

CENTRO PAULA SOUZA
FACULDADE DE TECNOLOGIA DE ITAPETININGA
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS

EFRAIM DE ANDRADE MORAIS JUNIOR
GABRIEL DISSOTTI DO NASCIMENTO RODRIGUES
MARCOS DISSOTTI DO NASCIMENTO RODRIGUES

FATEC EDU COLABORATIVO: DESENVOLVIMENTO DO APLICATIVO HELPIN

CENTRO PAULA SOUZA
FACULDADE DE TECNOLOGIA DE ITAPETININGA
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS

EFRAIM DE ANDRADE MORAIS JUNIOR
GABRIEL DISSOTTI DO NASCIMENTO RODRIGUES
MARCOS DISSOTTI DO NASCIMENTO RODRIGUES

FATEC EDU COLABORATIVO: DESENVOLVIMENTO DO APLICATIVO HELPIN

Projeto de Pesquisa apresentado à disciplina de Metodologia de Pesquisa Científica do Curso Superior de Tecnologia em Comércio Exterior da Faculdade de Tecnologia de Itapetininga sob orientação da Prof^a Dr^a Andressa Silvério Terra França.

RESUMO

Este trabalho apresenta as tecnologias e o processo de desenvolvimento de uma plataforma móvel para promover integração entre os alunos, monitores e professores da instituição Fatec Itapetininga, como uma forma de poder ajudar os alunos com dificuldades nas matérias e aproximá-los de outras pessoas que detém o conhecimento que eles precisam. Tais pessoas podem ser professores, monitores e os próprios alunos da faculdade, que exerceriam o papel de ensinar por meio de um chat em tempo real e um fórum. Além disso, os conhecimentos adquiridos por meio do fórum do aplicativo ficaram disponíveis para futuras consultas, de modo que até mesmo os alunos que estão ensinando poderão revisar o que ensinaram anteriormente e consequentemente aprimorarão seus conhecimentos sobre as matérias da faculdade e ao mesmo tempo criarão *networking* ao interagir com outros alunos.

SUMÁRIO

1	INTRODUÇÃO	3
2	DELIMITAÇÃO DO TEMA.....	4
3	JUSTIFICATIVA	4
4	OBJETIVOS.....	5
1.1	OBJETIVO GERAL	5
1.2	OBJETIVOS ESPECÍFICOS	5
5	PROBLEMATIZAÇÃO	6
6	HIPÓTESES.....	6
7	METODOLOGIA	6
8	REVISÃO DA LITERATURA	7
7.1	APRENDIZAGEM COLABORATIVA E A ABORDAGEM SOCIAL.....	7
7.2	AS TECNOLOGIAS E O DESENVOLVIMENTO DE SOFTWARE	8
7.3	AS TECNOLOGIAS DE VERSIONAMENTO DE CÓDIGO.....	9
7.4	O PROCESSO DE DOCUMENTAÇÃO DE SOFTWARE.....	11
7.5	A GESTÃO DE PROJETO	12
9	CRONOGRAMA	14
10	DESENVOLVIMENTO	15
8.1	LEVANTAMENTO DE REQUISITOS	15
RF-001	Registrar-se	15
RF-002	Realizar login	15
RF-003	Listar cursos.....	16
RF-004	Listar perguntas	16
RF-005	Fazer pergunta.....	17
RF-006	Responder pergunta	17
RF-007	Exibir detalhes da pergunta	18
RF-008	Editar configurações da conta	18
RF-009	Cadastrar conhecimentos	19
RF-010	Remover conhecimento.....	19
RF-011	Exibir alunos no <i>chat</i>.....	20
RF-012	Enviar mensagem.....	20

RF-013 Receber mensagem.....	21
RF-015 Listar minhas dúvidas.....	21
RF-016 Recuperar senha	22
8.2 OS DIAGRAMAS DA UML	22
8.3 PROTOTIPAGEM DO APLICATIVO	26
11 CONSIDERAÇÕES FINAIS.....	36
REFERÊNCIAS.....	37

ÍNDICE DE FIGURAS

Figura 1: Exemplo do fluxo de desenvolvimento com a padronização do GitFlow ...	10
Figura 2: Exemplo do fluxo de desenvolvimento com SCRUM	12
Figura 3: Tela de cadastro	26
Figura 4: Tela de login.....	27
Figura 5: Tela principal e tela de listagem de conhecimentos.....	29
Figura 6: Tela de visualização de postagem e escrevendo uma resposta	33
Figura 7: Tela de chat ou bate-papo e tela de conversação	35

1 INTRODUÇÃO

De acordo com Lamin (2018) em seu artigo acerca do aprendizado, a melhor maneira de aprender é ensinando. Comparado com outras formas de aprendizado como ler um livro, assistir uma aula ou uma videoaula, estamos fixando muito mais o conhecimento em nosso cérebro, tendo isso em mente deu início ao projeto Helpin onde foca em ajudar os alunos com dificuldades e também os usuários que estão ensinando e irão aumentar seu próprio conhecimento ao ensinar, isso por meio desta plataforma mobile que intermedia a interação entre os usuários com recursos como chat em tempo real e um fórum para centralizar as dúvidas para consultar a qualquer momento. Atualmente um aluno com dificuldades em alguma matéria irá buscar na internet alguma solução ou até mesmo tirar a dúvida presencialmente com algum professor ou monitor do curso, o problema surge quando o usuário não tem tempo para comparecer na faculdade no horário disponibilizado para tirar dúvidas seja por morar longe, estar trabalhando ou ter algum outro tipo de compromisso que o impede de ir até o local. O objetivo deste projeto é quebrar a barreira da distância entre o público alvo que seria os alunos e os detentores do conhecimento como monitores, professores e alunos dispostos a compartilhar seu conhecimento. Neste documento será exposto todas as etapas para o desenvolvimento deste projeto, tal como os requisitos funcionais e não funcionais, os diagramas que mais se encaixaram para a situação como o de componentes, casos de uso e de atividade logo mais é explicado porque seriam os mais plausíveis de utilizar, as tecnologias que compõem a estrutura do aplicativo, todo o fluxo do desenvolvimento e como scrum foi aplicado neste fluxo.

2 DELIMITAÇÃO DO TEMA

Desenvolvimento de um aplicativo que deverá facilitar a comunicação entre os alunos para compartilhar conhecimento e suscitar o aprendizado colaborativo na Faculdade de Tecnologia de Itapetininga.

3 JUSTIFICATIVA

O desenvolvimento de ferramentas que estimulam o aprendizado colaborativo é de suma importância para se alcançar melhorias efetivas no desempenho escolar dos alunos. A Fatec Itapetininga não dispõe de uma ferramenta onde os alunos que anseiam por aprender possam encontrar outros alunos dispostos a ensinar. Com isso, esse trabalho justifica-se para o desenvolvimento de uma ferramenta que facilite a troca de conhecimentos entre os alunos.

Até o momento na Fatec Itapetininga, quando os alunos necessitam compreender um determinado assunto e não conseguem através de fóruns ou pesquisa na internet, eles procuram por monitores de determinadas matérias para fazerem suas perguntas e estes devem tentar respondê-las. O problema é que os programas de monitorias possuem dias e horários específicos, tais que muitas vezes não atendem as necessidades dos alunos em função das suas ocupações como trabalho ou por questões geográficas como a distância, falta de transporte e muitas outras variáveis complicações.

Atualmente, existe a plataforma Brainly, onde uma pessoa pode publicar questões para outras pessoas responderem no formato de fórum. A Brainly foi fundada em 2009, logo em 2015 já possuía 40 milhões de usuários em 35 países, isto indica que os investimentos em aplicações desse nicho têm taxa alta de conversão de usuários. De acordo com os criadores, o sucesso dessa rede social educativa deve-se ao fato de que os alunos enfrentam problemas similares e a Brainly oferece como solução respostas corretas e quase imediatas (QUAINO, 2015). Um problema causado por essa plataforma é que ela pode ser usada apenas para copiar respostas prontas, obstruindo o aprendizado.

O diferencial proposto para a implementação do aplicativo é que ele deve estimular a comunicação direta entre duas ou mais pessoas à fim e solucionar os problemas conjuntamente. Os alunos poderão encontrar outras pessoas que podem responder a suas questões através de um dos meios disponíveis entre ambos e visíveis no aplicativo, como Skype, Hangouts ou chat, dessa forma os alunos que possuem determinado conhecimento serão rastreáveis por quem precisa de ajuda, e acessíveis devido a visibilidade que o aplicativo deve causar para os usuários.

4 OBJETIVOS

1.1 OBJETIVO GERAL

O objetivo desse trabalho é desenvolver o aplicativo Helpin para estimular o aprendizado colaborativo.

1.2 OBJETIVOS ESPECÍFICOS

- Conceituar o aprendizado colaborativo e as tecnologias no ensino;
- Conceituar as Tecnologias de Informação e Comunicação (TIC);
- Descrever as tecnologias e processos de desenvolvimento;
- Especificar os requisitos do Produto Mínimo Viável (PMV);
- Elaborar o protótipo do aplicativo;
- Descrever a estratégia de implementação;
- Apresentar os resultados obtidos com o uso do aplicativo.

5 PROBLEMATIZAÇÃO

Quais as principais dificuldades encontradas no processo de aprendizado dos alunos da Faculdade de Tecnologia de Itapetininga?

6 HIPÓTESES

A hipótese levantada na pesquisa é de que os alunos enfrentam vários obstáculos. Dentre as principais dificuldades que permeiam o aprendizado dos alunos estão a falta de troca de informações com pessoas que entendem dos assuntos em que os alunos têm dificuldade a fim de suprir suas dúvidas quando há impedimentos geográficos, climáticos, em função de suas ocupações ou quaisquer outras variantes que desmotivam ou impedem os alunos de ir em busca de monitores das matérias em que necessitam de ajuda.

7 METODOLOGIA

A pesquisa, ora em projeto, pode ser definida como uma pesquisa aplicada, pois seu objetivo é gerar conhecimentos para aplicação prática a fim de solucionar problemas específicos e imediatos (ROSA, 2018).

Os dados serão inicialmente coletados por meio da pesquisa bibliográfica em livros, periódicos e revistas, a partir dos quais serão buscados os temas sobre aprendizado colaborativo.

Para o design do aplicativo será elaborado com conceitos UI (*User Interface*) será analisada a experiência do usuário com o aplicativo para dispositivos móveis. Logo será desenvolvido um protótipo de alta fidelidade por meio da ferramenta de prototipagem Adobe XD, tal que servirá de base para o desenvolvimento do aplicativo.

No que tange aspectos de implementação *frontend* do aplicativo serão utilizados a linguagem de programação ECMAScript com a biblioteca *React Native*,

responsável por converter o código ECMAScript em código nativo para as plataformas Android e IOS. A metodologia de organização adotada para o código de folhas de estilos será RSCSS e a arquitetura de pastas ITCSS, visando gerar uma aplicação limpa e organizada. Para armazenamento dos dados será implementado o SGBD Cloud Firestore da plataforma Firebase. No *backend* também será utilizada a linguagem de programação ECMAScript a partir da criação de *cloud functions*, tais que serão usadas para realizar todo o controle por traz da aplicação.

Para o planejamento das *sprints* e organização de tarefas é usada a ferramenta online ClickUp onde é possível gerenciar quadros e cartões para uso de metodologias ágeis como o SCRUM 4 Kanban. No levantamento de requisitos, foram realizadas reflexões referentes as necessidades do aluno e como a aplicação poderia ser mais bem-disposta aos instrutores.

8 REVISÃO DA LITERATURA

7.1 APRENDIZAGEM COLABORATIVA E A ABORDAGEM SOCIAL

A aprendizagem colaborativa é uma estratégia de ensino onde o conhecimento é resultante de um consenso entre as pessoas que o constroem trabalhando juntas, direta ou indiretamente, sem distinções hierárquicas entre os membros do grupo. Essa estratégia promove o aprendizado através da aculturação, isto é, quando os membros se adaptam às culturas uns dos outros (TORRES; ALCANTARA; IRALA, 2014).

A evolução das Tecnologias de Informação e Comunicação (TIC) utilizadas no ambiente escolar possibilitaram a flexibilização de horários, reconhecida como benefício para os estudantes, visto que esses poderiam sanar dúvidas ou estudar em horários apropriados às suas rotinas. As tecnologias de ensino-aprendizagem à distância impulsionam o desenvolvimento de ideias, rompendo a hegemonia das disciplinas e promovendo a educação interdisciplinar por meio da troca de conhecimentos (BASSO, 2008).

Segundo Moura e Oliveira (2015), a maior dificuldade de introduzir tecnologia no ensino é que o professor é apontado como o detentor de todo conhecimento. Todavia, o professor precisa ser capaz de reconhecer as diferentes formas de pensar e evitar a imposição de suas ideias. A incorporação das TIC nos ambientes escolares deve promover ações educativas que instigam o educando a ver o mundo além da sala de aula, respeitando os pensamentos e princípios do outro.

7.2 AS TECNOLOGIAS E O DESENVOLVIMENTO DE SOFTWARE

A linguagem de programação definida para o desenvolvimento do aplicativo foi JavaScript, pois essa participa de todo o processo de desenvolvimento *frontend* e *backend*, sendo “uma linguagem leve, interpretada e baseada em objetos com funções de primeira classe, mais conhecida como a linguagem de script para páginas Web, mas usada também em vários outros ambientes sem browser, tais como node.js” (MOZILLA e COLABORADORES, 2019).

Segundo a fundação NODE.JS (2019), Node.js é uma plataforma em JavaScript desenhada para construir aplicações escaláveis em rede, que utiliza um processo único criando uma nova *thread* para cada requisição de modo assíncrono, ou seja, não é necessário um processo ser encerrado para dar início a outro, isso é conhecido como *non-blocking I/O*, proporcionando muito mais performance utilizando menos processamento do servidor.

O Cloud Firestore utilizado nesse projeto como serviço de armazenamento escalável, “é um banco de dados flexível e escalável para desenvolvimento em aplicações web, mobile de servidores do Firebase e Google Cloud Platform” (GOOGLE DEVELOPERS, 2019).

Na etapa do desenvolvimento do *frontend* da aplicação foi definido o *framework* em JavaScript conhecido como React Native, que torna possível o desenvolvimento de aplicativos mobile para Android e iOS simultaneamente, reaproveitando o mesmo código por meio de dois núcleos, um para o Android e outro para o iOS. Após o código ser compilado para um aplicativo base por meio de uma ponte de duas vias, os

códigos são gerados para ambas as plataformas. Isso é possível por meio de um *bundle* conhecido como Metro (ZAGUINI, 2018).

Toda a estrutura do React é baseada em componentes, o que promove a reutilização de código à partir da importação de componentes e passagem de propriedades programadas, logo o conjunto de uma série de componentes forma uma tela do aplicativo (ZAGUINI, 2018).

Para facilitar o uso dos componentes e melhorar a organização do estado dos componentes faz-se necessário a utilização de uma biblioteca para organização dos dados de forma global, conhecida como Redux. A definição de “Redux é um controlador de estados previsível para aplicações Javascript” (ABRAMOV D., 2015). Essa biblioteca também auxilia a centralização da parte lógica do aplicativo, como no consumo de API’s e na comunicação com a Cloud Firestore.

A padronização de código fica sob responsabilidade do *linter* de JavaScript denominado ESLint, no qual foi estabelecido o padrão de código da empresa Airbnb. O *linter* deve apontar cada linha de código não padronizado, a fim de garantir a manutenibilidade do código (ESLINT, 2019).

7.3 AS TECNOLOGIAS DE VERSIONAMENTO DE CÓDIGO

A tecnologia para versionamento de código utilizada no projeto é o GIT, “uma ferramenta de código aberto distribuída de versionamento de código feita para lidar desde pequenos até os mais complexos projetos com velocidade e eficiência” (GIT, 2019). O GitHub é a plataforma escolhida nesse projeto para o gerenciamento dos repositórios GIT online.

O processo de versionamento adotado foi o GIT Flow, que padroniza o desenvolvimento baseado na criação de ramos no repositório, sendo eles *feature*, *bugfix*, *release*, *hotfix*, *develop* e *master*, a seguir cada uma dessas etapas serão descritas e será possível visualizar graficamente o fluxo do processo na Figura 1 (MOTA, 2019).

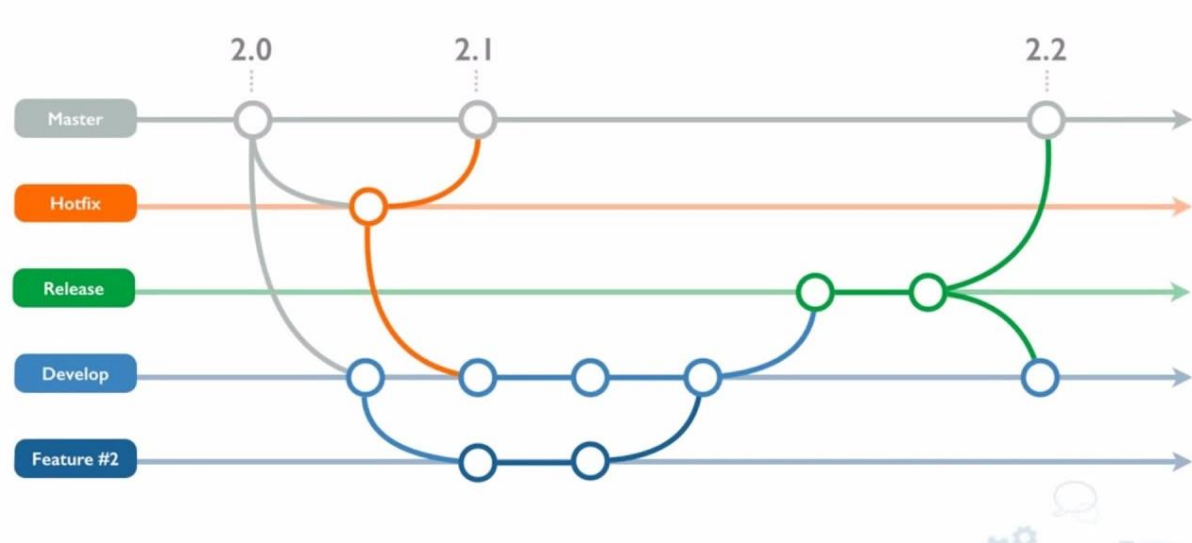


Figura 1: Exemplo do fluxo de desenvolvimento com a padronização do GitFlow – Fonte: <<https://www.youtube.com/watch?v=gLWSJXBbJuE>>

- **feature** é o prefixo utilizado para uma nova funcionalidade em desenvolvimento;
- **develop** é o ramo que contém todas as funcionalidades que já foram validadas e testadas isoladamente nos ramos *feature*, mas que ainda não se encontram em ambiente de produção;
- **bugfix** são reservados para correções após os ramos de *feature* já terem sido mesclados com o ramo *develop*;
- **release** é o prefixo e etapa onde é gerado uma versão destinada a produção, mas ainda em processo de homologação, podendo ser feitas alterações antes de ser enviada para o ramo de produção.
- **master** é o ramo que contém o código que se encontra nos servidores de produção;
- **hotfix** é o prefixo usado nos ramos onde são adicionadas correções que de problemas críticos que se encontram nos servidores de produção, ao finalizar o *hotfix*, a correção é enviada para o ramo de produção e para o ramo de desenvolvimento ao mesmo tempo, *master* e *develop* respectivamente;

O padrão de ramificações do GIT Flow segue o padrão de versionamento semântico, a técnica que define que as versões devem seguir o padrão x.y.z, onde x

é uma versão principal, y são segundas implementações de uma versão, como melhorias, e z são correções (WERNER, 2019).

Para a padronização dos *commits* entre os membros da equipe de desenvolvimento é utilizado a ferramenta Commitzen, que gerencia o padrão de escrita de cada *commit*, separando por tipo, como um *fix* para correção de algum problema, *feature* para *commits* que se tratam da implementação de uma funcionalidade, dentre outras nomenclaturas técnicas à fim de aperfeiçoar o desenvolvimento da aplicação (COMMITZEN, 2019).

7.4O PROCESSO DE DOCUMENTAÇÃO DE SOFTWARE

A UML (*Unified Modeling Language*), é a principal linguagem utilizada para a documentação de artefatos de software, consiste em notações gráficas padronizadas que são amplamente utilizadas no levantamento de requisitos, diagramação de comportamentos e modelagem de estruturas no desenvolvimento de software (MELO; MENDONÇA; DIAS, 2019).

O diagrama de caso de uso da UML é utilizado para representar os requisitos funcionais e suas relações com os envolvidos ao caso de uso em questão. Esse modelo auxilia no entendimento dos requisitos funcionais do sistema pela equipe de desenvolvimento e pode ser entendido pelos clientes ou outros envolvidos com o sistema, conhecidos como stakeholders (JACOBSON; RUMBAUGH; BOOCH, 2006).

O diagrama de componentes da UML é utilizado para diagramar a estrutura de como os códigos são organizados em sua programação, possibilitando uma visão mais clara de todos os componentes de software, podendo ser utilizada no momento em que a equipe de desenvolvimento deseja realizar o levantamento de ativos de software, isto é, aquilo que já existe desenvolvido e que pode ser reaproveitado. A UML também contém o diagrama de atividades, utilizado para modelar o fluxograma da execução de uma atividade, representando as variações condicionais que podem seguir atividades divergentes de acordo com as condições atendidas (JACOBSON; RUMBAUGH; BOOCH, 2006).

7.5A GESTÃO DE PROJETO

A metodologia Scrum descreve que um projeto deve ser dividido em partes, denominadas *sprints*, que deve ter de duas a quatro semanas de duração. As sprints são divididas em pequenas tarefas para que possam ser executadas pelos membros do time de desenvolvimento (SCRUM, 2019).

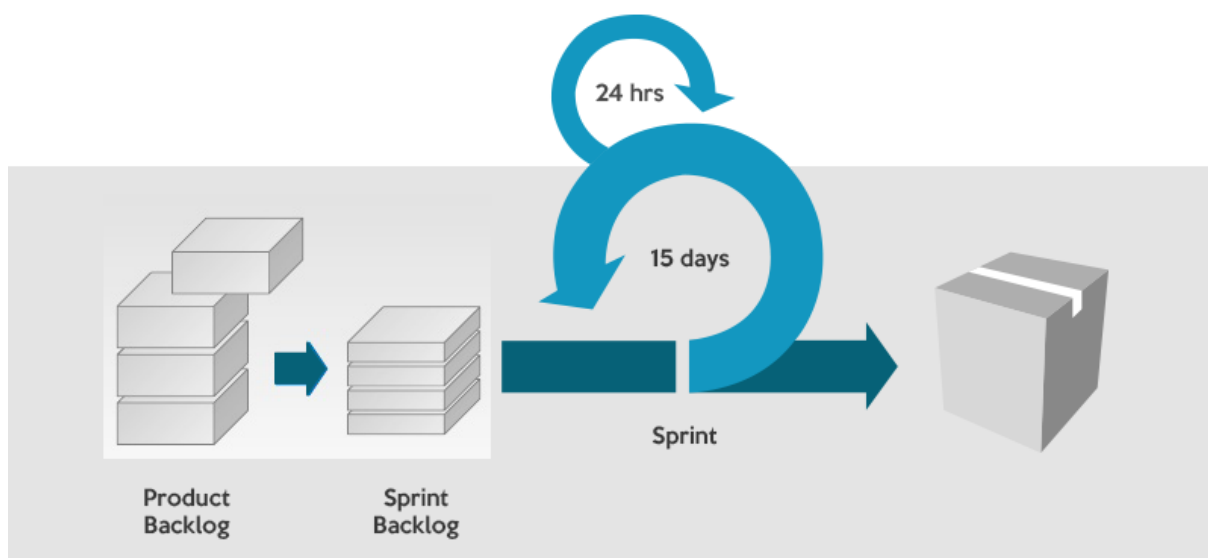


Figura 2: Exemplo do fluxo de desenvolvimento com SCRUM - fonte: <<https://luis-goncalves.com/pt-pt/o-que-e-scrum/>>

O ciclo do Scrum inicia-se na etapa de *Product Backlog*, onde são contidos os itens elicitados em entrevistas, análise de documentos e na observação do sistema de análise em questão. Em seguida, na etapa de *Sprint Backlog* é feito o planejamento de quais funcionalidades devem ser desenvolvidas. Nesse momento é realizado um processo denominado *planning poker*, onde todos os integrantes da equipe se juntam e juntos definem os prazos para a conclusão de cada *sprint* de acordo com sua dificuldade (SCRUM, 2019).

Após o início de uma sprint a equipe faz uma reunião diária conhecida como *daily meeting* onde cada membro fala das tarefas que foram concluídas, quais as dificuldades encontradas e o que irão fazer no próximo dia, assim todos os membros ficam atualizados em relação ao que os outros estão fazendo e podem oferecer ajuda a outros membros, se necessário. Após a finalização de uma *sprint*, acontece a

revisão e testes gerais para a validando das tarefas, sendo possível entregar a maior qualidade possível na entrega da nova funcionalidade da aplicação (DRUMOND, 2019).

10 DESENVOLVIMENTO

Antes de dar início às atividades de desenvolvimento, foram realizadas seções de *brainstorming* entre os membros da equipe com objetivo de escolher as melhores ideias e definir as que podem ser implementadas. Durante a análise das ideias e das restrições de prazo e recursos disponíveis para o desenvolvimento, foi proposto o desenvolvimento de um módulo de fórum em que um aluno da Fatec poderá realizar seu cadastro, fazer perguntas e responder dúvidas de outros alunos.

As funcionalidades do módulo de fórum foram documentadas utilizando as convenções de requisitos funcionais e requisitos não funcionais associados da engenharia de software. Cada tabela disponíveis no tópico 8.1 representa um requisito funcional, que contempla uma descrição e seus requisitos não funcionais associados.

8.1 LEVANTAMENTO DE REQUISITOS

RF-001 Registrar-se			
O aluno deverá preencher um formulário de cadastro para ingressar no aplicativo.			
REQUISITOS NÃO FUNCIONAIS ASSOCIADOS			
RNF	Restrição	Tipo	Classificação
001.1	O aluno deverá informar um e-mail válido.	Segurança	Obrigatório
001.2	O aluno deve inserir um e-mail e uma senha.	Padrão	Obrigatório

RF-002 Realizar login			
O aluno poderá autenticar-se no aplicativo.			
REQUISITOS NÃO FUNCIONAIS ASSOCIADOS			
RNF	Restrição	Tipo	Classificação

002.1	Caso o aluno informe um e-mail inválido, o aplicativo deve informá-lo.	Usabilidade	Obrigatório
002.2	O aluno deve inserir um e-mail e uma senha.	Segurança	Obrigatório

RF-003 Listar cursos			
O aluno poderá visualizar as opções de cursos para poder selecioná-los.			
REQUISITOS NÃO FUNCIONAIS ASSOCIADOS			
RNF	Restrição	Tipo	Classificação
003.1	Os cursos devem ser exibidos com cores distintas.	Usabilidade	Obrigatório
003.2	A exibição dos cursos deve ser em cartões.	Usabilidade	Obrigatório

RF-004 Listar perguntas			
O aluno poderá visualizar as perguntas feitas por ele e outros alunos.			
REQUISITOS NÃO FUNCIONAIS ASSOCIADOS			
RNF	Restrição	Tipo	Classificação
004.1	O aplicativo deve exibir a situação ou <i>status</i> da pergunta na listagem.	Padrão	Obrigatório
004.2	O aplicativo deve exibir o curso em que a pergunta se enquadra e a data de publicação da pergunta.	Usabilidade	Obrigatório
004.3	O aplicativo deve exibir o título da pergunta em destaque.	Usabilidade	Obrigatório

RF-005 Fazer pergunta			
O aluno poderá fazer perguntas no aplicativo.			
REQUISITOS NÃO FUNCIONAIS ASSOCIADOS			
RNF	Restrição	Tipo	Classificação
005.1	O aluno deve inserir o título da dúvida.	Padrão	Obrigatório
005.2	O aluno deve preencher a descrição da dúvida.	Padrão	Obrigatório
005.3	O aluno poderá adicionar imagens.	Usabilidade	Opcional
005.4	O aluno deverá pressionar um botão para salvar as informações.	Usabilidade	Obrigatório
005.5	O aplicativo deverá exibir uma confirmação após salvar a pergunta.	Usabilidade	Obrigatório

RF-006 Responder pergunta			
O aluno poderá responder a perguntas no aplicativo.			
REQUISITOS NÃO FUNCIONAIS ASSOCIADOS			
RNF	Restrição	Tipo	Classificação
006.1	O aluno deverá inserir uma resposta em um campo de texto.	Padrão	Obrigatório
006.2	O aluno poderá adicionar imagens para complementar a resposta.	Padrão	Opcional

006.3	O aluno deverá clicar em um botão para enviar a resposta.	Usabilidade	Obrigatório
-------	---	-------------	-------------

RF-007 Exibir detalhes da pergunta

O aluno poderá selecionar uma pergunta para exibir seus detalhes.

REQUISITOS NÃO FUNCIONAIS ASSOCIADOS

RNF	Restrição	Tipo	Classificação
007.1	O aplicativo deve exibir o título da pergunta em destaque.	Padrão	Obrigatório
007.2	O aplicativo deve exibir as imagens salvas com a pergunta.	Padrão	Opcional
007.3	O aplicativo deve exibir a data de publicação da pergunta.	Usabilidade	Obrigatório
007.4	O aluno poderá clicar em um botão para responder à pergunta.	Padrão	Obrigatório

RF-008 Editar configurações da conta

O aluno poderá alterar todos os dados de configurações da conta detalhados.

REQUISITOS NÃO FUNCIONAIS ASSOCIADOS

RNF	Restrição	Tipo	Classificação
008.1	A tela usada para exibição também deve ser usada para cadastro e edição dos dados da conta do usuário.	Usabilidade	Obrigatório

008.2	O aplicativo deve exibir os campos nome, curso, número de WhatsApp e conta do Skype.	Padrão	Opcional
008.3	O aplicativo deve exibir um botão para o aluno salvar as informações.	Usabilidade	Obrigatório

RF-009 Cadastrar conhecimentos			
O aluno poderá cadastrar seus conhecimentos.			
REQUISITOS NÃO FUNCIONAIS ASSOCIADOS			
RNF	Restrição	Tipo	Classificação
009.1	O aluno deverá inserir um título.	Padrão	Obrigatório
009.2	O aluno deve pressionar um botão para adicionar conhecimento.	Usabilidade	Obrigatório
009.3	O aplicativo deve exibir um botão para o aluno salvar as alterações.	Usabilidade	Obrigatório

RF-010 Remover conhecimento			
O aluno poderá remover seus conhecimentos cadastrados.			
REQUISITOS NÃO FUNCIONAIS ASSOCIADOS			
RNF	Restrição	Tipo	Classificação
010.1	O aplicativo deve exibir um botão para remover o conhecimento na listagem.	Padrão	Obrigatório

010.2	O aplicativo não deve solicitar confirmação de exclusão.	Padrão	Obrigatório
-------	--	--------	-------------

RF-011 Exibir alunos no *chat*

O aluno poderá visualizar outros alunos que usam o aplicativo pelo chat.

REQUISITOS NÃO FUNCIONAIS ASSOCIADOS

RNF	Restrição	Tipo	Classificação
011.1	O aplicativo deve exibir o nome e o curso dos alunos.	Padrão	Obrigatório
011.2	O aplicativo deve exibir a situação do aluno (<i>online</i> ou <i>offline</i>) com cores verde e vermelho respectivamente.	Padrão	Obrigatório
011.3	O aplicativo deve exibir um ícone que contemple as iniciais do nome do aluno.	Padrão	Obrigatório

RF-012 Enviar mensagem

O aluno poderá enviar mensagens para outros aluno.

REQUISITOS NÃO FUNCIONAIS ASSOCIADOS

RNF	Restrição	Tipo	Classificação
012.1	O aluno deve preencher o campo da mensagem.	Padrão	Obrigatório
012.2	O aplicativo deve exibir uma legenda no campo para o aluno inserir sua mensagem.	Padrão	Obrigatório

012.3	O aplicativo deve exibir um botão para o aluno enviar sua mensagem.	Padrão	Obrigatório
012.4	O aplicativo deve limpar o campo de mensagem após o aluno enviá-la.	Usabilidade	Obrigatório

RF-013 Receber mensagem			
O aluno poderá enviar mensagens para outros alunos.			
REQUISITOS NÃO FUNCIONAIS ASSOCIADOS			
RNF	Restrição	Tipo	Classificação
013.1	O aplicativo deve exibir as mensagens de ambos os usuários no histórico da conversa.	Padrão	Obrigatório
013.2	O aplicativo deve exibir a situação ou <i>status</i> do destinatário.	Padrão	Obrigatório
013.3	O aplicativo deve exibir um ícone com as iniciais do nome dos usuários.	Padrão	Obrigatório

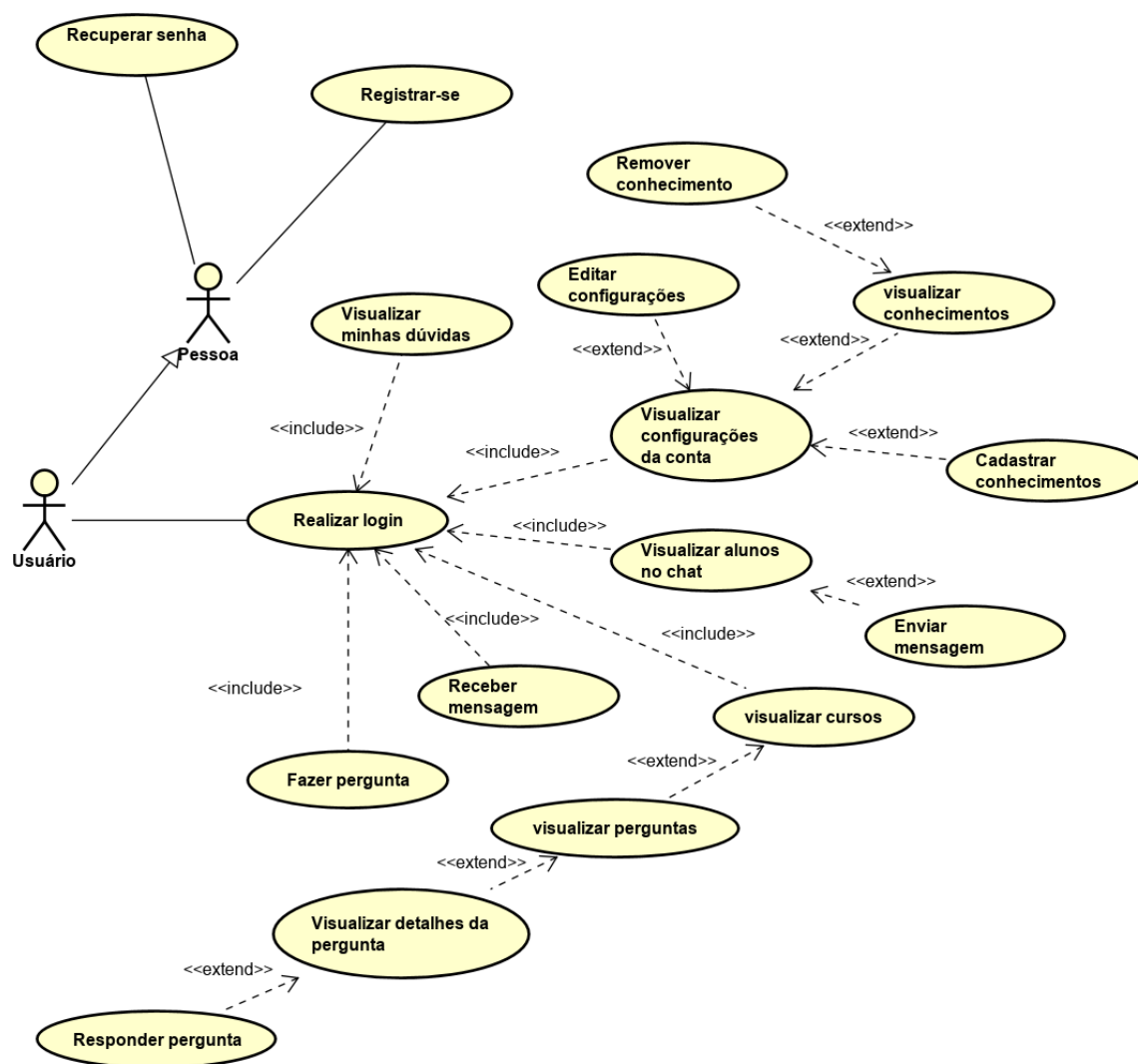
RF-015 Listar minhas dúvidas			
O aluno poderá visualizar todas as suas dúvidas já postadas			
REQUISITOS NÃO FUNCIONAIS ASSOCIADOS			
RNF	Restrição	Tipo	Classificação
015.1	O aplicativo deve exibir as últimas dúvidas postadas na página principal.	Padrão	Obrigatório

015.2	O aplicativo deve exibir todas as dúvidas já postadas pelo usuário quando ele selecionar a opção “exibir todas” na página principal	Padrão	Obrigatório
-------	---	--------	-------------

RF-016 Recuperar senha			
O aluno poderá recuperar sua senha quando esquece-lá			
REQUISITOS NÃO FUNCIONAIS ASSOCIADOS			
RNF	Restrição	Tipo	Classificação
016.1	O aplicativo deve enviar um link para redefinir sua senha para o seu e-mail.	Padrão	Obrigatório
016.2	O aluno deverá inserir sua nova senha ao acessar o link para recuperação	Padrão	Obrigatório
016.3	O aplicativo não deve permitir o usuário redefinir a senha se o <i>token</i> for inválido.	Segurança	Obrigatório

8.2 OS DIAGRAMAS DA UML

Os requisitos do aplicativo podem ser representados de maneira gráfica utilizando o diagrama de caso de uso da UML. São expostos da **Figura X** os casos de uso do módulo de fórum para os *stakeholders* terem a visão geral das atividades que usuários ou pessoas não autenticadas poderão realizar no aplicativo.

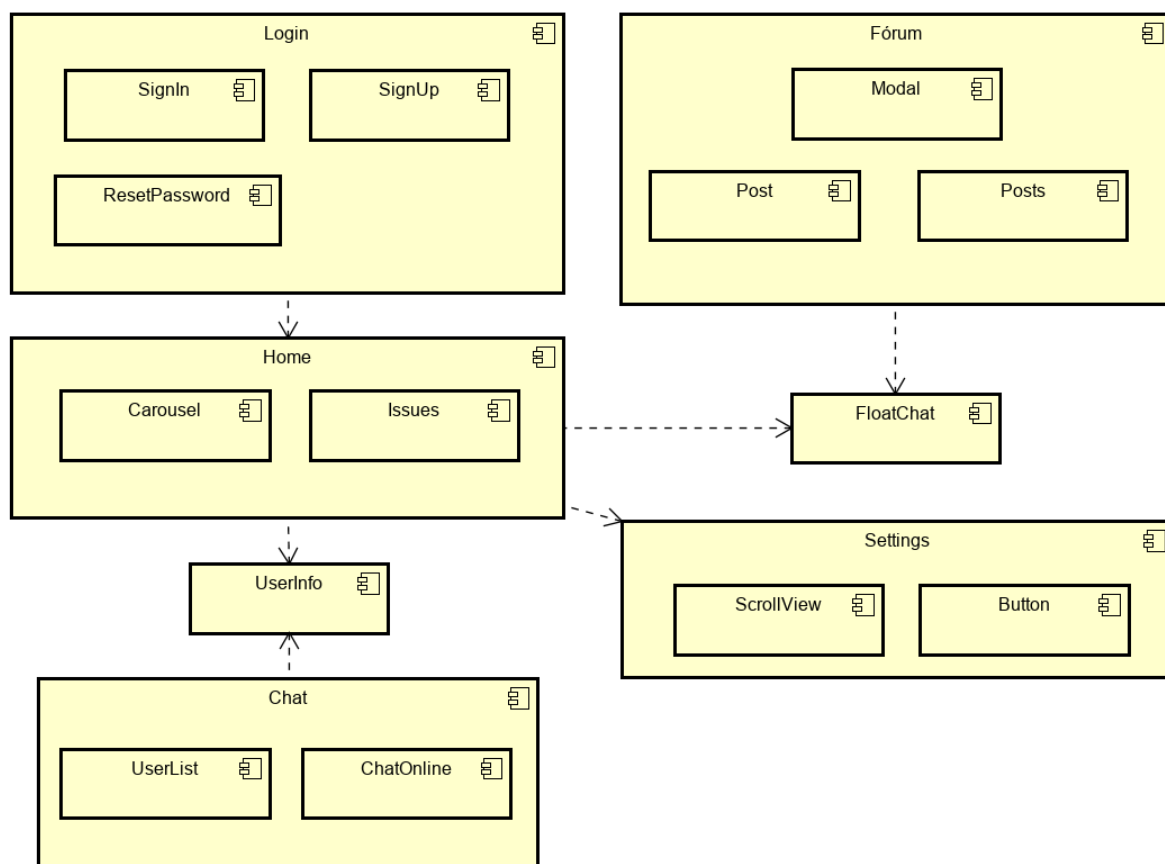


A biblioteca usada no desenvolvimento do aplicativo React Native funciona com o paradigma de componentes, onde tudo deve ser um componente, por essa razão foi utilizado o diagrama de componentes da UML, representado na **Figura X**, facilitando a compreensão pelos desenvolvedores acerca das dependências entre cada componente e servindo como um mapa de ativos para reaproveitamento de

código

futuramente.

cmp Components



No diagrama (Figura X) foram representados módulos que contemplam componentes, o módulo login é composto por outros três componentes, o SignIn, que representa os componentes usados na tela para fazer *login*, o SignUp, componentes da tela para registrar-se, e o ResetPassword, que representa os componentes utilizados na tela de redefinir senha.

Logo é representado o módulo Home, que representa os componentes utilizados na tela principal, composto pelos componentes UserInfo, informações do usuário, Issues, as postagens do usuário, o Carousel, carrossel de eventos e o botão do chat, denominado FloatChat. Fazendo uso do reaproveitamento de código, os componentes do módulo Home podem acessar todos os outros componentes do módulo de configurações.

O módulo de configurações (Settings) é composto por botões e por uma lista para edição de usuários, também existe o módulo do fórum composto pela lista de postagens (Posts) contendo as publicações (Post), a tela para a criação da postagem

(Modal), o botão flutuante do chat (FloatChat) e por fim o módulo do chat composto pelas informações do usuário no componente (UserInfo), uma lista de usuários (UserList) e o próprio bate-papo (ChatOnline).

8.3 PROTOTIPAGEM DO APLICATIVO

Durante a exposição do protótipo das telas, também serão apresentados os diagramas de atividade da UML, com o objetivo de explicar como o usuário poderá utilizar das funcionalidades do aplicativo.



Figura 3: Tela de cadastro

Na tela de cadastro (Figura X), de acordo com o diagrama de atividade “Registrar-se” (Figura X), o usuário deve informar os dados mínimos para o cadastro na plataforma, sendo eles e-mail e senha.

Foi adicionado um botão no campo de entrada para senha com a representação visual de um olho, que ao clicar sobre ele, o aplicativo revela a senha digitada para o usuário, tornando desnecessário a utilização de uma entrada de confirmação da senha e consequentemente o usuário irá digitar menos. Essa peculiaridade está disponível em todos os campos de entrada de senha na plataforma. Ao clicar em finalizar e concluir o seu cadastro, o usuário será redirecionado para a tela principal do aplicativo.

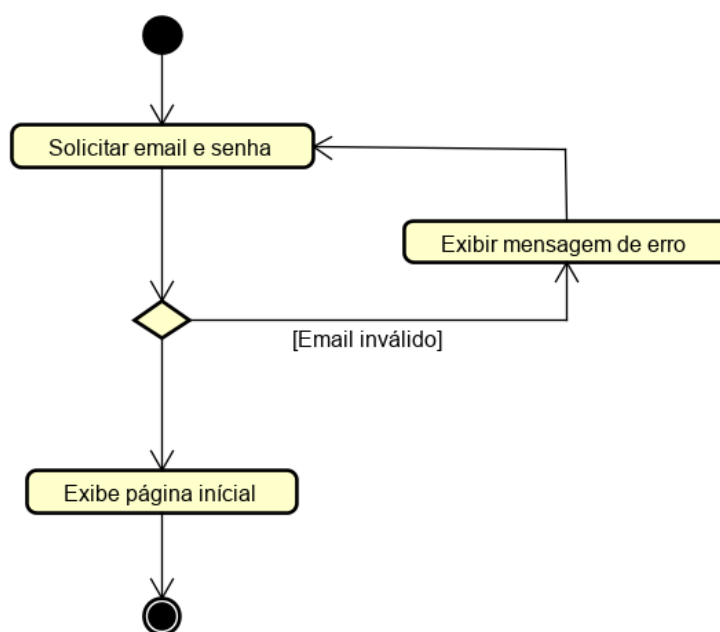


Figura 4: Diagrama de Atividade: Registrar-se

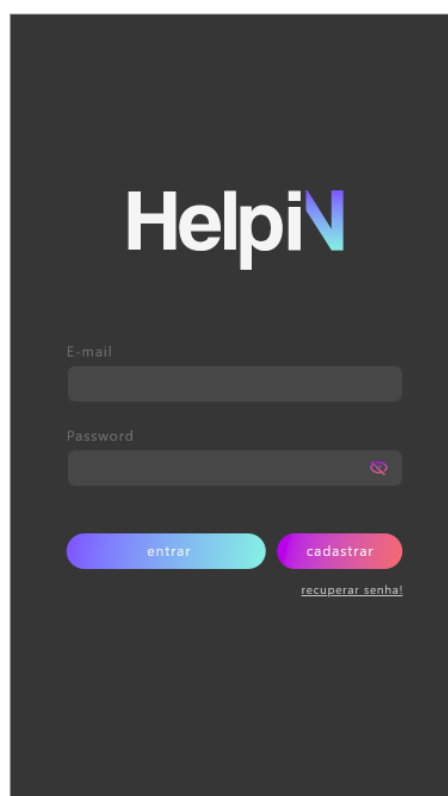


Figura 5: Tela de login

Na tela de login (**Figura X**), o usuário deverá inserir o usuário e senha do seu cadastro para prosseguir para a tela principal ou utilizar o botão cadastrar para

navegar até a tela de cadastro. Caso o usuário tenha esquecido a senha, foi implementado um botão para recuperação da senha, que enviará um e-mail com o link para recuperá-la ao usuário em questão.

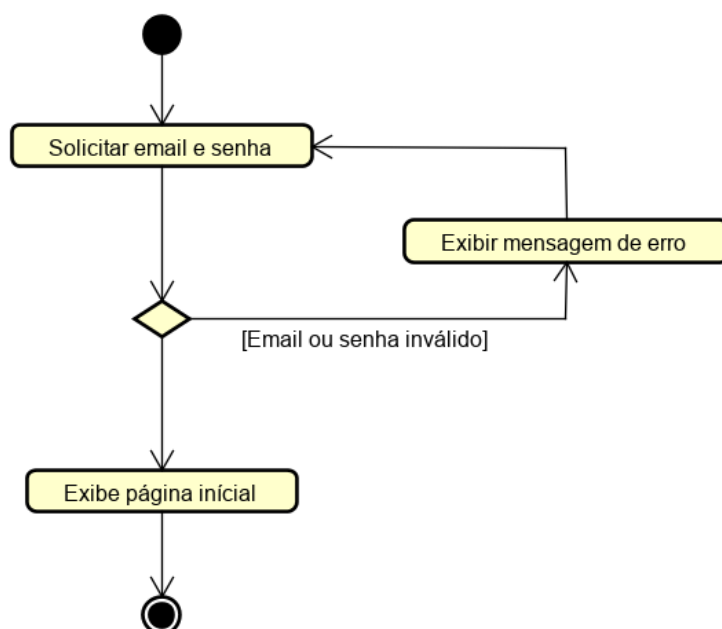


Figura 6: Diagrama de Atividade: Fazer login

Na tela principal (Figura X) é apresentado um resumo de todas as funcionalidades da plataforma, iniciando com a foto de perfil, nome e curso que usuário está matriculado, o ícone da engrenagem representa a ponte para a tela das configurações. Para mostrar os últimos eventos adicionados existe a representação em cartões agrupados por um *carousel*. Foi adicionado um botão em destaque mostrando os usuários online no *chat* quando o usuário está utilizando o aplicativo. Também existem dois botões que levam as principais funcionalidades da plataforma, sendo eles “ajudar”, respondendo perguntas feitas por outros usuários e “perguntar” onde o usuário autenticado poderá criar uma questão no fórum de aprendizado.

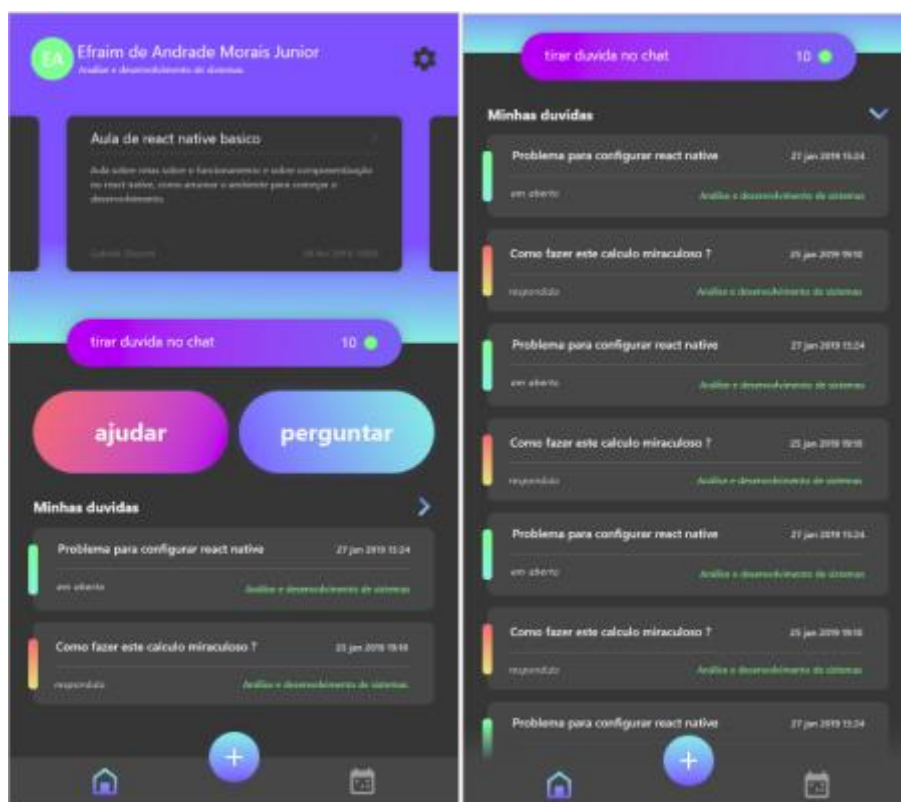


Figura 7: Tela principal e tela de listagem de conhecimentos

Após os botões de ações, existe a sessão de “minhas dúvidas” onde são exibidas as últimas postagens do usuário no aplicativo, representados com um cartão com informações breves da postagem como o título, data, curso relacionado e *status* ou situação, representado tanto em texto quanto em cores. A última informação adicionada nesta tela foi o menu de navegação que deve ajudar o usuário a navegar por toda a plataforma.

Na tela de configurações o usuário poderá completar seu perfil preenchendo dados pessoais como curso, redes sociais, foto de perfil e seus conhecimentos, como JavaScript, PHP, inglês, dentre outros. Com os conhecimentos cadastrados, outros usuários poderão ao realizar uma busca por determinado conhecimento, visualizar os usuários que detêm o conhecimento procurado.

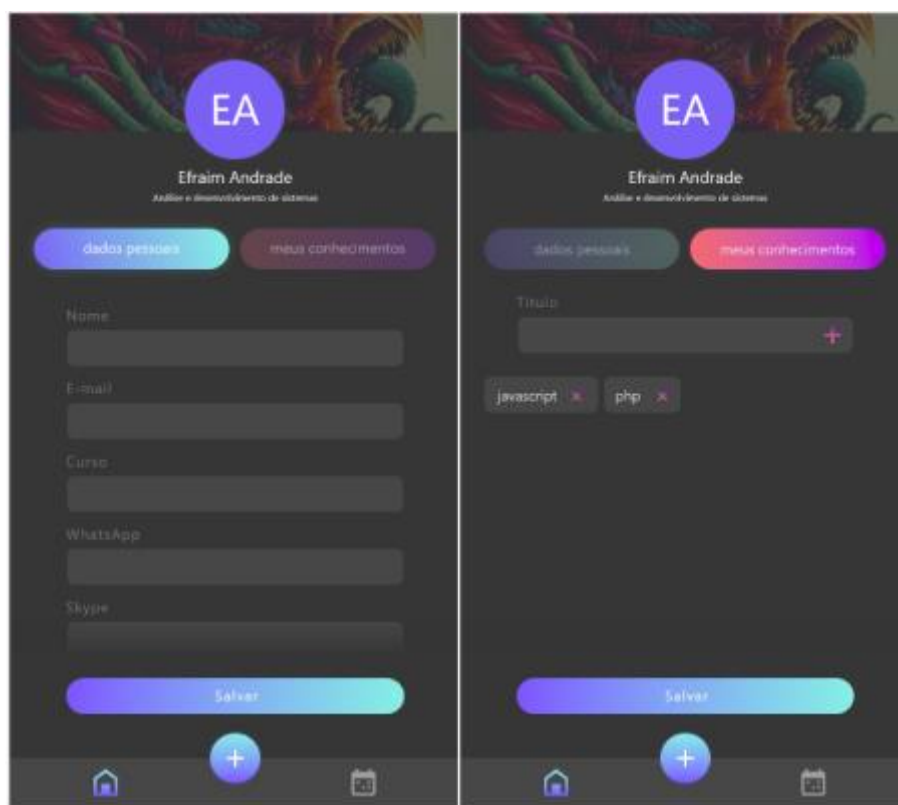


Figura 6: Tela de Configurações e tela de cadastro de conhecimento

Na tela de “ajudar” existe uma listagem com os cursos adicionados à plataforma onde eles são separados por categoria, os cursos são representados por suas respectivas cores para criar uma relação mais fácil de identificar ao visualizar uma postagem. Ao clicar em uma categoria, o usuário é redirecionado para a tela das postagens em aberto referente a categoria escolhida. Além disso, nesta tela existe a possibilidade de alterar a categoria por meio de um seletor.



Figura 7: Tela de ajudar: listagem de categorias e listagem de conhecimentos por categoria

Na tela de visualização de uma postagem são exibidas todas as informações da postagem, inicialmente o usuário que fez a postagem, então as informações expandidas e toda a descrição e conteúdo anexado do problema postado. Também é exibido a informação do status da postagem com a cor verde ou vermelha em questão e o curso escolhido ao lado da pergunta.

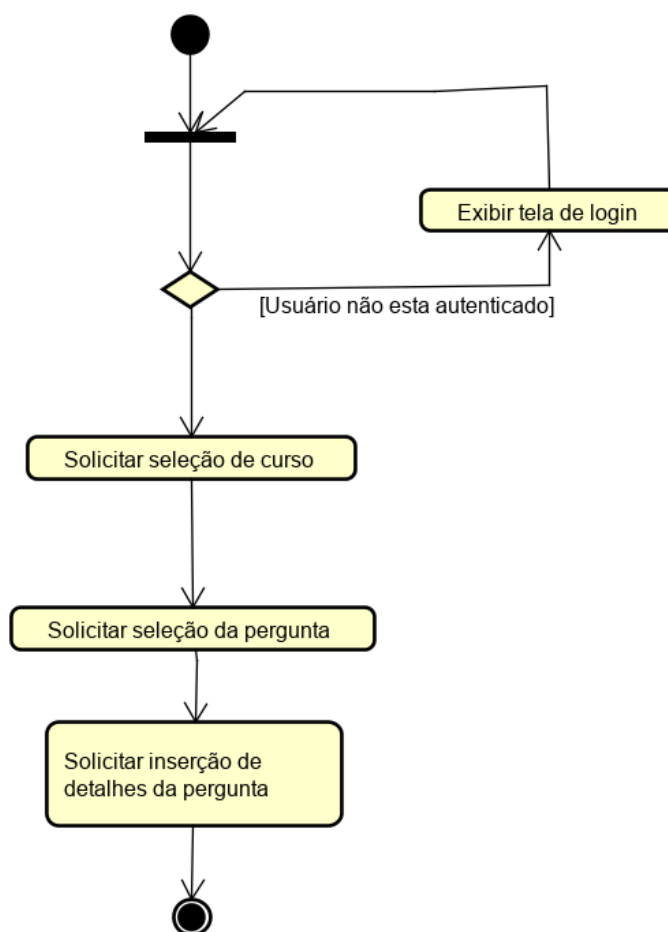


Figura 8: Diagrama de Atividade: Responder pergunta

Os usuários que desejarem responder à pergunta, conforme o diagrama de atividade “Responder pergunta” (Figura X), ao pressionar o botão “responder”, que está em destaque, o modal de resposta deve abrir e o usuário poderá escrever sua resposta e anexar arquivos ou imagens para ajudar na compreensão na resolução da dúvida, clicando em salvar, a resposta é lançada e vai estar logo abaixo da pergunta com a foto, nome do autor da postagem e a data de publicação.

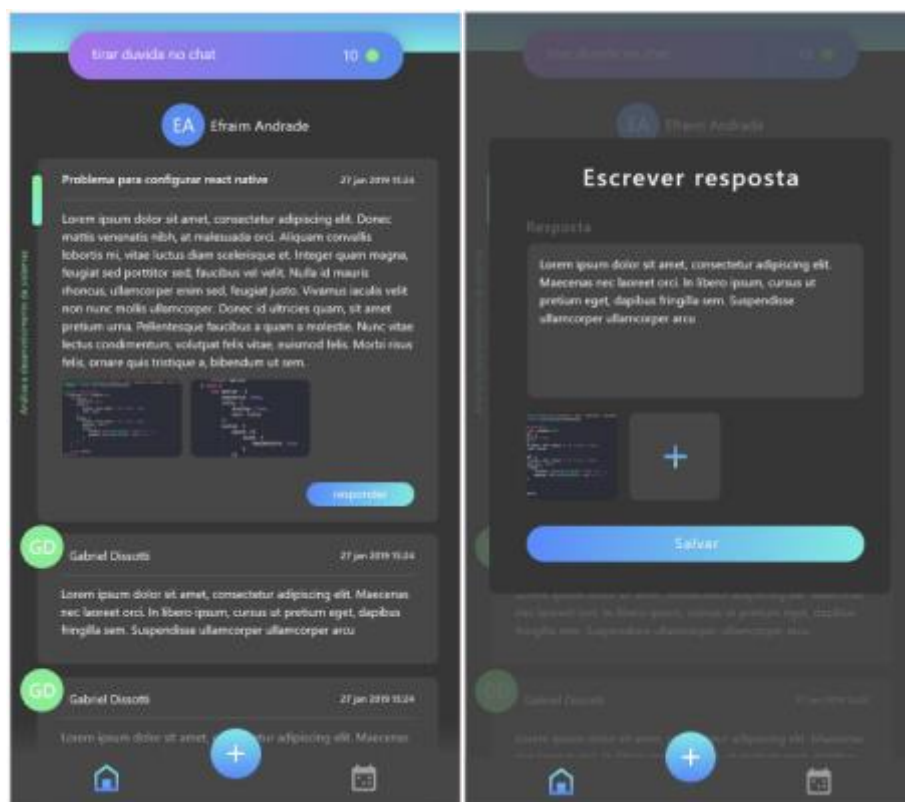


Figura 89: Tela de visualização de postagem e escrevendo uma resposta

Na tela de pergunta o usuário escolhe uma categoria para sua dúvida e o ícone de criação de postagem no menu torna-se um botão com gradiente voltado para roxo e vermelho, com a intenção de cancelar a criação.



Figura 9: Tela de perguntar: seleção de categoria

Ao avançar na criação da dúvida é apresentada ao usuário a tela em questão onde ele poderá inserir um título, uma descrição e anexar arquivos e fotos para uma melhor descrição de sua dúvida. Ao salvar a postagem, o usuário receberá a tela de sucesso como resposta de que a sua dúvida foi postada com sucesso no fórum.

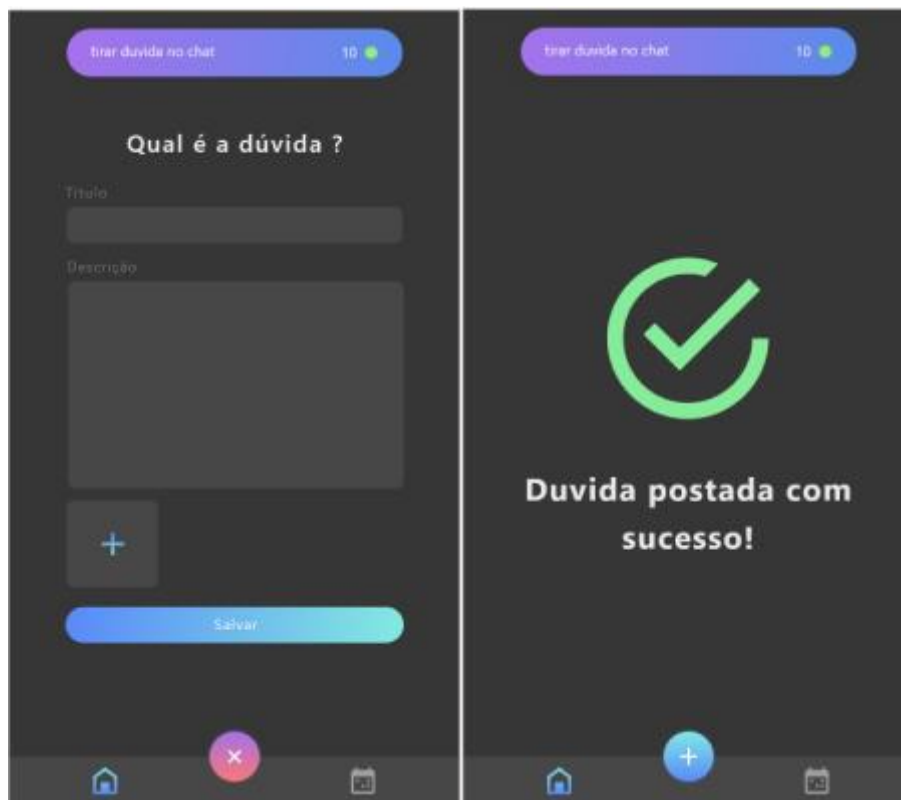


Figura 10: Tela de criação de dúvida e tela de confirmação de cadastro de dúvida

Nas telas do chat primeiramente existe a listagem de usuários com um campo de busca onde o usuário pode filtrar pelo nome, curso ou algum conhecimento. Então será exibido apenas os usuários que cumprirem com os requisitos do filtro. Ao clicar em algum usuário, a tela de chat é iniciada onde a interação em tempo real irá acontecer, por meio de um campo para a escrita das mensagens.

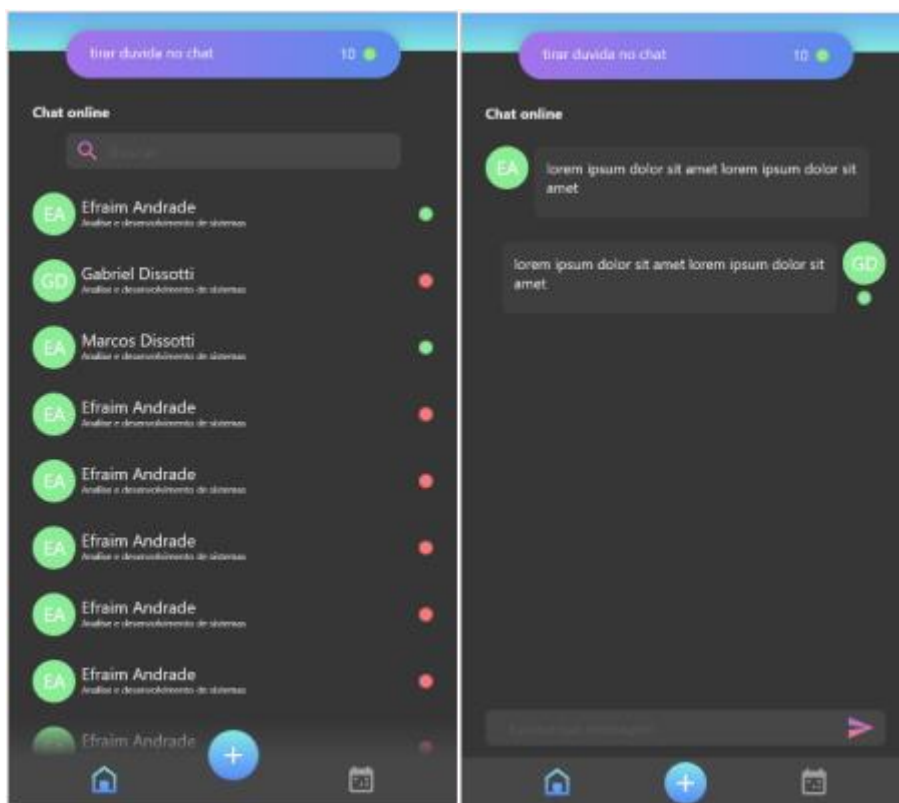


Figura 101: Tela de chat ou bate-papo e tela de conversação

11 CONSIDERAÇÕES FINAIS

De acordo com o objetivo do trabalho, foi iniciado o desenvolvimento do aplicativo que deve melhorar as condições para o aprendizado dos alunos, explicando processos e tecnologias utilizadas no mercado de trabalho. A utilização de processos de engenharia de software apresentadas aos discentes de Análise e Desenvolvimento de Sistemas pela Fatec junto a pesquisa com maior aprofundamento tem somado para a organização e desenvolvimento do projeto.

Espera-se que, após o lançamento do aplicativo por meio da Google Play Store, ocorra aceitação pelos alunos da Fatec, assim como ocorreu-se em apresentações anteriores, funcionando como prova de conceito para que este modelo acadêmico de aplicativo seja visto como forte no mercado e ajude toda a comunidade acadêmica, para que assim sejam incrementadas futuras funcionalidades.

As futuras implementações ponderadas até o momento são agendamento de videoaulas, aulas presenciais e talvez essa ferramenta possa ser utilizada para alguns eventos promovidos pela Fatec, como workshops, a semana de tecnologia ou excursões. Também está em análise uma versão web, em suma, os objetivos serão buscar melhorar a experiência do usuário constantemente e sempre que necessário inovar os meios disponíveis para o aprendizado de todas as pessoas.

REFERÊNCIAS

- ABRAMOV D., Redux. 2015. Disponível em: <<https://redux.js.org/introduction/getting-started>>. Acesso em: 13 mai. 2019.
- BASSO, M. D. O ambiente virtual no auxílio do ensino presencial. Brasília, 2008. Disponível em: <<http://www.ucb.br/sites/100/103/TCC/22008/MaxwellDavidBasso.pdf>>. Acesso em: 19 mai. 2019.
- COMMITZEN, 2019. Disponível em: <<https://github.com/commitizen/cz-cli>>. Acesso em: 20 mai. 2019.
- DRUMOND, C. O que é o scrum?. Disponível em: <<https://br.atlassian.com/agile/scrum>>. Acesso em 20 mai. 2019.
- ESLINT: The pluggable linting utility for JavaScript and JSX. Disponível em: <<https://eslint.org/>>. Acesso em 20 mai. 2019.
- GIT. Disponível em: <<https://git-scm.com/>>. Acesso em: 14 mai. 2019.
- GOOGLE, 2019, Cloud Firestore. Disponível em: <<https://firebase.google.com/docs/firestore>>. Acesso em 10 mai. 2019.
- BOOCH, G; RUMBAUGH, J; JACOBSON, I. **UML: guia do usuário**. Ed. Elsevier Brasil, 2006. cap. 20. p. 98-99.
- MOTA. F. J. Git Flow: Uma forma legal de organizar repositórios git. Disponível em: <<https://fjorgemota.com/git-flow-uma-forma-legal-de-organizar-repositorios-git/>>. Acesso em 20 mai. 2019.
- MELO, J. D; MENDONÇA, M. D; DIAS, W. R. A. Estudo de caso no contexto da Engenharia de Software: SGCOPEx. **Brazilian Journal of Development**, v. 5, n. 2, p. 1836-1851, 2019. Disponível em: <<http://www.brjd.com.br/index.php/BRJD/article/view/1186>>. Acesso em: 21 mai. 2019.
- MOURA, S. P; OLIVEIRA, C. TIC's na educação: A atualização das tecnologias de informação e comunicação na aprendizagem do aluno. Pedagogia em Ação. Belo Horizonte, 2015. Disponível em: <<http://periodicos.pucminas.br/index.php/pedagogiacao/article/view/11019/8864>>. Acesso em: 19 mai. 2019.
- MOZILLA e COLABORADORES, Javascript. Disponível em: <<https://developer.mozilla.org/pt-BR/docs/Web/JavaScript>>. Acesso em 10 mai. 2019.

NODE.JS FOUNDATION, NodeJS. Disponível em: <<https://nodejs.org/>>. Acesso em 9 mai. 2019.

QUAINO, L. Rede social educativa inicia 2015 com 40 milhões de usuários em 35 países. G1, Rio de Janeiro, 30 jan. 2015. Disponível em: <<http://g1.globo.com/tecnologia/noticia/2015/01/rede-social-educativa-inicia-2015-com-40-milhoes-de-usuarios-em-35-paises.html>>. Acesso em: 06 mar. 2019.

ROSA, M. N. L. M. O Programa de Iniciação Científica e seu impacto nas atividades de pesquisa da FACISB. **Manuscripta Médica**, São Paulo, 2018. Disponível em: <<https://facisb.edu.br/ojs/index.php/mm/article/view/14>>. Acesso em: 3 mar. 2019.

SCRUM, Scrum: metodologia ágil para gestão e planejamento de projetos. Disponível em: <<https://www.desenvolvimentoagil.com.br/scrum/>>. Acesso em: 14 mai. 2019.

TORRES, P. L; ALCANTARA, P. R; IRALA E. A. F. Grupos de Consenso: Uma Proposta de aprendizagem colaborativa para o processo de ensino-aprendizagem. **Revista Diálogo Educacional**, Curitiba, v. 4, n.13, p.129-145, set./dez. 2004. Disponível em: <<https://periodicos.pucpr.br/index.php/dialogoeducacional/article/view/7052/6932>>. Acesso em: 13 mai. 2019.

WERNER T. P., Versionamento Semântico 2.0.0, 2019. Disponível em: <<https://semver.org/lang/pt-BR/>>. Acesso em: 14 mai. 2019.

ZAGUINI L. F. Perguntas e respostas sobre React Native. 11 fev. 2018. Disponível em: <<https://medium.com/reactbrasil/perguntas-e-respostas-sobre-react-native-c56c4d8dff8>>. Acesso em: 13 mai. 2019.