



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE  
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA  
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

**CILAS MEDEIROS DE FARIAS MARQUES**

**COMPUTACAO@UFCG: UM CRM PARA A COORDENAÇÃO DO  
CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO DA  
UFCG**

**CAMPINA GRANDE - PB**

**2023**

**CILAS MEDEIROS DE FARIAS MARQUES**

**COMPUTACAO@UFCG: UM CRM PARA A COORDENAÇÃO DO  
CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO DA  
UFCG**

**Trabalho de Conclusão Curso  
apresentado ao Curso Bacharelado em  
Ciência da Computação do Centro de  
Engenharia Elétrica e Informática da  
Universidade Federal de Campina  
Grande, como requisito parcial para  
obtenção do título de Bacharel em  
Ciência da Computação.**

**Orientador: Professor Dr. Francisco Vilar Brasileiro.**

**CAMPINA GRANDE - PB**

**2023**

**CILAS MEDEIROS DE FARIAS MARQUES**

**COMPUTACAO@UFCG: UM CRM PARA A COORDENAÇÃO DO  
CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO DA  
UFCG**

**Trabalho de Conclusão Curso  
apresentado ao Curso Bacharelado em  
Ciência da Computação do Centro de  
Engenharia Elétrica e Informática da  
Universidade Federal de Campina  
Grande, como requisito parcial para  
obtenção do título de Bacharel em  
Ciência da Computação.**

**BANCA EXAMINADORA:**

**Professor Dr. Francisco Vilar Brasileiro  
Orientador – UASC/CEEI/UFCG**

**Professora Dr. Reinaldo César de Moraes Gomes  
Examinador – UASC/CEEI/UFCG**

**Professor Tiago Lima Massoni  
Professor da Disciplina TCC – UASC/CEEI/UFCG**

**Trabalho aprovado em: 14 de fevereiro de 2023.**

**CAMPINA GRANDE - PB**

## **ABSTRACT**

Customer Relationship Management is a widely adopted business strategy, with the objective of improving the interaction between a company and its customers. Its implementation uses software tools to compile, store and analyze customer information, enabling the company to better understand who they are, how they interact with the brand and what services they want. In recent years, some Higher Education Institutions have studied the use of this practice, due to its great potential to influence academic management and student experience. Therefore, the COMPUTACAO@UFCG system was designed in order to improve communication, engagement and connections between course participants, providing automation of the most common tasks, time optimization, error mitigation and reduction of resources involved in the course processes.

# COMPUTACAO@UFCG: Um CRM para a Coordenação do Curso de Graduação em Ciência da Computação da UFCG

**Cilas Medeiros De Farias Marques**  
Universidade Federal de Campina Grande  
cilas.marques@ccc.ufcg.edu.br

**Francisco Vilar Brasileiro**  
Universidade Federal de Campina Grande  
fubica@computacao.ufcg.edu.br

## RESUMO

O Customer Relationship Management é uma estratégia de negócios amplamente adotada, com o objetivo de melhorar a interação entre uma empresa e seus clientes. Sua implementação utiliza ferramentas de software para compilar, armazenar e analisar informações sobre os clientes, permitindo que a empresa compreenda melhor quem são, como interagem com a marca e quais serviços desejam. Nos últimos anos, algumas Instituições de Ensino Superior têm estudado o uso dessa prática, devido ao seu grande potencial de influenciar a gestão acadêmica e a experiência dos estudantes. Diante disso, o sistema COMPUTACAO@UFCG foi concebido, a fim de melhorar a comunicação, o envolvimento e as conexões entre os participantes do curso, proporcionando automatização de tarefas usuais, otimização de tempo, mitigação de erros e redução de recursos envolvidos nos processos do curso.

## PALAVRAS-CHAVE

COMPUTAÇÃO@UFCG, CRM, UFCG, Atividades complementares, Sistema de gestão

## 1. INTRODUÇÃO

A Gestão de Relacionamento com o Cliente (Customer Relationship Management - CRM) é uma estratégia de negócios amplamente adotada, que visa melhorar o relacionamento entre uma organização e seus clientes. Ela envolve a integração de processos, tecnologias e abordagens projetadas para gerenciar e aprimorar as interações com os usuários. A implementação de um sistema de CRM pode fornecer à organização informações valiosas sobre o comportamento do cliente, além de automatizar tarefas de rotina e melhorar a comunicação e a colaboração entre diferentes departamentos.

Devido ao potencial desse método, nos últimos anos algumas Instituições de Ensino Superior têm explorado o seu uso [1][2][3]. Um modelo de CRM bem estruturado pode acarretar em um impacto significativo na gerência acadêmica e conseqüentemente na experiência dos estudantes. Esse fator acaba tendo extrema relevância em

Os autores retêm os direitos, ao abrigo de uma licença Creative Commons Atribuição CC BY, sobre todo o conteúdo deste artigo (incluindo todos os elementos que possam conter, tais como figuras, desenhos, tabelas), bem como sobre todos os materiais produzidos pelos autores que estejam relacionados ao trabalho relatado e que estejam referenciados no artigo (tais como códigos fonte e bases de dados). Essa licença permite que outros distribuam, adaptem e evoluam seu trabalho, mesmo comercialmente, desde que os autores sejam creditados pela criação original.

cenários onde é estabelecido um compromisso de longo prazo e onde a harmonia dos envolvidos é fundamental para o crescimento mútuo.

Possibilitando a automação de tarefas usuais, o CRM é muito eficaz para a adaptação de serviços, otimização de tempo, mitigação de erros e redução de recursos envolvidos nos processos de gerência universitária. Além disso, essa estratégia permite obter informações valiosas sobre as necessidades e preferências dos estudantes, auxiliando no aprimoramento da comunicação e na tomada de decisões acadêmicas [4][5].

Neste contexto, o presente trabalho tem por objetivo apresentar o sistema de gestão de relacionamento com clientes COMPUTAÇÃO@UFCG. Concebido para a Unidade Acadêmica de Sistemas e Computação (UASC) da Universidade Federal de Campina Grande (UFCG). Seu propósito é aprimorar a comunicação, o engajamento e as relações entre os integrantes da unidade, inicialmente focado nos alunos de graduação. Com esta finalidade, foi considerado o cenário para declaração do processo formal de aproveitamento de atividades complementares desses alunos.

## 2. MOTIVAÇÃO

A declaração de atividades complementares envolve muitas informações e serviços diretamente associados à eficiência de seus processos. Nesta seção detalhamos o que são as atividades complementares e quais seus requisitos; em seguida explicamos como a declaração delas funciona atualmente; e por fim listamos requisitos que uma nova solução precisa atender.

### 2.1 ATIVIDADES COMPLEMENTARES E SEUS REQUISITOS

As Atividades Complementares Flexíveis visam o desenvolvimento de habilidades e competências do aluno, inclusive adquiridas fora do ambiente acadêmico. Elas abrangem estudos e atividades independentes, transversais, interdisciplinares, de permanente contextualização e atualização para o enriquecimento do perfil do formando [6].

No curso de Bacharelado em Ciência da Computação da UFCG/CEEI/UASC, a declaração das atividades complementares flexíveis é obrigatória para a integralização curricular, possuindo uma carga horária máxima de 330 horas (22 créditos). Os tipos de atividades válidas estão

descritos na resolução N° 01/2022 estabelecida pelo Colegiado do Curso de Graduação em Ciência da Computação da UFCG e engloba atividades como estágios, participação em programas institucionais, projetos de pesquisa e desenvolvimento, eventos, entre outros. O aproveitamento da carga horária é contabilizado dependendo da natureza da atividade, podendo ser disposta em horas, meses ou semestres letivos.

## 2.2 DECLARAÇÃO DAS ATIVIDADES

Para entender melhor o problema, podemos nos aprofundar no procedimento necessário para o aproveitamento das atividades. O aluno deve encaminhar uma solicitação formal à coordenação do curso, via SEI [7], para que suas atividades sejam validadas. Seguindo os critérios, unidades e limites definidos na resolução N° 01/2022 citada anteriormente, a documentação deve conter a devida comprovação e carga horária acerca das tarefas exercidas.

Atualmente esse procedimento é feito de forma manual através da confecção de um pedido formal de aproveitamento de atividades complementares contendo as seguintes informações:

- Tabela contendo a listagem e características das atividades realizadas.
- Documentos comprobatórios de todas as atividades enumeradas.

É necessário que o pedido de aproveitamento seja devidamente submetido ao relator e, posteriormente, aprovado pelo colegiado para que seu reconhecimento seja válido. Vale salientar que essa validação não pode ser realizada de forma paulatina, à medida que as atividades são cumpridas. Isso gera não só atrasos, mas também insegurança para os alunos, que não têm certeza de que as atividades serão contabilizadas e integralizadas até que o processo que as contém seja totalmente analisado.

## 2.3 REQUISITOS DE UMA NOVA SOLUÇÃO

Diante disso, junto ao coordenador do curso, foram identificadas necessidades a serem atendidas com o uso do sistema COMPUTAÇÃO@UFCG, o que gerou a seguinte lista de requisitos:

- RF01: O estudante deve conseguir autenticar-se no aplicativo mobile através da sua conta google @ccc.
- RF02: A coordenação do curso deve conseguir autenticar-se na interface web através da sua conta google @computacao ou computacao.ceei@graduacao.ufcg.edu.br.
- RF03: O estudante deve ter acesso a um guia informando os passos para validar suas atividades.
- RF04: O estudante deve poder solicitar a validação de suas atividades individualmente, à medida que as mesmas sejam concluídas.

- RF05: Um e-mail deve ser enviado ao coordenador quando a solicitação de validação de uma atividade for criada.
- RF06: A solicitação de validação da atividade do estudante deve ser atribuída a um revisor pela coordenação.
- RF07: Um e-mail deve ser enviado ao revisor quando uma solicitação de validação de atividade for atribuída a ele.
- RF08: O revisor deve poder aprovar ou negar a solicitação. Solicitações negadas precisam de uma justificativa, enquanto as aprovadas exigem a definição dos créditos aproveitados.
- RF09: Um e-mail deve ser enviado ao estudante quando sua atividade for validada ou negada.
- RF10: Ao completar a carga de 22 créditos, o estudante deve conseguir gerar o processo formal de aproveitamento de atividades complementares flexíveis.
- RF11: A coordenação do curso deve ser capaz de verificar se o processo final submetido pelo aluno foi gerado pelo sistema COMPUTAÇÃO@UFCG.
- RF12: Com base na natureza da atividade, a unidade da carga horária de uma atividade deve ser definida automaticamente durante o registro.

## 3 ARQUITETURA

### 3.1 VISÃO GERAL

Os componentes que constituem a arquitetura de software da plataforma COMPUTAÇÃO@UFCG, bem como suas interações são ilustrados na Figura 1.

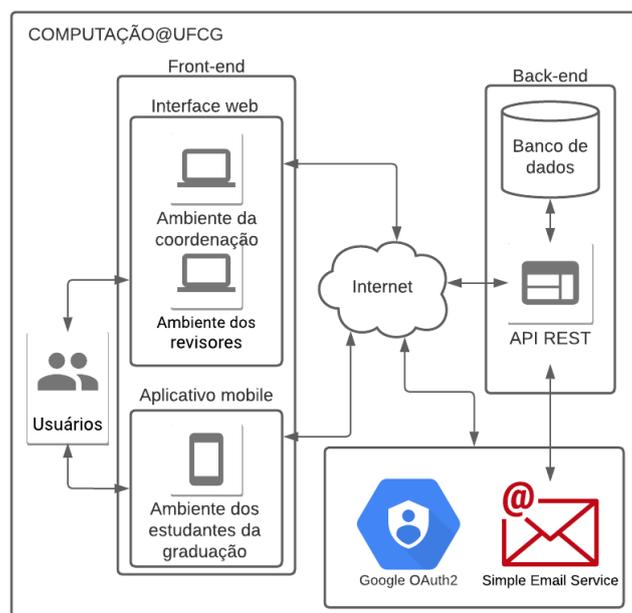


Figure 1. Arquitetura da solução.

O projeto possui 3 tipos de usuário cada um com particularidades em relação aos níveis de acesso e permissões no sistema. Seus perfis e características são as seguintes:

- **Coordenação:** Possui como principal ação o gerenciamento da plataforma. Esse usuário tem conhecimento de todos os revisores cadastrados na aplicação. Dessa forma, sendo o responsável pela atribuição das tarefas registradas no sistema.
- **Revisor:** Orientado aos docentes da instituição, a principal ação desse usuário é desempenhar tarefas que lhe são atribuídas (e.g. averiguar a correteza das informações presentes nos registros das atividades complementares flexíveis).
- **Estudante:** Seu papel é fazer uso do sistema para gerenciar sua experiência e interação com a instituição educacional. Isso pode incluir declarar e solicitar informações acadêmicas, melhorar seu contato com professores e funcionários, ter acesso a recursos e informações de apoio, enviar feedback para a instituição e muito mais.

A interface web é exclusiva para a coordenação e os revisores do curso. Já o aplicativo mobile é destinado aos estudantes da graduação. Ambos os ambientes tem suas requisições endereçadas ao back-end.

A API REST é o componente encarregado por receber e tratar as requisições do front-end, padronizando o acesso e manipulando os recursos através da rede. Ela permite a comunicação entre diferentes serviços e plataformas de forma fácil e eficiente. Duas das suas principais funcionalidades nesse sistema são o envio de e-mail e a comunicação com o banco de dados.

O banco de dados serve como um repositório central de informações e pode ser usado para uma ampla variedade de propósitos. Sua principal função é fornecer uma maneira eficiente e confiável de armazenar, acessar e gerenciar grandes quantidades de dados. Ele também inclui uma variedade de ferramentas e recursos, como linguagens de consulta e ferramentas de relatório, que permitem a fácil recuperação e análise dos dados armazenados.

### 3.2 APLICAÇÃO NO PROBLEMA

O uso da plataforma é restrita aos usuários ligados à UASC. Para tal intuito, após a autenticação, o domínio do email utilizado é validado pelo ambiente acessado. Nesse passo é identificada a permissão de acesso do usuário e se o mesmo é associado à UASC. A interação de cada usuário com o programa é mostrada no diagrama de casos de uso da Figura 2.

Acessando o aplicativo mobile, o estudante poderá registrar suas atividades, uma de cada vez. Para isso, é preciso preencher um formulário anexando o documento de comprovação e informando o tipo, a carga horária e a descrição da atividade. Para assegurar a consistência dos dados, antes de concluir o envio, é exibido um resumo do conteúdo inserido e o estado inicial do registro. Após a finalização, o registro é salvo no sistema, podendo ser

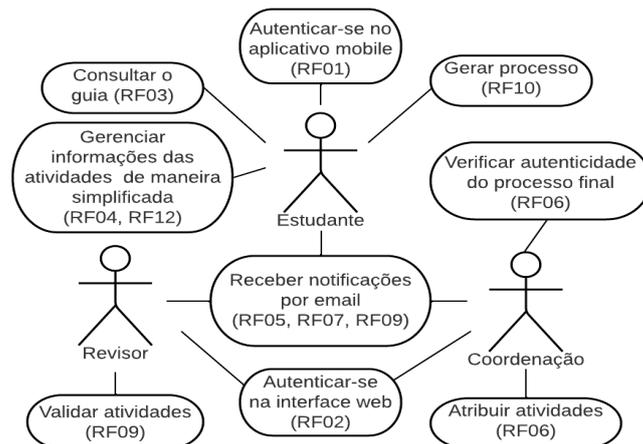


Figure 2. Casos de uso.

consultado a qualquer momento. Caso haja alguma falha durante a submissão, uma mensagem de erro é retornada e exibida ao usuário.

As atividades adicionadas ao banco de dados são configuradas com o estado inicial Created. Elas iniciam um processo no qual sempre que o estado da tarefa é atualizado, os usuários responsáveis pelo próximo passo são notificados por email. A sequência de estados segue o fluxo apresentado na Figura 3.

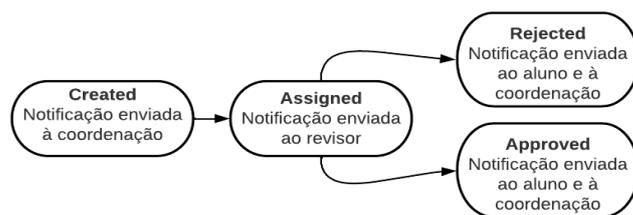


Figure 3. Fluxo de notificações.

Ingressando na interface web, a coordenação tem elencada todas as atividades complementares catalogadas no banco de dados. O coordenador pode atribuir a tarefa de validação dos registros com o estado Created aos revisores cadastrados no sistema. Essa ação faz com que a API vincule o email do revisor à atividade e atualize seu estado para Assigned.

No ambiente do revisor, são listadas apenas as atividades que lhe foram concedidas. Esse usuário deve realizar a validação da tarefa e definir uma avaliação final. Para isso, são dispostas as opções de aprovação e rejeição, atualizando o estado das atividades para Approved e Rejected, respectivamente. Em casos aprovados, é requisitada a definição da quantidade de créditos a serem aproveitados. Já para situações de rejeição, é demandado o motivo da rejeição.

Após a aprovação das atividades e a obtenção dos 22 créditos, o estudante poderá solicitar a geração do processo formal de aproveitamento de atividades complementares pelo aplicativo mobile. A API é a encarregada

pela criação automática do arquivo, retornando um PDF contendo as seguintes informações:

- Nome, Email e Matrícula do aluno
- Tabela com a listagem das atividades realizadas
- Sequência dos documentos comprobatórios, de todas as atividades enumeradas, com seus respectivos revisores.

Finalizada a geração do PDF é calculado o checksum do arquivo a fim de auxiliar a verificação de integridade do documento. Por fim, o processo é automaticamente baixado no celular do estudante.

No ambiente da coordenação é possível submeter o processo do aluno a uma verificação de integridade. Nesse passo é averiguada a igualdade do checksum entre o arquivo submetido e o registro existente no banco de dados. Caso haja alguma divergência, é retornada uma resposta ao front-end informando sobre a inconsistência do PDF.

## 4. IMPLEMENTAÇÃO

O código fonte do sistema foi disponibilizado no github, nos seguintes repositórios:

- <https://github.com/cilasmarques/ufcg-comp-app-api>
- <https://github.com/cilasmarques/ufcg-comp-app-web>
- <https://github.com/cilasmarques/ufcg-comp-app-mobile>

### 4.1 AUTENTICAÇÃO

Para a autenticação dos usuários foi utilizado o serviço OAuth2 do Google [8]. Com essa abordagem, os usuários podem autorizar o acesso às suas informações (e.g. email, nome) sem compartilhar suas credenciais de login. Isso é feito através de um fluxo de autorização, no qual o usuário é redirecionado para uma página de login do provedor de recursos (o Google) e, em seguida, autoriza o acesso ao aplicativo. O OAuth2 é uma norma aberta para autorização que permite que os clientes concedam acesso limitado aos seus recursos de forma segura a terceiros, sem compartilhar suas credenciais. Ele é amplamente utilizado em aplicativos e serviços na web para autorizar o acesso a contas do usuário.

### 4.2 API REST

A API é baseada no padrão MVC (Model-View-Controller), onde cada componente tem uma função específica na arquitetura da aplicação. Sua estrutura principal é dividida 4 partes:

- Routes: utilizado na gerência das rotas da aplicação, definindo quais ações devem ser tomadas em cada endpoint acessado.
- Controllers: encarregados do gerenciamento, averiguação e tratamento dos dados.
- Middlewares: responsável pela conexão de serviços externos à aplicação, tais como envios de email.

- Models: Onde são representados os dados e as regras de negócio da aplicação. É nesse componente onde são gerados os documentos PDF.

A tecnologia escolhida para seu desenvolvimento foi o Flask [9] devido sua facilidade de uso, desempenho, escalabilidade e segurança. O serviço de envio automático de emails opera por meio da biblioteca SMTPLib [10] do Python. Ela possibilita enviar e-mails através do protocolo SMTP (Simple Mail Transfer Protocol), oferecendo suporte a vários provedores de email, incluindo o Gmail, Yahoo, Outlook e Hotmail.

### 4.3 BANCO DE DADOS

Acerca da persistência dos dados, optou-se pela adoção de um modelo de banco de dados relacional, dada a natureza dos dados[11]. O PostgreSQL[12] é considerado um dos melhores sistemas de gerenciamento de banco de dados relacional open source, pois oferece uma ampla variedade de recursos avançados, como suporte a múltiplos usuários, extensões, suporte a procedimentos armazenados, suporte a vários tipos de dados, tratamento de erros, garantias de transação, controle de acesso, etc. Ele se tornou extremamente popular por causa de sua consistência, segurança e desempenho. Além disso, o PostgreSQL é altamente escalável, permitindo-o lidar com grandes conjuntos de dados e cargas de trabalho. Na Figura 4 podemos visualizar a estrutura inicial do banco.

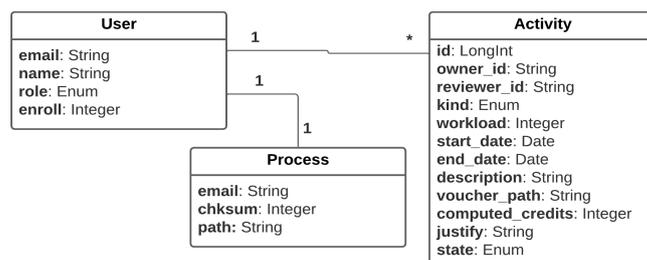


Figure 4. Estrutura do banco de dados.

### 4.4 A INTERFACE WEB

#### 4.4.1 TECNOLOGIAS

A implementação da interface web foi realizada com a utilização da biblioteca ReactJs [13], desenvolvida em JavaScript e mantida pelo Facebook. A seleção desta biblioteca foi orientada pela experiência do autor em projetos que utilizam o ReactJs, assim como pela estrutura baseada em componentes, que possibilita uma maior eficiência no desenvolvimento e expansão da aplicação.

#### 4.4.2 TELAS E FUNCIONALIDADES

A Figura 5 mostra como é a tela de listagem das atividades complementares na visão do coordenador. O ambiente do revisor é semelhante, contando com as opções NEGAR e VALIDAR ao invés do botão de ATRIBUIR; sem a visualização da coluna Revisor e sem a presença da Verificação de Documentos na barra lateral. Na Figura 6 é possível visualizar o ambiente para verificação do processo formal de aproveitamento de atividades complementares.

Usuário	Estado	Tipo	Período de tempo	Descrição	Comprovação	Revisor	Opções
cilas.marques@ccc.ufcg.edu.br	Criado 25/01/2023	Participação em Monitoria Reconhecida Institucionalmente pela UFCG.	1 semestre(s)	Monitoria de ICC	monitoria-2021.1.pdf	cilas.marques@ccc.ufcg.edu.br	ATRIBUIR

Figure 5. Tabela de atividades complementares

Figure 6. Tabela de atividades complementares

## 4.5 O APLICATIVO MOBILE

### 4.5.1 TECNOLOGIAS

Uma das ferramentas mais populares para desenvolvimento de aplicações mobile é o React Native, que permite criar aplicações nativas para iOS e Android usando JavaScript e React. No entanto, o gerenciamento de configurações e dependências pode ser complexo e demorado. Por esse motivo, para o desenvolvimento desse componente foi utilizada a biblioteca de código aberto React Native [14] em conjunto com a plataforma Expo [15].

A Expo é uma plataforma open-source que fornece uma abstração sobre o React Native, tornando o desenvolvimento mais fácil e rápido. Ela inclui recursos como push notifications, acesso à câmera, gerenciamento de pacotes, uma interface de linha de comando para construir e testar aplicações além de permitir a execução de aplicações diretamente no dispositivo sem precisar de configurações adicionais.

### 4.5.2 TELAS E FUNCIONALIDADES

O aplicativo mobile visa auxiliar o estudante, centralizando informações e deixando os processos da graduação mais intuitivos.

Para a entrada no sistema o usuário faz login na interface apresentada na Figura 7(A), sendo encaminhado para a tela principal das Atividades Complementares (Figura 7(B)) após a autenticação.

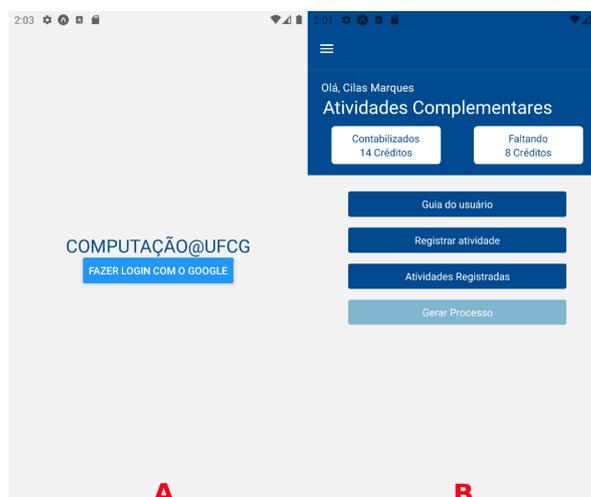


Figure 7. Telas iniciais da aplicação.

A tela principal (Figura 7(B)) sumariza a quantidade de créditos contabilizados e remanescentes de acordo com os cadastros feitos pelo usuário. Somado a isso, é disposto um botão referente à geração do processo, o qual só é habilitado após o contador de créditos faltantes chegar em zero (Figura 8(A)). Para a geração do processo, é necessário que o aluno informe sua matrícula (Figura 8(B)) antes de finalizar a requisição.

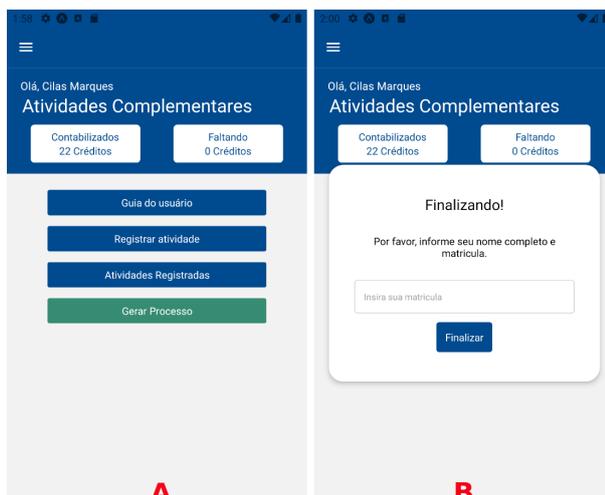


Figure 8. Telas de geração de processo.

Em vez de realizar manualmente o registro de todas as atividades, o aluno pode contar com um formulário que organiza de maneira simples o registro individual das atividades. Como mostrado na Figura 9, esse formulário é capaz de identificar automaticamente o tipo do período de participação, de acordo com a atividade que está sendo registrada.

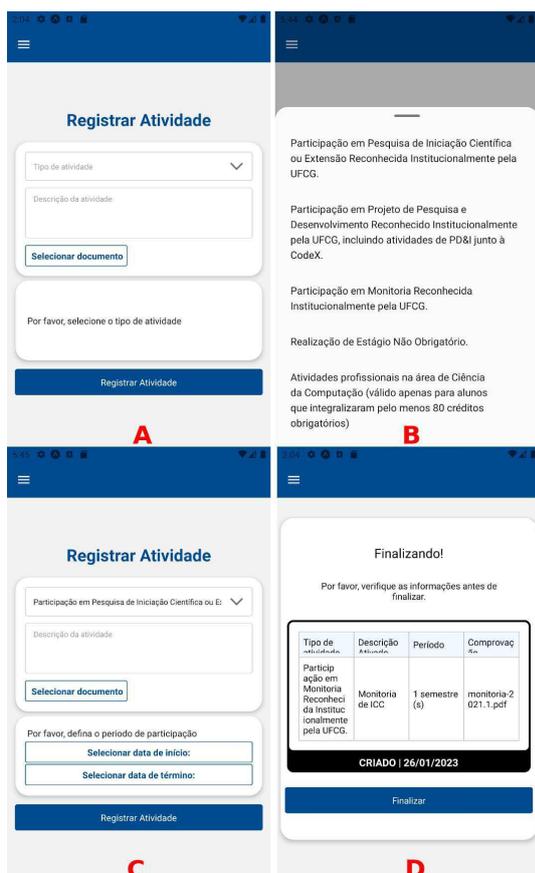


Figure 9. Telas de registro de atividades.

Por fim, o usuário terá fácil acesso às informações sobre suas submissões (Figura 10(A)), além de todos os passos necessários para declaração do processo (Figura 10(B)).

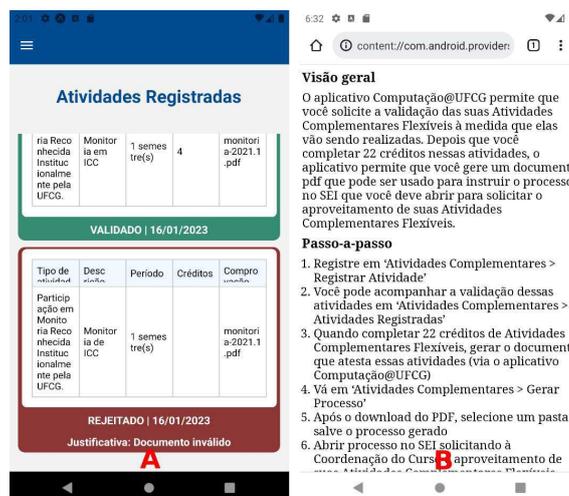


Figure 10. Telas informativas.

## 5. CONCLUSÃO E TRABALHOS FUTUROS

Com a solução elaborada e desenvolvida, podemos discutir seu comportamento. Devido ao tempo para realização deste trabalho e ao custo de implantação, não foi viável que tal ferramenta fosse testada e validada pelo público alvo. Nesta seção, serão discutidas algumas funcionalidades do sistema elaborado, bem como perspectivas futuras.

Atendo-se ao método atual de declaração de atividades utilizado pela UFCG, não há nenhuma etapa de validação prévia à geração do processo formal. Isso destaca o controle que a ferramenta pode promover. Ademais, devido ao fato do processo ser criado pelo sistema, é possível ter maior segurança e controle das informações presentes no arquivo. Além disso, também existe uma amplificação do domínio, o que proporciona uma melhor condução do procedimento de geração em si.

Além de reforçar a confiabilidade dos registros, a aplicação permite que a validação seja distribuída entre diferentes revisores e que a quantidade de créditos a serem aproveitados seja definitiva. No cenário em que a atividade é orientada por um professor de computação, é factível que este seja responsável por validar e determinar a quantidade de créditos que devem ser aproveitados.

Outro ponto a ser destacado é a simplificação e automação da comunicação disponibilizada pela ferramenta. O envio de e-mails acompanha todo o curso para validação de uma atividade. Além disso, o estudante pode receber justificativas específicas para cada uma de suas submissões.

Por fim, por se tratar de um CRM, a expectativa é o surgimento de diversas novas funcionalidades, seja para atender demandas dos estudantes e professores, bem como para o estudo e conhecimento das necessidades que

permeiam a graduação. Uma ideia que já começou a ser explorada foi a criação de um Chatbot para atender perguntas recorrentes feitas à coordenação da graduação.

## 6. REFERÊNCIAS BIBLIOGRÁFICAS

[1] G. P. A. Prieto and N. Piedra, "Exploring success factors of CRM strategies in Higher Education Institutions: a case study of CRMUTPL for conversion of prospects in university students," 2021 XVI Latin American Conference on Learning Technologies (LACLO), Arequipa, Peru, 2021, pp. 474-477, doi: 10.1109/LACLO54177.2021.00083.

[2] A. Hamid et al., "Social CRM Analytics Model on Malaysian Institutes of Higher Learning (IHLs)," 2018 3rd Technology Innovation Management and Engineering Science International Conference (TIMES-iCON), Bangkok, Thailand, 2018, pp. 1-5, doi: 10.1109/TIMES-iCON.2018.8621834.

[3] L. Lechtchinskaia, I. Friedrich and M. H. Breitner, "Requirements Analysis for a Student Relationship Management System—Results from an Empirical Study in Ivy League Universities," 2012 45th Hawaii International Conference on System Sciences, Maui, HI, USA, 2012, pp. 5132-5141, doi: 10.1109/HICSS.2012.502.

[4] N. Selimi, M. A. Trpkovska, L. A. Bexheti and M. Fetaji, "Utilization of Customer Relationship Management (CRM) theory, prototype and tools for improved strategic marketing in HE," 2018 7th Mediterranean Conference on Embedded Computing (MECO), Budva, Montenegro, 2018, pp. 1-4, doi: 10.1109/MECO.2018.8405992.

[5] BADWAN, Jehad J. et al. Adopting technology for customer relationship management in higher educational institutions. 2017.

[6] Aproveitamento de Atividades Complementares. Disponível em: <https://www.computacao.ufcg.edu.br/graduacao/procedimentos-graduacao/aproveitamento-de-atividades-complementares>. Acesso em 29 de Janeiro de 2023.

[7] Sistema Eletrônico de Informações - SEI - UFCG. Disponível em: <https://prgaf.ufcg.edu.br/sei-ufcg.html>. Acesso em 29 de Janeiro de 2023.

[8] Using OAuth 2.0 for Web Server Applications. Disponível em: <https://developers.google.com/identity/protocols/oauth2/web-server>. Acesso em 29 de Janeiro de 2023.

[9] Flask. Disponível em: <https://flask.palletsprojects.com/en/2.2.x/>. Acesso em 29 de Janeiro de 2023.

[10] smtplib - SMTP protocol client. Disponível em: <https://docs.python.org/3/library/smtplib.htm>. Acesso em 29 de Janeiro de 2023.

[11] M. A. Hassan, "Relational and NoSQL Databases: The Appropriate Database Model Choice," 2021 22nd

International Arab Conference on Information Technology (ACIT), Muscat, Oman, 2021, pp. 1-6, doi: 10.1109/ACIT53391.2021.9677042.

[12] PostgreSQL. Disponível em: <https://www.postgresql.org/>. Acesso em 29 de Janeiro de 2023.

[13] React. Disponível em: <https://reactjs.org/>. Acesso em 29 de Janeiro de 2023.

[14] React Native. Disponível em: <https://reactnative.dev/>. Acesso em 29 de Janeiro de 2023.

[15] Expo. Disponível em: <https://expo.dev/>. Acesso em 29 de Janeiro de 2023.