



CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

RAUL SIMIONI

GERAÇÃO DE OFERTAS (TURMA)

Londrina
2023

SUMÁRIO

1	INTRODUÇÃO	2
1.1	PROPOSTA E OBJETIVOS	2
1.2	JUSTIFICATIVA	3
2	DIAGRAMA DE CASO DE USO	4
2.1	Telas	5
3	WORKFLOW (AS IS) NA NOTAÇÃO BPMN	6
4	RECURSOS E AMBIENTE DE DESENVOLVIMENTO	7
4.1	Linguagem de Programação	7
4.2	Banco de dados	7
4.3	Padronização de API	7
4.4	Versionamento	7
5	CRONOGRAMA	8
	REFERÊNCIAS	9

1 INTRODUÇÃO

A proposta desse microssistema consiste em validar Unidades Curriculares oferecidas aos alunos, disponibilizar uma oferta ideal e verificar a carga horária cumprida. Trata-se de um microssistema do sistema de gerenciamento da matriz curricular denominado CUBO, que será utilizado pelo Centro Universitário Filadélfia (UNIFIL). O microssistema está sendo desenvolvido pelos estagiários do Núcleo de Práticas em Informática (NPI), e pretende-se que seja implementado no final do ano de 2023.

Esse microssistema está dividido entre duas pessoas. Ryan Lucas será responsável por desenvolver as funcionalidades de verificação das unidades curriculares cumpridas, verificação do histórico do aluno e verificação das cargas horárias obrigatórias (core) e não obrigatórias (flex). Por sua vez, as funcionalidades de geração de ofertas (turmas) e validação de ofertas serão desenvolvidas por Raul Simioni.

A utilização desse sistema será de grande ajuda, pois com o desenvolvimento será possível uma forma mais eficiente e automática de realizar todo o processo de validação de unidades curriculares cumpridas e gerar a oferta ideal para o aluno, sendo assim, esse microssistema será capaz de suprir uma grande necessidade dos cursos de computação da UNIFIL, pois ainda não há nenhum tipo de software que realiza todo esse processo de forma eficaz, ou seja, está sendo feito manualmente pela coordenação.

1.1 PROPOSTA E OBJETIVOS

O objetivo do projeto é desenvolver um sistema eficiente de validação de Unidades Curriculares ofertadas e oferecer a melhor oferta aos alunos de forma automática. Esse sistema terá como finalidade principal facilitar o processo de verificação de carga horária cumpridas, sendo elas obrigatórias (Core) ou não obrigatória (Flex), oferta automática de Unidades Curricular ideal aos alunos (Automatizando o gerenciamento de ofertas para trancamentos).

Com o esse projeto, os gestores da instituição de ensino terão uma ferramenta para validar a oferta de Unidades Curriculares aos alunos de forma mais eficiente, otimizando esse processo e oferecendo a Unidade Curricular ideal. Isso permitirá que a instituição de ensino tenha um controle mais preciso sobre a carga horária de seus alunos, garantindo que eles cumpram as exigências do currículo escolar.

1.2 JUSTIFICATIVA

A gestão eficiente da matriz curricular é essencial para a oferta de um ensino de qualidade, que atenda às necessidades dos alunos e do mercado de trabalho. Uma matriz curricular mal planejada pode resultar em atrasos na formação dos alunos, falhas no processo de ensino-aprendizagem e baixa empregabilidade dos egressos.

Além disso, a gestão da matriz curricular é uma tarefa complexa e que demanda muita atenção e cuidado por parte dos gestores da instituição de ensino. É necessário levar em conta diversos fatores, como a carga horária das disciplinas, os pré-requisitos para cada uma delas, a disponibilidade de professores e salas de aula, entre outros. Sem um sistema adequado de gerenciamento, essa tarefa pode se tornar extremamente trabalhosa e propensa a erros. Sendo assim, o desenvolvimento desse microsistema será de grande importância, pois será implementada em um sistema maior, ou seja, vai ser necessário que seja verificado Unidades Curriculares cumpridas e carga horária cumprida, e enviando os resultados para outros microsistemas que necessitam dessas informações.

2 DIAGRAMA DE CASO DE USO

O diagrama de caso de uso desenvolvido na figura 1 possui duas funcionalidades principais, sendo elas verificar oferta e verificar unidade curricular cumprida. Na funcionalidade de verificar oferta, o usuário poderá ver todas as ofertas disponíveis ao aluno, sendo essas ofertas geradas automaticamente com base em unidades curriculares já cumpridas, unidades curriculares que possui dependência das que foram feitas e também unidades curriculares obrigatórias (core) que possui prioridade. Na funcionalidade de verificar UC cumprida o usuário poderá ver unidades curriculares que já foram realizadas. Sendo as outras funcionalidades como verificar histórico do aluno e gerar oferta, processos necessários para as funcionalidades principais citadas anteriormente.

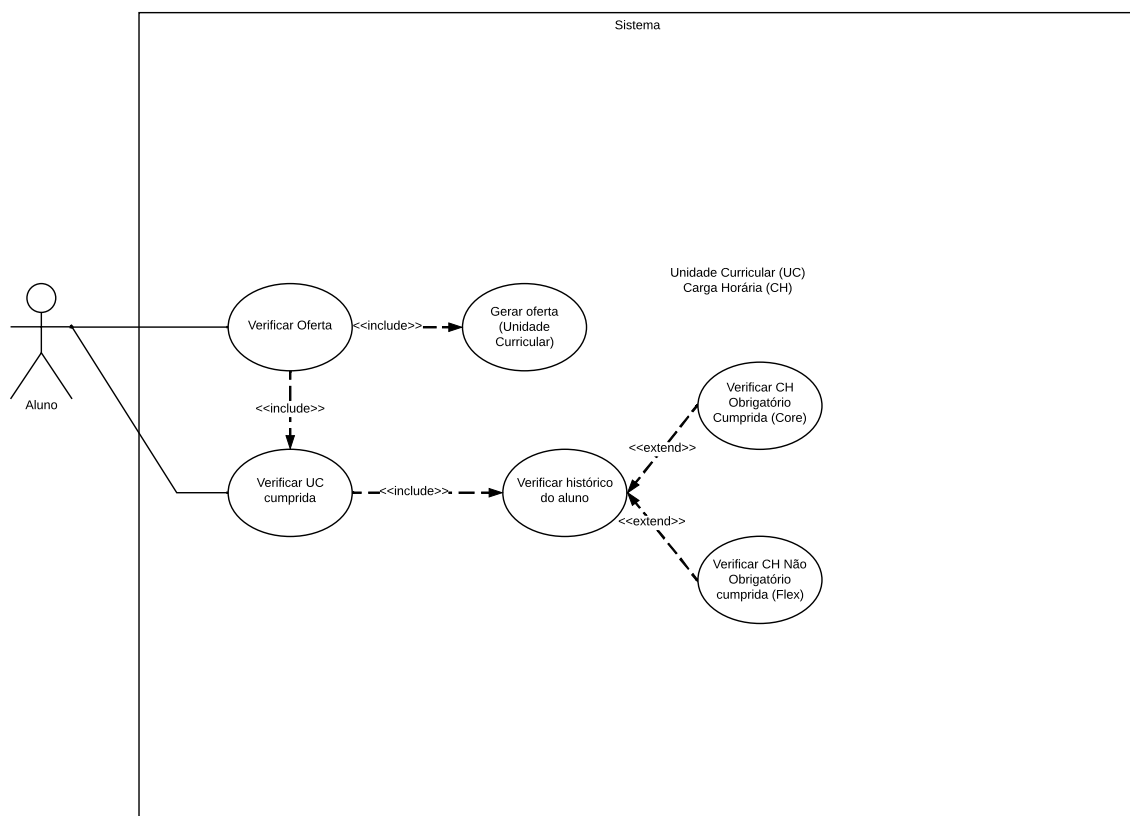


Figura 1 – Diagrama de Caso de Uso

As funcionalidades de verificar oferta, gerar oferta (unidade curricular) e verificar Unidades Curriculares cumpridas serão desenvolvidas pelo Ryan Lucas, consequentemente, as funcionalidades de verificar histórico do aluno, verificar carga horária obrigatória (Core) e não obrigatória (Flex) serão desenvolvidas pelo Raul Simioni.

2.1 TELAS

Na figura 2 será possível ver ofertas para o aluno, essas ofertas são geradas automaticamente com base em unidades curriculares já cumpridas, unidades curriculares que possui dependencia das que foram feitas e também unidades curriculares obrigatórias (core) que possui prioridade.

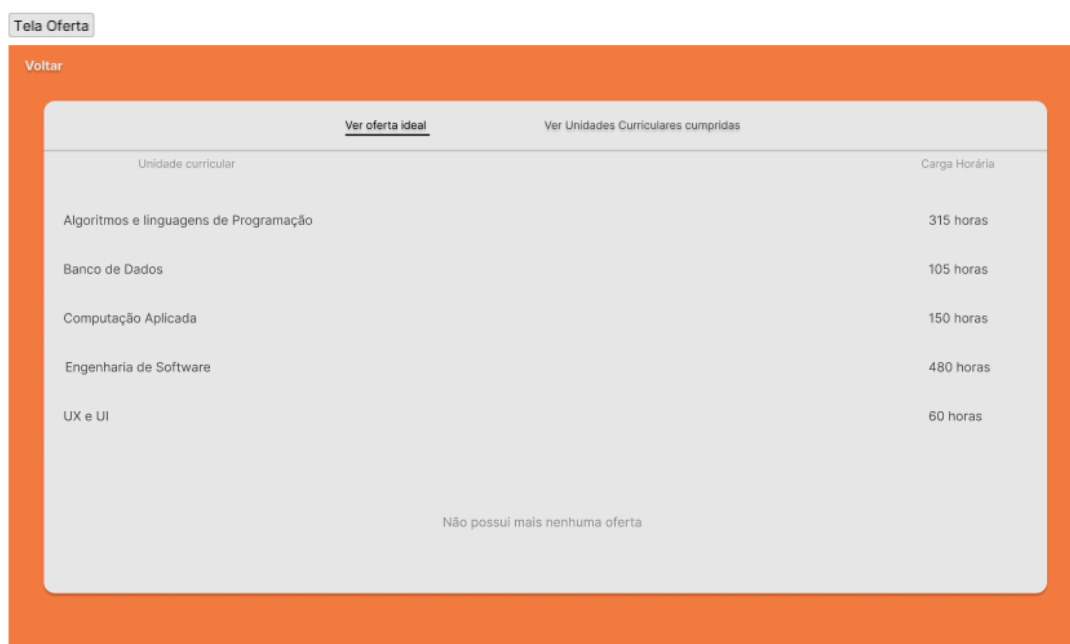


Figura 2 – Tela para ver oferta

Na figura 2 será possível ver unidades curriculares que já foram realizadas.

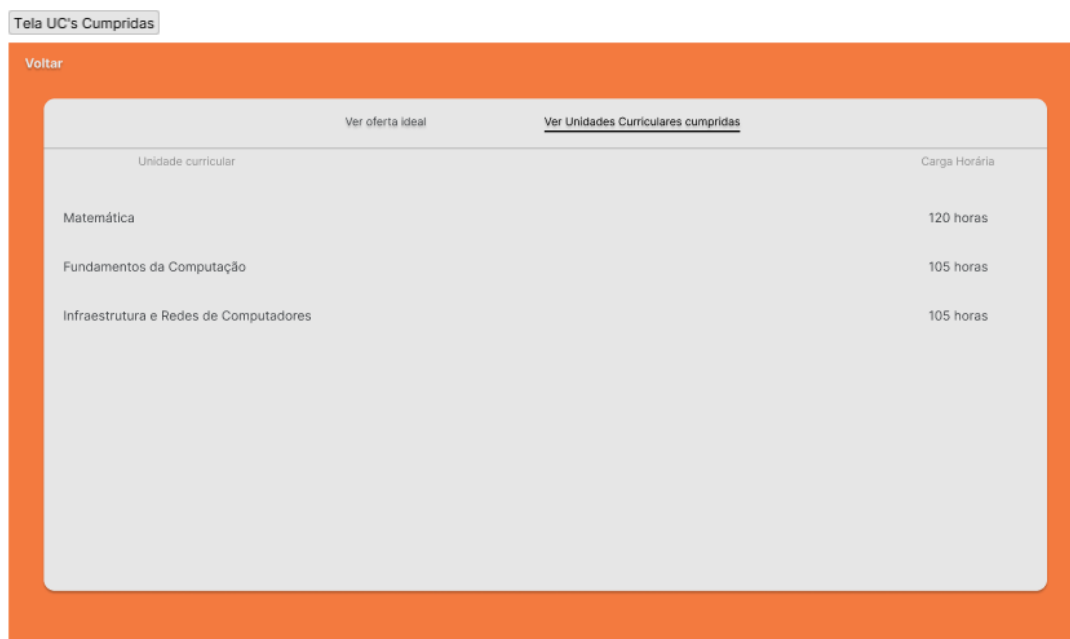


Figura 3 – Tela para ver unidades curriculares cumpridas

3 WORKFLOW (AS IS) NA NOTAÇÃO BPMN

Na figura 4 está sendo apresentado a maneira como é feito o processo de verificação e oferta de unidades curriculares atualmente, ou seja, sendo feito de maneira manual e com pouca eficiência.

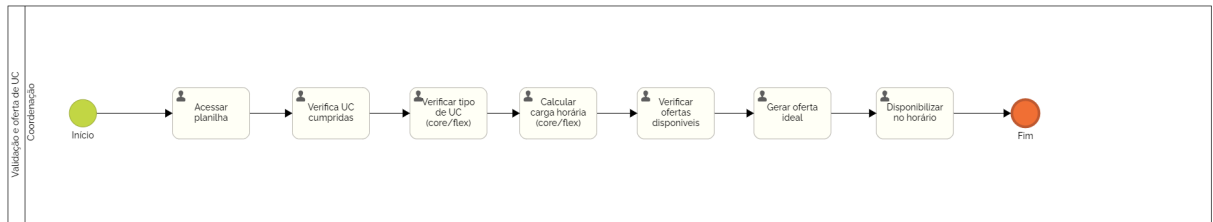


Figura 4 – Workflow BPMN (AS IS)

4 RECURSOS E AMBIENTE DE DESENOVOLVIMENTO

4.1 LINGUAGEM DE PROGRAMAÇÃO

O front-end e a API, serão desenvolvidas utilizando Laravel, um framework em PHP, que será responsável pela construção da interface com o usuário, ou seja, a aparência do aplicativo, a interação com os usuários e a comunicação com o servidor. Além disso, será usado para a criação do API, para que haja comunicação com outros sistemas sendo desenvolvidos.

O back-end será desenvolvido com MYSQL, um sistema de gerenciamento de banco de dados relacional, sendo responsável pelo gerenciamento e armazenamento de dados do sistema.

4.2 BANCO DE DADOS

Em relação ao banco de dados, será utilizado o MYSQL, pois além de ser muito utilizado, também facilitará o processo e integração com relação as demais tecnologias.

4.3 PADRONIZAÇÃO DE API

Será utilizado o postman, com a utilização do permitirá a visualização da API e documentação dos outros integrantes do projeto, ou seja, estagiários que estão desenvolvendo as outras funcionalidades, tendo assim, a comunicação entre os sistemas.

4.4 VERSIONAMENTO

O Github será utilizado para hospedagem do código desenvolvido do sistema.

5 CRONOGRAMA

Na Figura 5 está todo o processo de desenvolvimento do primeiro semestre do projeto.

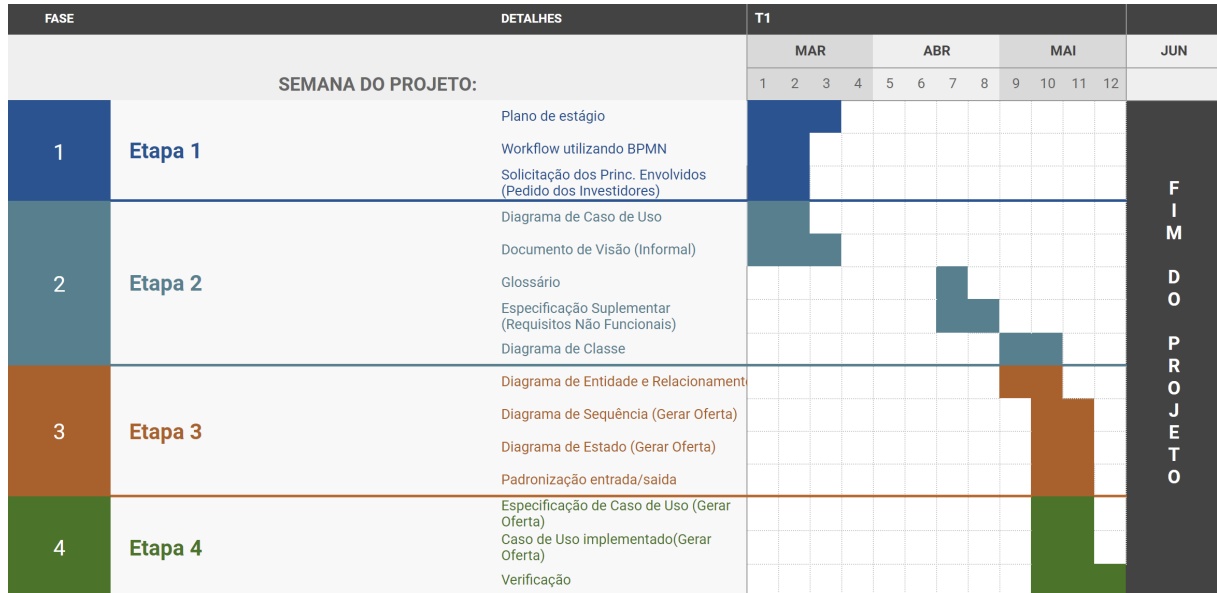


Figura 5 – Cronograma de desenvolvimento do projeto

REFERÊNCIAS

FIGMA. *Figma: the collaborative interface design tool*. 2022. Disponível em: <<https://www.figma.com/?fuid=>>>.

HEFLO PT. *HEFLO BPM | Gerenciamento de Processos de Negócio*. 2023. Disponível em: <<https://www.heflo.com/pt-br/>>>.

LUCIDCHART. *Diagramas UML Online*. Disponível em: <<https://www.lucidchart.com/pages/pt/exemplos/uml-online>>>.