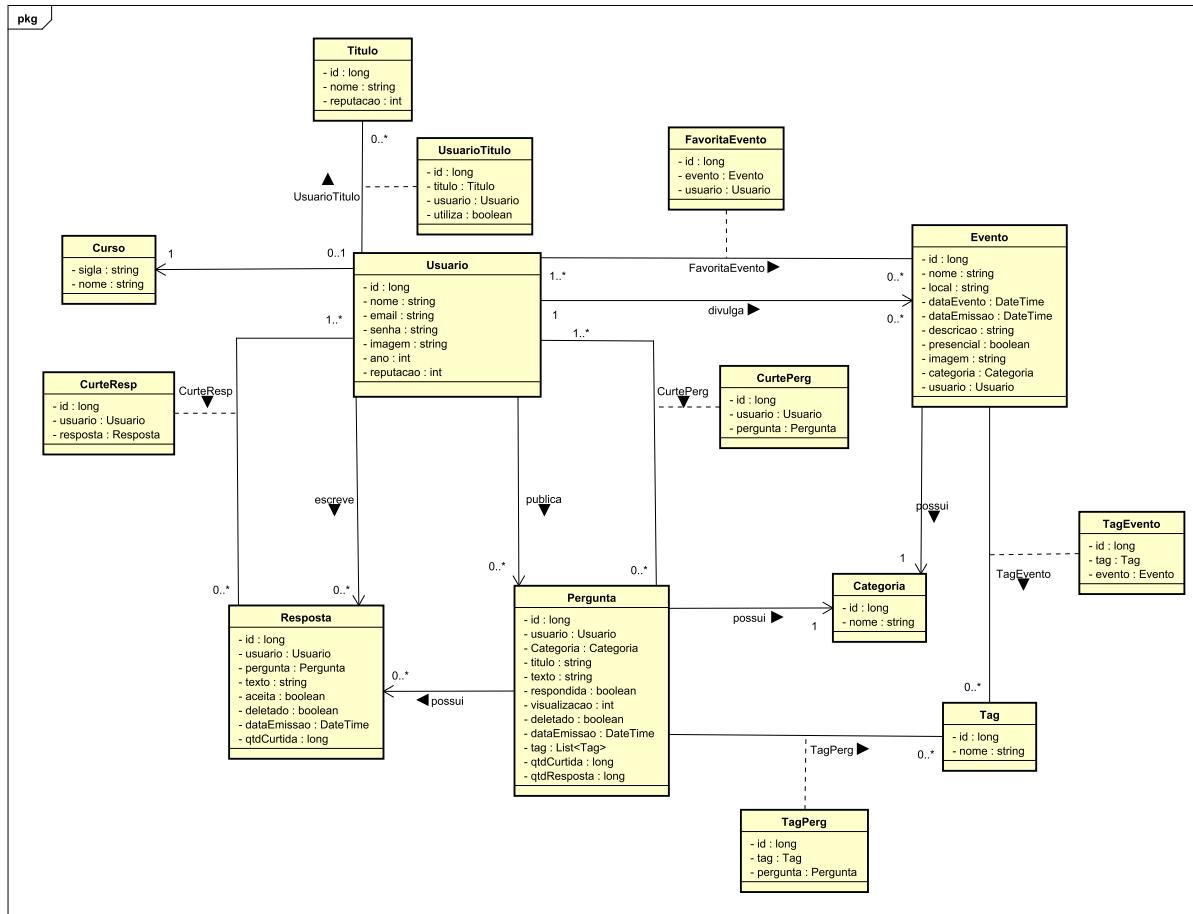
Segundo Sommerville (2019) os diagramas de classe são usados no desenvolvimento de um sistema orientado a objetos para realizar a demonstração das classes e suas associações entre elas. O autor ainda complementa com a seguinte ideia geral:

De maneira geral, uma classe pode ser encarada como uma definição geral de um tipo de objeto de sistema. Uma associação é um vínculo entre classes, que indica a existência de um relacionamento entre elas.

Consequentemente, cada classe pode precisar de alguns conhecimentos a respeito de sua classe associada ([SOMMERVILLE, 2019](#)).

Dessa forma, a [Figura 9](#) representa as entidades do diagrama de Classes do projeto IFriends.

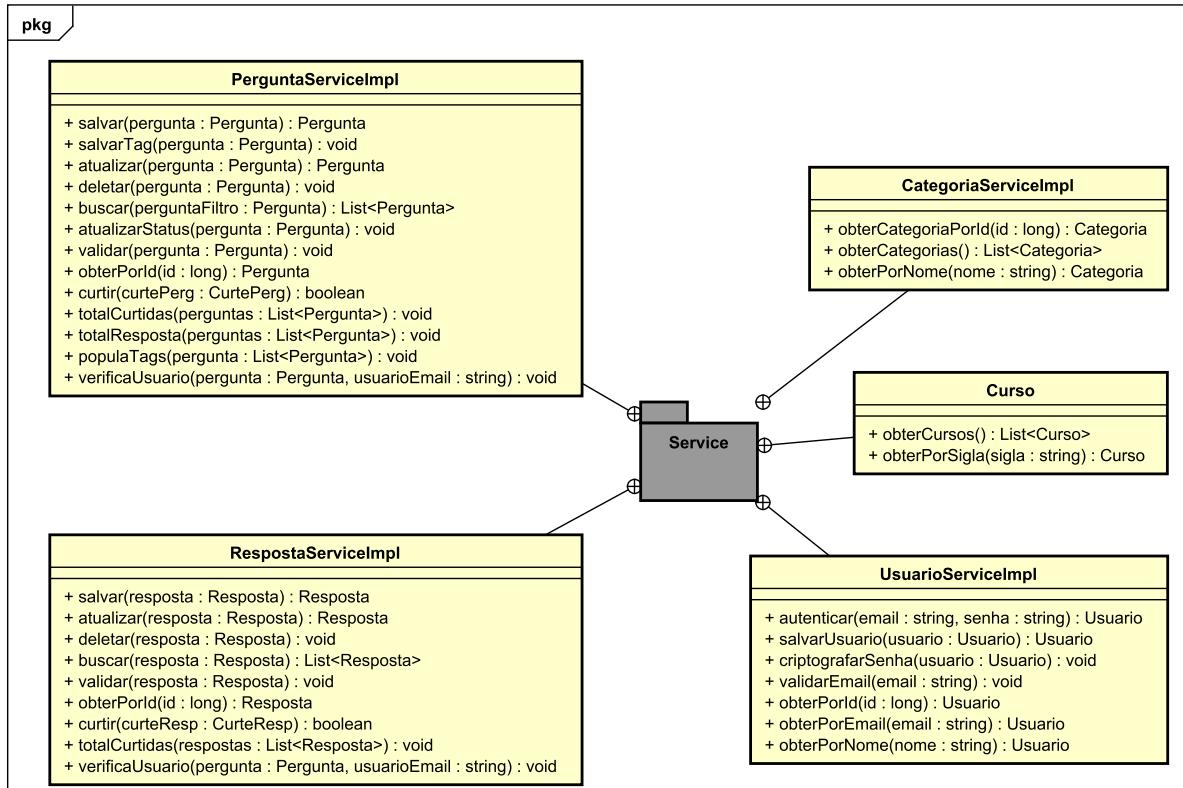
Figura 9 – Diagrama de Classes



Fonte: Os autores.

Já a [Figura 10](#) representa as classes de serviço e seus respectivos métodos dentro da *Application Programming Interface (API)* IFriends.

Figura 10 – Diagrama de Classes



Fonte: Os autores.

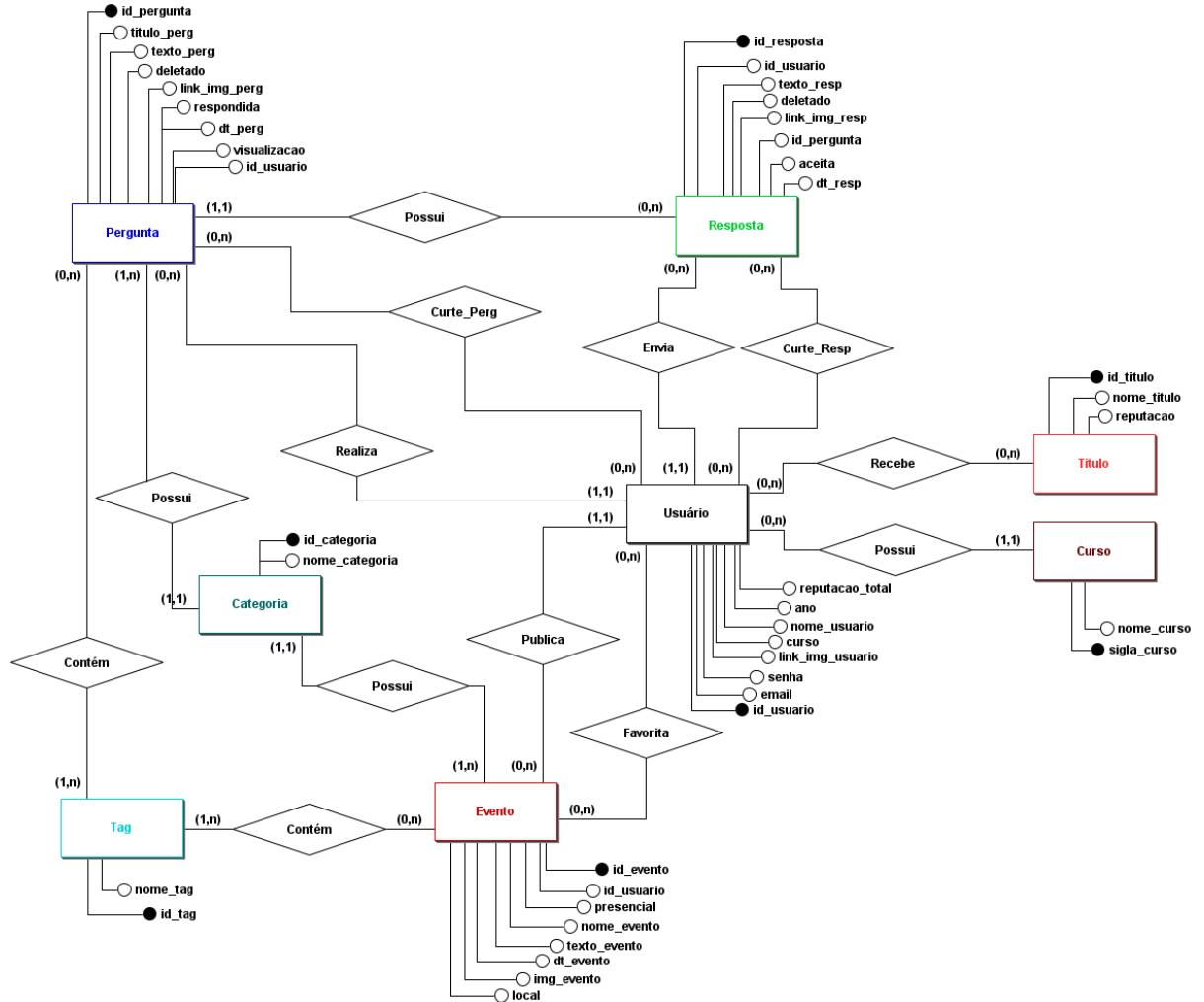
2.5.3 Diagrama de Entidade e Relacionamento

De acordo com [Leal \(2015\)](#), a abordagem entidade-relacionamento é a técnica de modelagem de dados mais difundida e utilizada e representa a modelo conceitual do banco de dados. Nela, a estrutura do banco de dados é descrita como coleção de entidades, relacionamentos e representada graficamente por meio do Diagrama Entidade Relacionamento.

Através dele, é possível descrever um subconjunto do mundo real que será retratado no banco de dados com um alto nível de abstração. Além disso, o modelo Entidade Relacionamento é um modelo formal e caracteriza-se por ter uma grande capacidade semântica, o que garante que todos possam ter o mesmo entendimento ([LEAL, 2015](#)).

A [Figura 11](#) representa o Diagrama Entidade Relacionamento (DER) do projeto de sistema [IFriends](#).

Figura 11 – Diagrama de Entidade e Relacionamento



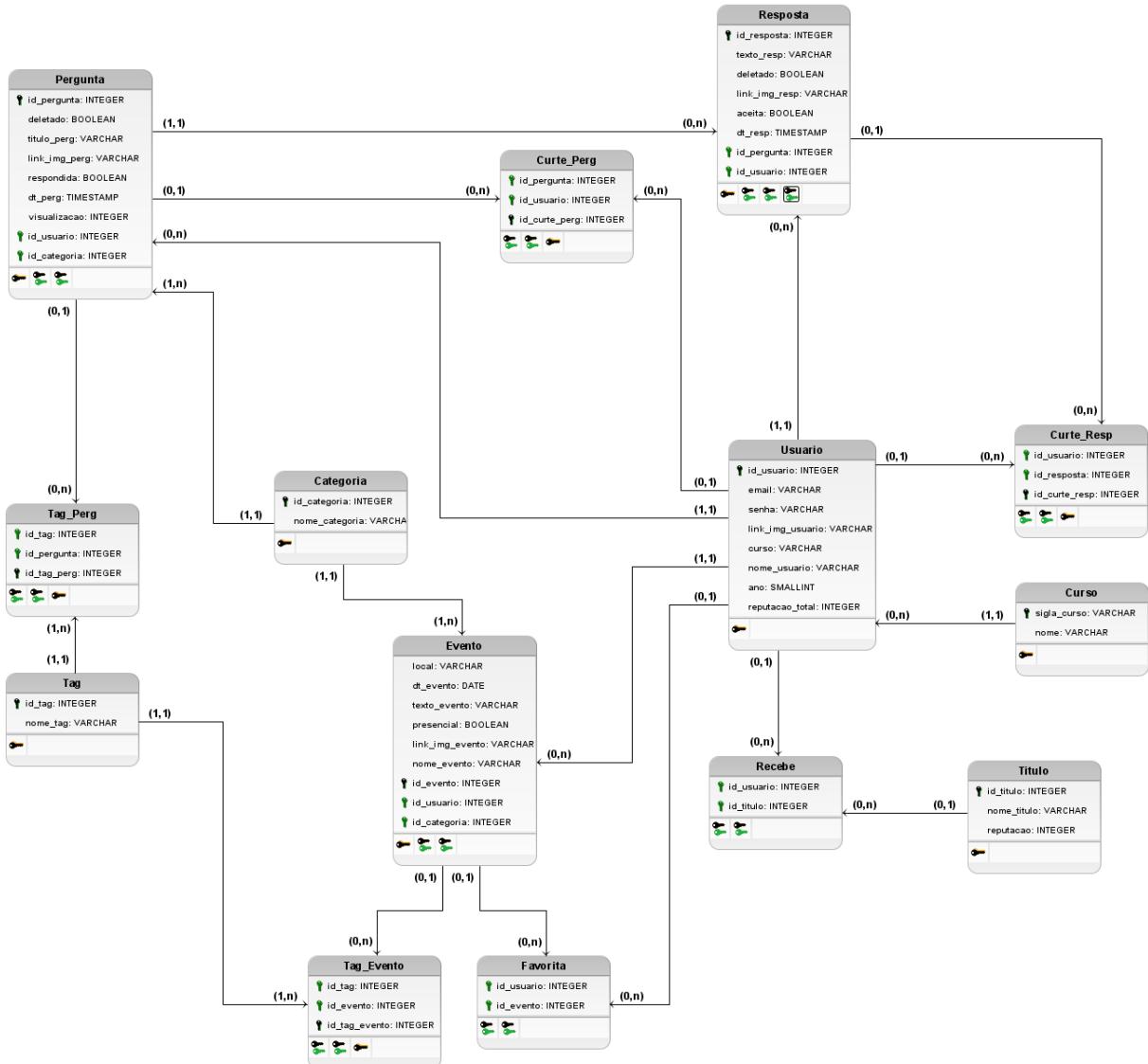
Fonte: Os autores.

2.5.4 Diagrama de Tabelas Relacionais

O Diagrama de Tabelas Relacionais DTR representa o modelo lógico do Banco de Dados. Segundo Ribeiro (201?), através do modelo lógico é representado de maneira mais clara as entidades e os relacionamentos, pois considera algumas limitações e implementa recursos como adequação de padrão e nomenclatura, define as chaves primárias e estrangeiras, normalização, integridade referencial, entre outras.

Deste modo, a Figura 12 representa o Diagrama de Tabelas Relacionais (DTR) do projeto de sistema IFriends.

Figura 12 – Diagrama de Tabelas Relacionais



Fonte: Os autores.

2.5.5 Dicionário de Dados

O **Dicionário de Dados (DD)** é responsável por armazenar as informações de configuração do banco de dados e as estruturas que compõem suas respectivas tabelas. As estruturas definem os campos e suas propriedades (ALVES, 2020).

Conforme Date (2004), o **DD** é o lugar em que, dentre outras coisas, todos os diversos esquemas (externo, conceitual, interno) e todos os mapeamentos correspondentes são mantidos.

O **DD** contém os metadados, dados que explicam dados, com relação aos diversos objetos que são de interesse do próprio sistema. Exemplos desses objetos incluem índices, usuários, restrições de integridade, restrições de segurança, e assim por diante, informações que essenciais para que o sistema faça seu trabalho apropriadamente (DATE, 2004).

De tal modo, no [Apêndice D](#) encontram-se os quadros que representam o [DD](#) das tabelas de banco de dados do projeto [IFriends](#).

2.6 Design do projeto

Cogitando a missão do projeto, surgiu a ideia de dar o nome [IFriends](#) para o sistema, cuja origem é a junção de duas palavras: IF e *friends*. Isto devido a elas retratarem bem o âmbito que será atingido, já que tais palavras em conjunto transmitem o significado de “amigos do [IFSP](#)”, nome ideal para um projeto que visa tornar a interação dos alunos mais favorável.

A próxima etapa do desenvolvimento inicial da marca foi a elaboração de uma logo, assim como a definição das cores iniciais do sistema. A logo foi desenvolvida por meio do [Canva](#), pois a plataforma se encontrava nos intermédios necessários para a elaboração da mesma.

Figura 13 – Logo do projeto

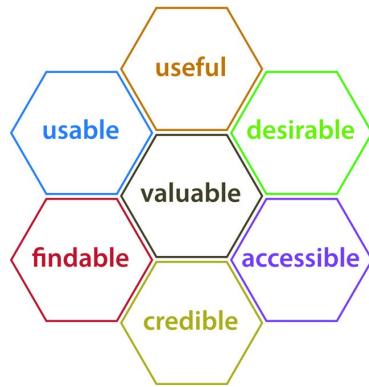


Fonte: Os autores.

Já a seleção das cores iniciais do sistema traçou um caminho através de um estudo a respeito da psicologia das cores, visto que a equipe se preocupou em passar uma boa experiência até mesmo no quesito visual. Dessa forma, se definiu o azul e suas variações como a cor principal do sistema, já que segundo [Tornos \(2021\)](#), os tons de azul se associam a princípios como: proteção, tranquilidade, fidelidade, compromisso, verdade, estabilidade, criatividade, entre outros. Vale ressaltar que o sistema ainda contará com outras cores, como roxo e algumas de suas variações, cores de sistema: variações de verde e vermelho, e cores neutras: variações de preto, cinza e branco.

Ainda, outro ponto considerado na criação da proposta foi a experiência do usuário final, pois mostrado assim como na [seção 1.2](#), a equipe se preocupou em estudar e conhecer melhor as dores deles. Visto que, segundo [Reyes \(2020\)](#), para tornar essa experiência agradável o sistema deve recorrer aos requisitos do modelo de colmeia desenvolvido por Peter Morville, sendo eles: útil, utilizável, desejável, acessível, confiável, localizável e valioso.

Figura 14 – Modelo Colmeia



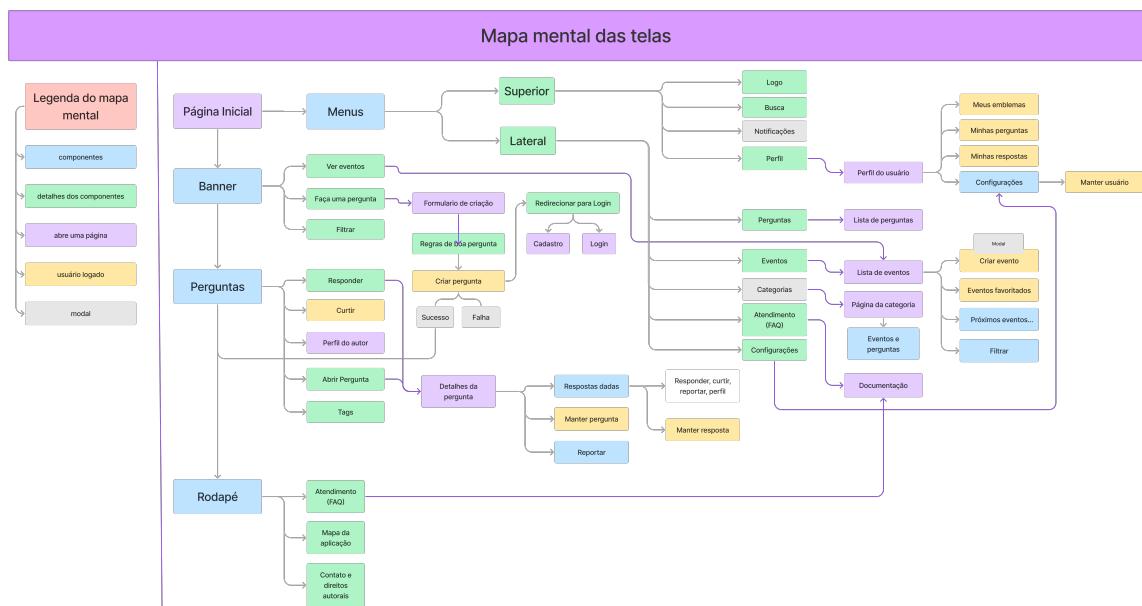
Fonte: liferay.com

2.6.1 Prototipagem

Para a prototipação se considerou os apontamentos de Ferreira (2020) qual considera que “prototipar é trazer, para o mundo real, o mundo palpável, as ideias de negócio construídas no mundo abstrato, na teoria”. Isto é, o autor comenta que um protótipo é um recurso utilizado para demonstrar e escolher a solução para representar uma ideia, podendo ser efetuado com entregas digitais, como telas de sistema. Dado isto, a próxima seção apresentará as telas prototipadas do projeto de sistema IFriends.

Ainda, para auxiliar na prototipação das telas, foi elaborado um mapa mental de modo a representar melhor o fluxo do nosso projeto, que pode ser conferido na Figura 15.

Figura 15 – Mapa mental



Fonte: os autores

2.6.1.1 Protótipos de alta fidelidade

As figuras referentes a prototipagem de alta fidelidade podem ser encontradas no [Apêndice E](#), onde cada tela apresenta uma breve contextualização sobre o seu conteúdo. De todo modo, a apresentação pode ser visualizada também pelo [Figma](#).

2.7 Swagger na API

O Swagger, de acordo com sua documentação, é um [Framework](#) de código aberto que possui ferramentas que auxiliam na documentação dos serviços de uma [REST API](#) com especificação [OpenAPI](#), o sistema deve apenas conter protocolos [HTTP](#) para aplicá-lo, independente da linguagem que está sendo utilizada. O Swagger realiza a descrição de todos os recursos da [API](#), mostrando quais métodos [HTTP](#), entidades, operações possíveis e parâmetros a serem enviados de cada [endpoints](#) da [REST API](#), por uma interface que a tecnologia disponibiliza, o Swagger UI. Portanto, este [Framework](#) facilita o entendimento do usuário em relação aos serviços de uma [API](#), proporcionando produtividade ao desenvolvedor.

Para a [API](#) do projeto, está sendo utilizado a versão 3 do Swagger, pois esta versão é a mais atual e exige menos linhas de código para implementá-la, além de nos dar mais recursos, principalmente em relação à autenticidade do sistema, segundo sua documentação. O Swagger foi implementado em um arquivo dependências gerenciado pelo [Maven](#), após isso foi ajustada através de uma classe de configuração do [SpringBoot](#) para geração do documento [OpenAPI](#) do sistema. A [Figura 16](#) mostra a interface do Swagger UI da [API](#) que está sendo desenvolvida.

Figura 16 – Swagger UI da API

The screenshot shows the Swagger UI for the iFriends API. At the top, there's a header with the Swagger logo, the URL '/v3/api-docs', and a green 'Explore' button. Below the header, the title 'iFriends' is displayed along with its version '1.1.2' and the 'OAS3' logo. A sub-header indicates it's a 'Comunidade online' by 'Bunka Bytes'. On the left, there's a sidebar titled 'Servers' showing the URL 'https://ifriends-api.herokuapp.com - Generated server url'. On the right, there's a 'Authorize' button with a lock icon.

The main content area is organized into sections for different resources:

- resposta-resource** (blue header):
 - GET /api/respostas/{id}
 - PUT /api/respostas/{id}
 - DELETE /api/respostas/{id}
 - POST /api/respostas
 - POST /api/respostas/{id}/curtir
 - GET /api/perguntas/{id}/respostas
- pergunta-resource** (blue header):
 - GET /api/perguntas/{id}
 - PUT /api/perguntas/{id}
 - DELETE /api/perguntas/{id}
 - GET /api/perguntas

Fonte: os autores

Como foi possível constatar na [Figura 16](#), no canto superior da interface indica informações gerais do projeto como o nome da [API](#), versão, descrição e nome da equipe. Já a parte inferior é mostrado o nome da classe de controle e a URI de cada requisição com seus métodos [HTTP](#). O botão “*Authorize*” é a principal função, pois este nos permite colocar o [JWT](#) para verificar o usuário que está realizando a requisição. Após abrir um terminal, é apresentado os parâmetros, o corpo da requisição e a descrição da resposta com o código de estado, como é possível observar na [Figura 17](#).

Figura 17 – Exemplo Swagger na API

The screenshot shows the Swagger UI interface for a POST request to the endpoint `/api/perguntas`. The top navigation bar indicates the method as `POST` and the URL as `/api/perguntas`. Below this, there are sections for **Parameters**, **Request body**, and **Responses**.

- Parameters:** A single parameter named `Authorization` is defined, which is a `string` type header.
- Request body:** The schema for the request body is shown as a JSON object with fields: `"titulo": "string"`, `"texto": "string"`, `"imagem": "string"`, `"tags": ["string"]`, and `"categoria": 0`.
- Responses:** The `200` status code response is listed with the description `OK`. The media type is set to `*/*`, which controls the `Accept` header.

Fonte: os autores

2.8 Análise de segurança

Para a análise de segurança das aplicações, cada qual publicada no Heroku através de suas URLs próprias (`ifriends-api` e `app-ifriends`), foi feita uma análise pelo [SSL Labs](#), visto que o [Heroku](#) utiliza um certificado básico com suporte para [TLS 1.2](#) e [HSTS](#), já que o suporte completo com o [ACME](#) na comunicação via [Let's Encrypt](#) e adição de certificados manuais, o [Heroku](#) só os disponibiliza [em versões pagas](#). De todo modo, é possível observar que ambas as aplicações receberam a nota mínima estipulada, conforme demonstrado na Figura 18 e na Figura 19.

Figura 18 – Análise de segurança na API

SSL Report: ifriends-api.herokuapp.comAssessed on: Sun, 19 Jun 2022 01:03:51 UTC | [Clear cache](#)[Scan Another >>](#)

	Server	Test time	Grade
1	54.243.238.66 ec2-54-243-238-66.compute-1.amazonaws.com Ready	Sun, 19 Jun 2022 00:57:52 UTC Duration: 89.684 sec	A+
2	107.22.57.98 ec2-107-22-57-98.compute-1.amazonaws.com Ready	Sun, 19 Jun 2022 00:59:21 UTC Duration: 90.520 sec	A+
3	3.209.172.72 ec2-3-209-172-72.compute-1.amazonaws.com Ready	Sun, 19 Jun 2022 01:00:52 UTC Duration: 89.437 sec	A+
4	23.22.130.173 ec2-23-22-130-173.compute-1.amazonaws.com Ready	Sun, 19 Jun 2022 01:02:21 UTC Duration: 89.923 sec	A+

Fonte: os autores

A análise completa do que está descrito na [Figura 18](#) também se encontra disponível para visualização [aqui](#).

Figura 19 – Análise de segurança no Front-end

SSL Report: app-ifriends.herokuapp.comAssessed on: Sun, 19 Jun 2022 00:57:27 UTC | [Clear cache](#)[Scan Another >>](#)

	Server	Test time	Grade
1	54.243.238.66 ec2-54-243-238-66.compute-1.amazonaws.com Ready	Sun, 19 Jun 2022 00:51:28 UTC Duration: 89.913 sec	A
2	107.22.57.98 ec2-107-22-57-98.compute-1.amazonaws.com Ready	Sun, 19 Jun 2022 00:52:58 UTC Duration: 89.373 sec	A
3	3.209.172.72 ec2-3-209-172-72.compute-1.amazonaws.com Ready	Sun, 19 Jun 2022 00:54:27 UTC Duration: 89.630 sec	A
4	23.22.130.173 ec2-23-22-130-173.compute-1.amazonaws.com Ready	Sun, 19 Jun 2022 00:55:57 UTC Duration: 89.717 sec	A

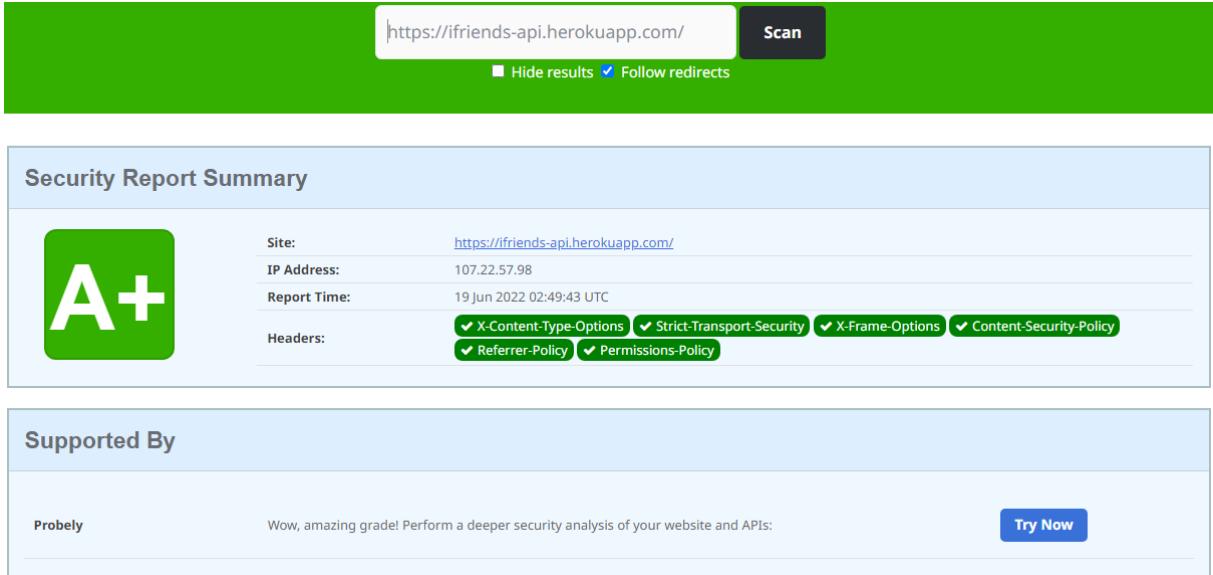
Fonte: os autores

A análise completa do que está descrito na [Figura 19](#) também se encontra disponível para visualização [aqui](#).

Além disso, conforme os requisitos da disciplina, a **API** teve suas respostas HTTP

analisadas pelo *Security Headers*, e conforme observado na [Figura 20](#), a aplicação conseguiu configurar todos os cabeçalhos básicos de segurança requeridos na análise.

Figura 20 – Análise de respostas HTTP na API



The screenshot shows a security analysis interface. At the top, there's a search bar with the URL <https://ifriends-api.herokuapp.com/> and a 'Scan' button. Below the search bar are two checkboxes: 'Hide results' (unchecked) and 'Follow redirects' (checked). The main section is titled 'Security Report Summary'. It features a large green 'A+' grade icon. To the right of the icon, the site details are listed: Site: <https://ifriends-api.herokuapp.com/>, IP Address: 107.22.57.98, Report Time: 19 Jun 2022 02:49:43 UTC. Under 'Headers:', several green checkmarks indicate the presence of security headers: X-Content-Type-Options, Strict-Transport-Security, X-Frame-Options, Content-Security-Policy, Referrer-Policy, and Permissions-Policy. Below this summary is a 'Supported By' section, which includes the Probely logo and a 'Try Now' button.

Fonte: os autores

Ainda assim, na procura por alternativas ao Heroku, a equipe pôde verificar que alguns servidores oferecem suporte completo ao [TLS](#) de forma gratuita, como o caso do [Vercel](#) para o lado do cliente. Já para o servidor, a única alternativa encontrada é criar os certificados manualmente e renová-los a cada três meses (para o cliente, esta também é uma possibilidade), porém a praticidade de hospedagem disponibilizada pelo [Heroku](#) nesse caso, com [SGBD](#) já nativo no servidor, é um dos desafios que dificultariam a migração para um novo serviço de hospedagem. De todo modo, cabe uma nova investigação dos meios para atender aos requisitos da disciplina.

2.9 Plano de testes

Segundo [Polo \(2020\)](#), por definição, os testes de *software* visam assegurar que um sistema/programa atenda às necessidades dos seus usuários, assim como também permitem descobrir defeitos no funcionamento antes de disponibilizá-lo para uso. Na realização dos testes, segundo [Sommerville \(2019\)](#), são usados dados artificiais para a sua execução.

Os resultados dos testes são verificados para descoberta de erros, anomalias ou informações não funcionais sobre a sua execução, por exemplo, análise de desempenho, utilização de memória, etc. ([POLO, 2020](#))

Considerando os apontamentos dos autores, e de modo a garantir um bom funcionamento do sistema [IFriends](#) além de prevenir eventuais erros, foi planejado a realização

de testes considerados necessários. Dessa forma, esta seção visa apresentar os testes aplicados no sistema [IFriends](#).

2.9.1 Testes unitários

Segundo [Sommerville \(2011\)](#), por testes unitários se comprehende o processo de testar componentes de programa, sendo eles métodos ou classes de objeto. Funções individuais ou métodos são os tipos mais simples de unidades. Seus testes devem ser chamados de para essas rotinas sob diferentes parâmetros. O autor ainda complementa dizendo:

Quando você está testando as classes de objeto, deve projetar os testes para fornecer uma cobertura de todas as características do objeto. Isso significa que você deve: testar todas as operações associadas ao objeto, definir e verificar o valor de todos os atributos associados ao objeto e colocar o objeto em todos os estados possíveis, o que significa simular todos os eventos que causam mudanças de estado ([SOMMERVILLE, 2011](#)).

Dessa forma, para o sistema [IFriends](#), cada novo método será feito pensando na criação dos testes unitários, de modo a também facilitar, posteriormente, seu planejamento. Os testes serão construídos efetivamente sempre após a criação de seus respectivos métodos ou classes, ou seja, após criar um método de cadastrar pergunta será feito o teste para verificar se todas as funcionalidades estão implementadas corretamente.

Além disso, considerando os apontamentos realizadas por [Bechtold \(2021\)](#) foi escolhido o [JUnit 5](#) como facilitador para execução dos testes automatizados na linguagem Java. Segundo a documentação do [JUnit](#), o [Framework](#) possui vários métodos para facilitar a verificação dos resultados, eles fazem parte da classe *Assertions*, e funcionam através de anotações “`@Test`” para dizer ao sistema que o que está abaixo da anotação é um método ou uma classe de teste. O escopo sempre possui três divisões, sendo elas: cenário, que visa a preparação dos dados a serem testados; ação, que se refere ao que o teste deve executar, e a verificação, para conferir se o teste foi realizado conforme planejado.

A [Figura 21](#) é um exemplo da estrutura de um teste, sendo este criado para o testar o método de salvar um usuário no banco de dados:

Figura 21 – Teste na API

```

25  @SpringBootTest
26  @ExtendWith(SpringExtension.class)
27  @Profile("teste1")
28  public class UsuarioServiceTest {
29
30    @SpyBean
31    UsuarioServiceImpl service;
32    PasswordEncoder encoder;
33
34    @MockBean
35    UsuarioRepository repository;
36
37
38    @Test
39    public void deveSalvarUmUsuario() {
40        Assertions.assertDoesNotThrow(() -> {
41            // cenário
42            Mockito.doNothing().when(service).validarEmail(Mockito.anyString());
43            Usuario usuario = criarUsuario();
44            usuario.setId(11);
45
46            Mockito.when(repository.save(Mockito.any(Usuario.class))).thenReturn(usuario);
47
48            // ação
49            Usuario usuarioSalvo = service.salvarUsuario(usuario);
50
51            // verificação
52            Assertions.assertNotNull(usuarioSalvo);
53            Assertions.assertEquals(usuarioSalvo.getId(), 11);
54            Assertions.assertEquals(usuarioSalvo.getNome(), "usuario");
55            Assertions.assertEquals(usuarioSalvo.getEmail(), "teste@aluno.ifsp.edu.br");
56            Assertions.assertNotEquals(usuarioSalvo.getSenha(), "123");
57            Assertions.assertEquals(usuarioSalvo.getCurso().getSigla(), "INFO");
58            Assertions.assertEquals(usuarioSalvo.getAno(), 4);
59        });
60    }

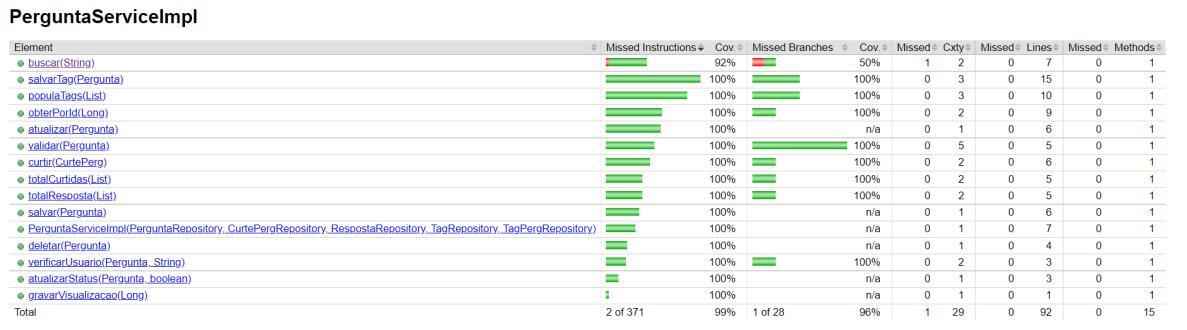
```

Fonte: os autores

Conforme mostrado na Figura 21, é possível visualizar as anotações “@SpyBean” e “@MockBean”, eles são responsáveis por dizer ao [SpringBoot](#) quais classes ele deve gerenciar e trazer para dentro do seu próprio contexto. Com a anotação “@Profile” é possível definir qual arquivo de propriedades será utilizado, no caso da Figura 21 é o arquivo que contém as configurações para o banco de dados teste.

Já para a cobertura de testes unitários está sendo utilizado o [JaCoCo](#). De acordo com sua documentação, a tecnologia oferece recursos integráveis com o [JUnit](#), e o [Eclipse](#), IDE utilizada para construção da API. A Figura 22 mostra o gráfico de cobertura de testes, realizado para a classe serviço de pergunta do [IFriends](#):

Figura 22 – Cobertura de testes



Fonte: os autores

Após a observação feita do gráfico na [Figura 22](#), a segunda coluna é possível saber a porcentagem total do percurso testado por cada método; a terceira coluna mostra todas as operações testadas, e o restante das colunas mostram a quantidade de operações, linhas e métodos da classe.

2.9.2 Testes de usabilidade

Na parte de usabilidade, a equipe definiu que, primeiramente, será validada a interface de usuário, verificando se o sistema segue uma coerência baseada nas 10 heurísticas de Nielsen, ou seja, será um teste feito inicialmente pelos próprios desenvolvedores e conforme a frequência de entregas. Ainda para a interface, conforme os requisitos da disciplina, o sistema deve ser inserido em um validador de interfaces ainda não definido.

Posteriormente, será utilizada uma técnica de caixa preta para a preparação de teste de usabilidade, com a criação de cenários de teste, contando com a atuação de pelo menos 10 estudantes, que objetivem medir o desempenho, a precisão, a lembrança e a resposta emocional entre a interação do usuário com o sistema. As descrições de tais cenários estão em fase de planejamento, porém, a equipe também deseja realizar uma pesquisa de satisfação geral ao final dos testes, para verificar a satisfação geral do cliente com relação ao sistema desenvolvido.

3 Histórico de Desenvolvimento

Assim como mencionado na [seção 2.2](#), a equipe optou pelo uso da metodologia ágil Scrum, dessa forma o processo de desenvolvimento foi organizado por meio de [Sprints](#), estas foram planejadas a partir das reuniões organizadas entre os membros da equipe. Logo, a seguinte seção visa descrevê-las, assim como apresentar a sua duração.

- **Sprint I: 17/04 a 01/05**

A primeira [Sprint](#) do projeto em suma visou a inicialização do épico de Gestão de Perguntas, visto que após a realização de estimativas com o *Scrum Poker* das histórias de usuário, a equipe concordou que este épico era o indicado para dar início ao desenvolvimento do projeto. Dessa forma, foi planejada ainda para a mesma [Sprint](#) a inicialização da prototipagem das telas, assim como a uma modelagem de dados inicial.

- **Sprint II: 02/05 a 09/05**

A segunda [Sprint](#) do projeto visou a finalização de dois épicos, isto é, a Gestão de Respostas e a Gestão de Perguntas. Além disso para esta [Sprint](#) estava planejada a apresentação da [*Proof of Concept \(POC\)*](#), logo, também era esperada uma apresentação bem estruturada, assim como o documento de visão/relatório da [POC](#).

Porém, esta [Sprint](#) não foi concluída com todas entregas de valor completas, visto que os épicos não foram finalizados com todos os critérios de aceitação prontos, porém foi possível entregar todos os que eram relacionados a construção da acsapi. Logo, passamos as tarefas faltantes para a próxima Sprint.

- **Sprint III: 16/05 a 29/05**

Após a apresentação da [POC](#) do projeto, para a nossa terceira [Sprint](#) planejou-se realizar os ajustes que ficaram pendentes, sendo na documentação, no *deploy* e na segurança da aplicação.

- **Sprint IV: 30/05 a 12/06**

Após concluir os ajustes pendentes, para a quarta [Sprint](#) se planejou estabelecer de vez uma boa conexão com a API no [*Front-end*](#) para que todas as requisições do épico de Gestão de Perguntas e o épico de Gestão de Respostas tenham um bom resultado, utilizando a autenticação do usuário. Na parte da documentação se planejou fazer a organização do documento, assim como atualizar as modificações realizadas no desenvolvimento até o momento.

- **Sprint V: 12/06 a 20/06**

Devido aos contratemplos encontrados na disciplina de **PDS** foi possível criar uma **Sprint** extra antes da 1º apresentação parcial, logo, a quinta **Sprint** visou a finalização e melhoria das tarefas pendentes da **Sprint** anterior assim como a preparação da apresentação do projeto.

3.1 Descartes

A seguinte seção foi criada visando fornecer informações a respeitos dos descartes realizados a partir do escopo inicial proposto para o projeto.

O primeiro descarte a ser feito foi em relação ao uso do **Scoold** no desenvolvimento, inicialmente a equipe pretendia fazer uso dessa plataforma visto que as funcionalidades que ela trazia se assemelhavam a boa parte do que se pretendia desenvolver no **IFriends**.

Mas, após fazer os estudos iniciais a respeito de como fazer essa implementação e de como fazer a integração com os outros recursos que ainda precisariam ser desenvolvidos, a equipe denotou que a tarefa se tornaria mais complexa do que o esperado, já que era necessário um estudo mais aprofundado, e como não dispomos um tempo de desenvolvimento significativo decidiu-se usar o **Scoold** apenas como uma inspiração. Vale lembrar que o descarte da plataforma já tinha sido previsto no escopo inicial do projeto, então o tempo aplicado no estudo não foi perdido e se tomou como aprendizado.

Referências

ALVES, W. *Banco de dados: Teoria e Desenvolvimento*. Saraiva Educação S.A., 2020. ISBN 9788536533759. Disponível em: <<https://books.google.com.br/books?id=qD0IEAAQBAJ>>. Citado na página 37.

AMBLER, S. W. *Modelagem ágil: práticas eficazes para a Programação Extrema e o Processo Unificado*. [S.l.]: Bookman, 2004. Citado na página 19.

BECHTOLD, S. B. S. *JUnit 5 User Guide*. 2021. [Online; accessed 19. Jun. 2022]. Disponível em: <<https://junit.org/junit5/docs/current/user-guide>>. Citado na página 45.

BLOG. *Um usuario, mais o que é isso afinal? é um rotulo atribuído por que ?* 2017. [Online; accessed 17. Jun. 2022]. Disponível em: <<https://blogdosistema.com.br/usuario-o-que-e>>. Citado na página 31.

CRUZ, F. *Scrum e Agile em Projetos: Guia Completo*. 2. ed. Rio de Janeiro: Brasport Livros e Multimídia Ltda., 2018. Citado 11 vezes nas páginas 20, 21, 23, 24, 25 e 27.

DALLAVALLE, S. I.; CAZARINI, E. W. Regras de negócio, um fator chave de sucesso no processo de desenvolvimento de sistemas de informação. *Anais do XX ENEGEP-Encontro Nacional de Engenharia de Produção*. São Paulo, 2000. Citado na página 30.

DATE, C. *Introdução a sistemas de bancos de dados*. ELSEVIER EDITORA, 2004. ISBN 9788535212730. Disponível em: <<https://books.google.com.br/books?id=xBeO9LSIK7UC>>. Citado na página 37.

FERNANDES, L. Redes sociais online e educação: contributo do facebook no contexto das comunidades virtuais de aprendentes. *Universidade Nova de Lisboa, Portugal*, 2011. Citado 3 vezes nas páginas 12 e 13.

FERREIRA, M. B. *Prototipagem e testes de usabilidade*. 1. ed. Curitiba: Contentus, 2020. Citado na página 39.

GUEDES, G. T. A. *UML 2 - Uma Abordagem Prática*. 2. ed. [S.l.]: novatec, 2011. Citado na página 32.

JÚNIOR, W. M. P. *Apostila Engenharia de Software*. Minas Gerais, 2010. Disponível em: <http://www.waltenomartins.com.br/ap_es_v1.pdf>. Acesso em: 23 jul. 2021. Citado na página 16.

LEAL, G. *Linguagem, Programação E Banco De Dados: GUIA PRÁTICO DE APRENDIZAGEM*. INTERSABERES, 2015. Disponível em: <<https://plataforma.bvirtual.com.br/Acervo/Publicacao/30495>>. Citado na página 35.

MACHADO, F. N. R. *Análise e Gestão de Requisitos de Software–Onde nascem os sistemas*. [S.l.]: Saraiva Educação SA, 2018. Citado na página 27.

- PEINADO, J.; AGUIAR, G. d. F. Compreendendo o kanban: um ensino interativo ilustrado. *Revista DaVinci. Curitiba (PR)*, v. 4, n. 1, p. 133–146, 2007. Citado na página 20.
- POLO, R. C. *Validação e teste de software*. Curitiba: Contentus, 2020. Citado na página 44.
- PONTOTEL, R. *Stakeholders: o que é, quem são, veja como identificá-los e quais os impactos para a empresa!* 2021. [Online; accessed 17. Jun. 2022]. Disponível em: <<https://www.pontotel.com.br/stakeholders>>. Citado na página 30.
- REYES, C. *Experiência do Usuário (UX): Entenda o que é de forma prática*. 2020. [Online; accessed 10. Apr. 2022]. Disponível em: <<https://www.liferay.com/pt/resources/1/user-experience>>. Citado na página 38.
- RIBEIRO, L. *Modelagem de Dados: Modelo Conceitual, Modelo Lógico e Físico*. 201? Disponível em: <<https://www.luis.blog.br/modelagem-de-dados-modelo-conceitual-modelo-logico-e-fisico.html>>. Acesso em: 29 setembro 2021. Citado na página 36.
- ROSA, G. C. d. Identidade cultural em comunidades de usuários e desenvolvedores de software livre: o caso debian-rs. Pontifícia Universidade Católica do Rio Grande do Sul, 2008. Citado na página 10.
- SGANDERLA, M. A.; LACERDA, G.; RIBEIRO, V. G.; SILVEIRA, S. R. Aprimorando a gerência e o desenvolvimento de software com metodologias ágeis. *RCT-Revista de Ciência e Tecnologia*, v. 2, n. 2, 2016. Citado 2 vezes nas páginas 19 e 20.
- SILVA, D. V. d. S.; SANTOS, F. A. d. O.; NETO, P. S. Os benefícios do uso de kanban na gerência de projetos de manutenção de software. In: SBC. *Anais do VIII Simpósio Brasileiro de Sistemas de Informação*. [S.l.], 2012. p. 715–725. Citado 2 vezes nas páginas 20 e 21.
- SOMMERVILLE, I. *Engenharia de Software*. 9. ed. São Paulo: Pearson Education do Brasil Ltda., 2011. Citado na página 45.
- SOMMERVILLE, I. *Engenharia de Software*. 10. ed. São Paulo: Pearson Education do Brasil Ltda., 2019. Citado 8 vezes nas páginas 15, 27, 31, 32, 33, 34 e 44.
- TORNOS, C. T. Significado da cor azul na psicologia. *Br.psicologia-online*, Br.psicologia-online.com, Sep 2021. Disponível em: <<https://br.psicologia-online.com/significado-da-cor-azul-na-psicologia-917.html>>. Citado na página 38.

Glossário

Ant	biblioteca React UI antd que auxilia na criação de interfaces interativas. - Citado em 17
Back-end	Refere-se a parte que está por trás da aplicação, responsável pela manipulação de dados voltada para o funcionamento interno de um sistema - Citado em 17, 19, 21, 22, 90, 102, 103, 105
Canva	Plataforma de design gráfico que permite a criação de gráficos de mídia social, apresentações, infográficos, pôsteres e outros conteúdos visuais - Citado em 38
Discord	Aplicativo de comunicação instantânea, muito utilizado pela comunidade gamer, por sua simplicidade em possibilitar troca de mensagens, áudio, texto e vídeo. - Citado em 78, 79, 80, 81, 82, 83, 85, 88
Eclipse	É uma IDE para desenvolvimento Java, com suporte a várias outras linguagens a partir de plugins. Ele foi feito em Java e segue o modelo open source de desenvolvimento de software. - Citado em 46
endpoint	URL que permite o acesso ao serviço por meio de uma aplicação cliente. - Citado em 40
Figma	Plataforma de criação de gráficos e prototipagem de projetos, focados principalmente em aplicações web. - Citado em 78, 85
Framework	Uma abstração que une códigos comuns entre vários projetos de software provendo uma funcionalidade genérica. - Citado em 17
Framework	Um framework ou arcabouço conceitual, é um conjunto de conceitos usado para resolver um problema de um domínio específico. - Citado em 19, 20, 40, 45
Friend	Nome dado aos usuários do sistema IFriends para exemplificar por meio deles as funcionalidades da aplicação. - Citado em 8, 31, 58, 59, 60, 61, 62, 63, 69, 71, 72, 73, 75, 77, 105
Front-end	Refere-se a parte visual e gráfica da interface de um sistema, elaboro-

gamificação	Aplicação das estratégias dos jogos nas atividades do dia a dia, com o objetivo de aumentar o engajamento dos participantes. Se baseia no game thinking, que integra a gamificação com outros saberes do meio corporativo e do design. - Citado em 12
GitHub	Ferramenta de hospedagem e versionamento de código. - Citado em 106
Google Forms	Ferramenta da Google para gerenciamento de pesquisas e formulários, utilizada para coletar e registrar informações de outras pessoas. - Citado em 11 , 86 , 87
Gource	Ferramenta utilizada para visualização em forma de diagramas e vídeos o desenvolvimento de um software. - Citado em 81 , 89 , 90 , 92 , 98 , 99 , 101 , 102 , 106
Heroku	Plataforma de nuvem como serviço que suporta várias linguagens de programação. - Citado em 15 , 18 , 42 , 44 , 91 , 102
IFriends	Nome dado ao projeto de sistemas desenvolvido, cujo significado se dá num trocadilho na junção das palavras friends (amigos, em inglês) e IF (Instituto Federal). - Citado em 10 , 11 , 13 , 19 , 23 , 30 , 31 , 32 , 34 , 35 , 36 , 38 , 39 , 44 , 45 , 46 , 49 , 58 , 64 , 69 , 86 , 87 , 91 , 105 , 106
ImgBB	Site de hospedagem de imagens. - Citado em 105
JUnit	É um Framework de código aberto que facilita a criação de testes automatizados. - Citado em 45 , 46
Maven	ferramenta de compilação e gerenciamento de projetos Java. - Citado em 17 , 40
Moodle	Ambiente virtual de ensino com o objetivo de auxiliar à aprendizagem dos alunos. - Citado em 13 , 79 , 85 , 106
Node.js	Software de código aberto que possibilita a execução de códigos JavaScript fora dos navegadores Web. - Citado em 95
Notion	Aplicação que contém ferramentas úteis para o gerenciamento do projeto e da equipe, uma delas sendo o quadro de kanban. - Citado em 78 , 85

OpenAPI	Conhecida anteriormente com Swagger, é uma especificação para arquivos de interface entendível por máquina. - Citado em 40
PostgreSQL	Sistema gerenciador de banco de dados objeto relacional, desenvolvido como projeto de código aberto. - Citado em 18
Python	Linguagem de programação de alto nível, interpretada por <i>Script</i> , de tipagem dinâmica e forte. - Citado em 93 , 94 , 95
React-Bootstrap	biblioteca que oferece os componentes clássicos do Bootstrap construídos em React. - Citado em 17
REST API	Arquitetura de sistema para serviços web. - Citado em 17 , 40
Scoold	Código aberto que funciona como uma plataforma de perguntas e respostas, fórum, base de conhecimento ou suporte ao cliente. - Citado em 13 , 49 , 106
SpringBoot	É uma ferramenta que visa facilitar o processo de configuração e publicação de aplicações que utilizem o ecossistema Spring. - Citado em 40 , 46
Sprint	Tempo estimado para promover um desenvolvimento mais focalizado no projeto. - Citado em 26 , 48 , 49 , 82 , 102 , 103 , 104 , 105
StatSVN	Ferramenta que funciona a partir de arquivos de log extraídos do repositório do SVN, fornecendo gráficos e dados estatísticos a partir do cruzamento dessas informações. - Citado em 104 , 106
SVN	Sistema de controle de versão de uso obrigatório na disciplina. - Citado em 91 , 96 , 97 , 98 , 99 , 102 , 106
WhatsApp	Aplicativo de mensagens instantâneas e chamadas de voz para smartphones. - Citado em 11 , 87 , 104 , 105
Yamllint	Validador de arquivos .yaml - Citado em 92 , 93 , 96 , 97
YouTube	Plataforma destinada a compartilhamento de vídeos. - Citado em 15 , 22 , 78 , 85 , 91 , 98 , 101

Apêndices

APÊNDICE A – Perguntas da pesquisa de viabilidade

1. Qual é o seu grau de escolaridade?
2. Qual é o seu curso?
3. Você trabalha ou faz estágio?
4. Você sentiu dificuldade em se adaptar ao entrar no IF?
5. Descreva como foi a sua experiência (com relação às dificuldades na instituição e no ensino).
6. Com que frequência você costuma ir às monitorias?
7. Você usaria um sistema de perguntas e respostas do IF?
8. Você acredita que uma comunidade de perguntas e respostas te ajudaria na sua vida acadêmica?
9. De que forma isso faria/não faria diferença para você? (Fique a vontade de responder com toda sinceridade!).
10. Gostaria de compartilhar mais alguma coisa sobre o tema? Bem, sinta-se a vontade!

APÊNDICE B – Histórias de usuário

A seguinte seção apresentará as histórias de usuário do projeto **IFriends** classificadas por seus respectivos épicos.

B.1 Épico - Gestão de perguntas

Quadro 12 – História: Manter uma pergunta

Descrição	Épico	Pontuação	Tamanho	Prioridade
Como Friend , eu gostaria de manter uma pergunta na comunidade para retirar uma dúvida	Gestão de Perguntas	13	Grande	ALTA

Fonte: Os autores

Para esta história de usuário foram definidos os seguintes itens como critérios de aceitação:

- Mostrar “como fazer uma boa pergunta”;
- O usuário deve conseguir somente criar e remover uma pergunta da visualização;
- O usuário deve conseguir fechar o espaço de resposta para a pergunta;

Quadro 13 – História: Filtrar perguntas

Descrição	Épico	Pontuação	Tamanho	Prioridade
Como Friend , eu gostaria de filtrar perguntas para que possa encontrar as mais relevantes.	Gestão de Perguntas	8	Médio	MÉDIA

Fonte: Os autores

Para esta história de usuário foram definidos os seguintes itens como critérios de aceitação:

- Tipos de filtro de perguntas: mais populares, recentes, por assuntos, sem resposta;

Quadro 14 – História: Adicionar assuntos

Descrição	Épico	Pontuação	Tamanho	Prioridade
Como Friend , eu gostaria de adicionar assuntos nas minhas perguntas para encontrá-las mais facilmente	Gestão de Perguntas	3	Pequeno	ALTA

Fonte: Os autores

Para esta história de usuário foram definidos os seguintes itens como critérios de aceitação:

- Devem existir tipos de tags (exemplo: o tipo é informática, que possui várias tags);
- As tags serão divididas em uma lista pré-definida de acordo com seu tipo e um campo outras para tags que não existem.

Quadro 15 – História: Curtir uma pergunta

Descrição	Épico	Pontuação	Tamanho	Prioridade
Como Friend , eu gostaria de votar em uma pergunta para indicar se ela me foi útil ou não.	Gestão de Perguntas	2	Pequeno	ALTA

Fonte: Os autores

Para esta história de usuário foram definidos os seguintes itens como critérios de aceitação:

- Um usuário só poderá votar uma única vez;
- Cada voto equivale a um ponto;
- Soma dos pontos por pergunta deve ser exibida;

Quadro 16 – História: Buscar perguntas

Descrição	Épico	Pontuação	Tamanho	Prioridade
Como Friend , eu gostaria de buscar perguntas feitas para que possa consultar uma pergunta específica	Gestão de Perguntas	2	Pequeno	MÉDIA

Fonte: Os autores

Para esta história de usuário foram definidos os seguintes itens como critérios de aceitação:

- O usuário precisa informar total ou parcialmente o título da pergunta desejada;
- As perguntas serão exibidas conforme as informações passadas, podendo ser semelhantes parcial ou totalmente;

B.2 Épico - Gestão de respostas

Quadro 17 – História: Manter uma resposta

Descrição	Épico	Pontuação	Tamanho	Prioridade
Como Friend , eu gostaria de manter uma resposta para retirar uma dúvida de um colega.	Gestão de Respostas	5	Médio	ALTA

Fonte: Os autores

Para esta história de usuário foram definidos os seguintes itens como critérios de aceitação:

- As respostas mais curtidas devem ser exibidas antes das demais;
- O usuário deve conseguir somente criar e deletar uma resposta;
- Todas as respostas devem ser exibidas sem exceção;

Quadro 18 – História: Curtir uma resposta

Descrição	Épico	Pontuação	Tamanho	Prioridade
Como Friend , eu gostaria de curtir uma resposta para indicar se ela me foi útil ou não.	Gestão de Respostas	1	Pequeno	ALTA

Fonte: Os autores

Para esta história de usuário foram definidos os seguintes itens como critérios de aceitação:

- Um usuário só poderá curtir uma única vez;
- Cada curtida equivale a um ponto;
- Soma das curtidas por pergunta deve ser exibida;

B.3 Épico - Gestão de eventos

Quadro 19 – História: Manter evento

Descrição	Épico	Pontuação	Tamanho	Prioridade
Como Friend , eu gostaria de manter eventos para ser ajudado e ajudar meus colegas com suas dúvidas sobre um assunto.	Gestão de Eventos	3	Pequeno	BAIXA

Fonte: Os autores

Para esta história de usuário foram definidos os seguintes itens como critérios de aceitação:

- É necessário uma aba com a listagem de todos os eventos publicados separados em ordem cronológica;

Quadro 20 – História: Favoritar evento

Descrição	Épico	Pontuação	Tamanho	Prioridade
Como Friend , eu gostaria de favoritar um evento para expressar meu interesse em atendê-lo.	Gestão de Eventos	2	Pequeno	BAIXA

Fonte: Os autores

Para esta história de usuário foram definidos os seguintes itens como critérios de aceitação:

- Deve ser um contador de número de inscritos no evento;

B.4 Épico - Gestão de usuários

Quadro 21 – História: Autenticação do usuário

Descrição	Épico	Pontuação	Tamanho	Prioridade
Como Friend , eu gostaria de me autenticar no sistema para realizar ações personalizadas.	Gestão de Usuários	5	Médio	ALTA

Fonte: Os autores

Para esta história de usuário foram definidos os seguintes itens como critérios de aceitação:

- Layout das telas de *login* e cadastro finalizado;
- Verificação do *Token* da *API* no *login* via *JWT*.

B.5 Épico - Usabilidade

Quadro 22 – História: Internacionalização

Descrição	Épico	Pontuação	Tamanho	Prioridade
Como Friend , eu gostaria de escolher o idioma da aplicação para visualização em inglês e português.	Usabilidade	5	Médio	ALTA

Fonte: Os autores

Para esta história de usuário foram definidos os seguintes itens como critérios de aceitação:

- Textos disponíveis em ambos os idiomas a qualquer momento na aplicação.

APÊNDICE C – Boas práticas para uma pergunta

Para auxiliar na elaboração de perguntas, foi disponibilizado aos Friends um pequeno tutorial de como pode ser feita uma pergunta.

1. Resuma o seu problema:

- Antes de fazer uma pergunta tenha em mente "qual é o problema". Para isso é recomendado que primeiramente se atente a reunir detalhes e informações que poderão ser úteis.
- Considere que uma pergunta não se caracteriza como boa devido ao seu tamanho, mas sim devido às informações fornecidas.
- Opte por um título sucinto e detalhado, e recorra a termos chaves, as tags podem ajudar nesse sentido.

2. Descreva o seu problema:

- Apresente seu problema com o máximo de detalhes, o que você já tentou e conte-nos o que você conseguiu até então.
- Lembre-se que conseguirá melhores respostas quando você fornecer e detalhar melhor os seus dados.
- Quando for apropriado, recorra a imagens para exemplificar melhor o problema.

3. Objetivo final:

- O que é preciso para chegar a um resultado viável, tente ser o mais claro possível em expressar qual é o seu objetivo.

APÊNDICE D – Dicionário de Dados

A seguinte seção apresentará o **DD** das tabelas de banco de dados do projeto **IFriends**.

Quadro 23 – Usuário.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_usuario	INT	11	N	PK	Chave primária do usuário
nome	VARCHAR	150	N		Nome do usuário
email	VARCHAR	100	N		E-mail institucional do usuário
senha	VARCHAR	255	N		Senha de acesso ao sistema
link_img_usuario	VARCHAR	255	S		link da imagem de perfil
curso	VARCHAR	20	S		Curso atual
ano	SMALLINT	1	S		Ano que o usuário cursa, ex.: 1ºano
reputacao_total	INT	11	N		Pontuação da reputação do usuário

Fonte: Os autores

Quadro 24 – Curte_Perg.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_curte_perg	INT	11	N	PK	Chave primária da curte_perg
id_usuario	INT	11	N	FK	Chave estrangeira no usuário
id_pergunta	INT	11	N	FK	Chave estrangeira na pergunta

Fonte: Os autores

Quadro 25 – Curte_Resp.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_corte_resp	INT	11	N	PK	Chave primária da curte_resp
id_usuario	INT	11	N	FK	Chave estrangeira no usuário
id_resposta	INT	11	N	FK	Chave estrangeira na resposta

Fonte: Os autores

Quadro 26 – Usuário_Título.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_usuario	INT	11	N	FK	Chave estrangeira no usuário
id_titulo	INT	11	N	FK	Chave estrangeira no título

Fonte: Os autores

Quadro 27 – Título.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_titulo	INT	11	N	PK	Chave primária do título
nome_titulo	VARCHAR	50	N		Nome do título
reputacao	INT	11	N		Acumulo da pontuação

Fonte: Os autores

Quadro 28 – Curso.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
sigla_curso	VARCHAR	20	N	PK	Sigla e chave primária do curso
nome_curso	VARCHAR	50	N		Nome do curso

Fonte: Os autores

Quadro 29 – Resposta.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_resposta	INT	11	N	PK	Chave primária da resposta
id_usuario	INT	11	N	FK	Chave estrangeira de usuário
id_pergunta	INT	11	N	FK	Chave estrangeira de pergunta
texto_resp	VARCHAR	800	N		Conteúdo da resposta
deletado	BOOLEAN		N		Resposta foi deletada ou não
link_img_resp	VARCHAR	255	S		Imagem da resposta
aceita	BOOLEAN		N		Se a resposta foi aceita como solução válida para o autor da pergunta
dt_resp	TIMESTAMP		N		Data em que a resposta foi publicada

Fonte: Os autores

Quadro 30 – Tag.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_tag	INT	11	N	PK	Chave primária da tag
nome_tag	VARCHAR	20	N		Nome da tag

Fonte: Os autores

Quadro 31 – Pergunta.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_pergunta	INT	11	N	PK	Chave primária da pergunta
id_usuario	INT	11	N	FK	Chave estrangeira de usuário
dt_per	TIMESTAMP		S		Data da pergunta
titulo_per	VARCHAR	50	N		Título da pergunta
texto_per	VARCHAR	800	N		Descrição da pergunta
deletado	BOOLEAN		N		Pergunta foi deletada ou não
link_img_per	VARCHAR	255	S		Link da imagem da pergunta
respondida	BOOLEAN		N		Se a pergunta já teve uma resposta útil a quem perguntou
visualizações	INT	11	S		Quantidade de visualizações da pergunta

Fonte: Os autores

Quadro 32 – Tag_Perg.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_tag_per	INT	11	N	PK	Chave primária da tag_per
id_assunto	INT	11	N	FK	Chave estrangeira no assunto
id_pergunta	INT	11	N	FK	Chave estrangeira na pergunta

Fonte: Os autores

Quadro 33 – Tag_Evento.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_tag_evento	INT	11	N	PK	Chave primária do tag_evento
id_assunto	INT	11	N	FK	Chave estrangeira no assunto
id_evento	INT	11	N	FK	Chave estrangeira no evento

Fonte: Os autores

Quadro 34 – Evento.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_evento	INT	11	N	PK	Chave primária do evento
id_usuario	INT	11	S	FK	Chave estrangeira de usuário
presencial	char	1	N		Local do evento, sendo presencial, online ou ambos
nome_evento	VARCHAR	50	N		Nome do evento
texto_evento	TEXT		N		Descrição sobre o evento
dt_evento	DATE		N		Data que o evento ocorrerá
img_evento	TEXT		S		Imagen do evento
local	TEXT		N		Local onde será realizado, tanto presencial como online

Fonte: Os autores

Quadro 35 – Categoria.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_categoria	INT	11	N	PK	Chave primária da categoria
nome_categoria	INT	50	N	FK	Nome da categoria

Fonte: Os autores

APÊNDICE E – Prototipação das telas

A seguinte seção apresentará as figuras referentes a prototipagem de alta fidelidade do projeto **IFriends**. Para a prototipação levou-se em consideração, dois estados, sendo eles o do usuário com *login* e o do usuário sem *login*.

E.1 Friend sem *login*

Quando um usuário externo ou até mesmo um **Friend** que ainda não tenha cadastro na aplicação decidirem visualizar ou conhecer a comunidade eles podem, mas os mesmos terão algumas funcionalidades bloqueadas, podendo apenas visualizar o sistema.

Dessa forma, a [Figura 23](#) apresenta a primeira tela exibida para o **Friend** ou usuário externo quando ele faz o primeiro contato com o sistema. Nela o **Friend** pode navegar através de dois menus disponíveis na página: o lateral e o superior, usar a barra de pesquisa e *logar* no seu perfil.

Figura 23 – Página inicial sem login

The screenshot shows the homepage of a platform called iFriends. On the left, there is a vertical sidebar with icons for Home, Dashboard, Help, Notifications, and Settings. The main content area features a large central banner with two illustrations: a person thinking next to a large question mark and another person looking at a phone. The text in the banner reads: "Está com dúvidas ou quer compartilhar o que sabe? Navegue pelo iFriends!" Below the banner, it says: "Aqui você pode ser perguntar, responder e ainda participar ou dar mentorias sobre um assunto que ache legal!". There are two buttons: "Perguntar" (Ask) and "Ver eventos" (View events). A "Filtros" (Filters) button is also present. Below the banner, a section titled "Últimas perguntas:" (Last questions) displays four recent posts from users. Each post includes the user's profile picture, the question title, the user's name, the time ago, like count, comment count, and a "Responder" (Answer) button. At the bottom of the page, there are navigation buttons for "Anterior" (Previous) and "Próximo" (Next), and a page number indicator showing pages 1 through 9.



Atendimento

- [FAQ](#)
- [Entrar em contato](#)
- [Políticas da comunidade](#)
- [Termos de uso e condições](#)

Mapa do site

- [Página inicial](#)
- [Eventos](#)
- [Categorias](#)
- [Configurações](#)
- [Contato](#)

A Figura 24 representa a página de perguntas e suas respostas.

Figura 24 – Página da pergunta sem *login*

Pesquisar por perguntas e interesses **Entrar** **Cadastre-se**

boletim escolar suap

Sabrina da Silva 2 dias atrás

Como visualizar o boletim escolar?

Entre essa semana no IF e gostaria de saber como posso visualizar meu boletim escolar, me disseram que tenho que acessar por um tal de suap, podem me dar uma dica?

Curtir **Reportar**

2 respostas

Melhor resposta

Juninha da silva 45 minutos atrás

Olá, tudo bem?
Sim, é possível ver as notas por meio do suap. Para isso você precisa fazer login com seu prontuário e senha no seguinte link: "https://suap.ifsp.edu.br/accounts/login/". Após efetuar o login você já se depara com o menu principal do sistema, nele você tem várias opções, mas a que nos interessa é "Ensino", nele clique em "Boletins" é pronto ta pronto o sorvetinho!
Espero ter ajudado =)

Responder

Maria Souza 1 dia atrás

Você pode ver suas notas através do suap, basta você fazer login e entrar na aba correspondente!

Responder

Sua resposta

Conhece uma resposta para essa dúvida? Mande a sua resposta!

Sua resposta **Enviar**

Registre-se

Não arrisque perder o seu progresso, registre-se agora é rápido e fácil!

Registre-se usando email e senha



Atendimento

- FAQ
- Entrar em contato
- Políticas da comunidade
- Termos de uso e condições

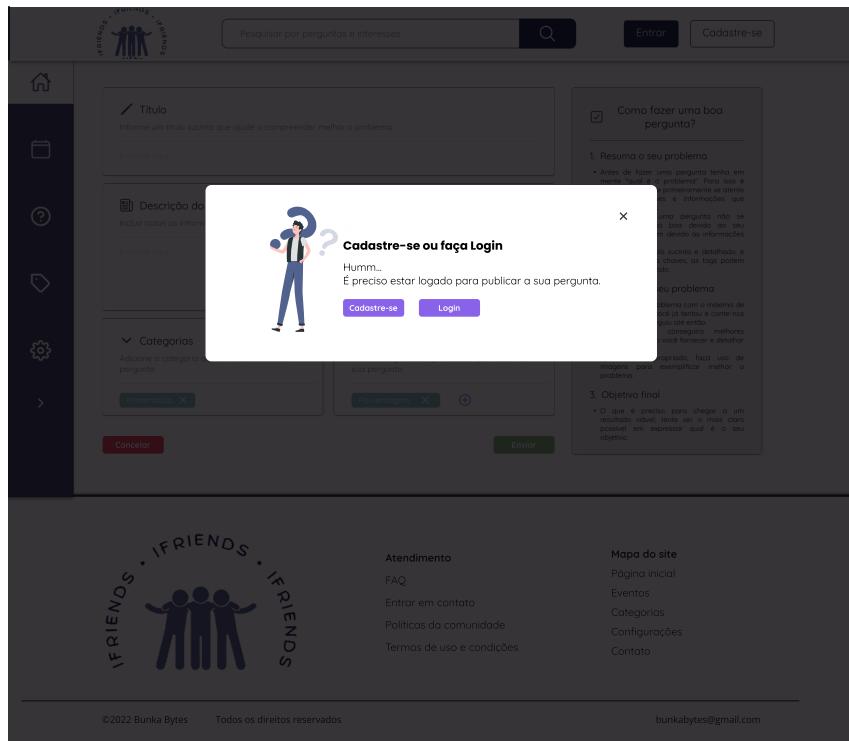
Mapa do site

- Página inicial
- Eventos
- Categorias
- Configurações
- Contato

Fonte: os autores

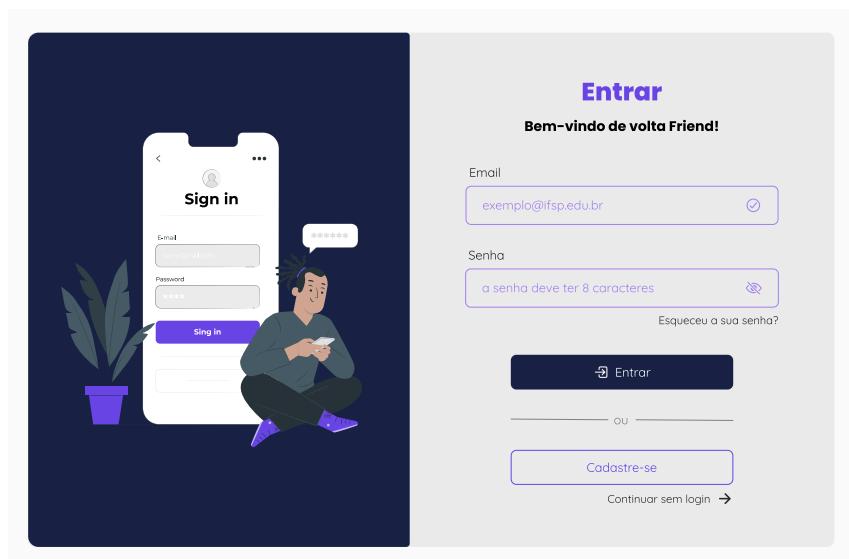
A Figura 25 corresponde a página de cadastro de perguntas, perceba que quando o **Friend** não está *logado* no sistema o mesmo bloqueia o processo pedindo primeiro a autenticação do mesmo.

Figura 25 – Página de cadastro de perguntas sem login



Fonte: os autores

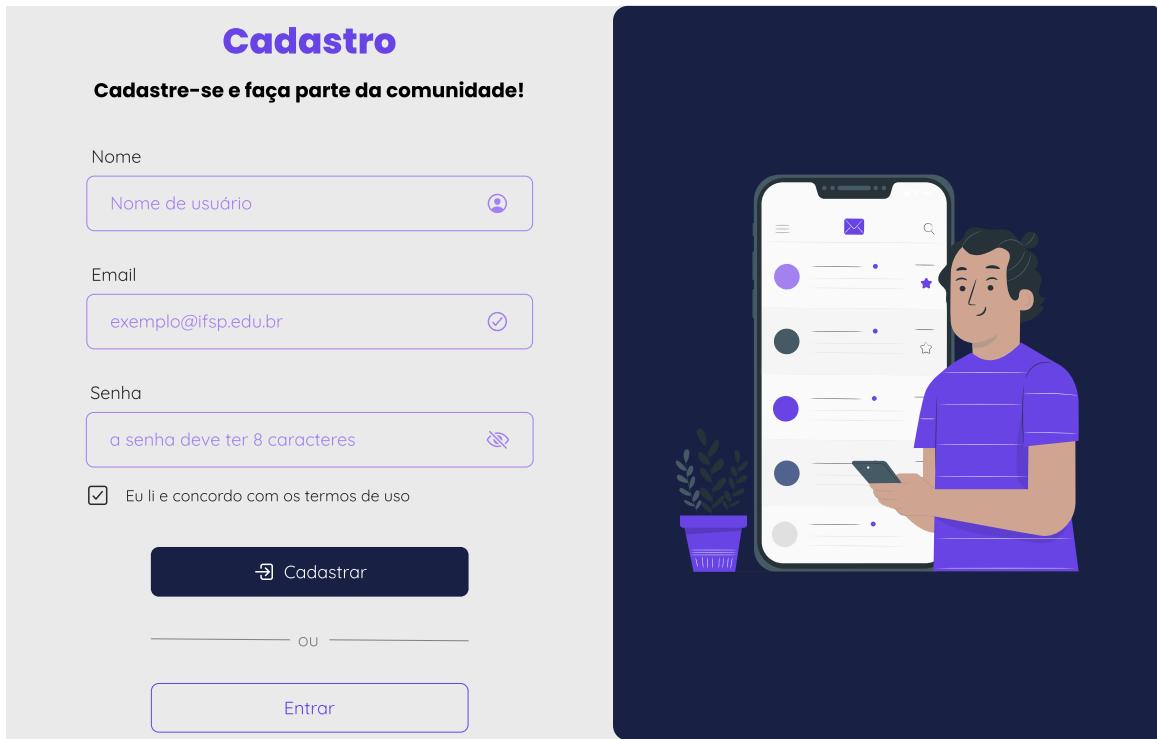
Caso o **Friend** escolha *logar* no sistema será exibida a tela de *login* (Figura 26). Nela, para fazer a autenticação é solicitada o e-mail e a senha cadastrados, e caso o **Friend** esqueça a sua senha ele consegue recurrir por meio do link “Esqueceu a sua senha?”, e se desejar continuar navegando sem fazer *login* ele também pode basta clicar no link “Continuar sem login”.

Figura 26 – Página de *login*

Fonte: os autores

Já, caso ele opte por realizar o cadastro será exibida a [Figura 27](#), que corresponde a página de cadastro, nela serão necessárias algumas informações como o nome,

Figura 27 – Página de cadastro



Fonte: os autores

E.2 Friend com login

Já para o [Friend logado](#) no sistema, não há restrições de funcionalidades, então desde que as use com responsabilidade e da forma certa a aplicação cumprirá o que foi proposto.

Logo, a [Figura 28](#) apresenta a página inicial do projeto, onde o [Friend](#) entra em contato pela primeira vez com o sistema. Nela o [Friend](#) pode navegar através de dois menus disponíveis na página: o lateral e o superior, usar a barra de pesquisa, *logar* no seu perfil, acessar as suas configurações, entre outras ações disponibilizadas. Na página encontram as questões mais relevantes da comunidade, assim como os espaços destinados para a realização de uma pergunta ou de uma monitoria.

Figura 28 – Página inicial

The screenshot shows the initial page of the iFriends platform. On the left is a dark sidebar with icons for Home, Calendar, Help, Notifications, Settings, and Logout. The main content area features a search bar at the top with placeholder text "Pesquisar por perguntas e interesses". Below the search bar is a large central box with a question mark icon and the text "Está com dúvidas ou quer compartilhar o que sabe? Navegue pelo iFriends!". It includes buttons for "Perguntar" and "Ver eventos". A user profile is shown on the right with options to change language (Português, English) and log out. Below this is a section titled "Últimas perguntas:" displaying four recent questions from users Sabrina da Silva, AlzhanMaratovna, and two instances of the same user. Each question card includes tags, the asker's profile picture, the question text, the asker's name, the time posted, like and comment counts, and a "Responder" button.



Atendimento

- [FAQ](#)
- [Entrar em contato](#)
- [Políticas da comunidade](#)
- [Termos de uso e condições](#)

Mapa do site

- [Página inicial](#)
- [Eventos](#)
- [Categorias](#)
- [Configurações](#)
- [Contato](#)

Quando o **Friend** clica em uma pergunta ou em “Responder” ele é direcionado à página dessa pergunta como mostra a [Figura 29](#), nela ele pode encontrar as respostas já fornecidas por outros membros da comunidade, assim como também pode deixar a sua contribuição.

Figura 29 – Pergunta e respostas

Pesquisar por perguntas e interesses

boletim escolar suap

Sabrina da Silva 2 dias atrás

Como visualizar o boletim escolar?

Entre essa semana no IF e gostaria de saber como posso visualizar meu boletim escolar, me disseram que tenho que acessar por um tal de suap, podem me dar uma dica?

Curtir Reportar

2 respostas

Melhor resposta

Juninha da Silva 45 minutos atrás

Olá, tudo bem?
Sim, é possível ver as notas por meio do suap. Para isso você precisa fazer login com seu prontuário e senha no seguinte link: "https://suapi.ifsp.edu.br/accounts/login/". Após efetuar o login você já se depara com o menu principal do sistema, nele você tem várias opções, mas a que nos interessa é "Ensino", nele clique em "Boletins" é pronto ta pronto o sorvetinho!
Espero ter ajudado =)

15 Responder

Maria Souza 1 dia atrás

Você pode ver suas notas através do suap, basta você fazer login e entrar na aba correspondente!

5 Responder

Sua resposta

Conhece uma resposta para essa dúvida? Mande a sua resposta!

Sua resposta

Enviar



Atendimento

- FAQ
- Entrar em contato
- Políticas da comunidade
- Termos de uso e condições

Mapa do site

- Página inicial
- Eventos
- Categorias
- Configurações
- Contato

Já a [Figura 30](#) corresponde a página de cadastro de perguntas, nessa tela são apresentados todos os elementos julgados necessários para a sua realização, nesta página ainda se encontra o manual de uma boa pergunta ([Apêndice C](#)), tal foi elaborado com o intuito de ajudar e auxiliar o [Friend](#) na preparação de sua problemática.

Figura 30 – Cadastro de perguntas



Atendimento

- FAQ
- [Entrar em contato](#)
- [Políticas da comunidade](#)
- [Termos de uso e condições](#)

Mapa do site

- [Página inicial](#)
- [Eventos](#)
- [Categorias](#)
- [Configurações](#)
- [Contato](#)

APÊNDICE F – Atas das Reuniões

F.1 1ºbimestre

F.1.1 Planejamento - 15/03/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto

Local: [Discord](#).

Pauta(s): Primeira reunião realizada pela equipe com o objetivo de planejar os passos iniciais do projeto, nela a equipe estabeleceu um contrato social na plataforma [Figma](#) contendo os combinados essenciais para o convívio social entre os componentes da equipe e também iniciou-se um *board* de ideias iniciais para o projeto.

F.1.2 Planejamento/Alinhamento - 17/03/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto

Local: [Discord](#).

Pauta(s): Reunião de planejamento e alinhamento, onde o foco da equipe esteve em discutir as ideias pré-selecionadas na reunião anterior para o projeto. Desde este primeiro momento a atenção da equipe se voltava principalmente para uma comunidade de dúvidas entre os estudantes do IF.

Para organização das tarefas e da equipe a plataforma [Notion](#), onde o quadro de *kanban* se encontra, foi criado e organizado inicialmente.

A equipe decidiu inicialmente que realizará reuniões nos dias e horários das janelas entre as aulas, permitidas pela grade curricular.

F.1.3 Planejamento/Alinhamento - 18/03/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto

Local: [Discord](#).

Pauta(s): Reunião de planejamento onde a equipe discutiu as tarefas a serem realizadas e discutiu outras ferramentas de organização.

Criamos o canal do [YouTube](#).

Realizamos a primeira postagem para o *blog* da equipe.

Amadurecemos ainda mais a ideia da comunidade, pensando em mais funcionalidades para agregar na aplicação e anotando as dúvidas a respeito do tema e do funcionamento.

F.1.4 Alinhamento - 21/03/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto
Local: Saguão [IFSP](#).

Pauta(s): Reunião realizada antes da aula de [PDS](#) para a equipe discutir algumas dúvidas, perspectivas e ideias sobre o projeto, visando otimizar o tempo em sala de aula.

F.1.5 Alinhamento - 25/03/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto
Local: [Discord](#).

Pauta(s): Reunião de alinhamento, onde a equipe pesquisou em trabalhos anteriores a formatação do documento da proposta inicial para preparar as etapas. Aproveitamos também para melhorar o *layout* do *blog* da equipe.

F.1.6 Planejamento/Alinhamento - 28/03/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto
Local: Laboratório [IFSP](#).

Pauta(s): Reunião de alinhamento e planejamento, onde a equipe conversou com os orientadores sobre soluções e funcionalidades para a proposta da comunidade. Neste ponto, resolvemos um problema antigo, validar que o usuário seja de fato aluno da instituição, a solução encontrada foi enviar um *e-mail* de validação para o *e-mail* institucional do aluno. Foi citado também outra proposta de projeto para a equipe realizar, uma plataforma de controle e gestão financeira. Porém, a equipe optou por seguir na proposta da comunidade. Partimos para as tarefas, iniciando o desenvolvimento de um questionário direcionado aos alunos do instituto para estudar a viabilidade de criação do projeto.

Sobre o *blog*, finalizamos a melhoria de seu *layout* e definimos que as publicações serão realizadas aos sábados pela manhã.

F.1.7 Planejamento/Retrospectiva - 02/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto
Local: [Discord](#).

Pauta(s): Reunião de retrospectiva e planejamento das atividades, onde revisamos a publicação da semana no *blog* e as perguntas para a pesquisa de viabilidade. Realizamos a entrega sobre as tecnologias que serão utilizadas no desenvolvimento do projeto no [Moodle](#) da disciplina e definimos duas tarefas: divulgação da pesquisa de viabilidade a partir de segunda-feira para alunos e ex-alunos da instituição e gerenciamento do *backlog* para cada parte do projeto.

F.1.8 Planejamento - 04/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto
Local: Laboratório [IFSP](#).

Pauta(s): Reunião de planejamento, onde a equipe após receber as orientações para a apresentação da proposta inicial, organizou as tarefas a serem realizadas por cada componente, organizou a formatação do documento para a proposta inicial e iniciou a divulgação do formulário para pesquisa de viabilidade da proposta.

F.1.9 Alinhamento - 09/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto
Local: [Discord](#).

Pauta(s): Reunião de alinhamento, onde a equipe se reuniu para realizar as atividades destinadas a apresentação da proposta inicial.

F.1.10 Alinhamento - 10/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto
Local: [Discord](#).

Pauta(s): Reunião de alinhamento, onde a equipe se reuniu para realizar as atividades destinadas a apresentação da proposta inicial.

F.1.11 Retrospectiva - 12/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto
Local: Biblioteca [IFSP](#).

Pauta(s): Reunião retrospectiva, onde a equipe analisou como foi o processo para realizar a entrega da proposta inicial, desta forma foram levantados os pontos positivos: todos estarem reunidos em chamada para realizar as tarefas, conseguimos entregar o que era esperado e apesar das dificuldades enfrentadas a apresentação fluiu bem. Os pontos negativos: realizar muitas tarefas no final de semana ficou puxado para a equipe e não ter realizado um ensaio antes da apresentação, como melhoria, queremos marcar mais reuniões com os professores.

F.1.12 Planejamento/Alinhamento - 17/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto
Local: [Discord](#).

Pauta(s): Reunião de planejamento e alinhamento para a semana, onde a equipe planejou as próximas *sprints* da [POC](#) juntamente com os professores e aproveitou para conversar sobre as avaliações das equipes em relação a apresentação da proposta inicial.

F.1.13 Planejamento - 18/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto.

Local: Laboratório [IFSP](#).

Pauta(s): Reunião de planejamento onde a equipe criou o *backlog* do produto com o uso das histórias de usuário e definiu as tarefas da semana, preparando-se para os dois épicos para a [POC](#).

F.1.14 Alinhamento - 21/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto

Local: [Discord](#).

Pauta(s): Reunião de alinhamento, onde a equipe terminou de elaborar as histórias de usuário, definiu as prioridades para a [POC](#) e votou por meio do *Planning Poker* - descrito pela metodologia Scrum - para estimarmos os esforços necessários para a conclusão de cada história. Durante as discussões da equipe para a execução desta tarefa, muitos pontos sutis, mas que poderiam ser perigosos no futuro, foram levantados e anotados para discutirmos com os orientadores.

F.1.15 Alinhamento - 25/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto

Local: Laboratório [IFSP](#).

Pauta(s): Reunião de alinhamento onde a equipe conversou com os orientadores sobre as dúvidas levantadas na reunião anterior, durante a elaboração da tarefa de definição das histórias de usuários, apresentou os diagramas de entidade e relacionamento e o diagrama de casos de uso para serem alinhados corretamente. Iniciamos também as configurações para criação do vídeo do [Gource](#).

F.1.16 Alinhamento - 01/05/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto

Local: [Discord](#).

Pauta(s): Reunião de alinhamento onde a equipe finalizou as histórias de usuário com base na discussão realizada em aula com os orientadores, alinhou o fluxo de usuário no sistema, os requisitos funcionais, não funcionais e as regras de negócio. Iniciamos o planejamento para realização da apresentação da [POC](#) e como orientação dos professores, decidimos deixar para desenvolver o épico de Gestão de Eventos apenas se sobrar tempo.

F.1.17 Alinhamento - 03/05/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto
Local: [Discord](#).

Pauta(s): Realização de tarefas para a [POC](#), configurações de ambiente e alinhamento da documentação.

F.2 2ºbimestre

F.2.1 Retrospectiva - 16/05/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto
Local: Laboratório [IFSP](#).

Pauta(s): Reunião retrospectiva referente ao desenvolvimento e entrega da [POC](#), onde a equipe chegou as seguintes conclusões: do que foi bom, entrega da [API](#) completa; do que foi ruim, falta de tempo, pois mesmo sabendo como desenvolver um dos requisitos não sobrou tempo para fazer; e um problema enfrentado por nós e pelas outras equipes da turma foi a incerteza e a definição errada que construímos sobre a [POC](#), isso gerou uma dificuldade na definição do que era extremamente essencial para a entrega. Como plano de ação, a equipe buscará definir melhor o que é simples e objetivo para as próximas entregas, evitando perca de tempo e desgaste com tarefas que não são prioridade.

F.2.2 Planejamento - 29/05/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto
Local: [Discord](#).

Pauta(s): Reunião de planejamento onde a equipe encerrou a [Sprint](#) que tinha como objetivo melhorar aquilo que tinha sido desenvolvido na [POC](#) e descreveu as tarefas que foram realizadas por cada membro da equipe.

F.2.3 Planejamento - 06/06/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto
Local: Laboratório [IFSP](#).

Pauta(s): Reunião de planejamento onde a equipe após ser orientada pelos professores definiu os tópicos que serão abordados na apresentação da primeira versão e a partir disso, definiu as tarefas a serem desenvolvidas por cada membro.

F.2.4 Planejamento 13/06/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto

Local: [Discord](#).

Pauta(s): Reunião de planejamento onde a equipe conversou e definiu melhor alguns pontos do cadastro, como: definição dos requisitos para uma senha válida e quais domínios de e-mails estarão disponíveis para que o usuário selecione. Como a entrega da primeira versão está prevista para a próxima semana, o objetivo é que até quinta feira esteja tudo documentado e até domingo as tarefas estejam concluídas.

Iniciamos uma discussão sobre abordagens para utilizarmos num futuro plano de testes e usabilidade. E também, aproveitamos para tirar do caminho uma tarefa pendente que tínhamos, da gravação do vídeo referente a apresentação da [POC](#).

APÊNDICE G – Publicações no blog da equipe

G.1 1^a Semana - 14/03 à 20/03

Figura 31 – Logo da equipe Bunka Bytes



Fonte: Os autores.

Primeiramente, bem vindos à primeira postagem da equipe! O principal objetivo deste blog é dar aos leitores a possibilidade de acompanhar nosso "diário semanal" de desenvolvimento dentro da disciplina de Prática de Desenvolvimento de Sistemas (ou [PDS](#) para os mais íntimos).

Mas antes de dar continuidade ao texto, precisamos apresentá-los a equipe Bunka Bytes, cujo nome é parte inspirado no tema do trabalho anterior (em [TDS](#)), já que este possuía um foco cultural e "Bunka" em japonês significa "cultura"; e outra parte se deve a uma referência/trocadilho com a área de informática, pois falando, é quase como se estivéssemos dizendo "Boom Kabbytes".

Integrantes da equipe:

- Anai Rojas
- Jamilli Gioielli
- José Roberto
- Julia Romualdo
- Kaiky Matsumoto

Dos integrantes, escolhemos a Jamilli como nossa representação de gerente, devido a seus conhecimentos em organização, metodologias e ferramentas de gerenciamento e por sua facilidade comunicativa.

Tendo isso em vista, nesta primeira semana da disciplina, realizamos três reuniões pela plataforma [Discord](#), nas quais conhecemos melhor nossos colegas de equipe, fizemos um Contrato Social e criamos um *Brainstorming* utilizando conceitos de *Design Thinking* na plataforma [Figma](#), que ajudou numa melhor visualização das ideias centrais para o projeto. Também tivemos acesso aos trabalhos anteriores e consultamos dois trabalhos que eram mais semelhantes ao nosso tema principal.

A partir do *Brainstorm*, a ideia que mais se consolidou foi a proposta de criação de uma comunidade de [Q/A](#) e mentorias como forma de apoio aos estudantes do [IFSP](#). Pensamos em muitas outras, mas esta acabou sendo nossa favorita, com base nos requisitos da disciplina de [PDS](#). Portanto, nosso objetivo para a segunda semana é conversar com os professores sobre a ideia e a desenvolver melhor com base nas nossas discussões e dúvidas a serem sanadas.

Por último, ao final dessa semana, criamos um *e-mail* conjunto para a criação deste *blog* e o canal no [YouTube](#), também conseguimos acessar o Subversion, criar um canal no [Discord](#), uma página de gerenciamento no [Notion](#) e uma logo para a equipe.

Por: Jamilli Gioielli e Julia Romualdo

G.2 2^a Semana - 21/03 à 27/03

Estamos de volta, leitor!

No início dessa semana, voltamos às atividades presenciais e tivemos um primeiro contato com a disciplina neste formato. A partir daí, separamos as 5 ideias que mais se destacaram do nosso *Brainstorm* da semana anterior e as compartilhamos com os professores. Recebemos algumas sugestões sobre a ideia de comunidade e nos foi recomendado documentar as ideias para enviar no ambiente do [Moodle](#) da disciplina.

Tendo isso em vista, precisamos nos reunir durante os próximos dias para passar a construção dos nossos pensamentos em formato textual e explicativo. Entretanto, enfrentamos algumas dificuldades nesse sentido, estando a maioria delas em torno do curto tempo que temos entre trabalho e escola para fazer as reuniões. Tentamos conversar na hora anterior às aulas, mas percebemos que faltava apenas documentar melhor o que havíamos conversado. Não foi possível realizar isto na escola devido aos problemas de conexão de rede, então optamos por fazermos aos poucos durante a semana e nos reunirmos depois da aula, via [Discord](#), para revisar o conteúdo. De todo modo, conseguimos fazer a entrega das duas tarefas da semana no [Moodle](#).

Nossos objetivos para próxima semana são: definir a proposta para o projeto com base no *feedback* dos professores e planejar melhor nossos dias e horários para reuniões.

Por: Jamilli Gioielli

G.3 3^a Semana - 28/03 à 03/04

Estamos de volta, Bunkers!

Nesta semana, trabalhamos na viabilidade da nossa proposta. Em aula, discutimos com os professores as funcionalidades que a comunidade pode ter, encontramos uma solução para validar os alunos cadastrados nela, enviando um *e-mail* de confirmação apenas ao *e-mail* institucional, que é de posse dos alunos, garantindo assim que os usuários sejam apenas alunos do instituto e por fim, falamos também sobre uma segunda proposta da equipe, que os professores gostaram bastante e deram forte impulso em desenvolve-la por atingir um grupo maior de usuários e ser algo que todos precisam cotidianamente, que é um sistema baseado em controlar gastos e auxiliar na organização financeira.

Entretanto, optamos por continuar com a comunidade, a qual batizamos com o nome: [IFriends](#), principalmente porque é mais próximo da nossa realidade como alunos do instituto e por não termos experiência em desenvolvimento de aplicações *mobile*, o modelo que enxergamos ser o ideal para a criação da segunda proposta. E como análise prática da viabilidade do [IFriends](#), elaboramos um formulário, via [Google Forms](#), para no início da próxima semana, enviarmos aos nossos colegas e alunos do instituto com o intuito de investigar se eles fariam uso da comunidade.

Falando agora sobre a nossa organização como equipe, ainda estamos nos adaptando com nossos horários entre estudos, trabalho e locomoção, então nosso foco será realizar reuniões rápidas e objetivas principalmente nos horários disponíveis antes das aulas começarem.

Resumo das atividades de cada membro da equipe:

- Anai - Elaborou formulário da pesquisa de viabilidade
- Jamilli - Organizou as atividades a serem realizadas pela equipe
- José, Julia e Kaiky - Trabalharam em melhorias de layout e postagens do blog

Para a próxima semana a equipe tem como objetivo estudar as tecnologias e linguagens a serem utilizadas no desenvolvimento do projeto, estruturar um *backlog* para gerenciamento eficiente das atividades, trabalhar na identidade visual e design da marca [IFriends](#).

Por: Julia Romualdo

G.4 4^a Semana - 04/04 à 10/04

Estamos de volta, Bunkers!

No inicio desta semana, realizamos a pesquisa de viabilidade da nossa proposta inicial da comunidade, para isto, elaboramos um formulário via [Google Forms](#), que ficou disponível para receber respostas de segunda-feira (04/04) à sexta-feira (08/04) e todos os integrantes da equipe ficaram responsáveis por enviar o endereço de compartilhamento - *link* - nos grupos de [WhatsApp](#) para os alunos da instituição - público-alvo da nossa proposta - respondessem a dez perguntas e compartilharem algumas experiências como alunos do [IFSP](#) que ajudassem a equipe a compreender se a proposta era ou não viável.

Ainda nesta semana, recebemos dos professores as orientações para apresentação da proposta inicial, juntamente com a entrega da documentação e estudo de dois projetos anteriores. Para realização desta tarefa, durante a semana a equipe se organizou da seguinte forma:

- Anai - Criou *branchmarketing* para a aplicação e estudou o projeto WebLab.
- Jamilli - Organizou as atividades a serem realizadas pela equipe e estruturou as documentações.
- José - Estudou as tecnologias a serem utilizadas no projeto e também do projeto Monitorando.
- Julia - Compilou dados da pesquisa de viabilidade de proposta e estudou o projeto Monitorando.
- Kaiky - Estudou as tecnologias a serem utilizadas no projeto e também do projeto WebLab.

Para a próxima semana a equipe tem como objetivo apresentar a proposta e estudar os *feedbacks* fornecidos pelos professores e colegas durante a apresentação.

Por: Julia Romualdo

G.5 5^a Semana - 11/04 à 17/04

Estamos de volta, Bunkers!

Iniciamos esta semana com a apresentação da proposta inicial da comunidade [IFriends](#) aos nossos professores e colegas de classe, dos quais recebemos orientações e *feedbacks*. Neste mesmo cenário, também participamos da apresentação das outras equipes e compartilhamos nossos *feedbacks* aos mesmos.

Após a apresentação, a equipe se reuniu na biblioteca do IFSP para realizar uma retrospectiva, com o objetivo de avaliar o funcionamento da equipe durante a intensa semana de trabalhos que tivemos para realizar a entrega da proposta inicial. Aproveitamos este momento, para decidir os pontos a serem melhorados na apresentação para realizarmos a gravação e entrega do vídeo da proposta e também ajustamos alguns pontos que faltavam ser encaixados sobre as tecnologias que serão utilizadas no projeto.

Além disso, aproveitamos o feriado para revisar alguns conceitos que tínhamos dúvidas e procurar possíveis melhorias para nossa apresentação e documentação. Uma das coisas que percebemos era que ainda precisávamos validar o arquivo equipe.yaml no yamllint, já que ele não estava sendo refletido na página de Blogs de Trabalhos, por isso aproveitamos para ajustá-lo de acordo com os apontamentos que foram dados pelo validador.

De todo modo, a equipe tem como meta para a próxima semana a atualização das fontes do projeto de acordo com o *feedback* que será dado pelos demais colegas e pelos professores após a apresentação da proposta. Além disso, pretendemos postar o vídeo da proposta - já com as melhorias - e também entregar nossas avaliações sobre as demais equipes.

Por isso, contamos com os feriados para adiantar o máximo de atividades possíveis e já pensar em alguns itens de *backlog* para que possamos iniciar a primeira *sprint* com o cronograma do projeto já bem definido. Pensando nisso também, a equipe não se dividiu como na semana anterior para a execução das tarefas, já que neste primeiro feriado focaremos juntos em tratar das melhorias, estudar as tecnologias propostas e trabalhar no planejamento do projeto.

Por: Julia Romualdo e Jamilli Gioielli

G.6 6^a Semana - 18/04 à 24/04

Estamos de volta, Bunkers!

Nesta semana, iniciamos a aula de segunda-feira de forma bem produtiva, pois conversamos bastante com os orientadores a respeito das próximas *sprints* a serem planejadas e executadas pela equipe. Assim partimos para a criação do *backlog* do produto com o uso das histórias de usuário, depois definimos as tarefas da semana já nos planejando para os três épicos a serem elaborados para a entrega da Prova de Conceito, sendo eles: Gestão de Perguntas, Gestão de respostas e Gestão de Eventos.

Além disso, aproveitamos o feriado de Tiradentes para nos reunirmos através da plataforma [Discord](#), com o objetivo de terminar a definição das histórias de usuário, neste momento muitos pontos sutis sobre a aplicação, que poderiam se tornar inimigos da equipe

futuramente, foram levantados e anotados para discutir com os professores. Também votamos por meio do *Planning Poker* - descrito pela metodologia *Scrum* - para entendermos sobre uma estimativa de esforço para que cada história seja concluída. Junto a isso, também podemos elencar alguns requisitos não funcionais e regras de negócio.

Desta forma, as tarefas realizadas pela equipe durante esta semana foram organizadas da seguinte forma:

- Anai e Jamilli - Iniciar a prototipagem das telas.
- José e Kaiky - Iniciar a modelagem de dados.
- Julia - Iniciar os ajustes na documentação.

Para a próxima semana, além de dar início as tarefas da *sprint*, a equipe tem o objetivo de terminar as tarefas de planejamento e apresentá-las aos orientadores para que possamos esclarecer dúvidas e saber os pontos a serem melhorados.

Por: Julia Romualdo

G.7 7^a Semana - 25/04 à 01/05

Estamos de volta, Bunkers!

Iniciamos esta semana validando com os professores as atividades que a equipe esteve realizando desde a semana passada, fomos orientados quanto a melhor organização dos tópicos do documento, a remover algumas histórias dos épicos para a **POC**, - pois alguns pontos estavam fugindo da ideia da **POC**, que é provar que o conceito principal, ou seja, o fluxo principal da aplicação está funcionando -, a realizar a entrega do épico de Gestão de Eventos, na **POC**, apenas se sobrar tempo, ajustar o diagrama de entidade relacionamento e o diagrama de casos de uso. Aproveitamos este momento ainda em aula para iniciar as configurações do **Gource**, pensando em futuras entregas da disciplina.

Devido ao final de bimestre, a equipe esteve ocupada com as atividades de outras disciplinas e não conversou muito durante esta semana, mas os componentes continuaram no desenvolvimento - visando o término - das atividades propostas na semana passada, organizadas da seguinte forma:

- Anai - Terminar a prototipagem das telas.
- José e Kaiky - Ajustar a modelagem de dados.
- Jamilli - Terminar a prototipagem das telas e ajustar os tópicos de gerenciamento e metodologias da documentação.

- Julia - Adicionar os apêndices na documentação e pesquisar sobre o [Gource](#).

Para a próxima semana a equipe tem como objetivo desenvolver a Prova de Conceito e a documentação relacionada a mesma.

Por: Julia Romualdo

G.8 8ª Semana - 02/05 à 08/05

Estamos de volta, Bunkers!

Esta semana devido a problemas na infraestrutura do encanamento do campus IFSP não tivemos aulas de maneira presencial e poucos professores ministraram no formato [EAD](#), aproveitamos então o plantão de segunda-feira com os orientadores para alinharmos o fluxo de usuário e o protótipo para a [POC](#). No decorrer da semana utilizamos os horários que seriam destinados às aulas para realizarmos as tarefas de desenvolvimento da [POC](#) (desenvolvimento, documentação e apresentação), para isto a equipe ficou organizada da seguinte maneira:

- Anai - Realizar ajustes finais no protótipo e ajustar a documentação.
- José e Jamilli - Configurar o ambiente e desenvolver o [Front-end](#)
- Julia - Realizar ajustes na documentação e gerar vídeo do [Gource](#).
- Kaiky - Configurar o ambiente e desenvolver [Back-end](#) e Banco de Dados.

Para a próxima semana a equipe tem como objetivo apresentar a Prova de Conceito e realizar reunião retrospectiva sobre o desenvolvimento da [POC](#).

Por: Julia Romualdo

G.9 9ª Semana - 09/05 à 15/05

Estamos de volta, Bunkers!

Iniciamos esta semana com a apresentação da Prova de Conceito para a turma e orientadores, e devemos dizer que nosso maior inimigo para esta entrega com certeza foi o tempo, pois deixamos de cumprir alguns requisitos da bíblia do Ivan porque nos faltou tempo hábil para concluir alguns tópicos ali estipulados. Ainda assim, acreditamos que conseguimos demonstrar que o fluxo principal da nossa aplicação estava funcionando.

Por outro lado, pudemos cumprir grande parte dos requisitos necessários para as entregas do primeiro bimestre, faltando apenas nossa planilha de avaliação da equipe, que

foi postada no [SVN](#) nessa semana, um pouco tarde devido a um imprevisto ocorrido com o sistema (que ficou fora do ar durante algumas horas).

A equipe tinha como um dos objetivos para essa semana a realização da reunião retrospectiva referente ao desenvolvimento da [POC](#), porém devido a alta demanda que as outras disciplinas exigiram durante a semana - provas, apresentações e outras atividades -, não encontramos um bom momento para nos reunirmos, ficando isso como uma meta para a próxima semana, juntamente com o planejamento para as próximas etapas do desenvolvimento do [IFriends](#). Além disso, precisamos atualizar nosso canal no [YouTube](#) com os vídeos para o segundo bimestre, visto que não conseguimos gravá-los e editá-los até o momento.

Por: Julia Romualdo e Jamilli Gioielli

G.10 10^a Semana - 16/05 à 22/05

Estamos de volta, Bunkers!

Podendo respirar um pouco mais após a intensa semana de entrega de atividades que tivemos durante a semana passada, iniciamos os trabalhos novamente com a reunião de retrospectiva referente a entrega da [POC](#), com isso chegamos as seguintes conclusões: do que foi bom, consideramos a entrega da [API](#) completa; para o que foi ruim, por outro lado, consideramos a falta de tempo, pois por mais que soubéssemos como desenvolver um requisito, como a internacionalização ou o vídeo de demonstração, não sobrou tempo para fazer; e por último, outro problema, enfrentado não apenas por nossa equipe mas de maneira geral na turma, foi a incerteza e a definição errada que construímos sobre a [POC](#) a partir de experiências anteriores, por isso percebemos que não conseguimos definir o que seria extremamente essencial para essa entrega e pode ser que tenhamos focado mais em coisas adicionais do que no principal. Assim, como plano de ação escolhemos que precisamos definir melhor o que é simples e objetivo para nossas entregas, para evitar que tenhamos confusões desnecessárias que acabem dificultando a entregarmos o que era necessário.

Os professores também realizaram alguns apontamentos a partir das nossas entregas relativas do primeiro bimestre, para que possamos realizar os alinhamentos necessários, tais como: representar o [Heroku](#) na arquitetura, postar os vídeos da apresentação da [POC](#) no [YouTube](#), realizar *commits* com mais frequência e enviar os arquivos do [LATEX](#) que estamos usando no Overleaf.

Além disso, os professores separaram uma parte da aula para assistirmos a alguns projetos dos alunos da turma 231 em [PJI](#), e permitiu que fizemos apontamentos e sugestões para os colegas sobre suas ideias.

Tendo realizado esses dois momentos de alinhamento a equipe documentou tudo, organizou as tarefas e ajustes que deveriam ser realizados durante a semana e partimos para os ajustes. Conseguimos enviar e ajustar o que faltava para o primeiro bimestre, como a questão do [Gource](#) e ainda incluímos alguns requisitos que faltavam na aplicação e na [API](#) (como a inclusão do *Swagger UI* e criptografia da autenticação). Como a próxima semana será de conselhos de classe, a equipe espera poder trabalhar mais no projeto e dar continuidade no desenvolvimento do mesmo, além de reavaliarmos o que for necessário conforme os demais *feedbacks* dos professores (que ainda serão feitos sobre a [POC](#)).

Por: Julia Romualdo e Jamilli Gioielli

G.11 Como utilizar o Yamllint: validador de arquivos .yaml

Um dos primeiros itens necessários para iniciar os trabalhos na disciplina de [PDS](#) é que "após a criação da pasta no repositório [Subversion](#) um arquivo **equipe.yaml** deve ser criado com os dados da equipe e projeto de acordo com modelo no repositório". Estes dados nada mais são do que aqueles existentes na página [Blogs de Trabalhos do Dicas Ivan](#), mas para que os trabalhos criados sejam expostos nesta página é preciso seguir alguns critérios na criação do arquivo, e o primeiro deles é a utilização de um validador de arquivos com extensão *.yaml*, isto é, o [yamllint](#).

Figura 32 – Logo Yamillint



Fonte: Icons For Free.com

Diante da dificuldade de alguns colegas em validar o arquivo e após fazer algumas pesquisas sobre o validador, percebi que a maior parte dos conteúdos sobre o assunto está em inglês e, arquivos dessa natureza possuem muitas especificações minuciosas para serem validadas, e muitas vezes a documentação do [Yamllint](#) não as deixa tão claras ao leitor. Por isso, resolvi compartilhar um pouco do que entendi sobre o processo de validação do arquivo, o que inclui a instalação do validador, erros comuns após a validação e a espera até que o projeto apareça na página de [Blogs de Trabalhos](#).

O que é o yamllint e por que devo validar um arquivo .yaml?

Segundo a documentação do [Yamllint](#), o validador é inspirado no [eslint](#) do JavaScript, uma ferramenta para análise **estática do código**. Isso quer dizer que seu objetivo é acusar erros (de programação e de estilo), *bugs* e construções suspeitas sem precisar executar o programa. Ou seja, a análise estática de um validador tem como objetivo trazer uma lista de erros, *warnings* e pontos de melhorias do seu código.

O que o [Yamllint](#) faz, basicamente, é este processo de *linting*/análise estática do conteúdo presente em um arquivo .yaml (ou .yml), procurando, não só problemas de sintaxe, como estranhezas e problemas estilísticos.

Ademais, a linguagem utilizada nesses arquivos, o YAML, é uma "linguagem de serialização de dados" e por isso é muito usada para arquivos de configuração, pois gerenciar configurações de um projeto é importante para manter a consistência do seu sistema e para documentar mudanças nas aplicações - mas isso é assunto pra outro *post*.

No final, o que você precisa saber é que a formatação do seu arquivo é necessária para que o sistema que vai "lê-lo" entenda corretamente as instruções que você deseja passar. Veja algumas coisas importantes sobre a formatação de arquivos.yaml:

- O YAML utiliza recuo no estilo [Python](#) para alinhamento;
- Caracteres de tabulação (os chamados "\t" ou apenas "TAB") não são permitidos, por isso utilize espaços em branco;
- A forma que os espaços em branco estão dispostos faz diferença na formatação do arquivo e pode apontar erros no validador. Por exemplo: esquecer de colocar um espaço após o caractere ":";
- Não há símbolos de formatos comuns, como chaves, colchetes, tags de fechamento, etc;
- Os arquivos .yaml são estruturados em mapas ou listas;
- As listas têm ordem específica e sua sequência começa com um traço (-) seguido de um espaço.

Existem outras coisas a se atentar, mas isso você vai descobrir quando rodar o [Yamllint](#). Se quiser saber mais sobre o YAML, recomendo ler [este artigo da Red Hat](#) (tirei algumas coisinhas de lá).

Como instalar o Yamllint?

Antes de começar a falar da instalação propriamente dita, considerando que você esteja utilizando o sistema operacional **Windows**, é importante que você **se certifique**

se possui o [Node.js](#) (com npm) ou o [Python](#) (com o pip) instalados em sua máquina. Caso esteja utilizando algum sistema operacional baseado em [Unix](#), você pode verificar se possui o gerenciador de pacotes mais utilizado nele e seguir as orientações da documentação do [yamllint](#) para instalar; ou utilizar o [Python](#) mesmo (também funcionará). Mas, se preferir um caminho alternativo, siga as próximas instruções.

Em sistemas tipo Unix (alternativa)

1. Primeiro, abra seu terminal Bash e verifique já se possui, na sua pasta \$HOME, o arquivo `.bash_profile` (é nele que ficam suas variáveis de ambiente), com o comando:

```
open $HOME/.bash_profile;
```

2. Caso ele não exista, crie-o na sua pasta de usuário e depois siga os passos para a instalação do gerenciador de pacotes [Homebrew](#);
3. Dependendo do seu tipo de [SO](#), pode ser que o comando: `brew` não funcione após a instalação, isso acontece porque você precisa criar uma variável de ambiente para ele no seu arquivo `.bash_profile`;
4. Para isso, se o próprio instalador não lhe alertar sobre os próximos passos, insira com o comando:

```
echo 'eval "$( /home/linuxbrew/.linuxbrew/bin/brew shellenv )"'>> $HOME/.bash_profile
```

5. Depois, insira o comando para atualizar o arquivo

```
source \$HOME/.bash\_profile
```

6. Agora, basta instalar o `yamllint` com o comando

```
brew install yamllint.
```

Dica: se o sistema for muito rebelde e não reconhecer os comandos, tente fechar e abrir novamente o terminal após a instalação. Para ter mais certeza se a variável foi inserida, abra seu arquivo `.bash_profile` e verifique se ela está lá. Veja mais dicas de instalação [aqui](#).

No Windows

Antes, já deixe seu Prompt de Comando/cmd aberto e vamos lá!

Dica: para essas instalações, prefira utilizar o [cmd](#) (executando como administrador) ao invés do *Windows PowerShell*, pois o segundo apresentou um erro inesperado durante a instalação do pacote.

Para usuários de Node.js

1. Baixar o pacote via

```
npm install -g yaml-lint
```

2. Verificar se a instalação foi feita com sucesso inserindo o comando

```
yamllint -v ou yamllint --version
```

"Obs.: Nesse caso, você deve tomar cuidado se seu arquivo constar como válido, mesmo não estando, pois a versão do yamllint para [Node.js](#), até este momento, vem apresentando algumas [instabilidades](#) que ainda estão sendo corrigidas. Por isso, tente atualizar sempre para a versão mais recente ou prefira utilizar o [Python](#)"

Para usuários de Python

1. Instale a última versão do gerenciador de pacotes pip, com o comando

```
pip install yamllint
```

depois, verifique a instalação foi inserido o comando

```
yamllint -v ou yamllint --version
```

Obs.: É melhor fazer a instalação global ao invés de dentro da pasta usuário, por isso, pode ser que utilizar o parâmetro **-user** na instalação cause instabilidades para rodar em outros lugares.

Como funciona a validação

Começar a validar é um processo muito simples, basta dar o comando, no mesmo diretório em que seu arquivo está,

```
yamllint equipe.yaml
```

O grande problema, na realidade, é descobrir o que deu de errado com ele. É bem provável que você encontre algo assim:

Figura 33 – Diretório

```

1:4      error    trailing spaces  (trailing-spaces)
4:4      error    wrong indentation: expected 4 but found 3  (indentation)
5:4      error    duplication of key "id-00042" in mapping  (key-duplicates)
6:6      warning  comment not indented like content  (comments-indentation)
12:6     error    too many spaces after hyphen  (hyphens)
15:12    error    too many spaces before comma  (commas)

```

Fonte: yamllint docs.

Obs.: Antes, é legal que você tenha olhado os arquivos das outras equipes que conseguiram validar o equipe.yaml, assim você consegue entender melhor como é a estrutura do arquivo. Se preferir baixar o nosso, é só acessar [aqui](#), mas também olhe lá no [Blog de Trabalhos](#) os que já estão validados (é só clicar no *link* chamado ""dos projetos).

Para entender melhor essas mensagens, recomendo que leia também as regras dispostas na documentação do [Yamllint](#), lá você vai encontrar o que significa cada erro/*warning* gerado nesse *log*, além de situações em que a estrutura disposta falhe ou passe nos testes. Porém, existem alguns erros comuns que pode ser que você se confunda mais caso leia a documentação. Alguns deles são:

- **Erro - Trailling spaces:** esse erro acontece quando você não remove o espaço depois da última palavra de uma linha. Por isso, fique atento aos números a direita do *log*, eles mostram em qual linha está aquele erro. O cursor precisaria estar "colado" no número 2 para que não apontasse o erro de *trailling spaces*. Por exemplo:

Figura 34 – Erro - *Trailling spaces*

Fonte: Os autores.

- **Erro - no new line at the end of file:** nesse caso você precisa se certificar de que a última linha do seu arquivo é uma linha em branco. Aqui percebemos que a linha 15 é a última linha do arquivo que está em branco. Dessa maneira:

Figura 35 – Erro - *no new line at the end of file*

Fonte: Os autores.

- **Erro - line too long (X > 80 characters):** esse erro indica que as linhas de um arquivo .yaml não podem ter mais do 80 caracteres, então se você se deparar com

ele, quer dizer que no lugar do "X" estará a quantidade de caracteres daquela linha, sendo essa maior do que 80. Uma dica para isso é quebrar as linhas do texto dando alguns espaços aqui e ali, e não fazer um texto corrido de uma vez (mas cuidado com as monografias).

- **Erro - found character '\t' that cannot start any token (syntax):** aqui você deve se lembrar lá de cima quando falei sobre a formatação do arquivo, pois esse erro é devido ao uso de tabulação em alguma linha do seu equipe.yaml.

Dicas extras e considerações finais Uma coisa muito importante a se atentar é o cuidado com os caracteres especiais, pois pode ser que seu arquivo lá no apareça de um jeito estranho, mais ou menos assim:

Figura 36 – Cuidado

```
descr: >-
  Pensando nas dificuldades enfrentadas pelos alunos dentro da instituição e
  perguntas e respostas mantida pelos próprios alunos. Por meio desta comunidade,
  atraídas ou sobre a instituição, dessa maneira outros alunos que faziam parte de
  onfortaveis em colaborar.
--
```

Fonte: Os autores.

Isso geralmente acontece pelo *Unicode* do seu arquivo não ser **UTF-8**. Por isso, ao salvar o arquivo, verifique se a codificação dele está no padrão esperado.

Outra coisa importante de se saber é se seu arquivo só vai ser atualizado na página do Dicas Ivan, quando o site for recompilado, o que ocorre a cada dois dias. Então, fique tranquilo se a atualização não for imediata: é normal.

Uma dica é sempre verificar o *log* do equipes.yaml, para ver quando/se a compilação já foi feita. Aliás, lá você vai perceber o mesmo tipo de validação que ocorre no [Yamllint](#), pois os arquivos que não foram compilados possuem seus erros logo abaixo.

Por último, não esqueça: o Google é seu amigo, se não encontrar algum erro super estranho na documentação, copie-o e cole-o no Google, pode ser que com um pouquinho de garimpo, você encontre a solução que procura.

Por hoje é isso pessoal, se tivermos mais dicas e relacionados para compartilhar com vocês, fiquem atentos nas próximas postagens.

Até mais! :)

Fontes:

[https://pt.stackoverflow.com/questions/330821/o-que-significa-executar-lint-no-código](https://pt.stackoverflow.com/questions/330821/o-que-significa-executar-lint-no-c%C3%B3digo)

<https://yamllint.readthedocs.io/en/stable/>

<https://www.redhat.com/pt-br/topics/automation/what-is-configuration-management>

<https://www.redhat.com/pt-br/topics/automation/what-is-yaml>

G.12 Gerando o Vídeo do Gource

Olá, Bunkers!

Estamos aqui hoje com a missão de ajudar vocês a gerar o vídeo no famoso [Gource](#). Nosso objetivo principal com este *post* é compilar o máximo de informações possíveis de forma simples e objetiva em um único lugar para que vocês não precisem ficar garimpando pedacinhos de informação em diversos *sites* para cumprir uma tarefa que não é difícil.

Primeiramente, o que é o Gource?

O [Gource](#), nada mais é do que uma ferramenta que permite visualizar o desenvolvimento de um *software* a partir dos *commits* realizados em um repositório (no nosso caso o [GitHub](#)) criando um gráfico em formato de árvore. Um dos requisitos da disciplina de [PDS](#) é que a cada bimestre, entrega e apresentação feita, deve ser publicado no canal do YouTube da equipe o vídeo gerado no Gource.

Segundo a [bíblia do Ivan](#), o vídeo do [Gource](#) deve cumprir as seguintes configurações:

- Alterar os userid do repositório por nomes dos participantes;
- Colocar uma imagem distinta e específica para cada usuário;
- Utilizar opção **key**;
- Utilizar as opções de *caption* para registrar as principais mudanças feitas no repositório;
- Os vídeos devem ter no máximo 1 minuto para cada bimestre;

Borá lá trabalhar

1. Instalação

Para instalar o [Gource](#) é só clicar [aqui](#) (pode confiar que não é vírus).

Use [este tutorial](#) encontrado no [YouTube](#) para instalar o *ffmpeg*, ele serve para converter o vídeo em .mp4, vamos entender isso melhor mais pra frente.

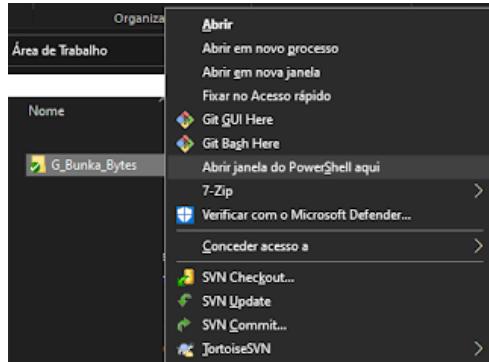
Dica: É importante que você verifique se as variáveis de ambiente estão devidamente configuradas, caso contrário o terminal irá reclamar quando você começar a rodar os comandos.

2. Testando 1,2,3

Clone o repositório da sua equipe e cole na área de trabalho ou em uma pasta de fácil acesso, faça o passo a passo localmente, nunca dentro do repositório oficial da disciplina.

Abra o terminal local ou cmd: para isso, aperte SHIFT + Botão direito do mouse e vá até a opção "Abrir o PowerShell aqui" ou "Abrir janela de comando aqui".

Figura 37 – Exemplo de como abrir o terminal



Fonte: Os autores.

Digite **gource** e aperte Enter

Apareceu um vídeo? ??????

Boooa, o **Gource** foi instalado corretamente, mas como alegria de Ifiano dura pouco, aperte ESC para sair pois ainda faltam os requisitos da bíblia.

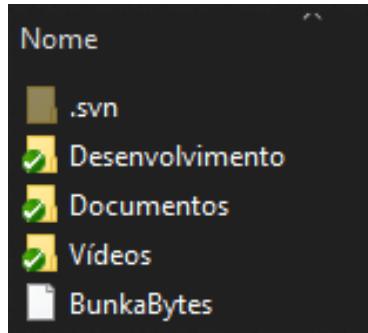
3. Alterando o log

Vamos precisar criar um log, para isso execute o seguinte comando usando o nome da sua equipe:

```
--output=custom-log BunkaBytes
```

Após executar o comando, volte na pasta clonada e veja que um arquivo foi criado.

Figura 38 – Arquivo de caption



Fonte: Os autores.

Abra este arquivo em um editor de texto de sua preferência e altere o prontuário correspondente a cada componente da equipe por seu primeiro nome, então vamos disso:

1649709679|sp1234567|A|/A2022-PDS-SEG/Bunka_Bytes/Documentos/AnaliseProjetos.pdf
1649709679|sp7654321|A|/A2022-PDS-SEG/Bunka_Bytes/Documento/PropostaInicial.pdf

Para isso:

1649709679|Anai|A|/A2022-PDS-SEG/Bunka_Bytes/Documentos/AnaliseProjetos.pdf
1649709679|Jose|A|/A2022-PDS-SEG/Bunka_Bytes/Documento/PropostaInicial.pdf

Dica: No bloco de notas padrão do **Windows**, use o Ctrl+H para localizar e substituir as linhas mais rápido.

Não é obrigatório, mas para você que prefere ir verificando se as coisas estão acontecendo certinho conforme estão sendo desenvolvidas, rode o seguinte comando para começar a ver a mágica acontecendo - lembre-se de que é com o nome da sua equipe:

```
gource BunkaBytes
```

4. Colocando vossos belos rostos no vídeo

Crie uma pasta "Imagens", "Avatares" ou algo do tipo dentro do seu repositório clonado e coloque dentro dela uma foto correspondente a cada membro da equipe, recomendamos que utilizem o formato png ou jpeg. O nome da imagem deve ser idêntico ao nome que foi inserido anteriormente no *log*.

Execute o seguinte código para verificar se funcionou:

```
gource --output-custom-log BunkaBytes --key --user-dir Imagens
```

5. Criando o Caption

Crie um arquivo de *caption*, nele deve constar os *commits* mais importantes dos que estão no *log*, porém ao invés do nome do arquivo, após o pipe (|) descreva o que foi feito, exemplo:

1651981117|Jose|M|/A2022-PDS-SEG/Bunka_Bytes/Desenvolvimento

6. Gerando o vídeo

Não existe um tempo padrão ou pré-definido pelo [Gource](#), o ideal é você utilizar as *flags* e ir testando, as que utilizamos são:

- **--caption-duration 5**: define o tempo para cada caption;
- **--max-file-lag 5**: define o tempo máximo para cada caption;

Para gerar o vídeo em si, nós basicamente vamos juntar todos os comandos e *flags* já citados aqui, ficando mais ou menos assim:

```
gource BunkaBytes --key --user-image-dir Imagens --caption-file
caption --caption-duration 5 --max-file-flag 5 -o gource.ppm
```

textbf{Atenção}: veja o vídeo rodando até o final, espere que a janela feche automaticamente, caso você pressione ESC e saia antes da execução terminar, o vídeo será gerado até aquele momento apenas.

7. Convertendo para mp4

Aqui entra a parte do *ffmpeg*, é fundamental que ele esteja instalado e com as variáveis de ambiente devidamente configuradas, caso contrário não irá funcionar. Mas se tudo estiver certinho basta digitar o seguinte comando:

```
ffmpeg -y -r 60 -f image2pipe -vcodec ppm -i gource.ppm -vcodec
libx264 -preset medium -pix_fmt yuv420p -crf 1 -threads 0 -bf 0
gource.mp4
```

8. Conclusão

Volte até a pasta e veja que o vídeo estará convertido em .mp4 e você pode visualizá-lo com qualquer reproduutor de vídeo.

E prontinho! o vídeo está pronto e já pode ser publicado no [YouTube](#).

Avisos:

- Este tutorialzinho está direcionado aos usuários de Windows, pois foi neste sistema operacional que geramos o vídeo.
- A cada bimestre apenas dê um: **SVN Update** na pasta, altere o caption e continue gerando o vídeo normalmente.

- Acesse nosso canal no [YouTube](#) para ver nossos vídeos do [Gource](#) como exemplo.

Fontes:

<https://aneisiata.blogspot.com/2019/04/gource.html>

<https://blog.dyegomaas.com.br/posts/artigo-como-visualizar-desenvolvimento-com-gource/>

<https://thiagogomesverissimo.github.io/posts/gource-organization>

G.13 11^a Semana - 23/05 à 29/05

Estamos de volta, Bunkers!

Visto que nesta semana ocorreram os conselhos de classe nós não tivemos aula em alguns dias e em outros apenas de forma remota ou com carga reduzida, nós conseguimos focar um pouco mais no projeto realizando ajustes na aplicação e documentação conforme haviam sido levantados por nós e pelos professores a partir das entregas do bimestre e da POC.

Decidimos então que durante essa semana trabalharíamos nestes ajustes e para a próxima iniciaremos uma nova [Sprint](#), trabalhando em outras histórias de usuário dos épicos de Gestão de Perguntas e Gestão de Respostas, além de finalizar a história de autenticação de usuários e iniciar o épico de Gestão de Eventos, visando a próxima entrega que já seria uma versão inicial do projeto.

Falando sobre os ajustes na documentação, adicionamos o [Heroku](#) no desenho da arquitetura, criamos uma sessão para descartes/mudanças, criamos os *QR Codes* dos *links* do *deploy* e enviamos o código inicial da documentação no [SVN](#).

No [Front-end](#) da aplicação, confirmamos a criptografia e passamos as URLs para análise no SSLabs (o [Front-end](#) recebeu notas A e a [API](#), notas A+), arrumamos o *deploy* da aplicação Web (estudando também novas plataformas para ter um plano B), ajustamos as rotas do *react-router-dom*, adicionamos as requisições da [API](#) e o *redux*. Já no [Back-end](#), foi adicionado o cadastro de *tag*, a busca de perguntas por id, verificações de usuário para algumas requisições, implementação do cargo de administrador na [API](#), totalizador de curtidas na pergunta e de respostas na pergunta, além da inclusão dos primeiros testes unitários. Foram alteradas as [URI](#) das requisições e a [API](#) agora faz uso do *Swagger 3* para a documentação, com autenticação na própria interface.

As tarefas realizadas por cada componente da equipe durante a semana foram:

- Anai e Julia - Realizaram os ajustes e configurações na documentação.
- Jamilli - Arrumou o deploy da aplicação Web, ajustou as rotas e iniciou a captura de *token* de usuário via [JWT](#).

- José - Adicionou as requisições faltantes da API no Front-end e o redux.
- Kaiky - Adicionou os ajustes e alterações no Back-end e banco de dados.

Por: Julia Romualdo

G.14 12^a Semana - 30/05 à 05/06

Estamos de volta, Bunkers!

Nesta semana, como previsto, demos início a uma nova Sprint para iniciar a história de usuário de Autenticação de Usuários e finalizar os épicos de Gestão de Perguntas e Respostas, principalmente na parte do Front-end, visto que para a POC demos mais atenção a nossa API - que já possui todos os endpoints esperados para estes épicos, passando apenas por alguns ajustes conforme novas necessidades foram surgindo. Entretanto, não conseguimos nos reunir durante a semana para discutirmos o andamento das tarefas em conjunto de forma síncrona/presencial, mas, além de termos falado mais de forma assíncrona, aproveitamos a aula de segunda-feira para conversarmos sobre o andamento do projeto e sanar as dúvidas pendentes com os professores, não somente sobre a aplicação, mas também sobre a documentação - visto que recebemos também o feedback sobre a documentação da POC. Dessa forma, as tarefas ficaram mais claras para que continuássemos com a mão na massa ao longo da semana, já que no mesmo dia definimos quais eram as prioridades faltantes e depois fomos apenas atualizando conforme o decorrer dos dias.

As tarefas realizadas por cada membro da equipe ficaram da seguinte forma:

- Anai - Ajustou a documentação para a entrega da primeira versão.
- Jamilli - Adicionou autenticação no Front-end, ajustou alguns erros e melhorou a parte visual.
- José - Alterou requisições para compatibilidade com a API e iniciou a funcionalidade de busca de perguntas.
- Julia - Estudou como gerar as métricas do StatSVN e iniciou os testes relacionados.
- Kaiky - Modelou o diagrama de classes e fez os ajustes necessários na API.

Para a próxima semana, pretendemos finalizar o que faltar no desenvolvimento da aplicação para a primeira entrega, assim como os documentos e a apresentação necessária.

Por: Julia Romualdo

G.15 13^a Semana - 06/06 à 12/06

Estamos de volta, Bunkers!

Nesta semana, nós demos continuidade a [Sprint](#) de desenvolvimento dos épicos de Gestão de Respostas e Gestão de Perguntas, além da autenticação de usuário, e fizemos bastante progresso no estabelecimento das conexões da [API](#), no entanto, deixamos muitas tarefas para o final de semana e percebemos que vamos precisar de mais uma [Sprint](#) para conseguir finalizar tudo com todos os ajustes que nos propomos. Temos a aplicação funcionando bem, porém precisamos finalizar detalhes importantes para a usabilidade. Com relação a [API](#), estamos bem encaminhados, inclusive na parte de testes, pois conseguimos incluir a biblioteca [JaCoCo](#) para cobertura de testes. No final, nosso maior inimigo tem sido o tempo, visto que também precisamos documentar algumas coisas da aplicação antes da primeira entrega.

No momento de aula, conversamos sobre o funcionamento e falhas que estavam havendo na comunicação da equipe, fomos orientados com relação ao que seria avaliado e a partir disso, definimos os tópicos que seriam abordados na apresentação e as prioridades para desenvolvimento, pois com muitas tarefas e pouco tempo é importante que todos saibam o que é essencial para a entrega.

Como temos tido dificuldade em nos reunirmos presencialmente, decidimos que aproveitaremos melhor os momentos em sala de aula que estaremos juntos - e com os professores por perto - para tirar dúvidas e definir as tarefas que serão realizadas no restante da semana e também, pelo mesmo motivo, priorizaremos a comunicação assíncrona da equipe, principalmente pelo [WhatsApp](#), mandando resumos do que foi feito por cada um durante a semana, dessa forma garantimos também que todos saibam o que está acontecendo em cada parte do projeto.

As tarefas realizadas por cada membro da equipe foram organizados da seguinte maneira:

- Anai - Trabalhou na documentação visando a próxima entrega, ajustou o protótipo e iniciou a criação dos slides para a apresentação.
- Jamilli - Arrumou o problema do [CORS](#), refatorou códigos e ajustou o [CSS](#) e iniciou a documentação sobre a segurança da aplicação.
- José - Refatorou alguns códigos, criou componentes reutilizáveis, ajustou *layout*, corrigiu *bugs* e criou a tela de cadastro.
- Julia - Ajustes na documentação e testes na geração das métricas no [StatSVN](#).

- Kaiky - Terminou os testes unitários de Pergunta, Adicionou o [JaCoCo](#) na [API](#) para cobrir os testes, arrumou o diagrama de classes, [DTR](#) e [DER](#) e iniciou a documentação do plano de testes.

Na quinta-feira (09/06), a diretoria do campus comunicou que as aulas estariam suspensas até o dia 19/06 por conta do crescimento do número de casos de Covid-19 entre alunos e servidores, então a apresentação da primeira versão aconteceria de forma remota, no mesmo dia previsto (13/06). Entretanto com muita argumentação e com os deuses da programação a nosso favor, os professores concordaram em deixar as apresentações e as entregas para o dia 20/06 quando retornarmos ao formato presencial, com isto nosso objetivo para a próxima semana é continuar desenvolvendo e aprimorando aquilo que foi proposto para a entrega.

Por: Julia Romualdo

G.16 14^a Semana - 13/06 à 19/06

Estamos de volta, Bunkers!

Nesta semana, devido ao adiamento da entrega e apresentação da primeira versão na semana passada, pudemos criar uma [Sprint](#) extra, que tinha como objetivo realizar melhorias do que havia sido proposto na [Sprint](#) anterior. Aproveitamos então para nos reunirmos no início da semana e trabalhar na definição dos atributos do cadastro, como: requisitos de senha e quais domínios de *e-mails* estariam disponíveis para seleção. Começamos também, a pensar em métodos e abordagens para aplicarmos planos de testes e usabilidade em um futuro próximo, com objetivo de validar com os [Friend's](#) - como chamamos os usuários do [IFriends](#) - se nosso sistema está intuitivo para uso. Ainda neste momento de reunião, aproveitamos para tirar uma tarefa pendente do caminho e gravamos o vídeo com relação a [POC](#). Com todos sabendo suas tarefas e o que tinha que ser feito, partimos para a execução durante a semana, mantendo a comunicação assíncrona via [WhatsApp](#).

No [Front-end](#) da aplicação, trabalhamos no *feedback* visual, padronização e validação de campos no *login* e no cadastro, internacionalização, menu superior com mais opções (como os dados do usuário logado, botão de internacionalização e *logout*), *timeout* na sessão para verificar expiração do *token*, tratamento e captura de erros da [API](#) com mensagens, além das rotas que foram ajustadas de modo que o [Friend](#) já logado não possa acessar as telas de *login* e cadastro. Além disso, também foi feito um teste inicial com a hospedagem de imagens via [API](#) do [ImgBB](#).

No [Back-end](#), os *bugs* de atualização da pergunta forma ajustados, uma requisição para devolver as categorias foram criadas, os testes do *service* da categoria foram feitos,

a segurança da [API](#) obteve nota A+ e criou-se a tabela "curso" no banco de dados.

Esta semana todos precisaram escrever um pouco na documentação, pois os pontos do desenvolvimento como testes, novas tecnologias e segurança, precisavam ser documentados. Entretanto, especificamente na estrutura da documentação, nós atualizamos os *posts* do *blog*, atas das reuniões e a modelagem do banco de dados ([DER](#), [DTR](#) e [DD](#)). Com base no *feedback* dos professores na documentação da [POC](#), melhoramos a introdução, de forma que ela ficasse mais aprofundada na temática do projeto. Adicionamos uma tabela de métricas do desenvolvimento geral do projeto, já que estamos com dificuldade e não conseguimos gerar as métricas no [StatSVN](#), criamos também a sessão de análise de concorrência para compararmos o [IFriends](#) com o [Moodle](#) e [Scoold](#) e por último, cuidamos dos apêndices, movendo para lá a prototipagem, dicionário de dados, histórias de usuários e documentos anteriores.

Cada componente da equipe realizou as seguintes atividades durante a semana:

- Anai - Criou a sessão de análise da concorrência, cuidou dos apêndices e montou o *layout* dos slides da apresentação.
- Jamilli - Traduziu dos textos para a internacionalização, fez ícone do usuário no menu ter mais ações, implementou *timeout* na sessão do usuário, a tratou erros das requisições vindas [API](#) e os redirecionamento das rotas.
- José - *Feedback* visual, padronização e validação de campos, tela de cadastro, inclusão de uma [API](#) para salvar imagens na nuvem e configuração da internacionalização.
- Julia - Atualização *posts blog*, atas das reuniões e modelagem de dados, melhoria da introdução e criação da tabela de métricas de desenvolvimento do projeto.
- Kaiky - Arrumou os *bugs* da pergunta, fez a requisição para devolver as categorias criadas, os testes do *service* da categoria, a segurança da [API](#), ajustes no banco e modelagem de dados.

Nós e as outras equipes da turma estamos enfrentando um problema desde sábado de manhã (18/06), que é a queda do [SVN](#), sem ele nós não estamos conseguindo *commitar* as coisas que estamos terminando e nem puxar os dados que estão no repositório para preencher a tabela de métricas e gerar o vídeo do [Gource](#). Por isso, estamos nos virando com um *backup* que fizemos no [GitHub](#) e tentando puxar as novas mudanças de cada um para a versão final.

Para a próxima semana temos como objetivo apresentar a primeira versão e realizar a reunião retrospectiva da entrega.

Por: Julia Romualdo

APÊNDICE H – Documentos Entregues

**IFSP - Instituto Federal de Educação, Ciência e Tecnologia
Câmpus São Paulo**

ANAÍ VILLCA ROJAS	SP3029085
JAMILI VITÓRIA GIOIELLI	SP3027473
JOSÉ ROBERTO CLAUDINO FERREIRA	SP3024369
JULIA ROMUALDO PEREIRA	SP3023061
KAIKY MATSUMOTO SILVA	SP185075X

**PROPOSTA INICIAL - IFRIENDS: COMUNIDADE DE
APOIO AOS ALUNOS**

Proposta de projeto de sistema para disciplina de PDS

Professor: Johnata Souza Santicioli

Professor: Carlos Henrique Veríssimo Pereira

IFSP - Instituto Federal de Educação, Ciência e Tecnologia
Câmpus São Paulo

Curso Técnico em Informática Integrado ao Ensino Médio

PDS - Prática para Desenvolvimento de Sistemas

São Paulo - SP - Brasil

2022

Listas de ilustrações

Figura 1 – Logo do projeto	7
Figura 2 – Modelo Colmeia	8

Listas de abreviaturas e siglas

CSS	<i>Cascading Style Sheets</i> - Folha de Estilo em Cascatas - Citado em 11
HTML	<i>HyperText Markup Language</i> - Linguagem de Marcação de Hipertexto - Citado em 11
IDE	<i>Integrated Development Environment</i> - Ambiente de Desenvolvimento Integrado - Citado em 12
IFSP	Instituto Federal de Educação, Ciência e Tecnologia de São Paulo - Citado em 5 , 7 , 9
MVC	<i>Model-View-Controller</i> - Modelo-Visão-Controle - Citado em 11
PAP	Programa de Auxílio Permanência - Citado em 10
PDS	Prática de Desenvolvimento de Sistemas - Citado em 5 , 13
SGBD	<i>Data Base Management System</i> - Sistema de Gerenciamento de Banco de Dados - Citado em 11
SQL	<i>Structured Query Language</i> - Linguagem de Consulta Estruturada - Citado em 11

Sumário

1	INTRODUÇÃO	5
1.1	Objetivo	5
1.2	Justificativa	6
2	ESPECIFICAÇÕES INICIAS PARA A PROPOSTA	7
2.1	Design da marca	7
2.2	Pesquisa de Viabilidade	8
2.2.1	Resultados	9
2.3	Funcionalidades iniciais	9
3	TECNOLOGIAS E FERRAMENTAS PRETENDIDAS	11
3.1	Ferramentas de Apoio	11
4	EQUIPE	13
	REFERÊNCIAS	14
	GLOSSÁRIO	14
	APÊNDICES	16
	APÊNDICE A – PERGUNTAS DESENVOLVIDAS PARA A PESQUISA DE VIABILIDADE	17

1 Introdução

O presente documento é resultado da proposta de um projeto cujo objetivo é sustentado no planejamento e na execução de um sistema para a Web através do aprendizado obtido nas matérias técnicas do Curso Técnico de Informática, realizado no IFSP, e como forma de trabalho de conclusão de curso.

Tendo isto em vista e diante dos desafios presentes na lista de requisitos e orientações propostas para a iniciação deste projeto, os integrantes da equipe Bunka Bytes reuniram-se em busca de encontrar a solução que melhor se encaixasse em seus objetivos e nos da disciplina de [PDS](#).

É pensando em soluções viáveis que a equipe voltou seu olhar para sistemas que contribuem para a criação de comunidades colaborativas dentro da área de desenvolvimento de sistemas, que, segundo [Rosa \(2008\)](#), foi fortemente difundida pelas comunidades de Código Aberto (do inglês, *Open Source*), sendo primeiramente criada pela cultura *hacker*, na qual afirma que a paixão e o interesse dos *hackers* nas soluções foi uma das principais propulsoras do espírito colaborativo.

Além deles, foi trazido ao debate as possibilidades de aliar as principais dificuldades que os integrantes observaram durante sua vida acadêmica no [IFSP](#), a um sistema que pudesse suprir determinadas necessidades dos alunos, como os questionamentos que começam a surgir com mais frequência conforme o início dos estudos é dado, sendo eles em âmbitos diversos como: sobre a instituição de ensino, matérias e assuntos tratados no ensino médio, dúvidas sobre os conteúdos técnicos ou até mesmo a busca por um apoio educacional - como ocorrem nas monitoriais.

O [IFriends](#) surge nesse cenário, no qual a criação de uma comunidade de estudantes que colaborassem entre si, pudesse instigar o interesse dos alunos em ajudarem uns aos outros de maneira acessível e prática, onde uma dúvida estivesse a um palmo de distância.

1.1 Objetivo

O objetivo deste projeto é tentar instigar o interesse dos estudantes que compreendem o [IFSP](#) para que possam criar espaços colaborativos entre si, através de um sistema no qual os usuários interajam entre perguntas e respostas, fornecendo caminhos para o esclarecimento de suas dúvidas sobre a instituição de ensino e as áreas e disciplinas que a ela pertencem.

Dessa forma, o objetivo será aplicado através da construção de uma plataforma de perguntas, respostas e mentorias para a Web, na qual qualquer usuário poderá sub-

meter uma pergunta para ser respondida pelos outros membros da comunidade; além de possibilitar que estudantes possam escolher se tornar mentores sobre determinados assuntos, disponibilizando recursos para a criação de anúncios de eventos de monitorias (cuja localidade a eles deve competir) dentro de seus perfis de usuário.

Tendo isso em vista e pensando numa melhor interatividade entre os usuários, o sistema deve passar por um processo de [gamificação](#) em algumas de suas funcionalidades, como as votações para respostas e perguntas mais relevantes e os atributos dos usuários mais ativos - assim como outros exemplos que devem ser adicionados durante o planejamento do projeto.

1.2 Justificativa

A reflexão com relação às formas complementares de aprendizagem é importante para a ampliação dos conteúdos interessados tanto aos alunos, quanto aos seus professores, pois permite que passem a enxergar, juntos, o ensino como um meio que evite a passagem de aprendizados de forma restrita e hierarquizada.

Por isso, [Fernandes \(2011\)](#) traz em sua pesquisa que o desafio da construção de sociedades de aprendizagem parte do pressuposto de que os recursos tecnológicos disponibilizados atualmente permitem aos estudantes aprenderem dentro e fora da escola e das mais variadas formas. Assim, para ele, a melhor forma se dá construindo comunidades sustentadas pelo uso de tecnologias Web.

O autor dá continuidade na exposição desse fenômeno ao atribuir o sucesso da potencialização da aprendizagem complementar e das relações sociais à Web 2.0. Isto pois, de acordo [Fernandes \(2011\)](#), a mesma permitiu novas formas e possibilidades de criação de conteúdos e possibilitou o enfoque a uma aprendizagem motivada pelos interesses do aluno, no qual ele deve assumir um papel exploratório nessa experiência, da qual poderá colher ensinamentos significativos, explica [Fernandes \(2011\)](#).

Com o objetivo de atrair atenção para o tema, o projeto tem como principal missão, permitir que os estudantes possam usufruir de uma ferramenta gratuita que proporcione a suavização do seu processo de aprendizagem, quando seus próprios colegas contribuirão com suas experiências passadas, além de deixarem um histórico para possibilitar um caminho menos árduo aos estudantes que virão. Por isso, espera-se que, com este projeto, a instituição de ensino também seja um agente na construção de uma comunidade propícia para estudantes, na qual poderão unir-se em razão de dúvidas comuns, e assim incentivarem a disseminação de uma cultura colaborativa dentro de seus espaços.

2 Especificações iniciais para a proposta

Os assuntos abordados nesta sessão têm como objetivo apresentar especificações sobre a proposta inicial do projeto, descrevendo os meios e resultados obtidos para a identidade visual, viabilidade e funcionalidades iniciais.

2.1 Design da marca

Cogitando a missão do projeto, surgiu a ideia de dar o nome **IFriends** para o sistema, cuja origem é a junção de duas palavras: IF e *friends*. Isto devido a elas retratarem bem o âmbito que pretende ser atingido, já que tais palavras em conjunto transmitem o significado de amigos do **IFSP**, nome ideal para um projeto que visa tornar a interação dos alunos mais favorável.

A próxima etapa do desenvolvimento inicial da marca foi a elaboração de uma logo, assim como a definição das cores iniciais do sistema. A logo foi desenvolvida por meio do **Canva**, pois a plataforma se encontrava nos intermédios necessários para a elaboração da mesma.

Figura 1 – Logo do projeto

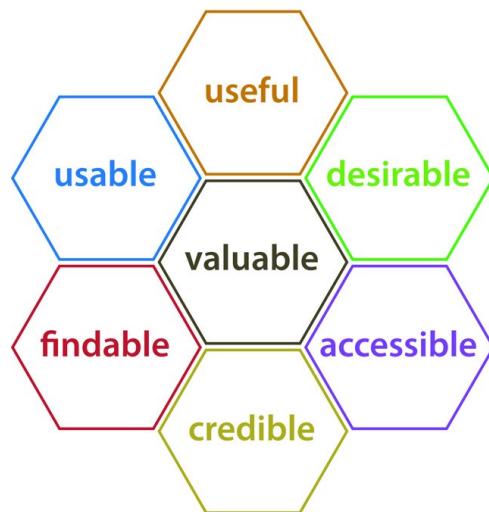


Fonte: Os autores.

Já a seleção das cores iniciais do sistema traçou um caminho através de um estudo a respeito da psicologia das cores, visto que a equipe se preocupou em passar uma boa experiência até mesmo no quesito visual. Dessa forma, se definiu o azul e suas variações como a cor principal do sistema, já que segundo **Tornos (2021)**, os tons de azul se associam a princípios como: proteção, tranquilidade, fidelidade, compromisso, verdade, estabilidade, criatividade, entre outros.

Ainda, outro ponto considerado na criação da proposta foi a experiência do usuário final, pois assim como mostrado na pesquisa realizada, a equipe se preocupou em estudar e conhecer melhor as dores deles. Visto que, segundo Reyes (2020), para tornar essa experiência agradável o sistema deve recorrer aos requisitos do modelo de colmeia desenvolvido por Peter Morville, sendo eles: útil, utilizável, desejável, acessível, confiável, localizável e valioso.

Figura 2 – Modelo Colmeia



Fonte: liferay.com

2.2 Pesquisa de Viabilidade

Foi realizada pela equipe uma pesquisa de viabilidade da aplicação, com o objetivo de verificar se o público alvo realmente estava sendo atingido, ou seja, se os alunos da instituição possuem interesse na aplicação e pretendem utilizar a mesma como ferramenta cotidiana para auxilia-los durante os estudos.

Para isto, houve a elaboração de um formulário por meio da ferramenta [Google Forms](#), no qual solicitamos que os alunos da instituição respondessem, com sinceridade, dez questões ([Apêndice A](#)) sobre a proposta. Sua divulgação ocorreu por meio do aplicativo de conversas [WhatsApp](#), onde os integrantes da equipe ficaram responsáveis por enviar o endereço de compartilhamento do formulário nos grupos de alunos conhecidos na instituição.

A realização desta pesquisa é de grande importância para a equipe avaliar a proposta que está sendo apresentada e para analisar sua viabilidade por meio dos resultados.

2.2.1 Resultados

A partir deste formulário, foram recebidas quarenta e cinco respostas acumuladas dentro de um período de cinco dias, e ao final, foi possível constatar as seguintes características dentro da maioria das respostas:

Algumas características do maior grupo que responderam ao questionário são:

- 86,7% dos estudantes são do ensino médio integrado ao técnico;
- 66,7% dos que sentiram dificuldade ao ingressar no IFSP, compartilhou em forma de resposta curta suas experiências (como, adaptação com as atividades, dificuldades em matérias específicas, falta de transparência nas informações institucionais);
- 60% nunca frequentaram ou vão raramente às monitorias;
- 85,5% usariam o sistema e acreditam que o mesmo o ajudaria academicamente;

Com base nesses resultados, a equipe entende que a proposta cumpre seu objetivo de atingir os alunos do IFSP com o desenvolvimento de uma comunidade de apoio aos assuntos enfrentados durante sua formação. Para pesquisas futuras, foi percebido, com base nas respostas, que uma melhoria na descrição das funcionalidades ajudaria o público-alvo a compreender melhor o objetivo da aplicação. Portanto, é possível validar, futuramente, dores dos usuários com pesquisas de usabilidade após melhores detalhamentos sobre o escopo do sistema.

2.3 Funcionalidades iniciais

Através da pesquisa de viabilidade e das discussões feitas anteriormente pela equipe, foram elencadas algumas funcionalidades que poderão compor o escopo inicial do projeto, conforme apresentadas nessa sessão.

- **Perguntar e responder questões:** Possibilitar que os usuários respondam e façam perguntas no fórum;
- **Gamificação para ganho de reputação a cada resposta dada:** Possibilidade de votar positiva e negativamente nas respostas e de ganhar emblemas de acordo com o aumento da sua reputação;
- **Trazer as perguntas mais relevantes na página inicial:** Exibir como página inicial para o usuário, as principais perguntas sobre temas específicos, como cursos, por exemplo;

- **Dividir as perguntas por categorias dentro da instituição:** Duas categorias: perguntas acadêmicas seriam mais voltadas aos cursos e assuntos relacionados a eles, já as perguntas institucionais seriam voltadas as entidades e burocracias da instituição, como perguntas sobre o grêmio, PAP, salas, entre outras;
- **Divulgação de monitorias no perfil do usuário:** Criação de uma aba no perfil para publicar, exclusivamente, eventos para monitorias de determinados assuntos.

3 Tecnologias e ferramentas pretendidas

Pensando no objetivo e alcance do projeto, decidimos por realizar uma aplicação Web inicialmente, a qual pode ser acessada de diferentes dispositivos de forma fácil, apenas tendo acesso a internet. Outros fatores levados em conta são a experiência da equipe com o mesmo e a disponibilidade de conteúdo sobre desenvolvimento Web.

A fim de desenvolver essa aplicação, será usado [HTML](#), [CSS](#) e [JavaScript](#) como base para o *Front-end*, podendo fazer uso de alguns [Frameworks](#) para agilizar o desenvolvimento.

Para o *Back-end* a linguagem Java, que é orientada a objetos; sua escolha foi feita pelo fato da equipe já possuir experiência com a linguagem, além de alguns já trabalharem com a mesma, ela também possui uma boa maturidade, contendo muitas informações disponíveis para ajudar no desenvolvimento. Em conjunto, será utilizado Spring Boot, o qual é um [Framework](#) Java de código aberto.

Quanto ao banco de dados, será utilizado a linguagem [SQL](#), que serve para executar comandos no banco, como consultas, e alterações nos dados ou na estrutura, ela é usada de maneira parecida entre os diferentes [SGBD](#) no mercado.

Como arquitetura de sistema, será aplicado o padrão [MVC](#), que separa o projeto em três principais camadas, model, view e controller.

Para hospedar a aplicação, será utilizado o [Heroku](#), o qual é uma plataforma de nuvem que suporta diversas linguagens, que permite a implantação, escalonamento e gerenciamento do sistema.

Como inspiração para o sistema, será estudada sobre a plataforma Scoold, a qual é uma aplicação Web de código aberto de perguntas em respostas inspirada no Stack Overflow. Ela usa de base as mesmas tecnologias do projeto, tendo isso em vista, no caso da inviabilidade de seu uso, pode ser reaproveitado ideias implementadas nessa plataforma no seu código-fonte.

3.1 Ferramentas de Apoio

Nesta seção serão especificados as ferramentas a serem utilizadas para a estruturação do sistema Web.

- **Notion:** Foi escolhido como plataforma de organização de projetos, pois este possui métodos de gerenciamento de equipe, fornecendo uma interface para vários desenvolvedores trabalharem utilizando o quadro de kanban, assim como o compartilha-

mento de arquivos, vídeos e artigos; permitindo também notificações via e-mail sobre o atualizações do projeto e de reuniões marcadas, por exemplo.

- **Discord:** Plataforma de comunicação online escolhida pela equipe, por ser a mais utilizada e conhecida entre os mesmos. Na qual é possível criar servidores privados para compartilhar informações e realizar reuniões sobre o projeto.
- **Overleaf:** Para a preparação do documento de visão, será usado o L^AT_EX, visto que este possui comando de padronização para documentos acadêmicos, facilitando a sua construção, juntamente, com editor Overleaf que oferece um ambiente compartilhado entre os membros das equipes.
- **Figma:** Na parte de modelagem do sistema e elaboração ideias de soluções de problemas, escolheu-se o Figma, visto que é uma ferramenta gratuita que possui diversas opções de edição.
- **Visual Studio Code:** A IDE escolhida para realizar a edição de código-fonte do sistema, por razões de ser um recurso da atualidade com possibilidades de várias adaptações de ambiente, principalmente as nossas tecnologias escolhidas.

4 Equipe

Tendo em vista a organização do projeto [IFriends](#), a equipe escolheu utilizar os princípios e valores do Manifesto Ágil como norteadores no seus processos de planejamento, modelagem e desenvolvimento do projeto. Deste modo, elencou-se duas metodologias ágeis que irão servir como referências para o trabalho da Bunka Bytes, sendo elas: os [Frameworks](#) Scrum e Programação eXtrema (do inglês, *eXtreme Programming*).

Portanto, para fins de organização, o Time de Desenvolvimento poderá ser dividido em subequipes, sendo elas: documentação e mídias, design e desenvolvimento ([Front-end](#) e [Back-end](#)). Nelas, estão inclusas as funções de programadores [Front-end](#) e [Back-end](#), que foram divididas entre os integrantes Jamilli Vitória Gioielli, José Roberto Claudinho Ferreira e Kaiky Matsumoto Silva; assim como as funções de designer de experiência/interface de usuário e gerenciamento da documentações e mídias do projeto, divididas entre os integrantes Anai Villca Rojas, e Julia Romualdo Pereira. De todo modo, a documentação do projeto deverá ser revisada por todo o time, a fim de manter um conhecimento melhor distribuído.

Entretanto, é importante salientar que as funções atribuídas a cada um não são exclusivas e podem variar de acordo com a necessidade de entrega do projeto. As subequipes são de responsabilidade de todos os integrantes e apenas foram divididas assim para fins de organização das tarefas do projeto. Levando isso em consideração, as funções de *Scrum Master* ou *Iteration Manager* e *Product Owner* foram adaptadas para um único papel, representado pela integrante Jamilli Vitória Gioielli. Porém, vale ressaltar que, nesse modelo escolhido, toda a equipe é responsável pela inspeção dos processos do projeto:

O Time deve ser multidisciplinar e multifuncional, possuindo todo o conhecimento necessário para criar um incremento no trabalho. [...] Não há títulos no Time, e não há exceção a esta regra. Não deve existir distinção de cargos ou funções, títulos ou senioridades, e muito menos áreas determinadas ou específicas de atuação. No Scrum todos os integrantes do Time são conhecidos como desenvolvedores.

Individualmente os integrantes do Time de Desenvolvimento podem ter habilidades específicas, mas, independentemente disso, a responsabilidade a respeito de uma entrega continua sendo do Time de Desenvolvimento como um todo ([CRUZ, 2018](#)).

Por fim, tendo em vista a finalidade acadêmica do projeto, foi estipulado um valor mínimo de duas reuniões semanais dedicadas na construção do mesmo, levando em conta a organização individual de cada membro da equipe, além da priorização para que a tomada de decisões e planejamentos sejam feitas no dia das aulas da disciplina de [PDS](#).

Referências

- CRUZ, F. *Scrum e Agile em Projetos: Guia Completo*. 2. ed. Rio de Janeiro: Brasport Livros e Multimídia Ltda., 2018. Citado na página 13.
- FERNANDES, L. Redes sociais online e educação: contributo do facebook no contexto das comunidades virtuais de aprendentes. *Universidade Nova de Lisboa, Portugal*, 2011. Citado na página 6.
- REYES, C. *Experiência do Usuário (UX): Entenda o que é de forma prática*. 2020. [Online; accessed 10. Apr. 2022]. Disponível em: <<https://www.liferay.com/pt/resources/1/user-experience>>. Citado na página 8.
- ROSA, G. C. d. Identidade cultural em comunidades de usuários e desenvolvedores de software livre: o caso debian-rs. Pontifícia Universidade Católica do Rio Grande do Sul, 2008. Citado na página 5.
- TORNOS, C. T. Significado da cor azul na psicologia. *Br.psicologia-online*, Br.psicologia-online.com, Sep 2021. Disponível em: <<https://br.psicologia-online.com/significado-da-cor-azul-na-psicologia-917.html>>. Citado na página 7.

Glossário

Back-end	Refere-se a parte que está por trás da aplicação, responsável pela manipulação de dados voltada para o funcionamento interno de um sistema - Citado em 11, 13
Canva	Plataforma de design gráfico que permite a criação de gráficos de mídia social, apresentações, infográficos, pôsteres e outros conteúdos visuais - Citado em 7
Framework	Uma abstração que une códigos comuns entre vários projetos de software provendo uma funcionalidade genérica. - Citado em 11
Framework	Um framework ou arcabouço conceitual, é um conjunto de conceitos usado para resolver um problema de um domínio específico. - Citado em 13
Front-end	Refere-se a parte visual e gráfica da interface de um sistema, elaborado por meio de outras linguagens e tecnologias. - Citado em 11, 13
gamificação	Aplicação das estratégias dos jogos nas atividades do dia a dia, com o objetivo de aumentar o engajamento dos participantes. Se baseia no game thinking, que integra a gamificação com outros saberes do meio corporativo e do design. - Citado em 6
Google Forms	Ferramenta da Google para gerenciamento de pesquisas e formulários, utilizada para coletar e registrar informações de outras pessoas. - Citado em 8
Heroku	Plataforma de nuvem como serviço que suporta várias linguagens de programação. - Citado em 11
IFriends	Nome dado ao projeto de sistemas desenvolvido, cujo significado se dá num trocadilho na junção das palavras friends (amigos, em inglês) e IF (Instituto Federal). - Citado em 5, 7, 13
WhatsApp	Aplicativo de mensagens instantâneas e chamadas de voz para smartphones. - Citado em 8

Apêndices

APÊNDICE A – Perguntas desenvolvidas para a pesquisa de viabilidade

1. Qual é o seu grau de escolaridade?
2. Qual é o seu curso?
3. Você trabalha ou faz estágio?
4. Você sentiu dificuldade em se adaptar ao entrar no IF?
5. Descreva como foi a sua experiência (com relação as dificuldades na instituição e no ensino).
6. Com que frequência você costuma ir às monitorias?
7. Você usaria um sistema de perguntas e respostas do IF?
8. Você acredita que uma comunidade de perguntas e respostas te ajudaria na sua vida acadêmica?
9. De que forma isso faria/não faria diferença para você? (Fique a vontade de responder com toda sinceridade!).
10. Gostaria de compartilhar mais alguma coisa sobre o tema? Bem, sinta-se a vontade!

APÊNDICE I – Prova de Conceito

IFSP - Instituto Federal de Educação, Ciência e Tecnologia
Câmpus São Paulo

ANAÍ VILLCA ROJAS	SP3029085
JAMILI VITÓRIA GIOIELLI	SP3027473
JOSÉ ROBERTO CLAUDINO FERREIRA	SP3024369
JULIA ROMUALDO PEREIRA	SP3023061
KAIKY MATSUMOTO SILVA	SP185075X

IFriends: uma comunidade virtual

São Paulo - SP - Brasil

2022

**IFSP - Instituto Federal de Educação, Ciência e Tecnologia
Câmpus São Paulo**

ANAÍ VILLCA ROJAS	SP3029085
JAMILI VITÓRIA GIOIELLI	SP3027473
JOSÉ ROBERTO CLAUDINO FERREIRA	SP3024369
JULIA ROMUALDO PEREIRA	SP3023061
KAIKY MATSUMOTO SILVA	SP185075X

IFriends: uma comunidade virtual

Projeto de sistema IFriends apresentado, conforme as normas ABNT, à disciplina de Prática para Desenvolvimento de Sistemas.

Professor: Johnata Souza Santicioli

Professor: Carlos Henrique Verissimo Pereira

IFSP - Instituto Federal de Educação, Ciência e Tecnologia
Câmpus São Paulo

Curso Técnico em Informática Integrado ao Ensino Médio

PDS - Prática para Desenvolvimento de Sistemas

São Paulo - SP - Brasil

2022

Resumo

O presente documento é o resultado obtido até a elaboração da POC do projeto que visa a criação de uma comunidade virtual do IFSP através do aprendizado adquirido na disciplina técnica de PDS no quarto ano do Curso Técnico de Informática, realizado no Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, Campus de São Paulo. O objetivo central deste trabalho, desse modo, é apresentar a projeção e a implementação de uma prova de conceito de uma comunidade virtual qual visa a criação de um espaço de acolhimento de alunos para alunos. Propõe-se, assim, utilizar de um método ágil e de ferramentas de desenvolvimento para passar pelos processos de engenharia do sistema, além de estimular o trabalho em equipe. Sob essa perspectiva, o projeto pôde ser apresentado abordando seu tema principal e focando nas suas funcionalidades mais essenciais, como a gerenciamento das perguntas e a gerenciamento das respostas.

Palavras-chaves: Projeto. IFriends. Comunidade virtual.

Listas de ilustrações

Figura 1 – Arquitetura Rest API	11
Figura 2 – Cronograma de Desenvolvimento Semanal	16
Figura 3 – Diagrama de Casos de Uso	24
Figura 4 – Diagrama de Entidade e Relacionamento	25
Figura 5 – Diagrama de Tabelas Relacionais	26
Figura 6 – Mapa mental	32
Figura 7 – Página inicial	33
Figura 8 – Pergunta e respostas	34
Figura 9 – Cadastro de perguntas	35
Figura 10 – API IFriends	36
Figura 11 – Aplicação IFriends	37

Listas de quadros

Quadro 1 – Distribuição de tarefas	15
Quadro 2 – História: Manter uma pergunta	20
Quadro 3 – História: Buscar perguntas	21
Quadro 4 – História: Curtir uma pergunta	21
Quadro 5 – História: Manter uma resposta	22
Quadro 6 – História: Curtir uma resposta	22
Quadro 7 – Regras de Negócio	23
Quadro 8 – Usuário	27
Quadro 9 – Usuário_Pergunta	27
Quadro 10 – Usuário_Resposta	28
Quadro 11 – Usuário_Título	28
Quadro 12 – Título	28
Quadro 13 – Resposta	29
Quadro 14 – Tag	29
Quadro 15 – Pergunta	30
Quadro 16 – Tag_Pergunta	30
Quadro 17 – Tag_Evento	30
Quadro 18 – Evento	31
Quadro 19 – Categoria	31

Listas de abreviaturas e siglas

API	<i>Application Programming Interface</i> - Interface de programação de aplicações - Citado em 11 , 36 , 57
CSS	<i>Cascading Style Sheets</i> - Folha de Estilo em Cascatas - Citado em 10
DD	Dicionário de Dados - Citado em 27
DER	Diagrama Entidade Relacionamento - Citado em 25
DTR	Diagrama de Tabelas Relacionais - Citado em 26
EAD	Ensino a Distância - Citado em 56
HTML	<i>HyperText Markup Language</i> - Linguagem de Marcação de Hipertexto - Citado em 10
HTTP	<i>Hypertext Transfer Protocol</i> - Protocolo de Transferência de Hipertexto - Citado em 11
IDE	<i>Integrated Development Environment</i> - Ambiente de Desenvolvimento Integrado - Citado em 12
IFSP	Instituto Federal de Educação, Ciência e Tecnologia de São Paulo - Citado em 2 , 8 , 46 , 47 , 48 , 49 , 50 , 52 , 53 , 55
MVC	<i>Model-View-Controller</i> - Modelo-Visão-Controle - Citado em 10
PDS	Prática de Desenvolvimento de Sistemas - Citado em 8 , 17 , 46 , 50 , 51
PJI	Projeto Integrador - Citado em 57
POC	<i>Proof of Concept</i> - Prova de Conceito - Citado em 2 , 20 , 32 , 35 , 36 , 48 , 49 , 55 , 56 , 57
Q/A	<i>Questions and Answers</i> - Perguntas e Respostas - Citado em 50
RN	Regra de Negócio - Citado em 23
SGBD	<i>Data Base Management System</i> - Sistema de Gerenciamento de Banco de Dados - Citado em 11
SQL	<i>Structured Query Language</i> - Linguagem de Consulta Estruturada - Citado em 11
TDS	Teoria e Desenvolvimento de Sistemas - Citado em 50
UML	<i>Unified Modeling Language</i> - Linguagem de Modelagem Unificada - Citado em 23

Sumário

1	INTRODUÇÃO	8
1.1	Objetivo	8
1.2	Justificativa	9
2	MATERIAIS E MÉTODOS	10
2.1	Tecnologias e ferramentas de desenvolvimento	10
2.1.1	Ferramentas de apoio	11
2.2	Métodos de gestão e desenvolvimento	12
2.2.1	Metodologia ágil: Scrum	12
2.2.1.1	O quadro de kanban	13
2.2.2	Gestão da equipe	14
2.2.3	Desenvolvimento: artefatos e eventos	17
3	DESENVOLVIMENTO	19
3.1	Análise de Requisitos	19
3.1.1	Histórias de Usuário	20
3.1.2	Regras de Negócio	22
3.2	Modelagem	23
3.2.1	Diagrama de Casos de Uso	23
3.2.2	Diagrama de Entidade e Relacionamento	24
3.2.3	Diagrama de Tabelas Relacionais	26
3.2.4	Dicionário de Dados	26
3.3	Prototipagem	31
3.3.1	Protótipos de alta fidelidade	32
4	PROVA DE CONCEITO	36
	REFERÊNCIAS	38
	GLOSSÁRIO	39
	APÊNDICES	43
	APÊNDICE A – PERGUNTAS DA PESQUISA DE VIABILIDADE	44

	APÊNDICE B – ATAS DAS REUNIÕES	45
B.1	1ºbimestre	45
B.1.1	Planejamento - 15/03/2022	45
B.1.2	Planejamento/Alinhamento - 17/03/2022	45
B.1.3	Planejamento/Alinhamento - 18/03/2022	45
B.1.4	Alinhamento - 21/03/2022	46
B.1.5	Alinhamento - 25/03/2022	46
B.1.6	Planejamento/Alinhamento - 28/03/2022	46
B.1.7	Planejamento/Retrospectiva - 02/04/2022	46
B.1.8	Planejamento - 04/04/2022	47
B.1.9	Alinhamento - 09/04/2022	47
B.1.10	Alinhamento - 10/04/2022	47
B.1.11	Retrospectiva - 12/04/2022	47
B.1.12	Planejamento/Alinhamento - 17/04/2022	48
B.1.13	Planejamento - 18/04/2022	48
B.1.14	Alinhamento - 21/04/2022	48
B.1.15	Alinhamento - 25/04/2022	48
B.1.16	Alinhamento - 01/05/2022	49
B.1.17	Alinhamento - 03/05/2022	49
B.1.18	Retrospectiva - 16/05/2022	49
	APÊNDICE C – PUBLICAÇÕES NO BLOG DA EQUIPE	50
C.1	1ª Semana - 14/03 à 20/03	50
C.2	2ª Semana - 21/03 à 27/03	51
C.3	3ª Semana - 28/03 à 03/04	51
C.4	4ª Semana - 04/04 à 10/04	52
C.5	5ª Semana - 11/04 à 17/04	53
C.6	6ª Semana - 18/04 à 24/04	54
C.7	7ª Semana - 25/04 à 01/05	55
C.8	8ª Semana - 02/05 à 08/05	55
C.9	9ª Semana - 09/05 à 15/05	56
C.10	10ª Semana - 16/05 à 22/05	57

1 Introdução

O presente documento é resultado da proposta de um projeto cujo objetivo é sustentado no planejamento e na execução de um sistema para a Web através do aprendizado obtido nas matérias técnicas do Curso Técnico de Informática, realizado no IFSP, e como forma de trabalho de conclusão de curso.

Tendo isto em vista e diante dos desafios presentes na lista de requisitos e orientações propostas para a iniciação deste projeto, os integrantes da equipe Bunka Bytes reuniram-se em busca de encontrar a solução que melhor se encaixasse em seus objetivos e nos da disciplina de [PDS](#).

É pensando em soluções viáveis que a equipe voltou seu olhar para sistemas que contribuem para a criação de comunidades colaborativas na área de desenvolvimento de sistemas, que, segundo [Rosa \(2008\)](#), foi fortemente difundida pelas comunidades de Código Aberto (do inglês, *Open Source*), sendo primeiramente criada pela cultura *hacker*, na qual afirma que a paixão e o interesse dos *hackers* nas soluções foi uma das principais propulsoras do espírito colaborativo.

Além deles, foi trazido ao debate as possibilidades de aliar as principais dificuldades que os integrantes observaram durante sua vida acadêmica no [IFSP](#), a um sistema que pudesse suprir determinadas necessidades dos alunos, como os questionamentos que começam a surgir com mais frequência conforme o início dos estudos é dado, sendo eles em âmbitos diversos como: sobre a instituição de ensino, matérias e assuntos tratados no ensino médio, dúvidas sobre os conteúdos técnicos ou até mesmo a busca por um apoio educacional - como ocorrem nas monitoriais.

O [IFriends](#) surge nesse cenário, no qual a criação de uma comunidade de estudantes que colaborassem entre si, pudesse instigar o interesse dos alunos em ajudarem uns aos outros de maneira acessível e prática, onde uma dúvida estivesse a um palmo de distância.

1.1 Objetivo

O objetivo deste projeto é tentar instigar o interesse dos estudantes que comprehendem o [IFSP](#) para poderem criar espaços colaborativos entre si, por um sistema onde os usuários interajam entre perguntas e respostas, fornecendo caminhos para o esclarecimento de suas dúvidas sobre a instituição de ensino, as áreas e disciplinas que a ela pertence.

Dessa forma, o objetivo será aplicado através da construção de uma plataforma de perguntas, respostas e mentorias para a Web, em que qualquer usuário poderá sub-

meter uma pergunta para ser respondida pelos outros membros da comunidade; além de possibilitar que estudantes possam escolher se tornar mentores sobre determinados assuntos, disponibilizando recursos para a criação de anúncios de eventos de monitorias (cuja localidade a eles deve competir) dentro de seus perfis de usuário.

Tendo isso em vista e pensando numa melhor interatividade entre os usuários, o sistema deve passar por um processo de [gamificação](#) em algumas de suas funcionalidades, como as votações para respostas e perguntas mais relevantes e os atributos dos usuários mais ativos - assim como outros exemplos que devem ser adicionados durante o planejamento do projeto.

1.2 Justificativa

A reflexão com relação às formas complementares de aprendizagem é importante para a ampliação dos conteúdos interessados tanto aos alunos, quanto aos seus professores, pois permite que enxerguem, juntos, o ensino como um meio que evite a passagem de aprendizados de forma restrita e hierarquizada.

Por isso, [Fernandes \(2011\)](#) traz em sua pesquisa que o desafio da construção de sociedades de aprendizagem parte do pressuposto de que os recursos tecnológicos disponibilizados atualmente permitem aos estudantes aprenderem dentro e fora da escola e das mais variadas formas. Assim, para ele, a melhor forma se dá construindo comunidades sustentadas pelo uso de tecnologias Web.

O autor dá continuidade na exposição desse fenômeno ao atribuir o sucesso da potencialização da aprendizagem complementar e das relações sociais à “Web 2.0”. Isto, pois, de acordo [Fernandes \(2011\)](#), a mesma permitiu novas formas e possibilidades de criação de conteúdos e possibilitou o enfoque a uma aprendizagem motivada pelos interesses do aluno, em que ele deve assumir um papel exploratório nessa experiência, da qual poderá colher ensinamentos significativos, explica [Fernandes \(2011\)](#).

Visando atrair atenção para o tema, o projeto tem como principal missão, permitir que os estudantes possam usufruir de uma ferramenta gratuita que proporcione a suavização do seu processo de aprendizagem, quando seus próprios colegas contribuirão com suas experiências passadas, além de deixarem um histórico para possibilitar um caminho menos árduo aos estudantes que virão. Por isso, espera-se que, com este projeto, a instituição de ensino também seja um agente na construção de uma comunidade propícia para estudantes, onde poderão unir-se em razão de dúvidas comuns, e assim incentivarem a disseminação de uma cultura colaborativa dentro de seus espaços.

2 Materiais e Métodos

Segundo Júnior (2010), engenharia de software é “um conjunto integrado de métodos e ferramentas utilizadas para especificar, projetar, implementar e manter um sistema”, e que, para tanto, reúne em si metodologias, métodos e ferramentas para que o projeto seja bem definido em todas as suas etapas, variando desde sua problemática inicial e indo até sua entrega enquanto um produto de software. É a partir disto, que Júnior (2010) distingue um método de uma ferramenta:

Um método é uma prescrição explícita de como chegar a uma atividade requerida por um modelo de ciclo de vida, visando otimizar a execução das atividades que foram especificadas. Já as ferramentas proporcionam apoio automatizado ou semi-automatizado aos métodos (JÚNIOR, 2010).

Ademais, Júnior (2010) ainda separa os métodos de desenvolvimento de sistema em três categoriais que permitem visualizar e solucionar o problema de diferentes maneiras para sua modelagem. Para a composição do presente projeto, no entanto, será utilizado como método principal a metodologia escolhida para a gestão do projeto e as ferramentas poderão ou não ser relacionadas a mesma, como há de ser especificado nas próximas seções.

2.1 Tecnologias e ferramentas de desenvolvimento

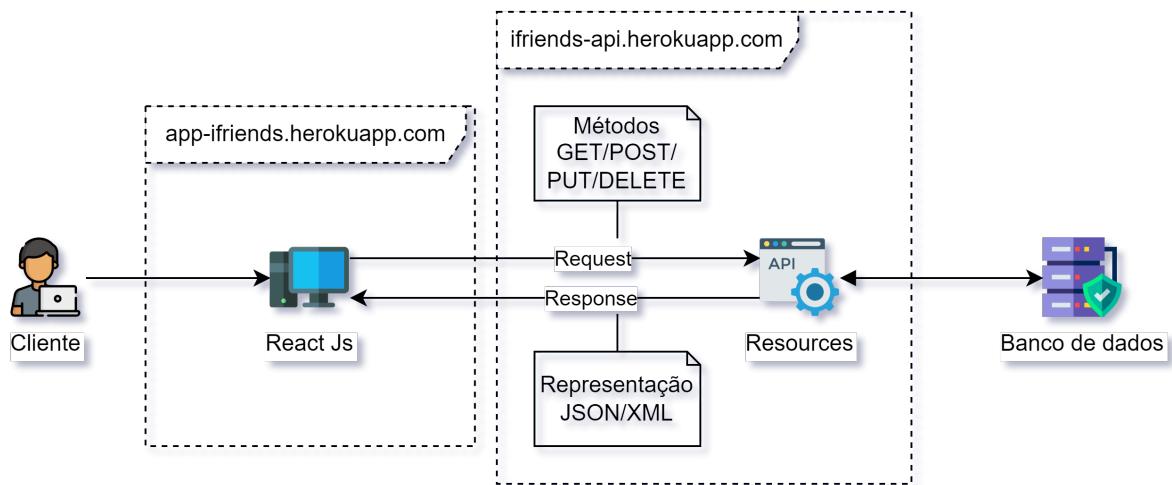
Pensando no objetivo e alcance do projeto, decidimos por realizar uma aplicação Web inicialmente, a qual pode ser acessada de diferentes dispositivos de forma fácil, apenas sendo necessário o acesso à *internet*. Outros fatores considerados são a experiência da equipe com o mesmo e a disponibilidade de conteúdo sobre desenvolvimento Web.

De modo a desenvolver essa aplicação, será usado **HTML**, **CSS** e **JavaScript** como base para o **Front-end**, já os **Frameworks** que servirão de auxílio serão o **React-Bootstrap** e o **Ant**.

Para o **Back-end** foi considerada a linguagem Java, orientada a objetos; sua escolha foi feita pelo fato da equipe já possuir experiência com a linguagem, além de alguns já trabalharem com a mesma, ela também possui uma boa maturidade, contendo muitas informações disponíveis para ajudar no desenvolvimento. Em conjunto, será utilizado **Spring Boot**: **Framework** Java de código aberto, e será juntamente usado o **Maven**: **Framework** de automação e gerenciamento de dependências no projeto. Já como padrão de desenvolvimento utilizado no **Back-end**, será aplicado o padrão **MVC**, que separa o projeto em três principais camadas, *model*, *view* e *controller*.

No entanto, como arquitetura de sistema, foi escolhido utilizar a arquitetura **REST API** na qual é possível separar o código *Back-end* do *Front-end* de maneira que as duas aplicações possam determinar a troca informações entre elas com requisições via protocolo **HTTP**. Esta solução também é útil para escalar a plataforma, já que a utilização de uma **API** pode ser feita tanto por aplicações móveis, como para a Web.

Figura 1 – Arquitetura Rest API



Fonte: Os autores.

Quanto ao banco de dados, será utilizado o **PostgreSQL**: um **SGBD** popular de **SQL**, que serve para executar comandos no banco, como consultas, e alterações nos dados ou na estrutura.

Para hospedar a aplicação, será utilizado o **Heroku**, o qual é uma plataforma de nuvem que suporta diversas linguagens, que permite a implantação, escalonamento e gerenciamento do sistema.

Como inspiração para o sistema, será estudada sobre a plataforma *Scoold*, a qual é uma aplicação Web de código aberto de perguntas em respostas, inspirada no *Stack Overflow*. Ela usa de base as mesmas tecnologias do projeto, tendo isso em vista, no caso da inviabilidade de seu uso, pode ser reaproveitado ideias implementadas nessa plataforma no seu código-fonte.

2.1.1 Ferramentas de apoio

Nesta seção serão especificados as ferramentas a serem utilizadas para a estruturação do sistema Web.

- **Notion:** Foi escolhido como plataforma de organização de projetos, pois este possui métodos de gerenciamento de equipe, fornecendo uma interface para vários desen-

volvedores trabalharem utilizando o quadro de kanban, assim como o compartilhamento de arquivos, vídeos e artigos; permitindo também notificações via e-mail sobre atualizações do projeto e de reuniões marcadas, por exemplo.

- **Discord:** Plataforma de comunicação online escolhida pela equipe, por ser a mais utilizada e conhecida entre os mesmos. Nela é possível criar servidores privados para compartilhar informações e realizar reuniões sobre o projeto.
- **Overleaf:** Para a preparação do documento de visão, será usado o L^AT_EX, visto que este possui comando de padronização para documentos acadêmicos, facilitando a sua construção, juntamente, com editor Overleaf que oferece um ambiente compartilhado entre os membros da equipe.
- **Figma:** Na parte de modelagem do sistema e elaboração ideias de soluções de problemas, escolheu-se o Figma, visto que é uma ferramenta gratuita que possui diversas opções de edição.
- **Visual Studio Code:** A IDE escolhida para realizar a edição de código-fonte na parte *Front-end* do sistema, por razões de ser um recurso da atualidade com possibilidades de várias adaptações de ambiente, principalmente as nossas tecnologias escolhidas.
- **Eclipse:** A IDE escolhida para realizar a edição de código-fonte na parte *Back-end*, por motivo de facilidade que o ambiente de desenvolvimento proporciona para rodar aplicações com Java e, também, seus recursos que fazem a integração com o banco de dados.

2.2 Métodos de gestão e desenvolvimento

Tendo em vista a organização do projeto **IFriends**, a equipe escolheu utilizar os princípios e valores do Manifesto Ágil como norteadores nos seus processos de planejamento, modelagem e desenvolvimento do projeto. Deste modo, elencou-se o **Framework Scrum** como metodologia ágil que irá servir como referência para o trabalho da Bunka Bytes.

2.2.1 Metodologia ágil: Scrum

Segundo Ambler (2004), o termo metodologias ágeis surgiu em fevereiro de 2001, quando especialistas em processos de engenharia de software se reuniram e estabeleceram princípios comuns entre todas as metodologias, criando a Aliança Ágil e o estabelecimento do “Manifesto Ágil” (*Agile Manifesto*).

O termo metodologia ágil consiste na otimização do tempo para a realização de determinado projeto, visando a rapidez na entrega e na qualidade do mesmo, surgindo assim, como uma resposta mais leve, mais assertiva e menos custosa em relação aos métodos pesados que eram utilizados para a construção de sistemas. Nas metodologias ágeis, os processos, as ferramentas, as documentações, as negociações e os planejamentos, possuem prioridade secundária, pois os indivíduos e suas interações são considerados essências e indispensáveis (SGANDERLA, 2016).

Para isso, o manifesto determina quatro valores principais, sendo eles: o enfoque nos indivíduos e nas interações, e não nos processos ou algoritmos; a adaptação e maior flexibilidade a novos fatores decorrentes do desenvolvimento do projeto; o foco na funcionalidade do sistema e na documentação mais simples e objetiva, e por último, a preferência por um ambiente de trabalho mais colaborativo e menos burocrático (SGANDERLA, 2016).

Segundo Cruz (2018), o Scrum pode ser definido como “um *Framework* para desenvolver e manter produtos complexos que também pode ser utilizado no gerenciamento ágil de projetos que se destinam também à criação de produtos”. Neste caso, sabida a escolha da utilização de uma metodologia ágil para o gerenciamento do presente projeto, foi decidido aplicá-la com base no Scrum e suas especificações, porém, como a finalidade do trabalho é acadêmica, resolveu-se adaptar algumas de suas características para que a metodologia pudesse ser implementada como uma referência na organização e gestão do projeto. Estas, por conseguinte, serão explicitadas mais a frente, conforme apresentadas as particularidades do *Framework*.

Dentre as principais características do Scrum, está a divisão do desenvolvimento em ciclos repetitivos e curtos, permitindo modificações, adaptações e correções no produto de forma iterativa e incremental, o que, segundo Cruz (2018) permite encontrar desvios mais rápido e com menos impacto. Cruz (2018) explica que para que os processos sejam otimizados de tal forma, o Scrum possui ainda três pilares de sustentação: transparência, a inspeção e a adaptação.

Além dos pilares da metodologia, há cinco valores importantes para sua construção e prática durante um projeto: coragem, foco, comprometimento, respeito e abertura. De acordo com Cruz (2018) os valores do Scrum são responsáveis por reforçar os princípios do manifesto ágil, principalmente considerando o comportamento e as pessoas maiores do que os processos e ferramentas.

2.2.1.1 O quadro de kanban

Segundo Peinado e Aguiar (2007), o nome Kanban vem do japonês “cartão” e a sua origem deu-se pela seguinte razão:

Este nome surgiu em razão do sistema de controle visual dos estoques de materiais, pois são frequentemente utilizados cartões para representar os contentores cheios ou vazios, estes cartões são retirados ou colocados em um quadro à medida que o material é utilizado, ou reposto (PEINADO; AGUIAR, 2007).

E através da implantação realizada por Silva, Santos e Neto (2012), devido aos problemas e necessidades encontrados ao longo dos *sprints*, houve a criação de um processo ágil, baseado nas práticas do Scrum com características do Kanban:

As tarefas ou itens de trabalho foram representadas através de cartões (do inglês *post-its*) fixados em um quadro (do inglês *cardwall*). Esse quadro, por sua vez, era dividido em colunas que representavam as fases do fluxo de trabalho (do inglês *workflow*) da equipe. As tarefas eram distribuídas sequencialmente nas colunas à medida que avançavam no fluxo de trabalho (SILVA, 2012).

No projeto a implementação do kanban foi similar, porém todo o esquema presencial foi adaptado ao quadro remoto. Para conciliar a participação continua da equipe, a ferramenta *Notion* é utilizada como auxiliar aos processos de organização, onde também foi possível criar o quadro de kanban do projeto.

O quadro de kanban usado pela equipe Bunka Bytes, responsável pelo projeto apresenta 5 colunas, nas quais são representadas visualmente em qual estágio a tarefa se encontra, as colunas se classificam como “para fazer”, “planejamento”, “em andamento”, “para revisar” e “feito”. E por “cartões virtuais” são colocadas as tarefas no quadro.

Foi decidido utilizar uma classificação em ordem de prioridade para cada tarefa, podendo ser “alta”, representada pela cor vermelha, “média”, representada pela cor laranja, e “baixa” representada pela cor verde, tais prioridades são atribuídas às tarefas assim que a *sprint* é definida.

De maneira geral, o uso do quadro de kanban é benéfico ao projeto, já que através de sua aplicação a equipe consegue conciliar de maneira clara e precisa as tarefas compartilhadas. A utilização do quadro remotamente, faz com que a equipe possa ainda acessá-lo de qualquer lugar, facilitando o processo de transparência dos itens e faz com que todos tenham em suas mãos as atividades a serem feitas, podendo ajustá-las no momento em que preferirem.

2.2.2 Gestão da equipe

A equipe Scrum é composta por três papéis: o *Scrum Master*, o *Product Owner* e o Time de Desenvolvimento:

O primeiro, chamado *Scrum Master*, é considerado o responsável por garantir que o Scrum seja entendido e aplicado, para que o Time Scrum

esteja aderindo os valores, as práticas e as regras do Scrum, e, portanto, trabalha como um líder ou técnico da equipe. Já o *Product Owner*, é o principal responsável pelo gerenciamento do *backlog* do produto, por garantir o valor do trabalho realizado pelo Time, e pela satisfação e atendimento das necessidades do cliente. O Time de Desenvolvimento, por outro lado, é responsável por executar o desenvolvimento e transformar o *backlog* do produto em incrementos de funcionalidades, criando um sistema pronto que possa ser entregue ao cliente (CRUZ, 2018).

Portanto, para fins de organização, o Time de Desenvolvimento poderá ser dividido em subequipes, sendo elas: desenvolvimento (*Front-end* e *Back-end*), design e documentação, e mídias. Nelas, estão inclusas as funções de programadores *Front-end* e *Back-end*, divididas entre os integrantes Jamilli Vitória Gioielli, José Roberto Claudinho Ferreira e Kaiky Matsumoto Silva; sendo a Jamilli responsável por supervisionar o *Front-end* da aplicação, e o José (que também auxiliará no *Front-end*) juntamente com o Kaiky pelo *Back-end* - que também inclui a administração do banco de dados.

Além dessas, as outras duas subequipes foram divididas entre as integrantes Anaí Villca Rojas, responsável por supervisionar o design de experiência/interface de usuário; e Julia Romualdo Pereira, responsável pelo gerenciamento da documentação e das mídias do projeto (como o canal no [YouTube](#) e as postagens no Blog).

De todo modo, a documentação e as mídias do projeto deverão ser revisadas, obrigatoriamente, por todo o time, de modo a manter um conhecimento sobre a aplicação melhormente distribuída. Entretanto, precisa-se compreender que a modelagem de dados e a elicitação de requisitos da aplicação serão feitas com o auxílio de todos os integrantes, visto que são partes de extrema importância para que a compreensão de todos sobre o cenário corresponda com o objetivo principal pretendido. No [Quadro 1](#), é possível observar uma relação das supervisões dos integrantes em cada área do projeto.

Quadro 1 – Distribuição de tarefas

Responsável	Front-end	Back-end	Documentação e Mídias	Design
Anaí			●	●
Jamilli	●			●
José	●	●		
Julia			●	
Kaiky		●		

Fonte: Os autores.

Entretanto, é importante salientar que as funções atribuídas a cada um não são exclusivas e podem variar conforme a necessidade de entrega do projeto. As subequipes são de responsabilidade de todos os integrantes e apenas foram divididas assim para fins de organização das tarefas do projeto. Levando isso em consideração, as funções de *Scrum Master* ou *Iteration Manager* e *Product Owner* foram adaptadas, ainda que não seja

o ideal na metodologia ágil, para um único papel, representado pela integrante Jamilli Vitória Gioielli. Porém, vale ressaltar ainda que, nesse modelo escolhido, toda a equipe é responsável pela inspeção dos processos do projeto, e não somente o *Scrum Master ou Iteration Manager*:

O Time deve ser multidisciplinar e multifuncional, possuindo todo o conhecimento necessário para criar um incremento no trabalho. [...] Não há títulos no Time, e não há exceção a esta regra. Não deve existir distinção de cargos ou funções, títulos ou senioridades, e muito menos áreas determinadas ou específicas de atuação. No Scrum todos os integrantes do Time são conhecidos como desenvolvedores.

Individualmente os integrantes do Time de Desenvolvimento podem ter habilidades específicas, mas, independentemente disso, a responsabilidade a respeito de uma entrega continua sendo do Time de Desenvolvimento como um todo (CRUZ, 2018).

Para organizar a produtividade continua de desenvolvimento do projeto, buscando seguir uma das vertentes propostas na metodologia ágil de entregas semanais, a equipe criou um cronograma de entregas por aula baseado a partir do plano de ensino da disciplina disponibilizado pelos orientadores, ou seja, a equipe busca realizar durante a semana a atividade proposta para a aula seguinte para aproveitar o tempo em sala de aula validando o que foi realizado com os orientadores.

O cronograma de entrega semanal utilizado para o desenvolvimento do projeto IFriends pode ser observado na [Figura 2](#).

Figura 2 – Cronograma de Desenvolvimento Semanal

A Conteúdo Planejado	Data / Semana	<input checked="" type="checkbox"/> Post no Bloq?	# Qtd de Reuniões	Resumo das entregas/atividades
Apresentação e definição dos procedimentos durante o ano letivo. Divisão das equipes.	14/03/2022	<input checked="" type="checkbox"/>	3	- Brainstorm e conversas iniciais sobre as ideias
Análise de temas de trabalhos	21/03/2022	<input checked="" type="checkbox"/>	3	- Acesso.txt - Documento sobre temas
Estudo de viabilidade	28/03/2022	<input checked="" type="checkbox"/>	2	- Inscrição do blog no feedburner - Criação da pesquisa de viabilidade
Análise de tecnologias	02/04/2022	<input checked="" type="checkbox"/>	2	- Resumo das tecnologias a serem utilizadas
Proposição de temas de trabalho	04/04/2022	<input checked="" type="checkbox"/>	3	- Divulgação da pesquisa - Análise dos trabalhos anteriores - Criação do documento para proposta
Apresentação das propostas iniciais	11/04/2022	<input checked="" type="checkbox"/>	2	- Apresentação da proposta - Envio do documento da proposta e da análise dos outros projetos - Envio dos Slides
Técnicas de gerenciamento de projeto	18/04/2022	<input checked="" type="checkbox"/>	2	- Entrega da análise sobre as outras equipes - Entrega do vídeo da proposta - Apresentação do rascunho do backlog do produto
Requisitos de negócio	25/04/2022	<input checked="" type="checkbox"/>	2	- Dúvidas sobre as histórias de usuário - Revisão da modelagem e das regras de negócio - Início da prototipagem
Modelagem de dados	02/05/2022	<input checked="" type="checkbox"/>	1	- Mostramos o protótipo e o fluxo do sistema - Mostramos a versão do banco de dados para a PoC

Fonte: Os autores.

2.2.3 Desenvolvimento: artefatos e eventos

Tendo em vista a finalidade acadêmica do projeto, foi estipulado um valor mínimo de duas reuniões semanais dedicadas na construção do mesmo, levando em conta a organização individual de cada membro da equipe, além da priorização para que a tomada de decisões e planejamentos sejam feitas no dia das aulas da disciplina de [PDS](#). Sobre os artefatos e eventos do Scrum, lista-se a seguir suas funcionalidades originais, descritos por [Cruz \(2018\)](#), e como serão adaptados:

- *Backlog*: é descrito como uma lista de todas as características, funções, tecnologias, melhorias e correções que constituem o produto a ser entregue. É também subdividido em “do produto”, “da versão de entrega” e “da Sprint”;
- *Sprint*: originalmente pensada para durar um mês ou menos e possuir uma meta estabelecida com um objetivo claro, foi pensada para durar até duas semanas, já que as orientações da disciplina exigem que existam entregas frequentes, isto é, todas as semanas. Portanto, o ideal é que se dê início ao trabalho a ser entregue pelo menos uma semana antes de sua entrega;
- *Time-boxed*: é esperado que o tempo estipulado para executar uma tarefa seja cumprido e que o trabalho proposto, seja realizado. Tendo em vista os curtos prazos, também poderá ser adaptado conforme a entrega;
- *Planejamento da Sprint*: ainda será utilizado para definir “o que será feito” e “como”, porém, ao contrário das oito horas mínimas estipuladas, a equipe definiu um mínimo de duas horas para tal reunião, que poderá acontecer nas segundas-feiras, durante a aula de [PDS](#) ou aos finais de semana, logo após a finalização dos entregáveis;
- *Reuniões Diárias*: não serão mantidas de maneira síncrona, visto que cada membro da equipe possui horários de disponibilidade diferentes. Porém, poderão ocorrer de forma assíncrona, de modo a compreender se existem impedimentos durante a execução das tarefas planejadas para a semana;
- *Revisão da Sprint*: seu maior objetivo é a revisão do *Product Owner*, ou do cliente, em todos os itens concluídos pelo Time. Porém, não terá *Time-boxed* de quatro horas, visto que a aprovação dos professores orientadores (o cliente) pode ser mais rápida e acontecer durante a execução das tarefas. No entanto, o Time a realizará em todo final de entrega, para que os retornos dos professores levem em consideração o trabalho final;
- *Retrospectiva da Sprint*: possui originalmente *Time-boxed* de até três horas e é feita para identificação de medidas de melhoria no processo do time que serão aplicadas na próxima *Sprint*. Foi escolhido adaptá-la para acontecer às terças-feiras, antes do

início das aulas, para que as entregas feitas às segundas-feiras estejam frescas e o processo realizado pelo time possa ser melhor avaliado para a melhoria contínua.

Para a definição do *backlog* do produto, o Scrum conta com um recurso chamado histórias de usuário, que pode ser definido como:

História é uma descrição resumida, porém clara e objetiva, de alguma funcionalidade que deverá ser fornecida pelo produto a ser entregue, sempre do ponto de vista do usuário final. Uma história não é uma especificação completa da funcionalidade, mas uma promessa de discutir uma funcionalidade, ou, simplesmente, um lembrete de que a discussão já aconteceu. Um modelo simples de como escrever uma história seria: Como um <tipo de usuário>, eu quero <um objetivo> para que <atenda a uma necessidade>. ([CRUZ, 2018](#))

Entretanto, vale ressaltar que as histórias de usuário do presente projeto serão montadas somente na parte de desenvolvimento, isto é, quando o produto começar a ser construído pelo Time Scrum e após os principais itens de *backlog* estarem descritos. Como ainda se encontra na fase de planejamento e modelagem, tal recurso ainda não foi acionado no projeto.

Outra ressalva diz respeito a utilização do *Scrum Poker* ou *Planning Poker Card*, que, segundo [Cruz \(2018\)](#), é definido como “uma técnica que auxilia na estimativa de histórias e tarefas com base no consenso de todo o Time”. Para tanto, o Time utiliza um conjunto de cartas com valores representando os pontos ou horas por história. Sobre o funcionamento do *Planning Poker Card*:

O seu uso é simples: o *Product Owner* ou um membro do Time apresenta a história, ou tarefa. Após uma breve discussão, cada um escolhe uma carta e a coloca virada sobre uma mesa, de forma que um não constate o valor da carta que o outro escolheu. Quando todos colocarem suas cartas, elas são desviradas para que todos vejam os valores.

Caso não haja consenso entre as cartas escolhidas, as diferenças são discutidas brevemente e uma nova rodada acontece, até que haja convergência e consenso. ([CRUZ, 2018](#))

Para o presente projeto, no entanto, o *Planning Poker Card* será utilizado durante o processo de desenvolvimento do produto e após as histórias de usuário da *sprint* estarem bem descritas. Vale ressaltar que este processo é apenas para estimar o trabalho da equipe e pode variar conforme o andamento do projeto.

3 Desenvolvimento

Baseando-se na conceituação de engenharia de software dada por Sommerville (2019), neste capítulo é descrito o desenvolvimento do sistema, cujas etapas estão apoiadas em técnicas que vão desde sua especificação até sua evolução. Neste sentido, Sommerville (2019) destaca quatro etapas como fundamentais para os processos de software, sendo elas sequenciadas em: especificação, desenvolvimento, validação e evolução. As etapas correspondentes a este capítulo consistem na definição, junto as suas restrições, e na projeção do sistema para ser programado - isto é, a especificação e o desenvolvimento.

Desse modo, as especificações descritas no capítulo estão separadas em: análise de requisitos, regras do negócio, modelagem do sistema e prototipação.

3.1 Análise de Requisitos

Segundo Machado (2018), ao definir as características de um requisito, é preciso salientar que não são dependentes da tecnologia empregada, visto que suas especificações estão contidas no campo do cumprimento das necessidades do usuário. Dessa forma, Machado (2018) define os requisitos como “objetivos ou restrições estabelecidas por clientes e usuários do sistema que definem suas diversas propriedades”.

Assim, tanto Machado (2018) como Sommerville (2019) concordam que a fase de definição de requisitos, a chamada engenharia de requisitos, é essencial para a tomada de decisões sobre os passos para adquirir ou desenvolver o sistema. Por outro lado, Sommerville ainda acrescenta sobre a necessidade de mudança durante o desenvolvimento:

Naturalmente, são feitas mudanças subsequentes nos requisitos de usuário, que podem ser ampliados para requisitos de sistema mais detalhados. Às vezes, pode-se utilizar uma abordagem ágil para eliciar simultaneamente os requisitos à medida que o sistema é desenvolvido, a fim de acrescentar detalhes e refinar os requisitos de usuário (SOMMERVILLE, 2019).

Tendo tais definições em vista, as próximas seções visam apresentar os requisitos funcionais e não funcionais, e as regras do negócio característicos do projeto tratado neste documento. Para representar os requisitos funcionais e os requisitos não funcionais se usou as histórias de usuário. Foi com base na utilização da abordagem ágil no projeto (mais especificada na seção 3), que se definiu três prioridades principais: alta, média e baixa.

A prioridade alta é aquela correspondente aos requisitos obrigatórios para o funcionamento do sistema, isto é, aqueles que caso faltem, o sistema em si não existe; já

na prioridade média, por exemplo, são característicos aqueles desejáveis, ou seja, são importantes para o sistema, mas não interferem diretamente numa mudança brusca de seu comportamento. Desse modo, os requisitos de prioridade baixa, são tidos como opcionais, aqueles que podem entrar no sistema eventualmente, mas que, numa fila de produção, não serão feitos antes dos demais. Vale ressaltar, por conseguinte, que tais decisões dependem da negociação entre os envolvidos no projeto e do método de produção utilizado, como descrevem [Simões e Vazquez \(2016\)](#). Os autores ainda completam, dizendo:

A priorização tem como função assegurar que os recursos do projeto sejam focados nos itens mais relevantes. Daí a importância de, na especificação, diferenciar cada requisito em termos de importância, dentre dezenas ou centenas de outros requisitos.

A responsabilidade por definir a prioridade do requisito deveria ser da parte interessada, facilitada pelo gerente de projetos ([SIMÕES; VAZQUEZ, 2016](#)).

3.1.1 Histórias de Usuário

Segundo [Cruz \(2018\)](#) as histórias de usuário se caracterizam como uma descrição resumida, clara e objetiva de uma funcionalidade fornecida pelo produto a ser entregue, visando o ponto de vista final do usuário. Ainda segundo o autor para uma história ser tida como completa, ela deve possuir uma descrição objetiva e critérios de aceitação, esses critérios representam o que ela precisa fazer para ser considerada válida.

No projeto, a equipe aproveitou as histórias de usuário para representar os requisitos funcionais e não funcionais, dessa forma os requisitos funcionais podem ser identificados a partir do nome das histórias, e o não funcionais por meio dos critérios de aceitação definidos para tais. Além disso, considerando a entrega da [POC](#), as histórias foram separadas conforme a elaboração dos dois épicos preparados para esta entrega, sendo elas a gestão de perguntas e a gestão de respostas.

Todas as histórias apresentam sete componentes: o seu nome, a sua descrição, seus critérios de aceitação, o épico a qual pertence, a pontuação que ela recebeu no *planning poker*, a estimativa de tamanho conforme a sua pontuação e a sua prioridade conforme o seu tamanho e pontuação.

Quadro 2 – História: Manter uma pergunta

Descrição	Épico	Pontuação	Tamanho	Prioridade
Como aluno, eu gostaria de manter uma pergunta na comunidade para retirar uma dúvida	Gestão de Perguntas	13	Grande	ALTA

Fonte: Os autores

Para esta história de usuário foram definidos os seguintes itens como critérios de aceitação:

- Mostrar “como fazer uma boa pergunta”;
- O usuário deve conseguir somente criar e remover uma pergunta da visualização;
- O usuário deve conseguir fechar o espaço de resposta para a pergunta;

Quadro 3 – História: Buscar perguntas

Descrição	Épico	Pontuação	Tamanho	Prioridade
Como aluno, eu gostaria de buscar perguntas feitas para que possa consultar uma pergunta específica	Gestão de Perguntas	2	Pequena	Média

Fonte: Os autores

Para esta história de usuário foram definidos os seguintes itens como critérios de aceitação:

- O usuário precisa informar total ou parcialmente o título da pergunta desejada;
- As perguntas serão exibidas conforme as informações passadas, podendo ser semelhantes parcial ou totalmente;

Quadro 4 – História: Curtir uma pergunta

Descrição	Épico	Pontuação	Tamanho	Prioridade
Como aluno, eu gostaria de votar em uma pergunta para indicar se ela me foi útil ou não.	Gestão de Perguntas	2	Pequena	Alta

Fonte: Os autores

Para esta história de usuário foram definidos os seguintes itens como critérios de aceitação:

- Um usuário só poderá votar uma única vez;
- Cada voto equivale a um ponto;
- Soma dos pontos por pergunta deve ser exibida;

Quadro 5 – História: Manter uma resposta

Descrição	Épico	Pontuação	Tamanho	Prioridade
Como aluno, eu gostaria de manter uma resposta para retirar uma dúvida de um colega.	Gestão de Respostas	5	Média	Alta

Fonte: Os autores

Para esta história de usuário foram definidos os seguintes itens como critérios de aceitação:

- As respostas mais curtidas devem ser exibidas antes das demais;
- O usuário deve conseguir somente criar e deletar uma resposta;
- Todas as respostas devem ser exibidas sem exceção;

Quadro 6 – História: Curtir uma resposta

Descrição	Épico	Pontuação	Tamanho	Prioridade
Como aluno, eu gostaria de curtir uma resposta para indicar se ela me foi útil ou não.	Gestão de Respostas	1	Pequena	Alta

Fonte: Os autores

Para esta história de usuário foram definidos os seguintes itens como critérios de aceitação:

- Um usuário só poderá curtir uma única vez;
- Cada curtida equivale a um ponto;
- Soma das curtidas por pergunta deve ser exibida;

3.1.2 Regras de Negócio

Regras de negócio são requisitos de domínio de aplicação tratado no desenvolvimento de um sistema, isso significa, as declarações sobre como determinada empresa faz negócio. É a partir dessas regras que se define como o negócio funciona e quais são suas especificações, além da sua importância para o desenvolvimento de um sistema, pois, auxiliam no controle dos processos, ajudam na tomada de decisões além de afetarem diretamente os requisitos funcionais, como descrevem Dallavalle e Cazarini (2000). Dessa forma, o Quadro 7 lista as regras de negócio levantadas para o projeto IFriends.

Quadro 7 – Regras de Negócio

ID	Descrição
RN01	Não serão permitidos usuários com os mesmos dados, ou seja, o sistema só permitirá a criação de uma conta por usuário
RN02	A fim de garantir a segurança de nossos usuários, o sistema deverá fazer uso de algumas diretrizes da Lei Marco Civil da Internet, lei nº12.965/2014, que tem como objetivo estabelecer princípios, garantias, direitos e deveres para o uso da internet no Brasil.
RN03	Dentro do direito a exclusão, ao excluir seu perfil, o usuário deve ter ciência de que suas perguntas e respostas serão mantidas na comunidade, porém como parte de um usuário anônimo (exemplo: user12345678).
RN04	O usuário deve resumir seu problema dentro do título da pergunta.
RN05	O usuário deve descrever seu problema, informar de forma clara o que ele precisa.
RN06	O usuário deve descrever qual cenário o colocou nessa situação, o que já tentou e qual seu objetivo final.
RN07	Se necessário, se deve fazer uso de imagens a fim de explicando melhor o problema.
RN08	O usuário deve lembrar que podem haver respostas diferentes, portanto deve manter a mente aberta a novas sugestões.

Fonte: os autores

3.2 Modelagem

Segundo Sommerville (2019), a modelagem de sistemas é definida como “um processo de desenvolvimento de modelos abstratos de um sistema, cada um apresentando uma visão diferente do mesmo”. Para isso, descreve também que são geralmente usadas notações gráficas baseadas nos tipos de diagrama em UML durante o processo de engenharia de requisitos “para ajudar a derivar os requisitos detalhados de um sistema; durante o processo [...] e depois da implementação, para documentar a estrutura e operação do sistema” (SOMMERVILLE, 2019).

3.2.1 Diagrama de Casos de Uso

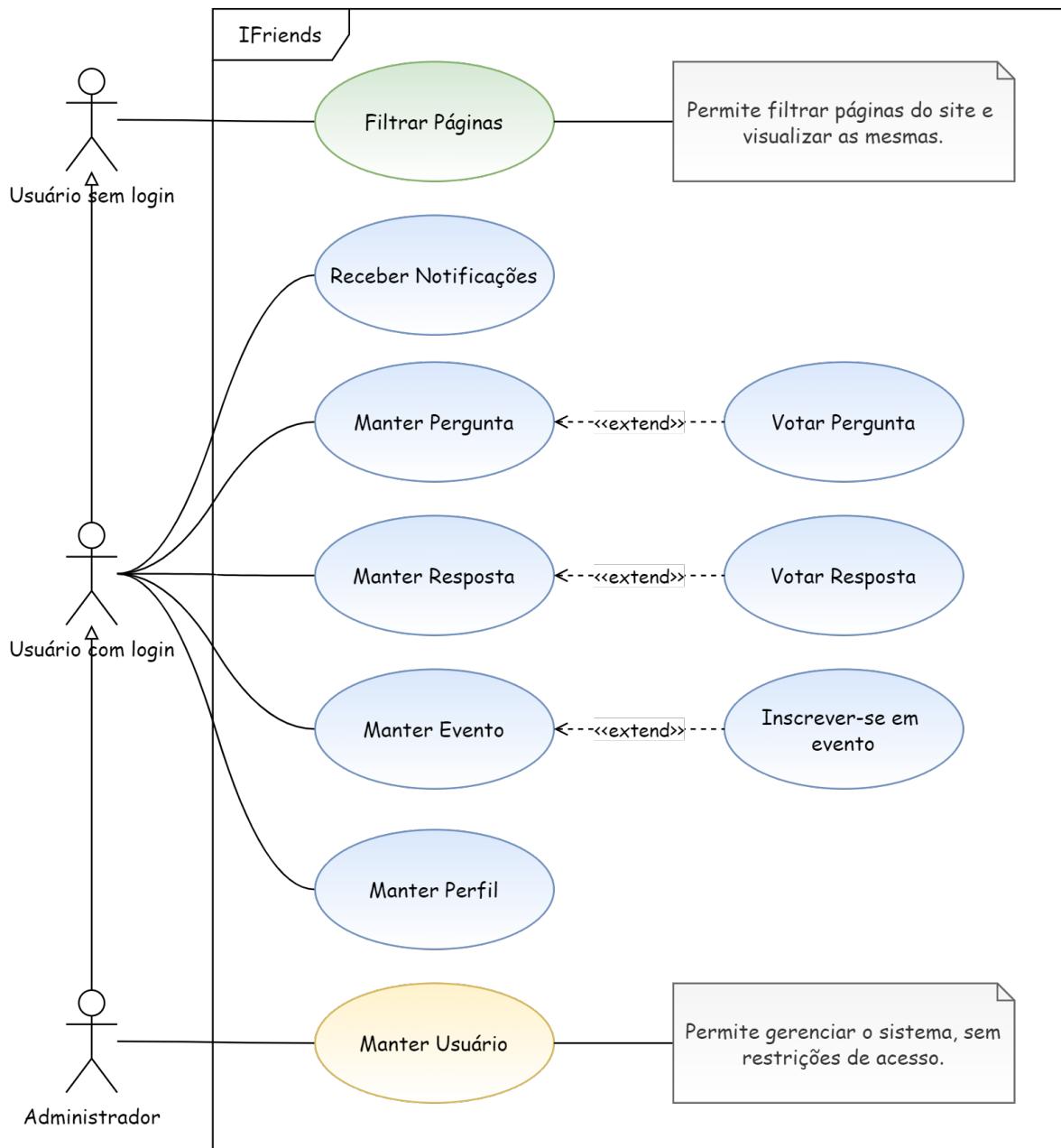
De acordo com Guedes (2011), o diagrama de casos de uso UML é descrito como:

O diagrama de casos de uso [...] tem por objetivo apresentar uma visão externa geral das funcionalidades que o sistema deverá oferecer aos usuários, sem se preocupar com a questão de como tais funcionalidades serão implementadas. [...] É de grande auxílio para a identificação e compreensão dos requisitos do sistema, ajudando a especificar, visualizar e

documentar as características, funções e serviços do sistema desejados pelo usuário (GUEDES, 2011).

Logo, a Figura 3 representa os casos de uso do projeto de sistema IFriends.

Figura 3 – Diagrama de Casos de Uso



Fonte: Os autores.

3.2.2 Diagrama de Entidade e Relacionamento

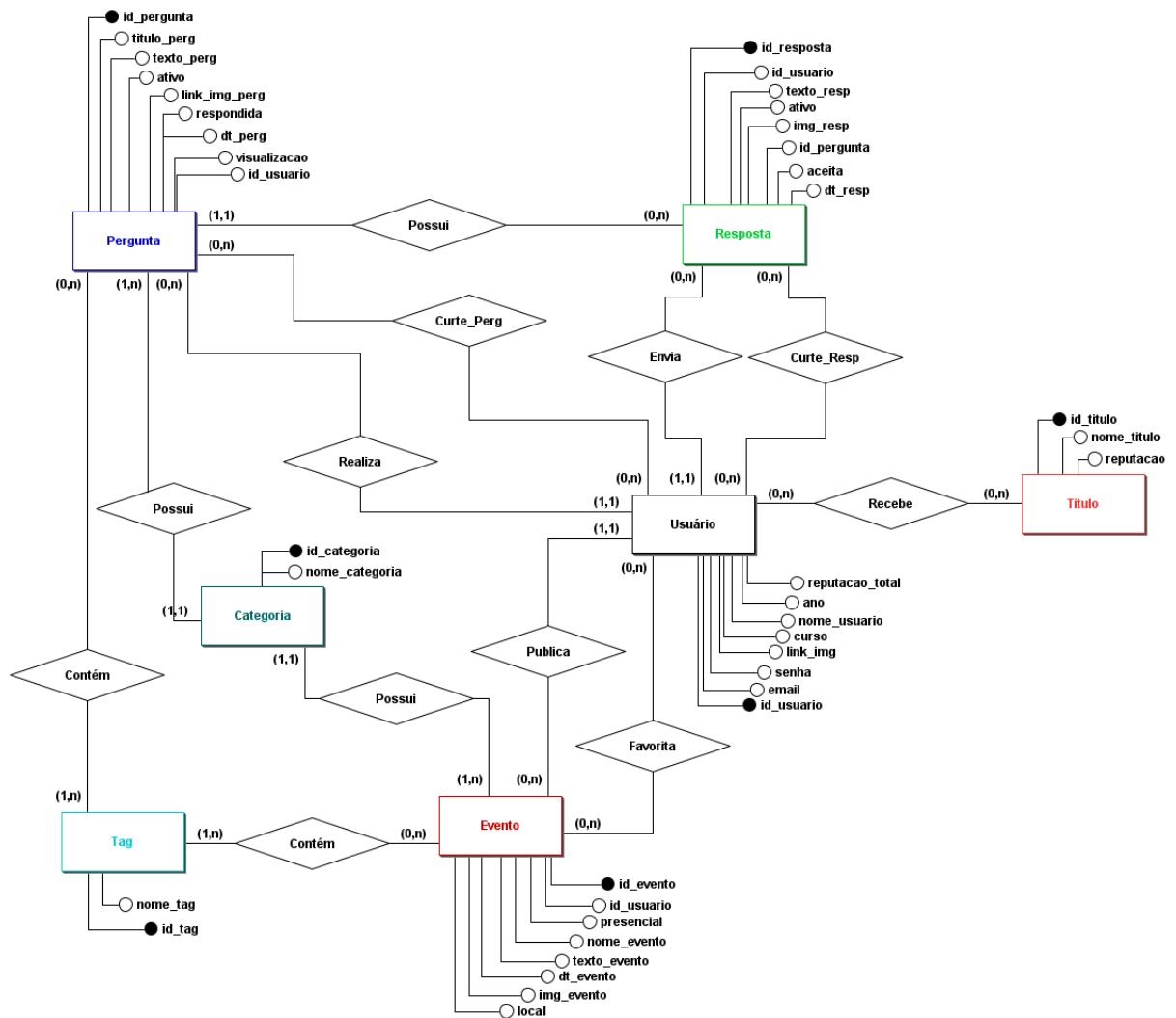
De acordo com Leal (2015), a abordagem entidade-relacionamento é a técnica de modelagem de dados mais difundida e utilizada e representa a modelo conceitual

do banco de dados. Nela, a estrutura do banco de dados é descrita como coleção de entidades, relacionamentos e representada graficamente por meio do Diagrama Entidade Relacionamento.

Através dele, é possível descrever um subconjunto do mundo real que será retratado no banco de dados com um alto nível de abstração. Além disso, o modelo Entidade Relacionamento é um modelo formal e caracteriza-se por ter uma grande capacidade semântica, o que garante que todos possam ter o mesmo entendimento (LEAL, 2015).

A Figura 4 representa o Diagrama Entidade Relacionamento (DER) do projeto de sistema IFriends.

Figura 4 – Diagrama de Entidade e Relacionamento



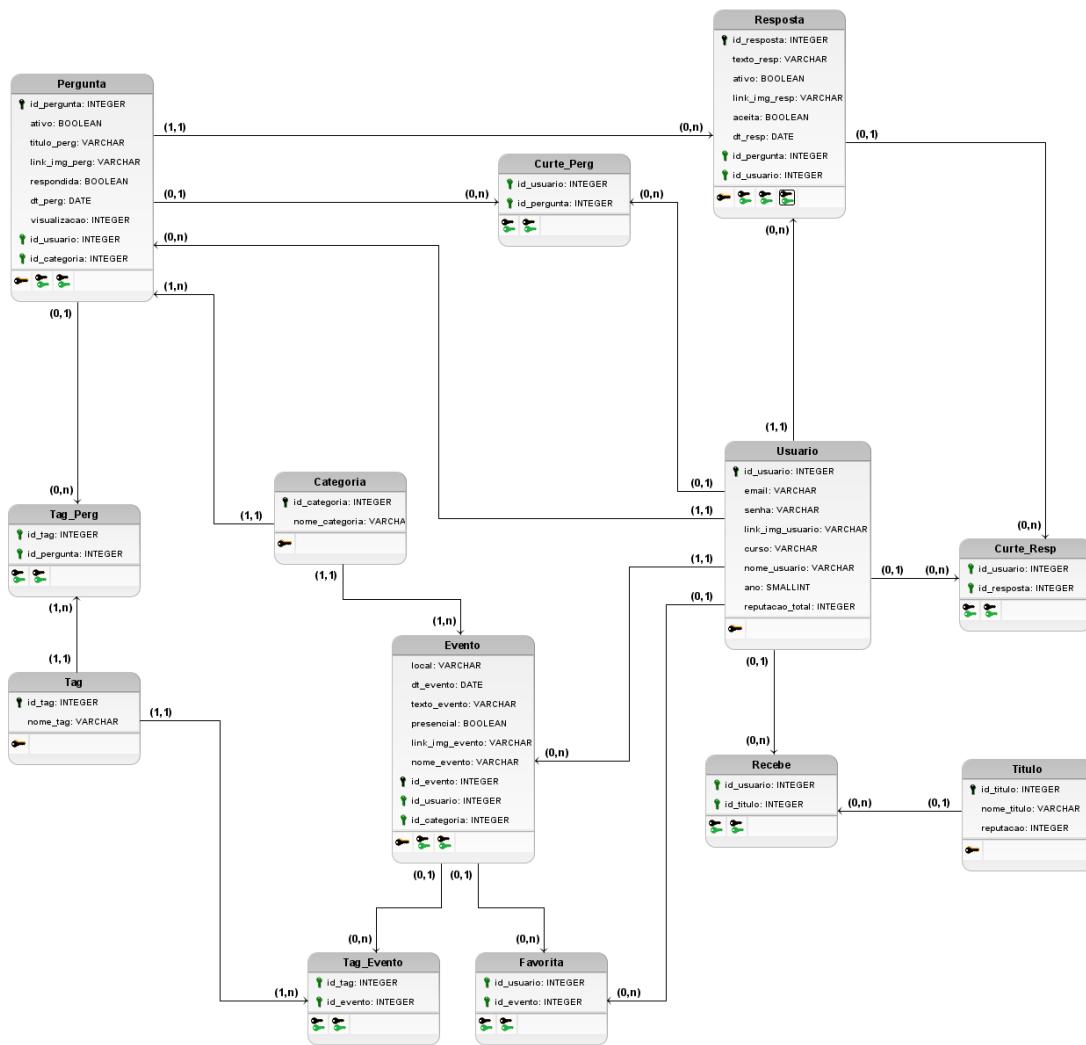
Fonte: Os autores.

3.2.3 Diagrama de Tabelas Relacionais

O Diagrama de Tabelas Relacionais DTR representa o modelo lógico do Banco de Dados. Segundo Ribeiro (201?), através do modelo lógico é representado de maneira mais clara as entidades e os relacionamentos, pois considera algumas limitações e implementa recursos como adequação de padrão e nomenclatura, define as chaves primárias e estrangeiras, normalização referencial, entre outras.

Deste modo, a Figura 5 representa o Diagrama de Tabelas Relacionais (DTR) do projeto de sistema IFriends.

Figura 5 – Diagrama de Tabelas Relacionais



Fonte: Os autores.

3.2.4 Dicionário de Dados

O Dicionário de dados é responsável por armazenar as informações de configuração do banco de dados e as estruturas que compõem suas respectivas tabelas. As estruturas definem os campos e suas propriedades (ALVES, 2020).

Conforme Date (2004), o Dicionário de dados é o lugar em que dentre outras coisas todos os diversos esquemas (externo, conceitual, interno) e todos os mapeamentos correspondentes são mantidos.

O Dicionário de dados contém os metadados, dados que explicam dados, com relação aos diversos objetos que são de interesse do próprio sistema. Exemplos desses objetos incluem índices, usuários, restrições de integridade, restrições de segurança, e assim por diante, informações que essenciais para que o sistema faça seu trabalho apropriadamente (DATE, 2004).

De tal modo, abaixo encontram-se os quadros que representam o **Dicionário de Dados (DD)** das tabelas de banco de dados do projeto **IFriends**.

Quadro 8 – Usuário.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_usuario	INT	11	N	PK	Chave primária do usuário
email	VARCHAR	50	N		E-mail institucional do usuário
senha	VARCHAR	50	N		Senha de acesso ao sistema
link_img	TEXT		S		link da imagem de perfil
curso	VARCHAR	50	S		Curso atual
nome_usuario	VARCHAR	120	N		Nome do usuário
ano	INT	1	S		Ano que o usuário cursa, ex.: 1ºano
reputacao_total	INT	11	N		Pontuação da reputação do usuário

Fonte: Os autores

Quadro 9 – Usuário_Pergunta.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_usuario	INT	11	N	FK	Chave estrangeira no usuário
id_pergunta	INT	11	N	FK	Chave estrangeira na pergunta

Fonte: Os autores

Quadro 10 – Usuário_Resposta.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_usuario	INT	11	N	FK	Chave estrangeira no usuário
id_resposta	INT	11	N	FK	Chave estrangeira na resposta

Fonte: Os autores

Quadro 11 – Usuário_Título.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_usuario	INT	11	N	FK	Chave estrangeira no usuário
id_titulo	INT	11	N	FK	Chave estrangeira no titulo

Fonte: Os autores

Quadro 12 – Título.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_titulo	INT	11	N	PK	Chave primária do título
nome_titulo	VARCHAR	50	N		Nome do título
reputacao	INT	11	N		Acumulo da pontuação

Fonte: Os autores

Quadro 13 – Resposta.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_resposta	INT	11	N	PK	Chave primária da resposta
id_usuario	INT	11	N	FK	Chave estrangeira de usuário
id_pergunta	INT	11	N	FK	Chave estrangeira de pergunta
texto_resp	TEXT		N		Conteúdo da resposta
ativo	BOOLEAN		N		Resposta ativa ou não
img_resp	TEXT		S		Imagen da resposta
aceita	BOOLEAN		N		Se a resposta foi aceita como solução válida para o autor da pergunta
dt_resposta	DATE	50	N		Data em que a resposta foi publicada

Fonte: Os autores

Quadro 14 – Tag.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_tag	INT	3	N	PK	Chave primária da tag
nome_tag	VARCHAR	50	N		Nome da tag

Fonte: Os autores

Quadro 15 – Pergunta.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_pergunta	INT	11	N	PK	Chave primária da pergunta
id_usuario	INT	11	N	FK	Chave estrangeira de usuário
dt_perg	DATE		N		Data da pergunta
titulo_perg	VARCHAR	50	N		Título da pergunta
texto_perg	TEXT		N		Descrição da pergunta
ativo	BOOLEAN		N		Pergunta ativa ou não
link_img_perg	TEXT		S		Link da imagem da pergunta
respondida	BOOLEAN		N		Se a pergunta já teve uma resposta útil a quem perguntou
visualizações	INT	5	N		Quantidade de visualizações da pergunta

Fonte: Os autores

Quadro 16 – Tag_Pergunta.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_assunto	INT	11	N	FK	Chave estrangeira no assunto
id_pergunta	INT	11	N	FK	Chave estrangeira na pergunta

Fonte: Os autores

Quadro 17 – Tag_Evento.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_assunto	INT	11	N	FK	Chave estrangeira no assunto
id_evento	INT	11	N	FK	Chave estrangeira no evento

Fonte: Os autores

Quadro 18 – Evento.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_evento	INT	11	N	PK	Chave primária do evento
id_usuario	INT	11	S	FK	Chave estrangeira de usuário
presencial	char	1	N		Local do evento, sendo presencial, online ou ambos
nome_evento	VARCHAR	50	N		Nome do evento
texto_evento	TEXT		N		Descrição sobre o evento
dt_evento	DATE		N		Data que o evento ocorrerá
img_evento	TEXT		S		Imagen do evento
local	TEXT		N		Local onde será realizado, tanto presencial como online

Fonte: Os autores

Quadro 19 – Categoria.

Atributo	Tipo	Tamanho	Nulo	Chave	Descrição
id_categoria	INT	11	N	PK	Chave primária da categoria
nome_categoria	INT	50	N	FK	Nome da categoria

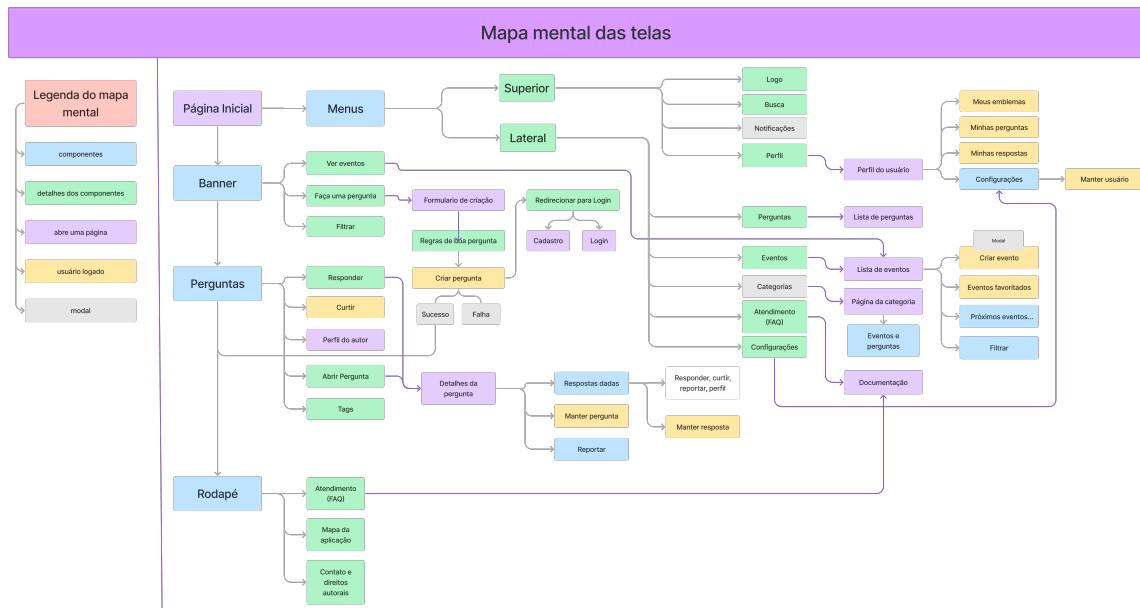
Fonte: Os autores

3.3 Prototipagem

Segundo Ferreira (2020), “prototipar é trazer, para o mundo real, o mundo palpável, as ideias de negócio construídas no mundo abstrato, na teoria”. Isto é, o autor comenta que um protótipo é um recurso utilizado para demonstrar e escolher a solução para representar uma ideia, podendo ser efetuado com entregas digitais, como telas de sistema. Dado isto, a próxima seção apresentará as telas prototipadas do projeto de sistema [IFriends](#).

Ainda, para auxiliar na prototipação das telas, foi elaborado um mapa mental de modo a representar melhor o fluxo do nosso projeto, que pode ser conferido na [Figura 6](#).

Figura 6 – Mapa mental



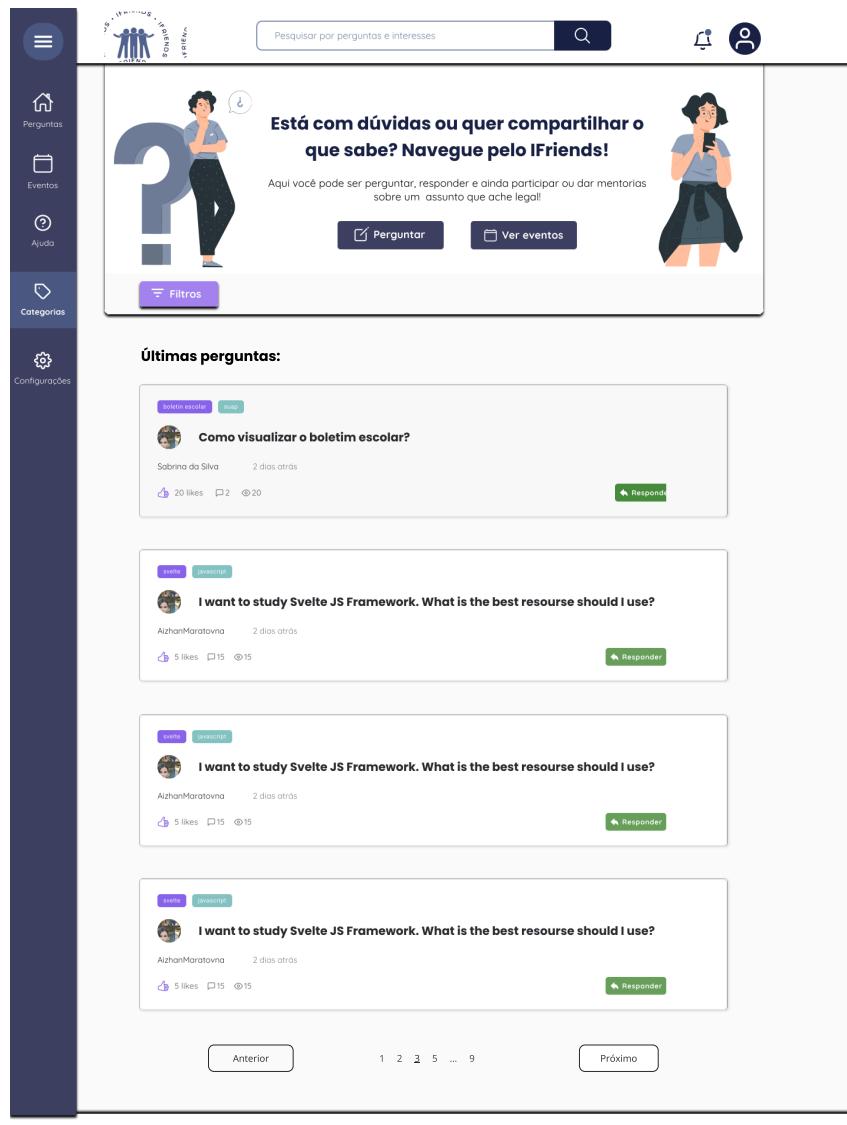
Fonte: os autores

3.3.1 Protótipos de alta fidelidade

Nesta seção estão contidas as figuras que representam as principais telas do sistema em relação a *Proof of Concept* (POC) do projeto, cada tela apresenta uma breve contextualização sobre o seu conteúdo. De todo modo, a apresentação pode ser visualizada também pelo [Figma](#).

A [Figura 7](#) apresenta a página inicial do projeto, onde o usuário entra em contato pela primeira vez com o sistema. Nela o usuário pode navegar através de dois menus disponíveis na página: o lateral e o superior, usar a barra de pesquisa, *logar* no seu perfil, acessar as suas configurações, entre outras ações disponibilizadas. Na página encontram as questões mais relevantes da comunidade, assim como os espaços destinados para a realização de uma pergunta ou de uma monitoria.

Figura 7 – Página inicial



Atendimento

FAQ
Entrar em contato
Políticas da comunidade
Termos de uso e condições

Mapa do site

Página inicial
Eventos
Categorias
Configurações
Contato

Fonte: os autores

Quando o usuário clica em uma pergunta ou em “Responder” ele é direcionado à página dessa pergunta como mostra a Figura 8, nela ele pode encontrar as respostas já fornecidas por outros membros da comunidade, assim como também pode deixar a sua contribuição.

Figura 8 – Pergunta e respostas

The screenshot shows a user interface for a community platform. On the left, a dark sidebar contains icons for Perguntas (Questions), Eventos (Events), Ajuda (Help), Categorias (Categories), and Configurações (Configurations). The main area features a search bar at the top with placeholder text "Pesquisar por perguntas e interesses" and a magnifying glass icon. Below the search bar is a circular logo with three stylized figures and the text "IFRIENDS". A navigation bar below the logo includes links for "boletim escolar" and "suap".

Question:

Como visualizar o boletim escolar?

Entre essa semana no IF e gostaria de saber como posso visualizar meu boletim escolar, me disseram que tenho que acessar por um tal de suap, podem me dar uma dica?

Responses:

Sabrina da Silva 2 dias atrás

Ola, tudo bem?
Sim, é possível ver as notas por meio do suap. Para isso você precisa fazer login com seu prontuário e senha no seguinte link: "<https://suap.ifsp.edu.br/accounts/login/>". Após efetuar o login você já se depara com o menu principal do sistema, nele você tem várias opções, mas a que nos interessa é "Ensino", nela clique em "Boletins" é pronto ta pronto o sorvetinho!
Espero ter ajudado =)

Juninha da silva 45 minutos atrás

Você pode ver suas notas através do suap, basta você fazer login e entrar na aba correspondente!

Sua resposta

Conhece uma resposta para essa dúvida? Mande a sua resposta!

Resposta:

Sua resposta

Enviar

Registration:

Registre-se
Não arrisque perder o seu progresso, registre-se agora é rápido e fácil!

Registrar usando email e senha

**Atendimento**

- FAQ
- Entrar em contato
- Políticas da comunidade
- Termos de uso e condições

Mapa do site

- Página inicial
- Eventos
- Categorias
- Configurações
- Contato

Fonte: os autores

Já a [Figura 9](#) corresponde a página de cadastro de perguntas, nessa tela são apresentados todos os elementos julgados necessários para a sua realização, nesta página ainda de encontra o manual de uma boa pergunta, tal foi elaborado com o intuito de ajudar e auxiliar o usuário na preparação de sua problemática.

Figura 9 – Cadastro de perguntas

Perguntas

Faça a sua pergunta

Título
Informe um título sucinto que ajude a compreender melhor o problema.

Descrição do problema
Inclua todas as informações necessárias para alguém conseguir responder sua pergunta.

Categorias
Selecione a categoria que melhor representa sua pergunta

Tags
Adicione algumas tags para complementar a sua pergunta

Como fazer uma boa pergunta?

- Resuma o seu problema**
 - Antes de fazer uma pergunta, tenha em mente "qual é o problema". Para isso é recomendado que primeiramente se atente a resumir detalhes e informações que podem ser úteis.
 - Considere que uma pergunta não se caracteriza como boa devido ao seu tamanho, mas sim devido às informações fornecidas.
 - Opte por um título sucinto e detalhado, e recorra a termos-chaves, as tags podem ajudar nesse sentido.
- Descreva o seu problema**
 - Apresente seu problema com o máximo de detalhes, o que você já tentou e conte-nos o que você conseguiu até então.
 - Lembre-se que conseguirá melhores respostas quando você fornecer e detalhar melhor seu problema.
 - Quando for apropriado, faça uso de imagens para exemplificar melhor o problema.
- Objetivo final**
 - O que é preciso para chegar a um resultado viável, tente ser o mais claro possível em expressar qual é o seu objetivo.

Registre-se
Não arrisque perder a sua pergunta, registre-se agora é rápido e fácil!

Enviar

Cancelar

Registre-se usando email e senha

**Atendimento**

- FAQ
- Entrar em contato
- Políticas da comunidade
- Termos de uso e condições

Mapa do site

- Página inicial
- Eventos
- Categorias
- Configurações
- Contato

Fonte: os autores

As telas apresentadas até o momento são aquelas que se encontram relacionadas com a **POC**, portanto vale salientar que esta seção será atualizada futuramente conforme o andamento e elaboração do projeto.

4 Prova de Conceito

De acordo com Lima, Rodrigues, Oliveira e Bernardino (2020), prova de conceito ou POC é o nome que se dá à demonstração da probabilidade de validação de uma ideia (ou conceito), podendo ser na área de TI ou na área dos negócios. A POC pode ser aplicada em um protótipo ou em um projeto em fase inicial, e normalmente segue um roteiro de testes. Esses testes são evidências, que demonstram que um conceito de design, proposta de negócio entre outros, são viáveis.

Segundo Lima, Rodrigues, Oliveira e Bernardino (2020), só é necessário realizar a prova de conceito, sempre que há desejo de implementar mudanças relacionadas a processos, ferramentas ou métodos, e isso se dá através dos testes designados pela mesma. Através da POC é possível determinar se o serviço ou produto funciona na prática e qual seu respectivo nível de eficácia e eficiência, além de ser extremamente importante tanto para o cliente como para o desenvolvedor, que no que lhe concerne, adquire a chance de implementar uma solução em um ambiente real de mercado onde todas as variáveis e possibilidades que podem acabar influenciando na solução, são expostas.

Para a prova de conceito do projeto de sistema IFriends, escolheu-se desenvolver, considerando a utilização do método ágil, apenas dois *Épicos*: Gestão de Perguntas e Gestão de Respostas, descritos na seção de análise de requisitos. Isto, pois, após discussões em equipe e analisando a definição de POC, os integrantes chegaram a um consenso de que estes são os itens mais fundamentais para que o funcionamento do projeto seja provado.

Utilizando as tecnologias elencadas anteriormente, foi possível criar todas as requisições escolhidas nos épicos para a API do projeto, e ainda pode-se publicá-la no Heroku, disponível na Figura 10 e conseguimos chamá-la no *front-end* da aplicação através da biblioteca Axios.

Figura 10 – API IFriends

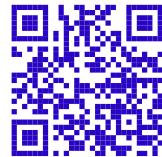


Fonte: Os autores

No entanto, ao publicar a aplicação ReactJS no Heroku, disponível na Figura 11, foi gerado um erro de limite de memória do Node.JS excedido, e portanto, aplicação saiu do ar. Isto ocorreu após terem sido adicionados mais recursos nela, ou seja, no último deploy feito. Outro ponto é que este problema acontece somente após serem feitas algumas requisições, e não no momento em que é feito o deploy e mesmo aumentando a capacidade

miníma de memória, o erro persistiu. A equipe pretende assim investigar o fato ocorrido para tomar possíveis planos de ação para as melhorias no projeto, partindo inicialmente da documentação do Heroku, que visa explicar o problema e trazer mais possíveis soluções.

Figura 11 – Aplicação IFriends



Fonte: Os autores

Com relação aos demais fatores do sistema, a equipe notou que foi possível testar a solução de maneira satisfatória, ainda que alguns problemas tenham acontecido no meio do caminho. A respeito da internacionalização e da criptografia, por exemplo, conseguiu-se encontrar recursos para fazê-las, visto que, no ReactJS, podemos definir a internacionalização com a biblioteca [Ant Design](#), porém a solução apresentou instabilidade após a implementação, e dessa forma, optou-se por não demonstrá-la na apresentação. Além disso, foi percebido que o próprio [Heroku](#) disponibiliza meios para inclusão da criptografia do sistema, mas conforme dito anteriormente, precisa-se solucionar o problema da hospedagem para que todos os recursos do sistema estejam funcionando on-line de maneira correta.

Dentre as demais mudanças feitas no projeto está a utilização do [Scoold](#) apenas como referência para o desenvolvimento, e não mais como uma tecnologia a ser incorporada no sistema.

Por outro lado, espera-se que após todos os apontamentos feitos, o projeto possa ser melhorado de forma iterativa para que a entrega da primeira versão seja feita sem instabilidades e, além disso, que a equipe possa realizar as entregas nos devidos prazos e com o cumprimento de todos os requisitos propostos.

Referências

ALVES, W. *Banco de dados: Teoria e Desenvolvimento*. Saraiva Educação S.A., 2020. ISBN 9788536533759. Disponível em: <<https://books.google.com.br/books?id=qD0IEAAQBAJ>>. Citado na página 26.

AMBLER, S. W. *Modelagem ágil: práticas eficazes para a Programação Extrema e o Processo Unificado*. [S.l.]: Bookman, 2004. Citado na página 12.

CRUZ, F. *Scrum e Agile em Projetos: Guia Completo*. 2. ed. Rio de Janeiro: Brasport Livros e Multimídia Ltda., 2018. Citado 11 vezes nas páginas 13, 15, 16, 17, 18 e 20.

DALLAVALLE, S. I.; CAZARINI, E. W. Regras de negócio, um fator chave de sucesso no processo de desenvolvimento de sistemas de informação. *Anais do XX ENEGEP-Encontro Nacional de Engenharia de Produção. São Paulo*, 2000. Citado na página 22.

DATE, C. *Introdução a sistemas de bancos de dados*. ELSEVIER EDITORA, 2004. ISBN 9788535212730. Disponível em: <<https://books.google.com.br/books?id=xBeO9LSIK7UC>>. Citado na página 27.

FERNANDES, L. Redes sociais online e educação: contributo do facebook no contexto das comunidades virtuais de aprendentes. *Universidade Nova de Lisboa, Portugal*, 2011. Citado na página 9.

FERREIRA, M. B. *Prototipagem e testes de usabilidade*. 1. ed. Curitiba: Contentus, 2020. Citado na página 31.

GUEDES, G. T. A. *UML 2 - Uma Abordagem Prática*. 2. ed. [S.l.]: novatec, 2011. Citado 2 vezes nas páginas 23 e 24.

JÚNIOR, W. M. P. *Apostila Engenharia de Software*. Minas Gerais, 2010. Disponível em: <http://www.waltenomartins.com.br/ap_es_v1.pdf>. Acesso em: 23 jul. 2021. Citado na página 10.

LEAL, G. *Linguagem, Programaç ao E Banco De Dados: GUIA PR'ATICO DE APRENDIZAGEM*. INTERSABERES, 2015. Disponível em: <<https://plataforma.bvirtual.com.br/Acervo/Publicacao/30495>>. Citado 2 vezes nas páginas 24 e 25.

LIMA, Y.; RODRIGUES, E.; OLIVEIRA, R.; BERNARDINO, M. Usando o teste ponta a ponta para garantia de confiabilidade de um sistema integrado de gestao: uma prova de conceito. In: SBC. *Anais da IV Escola Regional de Engenharia de Software*. [S.l.], 2020. p. 126–133. Citado na página 36.

MACHADO, F. N. R. *Análise e Gestão de Requisitos de Software–Onde nascem os sistemas*. [S.l.]: Saraiva Educação SA, 2018. Citado na página 19.

PEINADO, J.; AGUIAR, G. d. F. Compreendendo o kanban: um ensino interativo ilustrado. *Revista DaVinci. Curitiba (PR)*, v. 4, n. 1, p. 133–146, 2007. Citado 2 vezes nas páginas 13 e 14.

- REYES, C. *Experiência do Usuário (UX): Entenda o que é de forma prática.* 2020. [Online; accessed 10. Apr. 2022]. Disponível em: <<https://www.liferay.com/pt/resources/1/user-experience>>. Nenhuma citação no texto.
- RIBEIRO, L. *Modelagem de Dados: Modelo Conceitual, Modelo Lógico e Físico.* 201? Disponível em: <<https://www.luis.blog.br/modelagem-de-dados-modelo-conceitual-modelo-logico-e-fisico.html>>. Acesso em: 29 setembro 2021. Citado na página 26.
- ROSA, G. C. d. Identidade cultural em comunidades de usuários e desenvolvedores de software livre: o caso debian-rs. Pontifícia Universidade Católica do Rio Grande do Sul, 2008. Citado na página 8.
- SGANDERLA, M. A.; LACERDA, G.; RIBEIRO, V. G.; SILVEIRA, S. R. Aprimorando a gerência e o desenvolvimento de software com metodologias ágeis. *RCT-Revista de Ciência e Tecnologia*, v. 2, n. 2, 2016. Citado na página 13.
- SILVA, D. V. d. S.; SANTOS, F. A. d. O.; NETO, P. S. Os benefícios do uso de kanban na gerência de projetos de manutenção de software. In: SBC. *Anais do VIII Simpósio Brasileiro de Sistemas de Informação*. [S.l.], 2012. p. 715–725. Citado na página 14.
- SIMÕES, G. S.; VAZQUEZ, C. E. *Engenharia de Requisitos: software orientado ao negócio.* 1. ed. Rio de Janeiro: Brasport Livros e Multimídia Ltda., 2016. Citado na página 20.
- SOMMERVILLE, I. *Engenharia de Software.* 10. ed. São Paulo: Pearson Education do Brasil Ltda., 2019. Citado 6 vezes nas páginas 19 e 23.
- TORNOS, C. T. Significado da cor azul na psicologia. *Br.psicologia-online*, Br.psicologia-online.com, Sep 2021. Disponível em: <<https://br.psicologia-online.com/significado-da-cor-azul-na-psicologia-917.html>>. Nenhuma citação no texto.

Glossário

Ant	biblioteca React UI antd que auxilia na criação de interfaces interativas. - Citado em 10
Back-end	Refere-se a parte que está por trás da aplicação, responsável pela manipulação de dados voltada para o funcionamento interno de um sistema - Citado em 10, 11, 12, 15, 56
Discord	Aplicativo de comunicação instantânea, muito utilizado pela comunidade <i>gamer</i> , por sua simplicidade em possibilitar troca de mensagens, áudio, texto e vídeo. - Citado em 45, 46, 47, 48, 49, 50, 51, 54
Figma	Plataforma de criação de gráficos e prototipagem de projetos, focados principalmente em aplicações web. - Citado em 45, 50
Framework	Uma abstração que une códigos comuns entre vários projetos de software provendo uma funcionalidade genérica. - Citado em 10
Framework	Um framework ou arcabouço conceitual, é um conjunto de conceitos usado para resolver um problema de um domínio específico. - Citado em 12, 13
Front-end	Refere-se a parte visual e gráfica da interface de um sistema, elaborado por meio de outras linguagens e tecnologias. - Citado em 10, 11, 12, 15, 56
gamificação	Aplicação das estratégias dos jogos nas atividades do dia a dia, com o objetivo de aumentar o engajamento dos participantes. Se baseia no game thinking, que integra a gamificação com outros saberes do meio corporativo e do design. - Citado em 9
Google Forms	Ferramenta da Google para gerenciamento de pesquisas e formulários, utilizada para coletar e registrar informações de outras pessoas. - Citado em 52
Gource	Ferramenta utilizada para visualização em forma de diagramas e vídeos o desenvolvimento de um software. - Citado em 49, 55, 56, 57
Heroku	Plataforma de nuvem como serviço que suporta várias linguagens de programação. - Citado em 11, 36, 37, 57

IFriends	Nome dado ao projeto de sistemas desenvolvido, cujo significado se dá num trocadilho na junção das palavras friends (amigos, em inglês) e IF (Instituto Federal). - Citado em 8 , 12 , 16 , 22 , 24 , 25 , 26 , 27 , 31 , 36 , 52 , 53 , 56
Moodle	Ambiente virtual de ensino com o objetivo de auxiliar à aprendizagem dos alunos. - Citado em 47 , 51
Notion	Aplicação que contém ferramentas úteis para o gerenciamento do projeto e da equipe, uma delas sendo o quadro de kanban. - Citado em 45 , 51
PostgreSQL	Sistema gerenciador de banco de dados objeto relacional, desenvolvido como projeto de código aberto. - Citado em 11
React-Bootstrap	biblioteca que oferece os componentes clássicos do Bootstrap construídos em React. - Citado em 10
REST API	Arquitetura de sistema para serviços web. - Citado em 11
SVN	Sistema de controle de versão de uso obrigatório na disciplina. - Citado em 56
WhatsApp	Aplicativo de mensagens instantâneas e chamadas de voz para smartphones. - Citado em 52
YouTube	Plataforma destinada a compartilhamento de vídeos. - Citado em 15 , 45 , 51 , 56 , 57

Apêndices

APÊNDICE A – Perguntas da pesquisa de viabilidade

1. Qual é o seu grau de escolaridade?
2. Qual é o seu curso?
3. Você trabalha ou faz estágio?
4. Você sentiu dificuldade em se adaptar ao entrar no IF?
5. Descreva como foi a sua experiência (com relação as dificuldades na instituição e no ensino).
6. Com que frequência você costuma ir às monitorias?
7. Você usaria um sistema de perguntas e respostas do IF?
8. Você acredita que uma comunidade de perguntas e respostas te ajudaria na sua vida acadêmica?
9. De que forma isso faria/não faria diferença para você? (Fique a vontade de responder com toda sinceridade!).
10. Gostaria de compartilhar mais alguma coisa sobre o tema? Bem, sinta-se a vontade!

APÊNDICE B – Atas das Reuniões

B.1 1ºbimestre

B.1.1 Planejamento - 15/03/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto.

Local: [Discord](#).

Pauta(s): Primeira reunião realizada pela equipe com o objetivo de planejar os passos iniciais do projeto, nela a equipe estabeleceu um contrato social na plataforma [Figma](#) contendo os combinados essenciais para o convívio social entre os componentes da equipe e também iniciou-se um *board* de ideias iniciais para o projeto.

B.1.2 Planejamento/Alinhamento - 17/03/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto.

Local: [Discord](#).

Pauta(s): Reunião de planejamento e alinhamento, onde o foco da equipe esteve em discutir as ideias pré-selecionadas na reunião anterior para o projeto. Desde este primeiro momento a atenção da equipe se voltava principalmente para uma comunidade de dúvidas entre os estudantes do IF.

Para organização das tarefas e da equipe a plataforma [Notion](#), onde o quadro de *kanban* se encontra, foi criado e organizado inicialmente.

A equipe decidiu inicialmente que realizará reuniões nos dias e horários das janelas entre as aulas, permitidas pela grade curricular.

B.1.3 Planejamento/Alinhamento - 18/03/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto.

Local: [Discord](#).

Pauta(s): Reunião de planejamento onde a equipe discutiu as tarefas a serem realizadas e discutiu outras ferramentas de organização.

Criamos o canal do [YouTube](#).

Realizamos a primeira postagem para o *blog* da equipe.

Amadurecemos ainda mais a ideia da comunidade, pensando em mais funcionalidades para agregar na aplicação e anotando as dúvidas a respeito do tema e do funcionamento.

B.1.4 Alinhamento - 21/03/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto.

Local: Saguão [IFSP](#).

Pauta(s): Reunião realizada antes da aula de [PDS](#) para a equipe discutir algumas dúvidas, perspectivas e ideias sobre o projeto, visando otimizar o tempo em sala de aula.

B.1.5 Alinhamento - 25/03/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto.

Local: [Discord](#).

Pauta(s): Reunião de alinhamento, onde a equipe pesquisou em trabalhos anteriores a formatação do documento da proposta inicial para preparar as etapas. Aproveitamos também para melhorar o *layout* do *blog* da equipe.

B.1.6 Planejamento/Alinhamento - 28/03/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto.

Local: Laboratório [IFSP](#).

Pauta(s): Reunião de alinhamento e planejamento, onde a equipe conversou com os orientadores sobre soluções e funcionalidades para a proposta da comunidade. Neste ponto, resolvemos um problema antigo, validar que o usuário seja de fato aluno da instituição, a solução encontrada foi enviar um *e-mail* de validação para o *e-mail* institucional do aluno. Foi citado também outra proposta de projeto para a equipe realizar, uma plataforma de controle e gestão financeira. Porém, a equipe optou por seguir na proposta da comunidade. Partimos para as tarefas, iniciando o desenvolvimento de um questionário direcionado aos alunos do instituto para estudar a viabilidade de criação do projeto.

Sobre o *blog*, finalizamos a melhoria de seu *layout* e definimos que as publicações serão realizadas aos sábados pela manhã.

B.1.7 Planejamento/Retrospectiva - 02/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto.

Local: [Discord](#).

Pauta(s): Reunião de retrospectiva e planejamento das atividades, onde revisamos a publicação da semana no *blog* e as perguntas para a pesquisa de viabilidade. Realizamos a entrega sobre as tecnologias que serão utilizadas no desenvolvimento do projeto no *Moodle* da disciplina e definimos duas tarefas: divulgação da pesquisa de viabilidade a partir de segunda-feira para alunos e ex-alunos da instituição e gerenciamento do *backlog* para cada parte do projeto.

B.1.8 Planejamento - 04/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsu-moto.

Local: Laboratório [IFSP](#).

Pauta(s): Reunião de planejamento, onde a equipe após receber as orientações para a apresentação da proposta inicial, organizou as tarefas a serem realizadas por cada componente, organizou a formatação do documento para a proposta inicial e iniciou a divulgação do formulário para pesquisa de viabilidade da proposta.

B.1.9 Alinhamento - 09/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsu-moto.

Local: [Discord](#).

Pauta(s): Reunião de alinhamento, onde a equipe se reuniu para realizar as atividades destinadas a apresentação da proposta inicial.

B.1.10 Alinhamento - 10/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsu-moto.

Local: [Discord](#).

Pauta(s): Reunião de alinhamento, onde a equipe se reuniu para realizar as atividades destinadas a apresentação da proposta inicial.

B.1.11 Retrospectiva - 12/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsu-moto.

Local: Biblioteca [IFSP](#).

Pauta(s): Reunião retrospectiva, onde a equipe analisou como foi o processo para realizar a entrega da proposta inicial, desta forma foram levantados os pontos positivos: todos estarem reunidos em chamada para realizar as tarefas, conseguimos entregar o que era

esperado e apesar das dificuldades enfrentadas a apresentação fluiu bem. Os pontos negativos: realizar muitas tarefas no final de semana ficou puxado para a equipe e não ter realizado um ensaio antes da apresentação, como melhoria, queremos marcar mais reuniões com os professores.

B.1.12 Planejamento/Alinhamento - 17/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto.

Local: [Discord](#).

Pauta(s): Reunião de planejamento e alinhamento para a semana, onde a equipe planejou as próximas *sprints* da [POC](#) juntamente com os professores e aproveitou para conversar sobre as avaliações das equipes em relação a apresentação da proposta inicial.

B.1.13 Planejamento - 18/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto.

Local: Laboratório [IFSP](#).

Pauta(s): Reunião de planejamento onde a equipe criou o *backlog* do produto com o uso das histórias de usuário e definiu as tarefas da semana, preparando-se para os dois épicos para a [POC](#).

B.1.14 Alinhamento - 21/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto.

Local: [Discord](#).

Pauta(s): Reunião de alinhamento, onde a equipe terminou de elaborar as histórias de usuário, definiu as prioridades para a [POC](#) e votou por meio do *Planning Poker* - descrito pela metodologia Scrum - para estimarmos os esforços necessários para a conclusão de cada história. Durante as discussões da equipe para a execução desta tarefa, muitos pontos sutis mas que poderiam ser perigosos no futuro, foram levantados e anotados para discutirmos com os orientadores.

B.1.15 Alinhamento - 25/04/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto.

Local: Laboratório [IFSP](#).

Pauta(s): Reunião de alinhamento onde a equipe conversou com os orientadores sobre

as dúvidas levantadas na reunião anterior, durante a elaboração da tarefa de definição das histórias de usuários, apresentou os diagramas de entidade e relacionamento e o diagrama de casos de uso para que fossem alinhados corretamente. Iniciamos também as configurações para criação do vídeo do [Gource](#).

B.1.16 Alinhamento - 01/05/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto.

Local: [Discord](#).

Pauta(s): Reunião de alinhamento onde a equipe finalizou as histórias de usuário com base na discussão realizada em aula com os orientadores, alinhou o fluxo de usuário no sistema, os requisitos funcionais, não funcionais e as regras de negócio. Iniciamos o planejamento para realização da apresentação da [POC](#) e como orientação dos professores, decidimos deixar para desenvolver o épico de Gestão de Eventos apenas se sobrar tempo.

B.1.17 Alinhamento - 03/05/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto

Local: [Discord](#). Pauta(s): Realização de tarefas para a [POC](#), configurações de ambiente e alinhamento da documentação.

B.1.18 Retrospectiva - 16/05/2022

Integrantes: Anaí Rojas, Jamilli Gioielli, José Roberto, Julia Romualdo, Kaiky Matsumoto

Local: Laboratório [IFSP](#).

Pauta(s):

APÊNDICE C – Publicações no blog da equipe

C.1 1ª Semana - 14/03 à 20/03

Primeiramente, bem vindos à primeira postagem da equipe! O principal objetivo deste blog é dar aos leitores a possibilidade de acompanhar nosso "diário semanal" de desenvolvimento dentro da disciplina de Prática de Desenvolvimento de Sistemas (ou [PDS](#) para os mais íntimos).

Mas antes de dar continuidade ao texto, precisamos apresentá-los a equipe Bunka Bytes, cujo nome é parte inspirado no tema do trabalho anterior (em [TDS](#)), já que este possuía um foco cultural e "Bunka" em japonês significa "cultura"; e outra parte se deve a uma referência/trocadilho com a área de informática, pois falando, é quase como se estivéssemos dizendo "Boom Kabbytes".

Integrantes da equipe:

- Anai Rojas
- Jamilli Gioielli
- José Roberto
- Julia Romualdo
- Kaiky Matsumoto

Dos integrantes, escolhemos a Jamilli como nossa representação de gerente, devido a seus conhecimentos em organização, metodologias e ferramentas de gerenciamento e por sua facilidade comunicativa.

Tendo isso em vista, nesta primeira semana da disciplina, realizamos três reuniões pela plataforma [Discord](#), nas quais conhecemos melhor nossos colegas de equipe, fizemos um Contrato Social e criamos um *Brainstorming* utilizando conceitos de *Design Thinking* na plataforma [Figma](#), que ajudou numa melhor visualização das ideias centrais para o projeto. Também tivemos acesso aos trabalhos anteriores e consultamos dois trabalhos que eram mais semelhantes ao nosso tema principal.

A partir do *Brainstorm*, a ideia que mais se consolidou foi a proposta de criação de uma comunidade de [Q/A](#) e mentorias como forma de apoio aos estudantes do [IFSP](#).

Pensamos em muitas outras, mas esta acabou sendo nossa favorita, com base nos requisitos da disciplina de [PDS](#). Portanto, nosso objetivo para a segunda semana é conversar com os professores sobre a ideia e a desenvolver melhor com base nas nossas discussões e dúvidas a serem sanadas.

Por último, ao final dessa semana, criamos um *e-mail* conjunto para a criação deste blog e o canal no [YouTube](#), também conseguimos acessar o Subversion, criar um canal no [Discord](#), uma página de gerenciamento no [Notion](#) e uma logo para a equipe.

Por: Jamilli Gioielli e Julia Romualdo

C.2 2^a Semana - 21/03 à 27/03

Estamos de volta, leitor!

No início dessa semana, voltamos às atividades presenciais e tivemos um primeiro contato com a disciplina neste formato. A partir daí, separamos as 5 ideias que mais se destacaram do nosso *Brainstorm* da semana anterior e as compartilhamos com os professores. Recebemos algumas sugestões sobre a ideia de comunidade e nos foi recomendado documentar as ideias para enviar no ambiente do [Moodle](#) da disciplina.

Tendo isso em vista, precisamos nos reunir durante os próximos dias para passar a construção dos nossos pensamentos em formato textual e explicativo. Entretanto, enfrentamos algumas dificuldades nesse sentido, estando a maioria delas em torno do curto tempo que temos entre trabalho e escola para fazer as reuniões. Tentamos conversar na hora anterior às aulas, mas percebemos que faltava apenas documentar melhor o que havíamos conversado. Não foi possível realizar isto na escola devido aos problemas de conexão de rede, então optamos por fazermos aos poucos durante a semana e nos reunirmos depois da aula, via [Discord](#), para revisar o conteúdo. De todo modo, conseguimos fazer a entrega das duas tarefas da semana no [Moodle](#).

Nossos objetivos para próxima semana são: definir a proposta para o projeto com base no *feedback* dos professores e planejar melhor nossos dias e horários para reuniões.

Por: Jamilli Gioielli

C.3 3^a Semana - 28/03 à 03/04

Estamos de volta, Bunkers!

Nesta semana, trabalhamos na viabilidade da nossa proposta. Em aula, discutimos com os professores as funcionalidades que a comunidade pode ter, encontramos uma solução para validar os alunos cadastrados nela, enviando um *e-mail* de confirmação apenas ao *e-mail* institucional, que é de posse dos alunos, garantindo assim que os usuários sejam

apenas alunos do instituto e por fim, falamos também sobre uma segunda proposta da equipe, que os professores gostaram bastante e deram forte impulso em desenvolve-la por atingir um grupo maior de usuários e ser algo que todos precisam cotidianamente, que é um sistema baseado em controlar gastos e auxiliar na organização financeira.

Entretanto, optamos por continuar com a comunidade, a qual batizamos com o nome: [IFriends](#), principalmente porque é mais próximo da nossa realidade como alunos do instituto e por não termos experiência em desenvolvimento de aplicações *mobile*, o modelo que enxergamos ser o ideal para a criação da segunda proposta. E como análise prática da viabilidade do [IFriends](#), elaboramos um formulário, via [Google Forms](#), para no início da próxima semana, enviarmos aos nossos colegas e alunos do instituto com o intuito de investigar se eles fariam uso da comunidade.

Falando agora sobre a nossa organização como equipe, ainda estamos nos adaptando com nossos horários entre estudos, trabalho e locomoção, então nosso foco será realizar reuniões rápidas e objetivas principalmente nos horários disponíveis antes das aulas começarem.

Resumo das atividades de cada membro da equipe:

- Anai - Elaborou formulário da pesquisa de viabilidade
- Jamilli - Organizou as atividades a serem realizadas pela equipe
- José, Julia e Kaiky - Trabalharam em melhorias de layout e postagens do blog

Para a próxima semana a equipe tem como objetivo estudar as tecnologias e linguagens a serem utilizadas no desenvolvimento do projeto, estruturar um *backlog* para gerenciamento eficiente das atividades, trabalhar na identidade visual e design da marca [IFriends](#).

Por: Julia Romualdo

C.4 4^a Semana - 04/04 à 10/04

Estamos de volta, Bunkers!

No inicio desta semana, realizamos a pesquisa de viabilidade da nossa proposta inicial da comunidade, para isto, elaboramos um formulário via [Google Forms](#), que ficou disponível para receber respostas de segunda-feira (04/04) à sexta-feira (08/04) e todos os integrantes da equipe ficaram responsáveis por enviar o endereço de compartilhamento - *link* - nos grupos de [WhatsApp](#) para os alunos da instituição - público-alvo da nossa proposta - respondessem a dez perguntas e compartilharem algumas experiências como alunos do [IFSP](#) que ajudassem a equipe a compreender se a proposta era ou não viável.

Ainda nesta semana, recebemos das professoras as orientações para apresentação da proposta inicial, juntamente com a entrega da documentação e estudo de dois projetos anteriores. Para realização desta tarefa, durante a semana a equipe se organizou da seguinte forma:

- Anai - Criou *branchmarketing* para a aplicação e estudou o projeto WebLab.
- Jamilli - Organizou as atividades a serem realizadas pela equipe e estruturou as documentações.
- José - Estudou as tecnologias a serem utilizadas no projeto e também do projeto Monitorando.
- Julia - Compilou dados da pesquisa de viabilidade de proposta e estudou o projeto Monitorando.
- Kaiky - Estudou as tecnologias a serem utilizadas no projeto e também do projeto WebLab.

Para a próxima semana a equipe tem como objetivo apresentar a proposta e estudar os *feedbacks* fornecidos pelos professores e colegas durante a apresentação.

Por: Julia Romualdo

C.5 5^a Semana - 11/04 à 17/04

Estamos de volta, Bunkers!

Iniciamos esta semana com a apresentação da proposta inicial da comunidade [IFriends](#) aos nossos professores e colegas de classe, dos quais recebemos orientações e *feedbacks*. Neste mesmo cenário, também participamos da apresentação das outras equipes e compartilhamos nossos *feedbacks* aos mesmos.

Após a apresentação, a equipe se reuniu na biblioteca do [IFSP](#) para realizar uma retrospectiva, com o objetivo de avaliar o funcionamento da equipe durante a intensa semana de trabalhos que tivemos para realizar a entrega da proposta inicial. Aproveitamos este momento, para decidir os pontos a serem melhorados na apresentação para realizarmos a gravação e entrega do vídeo da proposta e também ajustamos alguns pontos que faltavam ser encaixados sobre as tecnologias que serão utilizadas no projeto.

Além disso, aproveitamos o feriado para revisar alguns conceitos que tínhamos dúvidas e procurar possíveis melhorias para nossa apresentação e documentação. Uma das coisas que percebemos era que ainda precisávamos validar o arquivo `equipe.yaml` no `yamllint`, já que ele não estava sendo refletido na página de Blogs de Trabalhos, por

isso aproveitamos para ajustá-lo de acordo com os apontamentos que foram dados pelo validador.

De todo modo, a equipe tem como meta para a próxima semana a atualização das fontes do projeto de acordo com o *feedback* que será dado pelos demais colegas e pelos professores após a apresentação da proposta. Além disso, pretendemos postar o vídeo da proposta - já com as melhorias - e também entregar nossas avaliações sobre as demais equipes.

Por isso, contamos com os feriados para adiantar o máximo de atividades possíveis e já pensar em alguns itens de *backlog* para que possamos iniciar a primeira *sprint* com o cronograma do projeto já bem definido. Pensando nisso também, a equipe não se dividiu como na semana anterior para a execução das tarefas, já que neste primeiro feriado focaremos juntos em tratar das melhorias, estudar as tecnologias propostas e trabalhar no planejamento do projeto.

Por: Julia Romualdo e Jamilli Gioielli

C.6 6^a Semana - 18/04 à 24/04

Estamos de volta, Bunkers!

Nesta semana, iniciamos a aula de segunda-feira de forma bem produtiva, pois conversamos bastante com os orientadores a respeito das próximas *sprints* a serem planejadas e executadas pela equipe. Assim partimos para a criação do *backlog* do produto com o uso das histórias de usuário, depois definimos as tarefas da semana já nos planejando para os três épicos a serem elaborados para a entrega da Prova de Conceito, sendo eles: Gestão de Perguntas, Gestão de respostas e Gestão de Eventos.

Além disso, aproveitamos o feriado de Tiradentes para nos reunirmos através da plataforma [Discord](#), com o objetivo de terminar a definição das histórias de usuário, neste momento muitos pontos sutis sobre a aplicação, que poderiam se tornar inimigos da equipe futuramente, foram levantados e anotados para discutir com os professores. Também votamos por meio do *Planning Poker* - descrito pela metodologia *Scrum* - para entendermos sobre uma estimativa de esforço para que cada história seja concluída. Junto a isso, também podemos elencar alguns requisitos não funcionais e regras de negócio.

Desta forma, as tarefas realizadas pela equipe durante esta semana foram organizadas da seguinte forma:

- Anai e Jamilli - Iniciar a prototipagem das telas.
- José e Kaiky - Iniciar a modelagem de dados.
- Julia - Iniciar os ajustes na documentação.

Para a próxima semana, além de dar início as tarefas da *sprint*, a equipe tem o objetivo de terminar as tarefas de planejamento e apresentá-las aos orientadores para que possamos esclarecer dúvidas e saber os pontos a serem melhorados.

Por: Julia Romualdo

C.7 7^a Semana - 25/04 à 01/05

Estamos de volta, Bunkers!

Iniciamos esta semana validando com os professores as atividades que a equipe esteve realizando desde a semana passada, fomos orientados quanto a melhor organização dos tópicos do documento, a remover algumas histórias dos épicos para a [POC](#), - pois alguns pontos estavam fugindo da ideia da [POC](#), que é provar que o conceito principal, ou seja, o fluxo principal da aplicação está funcionando -, a realizar a entrega do épico de Gestão de Eventos, na [POC](#), apenas se sobrar tempo, ajustar o diagrama de entidade relacionamento e o diagrama de casos de uso. Aproveitamos este momento ainda em aula para iniciar as configurações do [Gource](#), pensando em futuras entregas da disciplina.

Devido ao final de bimestre, a equipe esteve ocupada com as atividades de outras disciplinas e não conversou muito durante esta semana, mas os componentes continuaram no desenvolvimento - visando o termo - das atividades propostas na semana passada, organizadas da seguinte forma:

- Anai - Terminar a prototipagem das telas.
- José e Kaiky - Ajustar a modelagem de dados.
- Jamilli - Terminar a prototipagem das telas e ajustar os tópicos de gerenciamento e metodologias da documentação.
- Julia - Adicionar os apêndices na documentação e pesquisar sobre o [Gource](#).

Para a próxima semana a equipe tem como objetivo desenvolver a Prova de Conceito e a documentação relacionada a mesma.

Por: Julia Romualdo

C.8 8^a Semana - 02/05 à 08/05

Estamos de volta, Bunkers!

Esta semana devido a problemas na infraestrutura do encanamento do campus IFSP não tivemos aulas de maneira presencial e poucos professores ministraram no for-

mato [EAD](#), aproveitamos então o plantão de segunda-feira com os orientadores para alinharmos o fluxo de usuário e o protótipo para a [POC](#). No decorrer da semana utilizamos os horários que seriam destinados às aulas para realizarmos as tarefas de desenvolvimento da [POC](#) (desenvolvimento, documentação e apresentação), para isto a equipe ficou organizada da seguinte maneira:

- Anai - Realizar ajustes finais no protótipo e ajustar a documentação.
- José e Jamilli - Configurar o ambiente e desenvolver o [Front-end](#)
- Julia - Realizar ajustes na documentação e gerar vídeo do [Gource](#).
- Kaiky - Configurar o ambiente e desenvolver [Back-end](#) e Banco de Dados.

Para a próxima semana a equipe tem como objetivo apresentar a Prova de Conceito e realizar reunião retrospectiva sobre o desenvolvimento da [POC](#).

Por: Julia Romualdo

C.9 9^a Semana - 09/05 à 15/05

Estamos de volta, Bunkers!

Iniciamos esta semana com a apresentação da Prova de Conceito para a turma e orientadores, e devemos dizer que nosso maior inimigo para esta entrega com certeza foi o tempo, pois deixamos de cumprir alguns requisitos da bíblia do Ivan porque nos faltou tempo hábil para concluir alguns tópicos ali estipulados. Ainda assim, acreditamos que conseguimos demonstrar que o fluxo principal da nossa aplicação estava funcionando.

Por outro lado, pudemos cumprir grande parte dos requisitos necessários para as entregas do primeiro bimestre, faltando apenas nossa planilha de avaliação da equipe, que foi postada no [SVN](#) nessa semana, um pouco tarde devido a um imprevisto ocorrido com o sistema (que ficou fora do ar durante algumas horas).

A equipe tinha como um dos objetivos para essa semana a realização da reunião retrospectiva referente ao desenvolvimento da [POC](#), porém devido a alta demanda que as outras disciplinas exigiram durante a semana - provas, apresentações e outras atividades -, não encontramos um bom momento para nos reunirmos, ficando isso como uma meta para a próxima semana, juntamente com o planejamento para as próximas etapas do desenvolvimento do [IFriends](#). Além disso, precisamos atualizar nosso canal no [YouTube](#) com os vídeos para o segundo bimestre, visto que não conseguimos gravá-los e editá-los até o momento.

Por: Julia Romualdo e Jamilli Gioielli

C.10 10^a Semana - 16/05 à 22/05

Estamos de volta, Bunkers!

Podendo respirar um pouco mais após a intensa semana de entrega de atividades que tivemos durante a semana passada, iniciamos os trabalhos novamente com a reunião de retrospectiva referente a entrega da [POC](#), com isso chegamos as seguintes conclusões: do que foi bom, consideramos a entrega da [API](#) completa; para o que foi ruim, por outro lado, consideramos a falta de tempo, pois por mais que soubéssemos como desenvolver um requisito, como a internacionalização ou o vídeo de demonstração, não sobrou tempo para fazer; e por último, outro problema, enfrentado não apenas por nossa equipe mas de maneira geral na turma, foi a incerteza e a definição errada que construímos sobre a [POC](#) a partir de experiências anteriores, por isso percebemos que não conseguimos definir o que seria extremamente essencial para essa entrega e pode ser que tenhamos focado mais em coisas adicionais do que no principal. Assim, como plano de ação escolhemos que precisamos definir melhor o que é simples e objetivo para nossas entregas, para evitar que tenhamos confusões desnecessárias que acabem dificultando a entregarmos o que era necessário.

Os professores também realizaram alguns apontamentos a partir das nossas entregas relativas do primeiro bimestres, para que possamos realizar os alinhamentos necessários, tais como: representar o [Heroku](#) na arquitetura, postar os vídeos da apresentação da [POC](#) no [YouTube](#), realizar *commits* com mais frequência e enviar os arquivos do [L](#)^A_T[EX](#)que estamos usando no Overleaf.

Além disso, os professores separaram uma parte da aula para assistirmos a alguns projetos dos alunos da turma 231 em [PJI](#), e permitiu que fizemos apontamentos e sugestões para os colegas sobre suas ideias.

Tendo realizado esses dois momentos de alinhamento a equipe documentou tudo, organizou as tarefas e ajustes que deveriam ser realizados durante a semana e partimos para os ajustes. Conseguimos enviar e ajustar o que faltava para o primeiro bimestre, como a questão do [Gource](#) e ainda incluímos alguns requisitos que faltavam na aplicação e na *Application Programming Interface (API)* (como a inclusão do *Swagger UI* e criptografia da autenticação). Como a próxima semana será de conselhos de classe, a equipe espera poder trabalhar mais no projeto e dar continuidade no desenvolvimento do mesmo, além de reavaliarmos o que for necessário de acordo com os demais *feedbacks* dos professores (que ainda serão feitos sobre a [POC](#)).

Por: Julia Romualdo e Jamilli Gioielli