



UNIVERSIDADE DE SÃO PAULO

Taís Aparecida Pereira Pinheiro

**TUTORIA INTELIGENTE UTILIZANDO SISTEMAS DE
RECOMENDAÇÃO**

São Paulo

2018

Taís Aparecida Pereira Pinheiro

**TUTORIA INTELIGENTE UTILIZANDO SISTEMAS DE
RECOMENDAÇÃO**

Monografia final da disciplina MAC0499 -
Trabalho de Formatura Supervisionado.

Orientador: Kelly Rosa Braghetto

São Paulo

2018

Dedico este trabalho aos alunos do ensino fundamental da escola estadual Professor Rogério
Levorin, em especial à minha sobrinha Isabela.

It will always be "Day 1".

Agradecimentos

Agradeço aos meus pais Lúdia e Adão, aos meus irmãos Lidiane e Dionatas, aos meus amigos, o pessoal da Seção de Informática do IME e em especial aos meus professores do ensino básico e do superior.

Resumo

A internet tem proporcionado às pessoas um verdadeiro oceano de informações de todos os tipos, o que naturalmente acarreta na dificuldade de extrair dela as informações de que se necessita e de uma maneira proveitosa. Em um contexto onde a computação passou a ser tão fundamental na educação quanto as demais disciplinas, observamos também um demasiado crescimento de ferramentas e recursos digitais educacionais disponíveis na Web. Na abordagem desse problema aplicado ao contexto educacional, este trabalho estuda a teoria dos chamados Sistemas de Recomendação e como aplicar alguns dos algoritmos envolvidos em um Sistema de Tutoria Inteligente. A aplicação é uma projeção dos algoritmos estudados em protótipo de Sistema Tutor Inteligente para o projeto Pré-ETEC que é usado como estudo de caso.

Palavras Chave: Sistema Tutor Inteligente, Sistemas de Recomendação, Educação.

Abstract

The internet has given people a real ocean of information of all kinds, which naturally entails the difficulty of extracting the information that is needed and in a useful way. In a context where computing has become as fundamental in education as other disciplines, we also see too much growth of digital tools and educational resources available on the Web. In this approach to the educational context, this work studies the theory of so-called Recommendation Systems and applies this theory in an Intelligent Tutoring System. The application is a projection of the algorithms studied in prototype of a system for the Pre-ETEC project that is used as a case study.

Keywords: Intelligent Tutoring System, Recommendation Systems, Education.

Sumário

	Sumário	7
	Lista de ilustrações	9
	Lista de tabelas	9
1	INTRODUÇÃO	2
1.1	Justificativa	2
1.2	Projeto Pré-ETEC	2
1.3	Objetivos e Motivação	3
1.4	Estrutura do Documento	3
2	SISTEMAS DE RECOMENDAÇÃO	4
2.1	Fundamento	4
2.1.1	Baseado em conteúdo	5
2.1.2	Filtragem Colaborativa	6
2.2	Recomendação de recursos educacionais	7
2.2.1	LORSys: um Sistema de Recomendação de Objetos de Aprendizagem	8
2.2.2	Recomendação de Objetos de Aprendizagem Empregando Filtragem Colaborativa e Competências	9
3	SISTEMAS DE TUTORIA INTELIGENTE	11
3.1	Fundamento	11
3.2	Exemplos de STI's	12
4	MODELAGEM	14
4.1	Detalhamento	14
4.1.1	Quais usuários serão considerados?	14
4.1.2	Quais conteúdos serão trabalhados	14
4.1.3	Algoritmos de Recomendação	14
4.1.4	Critérios de similaridade	14
4.1.5	Estratégias de Recomendação	16
4.1.6	Perfis de Usuários	17
4.2	Projeção em um STI	17
4.2.1	Módulo Domínio	17
4.2.2	Módulo Aluno	17
4.2.3	Módulo Tutor	17
4.2.4	Módulo Interface	18
5	IMPLEMENTAÇÃO E TESTES	19

5.1	Implementação	19
5.1.1	Modulo Aluno	19
5.1.2	Modulo Domínio	20
5.1.3	Modulo Tutor	22
5.1.4	Modulo Interface	23
5.2	Testes	24
6	CONCLUSÃO	26
	REFERÊNCIAS	27

Lista de ilustrações

Figura 1 – Recomendação Baseado em Conteúdo	5
Figura 2 – Recomendação por Filtragem Colaborativa	6
Figura 3 – Exemplo de objetos digitais de aprendizagem	7
Figura 4 – LorSys - Definição de Relacionamentos	8
Figura 5 – LorSys - Recomendação	9
Figura 6 – Modelo proposto para o sistema	10
Figura 7 – Arquitetura Clássica - STI	11

Lista de tabelas

Tabela 1 – Objeto de aprendizagem: Questão	15
Tabela 2 – Objeto de aprendizagem: Questão Especial	15
Tabela 3 – Assuntos abordados nas questões de Matemática	20
Tabela 4 – Assuntos abordados nas questões de Humanidades	21
Tabela 5 – Assuntos abordados nas questões de Ciências	21
Tabela 6 – Assuntos abordados nas questões especiais	22

1 Introdução

Neste capítulo é apresentada resumidamente a pesquisa documentada nesta monografia, especificada na justificativa, motivação, no estudo de caso utilizado na abordagem do problema em questão, nos objetivos da dissertação, na metodologia utilizada e na organização do texto.

1.1 Justificativa

O advento da internet tem proporcionado um enorme volume de informações e também grande desenvolvimento de ferramentas e recursos digitais educacionais disponíveis na Web aberta e também em bibliotecas digitais privadas. Dada a grande oferta de informações e desses recursos, naturalmente torna-se uma tarefa complicada para um estudante ter acesso ao que mais se aproveita nas suas intenções de estudo ou trabalho, como observado por exemplo por [10]Costa, Aguiar e Magalhães (2013, p.58). Um usuário se depara com questões do tipo: *"Por onde começar?"*, *"São muitos materiais sobre o mesmo assunto. Qual ou quais deles seriam os ideais para mim?"* ou *"Quais materiais usar para testar o meu conhecimento?"*.

Na bordagem destes questionamentos em vários contextos, não só no âmbito educacional, surgiram os Sistemas de Recomendação Personalizados. Com enfoque em educação, surgiram também os Sistemas de Tutoria Inteligente. Esses tipos de sistemas podem ajudar significativamente no desempenho de atividades de usuários de recursos educacionais, como professores e alunos.

1.2 Projeto Pré-ETEC

A ETEC (Escola Técnica Estadual) de Francisco Morato tem crescido e se destacado entre as escolas da região no que se refere a preparar os alunos do Ensino Médio para o ingresso em universidades de prestígio, em especial, universidades públicas em todo o Brasil. Visando o bom desempenho da ETEC, o Pré-ETEC é um projeto voluntariado que tem como objetivo preparar alunos de Ensino Fundamental II para a prova de ingresso no Ensino Médio na ETEC de Francisco Morato.



O Projeto Pré-ETEC será o estudo de caso proposto. Ele consiste em aulas aos sábados para reforço do conteúdo que os alunos recebem durante as aulas regulares. Esse reforço é dado na Escola Estadual Professor Rogério Levorin e recebe alunos de várias escolas da cidade e da região.

1.3 Objetivos e Motivação

Na abordagem desse problema aplicado ao contexto educacional, este trabalho visa estudar a teoria dos Sistemas de Recomendação e alguns dos algoritmos utilizados neste contexto, além disso, essa pesquisa tem o objetivo de entender como alguns dos algoritmos estudados poderiam ser aplicados em um protótipo de Sistema de Tutoria Inteligente direcionado ao estudo de caso.

Objetiva-se especificamente:

1. Estudar alguns sistemas de recomendação utilizados na área da educação.
2. Estudar alguns algoritmos de recomendação que se aplicam no contexto educacional.
3. Com base nos itens anteriores, analisar como se poderia aplicar os algoritmos estudados em um protótipo de sistema de tutoria inteligente direcionado ao estudo de caso exibido.

1.4 Estrutura do Documento

Este documento está estruturada em 6 capítulos, ordenados pelo momento em que foram concluídos dentro do ciclo de vida desta pesquisa, a saber:

- O capítulo 1 apresenta a justificativa e motivação de uma forma macro e também um estudo de caso para o qual se direciona a pesquisa.

As bases teóricas são apresentadas em seguida:

- No capítulo 2 são apresentados os conceitos de Sistemas de Recomendação e alguns exemplos de aplicação no contexto educacional.
- No capítulo 3 é apresentado o conceito geral dos Sistemas de Tutoria Inteligente e alguns exemplos de aplicação no âmbito educacional.
- No capítulo 4 é apresentado como os conteúdos exibidos anteriormente poderiam ser utilizados na modelagem de um protótipo de Sistema Tutor Inteligente para o estudo de caso.
- No capítulo 5 verificamos como a modelagem poderia ser implementada na linguagem Python e os resultados dos testes da implementação.
- Finalmente, no capítulo 6 são apresentadas as conclusões obtidas no decorrer do desenvolvimento da pesquisa e sugestões para possível ampliação futura do protótipo.

2 Sistemas de Recomendação

Neste capítulo são apresentados conceitos de Sistemas de Recomendação e alguns exemplos de sistemas educacionais que utilizam estes conceitos.

2.1 Fundamento

As principais funções dos sistemas de recomendação são analisar os dados dos usuários e extrair informações úteis para futuras predições.

Em geral quando um usuário pesquisa um produto na internet como um livro ou um eletrônico, passa a receber sugestões de produtos relacionados ou de produtos do mesmo tipo mas de outros fabricantes ou fornecedores. Pois então, os dados que inseridos na busca pelo produto alimentam a base de um sistema de recomendação, neste caso, ligado ao e-commerce.

De uma maneira ampla, os sistemas de recomendação são recursos que tem por objetivo fazer sugestões que possam vir a ser de interesse do usuário. Essas sugestões se baseiam nas informações de perfil deste usuário ou nas escolhas feitas por outras pessoas com perfis semelhantes. Como é esperado, esse tipo de sistema é amplamente utilizado em estratégias de marketing por empresas de e-commerce, já que ao recomendar algo alinhado ao interesse do usuário, aumentam as chances de que ele venha adquirir algum produto.

Sistemas de recomendação também podem ser grandes aliados na área da educação e para isso precisam ser mais bem disseminado para os usuários como estudantes e professores, pois ajudarão significativamente no desempenho de suas atividades, por exemplo, ofertando a um professor materiais que seriam mais recomendados na área que ele leciona e que estejam ligados ao perfil dos alunos.

A partir desta descrição geral, como citado por [14] Souza (2012, p.15), observamos que os sistemas de recomendação se apoiam basicamente em duas coisas para poder funcionar de maneira adequada:

1. Informações provindas de usuários.
2. Um método ou estratégia para determinar se um item deve ser sugerido para um usuário.

Vamos ver algumas estratégias de recomendação e exemplos de como isso funciona:

2.1.1 Baseado em conteúdo

A ideia principal por trás dessa estratégia de recomendação é sugerir itens ou serviços similares a outros que foram adquiridos, utilizados ou avaliados anteriormente de forma positiva por um usuário, por exemplo, um usuário que adquire um artigo ou livro sobre um determinado assunto receberia como sugestão outros itens com assuntos relacionados.

Na Figura 1 é exemplificada a ideia de recomendação baseada em conteúdo. Em um cenário de recomendação de livros, um usuário que lê Harry Potter e a Pedra Filosofal teria, por exemplo, recomendações de livros do gênero fantasia, como Harry Potter e a Câmara Secreta ou O Hobbit.

Figura 1 – Recomendação Baseado em Conteúdo



Fonte: Elaborado pelo autor

Esta estratégia tem como vantagem a necessidade de receber apenas as informações do usuário em questão e sugerir itens com transparência. Em contrapartida não é capaz de fornecer itens que não estejam de acordo com o conteúdo já conhecido pelo usuário, podendo causar recomendações de contexto limitado.

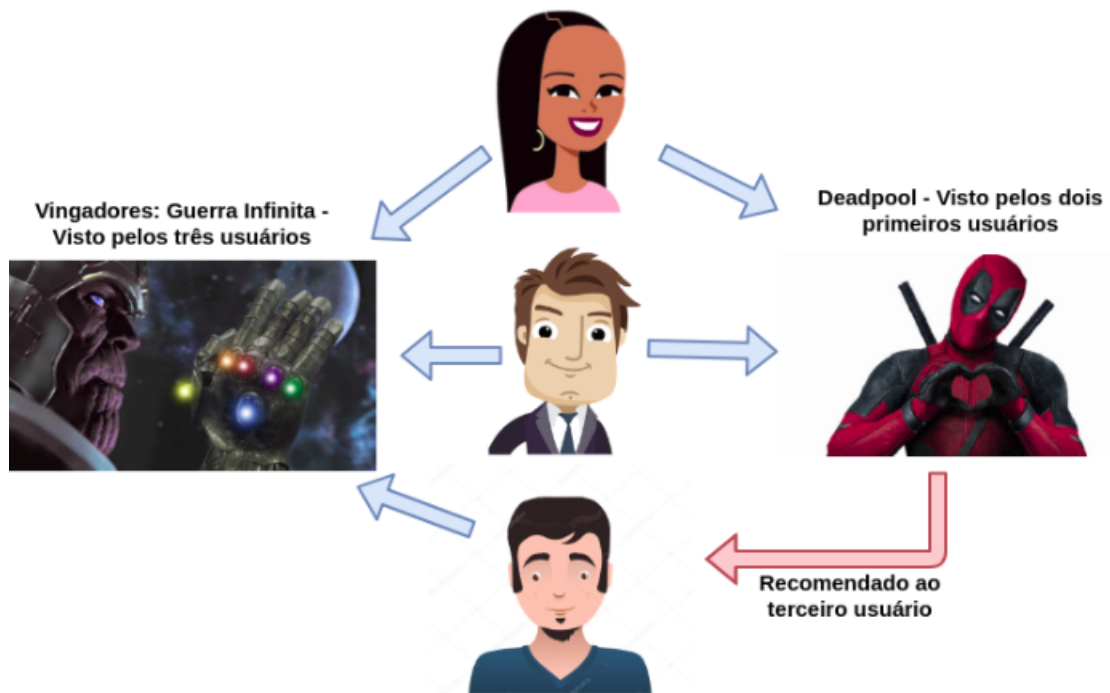
2.1.2 Filtragem Colaborativa

Recomendação baseada em filtragem colaborativa se apoia no julgamento de usuários com interesses em comum ou com perfis parecidos. Para essa estratégia, os itens devem ser avaliados e o sistema de recomendação pode descobrir padrões de comportamento e sugerir automaticamente os itens considerados mais interessantes por usuários com gostos similares.

Uma vantagem importante nesse tipo de abordagem é que a filtragem colaborativa é independente das propriedades específicas do item. Tudo o que se precisa é da identificação dos usuários e dos itens, além de alguma noção de preferência destes usuários. Contudo, como desvantagem, requer informações de usuários com perfis similares para funcionar corretamente.

Na Figura 2 em um ambiente de recomendação de filmes, por exemplo, se três usuários assistirem o filme “Vingadores: Guerra Infinita” e, em seguida, os dois primeiros avaliarem muito bem o filme “Deadpool”, é provável que o terceiro usuário receba este último filme como recomendação.

Figura 2 – Recomendação por Filtragem Colaborativa



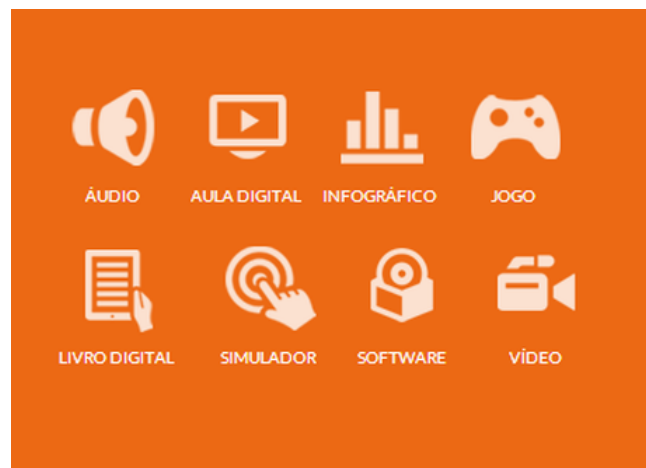
Fonte: Elaborado pelo autor

Pensando nisso, podemos recomendar também Recursos Educacionais?

2.2 Recomendação de recursos educacionais

O envolvimento do contexto educacional nos permite definir o termo *objetos de aprendizagem* que basicamente é uma forma de se apresentar algum conteúdo aos alunos, tanto na modalidade presencial ou a distância, ou seja, é uma unidade de instrução reutilizável [15]. Neste trabalho vamos abordar apenas os objetos de aprendizagem utilizados através de recursos digitais, aproveitando os benefícios que o emprego da tecnologia pode produzir. Sendo assim, um objeto de aprendizagem neste contexto é qualquer entidade digital que possa ser utilizada, reutilizada ou referenciada durante o processo de ensino apoiado pela tecnologia, como exemplo, vídeos, animações, aplicativos para smartphones, textos, figuras, músicas, etc.

Figura 3 – Exemplo de objetos digitais de aprendizagem



Fonte: <https://portal.aprendiz.uol.com.br/>

Assim como no domínio comercial, o ambiente educacional também pode ter estratégias de recomendação personalizadas como aliadas na melhoria do processo de ensino e aprendizagem. No entanto, precisamos compreender que, na maioria dos casos e neste contexto em especial, sistemas de recomendação apresentam algumas limitações. Isso porque a ensinar e aprender não se trata apenas de uma lógica matemática, é necessário considerar o tipo de assunto abordado, por exemplo, a maneira que aprendemos geometria é diferente da maneira que aprendemos idiomas, além disso, segundo Ricci et al. (2011) [9] é preciso considerar o tipo de aprendizagem que se deseja — aprender um novo conceito ou reforçar um conhecimento existente pode exigir tipos diferentes de recursos.

No enfoque deste trabalho, será considerada a ideia de um aluno que quer reforçar um conceito já adquirido.

Para entender com mais detalhes como sistemas de recomendação podem atuar em um contexto educacional, observaremos dois sistemas desenvolvidos por diversos pesquisadores:

2.2.1 LORSys: um Sistema de Recomendação de Objetos de Aprendizagem

Ferreira e Raabe (2010) [13] propõem um modelo de Sistema de Recomendação de objetos de aprendizagem para o ambiente Sophia, uma personalização do Moodle desenvolvida pela Universidade do Vale do Itajaí (UNIVALI). Ao se registrar a navegação do usuário, o ambiente Sophia possibilita a identificação da similaridade entre usuários e, além disso, os metadados que descrevem os objetos de aprendizagem permitem a identificação da similaridade entre conteúdos.

O sistema denominado LorSys (Learning Object Recommender System), gera recomendações por meio das técnicas de filtragem colaborativa e de filtragem baseada em conteúdo. A motivação desse sistema surgiu ao se perceber que muitos alunos não faziam uso dos objetos de aprendizagem da forma esperada pelos seus criadores.

No processo de recomendação as técnicas trabalham de maneira independente. Em relação à filtragem baseada em conteúdo, cabe ao professor definir o grau de relacionamento e afinidade entre os objetos de aprendizagem (“Pré-Requisito”, “Complemento”, “Similar” ou “Sem relação”). Já em relação à filtragem colaborativa, o sistema recomenda o objeto de aprendizagem mais acessado pelos outros usuários, além de recomendar o último objeto de aprendizagem acessado por usuários mais similares. Quando ambas as técnicas encontram o mesmo resultado para recomendação, o Lorys classifica como "dica quente".

Na Figura 4 é ilustrada a interface utilizada para a definição dos graus de relacionamento feitas pelo professor, na parte superior, as informações sobre um determinado objeto de aprendizagem, e, na parte inferior, os outros objetos de aprendizagem aos quais deve ser definido o grau de relacionamento.

Figura 4 – LorSys - Definição de Relacionamentos

Título:	Árvore Binária de Busca		
Assunto:	Estrutura de Dados, Árvore Binária de Busca, Aplicativo		
Descrição:	Permite ao acadêmico visualizar e interagir com a estrutura de dados de uma Árvore Binária de Busca.		
Autor:	Laboratório de Soluções em Software	Organizador:	UNIVALI
Contribuição:	Vinicius H. Ferreira (Produtor do Objeto), Elieser Ademir de Jesus (Desenvolvedor do Aplicativo), Rudimar Luís Scaranto Dazzi (Desenvolvedor do Aplicativo), Fabiane Vavassori (Produtora do Objeto).		
Licença:	(BY-NC-ND)Esta obra está licenciada sob uma Licença Creative Commons (http://creativecommons.org/licenses/by-nc-nd/3.0/deed.pt)		

Título	Relação
Utilizando estrutura de dados Lista: desafio Controle de Lista de Espera Companhia Aérea	Sem relação ▼
Utilizando estrutura de dados Pilha: desafio do Armazém	Pré-Requisito ▼
Preencha as lacunas	Complemento ▼

Relacionar

Fonte:[13]

Na Figura 5 vemos como a recomendação do objeto de aprendizagem é exibida pelo sistema.

Figura 5 – LorSys - Recomendação



Fonte [13]

2.2.2 Recomendação de Objetos de Aprendizagem Empregando Filtragem Colaborativa e Competências

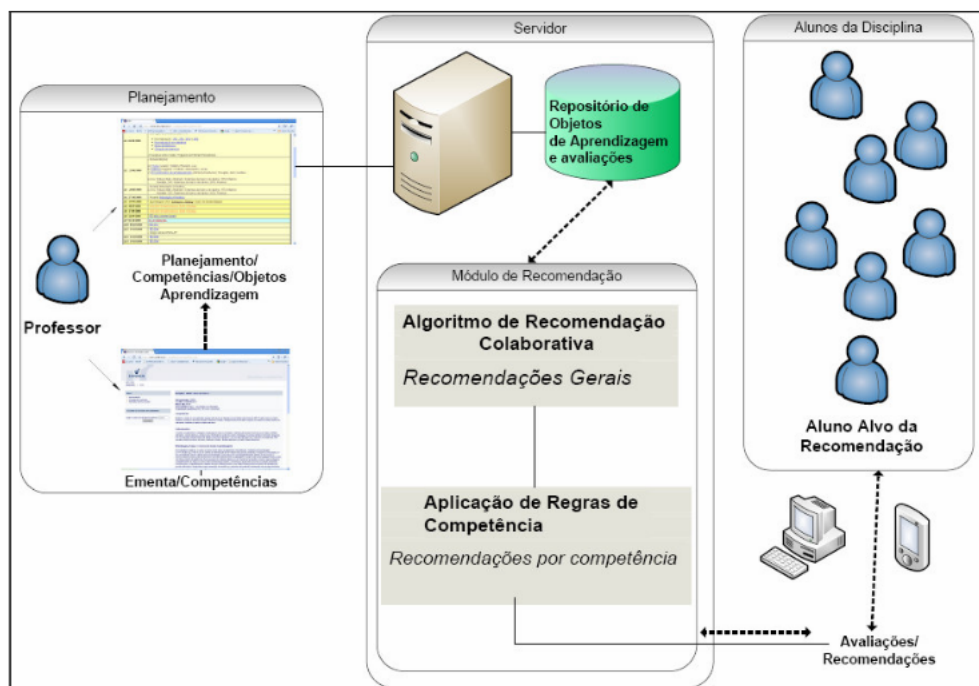
Este é um modelo de sistema de recomendação de objetos de aprendizagem baseado em Filtragem Colaborativa e competências aplicado a alunos de graduação, segundo Cazella et al. 2010 [6]. O modelo permite que alunos recebam a recomendação de objetos de aprendizagem de forma automática conforme interesses do aluno, e de acordo com as competências que devem ser desenvolvidas dentro de um plano de aula.

De um forma geral, o modelo proposto segue os seguintes passos:

- O professor planeja as aulas com base nas competências descritas na ementa da disciplina, disponibilizando os objetos de aprendizagem a serem utilizados para desenvolver as respectivas competências;
- O modelo realiza os cálculos de coeficiente de similaridade (utilizando o coeficiente de Pearson) e de predição;
- O modelo aplica as regras da competência, que filtram os conteúdos recomendados pela predição para permitir ao usuário o desenvolvimento de determinadas competências em um determinado tempo;

Na Figura 6 vemos a esquematização:

Figura 6 – Modelo proposto para o sistema



Fonte:[6]

Os sistemas apresentados ilustram como sistemas de recomendação podem auxiliar no ensino e aprendizagem. Em ambos os sistemas, os públicos alvos foram turmas de alunos de graduação, posteriormente neste trabalho veremos como podemos aplicar esses conceitos para o estudo de caso do projeto Pré-Etec.

3 Sistemas de Tutoria Inteligente

3.1 Fundamento

Basicamente um Sistema Tutorial Inteligente (STI) é qualquer sistema de computador que fornece feedbacks ou instruções diretas personalizadas aos alunos sem a intervenção de um outro ser humano, ou seja, é uma modalidade de software educacional que busca se adaptar de acordo com as necessidades dos estudantes. Esta adaptação é construída com combinação dinâmica de informações sobre o aluno, sobre o domínio dos assuntos estudados e sobre um processo pedagógico [11].

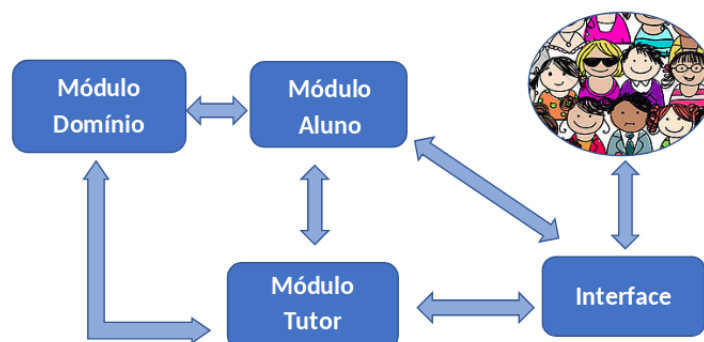
Em geral, de acordo com Raabe (2005) [12] se coloca no STI heurísticas e regras que permitem a construção de um diagnóstico sobre o aluno e a partir dele é possível tomar decisões sobre quais atividades devem ser propostas ou quais feedbacks devem ser apresentados. Em muitos casos, a base de conteúdos de um STI é composta por exercícios e problemas a serem resolvidos pelo aluno. Desta maneira, o STI não é utilizado para ensino de conteúdos novos e sim como elemento de fixação de conteúdos ou reforço de algo que já foi ensinado. Por esta razão, as disciplinas que envolvem a resolução de problemas são mais aderentes a modalidade de STI, uma vez que a estratégia de propor problemas de complexidade crescente acompanhando a trajetória de desenvolvimento do aluno pode ser mais facilmente sistematizada.

E como se estrutura um Sistema de Tutoria Inteligente?

Um STI pode ter mais do que uma arquitetura, neste trabalho vamos abordar a arquitetura clássica. Esta modalidade de arquitetura é formada basicamente de quatro partes fundamentais: o módulo de *interface*, que serve de elo entre o aluno e o sistema; o módulo do *aluno*, que armazena ou modela as características individuais do aluno; o módulo do *domínio*, que possui todo o “conhecimento” presente no tutor e; o módulo do *tutor* recebe as informações criadas pelo módulo do estudante e estabelece uma estratégia pedagógica para o aluno segundo seu perfil.

Na figura 7, podemos ilustrar a arquitetura.

Figura 7 – Arquitetura Clássica - STI



Fonte: Elaborado pelo autor

Na esquematização clássica, o módulo de Domínio contém detalhes sobre o assunto que é ensinado pelo STI e, portanto, contém informações sobre qualquer problema que pode ser proposto neste contexto, assim como solução correta para o problema.

Um problema é apresentado ao aluno através da Interface e em seguida, o aluno entra com sua solução do problema pelo mesmo meio. O módulo Tutor considera esta solução em um conjunto de informações obtidas a partir dos módulos Aluno e Domínio para decidir que tipo de retorno que deve fornecer ao aluno, apresentando então o retorno pela Interface.

Através deste processo, o STI como um todo forma uma base de dados sobre o conhecimento do aluno em determinado assunto e esta informação é atualizada para o módulo do Aluno, a fim de ter um modelo mais preciso em uma próxima avaliação.

3.2 Exemplos de STI's

- SOPHIE [7]

O sistema SOPHIE (Sophisticated Instructor for Eletronics) foi desenvolvido para possibilitar a simulação de um circuito eletrônico ajudando o estudante a depurar falhas hipotéticas.

O estudante recebe um circuito com uma falha e tem que determinar qual a peça defeituosa. O SOPHIE possui um gerador de hipóteses que simulam soluções sugeridas pelos estudantes, testando se eles estão corretos. Se o sistema considerar a hipótese do estudante inconsistente com os fatos já apresentados, ele questiona o estudante e argumenta em contrário.

- Tutor LISP [8]

Este STI conduz o aluno por uma lição de Lisp, que se constitui em uma sequência gradual de problemas a serem trabalhados. O sistema possui um guia de procedimentos para iniciantes ou para alunos mais experientes. A medida que o aluno insere um programa, o sistema vai analisando os símbolos e dando exemplos adequados a cada situação.

A teoria utilizada por este sistema sugere que os erros cometidos pelo aluno devem ser assinalados tão logo forem detectados, a fim de que os alunos não persistam nos mesmos. O sistema inicialmente escolhe o problema a ser trabalhado e depois ele faz comentários acerca das tentativas do aluno.

- MALT [8]

É um STI que opera sobre programas de código de máquina. Na estratégia de ensino, ele gera e apresenta um problema, tecendo comentários à medida em que o aluno fornece uma resposta que faz parte da sequência das instruções em código de máquina. Conforme o aluno vai ganhando experiência, a interrupção por parte do sistema passa a ser menor.

O problema inicialmente gerado pelo sistema é decomposto em vários sub-problemas com seus pontos centrais e suas funções primitivas. O sistema formula uma solução ideal e a

contrapõe com a tentativa do aluno. Conforme o aluno vai se desenvolvendo, o sistema apresenta variantes mais difíceis do sub-problema.

Neste caso, não existe uma representação real do conhecimento do aluno acerca dos conceitos de programação em código de máquina, apenas uma prévia atuação em determinados sub-problemas e funções primitivas.

4 Modelagem

Definidas algumas bases teóricas, vamos detalhar mais especificamente o objetivo deste trabalho:

4.1 Detalhamento

Como descrito anteriormente, o objetivo dessa pesquisa é estudar alguns sistemas de recomendação aplicados na área da educação, além de algoritmos utilizados neste contexto e a partir daí analisar como se poderia aplicar estes algoritmos em um protótipo de STI direcionado ao Projeto Pré-ETEC que é o estudo de caso.

Para melhor definição de como isso pode ser feito, vamos responder algumas questões:

4.1.1 Quais usuários serão considerados?

São considerados usuários, aqueles que podem atuar como estudantes, respondendo as questões propostas e recebendo como retorno o feedback de acordo com o considerado pelo o STI proposto.

4.1.2 Quais conteúdos serão trabalhados

Consideraremos os conteúdos de algumas disciplinas ensinadas no ensino fundamental II, a saber, Matemática, Humanidades (Ciências Humanas - Geografia e História) e Ciências (Ciências da Natureza) que envolvam resolução de problemas de maneira exata. Para avaliação do usuário, os principais objetos de aprendizagem serão apresentado na forma de questões de múltipla escolha, sendo que apenas uma das opções de resposta é correta. O assunto de uma disciplina é tratado de forma independente do assunto de outra. Os assuntos abordados nas questões foram escolhidos de acordo com o estabelecido para o Ensino Fundamental e que seguem a Base Nacional Comum Curricular (BNCC) homologada pelo Ministério da Educação (MEC) [4].

4.1.3 Algoritmos de Recomendação

São aplicadas ambas as técnicas vistas anteriormente, o algoritmo de filtragem por conteúdo e o algoritmo de filtragem colaborativa.

4.1.4 Critérios de similaridade

Como os principais objetos de aprendizagem são questões, a similaridade entre estes objetos é definida por um usuário que cadastra as questões, assim como feito no sistema LORSys. Cada questão tem uma identificação, uma disciplina associada, um título, um nível de dificuldade, uma pontuação, um enunciado e alternativas de resposta para o enunciado, sendo apenas

uma alternativa correta. **As questões de mesma disciplina e de mesmo nível de dificuldade são consideradas similares**, além disso, considera-se como requisito o seguinte: para que uma questão de nível i seja respondida, todas as questões do nível $i-1$ devem já ter sido corretamente solucionadas, considere i um número natural e maior do que 1.

Uma questão pode ser formulada da seguinte maneira:

Tabela 1 – Objeto de aprendizagem: Questão

Id:	111
Disciplina:	Matemática
Título:	Frações
Nível:	0
Pontuação:	2.5
Enunciado:	Qual das frações abaixo é equivalente a $2/5$?
a)	$4/10$
b)	$4/12$
c)	$5/10$
d)	$5/8$
e)	$19/2$

Fonte: Elaborado pelo autor

Além das questões comuns, existem questões especiais e que podem ser recomendadas quando um aluno já respondeu todas as questões de uma determinada disciplina. As questões especiais podem ser questões de olimpíadas ou de exames de anos anteriores, por exemplo, e exigem mais preparo no aluno. Abaixo podemos ver um exemplo de uma questão especial:

Tabela 2 – Objeto de aprendizagem: Questão Especial

Id:	m11e
Disciplina:	Especial
Título:	História
Pontuação:	10
Enunciado:	A República do Suriname localiza-se no norte da América do Sul e faz fronteira com a Guiana, com a Guiana Francesa e com o Brasil. Conta com uma área de aproximadamente 163 820 km ² e uma população de aproximadamente 560 000 habitantes. Sua capital é a cidade de Paramaribo. Sobre esse país é correto afirmar que:
a)	foi a única colônia europeia na América a não adotar mão de obra escrava.
b)	era colônia holandesa e tornou-se um país independente em meados da década de 1970.
c)	sua língua oficial é o inglês, mas o espanhol é bastante utilizado, assim como alguns dialetos locais.
d)	é banhado pelo Oceano Pacífico e seu relevo é marcado pela existência da Cordilheira dos Andes, que corta o país de norte a sul.
e)	seu principal produto de exportação é a banana, o que torna sua economia estável devido às pequenas variações no mercado de commodities

Fonte: Elaborado pelo autor

4.1.5 Estratégias de Recomendação

As regras básicas estabelecidas para recomendação são dadas por:

- Se nenhuma questão acaba de ser respondida:
 - Aguarde o usuário escolher uma disciplina.
 - Dada que a escolha da disciplina foi feita, se existe pelo menos uma questão não respondida nesta disciplina por este estudante, recomende uma questão não respondida de menor nível possível.
 - Se todas as questões da disciplina escolhida foram respondidas corretamente por este estudante, recomende uma questão especial.
 - Se todas as questões cadastradas já foram respondidas corretamente, inclusive as especiais, caso o usuário aceite, recomende a última questão que um usuário com perfil parecido respondeu de forma incorreta.
- Se uma questão acaba de ser respondida:
 - Questione se o usuário gostaria de continuar a responder questões.
 - Caso queira continuar, se a resposta anterior é incorreta, recomende a mesma questão.
 - Se a resposta anterior é correta, se possível recomende uma questão similar da disciplina em questão ainda não respondida pelo usuário.
 - Se não existe tal questão similar não respondida, recomende esta questão de mesma disciplina e do próximo nível.
 - Se não existir questão não respondidas de nível maior, recomende uma questão especial.
 - Se todas as questões cadastradas já foram respondidas corretamente, inclusive as especiais, caso o usuário aceite, recomende a última questão que um usuário com perfil parecido respondeu de forma incorreta.

Os algoritmos de recomendação vão operar em momentos separados:

1. A técnica de recomendação por filtragem de conteúdo é responsável por recomendar questões com base exclusiva no desempenho do usuário.
2. Caso o estudante tenha respondido corretamente todas as questões cadastradas em todas as disciplinas e também as questões especiais, a técnica de recomendação por filtragem colaborativa pode mostrar para este estudante questões que possam estar trazendo dificuldades para os seus colegas.

4.1.6 Perfis de Usuários

Um usuário deve prover algumas informações básicas no momento do cadastro no STI, sendo mandatórias, CPF, nome, idade, email, cidade, ano escolar e escola de origem. **Serão considerados com perfis comuns aqueles alunos que tiverem mesma idade escolar.**

4.2 Projeção em um STI

Adotando a estrutura clássica de um STI, os módulos funcionariam da seguinte maneira:

4.2.1 Módulo Domínio

Armazena todas as questões cadastradas e a alternativa correta para cada questão. Esse cadastro é feito previamente, carregando as questões em tempo de execução e mantendo as questões em memória enquanto a simulação está ativa.

4.2.2 Módulo Aluno

Para que o STI tenha total controle sobre os dados de cada usuário no módulo Aluno, definimos as seguintes variáveis:

- M: armazena a pontuação obtida pelas respostas do usuário nas questões de Matemática.
- H: armazena a pontuação obtida pelas respostas do usuário nas questões de Humanidades.
- C: armazena a pontuação obtida pelas respostas do usuário nas questões de Ciências.
- E: armazena a pontuação obtida pelas respostas do usuário nas questões especiais.

Além disso, são armazenadas as informações de disciplina e nível de cada questão respondida incorretamente.

Quando se é possível recomendar uma questão ainda não respondida de uma determinada disciplina, a pontuação total correspondente à disciplina é atualizada, existindo um acréscimo no valor da pontuação da questão em caso de acerto ou decréscimo de um valor fixo no caso erro. O mesmo se aplica para as questões especiais.

4.2.3 Módulo Tutor

Quando a requisição do estudante pela interface é para responder uma questão, o módulo Tutor analisa três possibilidades:

1. Recomendar uma questão e atualizar os dados referentes a esse aluno de acordo com a resposta da questão. Essa possibilidade é a prioritária.
2. Produzir um feedback com base nos valores de pontuações armazenadas no perfil do aluno para uma determinada disciplina. O feedback é oferecido quando todas as questões da disciplina em questão foram respondidas.

3. Em um cenário em que todas as questões cadastradas foram respondidas, recomendar a última questão que um aluno de perfil similar respondeu de forma incorreta.

4.2.4 Módulo Interface

Basicamente oferece ao usuário três opções:

1. Cadastrar um novo aluno/usuário.
2. Responder uma questão. É validada a identificação do usuário neste caso.
3. Encerrar a execução do STI.

5 Implementação e Testes

5.1 Implementação

O protótipo foi implementado na linguagem de programação Python. Toda a entrada de dados é tratada por um terminal simulado pela função principal e cada resposta fornecida pelo STI é enviada pela saída padrão.

5.1.1 Modulo Aluno

Para representar o módulo Aluno, temos o namespace *aluno.py* e a classe *Aluno* onde são tratadas todas as rotinas e sub-rotinas que inserem, obtém ou atualizam os dados de um ou todos os usuários. Como a criação de um usuário se dá em tempo de execução, para fins de testes é necessário o cadastramento de um novo usuário.

A classe *Aluno* é detalhada como:

```
class Aluno:
    def __init__(self, cpf, passwd, nome, idade, email, cidade, age, escola):
        self.cpf = cpf          #identificação única do usuário
        self.senha = passwd     #senha de acesso do usuário
        self.nome = nome        #nome
        self.idade = int(idade) #idade
        self.email = email      #email
        self.cidade = cidade    #cidade
        self.age = int(age)     #idade escolar
        self.escola = escola    #escola de origem
        self.mdone = []         #id's de questões de matemática respondidas corretamente
        self.hdone = []         #id's de questões de humanidades respondidas corretamente
        self.cdone = []         #id's de questões de ciências respondidas corretamente
        self.especiais = []     #id's de questões especiais respondidas corretamente
        self.merros = [0,0,0]   #num de questões de matemática erradas em cada nível
        self.herros = [0,0,0]   #num de questões de humanidades erradas em cada nível
        self.cerros = [0,0,0]   #num de questões de ciências erradas em cada nível
        self.ultimo = []       #id da última questão respondida de forma errada
        self.M = 0              #Pontuação acumulada das questões de matemática
        self.H = 0              #Pontuação acumulada das questões de humanidades
        self.C = 0              #Pontuação acumulada das questões de ciências
        self.E = 0              #Pontuação acumulada das questões especiais
```

5.1.2 Modulo Domínio

Para representar o módulo Domínio, temos o namespace *dominio.py*, a classe *Questao* e todas os objetos de questões armazenados em listas. Para esse armazenamento foram selecionadas 50 questões ao todo, sendo 15 questões de Matemática, 15 questões de Humanidades, 15 questões de Ciências e 5 questões especiais.

A classe *Questão* é detalhada como:

```
class Questao:
    def __init__(self, disciplina, titulo, nivel, pont, enunciado, a, b, c, d, e, r):
        self.disciplina = disciplina      #disciplina da questão
        self.titulo = titulo              #título
        self.nivel = nivel                #nível
        self.pont = pont                  #pontuação
        self.enunciado = enunciado        #enunciado
        self.a = a                       #alternativa a
        self.b = b                       #alternativa b
        self.c = c                       #alternativa c
        self.d = d                       #alternativa d
        self.e = e                       #alternativa e
        self.correct = r                 #alternativa correta
        self.set_id(disciplina,nivel)     #identificação da questão
```

Cada uma das disciplina trabalhadas possuem 3 níveis de questões, cada nível reserva 5 questões.

Os assuntos abordados em cada questão de Matemática foram os seguintes:

Tabela 3 – Assuntos abordados nas questões de Matemática

Questão	Assunto	Título	Nível
1	Matemática	Divisores e Múltiplos	0
2	Matemática	Divisores e Múltiplos	0
3	Matemática	Divisores e Múltiplos	0
4	Matemática	Divisores e Múltiplos	0
5	Matemática	Frações	0
6	Matemática	Equação do 1 grau	1
7	Matemática	Inequação	1
8	Matemática	Equação do 1 grau	1
9	Matemática	Equação do 1 grau	1
10	Matemática	Inequação	1
11	Matemática	Geometria	2
12	Matemática	Geometria	2
13	Matemática	Geometria	2
14	Matemática	Probabilidade	2
15	Matemática	Probabilidade	2

Fonte: Elaborado pelo autor

Os assuntos abordados em cada questão de Humanidades foram os seguintes:

Tabela 4 – Assuntos abordados nas questões de Humanidades

Questão	Assunto	Título	Nível
1	Geografia	Distribuição da população mundial e deslocamentos populacionais	0
2	Geografia	Distribuição da população mundial e deslocamentos populacionais	0
3	Geografia	Distribuição da população mundial e deslocamentos populacionais	0
4	Geografia	Distribuição da população mundial e deslocamentos populacionais	0
5	Geografia	Distribuição da população mundial e deslocamentos populacionais	0
6	Geografia	Diversidade e dinâmica da população mundial e local	1
7	Geografia	Diversidade e dinâmica da população mundial e local	1
8	História	Brasil: Primeiro Reinado	1
9	História	Brasil: Primeiro Reinado	1
10	História	Brasil: Primeiro Reinado	1
11	História	Brasil: Segundo Reinado	2
12	História	Brasil: Segundo Reinado	2
13	História	Brasil: Segundo Reinado	2
14	História	Partilha da África	2
15	História	Partilha da África	2

Fonte: Elaborado pelo autor

Os assuntos abordados em cada questão de Ciências foram os seguintes:

Tabela 5 – Assuntos abordados nas questões de Ciências

Questão	Assunto	Título	Nível
1	Ciências	Fontes de Energia	0
2	Ciências	Fontes de Energia	0
3	Ciências	Fontes de Energia	0
4	Ciências	Fontes de Energia	0
5	Ciências	Fontes de Energia	0
6	Ciências	Cálculo de consumo de energia elétrica	1
7	Ciências	Cálculo de consumo de energia elétrica	1
8	Ciências	Circuito Elétrico	1
9	Ciências	Circuito Elétrico	1
10	Ciências	Circuito Elétrico	1
11	Ciências	Sistema Sol, Terra e Lua	2
12	Ciências	Sistema Sol, Terra e Lua	2
13	Ciências	Sistema Sol, Terra e Lua	2
14	Ciências	Alterações ambientais por ação humana	2
15	Ciências	Alterações ambientais por ação humana	2

Fonte: Elaborado pelo autor

As questões foram escolhidas em alguns sites de conteúdo educacional [1] [2] [3].

As questões enquadradas como questões especiais foram retiradas do último exame da ETEC aplicado em dez/2018 [5]. Cada uma das 5 questões abordou uma disciplina propostas anteriormente.

Tabela 6 – Assuntos abordados nas questões especiais

Questão	Tipo	Disciplina
1	Especial	Matemática
2	Especial	Matemática
3	Especial	História
4	Especial	Geografia
5	Especial	Ciências

Fonte: Elaborado pelo autor

Todas as questões são armazenadas em quatro listas, cada lista armazena uma disciplina (Matemática, Humanidades, Ciências e Especiais). Assim que o protótipo do STI é iniciado estas listas são processadas e cada uma delas é transformada em uma matriz. A posição $[i][j]$ da matriz de uma disciplina armazena a questão j de nível i do conteúdo em questão.

5.1.3 Modulo Tutor

Para representar o módulo Tutor, temos o namespace *tutor.py* e neste módulo são administrados os algoritmos de recomendação. As principais rotinas são:

- `recomenda_conteudo(aluno,dis):`

Para uma disciplina *dis* o algoritmo acessa a matriz da disciplina, buscando a questão de menor nível não respondida por este usuário *aluno*. Se uma questão nestas condições for encontrada, a questão é apresentada para o usuário e os dados do usuário, conforme mostrado na classe *Aluno* são atualizados com base na resposta. Caso o usuário já tenha respondido todas as questões da disciplina, o algoritmo busca recomendar uma questão especial. No cenário em que todas as questões especiais já foram respondidas por este usuário, um valor default é retornado.

- `recomenda_colaborativa(aluno):`

Para um usuário *aluno*, é feita a busca dentre os demais usuários um que tenha perfil similar. Se possível retorna a última questão respondida de forma incorreta por este usuário similar.

- `feedback(aluno,dis):`

Para um usuário *aluno*, a rotina busca no seu perfil as informações armazenadas sobre a disciplina *dis*. Retorna uma mensagem de feedback com base no número de erros cometidos nas questões de cada nível da disciplina, assim como a pontuação geral da disciplina. Se o número de erros em um determinado nível comparado com o número de questões no nível for inferior a 30% o desempenho no nível é considerado bom, se esse valor for maior que ou igual a 30% porém inferior a 50% o desempenho é considerado regular e maior do que 50% é considerado ruim.

5.1.4 Modulo Interface

Para representar o módulo Interface, temos o namespace *interface.py* que é complementado com o módulo Tutor. O arquivo *interface.py* cria uma espécie de terminal que é exibido para o usuário. A mensagem default é a seguinte:

```
'Você tem as seguintes opções:'  
'[r]: responder questões'  
'[c]: cadastrar um novo aluno'  
'[s]: sair'
```

A opção [c] permite que um novo usuário seja cadastrado caso ele preencha todas as informações requeridas pela classe *Aluno*.

A opção [r] permite um usuário já cadastrado seja redirecionado ao sistema recomendador de questões caso informado corretamente o seu CPF e senha.

A opção [s] encerra a execução do protótipo do STI.

se a opção [r] for escolhida e a identificação do usuário validada com sucesso, o usuário poderá escolher a disciplina a ser respondida e o módulo Tutor então se encarrega de recomendar questões ou feedbacks.

```
'Escolha uma Disciplina:'  
'[1]: Questões de Matemática'  
'[2]: Questões de Humanidades'  
'[3]: Questões de Ciências'
```

Sempre que uma questão é respondida o usuário tem a opção de continuar a responder ou parar, assim como quando possível obter ou não feedback.

As mensagens seguem o seguinte padrão:

```
'O que fazer em seguida?'  
'[1]: continuar'  
'[2]: parar'  
  
'Você respondeu uma questão especial'  
'[1]: Obter Feedback'  
'[2]: Não Obter Feedback'
```

5.2 Testes

Os testes foram classificados em dois tipos:

1. Corretude dos dados.
2. Qualidade das questões e dos Feedbacks.

O primeiro tipo de testes foi realizado por quem escreveu o código. O objetivo destes testes é verificar se as informações armazenadas ao decorrer da execução do programa estavam corretas e coerentes. Para este teste verificou-se que:

- Cada lista de uma determinada disciplina monta corretamente cada questão a ser cadastrada.
- As matrizes de questões de cada disciplina foram preenchidas de forma correta.
- Um usuário que pode responder questões está devidamente registrado.
- O algoritmo de recomendação retorna uma questão do nível i apenas quando as questões de nível $i-1$ foram respondidas corretamente, i número natural maior do que 1.
- Os dados de um usuário que acaba de responder uma questão foram atualizados corretamente.
- As informações fornecidas pelo feedback foram coerentes.
- A vontade do usuário de parar ou continuar a responder questões é respeitada.

Cada resultado de insucesso foi reparado e o teste refeito até obter sucesso no resultado.

O segundo tipo de testes foi realizado por um grupo pequeno de professores voluntários do projeto Pré-ETEC. O objetivo deste teste é qualitativo, qualificando os seguintes tópicos como **Bom**, **Regular** ou **Ruim**:

1. A clareza das questões.
2. A forma com que as questões são apresentadas.
3. A usabilidade do protótipo.
4. A qualidade dos Feedbacks apresentados.

Considerando ser apenas um protótipo, a avaliação fornecida pelo grupo foi a seguinte:

1. A clareza das questões é Boa.
2. A forma com que as questões são apresentadas é Regular.

3. A usabilidade do protótipo é Regular.
4. A qualidade dos Feedbacks apresentados é Ruim (era esperado um feedback mais detalhado).

6 Conclusão

A grande oferta de informações, de ferramentas e recursos digitais educacionais provida pela internet, trouxe a difícil tarefa para um estudante ou professor de filtrar e selecionar conteúdos que mais se aproveita nas suas intenções de estudo ou trabalho. Na abordagem deste problema, não apenas exclusivo para o contexto educacional, surgiram os Sistemas de Recomendação personalizados e com enfoque em educação, surgiram também os Sistemas de Tutoria Inteligente (STI's). Esses tipos de sistemas tem grande potencial de auxílio no desempenho de atividades de usuários de recursos educacionais, como professores e alunos.

Tomando como estudo de caso o projeto Pré-ETEC, foi projetado um protótipo de STI utilizando Sistemas de Recomendação. Este protótipo se baseou também nos sistemas educacionais LORSys e Recomendação de Objetos de Aprendizagem Empregando Filtragem Colaborativa e Competências.

No desenvolvimento do protótipo foi possível observar que a ideia geral de tutoria inteligente envolve também conceitos de usabilidade, aparência, navegabilidade e uma estratégia pedagógica assertiva para que os usuários utilizem o STI de forma proveitosa.

Com o aprendizado adquirido fica clara a importância do tempo dedicado ao planejamento e desenho do protótipo, principalmente porque esta pesquisa foi direcionada a um grupo de pessoas. No início do projeto, a ideia era ter um protótipo web apto para usabilidade do usuário final, mas no decorrer do desenvolvimento se fez necessário refazer o planejamento algumas vezes por conta de mudanças no escopo.

Este protótipo tem muito espaço para evolução. Durante os testes foram recebidas uma série de feedbacks e sugestões para a continuidade do desenvolvimento, em especial para a qualidade dos feedbacks. Entre estas sugestões estão:

- Migrar o protótipo para uma plataforma web
- Incluir todas as disciplinas apresentadas no ensino fundamental II
- Incluir no protótipo a opção de simulados de provas e vídeo de aulas online.
- Opções de vínculo com perfis de redes sociais (Facebook e Instagram).
- Remodelagem do sistema de feedbacks fornecidos pelo protótipo para obter uma avaliação mais detalhada.

Referências

- [1] Base de questões. <https://exercicios.brasilecola.uol.com.br/>.
- [2] Base de questões. <http://tudodeconcursossevestibulares.blogspot.com/2012/12/>.
- [3] Base de questões. <https://rachacuca.com.br/quiz/>.
- [4] A base nacional comum curricular. http://basenacionalcomum.mec.gov.br/wp-content/uploads/2018/12/BNCC_19dez2018_site.pdf.
- [5] Questões do exame etec - 12/2018. http://fatweb.s3.amazonaws.com/vestibulinhoetec/gabarito/201917532/Prova_1modulo.pdf.
- [6] Sílvio César Cazella; Eliseo Berni Reategui; Munique Machado; Jorge Luis V. Barbosa. Recomendação de objetos de aprendizagem empregando filtragem colaborativa e competências. pages 1–10, 2009. <http://www.br-ie.org/pub/index.php/sbie/article/view/1158/1061>.
- [7] Johan de Kleer; John Seely Brown. Model-based diagnosis in sophie iii. pages 1 – 2. <http://www.dekleer.org/Publications/Model%20Based%20Diagnosis%20in%20SOPHIE%20III.pdf>.
- [8] G. H. Boulay; A. J. Woods; D. Teather; A. B. Teather; M. K. Wills; DJohan de Kleer Plummer; John Seely Brown. Towards an advisor for mri. pages 245 – 251.
- [9] Francesco Ricci; Lior Rokach; Bracha Shapira; Paul B. Kantor. *Recommender Systems Handbook*. Springer, Boston, MA, 2011. Bibliografia: p. 387–415.
- [10] Evandro Costa; Janderson Aguiar; Jonathas Magalhães. Sistemas de recomendação de recursos educacionais: conceitos, técnicas e aplicações. *Anais da Jornada de Atualização em Informática na Educação*, pages 57–69, 2013. <http://br-ie.org/pub/index.php/pie/article/view/2589/2245>.
- [11] Roger Nkambou; Jacqueline Bourdeau; Riichiro Mizoguch. *Advances in Intelligent Tutoring Systems*. Springer, 2010. Bibliografia: p. 1 – 12.
- [12] André Luís Alice Raabe. Uma proposta de arquitetura de sistema tutor inteligente baseada na teoria das experiências de aprendizagem mediadas. pages 14 – 19, 2005. <https://lume.ufrgs.br/handle/10183/12867>.
- [13] Vinicius Hartmann Ferreira; André Luís Alice Raabe. Lorsys - um sistema de recomendação de objetos de aprendizagem scorm. pages 1–9, 2010. <https://seer.ufrgs.br/renote/article/view/15223/8986>.

- [14] Bruno Souza. Modelos de fatoraão matricial para recomendaão de vdeos. *Pontifcia Universidade Catlica do Rio de Janeiro - PUC-Rio*, pages 15–26, 2012. https://www.maxwell.vrac.puc-rio.br/19273/19273_3.PDF.
- [15] David Wiley. Learning objects need instructional design theory. pages 115–126, 2002.