

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO  
INSTITUTO DE MATEMÁTICA  
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

IGOR CARVALHO DE PAIVA FONSECA  
KAIQUE RODRIGUES MENEZES  
MATHEUS PINHEIRO PINTO

SISTEMA PARA AUXÍLIO NO PLANEJAMENTO  
ACADÊMICO DE ALUNOS ATRAVÉS DE  
VISUALIZAÇÕES E RECOMENDAÇÕES

RIO DE JANEIRO

2018

IGOR CARVALHO DE PAIVA FONSECA  
KAIQUE RODRIGUES MENEZES  
MATHEUS PINHEIRO PINTO

SISTEMA PARA AUXÍLIO NO PLANEJAMENTO  
ACADÊMICO DE ALUNOS ATRAVÉS DE  
VISUALIZAÇÕES E RECOMENDAÇÕES

Trabalho de conclusão de curso de graduação apresentado ao Departamento de Ciência da Computação da Universidade Federal do Rio de Janeiro como parte dos requisitos para obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Prof. Giseli Rabello Lopes, D.Sc.

RIO DE JANEIRO

2018

F676

Fonseca, Igor Carvalho de Paiva

Sistema para auxílio no planejamento acadêmico de aluno através de visualizações e recomendações / Igor Carvalho de Paiva Fonseca, Kaique Rodrigues Menezes, Matheus Pinheiro Pinto. – Rio de Janeiro, 2018.

92 f.

Orientadora: Giseli Rabello Lopes.

Trabalho de Conclusão de Curso (graduação) - Universidade Federal do Rio de Janeiro, Instituto de Matemática, Bacharelado em Ciência da Computação, 2018.

1. Sistemas de recomendação. 2. Recuperação da informação. 3. Recomendação de disciplinas. I. Menezes, Kaique Rodrigues. II. Pinto, Matheus Pinheiro. III. Lopes, Giseli Rabello (Orient.). IV. Universidade Federal do Rio de Janeiro, Instituto de Matemática. V. Título.

IGOR CARVALHO DE PAIVA FONSECA  
KAIQUE RODRIGUES MENEZES  
MATHEUS PINHEIRO PINTO

SISTEMA PARA AUXÍLIO NO PLANEJAMENTO  
ACADÊMICO DE ALUNOS ATRAVÉS DE  
VISUALIZAÇÕES E RECOMENDAÇÕES

Trabalho de conclusão de curso de graduação apresentado ao Departamento de Ciência da Computação da Universidade Federal do Rio de Janeiro como parte dos requisitos para obtenção do grau de Bacharel em Ciência da Computação.

Aprovado em \_\_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_ .

BANCA EXAMINADORA:

---

Prof. Giseli Rabello Lopes, D.Sc.

---

Prof. João Carlos Pereira da Silva, D.Sc.

---

Prof. Maria Luiza Machado Campos, D.Sc.

## **AGRADECIMENTOS**

Gostaríamos de agradecer a equipe do SIGA e a coordenação do curso pela colaboração com a disponibilização dos dados para a realização deste trabalho. A todos os alunos que se voluntariaram a responder os questionários nas etapas de validação e testes de recomendação e usabilidade. Também gostaríamos de agradecer aos professores Anamaria Martins, Carla Amor Divino, João Carlos da Silva, João Paixão, Juliana Vianna, Maria Luiza Campos e S. Collier por sua colaboração nas entrevistas conduzidas.

Um agradecimento especial à professora e orientadora Giseli Lopes pelos conhecimentos passados na orientação deste projeto.

Igor Carvalho de Paiva Fonseca

Gostaria de agradecer a minha família que sempre me apoiou nessa jornada acadêmica, aos meus amigos que me ajudaram ao longo do curso em nossas inúmeras horas de estudo em grupo e a minha noiva Rayssa que estava sempre ali para não me deixar desistir do meu objetivo.

Agradeço também aos professores Vinícius Gusmão, Silvana, Collier e Giseli por terem sido excelentes professores, e por me darem mais motivos para querer continuar nesta profissão.

Obrigado também ao pessoal do CAPGov (Xiao, Débora e Edberg) por todo conhecimento e apoio transmitido para mim nos meus primeiros anos de faculdade.

Todos vocês foram essenciais em algum momento ao longo desses anos, obrigado por tudo!

Kaique Rodrigues Menezes

Gostaria de agradecer primeiramente a minha família: minha mãe Márcia e avó Zilda, por sempre me darem a base e apoio necessários desde que me entendo por gente; ao meu padrasto Ademilson, que sempre estava lá para ajudar com os favores mais inesperados e improváveis; também a minha sogra Luiza Helena e cunhada Isadora, que me acolheram como família em um momento muito importante em minha vida; e, por fim, a minha esposa Carla Beatriz, que esteve ao meu lado ao longo de toda essa jornada de pertinho, celebrando os bons momentos e trabalhando comigo nos difíceis.

Agradeço também aos meus amigos de curso, desde os novos até aqueles que vem comigo de outros momentos: Guilherme Herzog, Igor Fonseca, João Carvalho, Lucas Carneiro e Matheus Pinheiro, que sempre dividiam comigo a preocupação e tensão nos momentos de estudo e também partilhavam motivação para continuar em frente.

Agradeço também a UFRJ como um todo pela oportunidade e pelo aprendizado, em especial gostaria de agradecer a equipe do SIGA pela colaboração, a coordenadora Maria Luiza pela grande ajuda, a professora Giseli Lopes pelo trabalho impecável na orientação deste projeto. Finalmente, gostaria de agradecer aos excelentes professores S. Collier, Vinícius Gusmão, Silvana Rossetto, João Carlos e Giseli Lopes que foram essenciais para minha formação pela qualidade do ensino transmitido e seguem comigo como forma de inspiração na minha vida profissional.

Sem vocês seria impossível chegar até aqui. Essa vitória é nossa. Muito obrigado por tudo!

Matheus Pinheiro Pinto

Quero agradecer, em primeiro lugar, aos meus queridos pais (Milson e Ana Cristina), pela força e incentivo durante toda essa longa caminhada.

Agradeço à minha prima Camila, que me acolheu durante quase toda a graduação e esteve sempre disposta a me ajudar caso eu precisasse. Também à minha noiva Mayra que sempre me apoiava e me incentivava durante os períodos mais difíceis que passei.

Por fim, agradeço aos meus professores da graduação S. Collier, João Carlos, Silvana e Giseli, por ter tido a oportunidade de ser seu aluno e podem ter certeza que aprendi bastante. Também não poderia esquecer do meu querido amigo e professor Renato Mauro, que desde o técnico de informática no Cefet esteve do meu lado me ensinando, incentivando e acreditando no meu sucesso.

Todos vocês foram importantes de alguma forma para eu chegar até aqui e querer continuar nesta área. Muito obrigado!

## RESUMO

Este trabalho tem como objetivo o desenvolvimento de um sistema para auxílio no planejamento acadêmico através da visualização dos dados de progresso no curso e recomendação de disciplinas para os alunos do Bacharelado em Ciência da Computação da Universidade Federal do Rio de Janeiro. Inicialmente, foram realizadas pesquisas com professores orientadores acadêmicos e alunos, com o intuito de encontrar funcionalidades importantes para serem implementadas no sistema e entender quais características devem ser consideradas no momento da escolha da grade de disciplinas a serem cursadas. Em seguida, foram obtidos os dados dos alunos junto ao SIGA (Sistema Integrado de Gestão Acadêmico), para serem utilizados no sistema, com o consentimento da coordenação do curso. Estudando os dados e analisando o resultado das pesquisas, foi proposto um sistema de recomendação de disciplinas, aplicando os conceitos aprendidos durante a disciplina de Recuperação de Informação e com funcionalidades úteis para ambos, alunos e professores orientadores. No sistema, é possível acompanhar o progresso dos alunos, ver ranqueamento de disciplinas e receber recomendação de grades ao início do período. Por fim, o sistema foi apresentado e testado com alguns alunos e professores com a finalidade de obter avaliações e insumos para melhorias no futuro.

***Palavras-chave:*** Sistemas de Recomendação. Recuperação da Informação. Recomendação de Disciplinas.

## ABSTRACT

*This work has as objective the development of a system to assist the academic planning of the students of the Bachelor of Computer Science at the Federal University of Rio de Janeiro. The system provides data visualization of the course progress and subjects recommendation. Initially, surveys with some academic advisors and students were performed in order to find important functionalities to be implemented by the system. The surveys also helped to identify which features they considered at the moment of choosing the grid of subjects. After that, student data to be used in the system was obtained from SIGA (Integrated System of Academic Management), with the consent of the coordination of the course. Analysing the data and the surveys results, the system was proposed with useful functionalities for both, students and academic advisors. The functionalities include monitoring the progress of the student and recommendations of (grids of) subjects at the start of the semester. In the system development, were applied concepts learned during the Information Retrieval subject. Lastly, the system was tested by some students and professors, aiming at obtaining evaluations and inputs for future improvements.*

**Keywords:** Recommender Systems. Information Retrieval. Subject Recommendation.

## LISTA DE FIGURAS

Figura 1:	Período dos alunos entrevistados. . . . .	31
Figura 2:	Quantidade de alunos que fazem alguma atividade complementar. . .	31
Figura 3:	Período em que os alunos iniciam alguma atividade complementar. . .	32
Figura 4:	Número de disciplinas sem cursar atividade complementar. . . . .	32
Figura 5:	Número de disciplinas cursando atividade complementar. . . . .	33
Figura 6:	Critérios mais relevantes para escolha de eletivas. . . . .	33
Figura 7:	Diversificação de disciplinas eletivas. . . . .	34
Figura 8:	Critérios mais relevantes para escolha de obrigatórias. . . . .	34
Figura 9:	Cursar disciplinas reprovadas no período seguinte. . . . .	35
Figura 10:	Tela de estatísticas do aluno. . . . .	38
Figura 11:	Exemplo de ranqueamento de disciplinas obrigatórias e eletivas. . .	39
Figura 12:	Imagem para ilustrar a situação com disciplinas <i>A</i> e <i>B</i> no 6º pe- ríodo e disciplinas <i>C</i> e <i>D</i> do 7º. A seta vermelha de <i>B</i> para <i>D</i> representa uma situação de pré-requisito. . . . .	40
Figura 13:	Ilustração representando o resultado do ranqueamento. As disci- plinas aparecem na ordem <i>B</i> - <i>A</i> - <i>C</i> , e a disciplina <i>D</i> não compõe o <i>ranking</i> porque tem <i>B</i> como pré-requisito. . . . .	41
Figura 14:	Visualização gráfica do exemplo proposto. . . . .	48
Figura 15:	Lista de disciplinas ofertadas em 2017.2. . . . .	50
Figura 16:	Exemplo de parâmetros de recomendação com filtro de quantidade máxima de 4 disciplinas e apenas eletivas. . . . .	51
Figura 17:	Exemplo de filtro bloqueando horários da manhã para receber dis- ciplinas apenas no período de 13:00 às 17:00 todos os dias. . . . .	51
Figura 18:	Exemplo de grade recomendada utilizando os parâmetros das Fi- guras 16 e 17. . . . .	52
Figura 19:	Número de registros por período. . . . .	59
Figura 20:	Número de eletivas cursadas por período. . . . .	60
Figura 21:	Coleções do MongoDB criadas. . . . .	61
Figura 22:	Questionário para professores - Avaliação das funcionalidades. . . .	67
Figura 23:	Questionário para alunos - Avaliação das funcionalidades. . . . .	68

Figura 24: Avaliação de grade recomendada sem parametrização. . . . .	69
Figura 25: Avaliação de grade recomendada com parametrização. . . . .	69
Figura 26: Semelhança da grade parametrizada com a cursada pelo aluno. . .	70
Figura 27: Questionário do SUS. . . . .	72

## LISTA DE CÓDIGOS

5.1	Estrutura dos registros no Elasticsearch em formato JSON . . . . .	58
D.1	<i>Script</i> para obter a lista de pré-requisitos de cada disciplina de acordo com a tabela obtida na página do SIGA . . . . .	87
D.2	<i>Script</i> para popular as equivalências de cada disciplina de acordo com a tabela obtida na página do SIGA . . . . .	88
D.3	<i>Script</i> para obter as disciplinas obrigatórias faltantes para cada aluno . .	89
D.4	<i>Script</i> para obter as disciplinas ofertadas em 2017-2 . . . . .	90
E.1	Exemplo de <i>Controller</i> (Javascript) . . . . .	91
E.2	Exemplo de <i>Service</i> (Javascript) . . . . .	92
E.3	Exemplo de <i>Schema</i> (Javascript) . . . . .	93

## LISTA DE ABREVIATURAS E SIGLAS

API	Application Programming Interface
BCC	Bacharelado em Ciência da Computação
BOA	Boletim de Orientação Acadêmica
COAA	Comissão de Orientação e Acompanhamento Acadêmico
CR	Coeficiente de Rendimento
DCC	Departamento de Ciência da Computação
GA	Graduating Attribute
HTTP	Hypertext Transfer Protocol
JS	Javascript
JSON	JavaScript Object Notation
MEAN	Mongo + Express + Angular + Node
NoSQL	No Structured Query Language
REST	Representational State Transfer
SIGA	Sistema Integrado de Gestão Acadêmica
SUS	System Usability Scale
TCC	Trabalho de Conclusão de Curso
UFRJ	Universidade Federal do Rio de Janeiro

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>16</b>
1.1	MOTIVAÇÃO	17
1.2	OBJETIVO	18
1.3	ESTRUTURAÇÃO DO TEXTO	18
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA E TRABALHOS RELACIONADOS</b>	<b>20</b>
2.1	SISTEMAS DE RECOMENDAÇÃO	20
2.1.1	Filtragem Colaborativa	20
2.1.2	Filtragem baseada em Conteúdo	22
2.1.3	Filtragem baseada em Conhecimento	23
2.1.4	Abordagens Híbridas	24
2.2	TRABALHOS CORRELATOS	24
<b>3</b>	<b>PESQUISA DE VALIDAÇÃO</b>	<b>27</b>
3.1	PESQUISA COM PROFESSORES	27
3.1.1	Roteiro	27
3.1.2	Entrevistas	28
3.2	PESQUISA COM ALUNOS	30
<b>4</b>	<b>PROPOSTA DO SISTEMA</b>	<b>36</b>
4.1	ESTATÍSTICAS DE USUÁRIO	37
4.2	RANKING DE DISCIPLINAS	39
4.2.1	Ranking de disciplinas obrigatórias	39
4.2.2	Ranking de disciplinas eletivas	41
4.3	SUGESTÃO DE GRADE	49
<b>5</b>	<b>IMPLEMENTAÇÃO DO SISTEMA</b>	<b>54</b>
5.1	TECNOLOGIAS PRINCIPAIS	54
5.1.1	Aplicação WEB ( <i>Client-side</i> )	54
5.1.2	Servidor ( <i>Server-side</i> )	55

5.1.3	Banco de Dados . . . . .	56
5.2	TECNOLOGIAS AUXILIARES . . . . .	57
5.3	OBTENÇÃO E ANÁLISE INICIAL DOS DADOS . . . . .	57
5.4	TRATAMENTO E PERSISTÊNCIA DOS DADOS . . . . .	60
<b>6</b>	<b>AVALIAÇÃO DO SISTEMA . . . . .</b>	<b>65</b>
6.1	AVALIAÇÃO DAS FUNCIONALIDADES . . . . .	66
6.1.1	Resultado das Avaliações das Funcionalidades pelos Professores	66
6.1.2	Resultado das Avaliações das Funcionalidades pelos Alunos . .	67
6.2	AVALIAÇÃO DAS RECOMENDAÇÕES . . . . .	68
6.3	TESTES DE USABILIDADE . . . . .	70
6.3.1	Resultado dos Testes de Usabilidade com Professores . . . . .	73
6.3.2	Resultado dos Testes de Usabilidade com Alunos . . . . .	73
<b>7</b>	<b>CONCLUSÃO . . . . .</b>	<b>74</b>
7.1	DIFICULDADES ENCONTRADAS . . . . .	75
7.2	TRABALHOS FUTUROS . . . . .	76
	<b>REFERÊNCIAS . . . . .</b>	<b>78</b>
<b>A</b>	<b>ROTEIRO DE ENTREVISTA COM PROFESSORES . . . . .</b>	<b>80</b>
A.1	PERGUNTAS GERAIS . . . . .	80
A.2	CENÁRIOS ESPECÍFICOS PARA DISCUSSÃO . . . . .	80
A.3	UTILIDADE DE UM SISTEMA DE RECOMENDAÇÃO NESSE CON- TEXTO . . . . .	81
A.4	FINAL . . . . .	81
<b>B</b>	<b>QUESTIONÁRIO DE PESQUISA COM ALUNOS . . . . .</b>	<b>82</b>
B.1	PERGUNTAS GERAIS . . . . .	82
B.2	CRITÉRIOS PARA ESCOLHA DE DISCIPLINAS ELETIVAS . . . . .	82
B.3	CRITÉRIOS PARA ESCOLHA DE DISCIPLINAS OBRIGATÓRIAS	83
B.4	FUNCIONALIDADES DO SISTEMA DE RECOMENDAÇÃO DE DIS- CIPLINAS . . . . .	84

<b>C</b>	<b>QUESTIONÁRIO DE AVALIAÇÃO DO SISTEMA . . . . .</b>	<b>85</b>
C.1	AVALIAÇÃO DA RECOMENDAÇÃO . . . . .	85
C.2	AVALIAÇÃO DAS FUNCIONALIDADES . . . . .	85
C.3	AVALIAÇÃO DE USABILIDADE . . . . .	86
C.4	CONSIDERAÇÕES FINAIS . . . . .	86
<b>D</b>	<b>CÓDIGOS PYTHON . . . . .</b>	<b>87</b>
<b>E</b>	<b>ARQUITETURA BACK-END . . . . .</b>	<b>91</b>

## 1 INTRODUÇÃO

Devido à contínua evolução da era da informação nos anos recentes, principalmente depois do surgimento da Web, é produzido cada vez mais conteúdo em meio digital. Estes dados podem estar nos mais diversos formatos, por exemplo, textos não estruturados em redes sociais, fotos, vídeos, artigos científicos, catálogos de lojas, entre outros. É fácil perceber a importância de se ter maneiras eficientes de buscar informações em meio a tantos dados, caso contrário, estas informações não teriam muito uso devido à extrema dificuldade de recuperação. Um exemplo deste fenômeno seria tentar buscar alguma informação na Web sem o auxílio de um motor de busca.

Neste contexto, a Recuperação de Informação (RI) faz-se importante. Existem diversas conceituações de RI. Em uma delas, Manning et al. [13] afirmam que Recuperação da Informação consiste em encontrar material de natureza não estruturada que satisfaça uma necessidade de informação em meio a uma ampla coleção de dados. Os Sistemas de Recomendação fazem parte de uma importante área de pesquisa de Recuperação da Informação. Atualmente, existem componentes de recomendação implementados para os mais diversos contextos, por exemplo: ao acessar um sistema de *e-commerce* e ver mensagens do tipo “Usuários que compraram este item também compraram”, ao receber sugestões de artistas e músicas no Spotify<sup>1</sup> ou sugestões de séries e filmes no Netflix<sup>2</sup>.

Um desses contextos é relacionado à área acadêmica, onde cada vez mais são desenvolvidos sistemas para auxiliar alunos e professores. Existem sistemas, por exemplo, que ajudam alunos já formados na graduação a encontrar boas universidades para continuar os estudos em especialização, mestrado ou doutorado baseado em um modelo de filtragem colaborativa [11]. Também existem projetos para auxiliar alunos a encontrar materiais relevantes para estudo [15] e para recomendar disciplinas eletivas com base nos graus obtidos nas outras disciplinas, utilizando técnicas de *machine learning* para processar grandes volumes de dados [9].

---

<sup>1</sup><https://www.spotify.com>

<sup>2</sup><https://www.netflix.com>

## 1.1 MOTIVAÇÃO

A atual estrutura curricular do Bacharelado em Ciência da Computação (BCC) da UFRJ passou a vigorar no primeiro semestre de 2010, com integralização em 9 semestres letivos, e carga horária total de 3.140 horas, equivalentes a 190 créditos<sup>3</sup>. A atual grade curricular consiste em 34 disciplinas de caráter obrigatório, divididas em 8 períodos, mais 40 créditos em disciplinas eletivas (normalmente, 10 disciplinas eletivas que devem ser cursadas, escolhidas dentre as diversas que são ofertadas em cada período, podendo ser 1 delas de livre escolha, ou seja, pertencente a outro curso). Algumas disciplinas obrigatórias possuem pré-requisitos de outras disciplinas obrigatórias que devem ser cursadas previamente. Já as eletivas, podem ou não ter disciplinas de pré-requisito, dependendo do conteúdo abordado. Além disso, é obrigatória a realização de 200 horas em atividades complementares e, ao final da graduação, apresentar um Trabalho de Conclusão de Curso (TCC).

Atualmente, os alunos do BCC encontram dificuldades em visualizar de maneira rápida e adequada sua atual situação no curso: quais disciplinas foram concluídas e quais ainda estão pendentes, como está o seu rendimento, entre outras informações. Isso ocorre devido ao fato do Sistema Integrado de Gestão Acadêmico (SIGA<sup>4</sup>) atualmente utilizado não ter implementada uma interface de visualização da situação do aluno no curso. Em sua versão atual, é necessário efetuar o download em PDF do Boletim de Orientação Acadêmica (BOA), que contém, em formato textual, todas as informações acadêmicas do aluno. Além disso, por conta da vasta grade curricular do BCC, alguns alunos podem acabar encontrando dificuldade em descobrir suas áreas de interesse e especializações.

Este problema também é enfrentado pelos professores que são orientadores acadêmicos. Para fazer uma orientação adequada, os orientadores acadêmicos precisam analisar cuidadosamente a situação de cada orientando e planejar da melhor forma o plano de curso do aluno. Para auxiliar nessa tarefa, em geral, os professores analisam o BOA de cada aluno orientado, tentando entender tanto os aspectos acadêmicos

---

<sup>3</sup><http://dcc.ufrj.br/ensino/graduacao/sobre-o-curso>

<sup>4</sup><https://intranet.ufrj.br/>

(número de reprovações, disciplinas em média que são cursadas por período, CR - Coeficiente de Rendimento, etc.) quanto pessoais (o quanto o aluno se sente confortável fazendo determinada disciplina, número de disciplinas por período que são bem aproveitadas, etc.). O objetivo final é conseguir, conjuntamente com o aluno, planejar e recomendar cada disciplina, e isso tudo sem nenhum auxílio de visualização gráfica dos dados.

## 1.2 OBJETIVO

Diante do cenário exposto na motivação, o sistema proposto neste Projeto Final de Curso tem como objetivo principal auxiliar alunos e orientadores acadêmicos tanto na questão de visualização dos dados quanto na recomendação de disciplinas a serem cursadas nos próximos períodos letivos. A visualização dos dados apresenta dados estatísticos sobre o progresso do aluno no curso, tais como: período atual, CR por período, quantas disciplinas obrigatórias e eletivas já foram feitas e quantas ainda faltam, tabela com histórico de disciplinas cursadas, etc. A recomendação de disciplinas utiliza uma abordagem híbrida, combinando a técnica de filtragem colaborativa com a recomendação baseada em conhecimento (Seção 2.1.1), para o cálculo das recomendações, exibindo a lista de disciplinas ranqueadas e possibilitando parametrizações da grade, tais como: número de disciplinas a serem cursadas, horários, opções de fixar/remover disciplinas, etc. Para tanto, o sistema foi construído levando em consideração visões e opiniões de professores e alunos no que diz respeito aos aspectos adequados e inadequados durante a inscrição em disciplinas.

## 1.3 ESTRUTURAÇÃO DO TEXTO

O restante do texto está estruturado da seguinte forma:

- No Capítulo 2, é explicado o conceito de Sistemas de Recomendação dentro da área de Recuperação de Informação, descrevendo os diferentes tipos de abordagens e algoritmos existentes e definindo qual tipo se aplica no presente

projeto. Também são apresentados alguns trabalhos relacionados com o tema, mostrando as diferenças e semelhanças entre eles e o trabalho proposto.

- No Capítulo 3, é descrito todo o processo de pesquisa e validação do sistema com os professores, desde a criação do roteiro até as entrevistas individuais, e com os alunos, aplicando um questionário para entender um pouco sobre perfil de cada um deles.
- No Capítulo 4, é apresentada a proposta do sistema, sendo detalhados os algoritmos e estratégias utilizadas para a recomendação de disciplinas, funcionalidades implementadas e a visualização das estatísticas dos dados dos alunos.
- No Capítulo 5, é descrito como o sistema foi implementado, as principais tecnologias utilizadas e como os dados foram obtidos, tratados e analisados inicialmente.
- No Capítulo 6, são apresentados os resultados obtidos através das experimentações com alunos e professores, mostrando suas opiniões sobre o sistema e sugestões de melhoria.
- Por fim, no Capítulo 7, são apresentadas as conclusões sobre o desenvolvimento do projeto, trazendo dificuldades enfrentadas durante o processo, assim como possíveis ideias para trabalhos futuros.
- Nos Apêndices, encontram-se o roteiro de entrevista com os professores, o questionário de pesquisa com alunos, os questionários aplicados para avaliação do sistema e alguns códigos implementados.

## 2 FUNDAMENTAÇÃO TEÓRICA E TRABALHOS RELACIONADOS

### 2.1 SISTEMAS DE RECOMENDAÇÃO

Sistemas de Recomendação têm como objetivo mostrar ao usuário recomendações de itens que possivelmente sejam de seu interesse, normalmente buscando mostrar itens de maneira personalizada, ou seja, única para cada usuário. O poder destes sistemas é encontrar informações que o usuário não está ativamente procurando, mas que ainda assim são relevantes para ele. É importante perceber que, para ser capaz de personalizar a experiência de cada usuário de maneira adequada, estes sistemas precisam conhecer previamente informações sobre seus usuários.

Existem diversas técnicas e mecanismos que podem ser utilizados ao construir um Sistema de Recomendação. Nas subseções abaixo, serão discutidos estes mecanismos e técnicas, descritas em [12], ressaltando suas principais diferenças, vantagens, desvantagens e exemplos de aplicação.

#### 2.1.1 Filtragem Colaborativa

A ideia principal de abordagens colaborativas [10] é utilizar dados de interesse de outros usuários do sistema para realizar recomendações, acreditando no fato de que usuários que compartilharam dos mesmos interesses no passado continuam concordando no futuro. Por exemplo, se um usuário  $X$  de um *e-commerce* possui um padrão de compra semelhante ao de um usuário  $Y$  e o usuário  $X$  tem interesse em um item que o usuário  $Y$  não conhece, recomenda-se este item para o usuário  $Y$ . Estas abordagens tentam reproduzir o processo natural de recomendação que ocorre quando pessoas indicam produtos a outras.

Uma característica importante deste tipo de abordagem é que não é necessário possuir nenhum tipo de conhecimento sobre os itens que estão sendo recomendados, basta saber quais usuários têm interesse em quais itens para realizar a recomendação. Este fator possibilita a aplicação desta abordagem nos mais diversos contextos,

gerando uma versatilidade na utilização de dados que podem ser desde dados implícitos como histórico de compras ou de notas de alunos, até dados explícitos como avaliações de itens. Além disso, esse tipo de abordagem possui a capacidade de identificação de comportamentos similares, sendo muito popular e vastamente utilizada em diversos sistemas, como podemos ver em alguns trabalhos correlatos apresentados na Seção 2.2.

Existem variações deste tipo de abordagem, ambas visam recomendar itens que um dos usuários demonstrou interesse e o outro desconhece. Estas variações acontecem devido à maneira de calcular similaridades, se será entre usuários ou itens. Quando as similaridades calculadas são entre usuários, a solução denomina-se *User-user*. Também é possível olhar para os itens na hora de calcular as similaridades, esta última abordagem chama-se *Item-item*.

A abordagem *User-user* possui algumas limitações: (i) problemas de escalabilidade provenientes de grandes quantidades de usuários em sistemas deste tipo; (ii) *cold-start* (partida fria) de usuários, ou seja, quando um novo usuário aparece não é possível realizar nenhum tipo de recomendação para este até que ele estabeleça um padrão de comportamento (realize diversas avaliações em um sistema baseado em avaliações, ou realize compras em um sistema baseado em compras, por exemplo); (iii) problemas de *cold-start* de itens, ou seja, quando um novo item aparece ele inicialmente não será recomendado; (iv) grandes bases de dados podem possuir uma enorme quantidade de itens e usuários, fazendo com que seja difícil identificar usuários que avaliaram os mesmos itens para medir suas similaridades; (v) usuários com preferências distintas dos demais também podem ser um problema, já que sua similaridade com outros pode ser muito baixa.

A abordagem *Item-item* também possui algumas limitações, como, por exemplo o *cold-start* de itens, mas apresenta algumas vantagens sobre a anterior, como: (i) lidar com novos usuários passa a não ser tão problemático, visto que apenas uma avaliação/compra pode o tornar apto a receber recomendações de itens similares; (ii) a base de itens tende a crescer mais estável que a base de usuários, configurando um problema computacionalmente mais fácil em relação à abordagem *User-user*.

### 2.1.2 Filtragem baseada em Conteúdo

Sistemas de recomendação baseados em conteúdo usam dados associados com os itens para realizar a recomendação. Estes dados podem ser de qualquer natureza, como por exemplo, um sistema de recomendação de músicas poderia utilizar o artista, gênero da música, álbum, ano, entre outros. A intuição é de que estes dados podem definir a relevância de um item para um usuário, ou seja, no sistema de recomendação de músicas descrito acima, o gênero pode ser um fator determinante para um usuário.

As principais vantagens deste tipo de abordagem são: não é necessária uma base de outros usuários do sistema para realizar recomendações e não existem problemas de partida fria de itens, ou seja, quando o sistema recebe novos itens é possível recomendá-los, desde que eles possuam características associadas a ele. Porém, nesse tipo de abordagem, há problemas de *cold-start* para novos usuários já que é necessário conhecer os gostos do usuário para gerar recomendações a este.

Propriedades interessantes deste tipo de abordagem são: (i) a liberdade de escolha de características dos itens a serem utilizadas, o que permite uma etapa inicial de estudo sobre quais os atributos são mais relevantes para o usuário e (ii) a maneira de utilização destas características. Estas propriedades podem ser uma desvantagem em função dessa necessidade de haver uma descrição dessas características ou conteúdos dos itens a serem recomendados. O sistema pode tentar obter estes dados automaticamente, ou então eles podem ser inseridos manualmente. É importante salientar que qualquer dado associado ao item pode vir a ser utilizado na recomendação, mas somente a escolha adequada destes dados possibilitará que boas recomendações sejam geradas. Para gerar as recomendações, nesse tipo de abordagem, pode-se fazer uso de algumas das técnicas clássicas da área de Recuperação de Informação [3], tais como: os modelos booleano, vetorial ou probabilístico.

### 2.1.3 Filtragem baseada em Conhecimento

Uma abordagem de recomendação baseada em conhecimento [6] envolve uma maior interação com o usuário e menor necessidade de dados históricos. Por exemplo, um sistema deste tipo, para recomendar *smartphones*, deve ser capaz de recomendar itens para um novo usuário baseado em filtros e configurações fornecidas pelo mesmo, bem como dados sobre ele já previamente conhecidos (histórico). Uma possível utilização deste sistema seria o usuário configurar um intervalo de preços que ele está disposto a pagar, o tamanho da tela e uma marca específica, e então o sistema deve ser capaz de fazer uma recomendação personalizada para este usuário, baseada nesta configuração.

É importante notar que apenas exibir itens que satisfaçam regras ou restrições dos usuários não é suficiente, já que o ideal é um sistema de recomendação ser capaz de personalizar a experiência para cada usuário. Portanto, este tipo de sistema ainda precisa manter algum tipo de perfil de usuário. No exemplo de um sistema para recomendação de *smartphones*, mencionado anteriormente, o sistema poderia perguntar ao usuário quais características são mais importantes e calcular um ranqueamento baseado nestas informações.

Outra característica importante deste tipo de sistema é o conhecimento de domínio dos itens recomendados, em contraste a sistemas puramente colaborativos. A interação com o usuário faz este conhecimento necessário, já que, sem isso, seria impossível ou ineficiente construir um sistema desse tipo. Basta imaginar o sistema de recomendação de *smartphones*, citado como exemplo anteriormente, implementando apenas um filtro genérico por faixa de preços. Isso não auxiliaria tanto o usuário na hora da decisão, por não considerar outras características mais específicas sobre o produto selecionado, tais como: tamanho da tela, resolução da câmera, sistema operacional, marca, entre outras.

### 2.1.4 Abordagens Híbridas

Dependendo dos dados disponíveis para realização da recomendação, é possível construir sistemas que combinam as abordagens citadas nas seções anteriores (Seções 2.1.1, 2.1.2 e 2.1.3), com a intenção de minimizar seus pontos fracos e evidenciar seus pontos fortes. Estes sistemas, compostos de abordagens combinadas, são os sistemas de recomendação híbridos. Um possível exemplo da utilização desta técnica pode ser uma variação do sistema de músicas exemplificado na Seção 2.1.2: basta imaginar que, além dos dados como artista, gênero, álbum e ano, o sistema também tem acesso a dados históricos de preferências de usuários. É possível combinar a abordagem baseada em conteúdo com a abordagem colaborativa para obter um resultado mais refinado.

## 2.2 TRABALHOS CORRELATOS

No contexto de outras universidades, foram propostos sistemas de recomendação de disciplinas para seus alunos. Nesta seção, são descritos alguns trabalhos relacionados, sendo explicitadas as principais semelhanças e diferenças entre estes e o sistema de recomendação proposta no presente projeto.

Os principais aspectos analisados foram: (i) os tipos de dados usados para calcular as recomendações, (ii) os critérios que foram levados em consideração e (iii) as técnicas utilizadas.

Nos aspectos tipos de dados e critérios utilizados, alguns dos trabalhos estudados contam com dados diferentes dos disponíveis no SIGA. Um exemplo é o trabalho [4], correspondente ao Sistema da Universidade de Alberta - Canadá, que utiliza parâmetros denominados *Graduating Attributes* (GA's) na sua recomendação. Estes parâmetros representam habilidades que o graduando deve adquirir durante a graduação, são 7 *GA's* que vão além do conhecimento técnico adquirido na disciplina, e os alunos avaliam todas as disciplinas segundo estes parâmetros ao cursá-las.

Em contrapartida, existem vários trabalhos que utilizam dados relacionados com

o desempenho do aluno nas disciplinas, pré e pós requisitos, professores ministrantes, horários, dentre outros dados que o SIGA também disponibiliza. Em [8], por exemplo, utilizam-se notas nos cursos anteriores e professores ministrantes dos cursos em sua recomendação. Em [7], utilizam-se as inscrições de alunos similares, bem como cursos oferecidos por professores favoritos e cursos inscritos por amigos.

Também existem aqueles sistemas que fazem uso de avaliações e questionários preenchidos pelos alunos como algum tipo de critério para usar em suas recomendações. Em [14], por exemplo, é proposto um sistema em que os alunos avaliam disciplinas cursadas segundo um grau de adaptabilidade, que será utilizado posteriormente para traçar um perfil para o aluno, para realizar as recomendações.

Sobre as técnicas utilizadas, é possível perceber uma predominância na utilização de abordagens de Filtragem Colaborativa e suas variações, e uma considerável utilização de algoritmos baseados em Regras de Associação. Em [4], por exemplo, foi implementado um Sistema de Filtragem Colaborativa com diversos critérios (*multicriteria rating*), um para cada GA. Em [8], são utilizadas três técnicas diferentes em sua recomendação: Filtragem Colaborativa baseada em usuários, Regras de Associação baseadas em regras e conteúdo. Já em [7], faz-se tanto o uso de Filtragem Colaborativa baseada em usuários, quanto de algoritmos de classificação e regressão.

Existem também trabalhos que utilizam outros tipos de algoritmos em suas recomendações. Em [1], por exemplo, são utilizadas algoritmos de clusterização e classificação, em conjunto com algoritmos de regras de associação em sua recomendação.

O interessante dos trabalhos correlatos estudados é que, apesar da ideia e objetivos serem muito similares, as peculiaridades das universidades e alunos em cada uma delas têm muita relevância na proposição dos sistemas. Em [7], os autores constataram que os alunos têm uma preferência por disciplinas que têm alto nível de aprovação, e o sistema utilizava esse fato em sua proposição. Em [7] e [8], os autores indicaram que considerar os professores ministrantes era um critério importante. Em [4], é levado em conta o conceito de *Graduating Attributes* (GA's), e faz-se uso deste no núcleo de funcionamento do seu sistema.

A partir da revisão bibliográfica realizada, fez-se necessário um estudo aprofundado acerca das peculiaridades do BCC e dos dados disponíveis no SIGA para formular uma proposta apropriada de um sistema de recomendação para esta realidade. O sistema proposto faz uso das disciplinas cursadas pelos alunos, bem como seus desempenhos nelas, para gerar as recomendações. A recomendação de disciplinas proposta utiliza uma abordagem híbrida, combinando a técnica de filtragem colaborativa com a recomendação baseada em conhecimento em seu cálculo.

Além disso, o sistema possui diversas possibilidades de configuração, definidas com base no que os alunos do BCC relataram utilizar no momento de montar sua grade horária, tais como: filtros de horário, configuração de limite e tipos de disciplinas recomendados, entre outras. Ademais, são fornecidas visualizações sobre o progresso do aluno no curso, tais como: período atual, CR por período, quantas disciplinas obrigatórias e eletivas já foram feitas e quantas ainda faltam, tabela com histórico de disciplinas cursadas, etc. Estas definições foram obtidas a partir de uma etapa inicial de pesquisa, que é descrita no Capítulo 3. Já os detalhes da proposta realizada são discutidos no Capítulo 4.

## 3 PESQUISA DE VALIDAÇÃO

### 3.1 PESQUISA COM PROFESSORES

Com a finalidade de definir critérios adequados para serem utilizados nas recomendações e também funcionalidades da ferramenta, foi elaborado um roteiro de perguntas (Apêndice A). O roteiro foi aplicado em formato de entrevista individual, feita com professores do Departamento de Ciência da Computação (DCC) da UFRJ, que desempenham o papel de orientador acadêmico.

Nas seções a seguir (Seções 3.1.1 e 3.1.2), é descrito detalhadamente o roteiro utilizado para entrevistar os professores, bem como o processo das entrevistas em si.

#### 3.1.1 Roteiro

O roteiro de entrevista com os professores foi organizado em 3 seções principais: (i) perguntas gerais sobre a orientação acadêmica, (ii) cenários específicos para discussão e (iii) utilidade de um sistema de recomendação.

A primeira seção teve como objetivo descobrir quais estratégias são utilizadas durante a orientação acadêmica dos alunos, com a finalidade de entender quais critérios são considerados adequados e inadequados no momento de escolher quais disciplinas serão cursadas no período.

A segunda seção teve como objetivo discutir alguns cenários específicos que podem ser interessantes para a recomendação. Tais cenários são:

- É válido sempre tentar minimizar o tempo para conclusão de curso, mas levando em consideração as chances de aprovação do aluno?
- É importante considerar o grau de dificuldade das disciplinas neste momento?  
Se sim, como inferir este grau de dificuldade?
- O professor que ministrará a disciplina é um fator que deve ser considerado?

- Existe alguma recomendação especial para um aluno que reprovou disciplinas recentemente? É melhor cursar as disciplinas reprovadas o mais rápido possível ou aguardar um tempo? Como lidar com reprovação de disciplinas eletivas?
- Existem cuidados especiais que são tomados ao orientar alunos próximos de se enquadrar em algum critério de jubramento?
- E finalmente, quais fatores devem ser considerados na escolha de disciplinas eletivas? É importante focar em uma área de atuação ou é melhor diversificar o currículo? Como identificar a aptidão de um aluno por uma área de atuação? Será que é possível inferir essa aptidão através do desempenho em disciplinas obrigatórias?

A terceira e última parte teve como objetivo identificar características que um sistema de recomendação deve ter para ajudar no processo de escolha de disciplinas, e também saber se este tipo de sistema poderia ser útil para os professores orientadores. Nesta parte foi aberto um espaço para sugestões de ideias para o sistema (que seria futuramente implementado pelo presente projeto de conclusão).

### **3.1.2 Entrevistas**

Para realizar as entrevistas foram combinados horários com cada professor individualmente. As entrevistas tiveram uma duração média de 30 minutos, onde foram discutidos todos os assuntos descritos no roteiro descrito na Seção 3.1.1. O objetivo era entrevistar professores que têm ou já tiveram envolvimento com a Comissão de Orientação e Acompanhamento Acadêmico (COAA)<sup>1</sup> ou que possuam experiência em orientação acadêmica de alunos. Foram realizadas entrevistas com um total de sete professores orientadores.

Ao final das entrevistas, foi possível perceber que os professores possuem estratégias diferentes para abordar seus orientandos e chegar a uma proposta de orientação, mas existe um consenso sobre a maior parte dos temas discutidos. Dentre eles, estão

---

<sup>1</sup>Tem a função de, num trabalho conjunto com o orientador acadêmico do aluno, acompanhar, academicamente, os alunos.

as técnicas de escolha de disciplinas para compor a grade do aluno, levando em consideração a priorização de disciplinas atrasadas, para seguir com o planejamento do curso (em relação às disciplinas obrigatórias), e buscar escolhas de disciplinas eletivas que possam agregar um valor maior a cada aluno em particular, construindo um perfil mais conciso. Além disso, viu-se importante a análise individual de cada aluno com relação aos fatores pessoal, físico e mental, entendendo como cada um reage, e quantas disciplinas conseguem fazer com bom desempenho por período. Por fim, o desafio é propor uma orientação que possa balancear o número de disciplinas que não atrasem o aluno durante o curso, mas também que não o sobrecarregue de maneira exacerbada. Foi compreendido também que escolher disciplinas simplesmente pelo horário ou professor, são más práticas durante a escolha da grade, e para alguns, a escolha por minimizar o tempo de conclusão, sem considerar o aspecto pessoal, comentado anteriormente, também pode acabar prejudicando mais o aluno.

Com as entrevistas, também foi possível definir boas funcionalidades ao sistema que pudessem agregar valor ao orientador durante a conversa com cada orientando, mas que também auxiliassem os alunos na escolha de suas grades. Foram sugeridos: filtros de quantidade de créditos, número total de disciplinas, filtros por horários ou dias da semana, recomendar disciplinas que o aluno teve reprovação no último período, sempre priorizar disciplinas obrigatórias, exibir grades históricas para o aluno acompanhar seu progresso no curso.

Por fim, duas principais dificuldades foram apontadas por quase todos os professores. A primeira foi sobre como realimentar os dados provenientes do SIGA de maneira automática no sistema e fazer o com que ele seja produtivo em todo início de período letivo. A segunda foi a dificuldade de levar em consideração as questões pessoais do aluno, ou seja, por mais que tenhamos acesso aos dados do desempenho do aluno nos períodos anteriores, não seremos capazes de inferir a saúde mental do aluno, e isso tem uma grande influência no processo de inscrição em disciplinas.

### 3.2 PESQUISA COM ALUNOS

Assim como para os professores durante a orientação, a opinião dos alunos também é vista como de extrema importância para entender quais critérios são utilizados no momento da escolha de disciplinas a serem cursadas. Com isso, foi realizada uma pesquisa com os alunos através de um questionário (Apêndice B), com o objetivo de alcançar o maior número de alunos possível e validar as ideias do projeto.

O questionário foi dividido em três seções principais. Primeiramente, foram abordadas questões gerais, referentes ao progresso do aluno no curso, como, por exemplo: (i) em que período se encontra, (ii) qual período ele considera ideal para começar a cursar disciplinas eletivas, (iii) qual o número preferível de disciplinas para cursar a cada período e (iv) também qual o período mais adequado para realizar alguma atividade complementar. Na segunda seção, foram feitas perguntas sobre os critérios utilizados pelo aluno na escolha de disciplinas eletivas e obrigatórias. Por fim, na terceira seção, foram avaliados quais aspectos o aluno esperava que um sistema de recomendação de disciplinas contemplasse para que fosse útil para o aluno.

Um total de 48 respostas foram obtidas. Em resumo, tivemos 70,8% de alunos acima do sexto período (Figura 1). Dentre eles, 95,8% responderam que já fizeram alguma atividade complementar (Figura 2) e a maioria, 91,8%, iniciou essas atividades antes do quinto período (Figura 3). Outro tema abordado no questionário foi sobre o número de disciplinas que é considerado ideal a ser cursado em um período, sendo que 58,3% disseram que cinco disciplinas é o ideal quando não fazem alguma atividade complementar (Figura 4). Porém, quando fazem alguma atividade complementar, 66,7%, preferem cursar apenas quatro disciplinas (Figura 5).

48 respostas

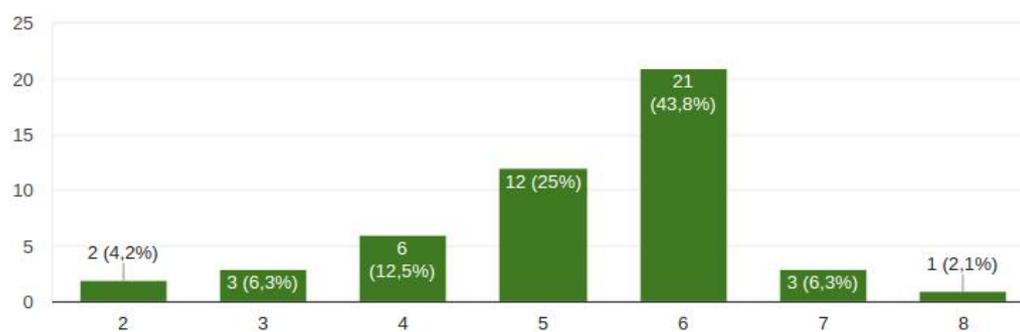


Figura 1: Período dos alunos entrevistados.

48 respostas

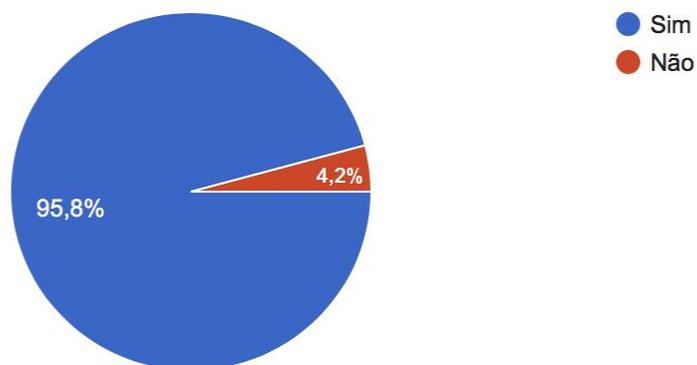


Figura 2: Quantidade de alunos que fazem alguma atividade complementar.

48 respostas

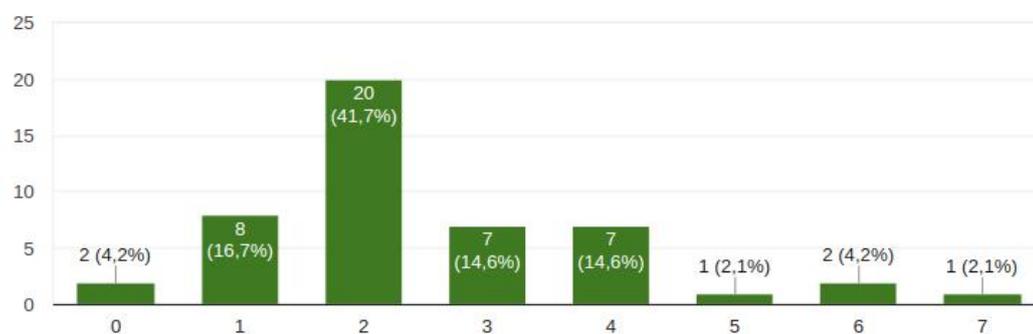


Figura 3: Período em que os alunos iniciam alguma atividade complementar.

48 respostas

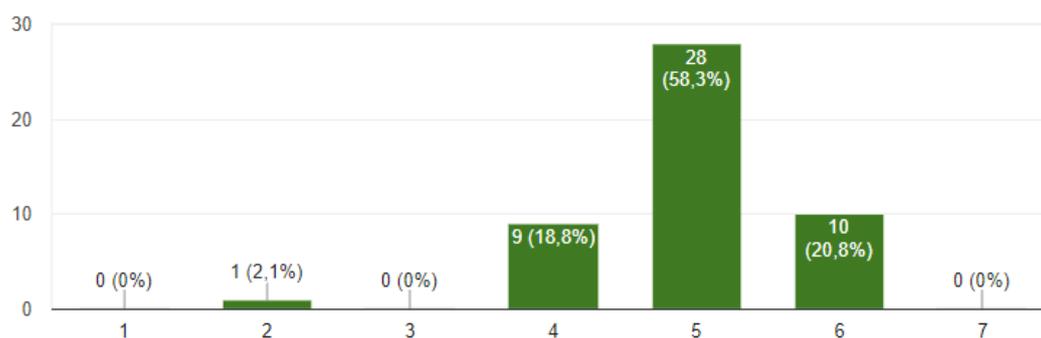


Figura 4: Número de disciplinas sem cursar atividade complementar.

48 respostas

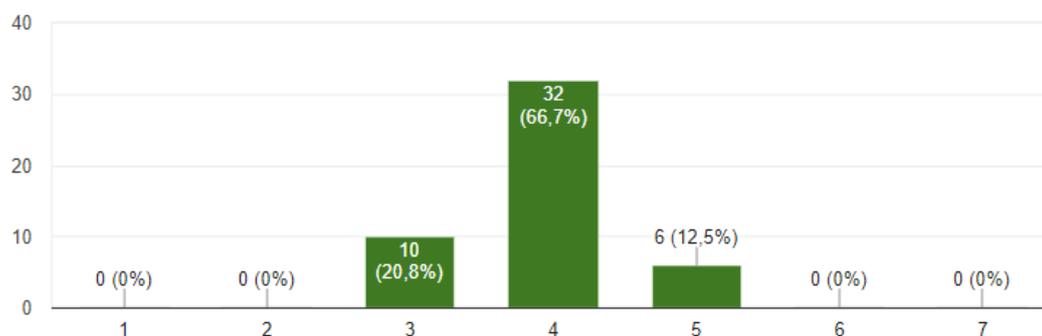


Figura 5: Número de disciplinas cursando atividade complementar.

Sobre a escolha de disciplinas eletivas, foram avaliados diversos critérios de acordo com seu grau de relevância. Os critérios que tiveram resultados que os alunos consideram mais relevantes são (Figura 6): (i) grau de dificuldade da disciplina, (ii) horário em que a disciplina é ofertada, ambos com 75% das opiniões e (iii) a área da disciplina, com 87,5%.

Outro ponto interessante sobre a escolha de disciplinas eletivas (Figura 7) foi que 56,3% dos alunos responderam que preferem cursar disciplinas de diversas áreas inicialmente, e, depois que obtém afinidade com alguma, procuram focar nela. Já 25% preferem sempre diversificar entre as áreas e 16,7% procuram focar em uma área específica desde o começo.

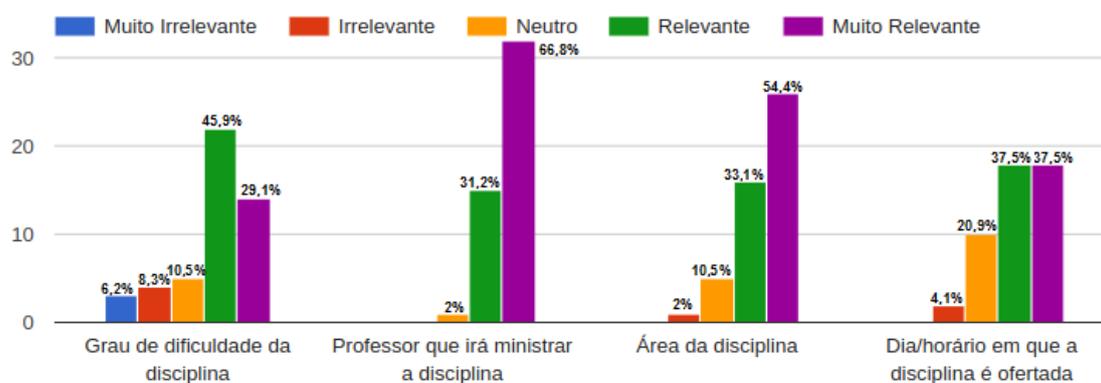


Figura 6: Critérios mais relevantes para escolha de eletivas.

48 respostas

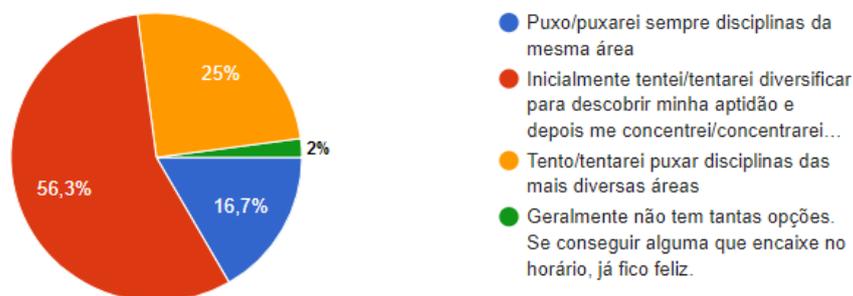


Figura 7: Diversificação de disciplinas eletivas.

Sobre a escolha de disciplinas obrigatórias (Figura 8), os critérios avaliados na pesquisa que chamaram mais atenção foram: a maioria dos alunos consideram mais relevante saber qual o professor irá ministrar a disciplina, assim como preferem priorizar as disciplinas que são pré-requisitos de outras.

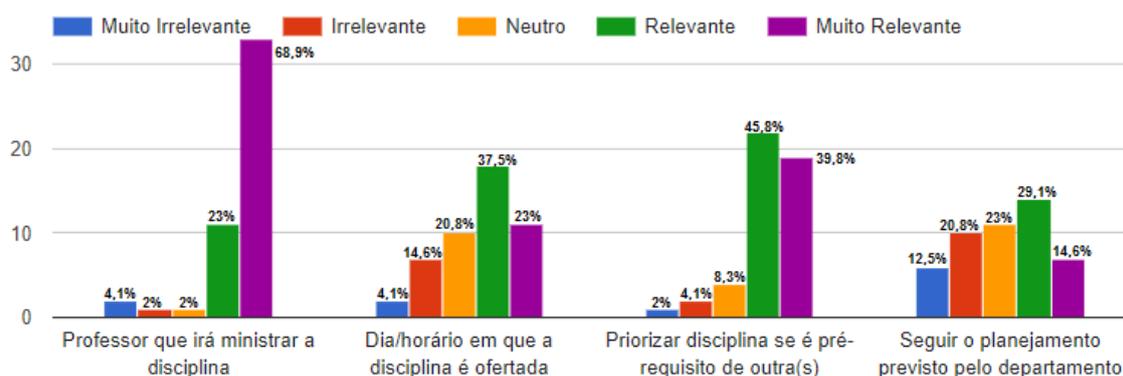


Figura 8: Critérios mais relevantes para escolha de obrigatórias.

Quando perguntados sobre casos de reprovação em disciplinas obrigatórias (Figura 9), 68,8% dos alunos responderam que preferem cursar aquela disciplina imediatamente no período posterior, já 27% disseram que depende de alguns aspectos, como o professor que irá lecionar ou se a disciplina é pré-requisito de outras.

48 respostas

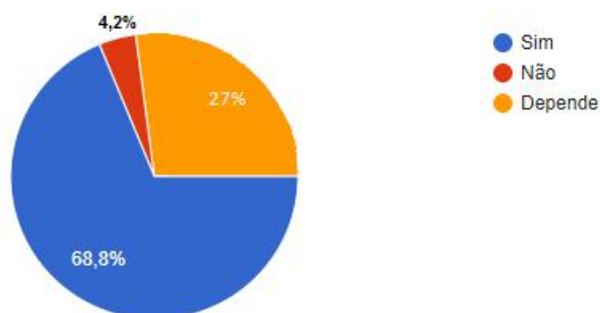


Figura 9: Cursar disciplinas reprovadas no período seguinte.

A respeito das funcionalidades que o sistema deveria contemplar, além da recomendação da grade do período seguinte, os alunos consideraram relevantes os seguintes itens:

- Visualizar histórico de grades cursadas anteriormente;
- Habilitar/Desabilitar recomendação de eletivas;
- Permitir filtrar disciplinas por horário na recomendação de grade;
- Permitir fixar disciplinas (as quais o aluno pretende cursar) para que o sistema só complete a grade com as recomendações;
- Exibir lista de disciplinas eletivas recomendadas por ordem de relevância.

## 4 PROPOSTA DO SISTEMA

Considerando os dados coletados nas pesquisas descritas no capítulo anterior (Capítulo 3), foi elaborada uma proposta de sistema. O sistema foi proposto a partir dos resultados obtidos nas pesquisas com professores e alunos e também com base nos trabalhos relacionados. Cabe salientar que o projeto se direciona somente aos alunos do BCC, como forma de reduzir o escopo e ser capaz de ser testado e avaliado. Entretanto, a proposta pode ser adaptada para ser utilizada em outros cursos.

A proposta consiste em três funcionalidades principais: (i) uma área de resumo de situação do curso, onde o aluno/professor orientador consegue rapidamente ter um panorama do progresso do aluno no curso; (ii) uma área de ranqueamento de disciplinas obrigatórias e eletivas, exibindo uma lista geral de todas as disciplinas oferecidas ao BCC, ordenadas por um *score* de relevância; e (iii) uma área de recomendação de grade, onde são apresentadas sugestões de grades prontas para o aluno.

O sistema possui três páginas principais, onde é possível visualizar as funcionalidades implementadas. A página inicial contém informações do aluno, como estatísticas do progresso no curso, gráficos com o CR por período e uma área com o histórico de grades, que foram percebidas nas pesquisas como informações que agregariam bastante valor tanto para alunos quanto para professores. Uma segunda página contempla uma lista ranqueada das disciplinas obrigatórias que o aluno está apto a cursar no próximo período e uma lista ranqueada das disciplinas eletivas oferecidas pelo BCC ordenadas por um *score*. Por fim, uma terceira página capaz de realizar recomendação de grades, onde o aluno pode definir diferentes parametrizações, como fixação/remoção de disciplinas e horários na grade final, recebendo, ao final, grades prontas como recomendação.

Detalhes de implementação e funcionalidade destas páginas são apresentadas nas seções a seguir (Seções 4.1, 4.2 e 4.3).

#### 4.1 ESTATÍSTICAS DE USUÁRIO

O objetivo desta página é auxiliar o aluno ou professor orientador a ter um panorama do progresso do aluno no curso. Isso é importante principalmente para os professores porque é uma tarefa trabalhosa interpretar o Boletim ou BOA do aluno, que são informações textuais. Seria melhor se esta informação fosse visual, e é isso que esta página propõe-se a apresentar.

As informações disponibilizadas nesta página, mostrada na Figura 10, são: Período Atual do aluno; Disciplinas obrigatórias cursadas sobre total de obrigatórias; Disciplinas Eletivas cursadas sobre quantidade necessária de eletivas; CR Acumulado; Gráfico de CR do aluno por período cursado; Gráfico de quantidade de disciplinas cursadas por período; Histórico completo do aluno, mostrando todas as disciplinas cursadas em cada período e a situação do aluno neste curso.

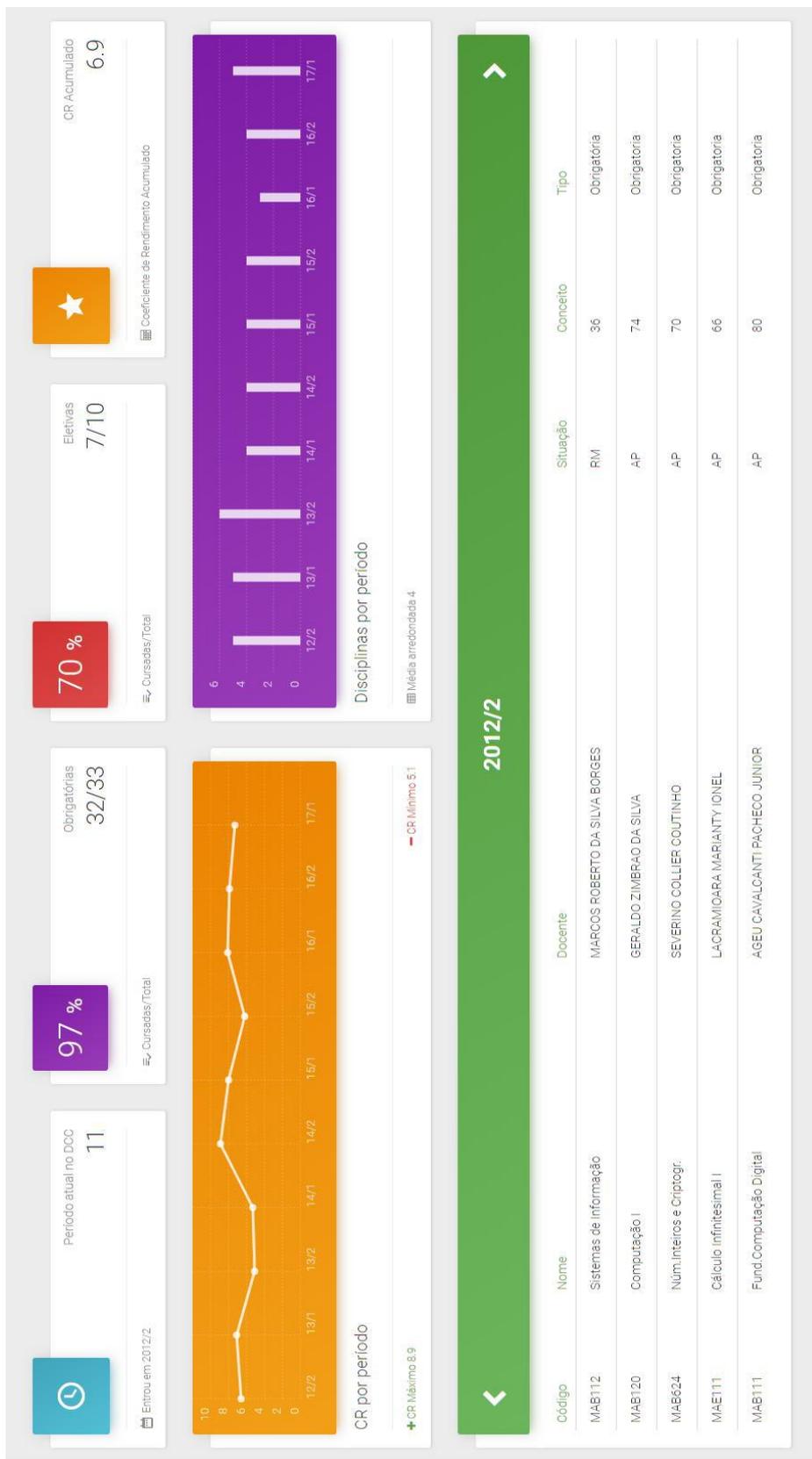


Figura 10: Tela de estatísticas do aluno.

## 4.2 RANKING DE DISCIPLINAS

Nesta página, é possível visualizar dois *rankings* separados: um para disciplinas obrigatórias e um para disciplinas eletivas. A ideia desta página é prover ao aluno a lista de disciplinas que o sistema considera adequada a ele, mesmo que estas disciplinas não estejam sendo ofertadas no período atual. Isso porque existem disciplinas eletivas que não são ofertadas sempre, mas podem ser pertinentes para algum aluno em particular.

Os *rankings* de obrigatórias (Seção 4.2.1) e eletivas (Seção 4.2.2) são construídos de maneiras completamente diferentes, visto que todas as disciplinas obrigatórias devem ser cursadas por todos os alunos em algum momento, ao contrário das eletivas. A Figura 11 mostra um exemplo desta página.

The image shows a screenshot of a web interface with two ranking tables. The top table is titled 'Disciplinas Obrigatórias' and has a purple header. The bottom table is titled 'Disciplinas Eletivas' and has a red header. Both tables list disciplines with columns for rank, code, name, recommended period, credits, and score.

Disciplinas Obrigatórias					
Ordenadas por grau de relevância, descartando disciplinas com pré-requisitos insuficientes e priorizando disciplinas atrasadas que são pré-requisitos					
#	Código	Nome	Período Recomendado	Creditos	Score
1	MAB515	Avaliação e Desempenho	7º Período	40	

Disciplinas Eletivas					
Ordenadas por grau de relevância, levando em consideração similaridade entre outras disciplinas					
#	Código	Nome		Creditos	Score
1	MAB632	Top Esp em Sist Inteligentes II		40	0.24
2	MAB621	Sistemas Embutidos I		40	0.23
3	MAB003	Top Esp em Segurança da Inform		40	0.21
4	MAB529	Sistemas Complexos Inteligentes I		40	0.20
5	MAB364	Linguagens de Programação		40	0.20
6	MAB616	Interface Humano-Computador		40	0.19
7	MAB470	Sistemas Operacionais II		40	0.18
8	MAB635	Laboratório de Sistemas Digitais		40	0.18

Figura 11: Exemplo de ranqueamento de disciplinas obrigatórias e eletivas.

### 4.2.1 Ranking de disciplinas obrigatórias

Para construir o *ranking* das obrigatórias, são utilizadas regras bem definidas que foram obtidas com a conclusão da etapa de pesquisa com os professores e alunos. A

intuição principal é priorizar disciplinas atrasadas, principalmente as que são pré-requisitos de outras disciplinas, seguindo a previsão do curso, na medida do possível. Na prática, as regras que o sistema segue são: (i) considerar apenas as disciplinas obrigatórias factíveis em termos de pré-requisitos, ordenadas crescentemente por período previsto; (ii) dentro de cada grupo de disciplinas de mesmo período previsto, ordenar por quantidade de disciplinas que têm a disciplina em questão como pré-requisito de maneira decrescente.

Para ilustrar a situação descrita acima, considere o seguinte cenário, ilustrado na Figura 12: restam para um aluno cursar apenas as disciplinas obrigatórias  $A$ ,  $B$ ,  $C$  e  $D$ ; as disciplinas  $A$  e  $B$  são previstas para serem cursadas no 6º período e as disciplinas  $C$  e  $D$  são previstas para o 7º período. A disciplina  $D$  tem como pré-requisito a disciplina  $B$ .

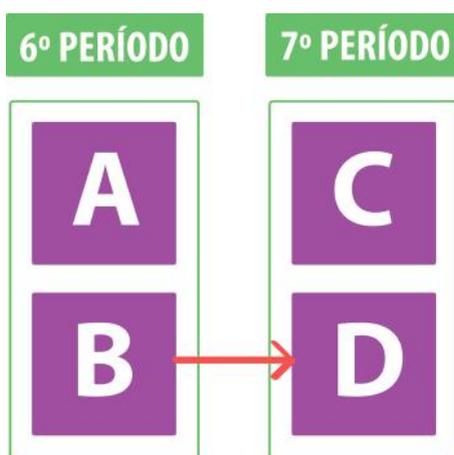


Figura 12: Imagem para ilustrar a situação com disciplinas  $A$  e  $B$  no 6º período e disciplinas  $C$  e  $D$  do 7º. A seta vermelha de  $B$  para  $D$  representa uma situação de pré-requisito.

Neste caso, o *ranking* obtido, conforme apresentado na Figura 13, seria:  $B$ ,  $A$  e  $C$  nesta ordem. A disciplina  $D$  não aparece porque o aluno não pode cursá-la ainda, já que ainda não cursou a disciplina  $B$ ; as disciplinas  $A$  e  $B$  são previstas para um período anterior em relação à  $C$ , portanto aparecem na frente, e a disciplina  $B$  é pré-requisito da disciplina  $D$ , ganhando prioridade sobre a disciplina  $A$ .



Figura 13: Ilustração representando o resultado do ranqueamento. As disciplinas aparecem na ordem  $B - A - C$ , e a disciplina  $D$  não compõe o *ranking* porque tem  $B$  como pré-requisito.

#### 4.2.2 Ranking de disciplinas eletivas

Para construir o *ranking* das eletivas, foram elaborados dois processos que fazem uso de técnicas de Filtragem Colaborativa. A intuição do cálculo deste ranqueamento é tentar identificar quais disciplinas eletivas são similares às disciplinas eletivas que o aluno alvo de recomendação já cursou. A ideia inicial era obter avaliações dos alunos para cada disciplina eletiva cursada, fazendo uso destas avaliações no momento da recomendação. O problema é que esta avaliação poderia ser muito trabalhosa, levando os usuários a rejeitarem o sistema. Então, a solução foi assumir que as disciplinas eletivas cursadas em que o usuário foi aprovado representam bem seu perfil e interesses no curso.

Existe um problema com esta abordagem, e é, por isso, que fez-se necessário um segundo processo de recomendação. O problema é que não é possível recomendar nada para alunos que nunca cursaram nenhuma disciplina eletiva, já que utilizamos este dado para calcular o perfil do aluno. Por este motivo, um segundo processo de recomendação foi desenvolvido, e sua intuição é de traçar o perfil do aluno baseado

em seu desempenho nas disciplinas obrigatórias. Ambos os processos são detalhados nas seções a seguir (Seções 4.2.2.1 e 4.2.2.2).

#### 4.2.2.1 Primeira abordagem para cálculo de *scores* de eletivas

A primeira etapa do algoritmo é a construção de uma matriz esparsa de alunos por disciplinas eletivas, onde 1 na posição  $(i,j)$  significa que o  $i$ -ésimo aluno cursou e foi aprovado na  $j$ -ésima disciplina eletiva e 0 significa que o  $i$ -ésimo aluno não cursou a  $j$ -ésima disciplina e/ou foi reprovado na mesma.

A próxima etapa é o cálculo de uma matriz de similaridades entre as disciplinas utilizando *Cosine Similarity* (similaridade de cosseno) entre cada par de vetores de disciplinas, como pode ser visto na equação 4.1.

$$sim(\vec{x}, \vec{y}) = \frac{\vec{x} \cdot \vec{y}}{\|\vec{x}\| \|\vec{y}\|} = \frac{\sum_{i \in A} X_i Y_i}{\sqrt{\sum_{i \in A} X_i^2} \sqrt{\sum_{i \in A} Y_i^2}} \quad (4.1)$$

Onde:

- $\vec{x}$ : vetor que representa a disciplina  $X$ , equivalente a uma coluna da matriz de alunos por disciplinas;
- $\vec{y}$ : vetor que representa a disciplina  $Y$ , equivalente a uma coluna da matriz de alunos por disciplinas;
- $A$ : conjunto de alunos;
- $X_i$ : valor do  $i$ -ésimo elemento do vetor de  $X$ , correspondente ao  $i$ -ésimo aluno;
- $Y_i$ : valor do  $i$ -ésimo elemento do vetor de  $Y$ , correspondente ao  $i$ -ésimo aluno.

Cabe salientar que a matriz resultante é simétrica uma vez que o *score* de similaridade entre o par de disciplinas  $X-Y$  e  $Y-X$  é o mesmo. Isso configura uma solução de filtragem colaborativa *Item-item*. A escolha de uma abordagem *Item-item* sobre *User-user* deu-se devido à escala do problema: a quantidade de disciplinas por pe-

ríodo é sempre a mesma salvo algumas exceções onde novas disciplinas são criadas, enquanto muitos alunos novos entram no curso a cada período.

Com a matriz de similaridades representada, já é possível calcular as recomendações. Para fazer isto, basta calcular a média aritmética das similaridades entre as disciplinas cursadas pelo aluno e as que ainda não cursou. O resultado deste cálculo será um vetor de *scores* indexado pelas disciplinas que o usuário ainda não cursou. Ao final do cálculo, basta remover os índices com valor 0 deste vetor, já que não serão recomendados, e realizar uma ordenação decrescente nos demais elementos do vetor gerando o ranqueamento final.

A seguir, será desenvolvido um exemplo para fins de esclarecimento. Considere então cinco alunos:  $A_1$ ,  $A_2$ ,  $A_3$ ,  $A_4$  e  $A_5$  e também 5 disciplinas eletivas:  $D_1$ ,  $D_2$ ,  $D_3$ ,  $D_4$  e  $D_5$ . O aluno  $A_1$  cursou e aprovou as disciplinas  $D_1$  e  $D_2$ ; o aluno  $A_2$  cursou e aprovou as disciplinas  $D_3$ ,  $D_4$  e  $D_5$ ; o aluno  $A_3$  cursou e aprovou apenas  $D_2$ ; o aluno  $A_4$  cursou e aprovou as disciplinas  $D_3$  e  $D_5$ ; e, por fim, o aluno  $A_5$  cursou e aprovou as disciplinas  $D_1$ ,  $D_2$  e  $D_3$ .

Nesse exemplo, a matriz alunos por disciplinas obtida seria:

$$\begin{array}{c}
 D_1 \quad D_2 \quad D_3 \quad D_4 \quad D_5 \\
 A_1 \left[ \begin{array}{ccccc} 1 & 1 & 0 & 0 & 0 \end{array} \right] \\
 A_2 \left[ \begin{array}{ccccc} 0 & 0 & 1 & 1 & 1 \end{array} \right] \\
 A_3 \left[ \begin{array}{ccccc} 0 & 1 & 0 & 0 & 0 \end{array} \right] \\
 A_4 \left[ \begin{array}{ccccc} 0 & 0 & 1 & 0 & 1 \end{array} \right] \\
 A_5 \left[ \begin{array}{ccccc} 1 & 1 & 1 & 0 & 0 \end{array} \right]
 \end{array}$$

A matriz de similaridades do exemplo proposto, sendo calculado o *Cosine Similarity* entre disciplinas, seria:

$$\begin{array}{c}
 D_1 \quad D_2 \quad D_3 \quad D_4 \quad D_5 \\
 \left[ \begin{array}{ccccc}
 D_1 & 1 & 0,81 & 0,4 & 0 & 0 \\
 D_2 & 0,81 & 1 & 0,33 & 0 & 0 \\
 D_3 & 0,4 & 0,33 & 1 & 0,57 & 0,81 \\
 D_4 & 0 & 0 & 0,57 & 1 & 0,7 \\
 D_5 & 0 & 0 & 0,81 & 0,7 & 1
 \end{array} \right]
 \end{array}$$

Para exemplificar o cálculo do *score* final das recomendações, vamos calcular e determinar as recomendações para os alunos  $A_3$  e  $A_4$ . O aluno  $A_3$  cursou apenas  $D_2$ , então serão calculados *scores* para as disciplinas  $D_1$ ,  $D_3$ ,  $D_4$  e  $D_5$ . Como este aluno só cursou uma disciplina, o vetor de *scores* resultante será a média de apenas um elemento, que é o vetor de similaridades entre a disciplina  $D_2$  e as demais, ilustrado abaixo:

$$\mathbf{A}_3 = \begin{array}{c}
 D_1 \quad D_2 \quad D_3 \quad D_4 \quad D_5 \\
 \left[ \begin{array}{ccccc}
 0,81 & X & 0,33 & 0 & 0
 \end{array} \right]
 \end{array}$$

O aluno  $A_4$  cursou duas disciplinas,  $D_3$  e  $D_5$ . Para calcular seu vetor de *scores* é calculada a média, posição a posição, das similaridades entre  $D_3$  e as demais disciplinas e  $D_5$  e as demais disciplinas:

- Para a primeira posição, referente à disciplina  $D_1$ , temos a média entre 0,4 (similaridade entre  $D_1$  e  $D_3$ ) e 0 (similaridade entre  $D_1$  e  $D_5$ ) que resulta em um *score* de 0,2.
- Para a segunda posição, referente à disciplina  $D_2$ , temos a média entre 0,33 (similaridade entre  $D_2$  e  $D_3$ ) e 0 (similaridade entre  $D_2$  e  $D_5$ ), resultando em um *score* de 0,16.
- As terceira e quinta posições não são calculadas, pois se referem às disciplinas  $D_3$  e  $D_5$ , que já foram cursadas pelo aluno.

- Para a quarta posição, referente à disciplina  $D_4$ , temos a média entre 0,37 (similaridade entre  $D_4$  e  $D_3$ ) e 0,70 (similaridade entre  $D_4$  e  $D_5$ ), resultando em um *score* de 0,63.

O vetor resultante está ilustrado abaixo:

$$\mathbf{A}_4 = \begin{matrix} & D_1 & D_2 & D_3 & D_4 & D_5 \\ \begin{bmatrix} 0,2 & 0,16 & X & 0,63 & X \end{bmatrix} \end{matrix}$$

O *ranking* de disciplinas recomendadas para cada aluno será exemplificado a seguir. Para o aluno  $A_3$ , o *ranking* final seria:  $D_1 - D_3$ . Para o aluno  $A_4$ , o *ranking* final seria:  $D_4 - D_1 - D_2$ .

#### 4.2.2.2 Segunda abordagem para cálculo de *scores* das eletivas

Para entender a necessidade desta segunda abordagem, basta imaginar um novo aluno  $A_6$  no exemplo proposto na seção 4.2.2.1. Se este aluno não tiver cursado nenhuma disciplina eletiva ainda, o terceiro passo do cálculo proposto, que é o passo que consiste em calcular a média entre as similaridades das disciplinas que o aluno já foi aprovado e as demais, não seria possível de realizar, impossibilitando recomendação neste caso.

Por conta disso, esta segunda abordagem leva em consideração as notas dos alunos nas disciplinas obrigatórias e eletivas, propondo que existe uma correlação entre o perfil e gostos do aluno por disciplinas eletivas e seu desempenho nas disciplinas ao longo do curso. Para exemplificar esta intuição, podemos pensar em um aluno com bom desempenho em disciplinas da área de análise de dados, mas mediano nas demais. Provavelmente, este aluno vai gostar de disciplinas eletivas relacionadas com este tema no futuro.

Para calcular cada elemento da matriz de similaridades, é usado o Coeficiente de Pearson, apresentado na equação 4.2, uma versão ajustada do *Cosine Similarity* (descrito anteriormente na equação 4.1). O Coeficiente de Pearson é responsável por

ênfatizar as tendências em vez de simplesmente a igualdade de valores absolutos. Em outras palavras, apesar de alguns alunos manterem a média alta e outros manterem a média baixa, os padrões podem ser similares e este coeficiente é capaz de endereçar estes casos.

$$sim(\vec{x}, \vec{y}) = \frac{\sum_{i \in A} (n_{i,x} - \bar{n}_i)(n_{i,y} - \bar{n}_i)}{\sqrt{\sum_{i \in A} (n_{i,x} - \bar{n}_i)^2} \sqrt{\sum_{i \in A} (n_{i,y} - \bar{n}_i)^2}} \quad (4.2)$$

Onde:

- $sim(\vec{x}, \vec{y})$ : similaridade entre vetores de representação das disciplinas  $X$  e  $Y$ ;
- $x$ : disciplina  $X$ ;
- $y$ : disciplina  $Y$ ;
- $A$ : conjunto de alunos;
- $i$ :  $i$ -ésimo aluno;
- $n_{i,x}$ : nota obtida pelo  $i$ -ésimo aluno na disciplina  $X$ ;
- $n_{i,y}$ : nota obtida pelo  $i$ -ésimo aluno na disciplina  $Y$ ;
- $\bar{n}_i$ : média das notas do  $i$ -ésimo aluno.

Neste momento, após obter a matriz de similaridades, é possível fazer o cálculo de *score* das disciplinas eletivas não cursadas pelos alunos. Para calcular o *score* de uma determinada disciplina para um aluno, basta calcular uma média ponderada das notas deste mesmo aluno em todas as disciplinas cursadas por ele, com pesos iguais às similaridades entre estas disciplinas e a disciplina em questão. São desconsiderados os valores negativos de similaridade. Este cálculo é descrito na equação 4.3.

$$score(a, d) = \frac{\sum_{i \in D_a} sim(i, d) * n_{a,i}}{\sum_{i \in D_a} sim(i, d)} \quad (4.3)$$

Onde:

- $a$ : aluno  $A$ ;
- $d$ : disciplina  $D$ ;
- $D_a$ : conjunto de disciplinas cursadas pelo aluno  $A$ ;
- $n_{a,i}$ : nota obtida pelo aluno  $a$  na  $i$ -ésima disciplina;
- $sim(i, d)$ : similaridade entre a  $i$ -ésima disciplina e a disciplina  $d$ .

Por fim, o resultado será uma matriz que corresponde aos vetores de *scores* de disciplinas dos alunos. Então, bastará ordenar os elementos dos vetores para se obter o ranqueamento desejado.

Para explicar os cálculos, neste caso, um novo exemplo será proposto. Considere a matriz abaixo, contemplando os alunos  $A_1, A_2, A_3, A_4$  e  $A_5$  e suas notas nas disciplinas obrigatórias  $O_1, O_2, O_3$  e  $O_4$  e disciplinas eletivas  $E_1$  e  $E_2$ :

	$O_1$	$O_2$	$O_3$	$O_4$	$E_1$	$E_2$
$A_1$	10	10	7	7	10	6
$A_2$	5	5	6	6	5	6
$A_3$	7	7	5	5	0	0
$A_4$	6	7	10	9	0	0
$A_5$	5	6	9	8	6	9

Cada célula  $(i, j)$  representa a nota em que o  $i$ -ésimo aluno obteve na  $j$ -ésima disciplina e as células preenchidas com 0 representam as disciplinas que o aluno ainda não foi aprovado.

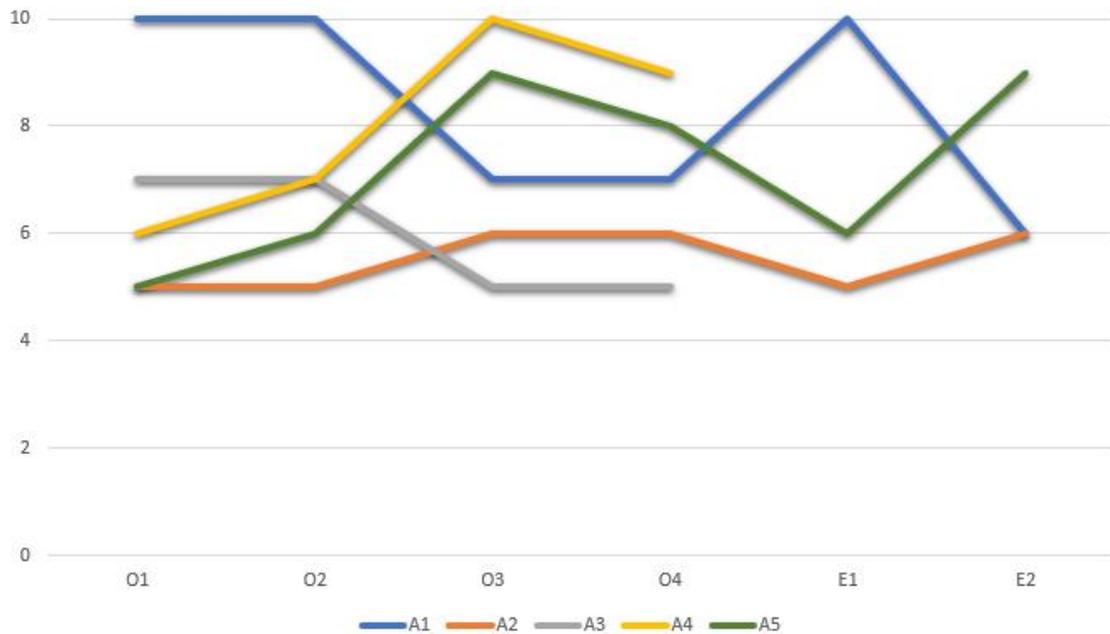


Figura 14: Visualização gráfica do exemplo proposto.

A Figura 14 mostra as notas dos alunos pelas disciplinas. É fácil perceber que as curvas dos alunos  $A_2$ ,  $A_4$  e  $A_5$  têm formato parecido, apesar da distância entre os pontos individualmente e o mesmo efeito acontece para os alunos  $A_1$  e  $A_3$ . Este efeito é o comportamento que o coeficiente de Pearson procura identificar, então, apesar de  $A_2$  e  $A_3$  terem notas parecidas (em termos numéricos absolutos), suas curvas se comportam de maneiras diferentes, indicando que eles não são similares por esse coeficiente.

A seguir, é calculada a matriz de similaridades, conforme explicado anteriormente, nas equações 4.1 e 4.2, e é obtida a matriz abaixo como resultado do exemplo:

$$\begin{array}{c}
O_1 \\
O_2 \\
O_3 \\
O_4 \\
E_1 \\
E_2
\end{array}
\begin{bmatrix}
O_1 & O_2 & O_3 & O_4 & E_1 & E_2 \\
1 & 0.94 & -0.99 & -0.93 & 0.95 & -0.96 \\
0.94 & 1 & -0.93 & -0.99 & 1 & -0.99 \\
-0.99 & -0.93 & 1 & 0.93 & -0.94 & 0.96 \\
-0.93 & -0.99 & 0.93 & 1 & -0.99 & 0.98 \\
0.95 & 1 & -0.94 & -0.99 & 1 & -0.99 \\
-0.96 & -0.99 & 0.96 & 0.98 & -0.99 & 1
\end{bmatrix}$$

Neste momento, é possível calcular os *scores* das disciplinas, segundo a equação 4.3, e é gerada a matriz dos *scores* das disciplinas  $E_1$  e  $E_2$  para os alunos  $A_3$  e  $A_4$  a seguir:

$$\begin{array}{c}
E_1 \quad E_2 \\
A_3 \begin{bmatrix} 0.70 & 0.60 \end{bmatrix} \\
A_4 \begin{bmatrix} 0.65 & 0.84 \end{bmatrix}
\end{array}$$

Ao final, basta ordenar de maneira decrescente os vetores de *scores* de disciplinas do aluno para obter o ranqueamento. No exemplo proposto, o aluno  $A_3$  receberia o ranqueamento:  $E_1 - E_2$ , enquanto o aluno  $A_4$  receberia o ranqueamento  $E_2 - E_1$ .

### 4.3 SUGESTÃO DE GRADE

Na funcionalidade de Recomendação de Grade, é possível realizar diversas configurações e receber uma grade pronta como recomendação. Essas possibilidades de configurações foram obtidas analisando as pesquisas realizadas com alunos e professores, que julgaram quais ajustes seriam relevantes para o momento da escolha das disciplinas para o próximo período. Essa interação com a ferramenta para configurar um cenário favorável para o aluno configura uma abordagem baseada em conhecimento (2.1.3), que eu conjunto com as técnicas de filtro colaborativo para cálculo de ranqueamento individual para cada aluno forma uma solução híbrida, como descrito na seção 2.1.4.

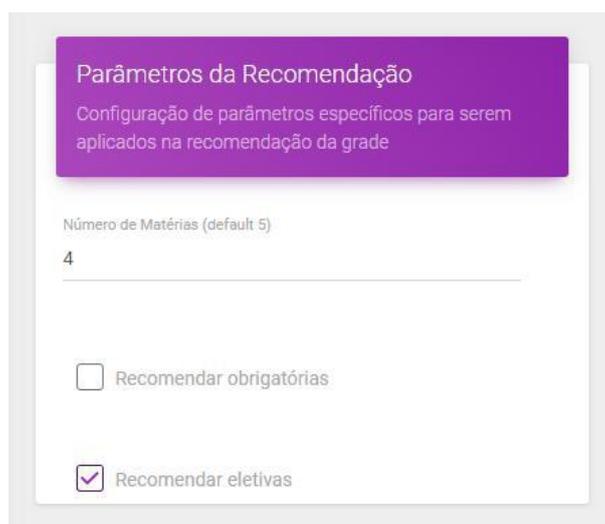
Primeiramente, é mostrada a lista contendo as disciplinas que serão ofertadas no próximo período letivo, exibindo apenas as disciplinas que o aluno ainda não cursou e pode cursar em termos de pré-requisitos, como é ilustrado na Figura 15. Estas disciplinas estão ordenadas usando o mesmo método de ranqueamento anterior, descrito na seção 4.2. Nesta lista, é possível fixar ou ignorar disciplinas, garantindo que as mesmas apareçam ou não no resultado da grade recomendada, independente do *ranking* calculado.



Fixar	Ignorar	Código	Nome	Professor	Horários	Período Recomendado
<input type="checkbox"/>	<input checked="" type="checkbox"/>	MAB515	Avaliação e Desempenho	PAULO HENRIQUE DE AGUIAR RODRIGUES	SEG 10:00-12:00, QUA 10:00-12:00	7º Período
<input type="checkbox"/>	<input checked="" type="checkbox"/>	MAB008	Tópicos Especiais em Segurança	CLAUDIO MICELI DE FARIAS	TER 15:00-17:00, QUI 15:00-17:00	Eletiva
<input type="checkbox"/>	<input checked="" type="checkbox"/>	MAB364	Linguagens de Programação	FABIO MASCARENHAS DE QUEIROZ	SEG 08:00-10:00, QUA 08:00-10:00	Eletiva
<input type="checkbox"/>	<input checked="" type="checkbox"/>	MAB635	Laboratório de Sistemas Digitais	NELSON QUILULÁ VASCONCELOS	TER 13:00-15:00, QUI 13:00-15:00	Eletiva
<input type="checkbox"/>	<input checked="" type="checkbox"/>	MAB602	Data Wareh Sup Tom de Decisão	MARIA LUIZA MACHADO CAMPOS	SEX 08:00-10:00, SEX 10:00-12:00	Eletiva
<input type="checkbox"/>	<input checked="" type="checkbox"/>	MAB519	Informática Aplicada ao Ensino	CARLA AMOR DIVINO MOREIRA DELGADO	SEG 13:00-15:00, SEG 15:00-17:00	Eletiva
<input type="checkbox"/>	<input checked="" type="checkbox"/>	MAB478	Métodos Numéricos	MAURO ANTONIO RINCON	TER 08:00-10:00, QUI 08:00-10:00	Eletiva
<input type="checkbox"/>	<input checked="" type="checkbox"/>	MAB125	Projeto de Jogos	GERALDO BONORINO XEXEO	SEG 13:00-15:00, SEG 15:00-17:00	Eletiva

Figura 15: Lista de disciplinas ofertadas em 2017.2.

Em seguida, o sistema permite que o usuário escolha o número máximo de disciplinas que serão recomendadas na grade, assim como a possibilidade de recomendar ou não disciplinas obrigatórias e eletivas (Figura 16). Ao lado existe uma grade horária vazia onde é possível filtrar horários que o usuário não deseja receber alguma recomendação (Figura 17). Esta configuração representa os horários que as disciplinas do BCC são oferecidas: pela manhã, de 08:00 às 10:00 e de 10:00 às 12:00; e pela tarde, de 13:00 às 15:00 e de 15:00 às 17:00.



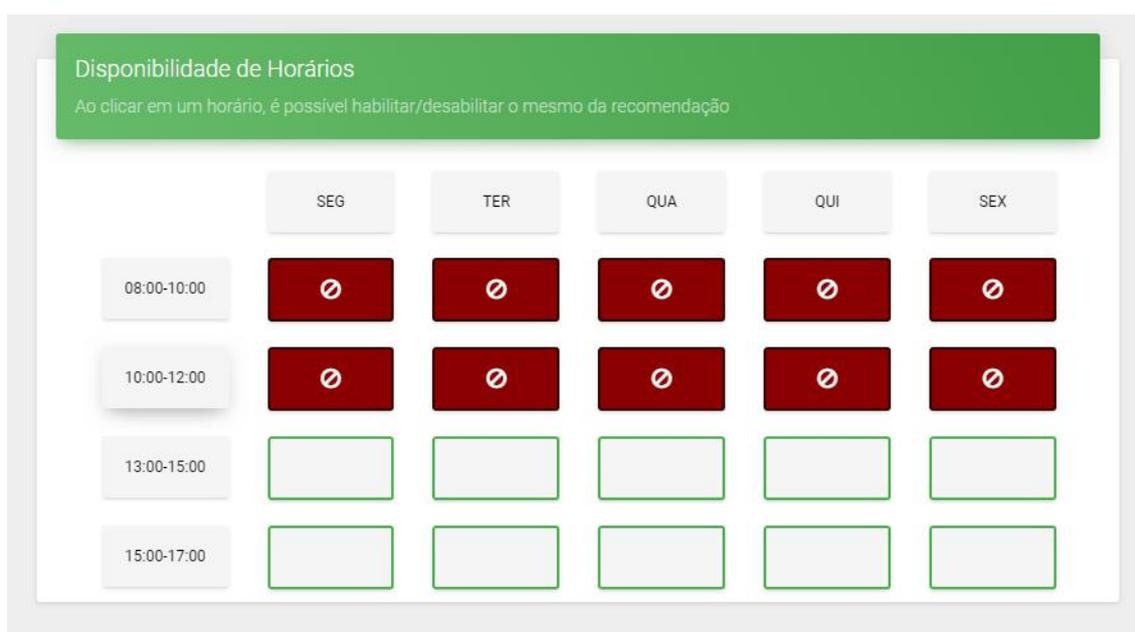
**Parâmetros da Recomendação**  
Configuração de parâmetros específicos para serem aplicados na recomendação da grade

Número de Matérias (default 5)  
4

Recomendar obrigatórias

Recomendar eletivas

Figura 16: Exemplo de parâmetros de recomendação com filtro de quantidade máxima de 4 disciplinas e apenas eletivas.



**Disponibilidade de Horários**  
Ao clicar em um horário, é possível habilitar/desabilitar o mesmo da recomendação

	SEG	TER	QUA	QUI	SEX
08:00-10:00	<input type="checkbox"/>				
10:00-12:00	<input type="checkbox"/>				
13:00-15:00	<input type="checkbox"/>				
15:00-17:00	<input type="checkbox"/>				

Figura 17: Exemplo de filtro bloqueando horários da manhã para receber disciplinas apenas no período de 13:00 às 17:00 todos os dias.

Por fim, a página exibe um botão para calcular a grade e, logo abaixo, é apresentado o resultado da recomendação, como mostra a Figura 18.

	SEG	TER	QUA	QUI	SEX
08:00-10:00					
10:00-12:00					
13:00-15:00	MAB519	MAB635	MAB001	MAB635	
15:00-17:00	MAB519	MAB003	MAB001	MAB003	

Figura 18: Exemplo de grade recomendada utilizando os parâmetros das Figuras 16 e 17.

O algoritmo para calcular a grade recomendada pode ser dividido em três etapas principais: filtragem, ranqueamento e montagem. Tais etapas serão explicitadas a seguir.

Na etapa de filtragem, são removidas da lista de disciplinas a serem recomendadas, todas as que não possuem pré-requisitos suficientes, bem como as disciplinas em horários que foram bloqueados pelo usuário durante a parametrização da recomendação. Também são removidas as disciplinas que foram selecionadas como “ignoradas” da tabela de ofertadas.

Depois disso, na etapa de ranqueamento, são realizados os cálculos de *scores* das disciplinas obrigatórias e eletivas, conforme descritos na Seção 4.2 seguindo a abordagem de filtro colaborativo detalhada na Seção 2.1.1, apenas para o resultado da filtragem na etapa anterior. Dessa forma, são geradas duas listas ordenadas de disciplinas: uma para obrigatórias e outra para eletivas.

Por fim, na etapa da montagem, percorre-se o ranqueamento de disciplinas obrigatórias, inserindo uma a uma na grade recomendada. Após o término da lista de disciplinas obrigatórias, é percorrido o ranqueamento de eletivas repetindo o mesmo processo de inserção. Cada inserção reserva um horário na grade recomendada, fa-

zendo com que uma próxima disciplina no mesmo horário não entre novamente na grade. Esta etapa encerra-se quando o limite de disciplinas configurado pelo usuário for atingido ou ambos os ranqueamentos forem esgotados e não for possível inserir mais nenhum novo elemento na grade recomendada. Se o número total de disciplinas a serem recomendados não for configurado previamente, o sistema adota valor padrão 5, que representa a média de disciplinas cursadas por período, obtida durante a pesquisa descrita previamente na Seção 3.2.

## 5 IMPLEMENTAÇÃO DO SISTEMA

Neste capítulo, serão descritas as tecnologias utilizadas na implementação do sistema, assim como toda a análise, processamento e persistência dos dados obtidos inicialmente.

### 5.1 TECNOLOGIAS PRINCIPAIS

Para a implementação do sistema, a principal linguagem utilizada foi JavaScript, tanto no *back-end* quanto no *front-end*. Para armazenamento dos dados, foi utilizado o banco de dados NoSQL MongoDB<sup>1</sup>, que é orientado a documentos. Na parte do *front-end*, o *framework* Angular<sup>2</sup>, baseado em JavaScript, foi utilizado. Já no *back-end* foi utilizado o ambiente de execução Node.js<sup>3</sup>, também baseado em JavaScript. Por último, o *framework* Express<sup>4</sup> foi utilizado para implementação do servidor web. Essas quatro tecnologias em conjunto, formam o chamado *MEAN stack*<sup>5</sup>, um *framework* que foi criado pensando em utilizar o JavaScript como linguagem única em todas as partes de um projeto.

#### 5.1.1 Aplicação WEB (*Client-side*)

O Angular é um *Web Framework* Javascript liderado pela equipe de Angular da Google, com o intuito de ajudar no desenvolvimento do *front-end* das aplicações, ou seja, a parte que é executada no navegador do usuário. Inicialmente foi adotado o padrão MVC (*Model-view-controller*), pois na época do lançamento de sua primeira *release*, era comum nas aplicações a utilização de jQuery<sup>6</sup>, ou outros recursos que não tinham uma padronização na organização dos arquivos ou na maneira como era desenvolvido o *front-end*. Depois de alguns anos, a equipe da Google reescreveu

---

<sup>1</sup><https://www.mongodb.com/>

<sup>2</sup><https://angular.io/>

<sup>3</sup><https://nodejs.org/en/>

<sup>4</sup><http://expressjs.com/pt-br/>

<sup>5</sup><http://mean.io/>

<sup>6</sup><https://jquery.com/>

todo o código do então chamado AngularJS, para criar o novo Angular, que aliado ao Typescript, que é uma linguagem tipada com programação orientada a objetos desenvolvida pela Microsoft, trouxe muitos benefícios e novos desenvolvedores ativos. Com essa nova versão, trouxeram o conceito de hierarquia de componentes, ao invés do conceito de escopo ou controladores. Muitas funcionalidades básicas passaram a compor os módulos, possui carregamento dinâmico, um modelo de compilação assíncrono e programação reativa de suporte usando o RxJS<sup>7</sup>.

Com isso, o Angular foi o *framework* escolhido para o desenvolvimento do *front-end* do projeto, não só devido à prévia experiência do assunto por parte dos integrantes, mas também pelo crescimento atual do mesmo.

A estrutura final consistiu em um componente pai, chamado *App*, sucedido dos seus filhos: *Shell*, utilizado para estruturar a “casca” da aplicação com o *header*, *sidebar*, *footer*; *Notifications*, responsável pela visualização das notificações das requisições; e os componentes *UserStatistics*, *Ranking* e *Grid*, que representam as funcionalidades do projeto, respectivamente, a página de estatísticas do usuário, a página de ranqueamento das disciplinas e a página de recomendação da grade.

Foi utilizado o *template* Material Dashboard Angular<sup>8</sup>, desenvolvido por Creative Tim<sup>9</sup> com suporte para os componentes do Angular e utilizando *Material Design*. Com ele foi possível montar as páginas e criar os elementos visuais para mostrar os dados, como gráficos, tabelas, etc., através de bons exemplos e uma documentação de fácil entendimento.

### 5.1.2 Servidor (*Server-side*)

Para a aplicação do *back-end* foi implementada uma API RESTful em Node.js, onde a aplicação Web se comunica através de requisições HTTP com o servidor. O Node.js, diferentemente de servidores Web convencionais, executa uma única *thread* esperando requisições e uma vez uma conexão aberta, ela recebe uma função de

---

<sup>7</sup><https://rxjs-dev.firebaseio.com/>

<sup>8</sup><https://www.creative-tim.com/product/material-dashboard-angular2>

<sup>9</sup><https://www.creative-tim.com/>

*callback* que será disparada assim que a requisição terminar, deixando a *thread* não bloqueada enquanto executa operações de I/O (entrada e saída).

A estrutura do projeto *back-end* foi separada em 3 camadas principais: *Controllers*, *Services* e *Models* (Apêndice E). Os modelos refletem os dados que estão nas coleções do MongoDB. Para isso foi utilizado o Mongoose<sup>10</sup>, que abstrai o que é necessário para interagir com a instância do banco. Os serviços, por sua vez, utilizam os *Schemas* criados no Mongoose para fazer a interação com o banco de dados e também contém toda a lógica de negócio da aplicação. Por fim, os controladores ficam com a camada HTTP, tratando as requisições da aplicação Web à API, sem conhecer nenhuma lógica de negócios, repassando o que é requisitado para os serviços correspondentes e retornando o dado devidamente tratado.

### 5.1.3 Banco de Dados

Como mencionado anteriormente, foi utilizado o banco de dados não relacional MongoDB, pois a forma em que os dados obtidos estavam estruturados facilitou a utilização de um banco de dados NoSQL. O MongoDB é um software de código aberto disponível para diversas plataformas, facilmente escalável, sua configuração é rápida e para aplicações mais simples e de baixo custo ele é ideal. A base de dados do MongoDB é organizada em coleções, que por sua vez possuem documentos, que são os dados estruturados em pares de chave-valor bem parecidos com objetos em JavaScript. Comparando com um banco de dados relacional, as coleções do MongoDB equivalem-se às tabelas, e os documentos às linhas. Assim como no banco de dados relacional, também é possível a criação de índices para melhorar o desempenho das consultas. Para facilitar o acesso e manutenção da base de dados, foi utilizado o serviço de nuvem MongoDB Atlas<sup>11</sup> que é disponibilizado pela própria empresa, também gratuitamente, para uma configuração básica de uma máquina na nuvem.

---

<sup>10</sup><http://mongoosejs.com>

<sup>11</sup><https://www.mongodb.com/cloud/atlas>

## 5.2 TECNOLOGIAS AUXILIARES

Todas as etapas e tarefas, durante o processo de pesquisa e desenvolvimento do projeto, foram elaboradas e organizadas em quadros na plataforma Trello<sup>12</sup>. Nela foi possível seguir a técnica Kanban, que consiste em dividir as etapas do processo de acordo com o progresso de cada tarefa, para se obter uma melhor visualização das mesmas. Com isso, foram criadas colunas que representavam o estado em que cada tarefa se encontrava, são elas: *To do*, *Doing*, *Done*. Assim, foi possível ter um controle melhor de todo o desenvolvimento, assim como a divisão entre os membros do grupo.

O projeto foi criado em um repositório com controle de versionamento Git, na plataforma BitBucket<sup>13</sup>. Com isso, foi possível ter todo o controle de fluxo do processo de desenvolvimento, onde todos puderam trabalhar em suas tarefas, resolvendo facilmente os conflitos, caso surgissem, durante a geração dos códigos.

## 5.3 OBTENÇÃO E ANÁLISE INICIAL DOS DADOS

Os dados foram disponibilizados pelo SIGA, através de um servidor *ElasticSearch*<sup>14</sup>, onde foi possível fazer a extração das informações a partir do período de 2010/1 até 2017/1. O ElasticSearch é um servidor de buscas distribuído baseado no Apache Lucene<sup>15</sup>, que possui uma API de indexação de documentos armazenados em uma estrutura NoSQL para buscas em grandes quantidades de dados de maneira eficiente.

Os dados obtidos foram estruturados em formato JSON, como mostra o código 5.1, de acordo com a solicitação das informações que seriam relevantes para o projeto. Nele, é possível perceber que todos os dados estão agrupados em uma única estrutura e alguns deles podem possuir mais de um valor fixo, como, por exemplo, a situação do aluno, ou o horário da disciplina. Com isso, fez-se necessário rea-

---

<sup>12</sup><https://trello.com/>

<sup>13</sup><https://bitbucket.org/>

<sup>14</sup> <https://www.elastic.co>

<sup>15</sup> <https://lucene.apache.org/>

lizar uma análise inicial dos dados para compreender toda a base de informações disponibilizada e realizar alguns pré-processamentos para auxiliar nos cálculos das estatísticas e recomendação.

Código 5.1: Estrutura dos registros no Elasticsearch em formato JSON

```
{
  "doCurriculo": 1,
  "matriculaDRE": "112999999",
  "codigo": "MABXXX",
  "periodoRecomendado": 2,
  "creditos": 40,
  "CH": 40,
  "tipo": "Obrigatoria",
  "nome": "Nome da Disciplina",
  "codigoTurma": 9999,
  "horarios": "['8-10;3','8-10;5']",
  "Docente": "Nome do Professor",
  "periodo": "201X/1",
  "conceito": 100,
  "situacao": "AP"
}
```

A análise inicial dos dados foi feita utilizando o Kibana<sup>16</sup>, que é um plugin de visualização dos dados do Elasticsearch. Nele, é possível realizar consultas nos documentos indexados, assim como construir gráficos e tabelas de diversos tipos para obter uma melhor visualização do conteúdo.

Com ele, foi possível analisar os dados obtidos, investigando possíveis ruídos, como campos vazios ou períodos fora do padrão da graduação (20XX/0 ou 20XX/3), que representam casos de alunos que tiveram o grau de alguma disciplina lançado no sistema SIGA após o período regular. Também foi possível entender melhor os campos `conceito` e `situacao`, para cada caso existente, principalmente para os alunos vindos de transferência para o curso de Bacharelado em Ciência da Computação.

Foi possível verificar que os registros que possuíam valor para `matriculaDRE`,

---

<sup>16</sup> <https://www.elastic.co/products/kibana>

Docente e codigoTurma tinham o restante das informações preenchidas, por conta disso, esses foram os considerados válidos para entender melhor os dados. A partir deste momento, foi possível obter algumas sumarizações dos dados, como por exemplo: (i) a quantidade de registros válidos por período (Figura 19); (ii) a quantidade de eletivas que foram realizadas em cada período (Figura 20).

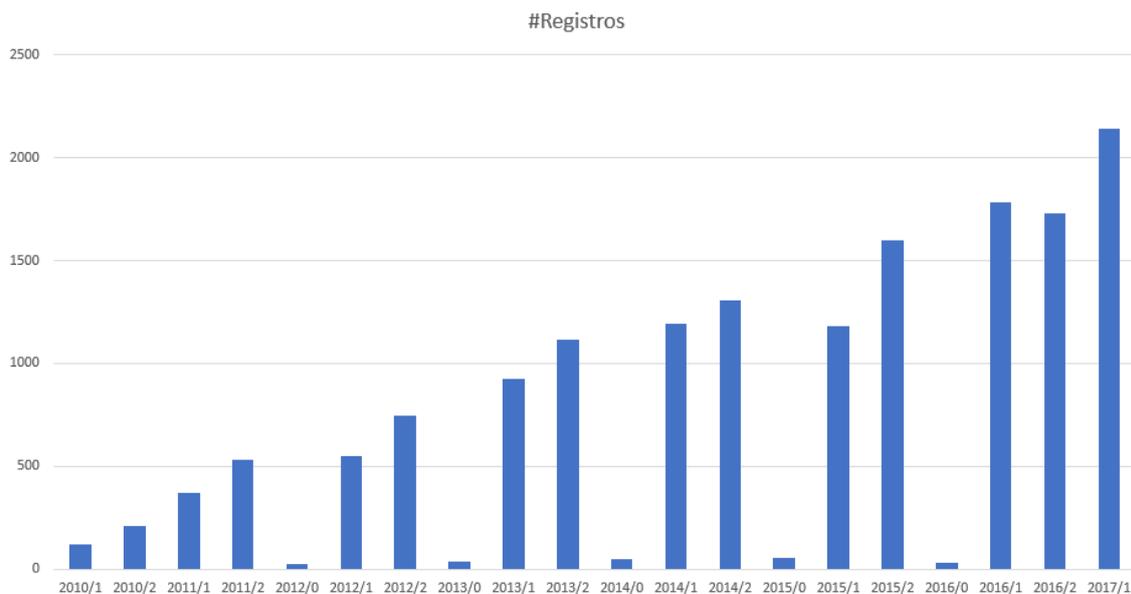


Figura 19: Número de registros por período.

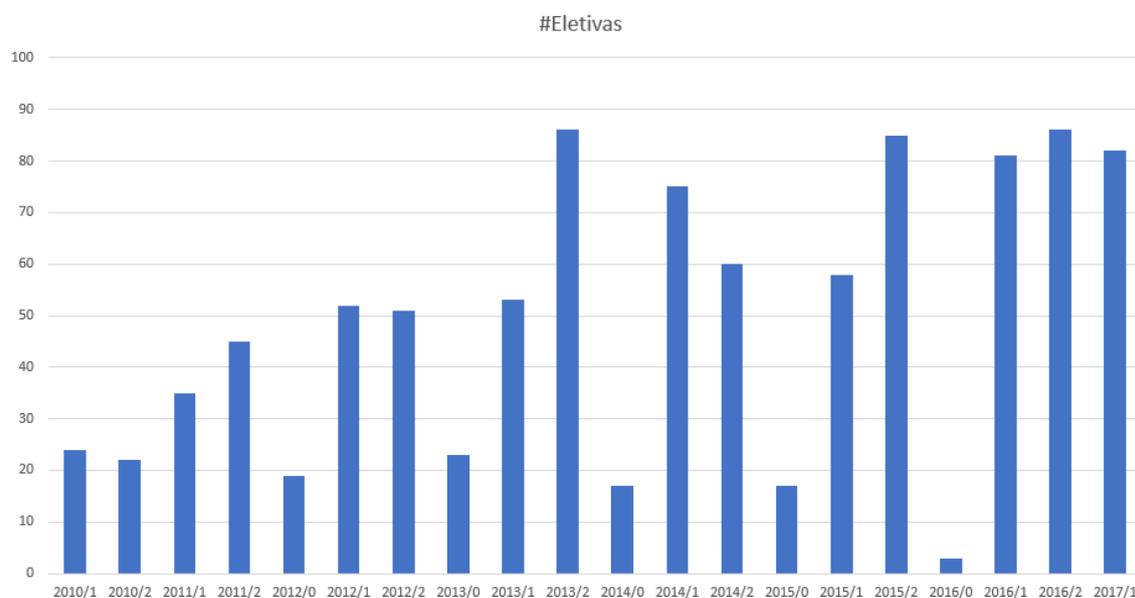


Figura 20: Número de eletivas cursadas por período.

Esse número de eletivas representa a quantidade de disciplinas distintas que foram cursadas por período, ou seja, consideram também disciplinas de livre escolha (de fora da grade curricular do BCC) e não necessariamente contemplam todas as disciplinas do curso, pois pode ocorrer de, em algum período, uma determinada disciplina não ter sido oferecida ou não ter nenhuma inscrição na mesma.

Após o fim da análise inicial das informações obtidas, deu-se início à etapa de desenvolvimento dos códigos de pré-processamento dos dados, para então persisti-los no banco. Esta etapa será detalhada a seguir, na Seção 5.4.

#### 5.4 TRATAMENTO E PERSISTÊNCIA DOS DADOS

Os dados foram armazenados no MongoDB, em 5 coleções, com o intuito de melhor estruturá-los, como ilustrado na Figura 21.

Collection Name ^	Documents
alunos	523
disciplinas	437
docentes	608
ofertas	70
turmas	16,551

Figura 21: Coleções do MongoDB criadas.

A coleção `alunos` possui, para cada documento, o DRE do aluno e uma lista com as disciplinas obrigatórias que ainda faltam para este aluno realizar.

A coleção `disciplinas`, contém os seguintes campos para cada documento:

- `cargaHoraria` - Carga horária da disciplina
- `nome` - Nome da disciplina
- `previstoCurriculo` - Booleano para verificar se a disciplina é do BCC
- `tipo` - Indica se a disciplina é obrigatória ou eletiva
- `codigo` - Código da disciplina
- `periodoRecomendado` - Para qual período aquela disciplina é recomendada. Se for eletiva, seu valor é 0
- `creditos` - Quantos créditos aquela disciplina possui
- `equivalencias` - Lista com os códigos de disciplinas que tem equivalência com a disciplina em questão
- `preReqs` - Lista com os códigos das disciplinas que são pré-requisitos da disciplina em questão

A coleção `docentes` possui apenas o campo nome para cada registro de professor encontrado nos dados obtidos.

A coleção `ofertas` foi populada com disciplinas que estavam sendo ofertadas no período de 2017/2 e contém os seguintes campos:

- `nome` - Nome da disciplina
- `professor` - O professor que irá ministrar a disciplina no período
- `horarios` - Os horários em que a disciplina está sendo ofertada
- `periodoOfertado` - O período em que a disciplina está sendo ofertada (no caso 2017/2)
- `codigo` - Código da disciplina
- `periodoDisciplina` - Período a que a disciplina pertence

A coleção `turmas` contém de fato os registros de cada aluno, em cada disciplina que cursou e o desempenho por ele obtido. Cada documento possui os seguintes campos:

- `situacao` - Se o aluno foi aprovado ou reprovado na disciplina
- `conceito` - A nota que o aluno obteve na disciplina
- `periodo` - O período em que o aluno cursou a disciplina
- `nomeDocente` - O professor que ministrou a disciplina
- `dre` - Registro de matrícula do aluno
- `horarios` - Horário em que a disciplina foi ofertada
- `codigoTurma` - Código da turma
- `codigoDisciplina` - Código da disciplina

Alguns dos campos que foram descritos acima, não foram utilizados no sistema final, mas estão estruturados com a finalidade de prover novas funcionalidades e cálculos em trabalhos futuros.

A próxima etapa do processo consistiu em criar *scripts* em Python<sup>17</sup> (Apêndice D) para persistir os dados do Elasticsearch no MongoDB e também para realizar alguns pré-processamentos, a fim de estruturá-los de uma forma melhor, para serem utilizados posteriormente. Foi utilizada a linguagem Python para implementação dos *scripts* por conta da familiaridade dos integrantes do grupo com a linguagem, e também pela praticidade e facilidade que a linguagem oferece para esse tipo de implementação.

Primeiramente, foi desenvolvido um *script* responsável por persistir todos os dados no banco. Esse *script* possui quatro estruturas de dicionário utilizadas para guardar as informações de docentes, alunos, disciplinas e turmas que são armazenadas nas coleções `docentes`, `alunos`, `disciplinas` e `turmas`, respectivamente. A partir daí, criou-se um *loop* pelos registros, adicionando as informações nos dicionários correspondentes e, por fim, salvando os dados no banco.

Em seguida, foi gerado um *script* para atualizar a coleção `disciplinas` com as disciplinas equivalentes e um *script* para calcular os pré-requisitos de cada disciplina. Ambos utilizam a página da Grade Curricular, presente no SIGA, para obter os dados correspondentes.

Como nem todos os dados obtidos sobre as disciplinas eletivas representam de fato a quantidade de disciplinas existentes no BCC, foi criado um *script* utilizando a lista de eletivas presentes no BOA, para popular o banco com as eletivas faltantes.

Para facilitar o cálculo das disciplinas obrigatórias faltantes para cada aluno, foi criado um *script* para pré-processar esta informação a partir da lista completa de disciplinas obrigatórias do BCC e armazenar, em cada registro da coleção `alunos`, a lista das obrigatórias faltantes.

No intervalo que antecede a inscrição das disciplinas no início de cada período, é

---

<sup>17</sup> <https://www.python.org/>

disponibilizado um documento contendo as disciplinas, docentes que a ministrarão, os horários e dias da semana. A partir desse documento, foi criado um *script* para percorrer as informações importantes e persisti-las na coleção **ofertas** do banco no MongoDB, e são essas disciplinas que serão utilizadas para a recomendação da grade. Como foram obtidos os dados até o período 2017/1, foi utilizado o documento de disciplinas ofertadas para o período de 2017/2 na recomendação.

## 6 AVALIAÇÃO DO SISTEMA

Neste capítulo, será descrita a etapa de experimentação com alunos e professores (Apêndice C), apresentando o sistema e permitindo testá-lo, a fim de validar as funcionalidades implementadas e avaliar as críticas de possíveis melhorias para o projeto futuramente. Durante toda a conversa houve liberdade para o recebimento de críticas sobre o sistema, assim como sugestões e questionamentos para melhorias do mesmo.

Nesta etapa, foram realizados encontros com alguns dos professores orientadores acadêmicos que participaram do processo de pesquisa e validação da ideia do sistema. A experimentação teve um total de 4 participantes e foi conduzida através, primeiramente, de uma conversa sobre o sistema, mostrando e explicando cada funcionalidade. A seguir, permitiu-se que os professores utilizassem e testassem o sistema. Em seguida, foi aplicado um questionário contendo duas seções: a primeira contemplando a avaliação de relevância das funcionalidades que foram implementadas e a segunda avaliando a usabilidade do sistema.

A experimentação com os alunos teve um total de 10 participantes e foi conduzida através da apresentação e utilização do sistema localmente, pois como os dados não estão anonimizados, foi de extrema importância acompanhar os alunos durante toda a etapa para garantir a privacidade dos dados. Em seguida, foi aplicado um questionário com perguntas divididas em 3 seções: recomendação, funcionalidades e usabilidade. Na primeira parte, os alunos avaliaram as grades recomendadas de acordo com a relevância das disciplinas sugeridas, se seguiria a recomendação ou não. Na segunda seção, foram avaliadas todas as funcionalidades do sistema, como já descritas anteriormente, desde a tela de estatísticas e histórico, passando pelo *ranking* até a página de recomendação de grade. Por último, foi avaliada a usabilidade do sistema como um todo.

## 6.1 AVALIAÇÃO DAS FUNCIONALIDADES

Em ambos os questionários, tanto para alunos quanto professores, foi pedido que avaliassem o sistema de acordo com as seguintes funcionalidades:

1. Visualizar dados estatísticos do progresso no curso
2. Visualizar histórico de disciplinas cursadas
3. Ranking de disciplinas obrigatórias
4. Ranking de disciplinas eletivas
5. Escolher horários disponíveis na recomendação de grade
6. Poder fixar e/ou ignorar disciplinas na recomendação de grade
7. Limitar quantidade de disciplinas recomendadas na grade
8. Filtrar recomendação para disciplinas obrigatórias/eletivas
9. Sugestão de grade como um todo

### 6.1.1 Resultado das Avaliações das Funcionalidades pelos Professores

Todos os professores entrevistados concordaram que as funcionalidades 1, 2, 3, 5, 6, 7 e 9 eram relevantes ou muito relevantes. Já para as funcionalidades 4 e 8, metade dos professores avaliaram como neutro e a outra metade como muito relevante ou relevante. Detalhes sobre os resultados podem ser vistos na Figura 22.

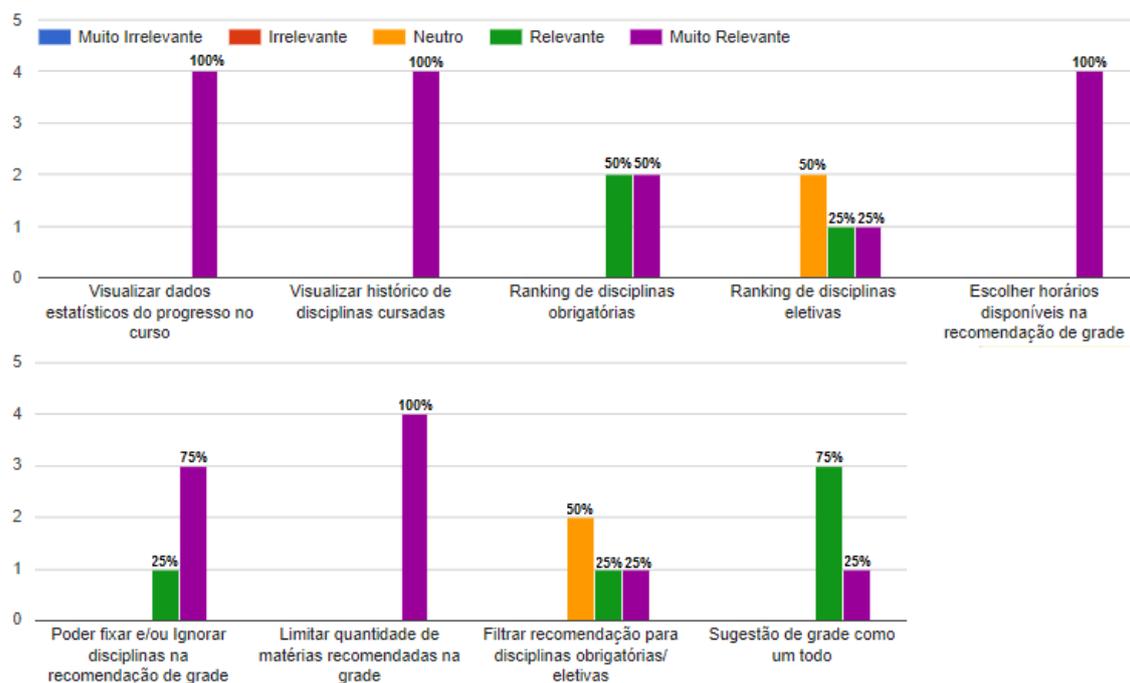


Figura 22: Questionário para professores - Avaliação das funcionalidades.

Com a análise das respostas e após uma conversa de *feedback* com os professores, foi possível perceber que o sistema seria de grande importância durante o processo de orientação acadêmica, pois diminuiria o trabalho de entender a situação atual do aluno no curso e facilitaria no entendimento de todo o progresso do mesmo. Assim, análises dos dados do aluno seriam mais fáceis e possibilitariam uma conversa mais objetiva sobre o que, de fato, poderia ser feito no decorrer do curso.

### 6.1.2 Resultado das Avaliações das Funcionalidades pelos Alunos

Todos os alunos que participaram do experimento avaliaram as funcionalidades 4, 5, 6, 7 e 9 como relevantes ou muito relevantes. Para as funcionalidades 1 e 3, 90% consideraram relevante ou muito relevante e apenas 10% votaram neutro. Para a funcionalidade 2, 40% votaram neutro e os 60% restantes votaram relevante ou muito relevante. Já para a funcionalidade 8, 80% acharam relevante ou muito relevante, 10% votaram neutro e os outros 10% acharam irrelevante. Os resultados

são apresentados na Figura 23.

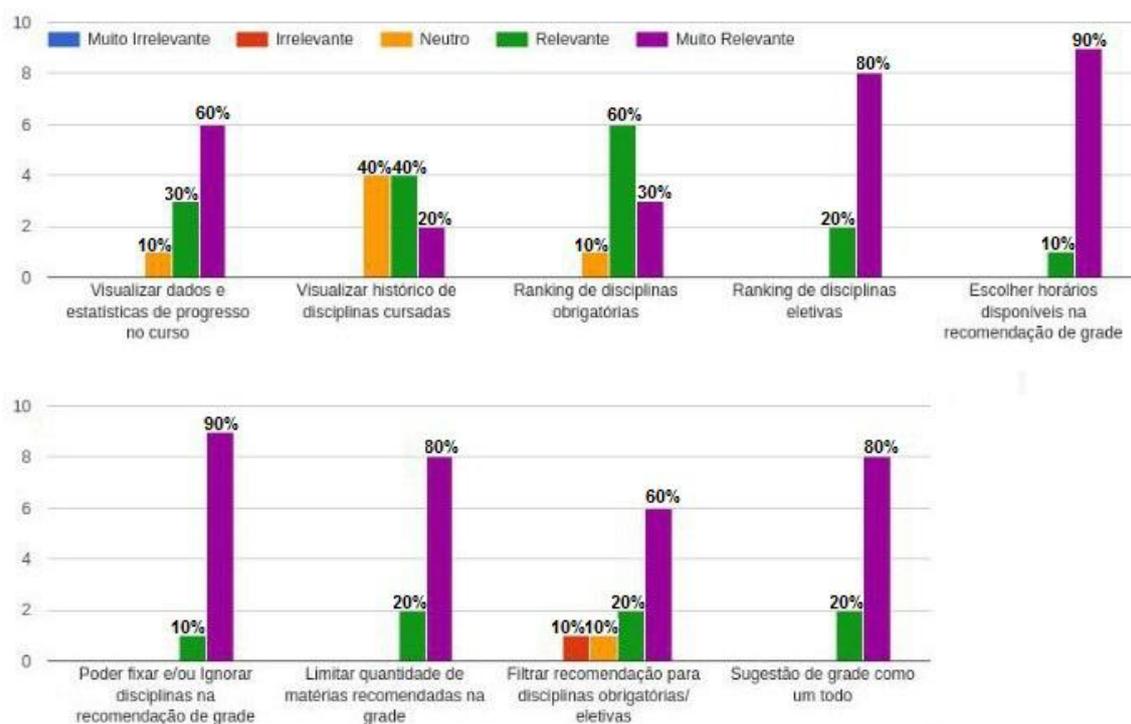


Figura 23: Questionário para alunos - Avaliação das funcionalidades.

## 6.2 AVALIAÇÃO DAS RECOMENDAÇÕES

No questionário dos alunos, além da avaliação das funcionalidades foi pedido para avaliarem a recomendação de grades em 3 tópicos:

1. Avaliar a grade SEM parametrizações
2. Avaliar a grade COM parametrizações
3. Avaliar a semelhança da grade com parametrizações com a de fato cursada pelo aluno em 2017/2 (essa avaliação foi possível porque foi realizada somente no final do período de 2018/1, então os alunos já sabiam a grade efetivamente cursada em 2017/2).

Os itens da pesquisa foram avaliados numa escala de 1 a 5, onde 1 é “Não cursaria

nenhuma disciplina sugerida” e 5 “Cursaria a grade inteira” para os itens 1 e 2 acima. Já para o item 3, 1 é “Totalmente distinta” e 5 “Idêntica”.

Para o item 1, 30% dos alunos disseram que cursariam a grade inteira recomendada, 60% cursariam parte da grade recomendada e apenas 10% não cursaria praticamente nenhuma disciplina recomendada, como observa-se na Figura 24.

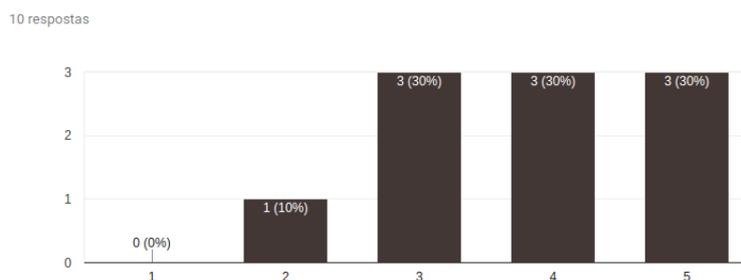


Figura 24: Avaliação de grade recomendada sem parametrização.

No item 2, 50% dos alunos responderam que cursariam a grade recomendada inteira, 40% cursariam grande parte da recomendação, e apenas 10% disseram que cursariam praticamente nenhuma das disciplinas recomendadas, como a Figura 25 ilustra.

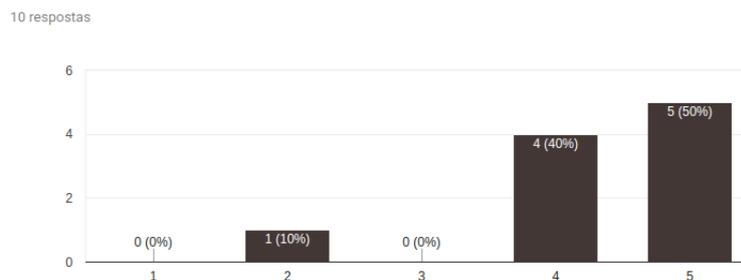


Figura 25: Avaliação de grade recomendada com parametrização.

Já no item 3, 40% dos alunos responderam que a grade recomendada com parametrizações ficou idêntica à cursada por eles no período 2017/2, 30% disseram que ficou bem parecida, 20% responderam que a grade estava pouco parecida e 10% disse que a grade estava totalmente diferente da por ele cursada. Como a Figura 26 ilustra.

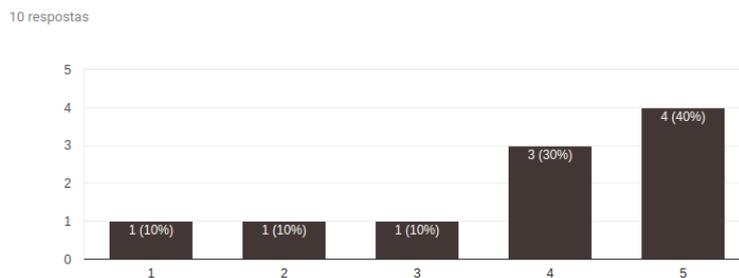


Figura 26: Semelhança da grade parametrizada com a cursada pelo aluno.

No geral, os alunos que responderam a pesquisa disseram que gostaram da maior parte das disciplinas que lhes foram recomendadas. Eles afirmaram que, mesmo que não fossem cursar imediatamente a mesma, já puderam ter alguma ideia do que escolher nos próximos períodos utilizando as recomendações do sistema.

Uma questão que foi levantada por boa parte dos alunos participantes durante o uso do sistema foi a recomendação de disciplinas eletivas baseadas em outras disciplinas similares já feitas, utilizando a área ou conteúdo das mesmas. Essa ideia foi inicialmente discutida no processo de desenvolvimento, pois se sabe que seriam informações relevantes não só para a integração com o sistema, mas também para a pesquisa durante a escolha de disciplinas a serem cursadas no início do período pelos alunos. Entretanto, análises sobre o conteúdo ou área das disciplinas ficaram fora do escopo do presente projeto.

Outra sugestão foi a possibilidade de ter informações sobre a ementa de cada disciplina ofertada dentro do sistema. Segundo a opinião de alguns alunos, mesmo a disciplina tendo um *score* alto na recomendação, o conteúdo que será apresentado é bastante relevante para a escolha de cursar a mesma. Assim, se o sistema já contemplasse esses dados, seria mais um fator que facilitaria a escolha da grade a ser cursada, pois não precisaria entrar no sistema do SIGA para obter esta informação.

### 6.3 TESTES DE USABILIDADE

Para realizar os testes de Usabilidade, foi utilizada a metodologia do SUS (*System Usability Scale*), que é uma técnica de medição quantitativa, precisa e simples,

amplamente utilizada, desenvolvida por John Brooke [5]. O SUS é composto por um questionário com 10 afirmações em escala de 5 pontos, onde 1 representa “discordo totalmente” e 5 significa “concordo totalmente”, para que seja avaliado o nível de concordância com o sistema. Metade das perguntas são redigidas de forma positiva, e a outra metade negativa, de maneira intercalada. O questionário pode ser observado na Figura 27.

A contribuição de cada item é feito da seguinte maneira: a pontuação varia de 0 a 4, onde os itens ímpares valem a posição da escala marcada menos 1, e os itens pares valem 5 subtraído da posição marcada. Por fim, multiplica-se a soma das pontuações por 2,5 para obter a pontuação geral do SUS, em uma escala de 0 a 100. Pontuações do SUS abaixo de 60 representam sistemas com experiências relativamente pobres e insatisfação do usuário, e pontuações acima de 80 pontos representam experiências muito boas com alto índice de satisfação dos usuários.

Segundo Albert et al. [2], o SUS é uma técnica de medição quantitativa precisa e simples, apresentando ótimo rendimento e consistência de resultados para testes com tamanhos relativamente pequenos de amostras. Com um número de 8 participantes já é possível identificar preferências e problemas através desse sistema de avaliação com 80% de precisão. Isso, de acordo com Tullis et al. [16], é possível devido ao uso de perguntas positivas e negativas que devem ser avaliadas e que deixam os avaliadores mais atentos.

	Discordo Totalmente	Discordo	Neutro	Concordo	Concordo Totalmente
Acho que gostaria de usar este sistema com frequência.	<input type="radio"/>				
Achei o sistema desnecessariamente complexo.	<input type="radio"/>				
Achei o sistema fácil de usar.	<input type="radio"/>				
Achei que seria necessário o apoio de um técnico para poder usar este sistema.	<input type="radio"/>				
As funções deste sistema estavam bem integradas.	<input type="radio"/>				
Achei este sistema muito inconsistente.	<input type="radio"/>				
Imagino que a maioria das pessoas aprenderiam a usar este sistema rapidamente.	<input type="radio"/>				
Achei o sistema muito complicado de usar.	<input type="radio"/>				
Eu me senti muito confiante com o sistema.	<input type="radio"/>				
Eu preciso aprender um monte de coisas antes de continuar usando este sistema.	<input type="radio"/>				

Figura 27: Questionário do SUS.

### **6.3.1 Resultado dos Testes de Usabilidade com Professores**

Os testes de usabilidade realizados com os professores, segundo a métrica do SUS, obtiveram um resultado de 92 pontos em um conjunto de apenas 4 professores, pois não foi possível encontrar em um tempo hábil mais professores para realizar os testes. Durante todas as conversas, foram discutidos os inúmeros ganhos de se ter uma visualização das informações de cada aluno de uma forma tão fácil e simples utilizando o sistema, que seriam de extrema importância durante as orientações acadêmicas. Também foram comentados sobre quais seriam os esforços para colocar o sistema em produção, junto ao SIGA.

### **6.3.2 Resultado dos Testes de Usabilidade com Alunos**

Os testes de usabilidade realizados com os alunos, segundo a métrica do SUS, obtiveram um resultado de 91 pontos em um conjunto de 10 alunos. O que foi levado em consideração nas críticas realizadas pelos alunos foram alguns detalhes na interface, sugestões de melhorias nos botões, posição dos elementos e textos e o comportamento do bloqueio de horários na grade. Contudo, todos elogiaram a facilidade de utilizar o sistema e sua interface, além de poder ter informações sobre o curso de maneira rápida e também permitir customizar de diversas formas a grade recomendada.

## 7 CONCLUSÃO

O desenvolvimento do trabalho foi muito interessante, pois proporcionou uma grande aplicação de diversos conteúdos aprendidos durante toda a graduação, passando por conhecimentos adquiridos nas disciplinas de Computação I e II, devido à lógica de programação envolvida; Engenharia de Software, para levantar requisitos e organizar o fluxo de desenvolvimento do projeto; Banco de Dados, para modelar, criar e dar manutenção na base de dados utilizada; Álgebra Linear, para realizar os cálculos matriciais e vetoriais utilizados na recomendação; mas, principalmente, a disciplina de Recuperação da Informação, que foi a responsável por mostrar as áreas de conhecimento e técnicas necessárias para a realização do projeto.

As entrevistas com os professores e alunos também foram interessantes, pois mostraram as visões dos dois lados no processo de escolha de disciplinas. Evidenciando o desafio por parte dos professores em entender a situação do aluno no curso, considerando os fatores físicos e psicológicos do mesmo, bem como fazer um planejamento que seja viável e que auxilie seu orientando durante o andamento do curso. Já, por parte dos alunos, as dificuldades evidenciadas são sobre quais disciplinas eletivas escolher, qual área de atuação seguir, como conciliar os horários de sua grade e o quê fazer em casos de reprovação nas disciplinas, principalmente as obrigatórias.

O sistema idealizado inicialmente era mais voltado para o desenvolvimento de múltiplas métricas de recomendação de disciplinas. Contudo, após as entrevistas em que foram discutidos problemas enfrentados por professores e alunos, as funcionalidades do sistema foram repensadas, evidenciando a importância de outras questões como as configurações na tela de recomendação de grade e estatística de usuário, por exemplo. Neste momento, a aplicação que inicialmente foi pensada para ser puramente um sistema de recomendação, tomou um escopo maior em auxiliar alunos e professores orientadores no planejamento da jornada acadêmica dos alunos.

O *feedback* obtido por alunos e professores no final evidenciou o valor do sistema desenvolvido. A grande maioria das pessoas que tiveram a oportunidade de utilizar o sistema, ou até alunos/professores que souberam da proposta, mostraram-se bas-

tante interessados e fizeram questão de dizer que gostariam de utilizar este tipo de sistema no seu dia a dia.

## 7.1 DIFICULDADES ENCONTRADAS

Uma das maiores dificuldades foi encontrar uma maneira de integrar o sistema desenvolvido com a base de dados do SIGA, de modo que o sistema obtivesse os dados necessários continuamente, por exemplo através de uma API que o realimentasse. Por conta deste problema de integração, os dados fornecidos foram um *snapshot* da base de dados entre os períodos letivos 2010/1 a 2017/1. Além disso, os dados foram enviados em um formato bruto, com objetos que representam todas as entidades envolvidas combinadas, o que tornou necessária uma etapa de pré-processamento e formatação dos dados antes que fossem efetivamente utilizados pelo sistema.

Esta segregação em relação à base de dados no SIGA afetou alguns outros aspectos do sistema. Um exemplo disso é a obtenção das disciplinas ofertadas no início do período, que é necessária para montar a tela de sugestão de grades. Estes dados foram obtidos através de um processamento da página com a lista de disciplinas ofertadas para o BCC. Os dados julgados necessários, inicialmente foram escolhidos através de uma reunião com o grupo e a orientadora e, em seguida, foi levado à coordenação do BCC, de modo que fosse obtido um memorando solicitando os dados ao SIGA. Esse processo se dava de maneira lenta, por questões burocráticas e de segurança, portanto, qualquer outro dado necessário durante o desenvolvimento era obtido através de outros meios, principalmente processando as páginas do DCC, que não necessariamente estavam sempre atualizadas. Detalhes de implementação de *scripts* necessários nesta parte foram descritos no Capítulo 5.

Outra dificuldade encontrada foi combinar métricas de recomendação que fossem úteis tanto aos alunos quanto aos professores. Por mais que existam pontos de concordância, como a ideia de priorizar disciplinas atrasadas, por exemplo, alunos e professores orientadores têm visões diferentes do processo, e portanto suas opiniões divergem em alguns aspectos, como foi discutido no Capítulo 3.

Com relação aos dados em si, alguns casos especiais mostraram-se uma das grandes dificuldades no desenvolvimento. Um deles era o caso de alunos que se transferiram para o curso de Bacharelado em Ciência da Computação, pois tinham cursado disciplinas que serviram como equivalência para as disciplinas do BCC e outro caso era de alunos que tiveram o grau de alguma disciplina lançado no SIGA após o período regular, aparecendo como cursada em um período /0 ou /3, como dito no Capítulo 5.

Por fim, o escopo do projeto reduziu-se explicitamente para o curso de Bacharelado em Ciência da Computação da UFRJ. Alguns casos especiais, apresentados nos dados disponibilizados pelo SIGA, foram tratados, mas outros podem ter passado despercebidos.

## 7.2 TRABALHOS FUTUROS

Existem diversos pontos de melhoria no sistema como um todo, que podem ser abordados no futuro. A melhoria imediata seria uma integração com o SIGA, permitindo possibilidade de funcionamento contínuo do sistema, bem como acesso a outros tipos de dados que não foram utilizados. Outra possibilidade seria integrar o projeto com o sistema de login único CAS<sup>1</sup> da UFRJ, e assim poder ter um controle de acesso seguro às informações.

O trabalho é perfeitamente generalizável para outros cursos ou até universidades apesar de ter sido desenvolvido tendo em mente os alunos de graduação do curso de Bacharelado em Ciência da Computação da Universidade Federal do Rio de Janeiro.

Sobre a recomendação em si, é possível refinar resultados utilizando diferentes abordagens, como por exemplo: adoção de um sistema onde os alunos avaliam cada disciplina realizada para melhorar a filtragem colaborativa, já que o sistema assume que o aluno gostou de toda disciplina eletiva cursada em que foi aprovado, e não necessariamente isso é verdade. Outra possível abordagem é combinar técnicas de recomendação baseada em conteúdo utilizando as áreas de atuação das disciplinas,

---

<sup>1</sup><https://cas.ufrj.br>

por exemplo. É importante notar que para utilizar este dado em particular seria ideal uma categorização formal das disciplinas por parte do BCC.

Um outro trabalho futuro é separar o acesso de professores e alunos. Desta forma, seria possível prover funcionalidades adequadas para cada tipo de usuário e proporcionar uma melhor experiência de usabilidade para ambos.

## REFERÊNCIAS

- [1] AHER, S. B.; LOBO, L. Course recommender system in e-learning. **International Journal of Computer Science and Communication**, [S.l.], v. 3, n. 1, p. 159-164, 2012.
- [2] ALBERT, W.; TULLIS, T. **Measuring the user experience**: collecting, analyzing, and presenting usability metrics. Amsterdam: Morgan Kaufmann, 2013.
- [3] BAEZA-YATES, R.; RIBEIRO-NETO, B. **Modern information retrieval**. New York: ACM Press, 1999.
- [4] BAKHSHINATEGH, B. **Design of a course recommender system as an application of collecting graduating attributes**. 2016. 64 f. Tese (Doutorado)-University of Alberta, Edmonton, 2016.
- [5] BROOKE, J. **SUS**: a quick and dirty usability scale. 1995. Disponível em: <<https://hell.meiert.org/core/pdf/sus.pdf>>. Acesso em: 10 jul. 2018.
- [6] BURKE, R. **Knowledge-based recommender systems**. 2000. Disponível em: <<https://pdfs.semanticscholar.org/1b50/ff4e5d8420f3ffae7de2a060b5a6fd4b8023.pdf>>. Acesso em: 10 jul. 2018.
- [7] BYDŽOVSKÁ, H. **Course enrolment recommender system**. 2013. Disponível em: <[http://www.educationaldatamining.org/EDM2016/proceedings/paper\\_83.pdf](http://www.educationaldatamining.org/EDM2016/proceedings/paper_83.pdf)>. Acesso em: 03 jul. 2018.
- [8] CHANG, P.-C.; LIN, C.-H.; CHEN, M.-H. A hybrid course recommendation system by integrating collaborative filtering and artificial immune systems. **Algorithms**, [S.l.], v. 9, n. 3, 2016.
- [9] DWIVEDI, S.; ROSHNI, V. K. Recommender system for big data in education. In: NATIONAL CONFERENCE ON E-LEARNING & E-LEARNING TECHNOLOGIES, 5., 2017, Hyderabad. **Proceedings...** New York: IEEE, 2017.
- [10] EKSTRAND, M. D.; RIEDL, J. T.; KONSTAN, J. A. Collaborative filtering recommender systems. **Journal Foundations Trends in Human-Computer Interaction**, [S.l.], v. 4, n. 2, p. 81-173, 2011.
- [11] IYENGAR, M.; SARKAR, A.; SINGH, S. A collaborative filtering based model for recommending graduate schools. In: INTERNATIONAL CONFERENCE ON MODELING, SIMULATION, AND APPLIED OPTIMIZATION, 7., 2017, Sharjah. **Proceedings...** New York: IEEE, 2017.
- [12] JANNACH, D. et al. **Recommender systems**: an introduction. Cambridge: Cambridge University Press, 2010.

- [13] MANNING, C. D.; RAGHAVAN, P.; SCHÜTZE, H. **Introduction to information retrieval**. Cambridge: Cambridge University Press, 2008.
- [14] NISHINO, K. et al. The development of a course recommendation system for e-learning students. **International Journal of Knowledge and Web Intelligence**, [S.l.], v. 3, n. 1, p. 19-32, 2012.
- [15] RODRÍGUEZ, P. A.; OVALLE, D. A.; DUQUE, N. D. A student-centered hybrid recommender system to provide relevant learning objects from repositories. In: ZAPHIRIS, P.; IOANNOU, A. (Ed.). **Learning and collaboration technologies: Second International Conference, LCT 2015, Held as Part of HCI International 2015, Los Angeles, CA, USA, August 2–7, 2015, Proceedings**. New York: Springer, 2015. p. 291-300.
- [16] TULLIS, T. S.; STETSON, J. N. **A comparison of questionnaires for assessing website usability**. 2004. Disponível em: <<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.396.3677&rep=rep1&type=pdf>>. Acesso em: 12 jul. 2018.

## **A ROTEIRO DE ENTREVISTA COM PROFESSORES**

### **A.1 PERGUNTAS GERAIS**

1. Você poderia descrever, em geral, as principais estratégias que costuma adotar na orientação dos alunos quanto à escolha das disciplinas a serem cursadas em um determinado período?

2. Poderia enumerar os principais critérios adequados e inadequados para a escolha de disciplinas?

### **A.2 CENÁRIOS ESPECÍFICOS PARA DISCUSSÃO**

1. Ao orientar um aluno na escolha de disciplinas, é importante levar em consideração alternativas que minimizem o tempo para conclusão de curso ao mesmo tempo que maximizem a chance de aprovação nas disciplinas cursadas. Você concorda com essa afirmação? Quais possíveis estratégias podem auxiliar a obter um “balanceamento” adequado entre carga horária por período e tempo para conclusão do curso?

2. Ao orientar um aluno na escolha de disciplinas, você acha que verificar o “grau de dificuldade” das disciplinas que serão realizadas é um fator importante a ser considerado? Uma estimativa em função do índice histórico de reprovações e/ou média das notas obtidas pelas turmas anteriores poderia auxiliar nessa identificação? Além disso, o professor que ministrará a disciplina é um fator que deve ser considerado na escolha da disciplina? Se sim, em quais casos?

3. Ao orientar um aluno que teve reprovação(ões) no período anterior, quais as possíveis recomendações? Por exemplo, reduzir, aumentar ou manter a mesma carga horária no próximo período? Se possível, cursar imediatamente as disciplinas reprovadas no período anterior ou aguardar para cursá-las em um período futuro? O que se deve analisar para tomar essas decisões na orientação? Recomenda-se cursar novamente uma disciplina eletiva na qual o aluno obteve alguma reprovação? Em

quais situações?

4. Ao orientar um aluno próximo a se enquadrar em algum critério de jubila-mento, existem cuidados adicionais a serem adotados na escolha das disciplinas? Quais as principais recomendações (da COAA ou coordenação de curso) frente a esses casos?

5. Ao orientar um aluno na escolha de disciplinas eletivas, quais fatores podem auxiliar nessa escolha? Por exemplo, é importante identificar uma linha/área de interesse ou uma diversificação de disciplinas de diferentes linhas/áreas? Como, em geral, é possível verificar o interesse/aptidão de um aluno em relação à(s) área(s) específica(s)? Acredita que o desempenho do aluno nas disciplinas obrigatórias possa auxiliar nessa identificação?

### A.3 UTILIDADE DE UM SISTEMA DE RECOMENDAÇÃO NESSE CONTEXTO

1. Você acha que um Sistema de Recomendação (grade horária para cada aluno) poderia auxiliar os alunos na escolha das disciplinas? Quais características o sistema deveria contemplar para ser útil nesse processo?

2. Você gostaria de utilizar um sistema como este para auxiliar no processo de Orientação Acadêmica? Quais características o sistema deveria contemplar para ser útil nesse processo?

### A.4 FINAL

1. Momento aberto para comentários e sugestões.
2. Agradecimento.

## **B QUESTIONÁRIO DE PESQUISA COM ALUNOS**

### **B.1 PERGUNTAS GERAIS**

1. Qual período você está cursando?
2. Em quantos períodos você pretende se formar a partir do atual?
3. Qual período você considera ideal para começar a cursar disciplinas eletivas?
4. Você faz/fez alguma atividade complementar?
5. Em qual período você começou a realizar atividades complementares?
6. Quantas disciplinas por período você considera ideal de serem cursadas, sem a realização de atividades complementares?
7. Quantas disciplinas por período você considera ideal de serem cursadas, caso haja a realização de atividades complementares?

### **B.2 CRITÉRIOS PARA ESCOLHA DE DISCIPLINAS ELETIVAS**

1. Avaliação dos critérios de acordo com sua relevância (1 a 5)
  - (a) Grau de dificuldade da disciplina
  - (b) Professor que irá ministrar a disciplina
  - (c) Área da disciplina
  - (d) Dia/horário em que a disciplina é ofertada
  - (e) Indicação de outros alunos
2. Você escolhe as disciplinas eletivas em função dos critérios de avaliação? Se sim, quais você prefere?
  - (a) Não ligo para o critério de avaliação
  - (b) Apenas provas

(c) Apenas trabalhos

(d) Provas e trabalhos

(e) Outros

3. Você prefere diversificar as áreas de suas disciplinas eletivas ou focar em uma área específica?

(a) Puxo/puxarei sempre disciplinas da mesma área

(b) Inicialmente tentei/tentarei diversificar para descobrir minha aptidão e depois me concentrei/concentrarei nela

(c) Tento/tentarei puxar disciplinas das mais diversas áreas

(d) Outros

4. Você já encontrou sua área de interesse? Se sim, seu desempenho nas outras disciplinas auxiliou neste processo?

(a) Não encontrei

(b) Encontrei, mas meu desempenho nas outras não auxiliou

(c) Encontrei apenas com auxílio das disciplinas obrigatórias

(d) Encontrei apenas com auxílio das disciplinas eletivas

(e) Encontrei com auxílio de disciplinas obrigatórias e eletivas

(f) Outros

### B.3 CRITÉRIOS PARA ESCOLHA DE DISCIPLINAS OBRIGATÓRIAS

1. Avaliação dos critérios de acordo com sua relevância (1 a 5)

(a) Professor que irá ministrar a disciplina

(b) Dia/horário em que a disciplina é ofertada

(c) Priorizar disciplina se é pré-requisito de outra(s)

(d) Seguir o planejamento previsto pelo departamento

2. Em caso de reprovação (incluindo múltiplas reprovações) em disciplinas obrigatórias, você considera cursá-las novamente já no período imediatamente posterior à reprovação?

(a) Sim

(b) Não

(c) Outros

#### B.4 FUNCIONALIDADES DO SISTEMA DE RECOMENDAÇÃO DE DISCIPLINAS

1. Avaliação dos critérios de acordo com sua relevância (1 a 5)

(a) Visualizar histórico de grades cursadas anteriormente

(b) Recomendar grades para o próximo período

(c) Permitir limitar quantidade total de créditos por período na sugestão de grade

(d) Habilitar/Desabilitar recomendação de eletivas

(e) Habilitar/Desabilitar recomendação de obrigatórias

(f) Permitir filtrar disciplinas por horário na recomendação de grade

(g) Permitir fixar disciplinas (as quais o aluno pretende cursar) para que o sistema só complete a grade com as recomendações

(h) Exibir lista de disciplinas eletivas recomendadas por ordem de relevância

2. Área livre para sugestões de outras funcionalidades relevantes

## **C QUESTIONÁRIO DE AVALIAÇÃO DO SISTEMA**

### **C.1 AVALIAÇÃO DA RECOMENDAÇÃO**

1. Avalie uma grade recomendada SEM parametrizações (1 a 5)
2. Avalie uma grade recomendada COM parametrizações (1 a 5)
3. Avalie a semelhança entre a grade COM parametrizações sugerida e a grade cursada no período 2017.2 (1 a 5)
4. Avalie se gostou ou não de cada uma das disciplinas recomendadas em sua grade com parametrizações
5. Campo livre para observação ou comentário sobre as recomendações

### **C.2 AVALIAÇÃO DAS FUNCIONALIDADES**

1. Avaliação dos critérios de acordo com sua relevância (1 a 5)
  - (a) Visualizar dados estatísticos do progresso no curso
  - (b) Visualizar histórico de disciplinas cursadas
  - (c) Ranking de disciplinas obrigatórias
  - (d) Ranking de disciplinas eletivas
  - (e) Escolher horários disponíveis na recomendação de grade
  - (f) Poder fixar e/ou ignorar disciplinas na recomendação de grade
  - (g) Limitar quantidade de disciplinas recomendadas na grade
  - (h) Filtrar recomendação para disciplinas obrigatórias/eletivas
  - (i) Sugestão de grade como um todo
2. Campo livre para observação ou comentário sobre as funcionalidades

### C.3 AVALIAÇÃO DE USABILIDADE

1. Avaliação dos critérios de acordo com sua relevância (1 a 5)
  - (a) Acho que gostaria de usar este sistema com frequência
  - (b) Achei o sistema desnecessariamente complexo
  - (c) Achei o sistema fácil de usar
  - (d) Achei que seria necessário o apoio de um técnico para poder usar este sistema
  - (e) As funções deste sistema estavam bem integradas
  - (f) Achei este sistema muito inconsistente
  - (g) Imagino que a maioria das pessoas aprenderiam a usar este sistema rapidamente
  - (h) Achei o sistema muito complicado de usar
  - (i) Eu me senti muito confiante com o sistema
  - (j) Eu preciso aprender um monte de coisas antes de continuar usando este sistema
2. Campo livre para observação ou comentário sobre a usabilidade do sistema

### C.4 CONSIDERAÇÕES FINAIS

1. Campo livre para observação final ou comentário sobre o sistema como um todo

## D CÓDIGOS PYTHON

Abaixo encontram-se os códigos Python criados para pré-processar e persistir os dados no banco de dados.

Código D.1: *Script* para obter a lista de pré-requisitos de cada disciplina de acordo com a tabela obtida na página do SIGA

```

thisPath=os.path.dirname(os.path.abspath(__file__))
filePath=os.path.join(thisPath,'datasets/grade.html')

htmlFile=open(filePath,'r',encoding='utf-8')
html=htmlFile.read().replace('\n','')

tree=etree.HTML(html)
nodes=tree.xpath("//*[contains(@class,'tableBodyBlue')]")

for node in nodes:
    tdCod=node.xpath("./td/a")
    if not tdCod:
        continue
    codigo=tdCod[0].text
    preReqs=node.xpath("./td[7]")[0].text.strip().split(',')
    if(preReqs[0] == ''):
        preReqs= []
    disciplinasCollection.update_one({'codigo':codigo}, {'$set':
        {'preReqs':preReqs}})

```

Código D.2: *Script* para popular as equivalências de cada disciplina de acordo com a tabela obtida na página do SIGA

```

thisPath=os.path.dirname(os.path.abspath(__file__))
filePath=os.path.join(thisPath,'datasets/tabelaEquivalencias.
    html')

htmlFile=open(filePath,'r',encoding='utf-8')
html=htmlFile.read().replace('\n','')

tree=etree.HTML(html)
nodes=tree.xpath("//tr[contains(@class,'tableBodyBlue')]")

for node in nodes: codigo=node.xpath("//td[2]//nobr")[0].text.strip()
    .split("
        ") [0]
    equivalencias=node.xpath("//td[4]//nobr")[0].text.strip().
        split("+")

    codigoEqs = []
    for eq in equivalencias:
        codigoEq=eq.split("
            ") [0]
        codigoEqs.append(codigoEq)
    disciplinasCollection.update_one({'codigo':codigo}, {'$set':
        {'equivalencias':codigoEqs}})

    if len(codigoEqs) > 1: disciplinasCollection.update_on
        e({'codigo':codigoEqs[0]},
            {'$set': {'equivalencias': [codigo,codigoEqs[1]]}})
        disciplinasCollection.update_one({'codigo':codigoEqs[1]},
            {'$set': {'equivalencias': [codigo,codigoEqs[0]]}})
    else:
        disciplinasCollection.update_one({'codigo':codigoEqs[0]},
            {'$set': {'equivalencias': [codigo]}})

```

Código D.3: *Script* para obter as disciplinas obrigatórias faltantes para cada aluno

```

def verificaEquivalencia(codigo, listaDisciplinas):
    disciplina = DisciplinasCollection.find_one({'codigo': codigo})
    if "equivalencias" in disciplina:
        for eq in disciplina["equivalencias"]:
            if eq in listaDisciplinas:
                return eq
    return ""

for disciplina in DisciplinasCollection.find():
    if (disciplina['tipo'] == 'Obrigatoria'):
        DisciplinasSet.add(disciplina['codigo'])

for aluno in AlunosCollection.find():
    AlunosDict[aluno['dre']] = set(DisciplinasSet)

for turma in TurmasCollection.find():
    dre = turma['dre']
    codigo = turma['codigoDisciplina']
    if codigo in AlunosDict[dre]:
        AlunosDict[dre].remove(codigo)
    else:
        eq = verificaEquivalencia(codigo, AlunosDict[dre])
        if eq != "":
            AlunosDict[dre].remove(eq)

for dre in AlunosDict:
    AlunosCollection.update_one({'dre': dre}, {'$set': {'obrigatoriasFaltantes': list(AlunosDict[dre])}})

```

## Código D.4: Script para obter as disciplinas ofertadas em 2017-2

```

connectToMongo("tcc-database", "ofertas")
html=readHtml('datasets/disciplinas2017-2.html')
tree=etree.HTML(html)
nodes=tree.xpath("//*[table[contains(@class,'lineBorder')]]")

disciplinasList= []
for node in nodes:
    disciplinaDict= {}
    disciplinaDict['periodoOfertado'] = "2017/2"
    periodo=node.xpath("//*[center//b]")[0].text[0]
    try:
        disciplinaDict['periodoDisciplina'] =int(periodo)
    except ValueError:
        disciplinaDict['periodoDisciplina'] = 0
    materiaNodes=node.xpath("//*[tr[contains(@class,'tableBody')]]")
    for mNode in materiaNodes:
        disciplinaDict['codigo'] =mNode.xpath("//*[td[1]//a]")[0].text.strip().replace("\t", "")
        disciplinaDict['nome'] =mNode.xpath("//*[td[2]]")[0].text.strip().replace("\t", "")
        disciplinaDict['professor'] =mNode.xpath("//*[td[3]]")[0].text.strip().replace("\t", "")
        horarios=mNode.xpath("//*[td[4]]")[0].text.strip().replace("\t", "")
        disciplinaDict['horarios'] =parseHorarios(horarios)
        disciplinasList.append(dict(disciplinaDict))
disciplinasCollection.insert_many(disciplinasList)

```

## E ARQUITETURA BACK-END

Este apêndice contempla exemplos de cada camada desenvolvida no *Back-end* da aplicação.

Código E.1: Exemplo de *Controller* (Javascript)

```
function disciplinaController(disciplinaService) {
  async function get(req, res) {
    try {
      const disciplinas = await disciplinaService
        .getByCodigo(req.params.codigo);
      res.json(disciplinas);
    }
    catch (err) {
      res.send(err);
    }
  };
  async function listar(req, res) {
    try {
      const disciplinas = await disciplinaService.listar();
      res.json(disciplinas);
    }
    catch (err) {
      res.send(err);
    }
  };
  return { get, listar };
}
module.exports = { disciplinaController }
```

Código E.2: Exemplo de *Service* (Javascript)

```

function disciplinaService(disciplinaCollection) {
  function listar() {
    return disciplinaCollection.find({}).exec();
  };
  function listarObrigatorias() {
    return disciplinaCollection.find({'tipo':'Obrigatoria'})
      .exec();
  }
  function listarEletivas() {
    return disciplinaCollection.find({'tipo':'Eletiva'}).
      exec();
  }
  function getByCodigo(codigo) {
    return disciplinaCollection.findOne({'codigo':codigo}).
      exec();
  }
  function disciplinasObrigatoriasCursadas(faltantes, disciplinas
  ) {
    return disciplinas.filter((cursada) => {
      if(faltantes.find(f=>f.codigo===cursada.codigo)
        !==undefined) {
        return false;
      }
      return true;
    });
  }
  return {listar,listarObrigatorias,listarEletivas,getByCodigo,
    disciplinasObrigatoriasCursadas}
}
module.exports= {disciplinaService}

```

Código E.3: Exemplo de *Schema* (Javascript)

```
var mongoose=require('mongoose');
var Schema=mongoose.Schema;
var DisciplinaSchema=new Schema({
  cargaHoraria: {type:Number},
  nome: {type:String},
  previstoCurriculo: {type:Number},
  tipo: {type:String},
  codigo: {type:String},
  periodoRecomendado: {type:Number},
  credits: {type:Number},
  preReqs: {type: [{type:String}] },
  equivalencias: {type: [{type:String}] }
});
module.exports=mongoose.model('Disciplinas',DisciplinaSchema);
```