



**UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS QUIXADÁ
CURSO DE GRADUAÇÃO EM ENGENHARIA DE SOFTWARE**

ISAAC JAMES MANGUEIRA DO NASCIMENTO

**DESENVOLVIMENTO DE UMA APLICAÇÃO MÓVEL PARA GERENCIAMENTO DE
ATIVIDADES COMPLEMENTARES EM INSTITUIÇÕES DE ENSINO SUPERIOR**

QUIXADÁ

2023

ISAAC JAMES MANGUEIRA DO NASCIMENTO

DESENVOLVIMENTO DE UMA APLICAÇÃO MÓVEL PARA GERENCIAMENTO DE
ATIVIDADES COMPLEMENTARES EM INSTITUIÇÕES DE ENSINO SUPERIOR

Trabalho de Conclusão de Curso apresentado
ao Curso de Graduação em Engenharia de
Software da Universidade Federal do Ceará,
como requisito parcial à obtenção do grau de
bacharel em Engenharia de Software.

Orientadora: Profª. Ma. Antônia Diana
Braga Nogueira

QUIXADÁ

2023

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas

Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

N195d Nascimento, Isaac James Mangueira do.

Desenvolvimento de uma aplicação móvel para gerenciamento de atividades complementares em instituições de ensino superior / Isaac James Mangueira do Nascimento. – 2023.

64 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Quixadá, Curso de Engenharia de Software, Quixadá, 2023.

Orientação: Profa. Ma. Antônia Diana Braga Nogueira.

1. Horas complementares. 2. Atividades complementares. 3. Engenharia de software. 4. Aplicativo móvel. 5. gerenciamento. I. Título.

CDD 005.1

ISAAC JAMES MANGUEIRA DO NASCIMENTO

DESENVOLVIMENTO DE UMA APLICAÇÃO MÓVEL PARA GERENCIAMENTO DE
ATIVIDADES COMPLEMENTARES EM INSTITUIÇÕES DE ENSINO SUPERIOR

Trabalho de Conclusão de Curso apresentado
ao Curso de Graduação em Engenharia de
Software da Universidade Federal do Ceará,
como requisito parcial à obtenção do grau de
bacharel em Engenharia de Software.

Aprovada em:____/____/_____

BANCA EXAMINADORA

Prof^a. Ma. Antônia Diana Braga
Nogueira (Orientadora)
Universidade Federal do Ceará (UFC)

Prof^a. Dr. Jeferson Kenedy Morais Vieira
Universidade Federal do Ceará (UFC)

Prof. Dr. Jefferson de Carvalho Silva
Universidade Federal do Ceará (UFC)

Dedico à Deus, minha família e meus verdadeiros amigos. Obrigado por tudo.

AGRADECIMENTOS

Agradeço primeiramente a Deus, por ter me dado saúde, esperança e sabedoria durante minha trajetória até aqui. Sem Ele e sem a fé que tenho, isso não seria possível.

À minha Mãe, Joelma Mangueira, por ter me criado tentando me dar o melhor que ela poderia me dar. Obrigado por nunca desistir de mim. Mãe, te amo.

À minha Vó, Maria Mateus, que deu à minha família base e suporte para sobrevivermos e vivermos, desde o início de nossas vidas. Obrigado por todas as orações. Afói, te amo.

Ao meu irmão, Diego Mangueira, por sempre ter um cuidado comigo, me dando conselhos sempre querendo o meu melhor. Irmão, sinto sua empatia comigo daqui. Te amo.

À minha cunhada, Genária e à minha sobrinha, Júlia Mangueira, obrigado por sempre torcerem por mim.

Ao meu pai, Josimar (Gutemboy), por ter me dado todo suporte durante toda a trajetória da minha educação. Pai, te amo. À minha Tia, Cláudia, obrigado por, lá atrás, no começo da minha vida como universitário na católica, ter acreditado em mim e me dado o suporte necessário para dar o start.

À minha psicóloga, Dra Laurilane Brandão, por me ouvir e me ajudar a me entender, aguçar minha sabedoria, superar traumas e me tornar uma pessoa melhor comigo mesmo e com os outros. Obrigado por me ajudar a me tornar um ser humano melhor.

Aos meus amigos que conheci durante essa trajetória: Jonas Deyvid, Isac, Rodrigo, Rosembergson, Matheus Rios, John que tornaram essa jornada inesquecível e ainda mais gratificante. Em especial, meus irmãos de coração, meu compadre Bruno Carvalho, por ter me dado a honra de sua amizade, por ter me dado suporte e parceria durante minha trajetória acadêmica e profissional. E ao Pedro, por me ajudar durante essa fase final da minha conclusão do curso. Obrigado pela sua amizade!

Aos meus amigos que conheci há quase 10 anos, que me influenciaram muito positivamente para o meu caráter e para iniciar a vida acadêmica na UFC Campus Quixadá. Em especial Thiago Pereira, Lucas Araújo, Sérgio Filho, Adail e Amaro César. Amaro, agradeço à você e à Deus por ter pulado naquele açude na Tapera e me salvar.

À minha orientadora, professora e mãe de curso, Diana Braga. Diana, obrigado por suas orientações no meu trabalho. Obrigado por seus conselhos e cuidado com nós, alunos da UFC. À Universidade Federal do Ceará Campus Quixadá e a seus docentes, por dar todo

o suporte necessário para a formação de seus alunos, tudo com excelência. A transformação transforma vidas, a minha é prova.

E por último, mas não menos importante, eu agradeço à mim. Por acreditar em mim, por ter me esforçado e feito acontecer, por ter focado nos momentos mais precisos, por manter a esperança viva de que tudo iria melhorar. E melhorou. O tempo é sábio, o tempo é ouro, o tempo é rei.

“O entusiasmo é a força da alma. Conserva-o e nunca te faltarão poder para conseguir o que desejas.”

(Napoleon Hill)

RESUMO

As atividades complementares desempenham um papel essencial na formação acadêmica, permitindo que os estudantes expandam seus conhecimentos além das disciplinas regulares. Essas atividades podem incluir participação em projetos de extensão, estágios, participação em projetos sociais, eventos, entre outras. Essas atividades são exigidas por algumas instituições de ensino superior, pois compõem o currículo acadêmico, e são importantes para a formação do aluno, pois possibilitam o desenvolvimento de habilidades e competências, além de proporcionar a vivência de situações reais de trabalho. Cada instituição pode ter suas estratégias tanto para incentivo quanto para o cumprimento, por parte dos alunos, das atividades complementares. Acontece que é comum os alunos terem dificuldades no cumprimento dessas atividades, pois não possuem um sistema que facilite o acompanhamento e o gerenciamento das mesmas. Diante do cenário global atual, onde a tecnologia móvel é cada vez mais presente, o desenvolvimento de um aplicação móvel para gerenciar tais atividades pode ser uma solução que amenizaria a problemática enfrentada pelos alunos no cumprimento e acompanhamento de tais atividades. Nesse contexto o objetivo do presente trabalho é desenvolver um aplicativo móvel para gerenciar as atividades complementares, que possibilite o acompanhamento e o gerenciamento de atividades complementares e atividades de extensão, de forma que o aluno possa ter um maior controle sobre as atividades que estão sendo cumpridas, e que possa ser utilizado por qualquer aluno de qualquer instituição de ensino superior. Para realização do desenvolvimento da aplicação móvel, foi aplicado técnicas de engenharia de software juntamente com modelos de desenvolvimento ágil scrum solo, com sprints semanais. Após a fase de desenvolvimento foi realizada a implantação da aplicação em uma loja de aplicativo e disponibilizada para os usuários. Por último a aplicação foi validada pelo público-alvo através de uma pesquisa, onde foi avaliada a usabilidade e a experiência dos usuários com a aplicação, o que demonstrou que a aplicação entregou o que foi proposto com resultados satisfatórios.

Palavras-chave: Horas complementares; Atividades complementares; Atividades de extensão; Engenharia de software; Aplicativo móvel; Mobile; Gerenciamento

ABSTRACT

Complementary activities play an essential role in academic education, allowing students to expand their knowledge beyond regular courses. These activities can include participation in extension projects, internships, social projects, events, among others. These activities are required by some higher education institutions as they are part of the academic curriculum and are important for students' education, as they enable the development of skills and competencies, as well as providing real-life work experiences. Each institution may have its own strategies to encourage and ensure compliance with these complementary activities by students. It is common for students to face difficulties in fulfilling these activities due to the lack of a system that facilitates their tracking and management. Given the current global scenario, where mobile technology is increasingly prevalent, the development of a mobile application to manage these activities could be a solution to alleviate the challenges faced by students in complying with and tracking these activities. In this context, the objective of this study is to develop a mobile application that enables the tracking and management of complementary activities, providing students with greater control over the activities they are completing, including both complementary activities and extension activities, and that can be used by any student from any higher education institution. To develop the mobile application, software engineering techniques were applied, along with agile Scrum Solo development models, using weekly sprints. After the development phase, the application was deployed in an app store and made available to users. Finally, the application was validated by the target audience through a survey, where usability and user experience were evaluated. This demonstrated that the application delivered what was proposed with satisfactory results.

Keywords: Complementary hours; Complementary activities; Extension activities; Software engineering; Mobile application; Mobile; Management

LISTA DE FIGURAS

Figura 1 – Diagrama de Casos de Uso Akademic	19
Figura 2 – Formulário de cadastro de horas complementares no Sistema Integrado de Gestão de Atividades Acadêmicas (SIGAA)	23
Figura 3 – Acompanhamento de horas complementares no SIGAA	23
Figura 4 – Um diagrama simples exemplificando a relação entre Model, View e Controller. As linhas sólidas indicam associação direta e as tracejadas indicam associação indireta.	26
Figura 5 – Fluxo do modelo de desenvolvimento <i>Scrum</i>	28
Figura 6 – Fluxo do modelo de desenvolvimento <i>Scrum Solo</i>	29
Figura 7 – Passos metodológicos	30
Figura 8 – Estrutura de pastas do projeto.	38
Figura 9 – Prints da sprint 2.	42
Figura 10 – Prints da sprint 2.	42
Figura 11 – Prints da sprint 3.	43
Figura 12 – Prints do artefato desenvolvido na sprint 4.	44
Figura 13 – Prints do artefato desenvolvido na sprint 4.	44
Figura 14 – Prints do artefato desenvolvido na sprint 4.	45
Figura 15 – Lista de atividades exportadas.	45
Figura 16 – Bibliotecas utilizadas no projeto.	49
Figura 17 – Como você avalia a facilidade de uso da aplicação?	51
Figura 18 – Como você avalia a aparência visual da aplicação?	52
Figura 19 – Como você avalia a velocidade e desempenho da aplicação?	52
Figura 20 – Quais funcionalidade você mais utilizou na aplicação?	53
Figura 21 – Você usaria esta aplicação?	53

LISTA DE QUADROS

Quadro 1 – Quadro comparativo de trabalhos - 1	21
Quadro 2 – Tipos de requisitos	25
Quadro 3 – Funcionalidades da Aplicação	32
Quadro 4 – Ferramentas	33
Quadro 5 – Avaliação da Aplicação	35
Quadro 6 – funcionalidades da aplicação proposta	37
Quadro 7 – Estrutura de pastas do aplicativo.	38
Quadro 8 – Bibliotecas utilizadas na aplicação.	39
Quadro 9 – Resumo das Sprints.	40
Quadro 10 – Priorização das tarefas por Sprint.	41
Quadro 11 – Bugs identificados na aplicação.	46
Quadro 12 – Respostas subjetivas categorizadas	54

LISTA DE ABREVIATURAS E SIGLAS

SIGAA	Sistema Integrado de Gestão de Atividades Acadêmicas
UFC	Universidade Federal do Ceará
ACs	Atividades Complementares
IES	Instituição de Ensino Superior
MEC	Ministério da Educação
AEs	Atividades Extensionistas
<i>Uniceplac</i>	<i>Centro Universitário do Planalto Central Apparecido dos Santos</i>
UniAcademia	Centro Universitário Academia
NoSQL	Not Only SQL
WEB	World Wide Web
ER	Engenharia de requisitos
ES	Engenharia de software
MVC	Model-View-Controller
PSP	Personal Software Process

SUMÁRIO

1	INTRODUÇÃO	15
2	TRABALHOS RELACIONADOS	17
2.1	<i>Desenvolvimento de um sistema para gerenciamento de atividades complementares (BARROS et al., 16 set. 2021)</i>	17
2.2	<i>HORAS COMPLEMENTARES: Um aplicativo de gestão de atividades acadêmicas (TOLEDO; SIRQUEIRA, 20 jul. 2021)</i>	17
2.3	<i>Implementação de um Sistema Multi-plataforma para Gerenciamento de Atividades Complementares em Cursos Superiores utilizando Banco de Dados noSQL (LIMA et al., 2018)</i>	18
2.4	Análise Comparativa	20
3	FUNDAMENTAÇÃO TEÓRICA	22
3.1	Atividades complementares em instituições de ensino superior	22
3.2	Atividades complementares na Universidade Federal do Ceará (UFC)	22
3.3	Desenvolvimento de aplicações móveis	23
3.4	Engenharia de Software	24
3.4.1	<i>Engenharia de requisitos</i>	24
3.4.2	<i>Arquitetura de software</i>	25
3.4.2.1	<i>Padrão MVC (Model, View and Controller)</i>	26
3.4.3	<i>Scrum Solo</i>	27
3.4.4	<i>Validação</i>	28
4	PROCEDIMENTOS METODOLÓGICOS	30
4.1	Identificação de aplicações relacionadas	30
4.2	Definição das funcionalidades da aplicação	31
4.3	Modelagem do sistema	32
4.4	Desenvolvimento	33
4.5	Implantação	34
4.6	Validação	34
5	DESENVOLVIMENTO DA APLICAÇÃO DE ATIVIDADES COMPLEMENTARES	36

5.1	Análise das aplicações dos trabalhos relacionados para identificação de funcionalidades	36
5.2	Definição das funcionalidades do aplicativo de Atividades Complementares	36
5.3	Modelagem do sistema	37
5.3.1	<i>Estrutura de pastas</i>	37
5.3.2	<i>Bibliotecas utilizadas</i>	39
5.4	Desenvolvimento em sprints	39
5.4.1	<i>Primeira sprint</i>	40
5.4.2	<i>Segunda sprint</i>	40
5.4.3	<i>Terceira sprint</i>	42
5.4.4	<i>Quarta sprint</i>	44
5.4.5	<i>Quinta sprint</i>	45
5.5	Implantação	47
5.6	Validação	48
6	RESULTADOS	51
6.0.1	<i>Resultados da pesquisa</i>	51
7	CONSIDERAÇÕES FINAIS	55
7.1	Limitações	56
7.2	Trabalhos Futuros	56
	REFERÊNCIAS	58
	APÊNDICES	59
	APÊNDICE A -QUESTIONÁRIO DE AVALIAÇÃO DA APLICAÇÃO	59

1 INTRODUÇÃO

De acordo com o Parecer CNE/CES nº 136/2012, aprovado em 9 de março de 2012, que institui as diretrizes Curriculares Nacionais para os cursos de graduação, as Atividades Complementares (ACs) é um componente curricular que têm como objetivo principal enriquecer o perfil do aluno e futuro profissional com atividades em diversos meios dentro e fora do ambiente acadêmico. Tais atividades são cruciais para o desenvolvimento do aluno estimulando-o de forma prática e interdisciplinar, tornando-o assim mais preparado para relações no âmbito profissional (BRASIL, 2012).

Como forma de preparação dos egressos para os ambientes onde esses irão desempenhar seus papéis após formados, as ACs podem ser realizadas em diversos cenários, categorias e instituições. De acordo com o Art. 9º do Parecer CNE/CES nº 136/2012, parágrafo único, as ACs podem ser exercidas tanto na Instituição de Ensino Superior (IES) a qual o aluno está matriculado quanto em outras IES, assim como em diversos ambientes sociais. As atividades a serem realizadas podem ser de cunho técnico-científicos ou formação profissional, incluindo experiências de trabalho, estágios não obrigatórios, extensão universitária, iniciação científica, participação em eventos técnico-científicos, publicações científicas, programas de monitoria e tutoria, disciplinas de outras áreas, representação discente em comissões e comitês, participação em empresas juniores, incubadoras de empresas ou outras atividades de empreendedorismo e inovação"(Ministério da Educação (MEC), 2012).

Cada IES possui um regimento próprio da quantidade de horas que cada curso deve ter. É comum que as ACs sejam distribuídas em categorias. Essa categorização assim como a quantidade mínima e/ou máxima de horas para cada categoria podem variar entre cursos e também IES.

No contexto da UFC, a integralização dessas ACs é feita através do SIGAA, onde o aluno pode submeter as ACs e visualizar seu progresso de maneira online e digitalizada. Essa sistematização ainda não é algo comum e presente nas outras IES. Em muitas universidades o controle de entregas das ACs é feito somente pela coordenação da IES, sendo permitido em determinado período do semestre o envio da documentação que comprove a participação do aluno em ACs.

Além das ACs, as IES vem se preparando a curricularização da extensão. A curricularização da extensão é uma estratégia nacional prevista na Meta 12.7 do Plano Nacional de Educação (PNE) 2014-2024 (Lei nº 13.005/2014). De acordo com a legislação, devem ser

assegurados, no mínimo, 10% do total de créditos curriculares exigidos para a graduação em projetos de extensão, prioritariamente em áreas de grande pertinência social. Ou seja, se um curso de graduação possui carga horária total de 4 mil horas, por exemplo, será necessário que pelo menos 400 horas sejam cumpridas pelo aluno em Atividades Extensionistas (AEs).

A gestão, controle e entrega das ACs e de AEs podem ser um processo confuso e custoso para os estudantes. Independentemente de terem participado de várias atividades ou não, os alunos podem enfrentar dificuldades ao centralizar e organizar a documentação e certificados de participação, especialmente quando o período de envio é aberto pela Instituição de Ensino Superior (IES). Nesse sentido, é interessante propor o uso de uma ferramenta tecnológica que possa resolver ou minimizar os impactos negativos causados por essa problemática.

O objetivo deste trabalho é desenvolver uma aplicação móvel genérica que possa ser utilizada em diferentes contextos, independentemente da IES. Essa aplicação permitirá que os alunos registrem e anexem, de maneira centralizada e organizada, todas as informações obrigatórias para a validação das ACs e AEs. A aplicação também exibirá o progresso do aluno de forma simplificada, intuitiva e dividida por categorias, mostrando a quantidade de horas já submetidas, pendentes e totais. Os alunos também poderão gerar um documento compilado de todos os registros feitos, com a opção de exportá-lo e enviá-lo para a coordenação da IES. Além disso, a aplicação permitirá que os usuários cadastrem metas, definindo um número específico de horas a serem cumpridas em um determinado período para uma determinada categoria.

Além da aplicação móvel proposta, este trabalho busca analisar e avaliar as escolhas técnicas de desenvolvimento utilizadas, bem como avaliar a aplicação com o público-alvo, que são os alunos das IES. Para alcançar esses objetivos, serão adotadas abordagens adaptadas do Scrum para equipes compostas por apenas um indivíduo no processo de desenvolvimento de software. Os resultados alcançados em cada etapa, os artefatos criados e as decisões tomadas ao longo do projeto foram documentadas no presente trabalho

2 TRABALHOS RELACIONADOS

Neste capítulo serão apresentados alguns trabalhos relacionados. Foram selecionados três trabalhos que se assemelham bastante ao objetivo do presente trabalho, onde também se é proposto ferramentas que automatizam o processo de gerenciamento de ACs em IES. Por fim é apresentado uma seção de comparação e diferenças entre tais trabalhos e a solução proposta no presente trabalho.

2.1 Desenvolvimento de um sistema para gerenciamento de atividades complementares (BARROS et al., 16 set. 2021)

No trabalho de Barros *et al.* (16 set. 2021), apresenta-se o contexto do curso de Sistemas de Informação do *Centro Universitário do Planalto Central Apparecido dos Santos (Uniceplac)* - Gama - Distrito Federal. Nesse trabalho é apresentado um sistema para gerenciar as ACs, tanto dos cursos do *Uniceplac* quanto de quaisquer instituições de ensino que queiram fazer uso do sistema. O sistema facilita e acelera o fluxo de validação das ACs, centralizando as informações necessárias para a submissão, de tal forma que possa gerenciar de maneira mais adequada todo esse processo.

No trabalho, é detalhado algumas fases do desenvolvimento: definição da tecnologia a ser utilizada, modelagem do sistema a ser desenvolvido e apresentação e análise de dados, onde é revelado detalhes sobre o fluxo de usabilidade e interface do sistema. Foram utilizadas tecnologias *open sources*: Laravel, MySql (para o back-end) e Vue.js (para o front-end).

O sistema tem como usuários finais estudantes e validadores. O estudante pode submeter atividades respondendo questionários e anexando certificados, caso necessário. Já o validador, é a pessoa funcionária da IES responsável por avaliar as informações que foram submetidas pelos estudantes ao sistema, decidindo se aprova ou não a atividade submetida. Ambos, estudante e validador, conseguem ver o progresso do aluno em relação as horas submetidas e pendentes.

2.2 HORAS COMPLEMENTARES: Um aplicativo de gestão de atividades acadêmicas (TOLEDO; SIRQUEIRA, 20 jul. 2021)

No trabalho de Toledo e Sirqueira (20 jul. 2021) apresenta-se o contexto dos cursos do Centro Universitário Academia (UniAcademia). A princípio a aplicação proposta se aplica

somente aos cursos de Engenharia de Software e Sistemas de Informação, podendo ser evoluída para abranger outros cursos.

O trabalho começa introduzindo o leitor à importância da tecnologia para a resolução de problemas do dia a dia. Em seguida descreve a relevância das ACs juntamente com as problemáticas que são encontradas no cotidiano de uma IES para gerenciar tais registros de atividades.

O trabalho então propõe uma aplicações *mobile*, disponível para *Android* e *iOS*, denominada *Horas Complementares*. A aplicação foi desenvolvida usando tecnologias *open source*, as principais sendo: *Node.js*, *React-Native*, *PostgreSQL*, entre outras. O artigo tem seções dedicadas a explanar detalhadamente tópicos como: requisitos funcionais, modelagem da aplicação, diagramas de caso de uso, documentos de caso de uso e modelagem do banco de dados. Por fim, o trabalho exibe imagens e comenta a interface gráfica da aplicação.

2.3 *Implementação de um Sistema Multi-plataforma para Gerenciamento de Atividades Complementares em Cursos Superiores utilizando Banco de Dados noSQL (LIMA et al., 2018)*

O trabalho de Lima tem como objetivo gerenciar as ACs, o processo de submissão das informações e os certificados necessários para a validação e acompanhamento do progresso já cumprido ou pendente.

Os autores propuseram um sistema chamado *Akademic*. Um sistema para envio de documentação e gerenciamento das ACs. O sistema não é focado em uma única IES, podendo ser adaptado para outras instituições e também para outros tipos de atividades.

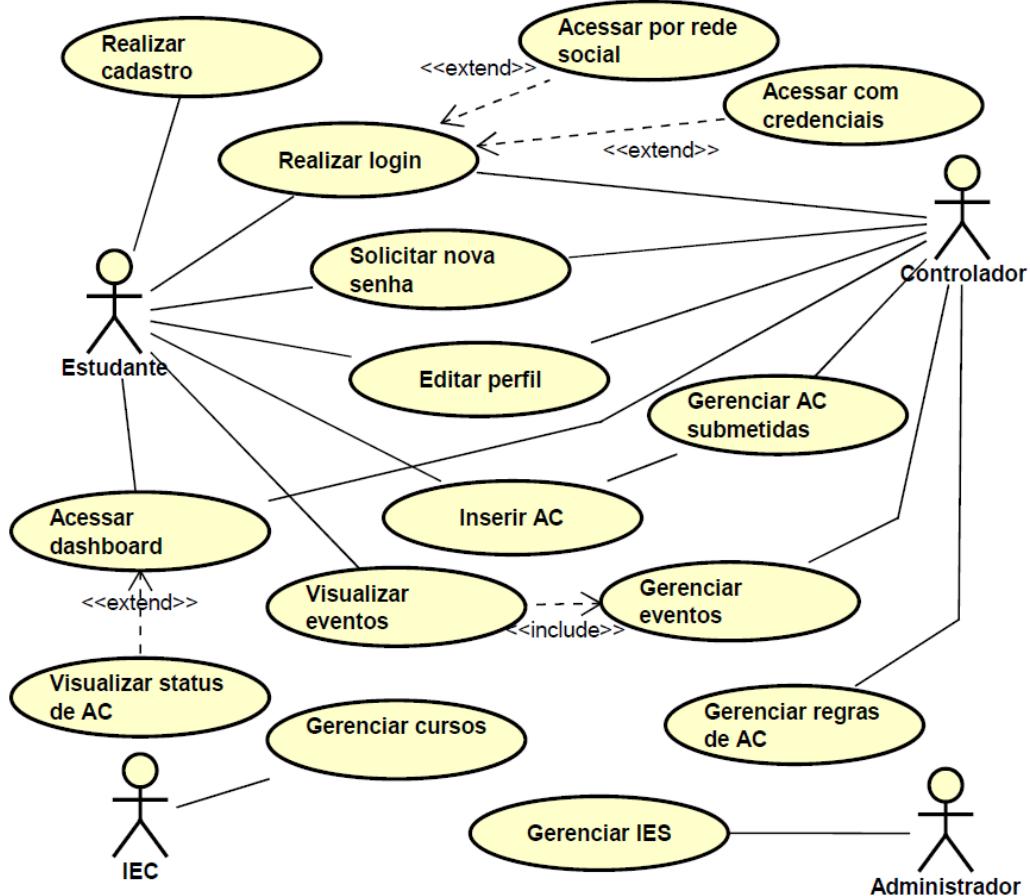
O trabalho documenta as seguintes fases de desenvolvimento da aplicação: levantamento de requisitos, definição da arquitetura para a aplicação, definição da modelagem de dados, implementação da aplicação.

O sistema foi desenvolvido utilizando uma ferramenta de desenvolvimento web chamada *Angular.js*, que possibilita que a aplicação seja acessada de diferentes dispositivos e em diferentes resoluções. Para o banco de dados foi utilizado uma ferramenta do google chamada *Firebase*, que utiliza banco de dados *Not Only SQL (NoSQL)*, facilitando o escala, caso necessário.

O sistema tem alguns perfis de usuários, sendo eles: O *administrador* do sistema (com privilégios máximos de acesso). O *controlador* (geralmente o coordenador de curso), cujo

a principal ação é a validação das ACs enviadas pelos estudantes, aprovando-as ou não. A *IES*, que pode cadastrar *controladores*, que são associados aos cursos. E por último, o *Estudante*, cujo, dentre outras, a principal ação é realizar a gerência das próprias ACs, submetendo-as para validação do *controlador*. A figura 1 ilustra um diagrama de caso de uso, exibindo as funcionalidades do sistema *Akademic*, assim como os atores envolvidos nas funcionalidades.

Figura 1 – Diagrama de Casos de Uso Akademic



Lima *et al.* (2018)

2.4 Análise Comparativa

Esta seção apresenta uma análise comparativa entre os trabalhos apresentados neste capítulo e a solução proposta no presente trabalho. Todos os trabalhos apresentados nesse capítulo propõem como solução uma ferramenta *WEB* ou *mobile*. Todos os trabalhos, assim como o presente trabalho, tem como foco desenvolver uma aplicação para o gerenciamento das ACs. No entanto somente o presente trabalho inclui o gerenciamento de AEs.

Com exceção do trabalho de Leoncio (2018), todos os demais trabalhos, assim como o presente trabalho, permite o anexo de arquivos digitais, removendo necessidade de manuseio de documentos físicos.

A grande semelhança entre Barros *et al.* (16 set. 2021) e o presente trabalho é a funcionalidade de anexar certificados para a aplicação. Uma outra semelhança é a visualização do progresso de horas submetidas e pendentes. A diferença se encontra em parte do público-alvo, pois neste trabalho, não tem o perfil de validador, os usuários-finais serão apenas os estudantes.

No trabalho de Toledo e Sirqueira (20 jul. 2021), existem dois perfis de usuários finais, o administrador do curso e o aluno. Já o presente trabalho, só terá uma persona de usuário final: O aluno. O presente trabalho também visa disponibilizar uma aplicação para qualquer aluno de qualquer curso, não somente para os cursos de Engenharia de Software e Sistemas de Informação.

A principal semelhança entre o presente trabalho e (LIMA *et al.*, 2018) é que os trabalhos usam a tecnologia Firebase para o banco de dados. No trabalho de (LIMA *et al.*, 2018) existem quatro perfis de usuários. Já o presente trabalho, terá somente um perfil de usuário: Estudante.

Os trabalhos que foram apresentados tem em comum otimizar através do uso de um sistema de informação um tema comum em IES. Cada um desses trabalhos, apesar de resolverem problemas parecidos, usam diferentes abordagens e soluções com diferentes funcionalidades, sendo algumas delas semelhantes ao do trabalho que está sendo desenvolvido. Dessa forma foram definidos alguns critérios de comparação entre os trabalhos relacionados e o presente trabalho:

- **Notificações Push:** Utilizado para checar se a aplicação proposta possui a funcionalidade de Notificações *Push*
- **Perfís de usuário:** Utilizado para listar os tipos de perfis de usuários que a aplicação proposta possui.

- **Envio de documentação:** Utilizado para verificar se a aplicação proposta possui alguma funcionalidade para anexar e armazenar arquivos.
- **Plataforma:** Utilizado para verificar com quais plataformas a aplicação é compatível para funcionamento
- **Planejamento:** Utilizado para verificar se a aplicação fornece alguma funcionalidade para o planejamento da execução das ACs
- **Exportação de informações:** Utilizado para verificar se a aplicação proposta possui alguma funcionalidade para exportar informações, em formato *PDF*, as informações dos usuários.

Este trabalho, diferente dos demais trabalhos, não depende da integração com os sistemas específicos das IES. Diferentemente dos outros trabalhos comparados, a aplicação desenvolvida pode ser baixada e utilizada por qualquer aluno, independentemente da instituição em que estão matriculados. Isso proporciona uma maior acessibilidade e flexibilidade aos usuários, tornando o trabalho um diferencial significativo. Além disso, o trabalho também aborda o planejamento da execução das ACs e AEs, agregando valor adicional à aplicação.

Por fim, o objetivo deste trabalho é propor uma aplicação de fácil acesso para o gerenciamento de ACs e AEs, que possa ser utilizada por alunos de qualquer IES. Que a aplicação proposta também permita que os alunos acompanhem seu progresso de forma simplificada e intuitiva, sem a necessidade de integrações com os sistemas específicos das IES. Isso elimina burocracias e torna o processo mais ágil e descomplicado para os usuários.

O quadro 1 apresenta o resumo dessa análise.

Quadro 1 – Quadro comparativo de trabalhos - 1

Trabalho	Notificações <i>Push</i>	Perfís de usuários	Plataforma	Exportação de informações	Permite anexar certificado	Planejamento de execução das atividades
Barros <i>et al.</i> (16 set. 2021)	Não	Administrador, Validor e Estudante	WEB	Não	Sim	Não
Toledo e Sirqueira (20 jul. 2021)	Não	Administrador do curso e Estudante	WEB	Não	Sim	Não
Lima <i>et al.</i> (2018)	Sim	Administrador do sistema, Controlador e Estudante	Android e iOS	Não	Sim	Não
Presente trabalho	Não	Estudante	Android	Sim	Sim	Sim

Fonte: Elaborado pelo autor.

3 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão apresentados os principais conceitos para o desenvolvimento deste trabalho.

3.1 Atividades complementares em instituições de ensino superior

As ACs e AEs são uma exigência das IES, sendo uma obrigatoriedade a ser cumprida pelo aluno, sendo também uma lei determinada pelas Diretrizes Curriculares dos Cursos de Graduação e pela Lei 9.394/96, que estabelece as Diretrizes da Educação Nacional.

As ACs e AEs, como já mencionado na introdução, tem o objetivo de enriquecer o perfil do aluno e expandir as suas experiências com situações além da técnica e da sala de aula.

As atividades podem ser de diferentes categorias e ambientes: cunho técnico-científicos ou formação profissional, incluindo experiências de trabalho, estágios não obrigatórios, extensão universitária, iniciação científica, participação em eventos técnico-científicos, publicações científicas, programas de monitoria e tutoria, disciplinas de outras áreas, representação discente em comissões e comitês, participação em empresas juniores, incubadoras de empresas ou outras atividades de empreendedorismo e inovação"(MEC, 2012).

Cada IES tem suas regras para o cumprimento das ACs e AEs, podendo ser dividida em categorias e contendo ou não carga horária máxima para cada categoria.

3.2 Atividades complementares na UFC

Atualmente na UFC, as submissões e consultas de horas das ACs e AEs são feitas integralmente através do SIGAA, um portal de integração acadêmica, que conecta reitoria, coordenadoria, professores e alunos da UFC.

Através do sistema SIGAA o aluno pode submeter a atividade, preenchendo um formulário e anexando um certificado (Figura 2). Também é possível consultar e acompanhar as horas pendentes, submetidas e deferidas (Figura 3). Neste cenário, a solução proposta no presente trabalho, utiliza o SIGAA como um ponto de referência para o desenvolvimento de uma aplicação móvel que permitirá os alunos submeterem e acompanharem as horas das ACs e AEs.

Figura 2 – Formulário de cadastro de horas complementares no SIGAA

Fonte: SIGAA - 11/2022

Figura 3 – Acompanhamento de horas complementares no SIGAA

Tipo de Carga Horária	: Integralizar Creditações				
	Horas Exigidas	Horas em Solicitações Submetidas	Horas em Solicitações Deferidas	Horas Integralizadas	Horas Pendentes de Integralização
EXTENSÃO	0	0	0	0	0
C.H. de Extensão em Disciplinas (Modalidade II)	--	--	--	0	--
C.H. de Extensão em Atividades (Modalidade I) - Somente se adotada pelo curso	--	0	0	0	--
ATIVIDADE COMPLEMENTAR	288	288	280	0	288
C.H. Complementar em Iniciação à Docência, Pesquisa e/ou Extensão (Máx: 96 hs)	--	96	96	0	--
C.H. Complementar em Participação e/ou Organização de Eventos (Máx: 32 hs)	--	32	32	0	--
C.H. Complementar em Atividades Artísticas-Culturais e Esportivas (Máx: 80 hs)	--	20	14	0	--
C.H. Complementar em Experiências Ligadas à Formação Profissional (Máx: 64 hs)	--	64	64	0	--
C.H. Complementar em Produção Técnica e/ou Científica (Máx: 96 hs)	--	4	4	0	--
C.H. Complementar em Vivências de Gestão (Máx: 48 hs)	--	24	24	0	--
C.H. Complementar em Outras Atividades (Máx: 48 hs)	--	48	46	0	--

Fonte: SIGAA - 11/2022

3.3 Desenvolvimento de aplicações móveis

Os programas móveis são diferentes dos programas de desktop que são estáticos. Eles são projetados com portabilidade e facilidade de uso em mente (LECHETA, 2009). Além disso, os programas móveis são executados em hardware móvel, como tablets e smartphones.

O software executado em dispositivos móveis cresceu com o desenvolvimento de dispositivos móveis. Muitos novos programas de software agora existem fora dos telefones celulares e tablets. Novos softwares estão disponíveis até mesmo para relógios inteligentes, como o Apple Watch e o Android Wear do Google (WASSERMAN, 07 Nov. 2010).

Ter milhares de usuários acessando aplicativos móveis por meio de lojas de aplicativos traz um grande potencial para o mercado de software. Por causa disso, é possível alcançar milhares de clientes que têm como pares potenciais compradores de aplicativos.

Os aplicativos móveis exigem um alto grau de garantia de qualidade. No entanto, existem poucos processos e procedimentos definidos para apoiar o desenvolvimento desses aplicativos. Consequentemente, a maioria dos desenvolvedores usa métodos não estruturados

sem nenhum rastreamento efetivo. Isso resulta em um alto grau de competitividade no mercado (WASSERMAN, 07 Nov. 2010).

A compreensão do desenvolvimento de software móvel é fundamental neste projeto, pois é essencial criar um aplicativo que possa ser facilmente distribuído em várias plataformas. Isso se deve ao fato de que a portabilidade permite que os aplicativos utilizem recursos limitados da Internet, o que é uma consideração importante ao criar uma solução de software.

3.4 Engenharia de Software

Segundo Pressman e Maxim (2016), a engenharia de software é uma abordagem disciplinada, sistemática e quantificável no ciclo de desenvolvimento de um sistema de software. A engenharia de software possui uma variedade de métodos e ferramentas. Essas auxiliam na criação de artefatos (documentos, modelos, dados, relatórios etc.). A engenharia de software faz com que essas ferramentas se integrem e se comuniquem entre si, de modo que um artefato, que é criado por uma ferramenta, seja utilizado em outra(s) ferramenta(s) ou método(s). A engenharia de software tem foco na qualidade e para isso ela propõe marcos, métricas de qualidade, metodologias para análises de viabilidade do sistema, gerenciamento e entrega.

A engenharia de software é essencial porque capacita o desenvolver e a equipe para o desenvolvimento de sistemas de software dentro do prazo e com alta qualidade. Ela permite o time de desenvolvimento adapte e aplique a sua própria abordagem, tornando o processo que poderia ser caótico em algo disciplinado e conveniente para as necessidades da equipe. (PRESSMAN; MAXIM, 2016)

Isto posto, as próximas subseções irão apresentar tópicos de comum importância na engenharia de software e para o desenvolvimento do presente trabalho.

3.4.1 Engenharia de requisitos

Os requisitos tem como papel descrever os serviços e tarefas que o sistema deve prestar e as restrições de suas operações. Tais requisitos representam as necessidades do cliente que atende a um determinado propósito, como encontrar informação, controlar um dispositivo ou fazer um pedido. O processo para a descoberta, análise, documentação e validação desses requisitos é chamado de Engenharia de requisitos (ER). (SOMMERVILLE, 2019).

Na ER normalmente é apresentada na primeira fase do processo de engenharia de

software. Na ER é documentado tudo que o cliente espera encontrar no sistema. Um bom levantamento e validação dos requisitos favorecem consideravelmente o processo de desenvolvimento da aplicação. Por isso é considerada uma das paradas mais importantes da Engenharia de software (ES). (SOMMERVILLE, 2019)

Existem alguns tipos de requisitos durante o desenvolvimento de um software. Eles são tipados como exibido no Quadro 2. Cada tipo de requisito tem sua importância no processo de ES. (WIEGERS; BEATTY, 2013)

Quadro 2 – Tipos de requisitos

Termo	Definição
Exigência de negócio	Um objetivo de negócios de alto nível da organização que constrói um produto ou de um cliente que o adquire.
Regra de negócios	Uma política, diretriz, padrão ou regulamento que define ou restringe algum aspecto do negócio. Não é um requisito de software em si, mas a origem de vários tipos de requisitos de software.
Restrição	Restrição imposta às escolhas disponíveis ao desenvolvedor para o projeto e construção de um produto.
Requisito de interface externa	Uma descrição de uma conexão entre um sistema de software e um usuário, outro sistema de software ou um dispositivo de hardware.
Recurso	Um ou mais recursos de sistema logicamente relacionados que fornecem valor a um usuário e são descritos por um conjunto de requisitos funcionais.
Requisito funcional	Uma descrição de um comportamento que um sistema exibirá sob condições específicas.
Requisito não funcional	Uma descrição de uma propriedade ou característica que um sistema deve exibir ou um condicionalidade que deve respeitar.
Atributo de qualidade	um tipo de requisito não funcional que descreve um serviço ou característica de desempenho de um produto.
Requisito do sistema	Um requisito de nível superior para um produto que contém vários subsistemas, que pode ser todo software ou software e hardware.
Requisito do usuário	Um objetivo ou tarefa que classes específicas de usuários devem ser capazes de executar com um sistema, ou um atributo de produto desejado.

Fonte: Traduzida pelo autor.

No âmbito do trabalho proposto, a ER se faz presente no levantamento e definição dos requisitos do sistema. A seção 5.2, é dedicada apresentar em mais detalhes tais requisitos.

3.4.2 Arquitetura de software

A arquitetura de software define como as partes do software são organizadas. Na arquitetura de software são listados quais componentes estruturais compõem aquele software e como eles se relacionam. Segundo (SOMMERVILLE, 2019), a arquitetura de software é o vínculo fundamental entre o projeto e a engenharia de requisitos. Na fase de modelagem de um sistema, a arquitetura é um dos primeiros passos a serem executados. O artefato produzido

durante a fase de arquitetura é um documento contendo um diagrama, ilustrando como o sistema está organizado e como os conjuntos de componentes se comunicam.

As definições feitas durante a fase de arquitetura impactam diretamente todo o ciclo de vida do sistema, já que essas definições servem como alicerce para o início do desenvolvimento do projeto, diminuindo as preocupações com decisões técnicas durante a fase de codificação. (BASS *et al.*, 2012). De acordo com (LIENTZ *et al.*, 1978), a manutenção e aprimoramento do sistema consomem de 75% a 80% do ciclo de vida de um sistema. Sabendo disso, uma arquitetura de software bem-feita, facilita a fase de desenvolvimento e reduz custos de evolução e manutenção que poderiam surgir a longo prazo.

3.4.2.1 Padrão MVC (*Model, View and Controller*)

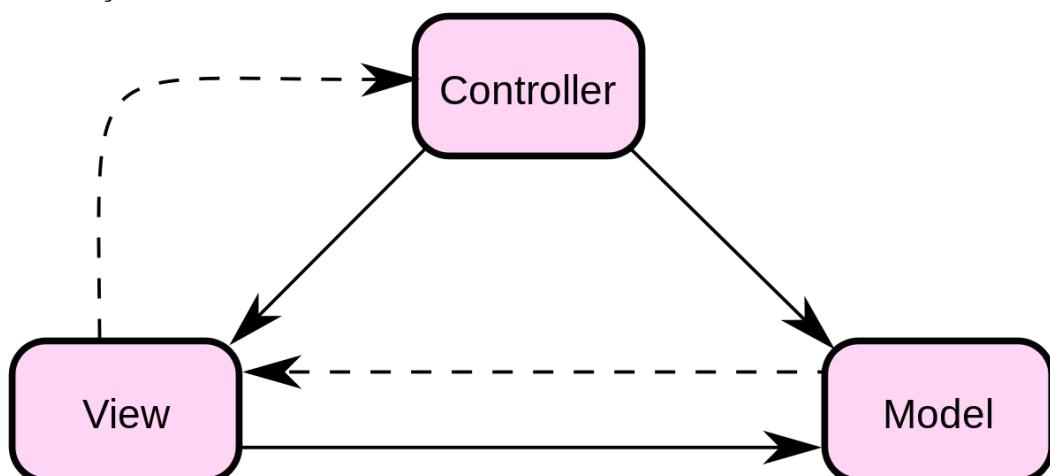
O *Model-View-Controller (MVC)* é um padrão de projeto de software. Ele estrutura o sistema em 3 camadas que interagem entre si. Essas camadas são:

- Model (Modelo): Responsável por centralizar os dados da aplicação.
- View (Interface): Define e gerencia como os dados são apresentados ao usuário.
- Control (Controle): Gerencia as interações do usuário e mudanças de estados.

Fornecendo comunicação entre Model e View.

Segundo Sommerville (2019), conceitos de independência e separação são fundamentais para o projeto de arquitetura e esses conceitos são muito bem aplicados no padrão MVC. A Figura 4 ilustra a relação entre as três camadas citadas.

Figura 4 – Um diagrama simples exemplificando a relação entre Model, View e Controller. As linhas sólidas indicam associação direta e as tracejadas indicam associação indireta.



Fonte: Elaborado pelo autor.

3.4.3 Scrum Solo

O *Scrum Solo* é um *framework* de gerenciamento de projetos criado em 2012 por Tiago Pagotto e sua equipe. O objetivo desse *framework* é sanar a ausência de uma técnica de desenvolvimento para desenvolvedores solos. Segundo Pagotto *et al.* (28 jul. 2016), o *Scrum Solo* é baseado no *Scrum* e que une as boas práticas delineadas pelo Personal Software Process (PSP)¹.

Para entender o *Scrum Solo*, primeiro é preciso uma contextualização sobre o *Scrum* tradicional. O *Scrum* é um *framework* para gerenciamento de projetos. Ele foi idealizado no início da década de 90 por Jeff Sutherland e sua equipe. O *Scrum* tem como base os princípios do manifesto ágil² que são usados para guiar as tarefas de desenvolvimento no âmbito de processos que incorpore as seguintes atividades metodológicas: requisitos, análise, projeto, evolução e entrega. (PRESSMAN; MAXIM, 2016)

No *Scrum* há alguns termos, papéis e atividades de desenvolvimento:

1. Papéis:

- *Scrum Master* (Mestre de *Scrum*)
- *Product Owner* (proprietário do produto)
- *Scrum Team* (Analista e programadores)

2. Eventos:

- *Sprint Planning* (Planejamento da *sprint*)
- *Daily Scrum* (Reunião de *Scrum* diária)
- Execução da *Sprint*
- Revisão da *Sprint*
- Retrospectiva da *Sprint*

3. Artefatos:

- *Product Backlog* (Requisitos ou tarefas a serem feitas)
- *Sprint Backlog* (Tarefas a serem feitas naquela *sprint*)
- Incremento/Entrega

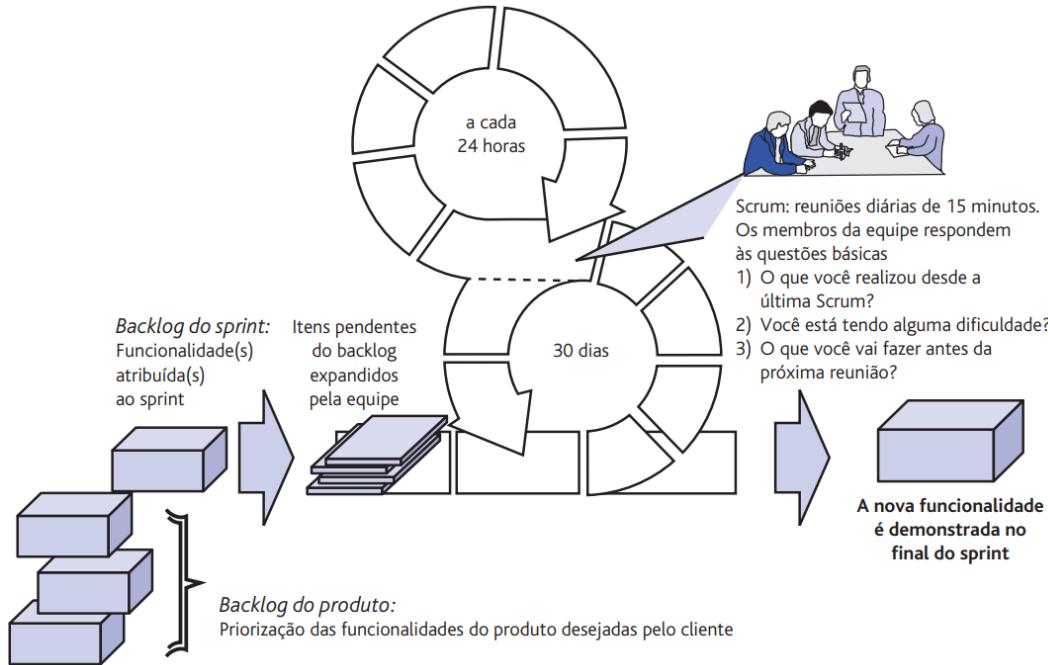
A Figura 5 ilustra o fluxo do modelo *Scrum*.

Sobre o *Scrum Solo*, é possível analisar a Figura 6 e perceber a semelhança entre ambos os *frameworks*, *Scrum Solo* e *Scrum*. O *product backlog* e o *sprint backlog* acontecem

¹ Concebido por Watts Humphrey em 1995 o Personal Software Process é um processo de desenvolvimento de software projetado para ser utilizado por engenheiros de software para a elaboração de projetos individuais.

² Manifesto Ágil: <https://agilemanifesto.org/iso/ptbr/manifesto.html>

Figura 5 – Fluxo do modelo de desenvolvimento *Scrum*



Fonte: (PRESSMAN; MAXIM, 2016)

de maneira idêntica em ambas as abordagens. No *Scrum Solo* é sugerido que as *sprints* tenham duração reduzida para uma semana e que não haja reuniões diárias; Ao fim de cada *sprint*, assim como feito no *Scrum*, a equipe de desenvolvimento deve entregar um artefato, como um protótipo com novas funcionalidades. No *Scrum Solo* também pode existir, quando houver a necessidade, reuniões de orientação entre o grupo de validação (clientes e usuários finais) e o desenvolvedor. (PAGOTTO *et al.*, 28 jul. 2016)

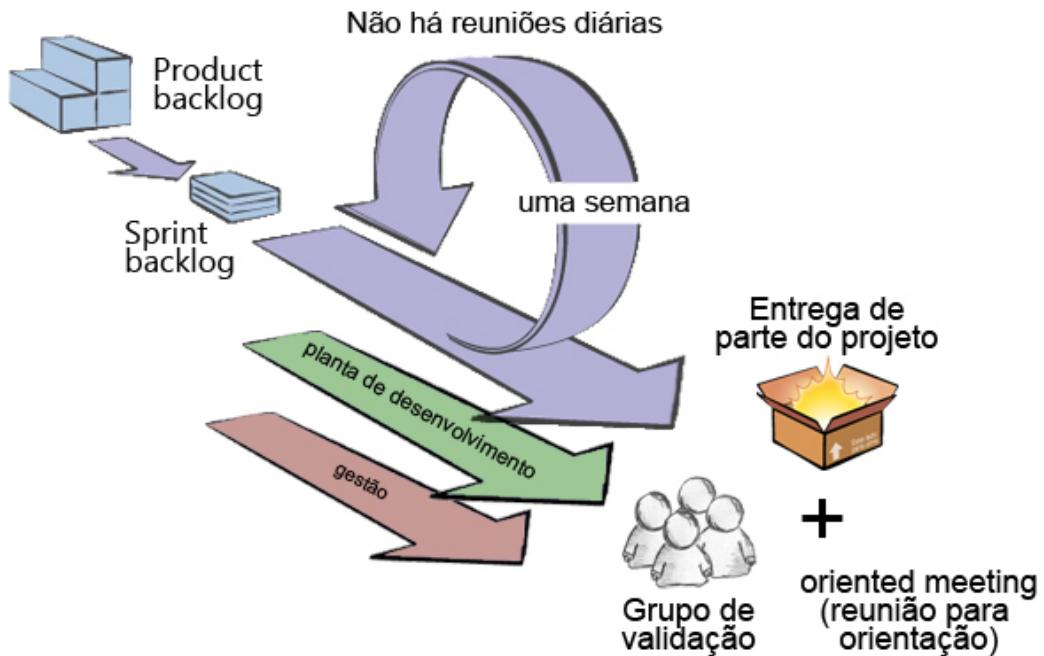
No âmbito do presente trabalho o *Scrum Solo* é usado pelo autor como modelo a ser seguido para o processo de desenvolvimento da aplicação proposta.

3.4.4 Validação

Ao longo do desenvolvimento de um software até a sua distribuição é essencial a sua validação, que parte desde a execução das funcionalidades até a conformidade com os requisitos definidos.

Os testes podem ser aplicados em vários níveis. Desde linhas ou bloco de código, passando por testes de funcionalidades, testes de interfaces, testes de integração (que valida o comportamento do sistema ao se comunicar com outros sistemas, serviços ou internet), testes das decisões arquiteturais e por último, os testes de aceitação (que é onde é validada a satisfação do usuário, baseado na expectativa dele em relação as funcionalidades do sistema) (PRESSMAN;

Figura 6 – Fluxo do modelo de desenvolvimento *Scrum Solo*



Fonte: (PAGOTTO *et al.*, 28 jul. 2016)

MAXIM, 2016).

Os testes funcionais ou testes de caixa-preta, são testes voltados para as funcionalidades da aplicação, onde é verificado a acurácia dos resultados esperados em determinadas funcionalidades da aplicação. (PRESSMAN; MAXIM, 2016).

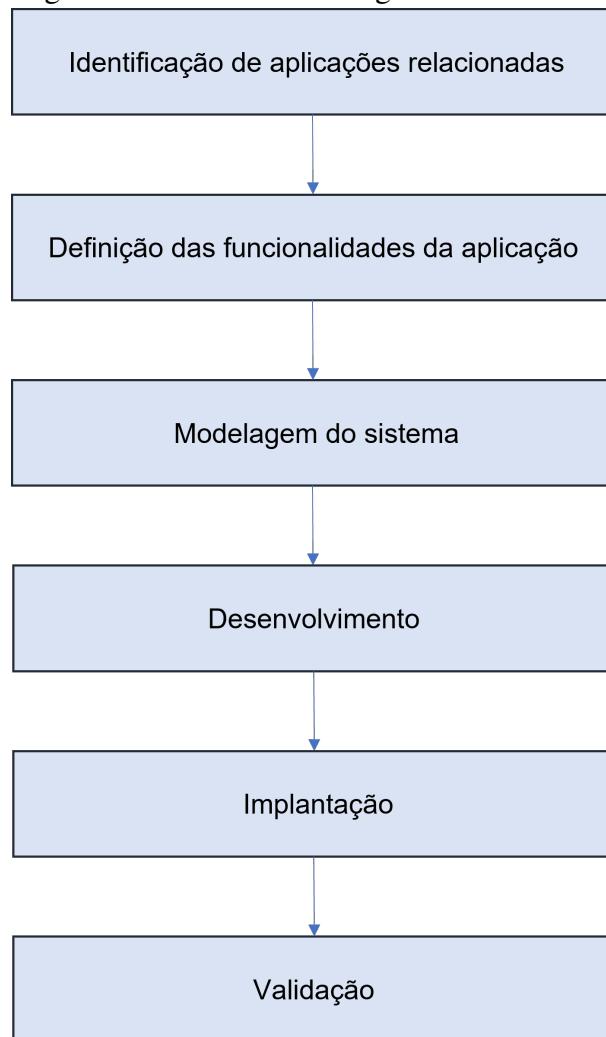
Os testes funcionais são aplicados no sistema proposto em uma fase de desenvolvimento já avançada, onde já é possível navegar durante o fluxo de submissão de atividades, cadastro de informações de universidade, usuários, categorias, metas e outras funcionalidades importantes.

No trabalho proposto a validação se faz importante também nos testes de validação, que são realizados propriamente com os usuários da aplicação. A aplicação desse teste se dá da seguinte forma: Os usuários respondem um questionário que contém perguntas distribuídas em 4 principais tópicos: Expectativa dos usuários em relação às funcionalidades, design das interfaces, experiência do usuário com a aplicação e fluidez e performance, do sistema.

4 PROCEDIMENTOS METODOLÓGICOS

Nesta seção serão apresentadas as etapas necessárias para a execução do trabalho e o planejamento de condução dessas tarefas. Com isso, a solução proposta nesse trabalho pode ser modelada, desenvolvida e validada. A Figura 7 ilustra a sequência de passos que foram utilizados na fase procedimento metodológico desse trabalho.

Figura 7 – Passos metodológicos



Fonte: Elaborado pelo autor.

4.1 Identificação de aplicações relacionadas

A primeira tarefa é buscar por aplicações de gerenciamento de ACs e AEs, com o objetivo de coletar as funcionalidades que cada uma possui em seu escopo. A principal aplicação utilizada como inspiração para o desenvolvimento da aplicação proposta foi o próprio

SIGAA, um sistema que, além de outras funções, permite o gerenciamento de ACs e AEs. Essa funcionalidade do mesmo foi ilustrada nas Figuras 2 e 3.

4.2 Definição das funcionalidades da aplicação

Após uma análise das aplicações semelhantes à proposta deste trabalho, é realizado um reconhecimento das funcionalidades comuns encontradas nessas aplicações. O objetivo dessa análise foi identificar as principais funcionalidades para compor o aplicativo de gerenciamento de ACs e AEs. Além disso, essa análise teve como objetivo identificar usabilidades mais interessantes presentes em outras aplicações. Ao fazer isso, buscou-se criar um escopo de funcionalidades e usabilidades que sejam imprescindíveis para a aplicação proposta, tornando-a mais atraente e intuitiva para os usuários.

Durante a análise, foram considerados os seguintes critérios para a seleção das funcionalidades:

- Relevância: As funcionalidades selecionadas devem ser relevantes para o gerenciamento eficiente de ACs e AEs, atendendo às necessidades dos usuários e aos objetivos do aplicativo.
- Usabilidade: As funcionalidades devem ser intuitivas e fáceis de usar, proporcionando uma experiência agradável para os usuários.
- Inovação: Buscou-se identificar funcionalidades inovadoras presentes em outras aplicações, que possam agregar valor e diferenciar o aplicativo no mercado.

Com base nessa análise, o Quadro 3 apresenta as principais funcionalidades que serão implementadas no aplicativo de gerenciamento de ACs e AEs.

É importante ressaltar que essa lista de funcionalidades foi uma lista preliminar e que passou por ajustes e refinamentos à medida que a fase de desenvolvimento do aplicativo foi avançando. A opinião e o feedback dos usuários também foram considerados para aprimorar e adicionar novas funcionalidades ao longo do tempo.

Essa definição inicial das funcionalidades serviu como base para a concepção de desenvolvimento do aplicativo, garantindo que as necessidades gerais dos usuários finais fossem atendidas e que a aplicação se tornasse competitiva no mercado.

Quadro 3 – Funcionalidades da Aplicação

Funcionalidade	Descrição
Cadastro e gerenciamento de AC	Permitir que os usuários cadastrem e gerenciem suas AC de forma organizada e eficiente.
Cadastro de categorias	Permitir que os usuários cadastrem categorias para as AC, de modo a facilitar a organização das AC e a busca por AC específicas.
Cadastro e visualização de progresso de metas	Permitir que os usuários cadastrem e visualizem metas para cumprimento de horas de AC, de modo a proporcionar um planejamento de execução das AC.
Exportação de AC	Possibilitar a exportação das AC em formato de PDF, para que os usuários possam submetê-las para as coordenadorias das IES.
Relatórios e métricas	Gerar relatórios e métricas sobre o progresso das AC, fornecendo uma visão geral do andamento das atividades.
Cadastro de informações referente a aluno, curso e instituição	Possibilitar que o aluno registre as informações da sua matrícula, curso e instituição

Fonte: Elaborado pelo autor.

4.3 Modelagem do sistema

Essa fase foi focada em definir os aspectos da aplicação que será desenvolvida. Os pontos considerados são: linguagem de programação, padrões de design, sistema operacional compatível, modelagem dos componentes e serviços para integração.

Com o escopo inicial da aplicação definido, foi definido os aspectos do desenvolvimento. Aspectos como definição e organização dos componentes com suas respectivas responsabilidades e as demais decisões técnicas de implementação dos requisitos inerentes as regras de negócios.

Por conta do fácil acesso ao ecossistema, tanto pelo desenvolvedor quanto pelo usuário, a aplicação foi desenvolvida para o sistema operacional *Android*, utilizando o Framework de desenvolvimento mobile *React Native*, que é uma ferramenta para desenvolvimento de aplicação híbridas. Isso facilitará, no futuro, o lançamento da aplicação para dispositivos da Apple, cujo sistema operacional é o *iOS*.

O *React Native* é uma tecnologia para desenvolvimento de aplicativos de forma híbrida (permite a reutilização do mesmo código para gerar aplicações para Android e iOS). Trata-se de uma biblioteca *JavaScript* lançada em 2015 pela *Meta Platforms, Inc* (na época chamada de *Facebook, Inc*), que facilita a criação de interfaces para aplicações móveis. A motivação para a escolha dessa ferramenta deu-se pela possibilidade de distribuição da aplicação para diferentes sistemas operacionais e também pelo conhecimento prévio que o autor do trabalho já tem sobre essa tecnologia.

A ferramenta utilizada para criação do código-fonte da aplicação é o *Visual Studio*

Code, que é também conhecido como *VS Code*. Trata-se de um editor de código-fonte bastante popular, feito pela *Microsoft*, para *Windows*, *Linux* e *macOS*. Os recursos incluem suporte para depuração, realce de sintaxe, conclusão de código inteligente, atalhos, refatoração de código e *Git* incorporado. Os usuários podem alterar o tema, atalhos de teclado, preferências e instalar extensões que adicionam funcionalidades adicionais.

Para a ilustração dos componentes, são utilizadas ferramentas para criação de diagramas. Diferentes diagramas são criados para representar aspectos no desenvolvimento da aplicação. Ilustrar e registrar essas decisões tem o objetivo de fazer com que a fase de desenvolvimento ocorra de forma fluída e concentrada, sem que haja a necessidade de tomar decisões ou de se preocupar com definições de técnicas cruciais durante essa fase.

Os artefatos entregáveis produzidos ao fim dessa etapa são as decisões técnicas de arquitetura e a modelagens do sistema. No Quadro 4 é possível visualizar a descrição das ferramentas escolhidas para execução desse trabalho.

Quadro 4 – Ferramentas

Nome	Tipo	Versão	Descrição
<i>React Native</i>	Biblioteca	0.70	Ferramenta em <i>Javascript</i> criada pela <i>Meta Platforms, inc</i> para desenvolver aplicativos para os sistemas <i>Android</i> e <i>iOS</i> de forma nativa.
<i>Node.js</i>	Editor de código	1.73.0	Ferramenta para edição de código-fonte
<i>Windows</i>	Sistema operacional	11 Pro	Sistema operacional para rodar as outras ferramentas
<i>Git</i>	Sistema de controle de versão	2.38.1	Ferramenta para controle de mudança de código e de diferentes versões
<i>Google Play</i>	Loja de aplicativos	-	Plataforma para a publicação de aplicações para dispositivos android

Fonte: Elaborado pelo autor.

4.4 Desenvolvimento

Após a definição dos requisitos, decisões técnicas de arquitetura e a modelagem do sistema serem definidas, junto a linguagem de programação, ambiente de desenvolvimento e outros aspectos da aplicação, foi iniciada a fase de desenvolvimento.

Essa etapa foi marcada pelo início do desenvolvimento do sistema. O escopo inicial levantado na fase anterior, serve com base para a concepção de requisitos mais específicos. Esses requisitos definidos e registrados, servem como fundamentos para a escrita do código-fonte e funcionalidades do aplicativo de gerenciamento de ACs e AEs.

Essa etapa foi concentrada não somente no desenvolvimento da aplicação, mas

também na validação, testes necessários e implantação da primeira versão do aplicativo.

O artefato produzido e entregue ao fim dessa etapa, foi a aplicação de gerenciamento de ACs e AEs pronta para ser lançada na loja online de aplicativos da google.

4.5 Implantação

Após a fase de desenvolvimento finalizar e o aplicativo estiver definido como pronto, foi gerada a versão de produção do aplicativo. Essa versão de produção foi disponibilizada na *Google Play*, onde o público-alvo, assim como os participantes da pesquisa de validação poderam baixar, instalar e usar o aplicativo.

4.6 Validação

A última etapa foi a de validação, onde o aplicativo proposto foi validado pelos usuários. O objetivo dessa validação foi medir o grau de satisfação dos usuários em sua experiência com a aplicação.

Foram avaliados aspectos como funcionalidade, design, usabilidade, desempenho e cumprimento daquilo que foi proposto.

Ao finalizar essa última etapa, foi gerado um artefato com os resultados da pesquisa mostrando o grau de satisfação dos usuários com o aplicativo.

As perguntas selecionadas para a pesquisa foram inspiradas e adaptadas do Questionário de Experiência do Usuário (*User Experience Questionnaire - UXQ*), que é um questionário utilizado para avaliar a usabilidade e a experiência do usuário. No UXQ, as perguntas são divididas em seis categorias principais: usabilidade, eficiência, satisfação, facilidade de uso, estética visual e confiabilidade.

A categoria de usabilidade avalia a facilidade com que o usuário consegue utilizar a aplicação. A categoria de eficiência avalia a rapidez e a eficiência com que o usuário consegue realizar suas tarefas. A categoria de satisfação avalia o nível de satisfação geral do usuário com a aplicação. A categoria de facilidade de uso avalia o quanto intuitiva e fácil de aprender a aplicação é para o usuário. A categoria de estética visual avalia a aparência visual e o design da aplicação. E a categoria de confiabilidade avalia a confiabilidade e a estabilidade da aplicação.

O questionário disposto no Apêndice A foi adaptado para a avaliação da aplicação de gerenciamento de ACs e AEs onde as perguntas foram selecionadas de acordo com o objetivo

da pesquisa. Diferentemente do UXQ, que utiliza uma escala de 1 a 7 para suas questões, as perguntas selecionadas para a pesquisa em questão utilizam a escala de *Likert*, que varia de 1 a 5. Nessa escala, 1 representa a avaliação mais baixa e 5 representa a avaliação mais alta para o ponto avaliado na questão.

As perguntas selecionadas para a pesquisa estão descritas no Quadro 5.

Quadro 5 – Avaliação da Aplicação

Pergunta	Objetivo
Como você avalia a facilidade de uso da aplicação? (De 1 a 5)	Avaliar a facilidade de uso e a experiência do usuário.
Como você avalia a aparência visual da aplicação? (De 1 a 5)	Avaliar o design e a estética da aplicação.
Como você avalia a velocidade e desempenho da aplicação? (De 1 a 5)	Avaliar a rapidez e o desempenho da aplicação.
Quais funcionalidades você mais utilizou na aplicação? (marque as opções)	Identificar as funcionalidades mais utilizadas pelos usuários.
Você usaria esta aplicação? (De 1 a 5)	Avaliar a intenção de uso da aplicação pelo usuário.
Você recomendaria esta aplicação para outros estudantes? (De 1 a 5)	Avaliar a satisfação e a probabilidade de recomendação da aplicação.
Você encontrou algum problema ou bug na aplicação? Se sim, qual foi? (opcional)	Identificar problemas ou bugs relatados pelos usuários.
Há alguma funcionalidade que você sentiu falta no aplicativo? Se sim, qual seria? (opcional)	Coletar sugestões de melhorias e novas funcionalidades.
Qual é a sua opinião geral sobre a aplicação? (opcional)	Permitir que os usuários expressem sua opinião geral sobre a aplicação.

Fonte: Elaborado pelo autor.

5 DESENVOLVIMENTO DA APLICAÇÃO DE ATIVIDADES COMPLEMENTARES

Nesta seção serão apresentados os detalhes do ciclo de vida desenvolvimento da aplicação proposta no presente trabalho. Primeiramente será contextualizado como se deu a definição das funcionalidades da aplicação. Em seguida será apresentada a modelagem do sistema, detalhes de implementação das funcionalidades com suas respectivas sprints e a implantação do sistema. Por fim, será apresentado o processo de validação e os resultados da pesquisa.

5.1 Análise das aplicações dos trabalhos relacionados para identificação de funcionalidades

Para a definição das funcionalidades da aplicação de ACs e AEs, foi realizada uma análise das aplicações dos trabalhos relacionados e também da funcionalidade de submissão de atividades do SIGAA, ilustradas nas Figuras 2 e 3. A análise foi realizada com o objetivo de identificar as funcionalidades que as aplicações possuem e que podem ser utilizadas na aplicação proposta. A análise foi realizada com base na descrição das funcionalidades das aplicações, que foram apresentadas nos artigos. A análise está detalhada na seção 2.4, na qual foram identificadas as funcionalidades das aplicações e as funcionalidades que podem ser utilizadas na aplicação proposta.

5.2 Definição das funcionalidades do aplicativo de Atividades Complementares

A partir da análise de aplicações similares, foram identificadas as funcionalidades necessárias para a aplicação de ACs e AEs. O autor, que também é um usuário da aplicação, contribuiu com sua visão ampla para a definição dessas funcionalidades. Para isso, foram consideradas as necessidades dos usuários e as funcionalidades das aplicações similares. Assim, as funcionalidades da aplicação foram definidas de forma a atender às necessidades dos usuários e oferecer uma experiência satisfatória.

No Quadro 6 apresenta a definição do *product backlog* (artefato mencionado na fundamentação teórica, subseção 3.4.3 do projeto, ou seja, as funcionalidades que a aplicação deve ter. O Quadro possui três colunas, sinalizando o tipo do requisito, o título do requisito e a descrição. A definição desses tipos de requisitos também foram abordados no Quadro 2

Quadro 6 – funcionalidades da aplicação proposta

Tipo	Título	Descrição
Funcional	Cadastrar usuário	Permite que o usuário se cadastre e tenha uma conta na aplicação.
Funcional	Login de usuário	Permite que usuário cadastrado acesse a sua conta na aplicação.
Funcional	Recuperar senha	Permite que o usuário consiga recuperar sua senha de acesso.
Funcional	Cadastrar ACs e AEs	Permite que o usuário registre a participação em uma ACs e AEs, preenchendo um formulário e anexando arquivo comprobatório.
Funcional	Editar ACs e AEs	Permite que o usuário edite uma ACs e AEs já cadastrada em sua conta.
Funcional	Exportar registros de ACs e AEs	Permite que o usuário exporte um ou vários registros de ACs e AEs para um único documento, no formato de PDF.
Funcional	Acompanhar atividades	Permite que o usuário visualize como estar seu progresso na submissão das horas.
Funcional	Cadastrar de metas	O usuário pode cadastrar metas de horas que ele deseja atingir para uma determinada categoria dentro de um determinado prazo.
Funcional	Cadastrar categorias	Permite que o usuário cadastre, em sua conta, categorias das ACs e AEs.
Funcional	Editar categorias	Permite que o usuário edite categorias já cadastradas em sua conta.
Funcional	Registrar dados do aluno	Permite que o usuário cadastre, em sua conta, dados do aluno.
Funcional	Editar dados do aluno	Permite que o usuário edite dados do aluno já cadastrados em sua conta.
Funcional	Registrar dados do curso e instituição	Permite que o usuário registre informações sobre o seu curso e instituição.
Não Funcional	Firebase	Usar banco de dados do firebase para registro informações sobre usuários.
Não Funcional	Segurança	O sistema deve ser protegido contra acesso não autorizado.
Não Funcional	Manutenção	O sistema deve ser fácil de manter e atualizar.
Não Funcional	Plataforma	O sistema deve ser compatível com o sistema operacional <i>Android</i> .

Fonte: Elaborado pelo autor.

5.3 Modelagem do sistema

Nesta seção, será apresentada a modelagem do sistema desenvolvido. Será abordada a estrutura de pastas adotada no projeto, que segue um padrão de organização de código comum em projetos *React Native* baseado no padrão de arquitetura de software MVC. Além disso, serão mencionadas as principais bibliotecas utilizadas durante o desenvolvimento da aplicação, destacando o papel fundamental que desempenharam no processo.

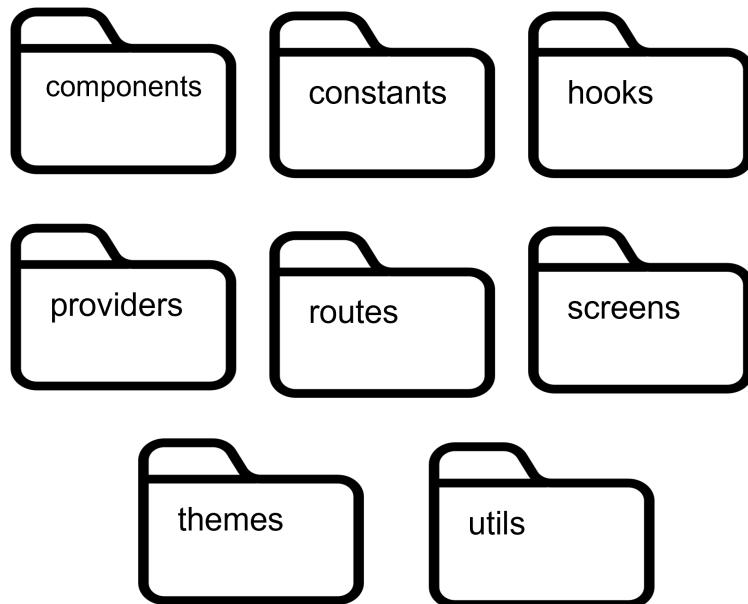
5.3.1 Estrutura de pastas

A estruturação das pastas do projeto foi feita de forma a facilitar a manutenção e a organização do código. Essa estrutura de pastas segue um padrão de organização de código comum em projetos React Native, que é baseado no padrão de arquitetura de software MVC. Nesse padrão, a pasta "screens" representa a camada de visualização (*View*), a pasta "components" representa a camada de controle (*Controller*) e a pasta "models"(que não foi

mencionada na atual estrutura) representaria a camada de modelo (*Model*).

A estrutura de pastas do projeto é apresentada na Figura 8. O Quadro 7 apresenta uma descrição mais detalhada de cada pasta.

Figura 8 – Estrutura de pastas do projeto.



Fonte: Elaborado pelo autor.

Quadro 7 – Estrutura de pastas do aplicativo.

Pasta	Descrição
<i>components</i>	Contém componentes reutilizáveis que são usados em várias partes do aplicativo, como um componente de botão personalizado ou um componente de lista.
<i>constants</i>	Contém constantes que são usadas em várias partes do aplicativo, como cores, tamanhos de fonte ou URLs de API.
<i>hooks</i>	Contém hooks personalizados que são usados em várias partes do aplicativo, como um hook para gerenciar o estado do formulário ou um hook para lidar com a autenticação.
<i>providers</i>	Contém provedores de contexto que são usados em várias partes do aplicativo, como um provedor de contexto de autenticação ou um provedor de contexto de tema.
<i>routes</i>	Contém as definições de rotas do aplicativo, como as rotas para a tela de login, a tela inicial e a tela de perfil.
<i>screens</i>	Contém as telas do aplicativo, como a tela de login, a tela inicial e a tela de perfil.
<i>themes</i>	Contém arquivos de estilo que são usados em várias partes do aplicativo, como estilos para botões, textos e fundos.
<i>utils</i>	Contém funções utilitárias que são usadas em várias partes do aplicativo, como funções para formatação de datas, validação de formulários ou manipulação de strings.

Fonte: Elaborado pelo autor.

5.3.2 Bibliotecas utilizadas

Durante o desenvolvimento da aplicação, foram utilizadas diversas bibliotecas que desempenharam um papel fundamental no processo de desenvolvimento. Algumas das principais bibliotecas utilizadas incluem o *react-native*, *expo*, *react-hook-form*, *yup*, *date-fns*, *expo-print*, *@react-navigation/native*, *@react-native-community/datetimepicker*, *@react-native-firebase/auth*, *@react-native-firebase/firestore* e *@react-native-firebase/storage*.

Essas bibliotecas foram selecionadas com base em sua popularidade, documentação abrangente e capacidade de atender aos requisitos específicos da aplicação. O Quadro 8 apresenta as bibliotecas utilizadas na aplicação e suas respectivas descrições.

Quadro 8 – Bibliotecas utilizadas na aplicação.

Biblioteca	Descrição
<i>React-Native</i>	Base do desenvolvimento da aplicação em React Native, fornecendo um conjunto de componentes e APIs para criar interfaces de usuário nativas para dispositivos móveis.
<i>Expo</i>	Plataforma que facilita o desenvolvimento de aplicativos móveis em React Native, fornecendo ferramentas e serviços para simplificar tarefas como configuração do ambiente de desenvolvimento e publicação nas lojas de aplicativos.
<i>React-hook-form</i>	Biblioteca para gerenciamento de formulários, permitindo definir regras de validação, controlar erros e exibir mensagens de erro personalizadas.
<i>Yup</i>	Biblioteca para validação de formulários, garantindo que os dados inseridos pelos usuários estejam de acordo com as regras definidas.
<i>@react-native-community/datetimepicker</i>	Biblioteca para seleção de datas em formulários.
<i>date-fns</i>	Biblioteca para manipulação e formatação de datas na aplicação.
<i>expo-print</i>	Biblioteca para funcionalidade de impressão, permitindo a geração de documentos em formato PDF.
<i>@react-navigation/native</i>	Biblioteca para implementar a navegação entre telas na aplicação.
<i>@react-native-firebase/auth</i>	Biblioteca para autenticação de usuários.
<i>@react-native-firebase/firestore</i>	Biblioteca para armazenamento, em núvem, de dados em tempo real.
<i>@react-native-firebase/storage</i>	Biblioteca para armazenamento, em núvem, de arquivos de certificados das ACs e AEs.

Fonte: Elaborado pelo autor.

5.4 Desenvolvimento em sprints

A seguir, serão apresentados os detalhes do desenvolvimento da aplicação proposta, detalhes de artefatos e eventos apresentados na seção sobre *Scrum Solo 3.4.3*, como *Sprint backlog*, execução da sprint, artefatos entregáveis ao fim das sprints. Também será detalhado as execuções das cinco sprints realizadas ao longo do processo.

Cada sprint teve a duração de uma semana. No Quadro 9, apresenta-se um resumo

das sprints realizadas. Em seguida, são apresentadas subseções detalhando cada uma delas.

Quadro 9 – Resumo das Sprints.

Sprint	Resumo da Sprint
1	Priorização do backlog, definição das sprints backlog, configuração do ambiente de desenvolvimento, criação do banco de dados, implementação do serviço de autenticação no Firebase, criação das telas do sistema com layouts básicos, implementação da navegação entre as telas.
2	Desenvolvimento de funcionalidades para fluxo de login, registro de dados, cadastro de atividades e upload de arquivos, cadastro de categorias e atualização dos dados do aluno, curso e instituição.
3	Desenvolvimento de funcionalidades para visualização das informações salvas no sistema, implementação da tela de acompanhamento do progresso de horas de AC.
4	Melhoria da experiência do usuário em relação à estilização da aplicação, desenvolvimento da funcionalidade de metas, adição de uma nova aba na tela inicial para listar atividades do tipo extensão.
5	Correção de bugs identificados na aplicação, remoção de necessidade de autenticação pelo usuário, testes finais para garantir o funcionamento correto da aplicação, preparação para publicação.

Fonte: Elaborado pelo autor.

5.4.1 Primeira sprint

Na primeira sprint, foram realizadas a priorização do *backlog* e a definição das *sprints backlog* a fim de estabelecer as funcionalidades prioritárias e o cronograma de desenvolvimento. Em seguida, foi criado o projeto e configurada a ferramenta de controle de versão *Git* para garantir a organização e o gerenciamento do código fonte.

Foram definidas as tecnologias a serem utilizadas e configurado o ambiente de desenvolvimento a fim de garantir a estabilidade e a eficiência do processo de desenvolvimento. Além disso, foi criado o banco de dados e implementado o serviço de autenticação no *Firebase* para garantir a segurança e a integridade dos dados do sistema.

Em seguida, foram criadas as telas do sistema com seus respectivos *layouts* básicos a fim de estabelecer a estrutura visual do aplicativo. A navegação entre as telas foi implementada para garantir a usabilidade e a experiência do usuário durante a utilização do aplicativo.

O principal artefato gerado foi um quadro com a priorização do *backlog*, que está disponível no Quadro 10. Não foram gerados capturas de tela dos demais artefatos desenvolvidos devido à sua natureza. Os mesmos consistiam principalmente em código fonte e configurações.

5.4.2 Segunda sprint

Durante a sprint 2 do projeto, foram desenvolvidas funcionalidades para garantir a segurança e a privacidade dos usuários do aplicativo, incluindo os fluxos de cadastro, login, logout e recuperação de senha. Além disso, foram implementadas funcionalidades para registro

Quadro 10 – Priorização das tarefas por Sprint.

Sprint	Descrição
1	<ul style="list-style-type: none"> – Priorização do backlog – Definição das sprints backlogs – Configuração do ambiente de desenvolvimento – Criação do banco de dados – Implementação do serviço de autenticação no Firebase – Criação das telas do sistema com layouts básicos – Implementação da navegação entre as telas
2	<ul style="list-style-type: none"> – Desenvolvimento de funcionalidades para fluxo de login – Registro de dados – Cadastro de atividades e upload de arquivos – Cadastro de categorias – Atualização dos dados do aluno, curso e instituição
3	<ul style="list-style-type: none"> – Desenvolvimento de funcionalidades para visualização das informações salvas no sistema – Implementação da tela de acompanhamento do progresso de horas de AC
4	<ul style="list-style-type: none"> – Melhoria da experiência do usuário em relação à estilização da aplicação – Desenvolvimento da funcionalidade de metas – Adição de uma nova aba na tela inicial para listar atividades do tipo extensão
5	<ul style="list-style-type: none"> – Exportar atividades cadastradas em um arquivo – Correção de bugs identificados na aplicação – Testes finais para garantir o funcionamento correto da aplicação – Preparação para publicação

Fonte: Elaborado pelo autor.

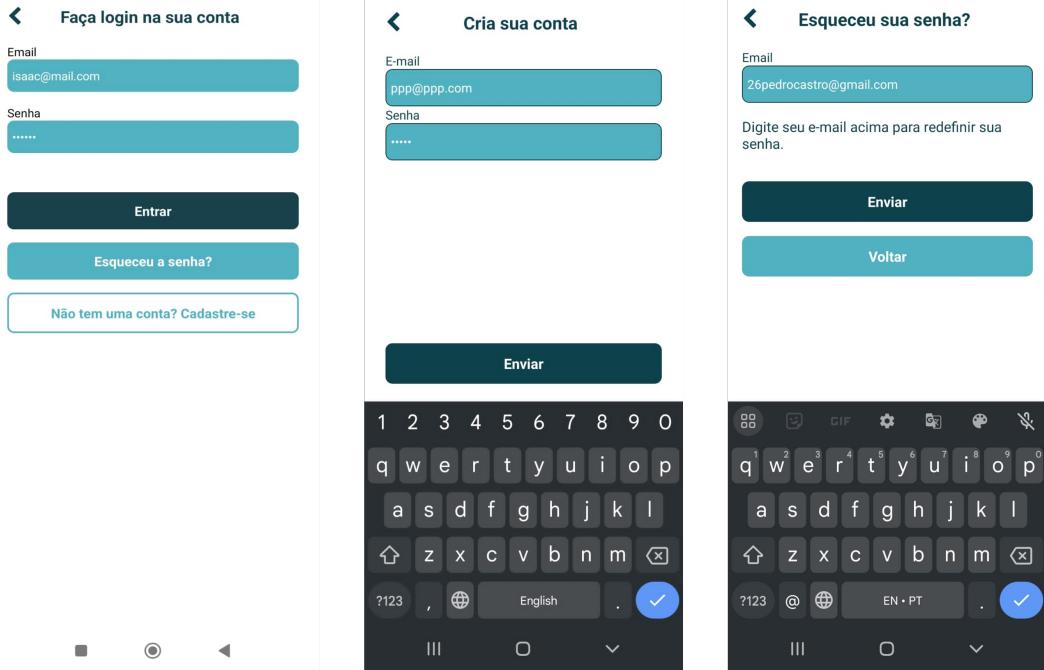
de dados, cadastro de atividades e upload de arquivos, permitindo que os usuários possam armazenar informações relevantes.

Também foram desenvolvidas funcionalidades para cadastro de categorias e atualização dos dados do aluno, curso e instituição. Vale destacar que a persistência desses dados é realizada no banco de dados configurado na sprint anterior.

A implementação dessas funcionalidades foi baseada nas prioridades definidas no backlog do produto e nas sprints backlogs, seguindo uma abordagem iterativa e incremental para o desenvolvimento do projeto.

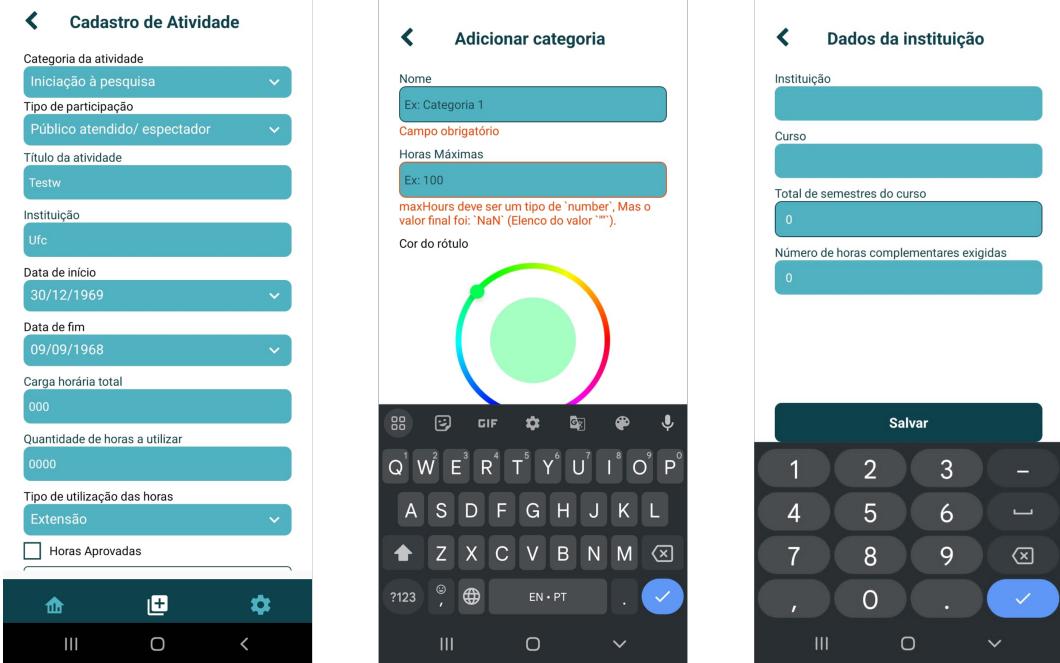
As Figuras 9 e 10 mostram os prints dos artefatos desenvolvidos durante a sprint 2 do projeto.

Figura 9 – Prints da sprint 2.



Fonte: Elaborado pelo autor.

Figura 10 – Prints da sprint 2.



Fonte: Elaborado pelo autor.

5.4.3 Terceira sprint

Durante a sprint 3 do projeto, foram desenvolvidas funcionalidades para permitir que o usuário possa visualizar as informações salvas no sistema. Foi implementada uma das telas principais da aplicação, a tela de acompanhamento do progresso de horas de ACs e AEs, que

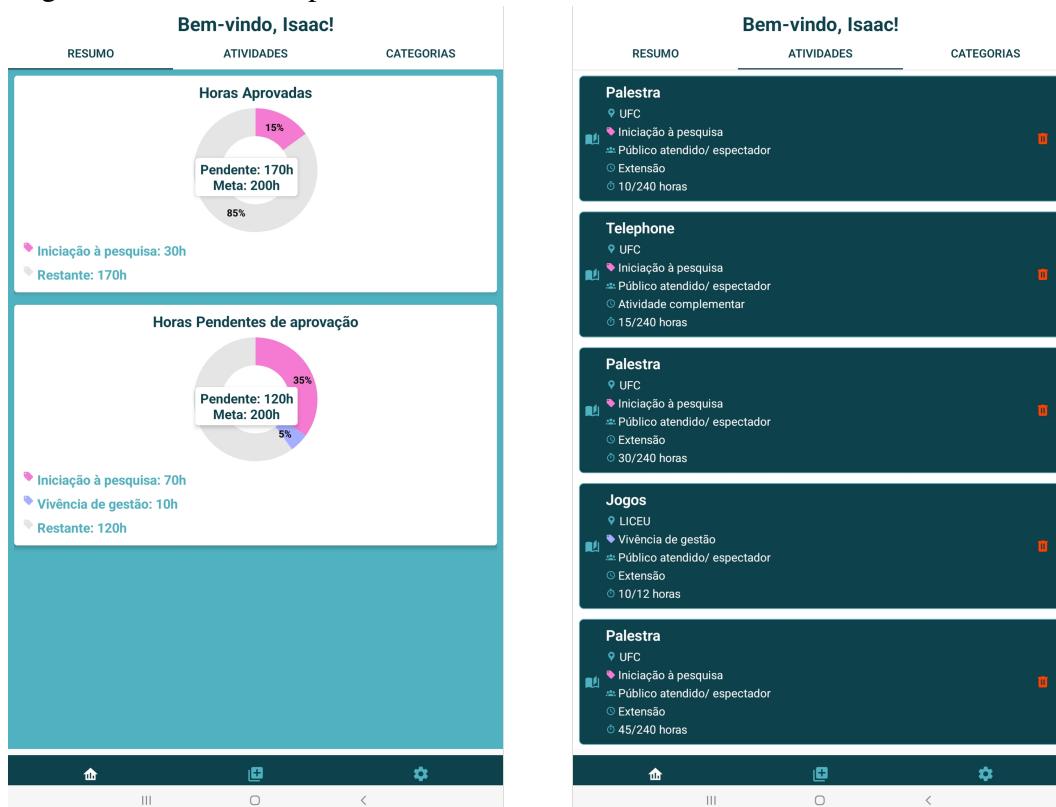
permite ao usuário acompanhar o seu progresso em relação às horas requeridas para a conclusão do curso. Essa tela foi desenvolvida com o objetivo de fornecer ao usuário uma visão geral do seu progresso em relação às horas de ACs e AEs requeridas para a conclusão do curso. Essa tela foi dividida em três abas: voltadas para atividades do tipo complementares, a primeira aba contém dois gráficos, um para atividades cadastradas e outro para atividades cadastradas e já aprovadas. Esses gráficos permitem ao usuário visualizar de forma clara e objetiva o seu progresso em relação às atividades cadastradas e aprovadas.

Na segunda aba, foi implementada a listagem das categorias que o aluno possui atividades. Essa funcionalidade permite ao usuário visualizar de forma organizada as atividades cadastradas em cada categoria, facilitando a gestão e o acompanhamento das atividades.

Na terceira aba, foi implementada a mesma funcionalidade da primeira aba, porém voltada para atividades de tipo extensão.

A Figura 11 mostra o print do artefato desenvolvido durante a sprint 3 do projeto.

Figura 11 – Prints da sprint 3.



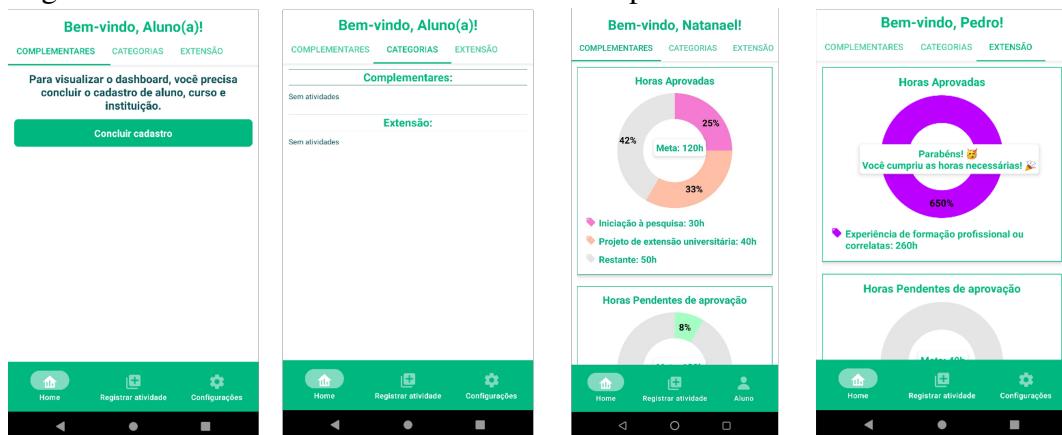
Fonte: Elaborado pelo autor.

5.4.4 Quarta sprint

Na sprint 4 do projeto, foram desenvolvidas funcionalidades para melhorar a experiência do usuário em relação à estilização da aplicação. Foram realizadas melhorias na estilização de botões, inputs, layouts, formulários, modal com loading, modal de feedback de sucesso ou falha ao salvar dados para tornar a interface mais atraente e intuitiva para o usuário. Foi desenvolvida também a funcionalidade de metas, que permite ao usuário definir metas de horas para cada categoria de atividade.

Em resumo, o objetivo dessa sprint foi garantir que a interface da aplicação fosse aprimorada e que novas funcionalidades importantes fossem implementadas para melhorar a experiência do usuário. As Figuras 12, 13 e 14 mostram os prints dos artefatos gerados ao fim da sprint 4.

Figura 12 – Prints do artefato desenvolvido na sprint 4.



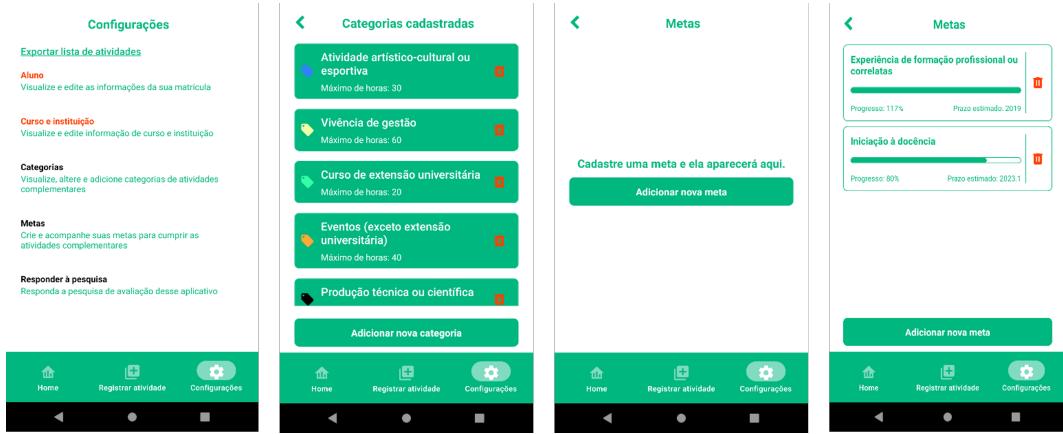
Fonte: Elaborado pelo autor.

Figura 13 – Prints do artefato desenvolvido na sprint 4.



Fonte: Elaborado pelo autor.

Figura 14 – Prints do artefato desenvolvido na sprint 4.



Fonte: Elaborado pelo autor.

5.4.5 Quinta sprint

Na sprint final do projeto, foi implementada a funcionalidade de exportação de atividades para um arquivo *PDF*. Nesse arquivo, cada atividade cadastrada é detalhada em um quadro. Há uma coluna com o link clicável do certificado anexado à atividade em questão, permitindo o download do mesmo. O artefato gerado por essa funcionalidade está ilustrado na Figura 15.

Figura 15 – Lista de atividades exportadas.

Informações do Estudante

Nome	Natanael Holanda
Matrícula	20182055323202
Instituição	Ifce
Curso	Engenharia civil
Horas requeridas	120

Atividades

Categoria	Tipo de participação	Título	Instituição	Uso das horas	Data de início	Data de término	Total de horas	Horas a serem usadas	Arquivo
Projeto de extensão universitária	Público atendido/ espectador	Pub atend	Ifce	Atividade complementar	06/03/2023	12/07/2023	40	40	Screenshot_20230703-094731.png
Iniciação à pesquisa	Público atendido/ espectador	Iniciação a pesquisa	Ifce	Atividade complementar	01/05/2023	03/07/2023	50	30	
Programa de extensão universitária	Público atendido/ espectador	Atend	Ifce	Atividade complementar	11/04/2023	13/07/2023	20	10	
Programa de extensão universitária	Membro da equipe de trabalho	Extensão	Ifce	Extensão	11/04/2023	14/07/2023	60	60	Screenshot_20230703-094731.png

Fonte: Elaborado pelo autor.

Além disso, foi dedicado um tempo para identificação e correção de bugs. Foi gerada uma versão de teste que foi utilizada pelo autor para usar a aplicação e identificar os bugs presentes na mesma. Foram feitos testes de caixa-preta ou testes funcionais, mencionados na subseção 3.4.4, para a estratégia de identificação de *bugs*. No Quadro 11 é possível visualizar os bugs identificados e a sua descrição.

Além de identificar *bugs*, esse processo permitiu identificar pontos de melhoria no aplicativo. Foi analisada a real necessidade da funcionalidade de login, visto que o usuário não precisaria se autenticar para utilizar o aplicativo. Após essa análise, foi decidido que o fluxo de login seria removido da aplicação, tendo em vista uma melhor experiência do usuário.

Quadro 11 – Bugs identificados na aplicação.

Bug	Descrição
Criação de categorias padrões duplicadas	Categorias padrões estão sendo duplicadas durante o processo de criação.
Imagen de carregamento do app	A imagem exibida durante o carregamento do aplicativo está incorreta.
Resetar formulário ao fim de cadastros	O formulário está sendo resetado automaticamente após a conclusão do cadastro.
Horas de extensão na aba de dashboard de extensão	As horas de extensão exibidas na aba de dashboard estão incorretas.
Texto incorreto no feedback ao salvar meta	O texto exibido no feedback ao salvar uma meta está incorreto.
Mensagem de erro em formulários - campos de números	Uma mensagem de erro é exibida ao preencher campos numéricos nos formulários.
Adicionar mensagem de sem atividades cadastradas ainda em tela de categorias	É necessário adicionar uma mensagem informando que não há atividades cadastradas na tela de categorias.
Adicionar mensagem de sem metas cadastradas ainda em tela de metas	É necessário adicionar uma mensagem informando que não há metas cadastradas na tela de metas.

Fonte: Elaborado pelo autor.

Com os *bugs* identificados, foi realizada a correção dos mesmos, garantindo que a aplicação estivesse funcionando corretamente e sem problemas para o usuário. Essa etapa foi fundamental para garantir a qualidade da aplicação e a satisfação do público alvo.

Com os *bugs* resolvidos, foram feitos alguns testes finais para garantir o funcionamento da aplicação e, assim, preparar a aplicação para publicação.

O artefato gerado nessa *sprint* foi a nova versão da aplicação, os quais se concentraram principalmente na resolução dos *bugs* citados anteriormente e a remoção do fluxo de login. Não houve alterações no layout em relação à sprint anterior. Desse modo, o objetivo dessa sprint

foi garantir que a aplicação estivesse funcionando corretamente e que estivesse pronta para ser publicada e utilizada pelos usuários.

5.5 Implantação

O processo de implantação de um aplicativo na loja de aplicativos da Google envolve diversas etapas que devem ser executadas em ambiente de desenvolvimento e na plataforma para desenvolvedores da loja. Uma das exigências da *Google Play* é que todos os aplicativos submetidos para a loja estejam digitalmente assinados com uma chave que informa dados sobre o desenvolvedor ou a empresa responsável. Para isso, é necessário executar um comando no terminal do sistema operacional que gera um arquivo no formato ".jks", adicioná-lo aos arquivos do aplicativo e referenciá-lo nos arquivos de configuração para a construção de novos aplicativos assinados com essa chave.

Após a adição da chave e das configurações nos arquivos de construção do aplicativo, é possível gerar novos aplicativos assinados com a chave criada. Para isso, foi utilizado o *Android Studio* - IDE oficial para desenvolvimento de aplicativos *Android*, oferecendo ferramentas e recursos para criar, depurar e testar apps móveis - para gerar o arquivo ".aab"(*Android App Bundle*). Esse arquivo é enviado para a loja de aplicativos *Google Play*, que se responsabiliza por extrair o arquivo enviado pelo desenvolvedor, gerando diferentes arquivos ".apk"(*Android Package*) para diferentes arquiteturas de celulares que recebem sua respectiva versão compatível para instalação quando um usuário acessa a página do aplicativo na loja, após sua disponibilização.

Antes de ser disponibilizado para os usuários, o aplicativo precisa passar por um conjunto de testes automatizados na *Google Play*. Esses testes avaliam diversos fatores, como o tempo de inicialização, o tamanho das opções clicáveis, a presença de falhas de inicialização e a conformidade com as políticas da loja. Além disso, é necessário enviar alguns artefatos para a loja, como o ícone do aplicativo, capturas de tela que mostram suas principais funções e textos que descrevem suas funcionalidades e oferecem uma visão mais detalhada sobre o que ele oferece ao usuário. Esses artefatos são exibidos na página do aplicativo na loja, permitindo que os usuários tenham uma ideia do que esperar do aplicativo antes de baixá-lo.

A *Google Play* também solicita que sejam respondidos alguns questionários sobre a categoria do aplicativo, conteúdo, palavras-chave relacionadas ao aplicativo, países onde ele estará disponível e sua classificação indicativa. Durante o processo de inserção de informações sobre as especificações do aplicativo, podem surgir algumas dificuldades para o preenchimento

dos dados, por conta da falta de familiaridade com a plataforma. Algumas mensagens de aviso são exibidas, exigindo a conclusão de outras etapas que se encontram em diferentes áreas da plataforma, o que pode levar o usuário novato a perder tempo navegando na plataforma.

Um outro ponto que também é exigido pela loja é a criação de uma Política de Privacidade do aplicativo. Essa política é um texto que descreve como os dados do usuário são coletados, armazenados e utilizados pelo aplicativo. É necessário que o desenvolvedor crie esse texto descrevendo as informações de política e privacidade do aplicativo e o envie para a loja, que o analisa e pode solicitar alterações caso seja necessário. Essa etapa também pode ser um pouco confusa para o desenvolvedor, pois é necessário que ele tenha conhecimento sobre a política de privacidade do aplicativo e como ele funciona para poder criar um documento que descreva essas informações. No entanto, o autor da aplicação proposta utilizou o site *privacy policy template*¹, que disponibiliza um modelo de política de privacidade para aplicativos, que pode ser utilizado como base para a criação da política de privacidade do aplicativo. Esse documento deve ser fornecido em formato de site e a URL deve ser cadastrada nas configurações do aplicativo na loja. A URL para a política de privacidade do aplicativo proposto é a seguinte: <https://atividades-complementare.web.app/privacy-policy.html>.

Outra exigência é que as imagens precisam ter um tamanho específico, seja para ícone ou banner interno da loja.

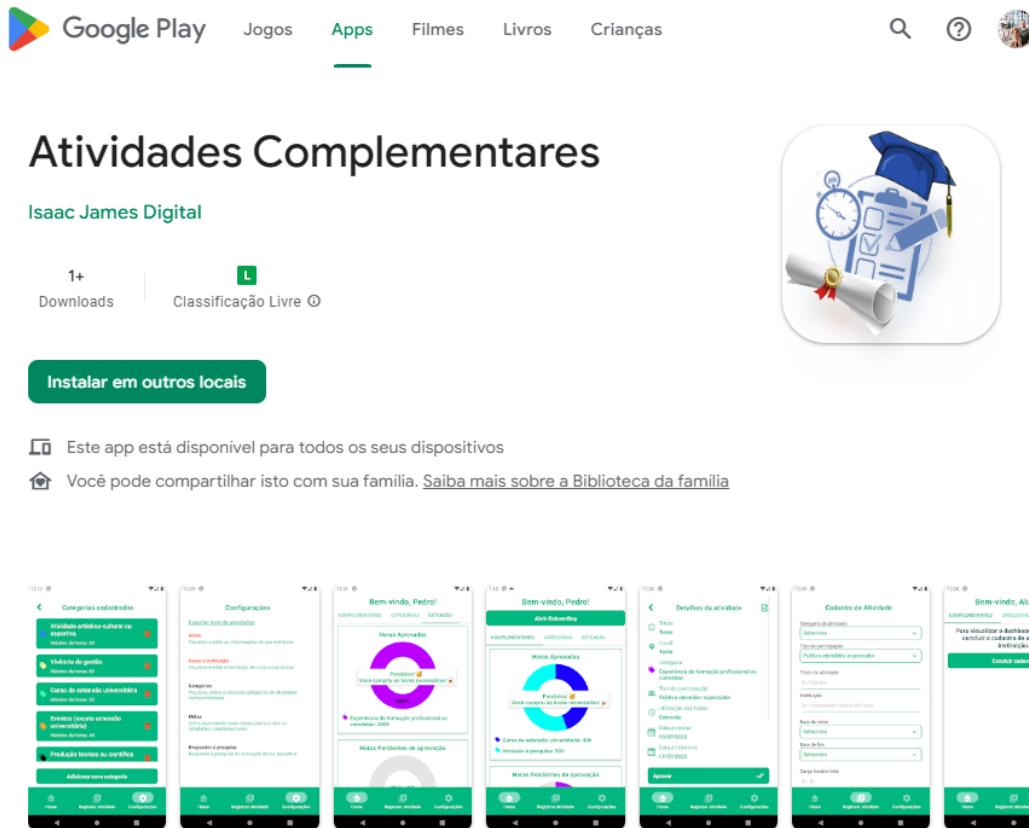
Após o envio de todos os pontos requeridos pela loja, o aplicativo passa por uma análise final que pode demorar até 7 dias, segundo a loja. Após essa análise, o aplicativo é liberado para os usuários e está disponível para download na *Google Play*.

A primeira versão do aplicativo em questão foi disponibilizado na *Google Play* em 16 de junho de 2023 com o nome "Atividades Complementares". A página do aplicativo na loja pode ser acessada pelo link <https://play.google.com/store/apps/details?id=com.isaacjames.tccatvdcomp>. A Figura 16 mostra a página do aplicativo na loja.

5.6 Validação

Para avaliar a experiência dos usuários com a aplicação, abrangendo diferentes aspectos como usabilidade, aparência visual, desempenho, funcionalidades e satisfação geral, foi realizada uma pesquisa com base em perguntas definidas previamente no Quadro 5. Essas perguntas têm como objetivo avaliar a facilidade de uso da aplicação, quais funcionalidades foram mais utilizadas, a aparência visual da aplicação, a velocidade e desempenho da aplicação,

Figura 16 – Bibliotecas utilizadas no projeto.



O que há de novo

- Adiciona modal de apresentação do app;
- Resolve bugs.

Fonte: Elaborado pelo autor.

se há alguma funcionalidade que o usuário sentiu falta, se encontrou algum problema ou bug na aplicação, se recomendaria a aplicação para outras pessoas e qual é a opinião geral sobre a mesma.

As perguntas do questionário disposto no Apêndice A foram adaptadas de modelos já existentes, como o User Experience Questionnaire¹, que é um questionário utilizado para avaliar a experiência dos usuários no uso de aplicações. A maioria das perguntas seguem um padrão consistente que avaliam diferentes aspectos da aplicação. Como por exemplo na questão 2, "Como você avalia a aparência visual da aplicação? (De 1 a 5)", onde o usuário pode avaliar a aplicação de 1 a 5, sendo 1 a pior nota e 5 a melhor nota. Essa questão avalia a aparência visual da aplicação, que é um aspecto importante para a experiência do usuário, pois a aparência visual pode influenciar na decisão de instalar ou continuar a usar o aplicativo ou não.

¹ <https://www.ueq-online.org/>

Para a realização da pesquisa, foi utilizada uma ferramenta de criação de formulários dinâmicos, o Google Forms da Google. O link da pesquisa² foi disponibilizado tanto na aplicação quanto em fóruns da internet e em grupos de aplicativos de mensagem para que qualquer pessoa com interesse na aplicação respondesse o questionário de forma voluntária.

A pesquisa ficou aberta para respostas de 01 de julho de 2023 até 05 de julho de 2023, coletando 19 respostas, onde as mesmas foram analisadas para avaliar o grau de satisfação dos usuários e aspectos de qualidade da experiência em relação à proposta entregue pela aplicação.

Além de coletar os pontos referentes a validação da aplicação, o propósito da pesquisa é coletar feedbacks dos usuários, identificar pontos fortes e pontos fracos, sugestões de melhorias e detectar possíveis falhas ou problemas que possam ser corrigidos. Com essas informações, é possível aprimorar a aplicação e oferecer uma experiência cada vez melhor aos usuários.

² <https://forms.gle/2FQMTMXhWNaAEGE76>

6 RESULTADOS

Nesta capítulo são apresentados os resultados da pesquisa realizada com os usuários do aplicativo de Atividades Complementares.

6.0.1 Resultados da pesquisa

A pesquisa foi dividida em nove perguntas, sendo uma pergunta de múltipla escolha, três de escala de likert, duas objetivas e três abertas. Nas Figuras 17, 18 e 19 são apresentados os resultados das perguntas da escala de likert, onde foi questionado sobre usabilidade, aparência e facilidade de uso da aplicação.

De modo geral, as perguntas receberam avaliações positivas, onde a maioria dos usuários avaliaram a aplicação com notas 4 e 5. No entanto, as perguntas sobre aparência da aplicação e facilidade de uso merecem uma atenção a mais. Essas perguntas receberam, respectivamente, quatro votos 3 e um voto 3. Isso pode indicar que há pontos de melhorias sobre a aparência e usabilidade da aplicação.

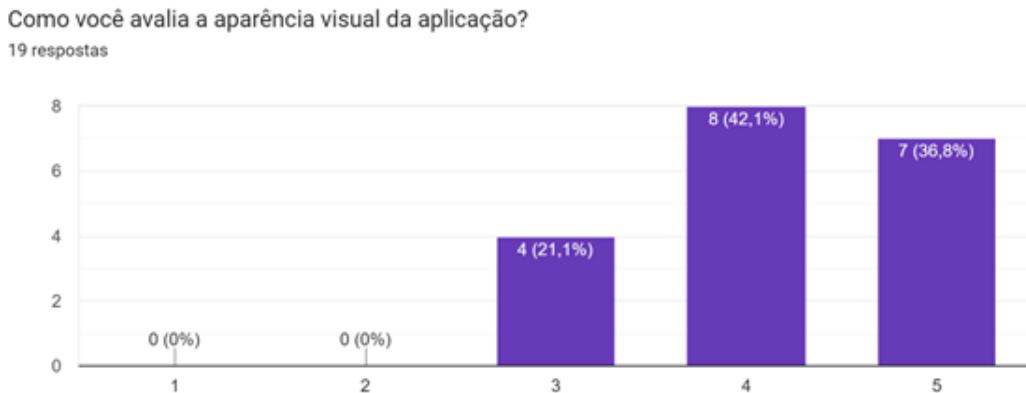
Figura 17 – Como você avalia a facilidade de uso da aplicação?



Fonte: Elaborado pelo autor.

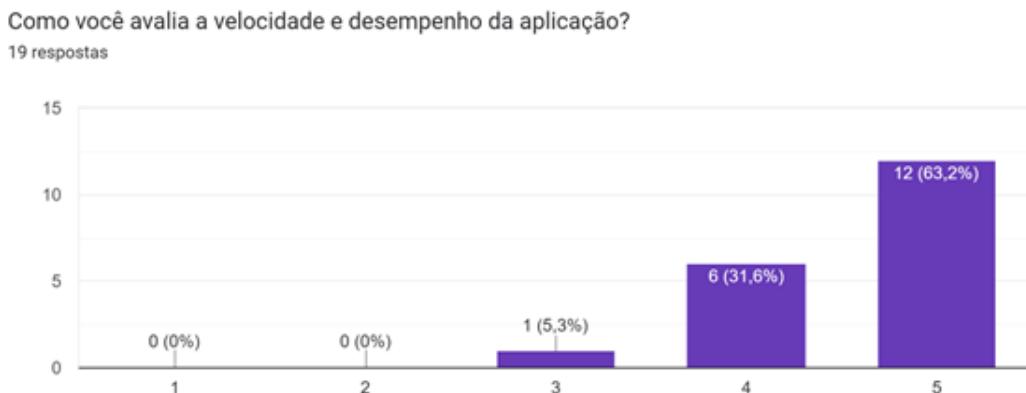
A única pergunta de múltipla escolha feita, foi sobre as funcionalidades mais utilizadas pelos usuários. A Figura 20 apresenta os resultados dessa pergunta. A funcionalidade de cadastro de atividade foi a mais utilizada, seguida de cadastro de categoria e exportação das atividades cadastradas. Esses resultados são esperados, pois essas funcionalidades são as principais do aplicativo. A funcionalidade de anexação de arquivos não recebeu nenhum voto, o que indica que essa funcionalidade não é utilizada pelos usuários. Isso pode ser explicado

Figura 18 – Como você avalia a aparência visual da aplicação?



Fonte: Elaborado pelo autor.

Figura 19 – Como você avalia a velocidade e desempenho da aplicação?



Fonte: Elaborado pelo autor.

pelo fato de que ou o aplicativo não foi utilizado para fins acadêmicos, mas sim para fins de avaliação ou que a funcionalidade não está em destaque o suficiente para os usuários conseguirem encontrá-la. Apesar de não ter sido utilizada pelos usuários da pesquisa, essa funcionalidade foi mantida e é considerada importante para o aplicativo e deve ser melhorada para que os usuários possam utilizá-la.

As perguntas objetivas foram feitas para avaliar a opinião dos usuários sobre o aplicativo. A Figura 21 apresenta os resultados dessas perguntas. Todos os usuários responderam que sim para ambas, utilizariam e recomendariam a aplicação para outros estudantes. Esses resultados são positivos, podendo indicar uma consistência de usuários ativos e uma tendência para novos usuários utilizarem a aplicação.

As respostas das três questões de escopo aberto, direcionadas para a identificação

Figura 20 – Quais funcionalidade você mais utilizou na aplicação?



Fonte: Elaborado pelo autor.

Figura 21 – Você usaria esta aplicação?



Fonte: Elaborado pelo autor.

de comportamento inesperado, falhas e pontos de melhoria na aplicação, podem contribuir para ajudar a entender quais pontos levaram a uma avaliação com notas baixas nas questões de escopo fechado. Para uma melhor avaliação dessas respostas, as mesmas foram organizadas e distribuídas em três categorias: bugs, usabilidade, funcionalidades.

O Quadro 12 apresenta as respostas subjetivas categorizadas. A categoria bug apresenta falhas que impedem o uso da aplicação. A categoria usabilidade apresenta pontos de melhorias que não impedem o uso da aplicação, mas que dificultam o uso da aplicação, deixando-a menos intuitiva e proporcionando uma experiência ruim para o usuário. A categoria funcionalidades apresenta funcionalidades que os usuários gostariam de ver implementadas no aplicativo.

De modo geral, a pesquisa mostra que a aplicação recebeu notas positivas da maioria do público alvo. Os resultados mostram que a aplicação foi avaliada como útil e objetiva. No

Quadro 12 – Respostas subjetivas categorizadas

Categorias	fallhas ou pontos de melhoria
Bug	falha ao exibir certificado anexado em atividade. Error 404.
Usabilidade	Loading demorado ao atualizar informações de aluno.
Usabilidade	No geral o app atende o esperado. Para versões futuras recomendo trazer um estilo mais amigável, com cores diferentes e que tenham uma sinergia e ícones para categorias, atividades, funcionalidades. Por último, dar uma atenção maior na questão de usabilidade, procurando deixar o processo de registro de horas mais simples, rápido e prático, visando o seu público principal que considero ser os jovens de 17 a 24 anos.
Usabilidade	Tem pontos de melhoria, especialmente visuais, mas é uma aplicação útil e que pode ajudar bastante no processo de cadastro e acompanhamento das atividades.
Usabilidade	A aplicação é funcional, mas precisa de alguns ajustes no design: 1 - Na tela de cadastro de atividade ou na tela de aluno não é necessário exibir o botão de "voltar", pois não houve uma navegação para uma nova tela, o usuário apenas trocou de aba. 2 - Na tela de "Aluno" a cor do título das opções não estão padronizadas, as duas primeiras estão em vermelho e as outras estão em preto.
Funcionalidades	Seria muito bom caso houvesse a possibilidade de integração com o SIGAA.
Funcionalidades	Uma sessão de apoio explicando de maneira fácil como funcionam as horas complementares. A regulamentação, horas que podem ser aprovadas em cada categoria e outras particularidades
Funcionalidades	Envio de notificação semanal para lembrar da meta seria ótimo.
Funcionalidades	Uma feature que complementaria bastante e facilitaria minha vida de estudante seria o aplicativo listar atividades do campus que acontecesse e que se enquadrasse em alguma categoria que ainda não complementei minhas horas.

Fonte: Elaborado pelo autor.

entanto, os resultados também mostram que a aplicação precisa de melhorias, especialmente em sua usabilidade. É possível fazer uma relação entre as notas baixas, em relação a usabilidade, e as respostas subjetivas que apontam para pontos de melhoria nas experiências do usuário. A aplicação precisa ser melhorada para que o usuário tenha uma experiência mais agradável e intuitiva. Os resultados também mostraram que os usuários gostariam de ver funcionalidades adicionais implementadas no aplicativo. A implementação dessas funcionalidades seria interessante para que o usuário tenha uma experiência mais completa e que atenda suas necessidades.

7 CONSIDERAÇÕES FINAIS

Neste capítulo, serão apresentadas as considerações finais do trabalho, Incluindo suas limitações e sugestões para trabalhos futuros.

Em suma, as ACs e AEs desempenham um papel fundamental na formação acadêmica e profissional dos alunos. Elas proporcionam uma oportunidade única para os estudantes expandirem seus conhecimentos além da sala de aula, adquirindo habilidades práticas e interdisciplinares que enriquecem seu perfil e os preparam para os desafios do mercado de trabalho.

Ao longo deste trabalho, foi possível compreender a diversidade de cenários e categorias das ACs e AEs, bem como os desafios enfrentados pelos alunos na gestão e controle dessas atividades. Foi identificada a necessidade de uma solução tecnológica eficiente para auxiliar os estudantes nesse processo organizacional e de acompanhamento dessas atividades.

O objetivo principal deste trabalho foi desenvolver uma aplicação móvel para o gerenciamento das ACs e AEs em IES e documentar o processo de desenvolvimento dessa aplicação. Os resultados alcançados foram satisfatórios com a implementação de funcionalidades relevantes e a aplicação sendo bem recebida pelos usuários.

A avaliação da aplicação por meio de uma pesquisa com os usuários revelou pontos positivos, como a utilidade e objetividade da aplicação. No entanto, também foram identificados pontos de melhoria, como bugs, usabilidade e funcionalidades adicionais desejadas pelos usuários. Esses pontos podem ser abordados em trabalhos futuros, visando aprimorar a experiência do usuário e tornar a aplicação ainda mais completa.

Este trabalho demonstrou as fases de desenvolvimento de uma aplicação móvel, desde a concepção até a publicação na loja de aplicativos da Google. Passando por áreas vistas em disciplinas do curso de Engenharia de Software, como análise de requisitos, arquitetura de software, desenvolvimento de software, testes e entre outras. A aplicação foi desenvolvida utilizando a metodologia *Scrum Solo*, que foi aplicada de forma eficiente e contribuiu para o sucesso do projeto.

Em conclusão, este trabalho demonstrou a importância das ACs e AEs, apresentou uma solução tecnológica para o gerenciamento dessas atividades, avaliou a aplicação desenvolvida e identificou áreas de melhoria. Espera-se que essa aplicação contribua para facilitar o processo de gestão das ACs e AEs, beneficiando tanto os alunos quanto as IES.

7.1 Limitações

Nesta seção, são apresentadas algumas limitações importantes que podem afetar o trabalho realizado na aplicação. A primeira limitação é o armazenamento grátis limitado fornecido pelo Firebase, que é usado como banco de dados para a aplicação. É importante estar ciente desse limite e considerar estratégias para gerenciar eficientemente o armazenamento de dados, como a exclusão de registros antigos ou a implementação de um sistema de arquivamento.

Outra limitação é que a aplicação só cadastra e carrega dados quando está conectada à internet. Isso pode ser um desafio em situações em que a conectividade é instável ou inexistente. Nesses casos, pode ser necessário explorar soluções alternativas, como o armazenamento local temporário dos dados e a sincronização automática assim que a conexão for restabelecida.

Além disso, é importante destacar que a aplicação não possui integração com sistemas de coordenadorias de IES para sincronia de dados. Isso pode limitar a capacidade de obter informações atualizadas e precisas. Nesses casos, é importante buscar alternativas para garantir a precisão dos dados, como a implementação de processos manuais de atualização ou a exploração de APIs disponíveis para obter informações relevantes.

Uma limitação sobre a pesquisa é a falta de dados demográficos. Não foi obtido informações como idade, curso, semestre, instituição, entre outros. Isso pode ter afetado a qualidade dos resultados, pois não foi possível identificar diferenças entre os grupos de usuários. Para trabalhos futuros, é importante considerar a obtenção de dados demográficos para uma análise mais completa e eficiente.

7.2 Trabalhos Futuros

Para trabalhos futuros, é possível sugerir a implementação de novas funcionalidades, pesquisas para avaliar e identificar melhorias na aplicação desenvolvida, como:

- Integração tanto com SIGAA, para UFC, quanto para outros sistemas de outras IES, para sincronização automática dos dados dos cursos, ACs, AEs e alunos. Essa integração permitiria uma atualização precisa e automática das informações, eliminando a necessidade de inserção manual e reduzindo possíveis erros de dados.
- Inserção da funcionalidade de importação/exportação de categorias. Isso permitiria aos usuários importar e exportar categorias de ACs e AEs, facilitando o compartilhamento e a reutilização dessas categorias entre diferentes usuários ou sistemas.

- Criação de uma tela informativa no aplicativo para auxiliar os usuários a se familiarizarem com as ACs e AEs. Essa tela poderia fornecer orientações sobre requisitos, prazos e critérios de avaliação das ACs e AEs, além de recursos e dicas para maximizar a participação dos usuários.
- Criação de uma funcionalidade de notificação semanal para lembrar os usuários de suas metas. Essa funcionalidade poderia ser configurada pelo usuário para lembrá-lo de suas metas semanais, facilitando o acompanhamento e o controle das ACs e AEs.
- Criação de uma funcionalidade para acompanhamento de eventos do campus. Essa funcionalidade poderia listar eventos do campus que se enquadrasssem em alguma categoria de ACs ou AEs, facilitando a participação dos usuários em eventos que contribuam para o cumprimento de suas metas.

REFERÊNCIAS

- BARROS, M. F.; SILVA, A. W. G. da; OLIVEIRA, G. R. d. C. **Desenvolvimento de um sistema para gerenciamento de atividades complementares**. 16 set. 2021. Disponível em: <https://dspace.uniceplac.edu.br/handle/123456789/918>. Acesso em: 24 jul. 2023.
- BASS, L.; CLEMENTS, P.; KAZMAN, R. **Software architecture in practice**: 3. ed. [S. l.]: Addison-Wesley Professional, 2012. ISBN 9780321815736.
- BRASIL. **Parecer CNE/CES nº 136/2012, aprovado em 9 de março de 2012**. 2012. Disponível em: https://normativasconselhos.mec.gov.br/normativa/view/CNE_PAR_CNECESN1362012.pdf?query=CURRICULARES. Acesso em: 11 05. 2021.
- LECHETA, R. R. **Google Android - Aprenda a criar aplicações para dispositivos Móveis com o Android**. [S. l.]: Novatec, 2009. ISBN 9788575221860.
- LEONCIO, D. **Sistema para Controle de Estágio Supervisionado do Departamento Acadêmico de Informática**. 2018. Trabalho de Conclusão de Curso (Tecnólogo em Análise e Desenvolvimento de Sistemas), UTFRP (Universidade Tecnológica Federal do Paraná), Paraná, Brasil. Disponível em: <http://repositorio.utfpr.edu.br/jspui/handle/1/16809>. Acesso em: 24 jul. 2023.
- LIENTZ, B. P.; SWANSON, E. B.; TOMPKINS, G. E. Characteristics of application software maintenance. **Commun. ACM**, Association for Computing Machinery, New York, NY, USA, v. 21, n. 6, p. 466–471, jun 1978. ISSN 0001-0782. Disponível em: <https://doi.org/10.1145/359511.359522>.
- LIMA, P. G. de; CORREIA, L. H.; SILVA, C. A. G. da. Implementação de um sistema multi-plataforma para gerenciamento de atividades complementares em cursos superiores utilizando banco de dados nosql. **Revista de Informática Aplicada**, v. 14, n. 2, out. 2018. Disponível em: https://seer.uscs.edu.br/index.php/revista_informatica_aplicada/article/view/6935.
- PAGOTTO, T.; FABRI, A. J.; L'ERARIO, A.; GONÇALVES, J. A. **Scrum solo: Software process for individual development**. 28 jul. 2016. Disponível em: <https://ieeexplore.ieee.org/document/7521555>. Acesso em: 24 jul. 2023.
- PRESSMAN, R.; MAXIM, B. **Engenharia de Software**: 8. ed. [S. l.]: McGraw Hill Brasil, 2016. ISBN 9788580555349.
- SOMMERVILLE, I. **Engenharia De Software**. [S. l.]: PEARSON BRASIL, 2019. ISBN 9788543024974.
- TOLEDO, P. D. A. de; SIRQUEIRA, T. F. M. **HORAS COMPLEMENTARES: Um aplicativo de gestão de atividades acadêmicas**. 20 jul. 2021. Disponível em: https://www.researchgate.net/publication/353246767_HORAS_COMPLEMENTARES_UM_APPLICATIVO_DE_GESTAO_DE_ATIVIDADES_ACADEMICAS. Acesso em: 24 jul. 2023.
- WASSERMAN, T. **Software engineering issues for mobile application development**. 07 Nov. 2010. Disponível em: <https://dl.acm.org/doi/abs/10.1145/1882362.1882443>.
- WIEGERS, K.; BEATTY, J. **Software Requirements**. [S. l.]: Microsoft Press, 2013. v. 3. ISBN 9780735679665.

APÊNDICE A – QUESTIONÁRIO DE AVALIAÇÃO DA APLICAÇÃO

Pesquisa de avaliação do aplicativo Atividades Complementares

Olá, seja bem vindo(a) !!

Me chamo Isaac James, sou aluno do curso de Engenharia de Software da Universidade Federal do Ceará e estou realizando essa pesquisa para o meu trabalho de conclusão de curso. Esta pesquisa tem como objetivo coletar informações sobre a avaliação do aplicativo de Atividades Complementares. O aplicativo tem como propósito auxiliar os usuários no cadastro, gerenciamento e acompanhamento de atividades complementares realizadas durante o curso.

O Aplicativo pode ser encontrado aqui: <https://play.google.com/store/apps/details?id=com.isaacjames.tccatvdcomp>

Sua participação é extremamente importante para o desenvolvimento deste trabalho acadêmico. As respostas fornecidas serão tratadas de forma confidencial e anônima, sendo utilizadas apenas para fins de pesquisa.

Agradeço antecipadamente por dedicar seu tempo para responder a este questionário. Sua contribuição será fundamental para a conclusão deste trabalho.

Obrigado pela sua colaboração!

* Indica uma pergunta obrigatória

1. E-mail *

CONSENTIMENTO DA PARTICIPAÇÃO COMO SUJEITO

Antes de responder a pesquisa, preciso que você esteja ciente do termo de consentimento e condições abaixo:

1. PARTICIPANTES DA PESQUISA: Pode participar da pesquisa qualquer pessoa que utilizou o aplicativo disponível no link acima, que deseje colaborar e esteja de acordo com o presente termo.
2. ENVOLVIMENTO NA PESQUISA: Caso você concorde em participar, você realizará a atividade de preenchimento de questionário online, que dura em média 3 minutos. Você é livre para deixar a pesquisa a qualquer momento, deixar de participar após iniciar não lhe trará penalidade.
3. CONFIDENCIALIDADE DA PESQUISA: Os pesquisadores tratarão a sua identidade com padrões profissionais de sigilo, atendendo a legislação brasileira (Resolução N° 466/12 do Conselho Nacional de Saúde).
4. BENEFÍCIOS E CUSTOS: Para participar deste estudo você não vai ter nenhum custo, nem receberá qualquer vantagem financeira.

Compreendi perfeitamente tudo o que me foi informado sobre a minha participação e estou consciente de que clicando em "Próxima", ACEITO PARTICIPAR DESTA PESQUISA VOLUNTARIAMENTE.

2. Como você avalia a facilidade de uso da aplicação? *

Marcar apenas uma oval.

Péssimo

1

2

3

4

5

Excelente

3. Como você avalia a aparência visual da aplicação? *

Marcar apenas uma oval.

Péssimo

1

2

3

4

5

Excelente

4. Como você avalia a velocidade e desempenho da aplicação? *

Marcar apenas uma oval.

Péssimo

1

2

3

4

5

Excelente

Seção 2: Utilização e Opiniões sobre a Aplicação

5. Quais funcionalidades você mais utilizou na aplicação? *

Marque todas que se aplicam.

- Cadastro de atividade
- Anexo de arquivos para certificado
- Cadastro de categoria
- Exportação das atividades cadastradas

6. Você usaria esta aplicação? *

Marcar apenas uma oval.

- Sim
- Não

7. Você recomendaria esta aplicação para outros estudantes? *

Marcar apenas uma oval.

Sim

Não

8. Você encontrou algum problema ou bug na aplicação? Se sim, qual foi?

9. Há alguma funcionalidade que você sentiu falta no aplicativo? Se sim, qual seria?

10. Qual é a sua opinião geral sobre a aplicação?

Agradecimentos!

Gostaria de agradecer imensamente pelo seu tempo dedicado a responder essa pesquisa. Sua resposta será fundamental para a contribuição do meu TCC e aplicação em questão.

Atenciosamente, Isaac James!