



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

**RENAN CARNEIRO BARBOSA DE ARAÚJO**

**EURECADASH:  
UM DASHBOARD COMPLETO PARA GESTÃO E  
ACOMPANHAMENTO DE MATRÍCULAS ACADÊMICAS**

**CAMPINA GRANDE - PB**

**2023**

**RENAN CARNEIRO BARBOSA DE ARAÚJO**

**EURECADASH:**

**UM DASHBOARD COMPLETO PARA GESTÃO E  
ACOMPANHAMENTO DE MATRÍCULAS ACADÊMICAS**

**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 : Francisco Vilar Brasileiro**

**CAMPINA GRANDE - PB**

**2023**

**RENAN CARNEIRO BARBOSA DE ARAÚJO**

**EURECADASH:  
UM DASHBOARD COMPLETO PARA GESTÃO E  
ACOMPANHAMENTO DE MATRÍCULAS ACADÊMICAS**

**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:**

**Francisco Vilar Brasileiro  
Orientador – UASC/CEEI/UFCG**

**Dalton Dario Serey Guerrero  
Examinador – UASC/CEEI/UFCG**

**Melina Mongiovi Brito Lira  
Professor da Disciplina TCC – UASC/CEEI/UFCG**

**Trabalho aprovado em: 01 de DEZEMBRO de 2023.**

**CAMPINA GRANDE - PB**

## RESUMO

Ao longo de todo o regime acadêmico é necessário que os discentes façam diversas matrículas, a fim de efetuar a inscrição em disciplinas do período acadêmico vigente. Nesse contexto, principalmente com o grande volume de matrículas realizadas, se faz necessário o melhoramento da organização, do gerenciamento, do controle e do acompanhamento das matrículas acadêmicas. Atualmente, mesmo diante de uma crescente modernização sistemática, o sistema de matrículas universitárias da Universidade Federal de Campina Grande é muito dependente do coordenador acadêmico, que deve manualmente: inscrever turmas, modificar quantidades de vagas e efetuar todo o planejamento de turmas manualmente. Além disso, o sistema não avisa aos estudantes sobre turmas ideais para suas matrículas e nem sobre o horário inicial da abertura do período de inscrição das disciplinas. Nesse viés, o Eureka Dashboard, surge como uma ferramenta facilitadora, na qual o coordenador acadêmico pode visualizar a oferta de vagas ideal para as disciplinas, modificar a disponibilização das disciplinas e acompanhar a inscrição dos discentes, tudo isso de forma unificada e centralizada, com um acesso facilitado e seguro. Com isso, o trabalho desses gerenciadores será facilitado, e através de uma entrevista, com um usuário alvo, será possível medir o real impacto da plataforma e se ela foi fundamental no gerenciamento de tempo desses profissionais.

**EURECADASH:**  
**A COMPLETE DASHBOARD FOR MANAGING AND TRACKING**  
**ACADEMIC ENROLLMENTS**

**ABSTRACT**

Throughout the academic regime, students must register several times in order to enroll in subjects during the current academic period. In this context, especially with the large volume of enrollments carried out, it is necessary to improve the organization, management, control and monitoring of academic enrollments. Currently, even in the face of increasing systematic modernization, the university enrollment system at the Federal University of Campina Grande is very dependent on the academic coordinator, who must manually: register classes, modify the number of places and carry out all class planning manually. Furthermore, the system does not notify students about ideal classes for their enrollment or about the initial opening time of the discipline registration period. In this sense, the Eureka Dashboard appears as a facilitating tool, in which the academic coordinator can view the ideal offer of vacancies for the subjects, modify the availability of the subjects and monitor the registration of students, all in a unified and centralized way, with easy and safe access. This will make the work of these managers easier, and through an interview with a target user, it will be possible to measure the real impact of the platform and whether it was fundamental in managing these professionals' time.

# EURECADASH: UM DASHBOARD COMPLETO PARA GESTÃO E ACOMPANHAMENTO DE MATRÍCULAS ACADÊMICAS

Renan Carneiro Barbosa de Araújo  
Universidade Federal de Campina Grande  
Campina Grande, Paraíba, Brasil  
renan.araujo@ccc.ufcg.edu.br

Francisco Vilar Brasileiro  
Universidade Federal de Campina Grande  
Campina Grande, Paraíba, Brasil  
fubica@computacao.ufcg.edu.br

## RESUMO

Ao longo de todo o regime acadêmico é necessário que os discentes façam diversas matrículas, a fim de efetuar a inscrição em disciplinas do período acadêmico vigente. Nesse contexto, principalmente com o grande volume de matrículas realizadas, se faz necessário o melhoramento da organização, do gerenciamento, do controle e do acompanhamento das matrículas acadêmicas. Atualmente, mesmo diante de uma crescente modernização sistemática, o sistema de matrículas universitárias da Universidade Federal de Campina Grande é muito dependente do coordenador acadêmico, que deve manualmente: inscrever turmas, modificar quantidades de vagas e efetuar todo o planejamento de turmas manualmente. Além disso, o sistema não avisa aos estudantes sobre turmas ideais para suas matrículas e nem sobre o horário inicial da abertura do período de inscrição das disciplinas. Nesse viés, o Eureka Dashboard, surge como uma ferramenta facilitadora, na qual o coordenador acadêmico pode visualizar a oferta de vagas ideal para as disciplinas, modificar a disponibilização das disciplinas e acompanhar a inscrição dos discentes, tudo isso de forma unificada e centralizada, com um acesso facilitado e seguro. Com isso, o trabalho desses gerenciadores será facilitado, e através de uma entrevista, com um usuário alvo, será possível medir o real impacto da plataforma e se ela foi fundamental no gerenciamento de tempo desses profissionais.

## PALAVRAS-CHAVE

Acadêmico, Dashboard, Desenvolvimento Web, Matrículas.

## 1. INTRODUÇÃO

Durante toda a jornada acadêmica um discente realiza diversas matrículas, isso ocorre pois ele precisa confirmar a sua inscrição nas disciplinas que irá cursar no período vigente. Desse modo, o coordenador do curso, exerce um papel fundamental no processo de matrícula que é de fornecer uma tutela, um acompanhamento e uma supervisão de todo esse processo de inscrição nas disciplinas ofertadas aos discentes, e devido ao grande número de discentes ativos e de disciplinas ofertadas o processo de matrículas se torna muitas vezes algo exaustivo, cauteloso e desgastante.

Atualmente, é necessário que coordenador acesse diversas páginas em busca de informações dos discentes e esse deve executar manualmente alguns scripts para tentar visualizar uma possível configuração ideal de oferta de disciplinas, portanto, é preciso facilitar o processo de matrícula para o coordenador do curso, melhorando o seu gerenciamento de tempo e expondo métricas precisas e necessárias para uma melhor gestão e controle acadêmico.

Nesse contexto, o Eureka DashBoard busca promover uma centralização dos dados em um único ambiente web, além disso, facilita no cálculo e ajustes das disciplinas ofertadas aos discentes, algo muito trabalhoso e sensível, visto que, poderá impactar diretamente na formação acadêmica pois uma falha resulta em um excesso ou falta de vagas em disciplinas fundamentais para a conclusão do curso. Nesse âmbito, o coordenador fará o seu login no sistema, logo após ele poderá ver métricas fundamentais do seu curso como a variação do CRA(Coeficiente de Rendimento Acumulado) ou a variação da velocidade média do seu curso, métricas baseadas nas disciplinas cursadas pelos discentes ativos no último período vigente. Ademais, ele também poderá visualizar alunos em situação de risco de evasão, facilitando a identificação desses discentes e uma possível comunicação preventiva a fim de evitar uma desistência ou evasão do curso. Por fim, o coordenador também pode visualizar uma possível oferta de disciplinas ideal para o próximo período acadêmico, tudo isso de forma automatizada, centralizada e concisa com os dados acadêmicos.

Para avaliar a integridade do sistema, e bem como validar toda a segurança, praticidade e funcionalidade do projeto, foi entrevistado um dos coordenadores de curso da UFCG [1]. No qual, ele ao utilizar o sistema, opinou e avaliou o Eureka DashBoard, o classificando positivamente e confirmando que irá auxiliá-lo no processo de matrículas e o fará economizar tempo e facilitará melhorias para a gestão do seu curso.

## 2. SOLUÇÃO

O presente trabalho tem como objetivo a criação de um dashboard no qual o coordenador de um curso da Universidade Federal de Campina Grande possa acessar estatísticas do seu curso e dos discentes ingressos, além disso, também objetiva-se facilitar a identificação das vagas necessárias para as turmas do próximo processo de matrículas.

Nesse âmbito, o EurekaDash possibilita que o coordenador tenha centralizado em um só ambiente e de acesso facilitado, dados essenciais para uma plena condução e gerenciamento do curso.

## 2.1. Funcionalidades

Ao abrir a página inicial, o usuário encontra a página de login do sistema, na qual é possibilitado que a autenticação do sistema seja realizada, desse modo, o usuário consegue acesso a todas as funcionalidades do sistema.

Para ser efetuado o login no dashboard, é preciso que o usuário já seja cadastrado previamente no sistema Eureka, sendo assim, ao acessar a tela de login, ele necessita apenas inserir seu nome de usuário e sua senha e o serviço de autenticação unificado irá verificar a integridade das informações, ademais, não é preciso que haja um cadastro de novos usuários ao sistema, pois eles já devem ter um cadastro efetivo no sistema Eureka.

Após efetuar o login na aplicação, é possibilitado a visualização completa da aplicação, inicialmente há um direcionamento para a página de desempenho, na qual consta as métricas gerais do curso, como a variação do CRA e da velocidade média dos discentes. Além disso, também há uma tabela interativa com a identificação dos alunos que apresentam risco de evasão do curso. Por fim, também há um menu no topo, que permite que o usuário alterne entre as funcionalidades do sistema.

Uma outra funcionalidade da aplicação é a parte de planejamento, na qual, é permitido a visualização da demanda de todas as disciplinas ofertadas no curso, nessa página, o coordenador pode visualizar a necessidade de vagas nas disciplinas e com isso, planejar de forma efetiva um melhor processo de matrículas.

## 3. ARQUITETURA

O sistema EurekaDash adotou uma arquitetura web baseada no modelo cliente-servidor [2]. Nesse modelo arquitetural, o cliente efetua solicitações ao servidor, e obtém uma resposta contendo os dados solicitados. A aplicação se divide em duas partes principais: o frontend, que faz o papel de cliente da arquitetura, fornecendo uma interface para o usuário; e o backend, no qual contém toda a lógica da aplicação, além de armazenar e manipular os dados, atuando como servidor no modelo arquitetural.

A comunicação entre os backends e o frontend segue o padrão REST(Representational State Transfer) [3], esse modelo arquitetural é usualmente utilizado na construção de aplicações web. Ele norteia princípios e restrições que resultam na criação de APIs robustas e escaláveis, no padrão REST, os recursos são fornecidos no formato de endpoints e as interações necessárias ocorrem através do protocolo HTTP [4]. Tal abordagem, é algo padronizado e independe de linguagens e frameworks utilizados e é cada vez mais utilizada pelas aplicações modernas devido a sua simplicidade e praticidade para aplicações e serviços web.

### 3.1 Tecnologias de Backend

Na construção do backend da aplicação, utilizou-se o framework Spring Boot [7], pois tal plataforma de desenvolvimento permite um desenvolvimento rápido, facilitado e facilmente escalável. Nota-se que, a prévia construção do projeto Eureka nesse mesmo framework influenciou na escolha dessa plataforma de desenvolvimento para o Eureka Dash, já que possibilita uma melhor

manutenibilidade e mais fácil legibilidade, manter um mesmo framework e linguagem em todo o sistema.

### 3.2 Estrutura do Backend

Para a estruturar o backend da aplicação, foi escolhido o padrão arquitetural MVC(model-view-controller) [5], pois ele possui uma enorme facilidade na separação das camadas do sistema, separando as responsabilidades das partes envolvidas em cada uma dessas camadas, produzindo assim uma API(Application Programming Interface) com código mais organizado, facilitando desse modo a escalabilidade e manutenibilidade da aplicação.

### 3.3 Funcionalidades do Backend

O backend da aplicação possui alguns endpoints que permitem o pleno funcionamento da ferramenta, dentre eles, podemos destacar o endpoint que calcula a variação média do CRA e da velocidade média dos estudantes ativos do curso do usuário autenticado. Outro endpoint destacável é o que seleciona os discentes que apresentam algum risco de evasão ao curso, possibilitando a tomada de medidas cautelares para com esses discentes. Por fim, tem-se o endpoint que calcula a necessidade de vagas em cada disciplina do curso, baseada na matrícula ideal de cada discente ativo, retornando portanto, uma métrica fundamental para facilitar o processo de matrícula dos estudantes.

### 3.4 Tecnologias do Frontend

Para a construção do frontend da aplicação, foi utilizado o framework Angular [6], com a linguagem TypeScript, a escolha desse framework juntamente com essa linguagem se deu pela popularidade do uso dessas tecnologias, que possuem um grande mercado tecnológico e pela robustez do sistema produzido, já que o framework Angular, por ser desenvolvido e mantido pela Google, possui uma solidez imensa, com vasta documentação e exemplificações. Desse modo, a construção da interface com o usuário através dessa plataforma e linguagem de desenvolvimento, se torna moderna, intuitiva e de fácil controle, acesso e manutenção.

### 3.5 Estruturação do Frontend

Apesar do framework utilizado, o Angular, não exigir uma estruturação pré definida, como boa prática foi seguida uma hierarquização na sua estruturação, sendo assim, os integrantes da interface foram seccionados, a fim de melhor gerenciar os arquivos e componentes do sistema, por critério de funcionalidade, facilitando o entendimento do código e melhorando a legibilidade, com isso, contribuindo para possíveis melhorias futuras, como adições e refatorações.

### 3.6 Funcionalidades do Frontend

O frontend da aplicação, tem 3 páginas principais, a primeira que é a página de login, que é a responsável pela interação de login com o usuário, é nela que ele se autentica e entra no EurekaDash, buscou-se fazer uma tela simples e eficaz a fim de simplificar o processo de login do usuário no sistema. A posteriori, tem-se a tela de desempenho, sendo essa mais complexa, na qual encontra-se um menu no topo, que pode alterar entre as páginas do sistema, além disso, nela encontra-se dois cards com as métricas dos discentes e uma tabela com os alunos que apresentam risco de evasão, na tabela, consta os dados dos alunos para facilitar a identificação e comunicação do coordenador para com o estudante. Por fim, apresenta-se a tela de planejamento, na qual é demonstrado a necessidade e demanda de vagas de todas as disciplinas do curso do qual o usuário é responsável,

facilitando a identificação e alocação das vagas para o período de matrículas seguinte.

## 4. PROCESSO DE DESENVOLVIMENTO

### 4.1 Investigação do projeto Eureka

Para iniciação do projeto, foi feita uma investigação e identificação do projeto Eureka, identificando os endpoints e a base de dados presente, a fim de proporcionar um entendimento de como funciona o sistema para tornar possível o desenvolvimento de um dashboard. Para tal prática investigativa, o swagger, ferramenta de auxílio para testagem e demonstração de APIs foi fundamental, já que foi através dessa ferramenta que o código pré existente do sistema Eureka foi testado e demonstrado, permitindo o entendimento da realização das suas operações e de como os dados eram fornecidos e enviados.

### 4.2 Levantamento dos requisitos funcionais

Após todo o estudo realizado na etapa 1, foi definido e posteriormente documentado os requisitos e comportamentos que o dashboard teria que atender. Desse modo, buscando nortear o desenvolvimento do projeto, comunicar as necessidades e objetivos do dashboard, gerenciar, atribuir e priorizar o que deverá ser construído e demonstrado, e por fim, validar e verificar o que foi apresentado, foi levantado uma série de requisitos funcionais, possibilitando toda a esquematização e gerenciamento do escopo do projeto EurekaDash. A seguir segue uma tabela com os requisitos do sistema:

| Requisito | Descrição  |
|-----------|--|
| R01       | Permitir a autenticação no sistema, utilizando o servidor de autenticação do projeto Eureka.                 |
| R02       | Demonstrar a variação do CRA dos discentes ativos.   |
| R03       | Demonstrar a variação da velocidade dos discentes ativos.  |
| R04       | Identificar os alunos com risco de evasão do curso.  |
| R05       | Calcular a demanda de vagas de todas as disciplinas do curso, baseadas nas matrículas dos estudantes ativos. |

[Tabela 1: Requisitos funcionais do EurekaDash]

### 4.3 Prototipação de telas

A partir do levantamento dos requisitos funcionais, se tornou necessário a prototipação das telas do sistema, que consiste basicamente, em uma modelagem visual de como ficaria um esboço de cada tela da aplicação, a prototipação, visa acelerar o processo de criação da interface com o usuário e nortear o desenvolvimento do sistema baseado em componentes visuais conhecidos. Para efetuar este passo, foi utilizado o Figma, que

consiste em uma ferramenta de design amplamente utilizada pelo mundo cooperativo e universitário, para desenvolvimento de telas e elementos criativos, com esse editor gráfico foi possível criar a prototipação das telas do sistema de uma forma agradável e de fácil compreensão e visualização prévia. A seguir, é possível ver como ficou a prototipação de uma das telas do sistema:



[Figura: Exemplo de prototipação da tela de desempenho]

### 4.4 Implementação do backend

Com base nos requisitos funcionais do sistema, foi iniciado o processo de desenvolvimento do backend do EurekaDash, para organizar esse desenvolvimento foi utilizado o modelo de desenvolvimento ágil Scrum, adaptado para desenvolvimento solo. Dessa maneira, foi organizado um backlog com base nos requisitos do sistemas e montado um cronograma de desenvolvimento do projeto, para isso a ferramenta Trello, foi utilizada como forma de facilitar a organização e cumprimento do cronograma do desenvolvimento da aplicação.

Em seguida, foi criado um novo projeto com a linguagem Java, e o framework Spring Boot, esse seria o backend da aplicação, após a criação do projeto, foi feita diversas chamadas ao backend do projeto Eureka, a fim de receber os dados necessários para cálculos e amostragens dos itens do dashboard, como por exemplo, verificar os alunos ativos do curso, ver dados e histórico dos discentes ativos e entre outros acessos aos endpoints do projeto Eureka, dessa maneira estabeleceu a conexão entre os backends. Por fim, foi desenvolvido os endpoints do backend do EurekaDash, que permitiu que o frontend da aplicação acessasse os dados resultados da manipulação das informações recebidas do projeto Eureka.

### 4.5 Implementação do frontend

Baseado na prototipação de tela e também baseado nos requisitos funcionais, foi desenvolvido o frontend da aplicação, que também contou com a utilização do desenvolvimento ágil, seguindo a mesma organização do backend, com o cronograma estabelecido, também foi iniciado um novo projeto em Angular, com a utilização da linguagem TypeScript. Ademais, também foi utilizado a linguagem de marcação HTML5 para colocação de elementos visuais e o SCSS, que é uma linguagem de folhas, que permite alterações



visuais na interface com o usuário. Também, foi utilizado bibliotecas de componentes, a fim de agilizar e facilitar a plotagem de componentes, nesta etapa o Angular Material e o Bootstrap foram essenciais para agilizar e simplificar o desenvolvimento. Por fim, foi feita a conexão do cliente, interface com o usuário, e servidor, fornecedor dos dados, utilizando chamadas com o protocolo HTTP, gerando assim as informações necessárias para visualização do usuário.

## 5. AVALIAÇÃO

Neste segmento, segue as informações sobre a avaliação do EurekaDash, bem como os resultados obtidos.

### 5.1 Metodologia da avaliação

Objetivando comprovar a eficiência do sistema desenvolvido, foi entrevistado um coordenador de um dos cursos da UFCG(Universidade Federal de Campina Grande) a fim de validar e testar a aplicação. Após, o usuário alvo usar o dashboard foram feitas alguns questionamentos acerca do sistema e da sua usabilidade, vale ressaltar que a opinião e resposta do entrevistado é de suma importância para o projeto, já que esse é quem será o maior utilizador de toda a aplicação e é quem irá vivenciar o seu uso no dia a dia.

Desse modo, ao ser questionado como estava a autenticação e navegabilidade da aplicação web, o entrevistado respondeu que se dava muito satisfeito com a navegação e entrada do sistema, sendo essas muito rápidas e de fácil compreensão. Logo após foi questionado se as métricas informadas eram condizentes e a resposta era que eram informações sim bem realistas e que poderiam auxiliá-lo na direção e gerenciamento do curso. A posteriori, foi conversado acerca da utilização do EurekaDash, se ele se fazia útil e se o coordenador iria utilizá-lo no seu dia a dia, a resposta foi de que com algumas melhorias, correções e ajustes ele usaria sim no dia a dia e acrescentou que finalizado, será uma ferramenta que auxiliará e muito os coordenadores nas suas tarefas, facilitando muito, principalmente o processo de matrícula, algo que demanda muito tempo e dedicação da sua parte.

### 5.2 Resultados

Ao final da entrevista, o entrevistado respondeu que o sistema se encontrava em um estágio satisfatório, e que era plenamente usável para realização das tarefas propostas, cumprindo fielmente o seu papel e que com pequenas ressalvas, relacionadas a pequenos ajustes e acréscimos necessários, o sistema seria de suma importância no auxílio do coordenador na realização das suas atividades a frente do curso acadêmico.

### 5.3 Conclusão

Com base na entrevista cedida, percebe-se uma avaliação positiva do EurekaDash, é notório que o dashboard é capaz de auxiliar e facilitar o papel dos coordenadores no gerenciamento dos cursos e fornecer métricas importantes sobre o andamento acadêmico da sua unidade e dos discentes ativos. Ademais, a opinião do entrevistado forneceu insights e opiniões importantes, além de pontos claros de melhorias e complementos para a aplicação, que busca se tornar mais completa e atuar de forma mais incisiva para facilitar e evitar a sobrecarga dos coordenadores acadêmicos.

## 6. DISCUSSÃO E TRABALHOS FUTUROS

Mesmo que o EurekaDash tenha agradado o entrevistado, e sendo classificado como útil para auxiliar na administração do curso, é visto que, melhorias e acréscimos podem ser realizados em trabalhos futuros, a fim de garantir uma maior usabilidade do sistema e maior popularização da aplicação web. Desse modo, será possível que o sistema atue de maneira mais incisiva e colaborativa na gestão de matrículas e atividades curriculares, além de uma forma mais informativa tangente aos dados dos discentes ativos dos cursos oferecidos pela UFCG e coordenados pelos responsáveis acadêmicos.

Nesse contexto, um possível trabalho futuro seria um acompanhamento do processo de matrículas, fornecendo os dados em tempo real, dos estudantes que estão efetuando o cadastro de disciplinas do curso, mostrando métricas como percentuais de matrículas realizadas, de trancamentos e de ausências, e posteriormente, alertando o coordenador alunos que não estão se matriculando, que estão muito atrasados de acordo com o cronograma do curso e que estão próximo da conclusão.

Outro destaque, como trabalho futuro, seria a visualização de discentes que estão com uma matrícula muito discrepante da que ele deveria realizar e o envio de alertas e mensagens aos alunos que não estão efetuando o cadastro nas disciplinas durante o seu período de inscrição.

Com esses trabalhos, o EurekaDash se tornaria um sistema mais completo e notoriamente mais essencial para coordenadores acadêmicos, que usariam esse dashboard como principal ferramenta de gestão e acompanhamento de matrículas e atividades curriculares do seu curso.

## 7. AGRADECIMENTOS

Em primeiro lugar, gostaria de agradecer a Deus, sem ele não seria possível a minha caminhada até aqui, porque dele, por ele e para ele são todas as coisas. Gostaria também de agradecer aos meus pais e a minha irmã, que foram de grande apoio e verdadeiros incentivadores durante toda a minha jornada acadêmica. Também expressei meus profundos agradecimentos ao professor Francisco Vilar Brasileiro, que durante todo meu processo de desenvolvimento do trabalho, me orientou com maestria, bondade, compaixão e empatia, sem sua fundamental colaboração não seria possível encerrar essa jornada acadêmica. Por fim, estendo minha gratidão, a todos os amigos e colegas, que de maneira incisiva e companheira me apoiaram e incentivaram durante todo o período acadêmico e ao curso de Ciência da Computação da UFCG, por me proporcionar tamanhos conhecimentos, aprendizados e crescimento pessoal e intelectual.

## 8. REFERENCES

- [1] Universidade Federal de Campina Grande (UFCG) official website. Disponível em: <https://portal.ufcg.edu.br/>.
- [2] Client-Server Architecture overview. Disponível em: <https://www.geeksforgeeks.org/client-server-model/>.
- [3] RESTful API (RESTapi) documentation. Disponível em: <https://www.ibm.com/topics/rest-apis>
- [4] Hypertext Transfer Protocol (HTTP) specification. Disponível em:

<https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview>.

- [5] Model-View-Controller (MVC) design pattern. Disponível em: <https://www.geeksforgeeks.org/mvc-framework-introduction/>.
- [6] Angular framework official documentation. Disponível em: <https://angular.io/docs>.
- [7] Spring Boot framework official documentation. Disponível em: <https://docs.spring.io/spring-boot/docs/current/reference/html/index.html>.

