

FERRAMENTA DE SUPORTE NA APRENDIZAGEM DE PROGRAMAÇÃO

Francisco Leocassio da Silva*

Adriana Mara Guimarães de Farias**

RESUMO

O presente trabalho tem como propósito desenvolver um sistema computacional para submissão e correção automática de códigos. Compreende-se que as disciplinas de programação são essenciais para curso relacionados a área da computação e que estudantes enfrentam dificuldades na aprendizagem dessas disciplinas. Partindo disso foi realizado um estudo buscando entender quais os obstáculos que esses alunos encontram. Foram implementados e analisados questionamentos para 14 alunos concluintes do curso de Licenciatura em computação da Universidade Federal Rural do Semi-Árido – UFERSA. A pesquisa incluiu questões sobre contratempos encontrados por eles no período em que cursavam essas disciplinas. Identificou-se que muitos desses alunos dedicam-se pouco tempo para resolver problemas de programação, que tem dificuldade em elevar o nível de programação e que foram reprovados em disciplinas de programação. A criação dessa aplicação fará com que os estudantes possam se dedicar a resolução de problemas tendo um *feedback* rápido, aumentando o seu nível de programação e diminuindo o índice de reprovação.

Palavras-chave: Sistema. Aprendizagem. Obstáculos. Disciplinas. Programação.

ABSTRACT

The present work has the purpose of developing a computational system for automatic code submission and correction. It is understood that programming disciplines are essential for course related to the area of computing and that students face difficulties in learning these subjects. From this, a study was carried out in order to understand the obstacles that these students encounter. Questionnaires were implemented and analyzed for 14 students graduating from the Licentiate degree course in Computing at the Federal Rural Semi-Arid University – UFERSA. The research included questions about setbacks encountered by them in the period in which they attended these subjects. It was identified that many of these students spend little time to solve programming problems that have difficulty raising the level of programming and

* Graduando de Licenciatura em Computação – Universidade Federal Rural do Semi-Árido. E-mail: leocassiosilva1234@hotmail.com

** Professora Assistente Universidade Federal Rural do Semi-Árido/Angicos. E-mail: adrianaguimaraes@ufersa.edu.br

have been disapproved in programming disciplines. Creating this application will allow students to focus on problem solving by having quick feedback, increasing their programming level, and decreasing the failure rate.

Keywords: System. Learning. Obstacles. Subjects. Programming.

1 INTRODUÇÃO

Os acadêmicos de cursos de computação, principalmente em seus períodos iniciais são submetidos a prática e repetição na realização de tarefas, que tem como objetivo a construção de inúmeros algoritmos, visando com que esses estudantes possam aprender a programar.

Sabe-se que as disciplinas de programação são de fundamental importância para cursos de computação pelo simples fato dessas matérias serem a base desses cursos. No entanto, alunos de computação sentem dificuldades de absorver o conteúdo, identificar soluções e convertê-las para algoritmos o que compromete a aprendizagem.

Nesse cenário, foram desenvolvidas diversas aplicações online, com propósito de auxiliar ao ensino-aprendizado. Um ambiente muito comum em competições de programação são os chamados Juízes Online [Kurnia et al. 2001], que tem como principal função a avaliação de códigos-fonte. A avaliação feita por esses juízes gera respostas como: certo, errado, saída mal formatada, erro de compilação, erro em tempo de execução, dentre outras [Campos e Ferreira 2004].

O estudo de programação demanda com que teoria e prática sejam indissociáveis, entretanto é na parte prática que o educando tem maior chance de aprender. Porém, encontram dificuldades de transição entre o período teórico e o prático, ou seja, no momento em que são desafiados a encontrar elucidaciones para algoritmos. Além disso, quando não obtém sucesso na resolução, muitos estudantes buscam em outras fontes por respostas para os problemas solicitados por seus professores.

É necessário frisar que a modalidade de educação presencial requer que docentes e discentes estejam em um mesmo local e ao mesmo tempo, entretanto a Educação a Distância (EaD) utiliza de ferramentas tecnológicas para mediar a comunicação, possibilitando com que educandos possam estar em lugares distintos de seus educadores.

Em geral, quer seja na modalidade de Educação a Distância (EaD) ou na modalidade de educação presencial, os sistemas existentes atualmente fornecem um ambiente que permite ao aluno criar seus algoritmos e codificá-los em alguma linguagem de programação, porém, para o professor torna-se difícil fornecer um feedback rápido e disponibilizar as devidas correções a seus alunos (CHAVES, 2013, p. 2).

Dessa forma, os professores precisam de alguma solução que lhe ampare e melhore o rendimento dos alunos, tanto em tarefas realizada em sala de aula como fora da classe. Em decorrência disso, nota-se que os obstáculos existentes na aquisição de conhecimento dos conceitos básicos relacionados a disciplinas de programação fazem com que os estudantes se sintam desmotivados e não atinjam desempenho satisfatório.

Embora componham a base de qualquer curso de Computação, um dos grandes problemas enfrentados nesses cursos recai justamente no ensino/aprendizagem das disciplinas ligadas à Programação. Grande parte dos estudantes apresenta dificuldade em entender os conceitos iniciais, aumentando, cada vez mais, o índice de evasão e reprovação nas disciplinas desta área (PRIETCH e PAZETO, 2010).

Por todos esses aspectos, o presente artigo tem como finalidade de apresentar a criação de um software para apoiar os professores no ensino de programação e os alunos na aprendizagem dessas técnicas.

2 METODOLOGIA

A metodologia usada nesse trabalho foi o estudo bibliográfico e duas pesquisas quantitativas. O estudo identificou, a existência de inúmeros sistemas voltado para o aprendizado de programação, que prezam por automatizar a submissão e correção de algoritmos. Ainda assim, estudantes encontram contratempo para receber um *feedback* rápido e os docentes em identificar as dificuldades dos alunos, seja na educação presencial ou na educação a distância (EaD).

No que tange a primeira pesquisa quantitativa, ela possibilitou perceber a importância das disciplinas de programação/algoritmos nos cursos na área de computação. Além disso, proporcionou o discernimento das principais dificuldades no ensino e aprendizagem de programação. Ademais, o questionário constituiu de nove indagações, sendo que uma questão aberta e oito fechadas. A aplicação do questionário aconteceu por meio do aplicativo Google Formulários, por causa que os discentes moram em cidades distintas.

Por outro lado, a segunda pesquisa quantitativa, permitiu identificar uma enorme aceitação da ideia de desenvolvimento de um sistema *online*, que possibilite além da submissão e correção automática, uma maior interação entre alunos e professores. Convém lembrar que o questionário foi construído com nove perguntas, sendo uma questão discursiva e oito questões alternativas. Os questionamentos foram realizados por meio da ferramenta Google Formulários.

Uma pesquisa para Gil (2007, p.17), é determinada como um

(...) procedimento racional e sistemático que tem como objetivo proporcionar respostas aos problemas que são propostos. A pesquisa desenvolve-se por um processo constituído de várias fases, desde a formulação do problema até a apresentação e discussão dos resultados.

O curso de Licenciatura em Computação a distância, foi escolhido para realização da pesquisa. A investigação constitui-se na elaboração de dois questionários e aplicação com os alunos. Os estudantes selecionados foram aqueles que já cursaram matérias relacionadas a algoritmos/programação.

Portanto, optou-se por apresentar as informações colhidas no primeiro formulário por meio de gráficos. Dessa forma, cada pergunta representa um gráfico e os dados são mostrado em porcentagem.

3 REFERENCIAL TEÓRICO

Pode-se afirmar que em razão das complicações encontradas na aprendizagem e também no ensino de disciplinas de algoritmos e programação, seja em cursos ofertado na modalidade presencial e/ou a distância, nos últimos tempos intensificou-se as pesquisas na busca por reduzir esses obstáculos.

Ao examinar alguns trabalhos, verificou-se que para minimizar esses problemas foram estudadas e desenvolvidas ferramentas web, com o propósito de auxiliar alunos e professores em atividades práticas de programação. Essas aplicações utilizam ou se baseiam em Juízes Online, que são aplicações que compilam, efetuam e avaliam códigos fontes. Esse tipo de sistema é muito aplicado em disputas de programação. Porém são adicionadas funcionalidades didáticas que facilite a interação aluno e professor, *feedback* rápido e personalizado.

De acordo Chaves (2013), apesar da utilização desses instrumentos, entretanto os professores encontram dificuldade no processo de ensino, como na avaliação de atividades em um curto espaço de tempo, de uma turma numerosa. Mesmo essas aplicações possibilitando com que os discentes desenvolvam algoritmos e códigos em algum tipo de linguagem de programação, ainda assim, o docente encontra obstáculos para proporcionar aos seus alunos *feedback* e correção rápida de exercícios por ele proposto.

Pode-se mencionar, por exemplo, o BOCA que é um sistema utilizado no apoio de competições de programação. De acordo com Campos e Ferreira (2004) Boca é um aparato que torna automático a submissão e entrega de questões de programação para que os professores possam fazer a correção. Ainda convém lembrar a possibilidade de tirar dúvidas por meio de uma ferramenta semelhante a um bate-papo/fórum.

Além disso, existe o JOnline uma aplicação que enriqueceu os juízes online com funcionalidades educativas para que pudessem ser melhor empregada na educação com proposito de ajudar os estudantes na aprendizagem.

O JOnline possui as seguintes funcionalidades didáticas: a apresentação de dicas (em língua portuguesa) para corrigir erros de compilação do código-fonte enviado, apresentação de casos de testes que geram resultados errados, organização de problemas por assunto e grau de dificuldade e votação do nível de dificuldade dos problemas pelos usuários, além da programação colaborativa (SANTOS, 2011).

Um outro ambiente que foi criado com o intuito de diminuir os obstáculos encontrados por discentes de disciplinas de algoritmos, complementando assuntos concebidos em sala de aula e que também serve para ajudar os educadores em suas tarefas de submissão e orientação individual é denominado de ALICE (*Algorithm Learning Internet-based Computer Environment*).

O sistema ALICE é fundamentado em Assistentes Inteligentes de Ensino (ITA – *Intelligent Teaching Assistant System*). Segundo Lesta e Yalef (2002), um assistente inteligente tem a capacidade de auxiliar tanto os alunos como também os professores em suas tarefas.

A ferramenta ALICE possibilita a submissão de algoritmos, questões objetivas e subjetivas, realiza uma análise previa da resposta e do desempenho do estudante. Além disto, permite com que o professor possa compartilhar diferentes matérias como apostilas, slides e códigos fontes. Por meio de mensagens ainda possibilita com que haja uma maior interação entre estudante e professor.

Além das aplicações anteriormente citadas, ainda existe o Inspetor Online que é um sistema online que automatiza a submissão e correção de exercícios práticos de programação e tem como base os Juízes online. Assim, seu uso faz com que docentes não realizem operações manuais, como submissão e revisão, acarretando no prazo para que o aluno seja informado do resultado.

Por outro lado, com o inspetor online, o professor faz o cadastro do problema e do caso de teste e o discente é responsável pelo problema. Uma característica importante é que o estudante pode fazer a escolha da linguagem de programação que ele deseja usar, claro que levando em consideração a linguagem solicitada por seu docente.

No entanto as duas principais funcionalidades do Inspetor online é fornecer para os alunos um feedback rápido e ajudar o professor na aplicação e correção de exercícios e provas. Além do mais, os alunos podem ter acesso ao gabarito dos exercícios propostos, o que é importante, já que os educandos podem comparar ambos os códigos, identificando assim suas maiores dificuldades.

Em relação a mais um sistema usado na web, para ajudar no estudo de disciplinas de programação, por meio de atividades realizadas no âmbito escolar e/ou fora do ambiente escolar temos o *Feeper*, que disponibiliza atividades práticas para que os alunos possam responder, por meio de um editor de código, e a validação desse código fonte acontece por intermédio dos Juízes online.

Convém lembrar que o *Feeper* viabiliza que os alunos tirem suas dúvidas por meio de comentário nos próprios códigos fontes assim como os estudantes os professores podem responde-los por meio de comentários. Por essas características ao solicitar a validação o estudante terá uma resposta em tempo real.

Dado o exposto, entende-se que, grande maioria dessas aplicações tem como principais características a automação da submissão e correção de códigos. Em consequência disso, nota-se que são de grande importância para facilitar as atividades dos docentes de submeter e corrigir. Por outro lado, auxilia os estudantes nas atividades práticas de programação.

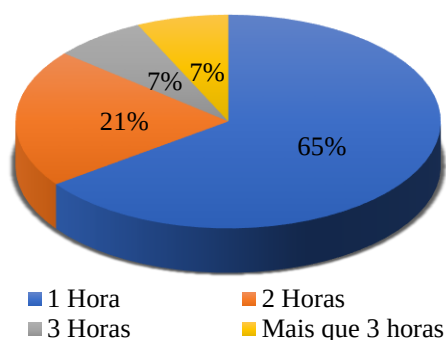
4 RESULTADO DA PESQUISA

Com a finalidade de verificar, discernir os aspectos e confirmar a dificuldade de aprendizagem enfrentada por estudantes nas disciplinas de programação e/ou algoritmo, decidiu-se executar uma pesquisa por meio de um questionário de nove questões, com catorze estudantes do oitavo período do curso de Licenciatura em Computação da Universidade Federal Rural do Semi-Árido – UFERSA, já que os mesmos tiveram contato com disciplinas de programação/algoritmos.

No entanto, o propósito da fase de obtenção dos dados levou em consideração identificar quanto tempo é gasto para resolução de problemas de computação, quais são as estratégias para solucionar esses problemas, quais ações são realizadas quando não obtém sucesso na solução, de qual forma tiram dúvidas, qual meio usado para se comunicar com os docentes, como identificaria o seu raciocínio lógico, quais as dificuldades nas disciplinas de programação, se foi reprovado em disciplinas de programação e quais as disciplinas na qual reprovou.

Como pode ser visto na Figura 1. Distribuição do percentual de respostas ao questionamento 01 da análise de perfil do aluno de Licenciatura em Computação à Distância. Dos estudantes que responderam o questionário, 65% dedicam-se diariamente uma hora para resolver problemas de programação, 21% dedicam-se duas horas, 7% três horas e 7% mais que três horas, o que provavelmente ocasiona dificuldade na aprendizagem, isso porque essas disciplinas requerem mais tempo de dedicação.

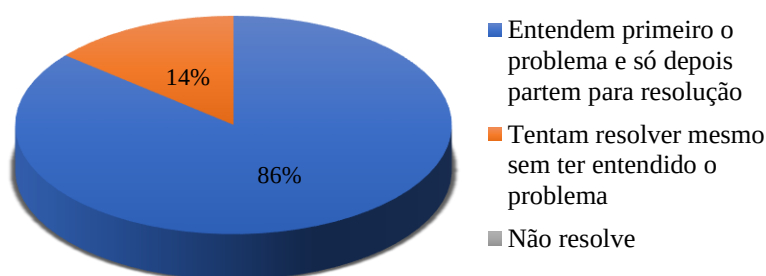
Figura 1 - Quanto tempo, diariamente, você se dedica para resolver problemas de programação?



Fonte: Autoria própria (2017).

Verificou-se na Figura 2, nas estratégias usadas para resolver problemas de programação, 86% dos alunos entendem primeiro o problema e só depois seguem para resolução e 14% tentam resolver mesmo sem ter entendido o problema quando não há um entendimento a resolução e a aprendizagem ficam comprometidas.

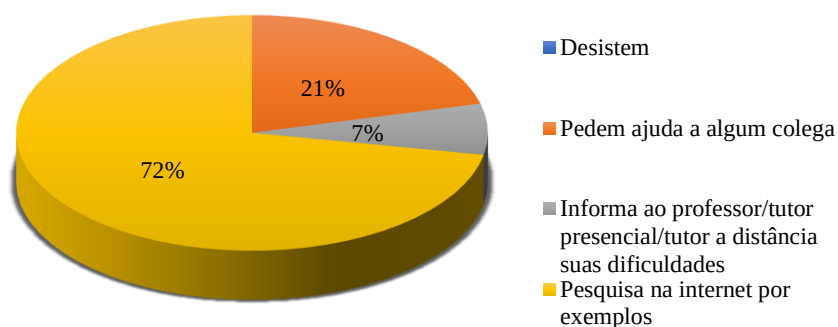
Figura 2 - Quais suas estratégias para resolver problemas de programação?



Fonte: Autoria própria (2017).

Entretendo, no que se refere em quais as atitudes que são realizadas quando não obtém sucesso na solução de um problema, na Figura 3 pode-se observar, que 72% pesquisam na internet por exemplos, enquanto que 21% pede ajuda para algum colega e 7% informa ao professor/tutor presencial/tutor a distância suas dificuldades. Dessa forma, quando o professor/tutor não tem o conhecimento de quais as dificuldades dos estudantes, ficando mais difícil que o processo de ensino-aprendizagem possa acontecer. Muitas vezes ao pesquisar na internet os estudantes encontram soluções já prontas para seus problemas.

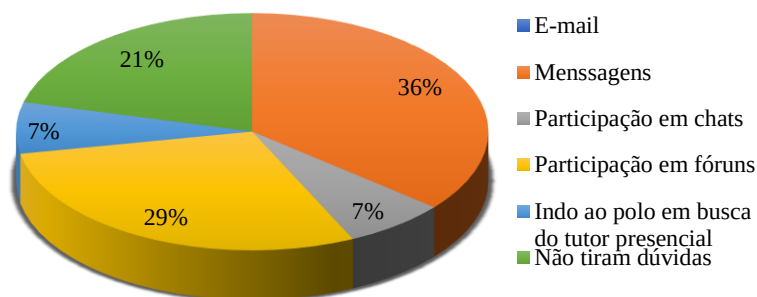
Figura 3 - Quais atitudes você realiza quando não obtém sucesso na solução de um problema?



Fonte: Autoria própria (2017).

No que concerne de qual a maneira de tirar dúvidas, como pode ser visto na Figura 4, 36% preferem tirar dúvidas por mensagens, 29% por fóruns, 7% indo ao polo em busca do tutor presencial, 7% participam de chats e 21% não tiram dúvidas, o que provavelmente pode afetar a aprendizagem e o conhecimento dos estudantes, isso porque o aluno pode não ter um feedback rápido em relação a uma dúvida que ele tenha.

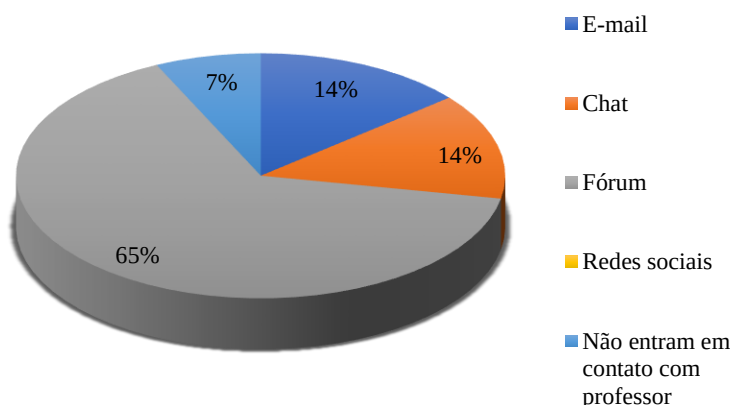
Figura 4 - De qual maneira você tira dúvidas ?



Fonte: Autoria própria (2017).

Contudo quanto ao meio utilizado para se comunicar com os professores sob um determinado conteúdo, como pode ser visto na Figura 5, 65% usam o fórum para comunicar-se com os docentes, enquanto que 14% fazem o uso de chat, 14% se comunicam por e-mail e apenas 7% não entram em contato com o professor. Quanto mais os estudantes entram em comunicação com os docentes, maior possibilidade da criação e compartilhamento de conhecimento entre ambas as partes.

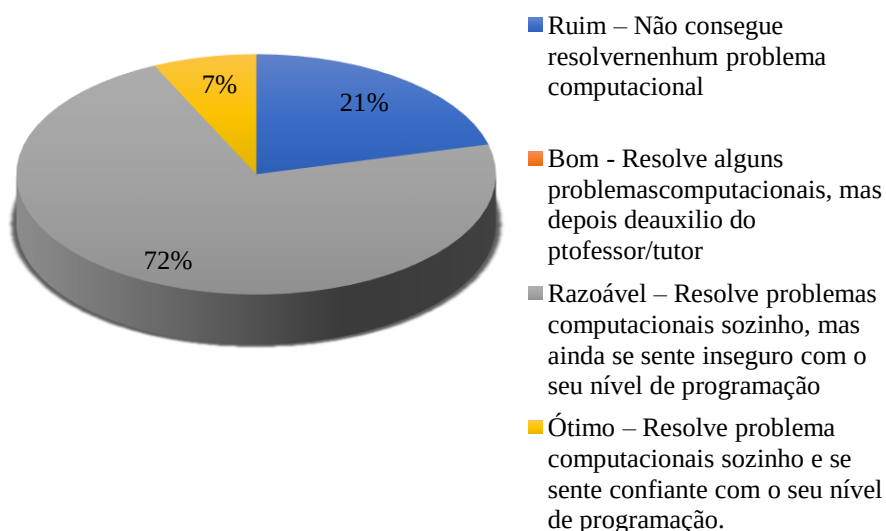
Figura 5 - Qual meio você utiliza para se comunicarem com os professores sobre um determinado conteúdo?



Fonte: Autoria própria (2017).

No que tange, como identificaria seu raciocínio em lógica de programação, como pode observar na Figura 6, 72% consideram ter um raciocínio razoável (resolve problemas computacionais sozinho, mas ainda se sente inseguro com seu nível de programação), 21% se consideram ruim (não consegue resolver nenhum problema computacional) e 7% acreditam ter um ótimo raciocínio (resolvem problema computacionais sozinho, mas ainda se sentem inseguros com seu nível de programação). Essa insegurança gera um desconforto no momento de programar porque muitas vezes o estudante pode se achar incapaz de resolver algum problema.

Figura 6 - Como você identificaria seu raciocínio em lógica de programação?

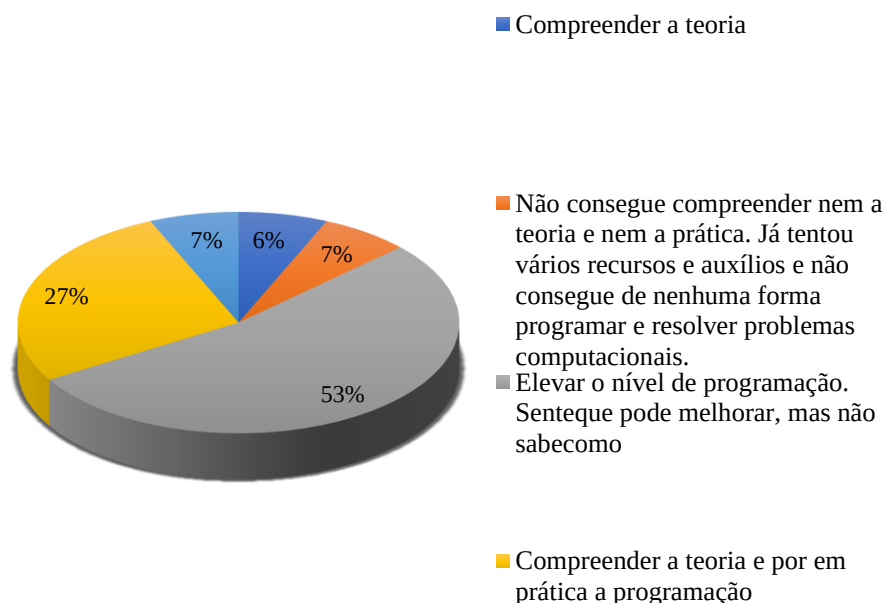


Fonte: Autoria própria (2017).

Sobre as dificuldades nas disciplinas de programação, é possível identificar na Figura 7, que 57% tem contratempo em elevar o nível de programação, sente que pode melhorar, mas não sabe como, 29% tem dificuldade em compreender a teoria e pôr em prática a programação, 7% tem como obstáculo compreender a teoria e 7% não consegue compreender nem a teoria e nem a prática. Neste caso tentou vários recursos e auxílios e não consegue de nenhuma forma

programar e resolver problemas, o que são dificuldades de grande relevância que pode afetar no desempenho e também na desistência de disciplinas.

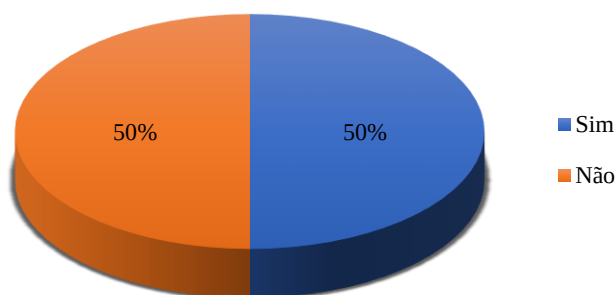
Figura 7 - Quais suas dificuldades nas disciplinas de programação?



Fonte: Autoria própria (2017).

Quanto à reprovação em disciplinas de algoritmos/programação, 50% dos estudantes já foram reprovados, enquanto que os outros 50% nunca foram reprovados, como pode ser visto na Figura 8, a reprovação pode ocasionar muitas vezes na desistência dos cursos.

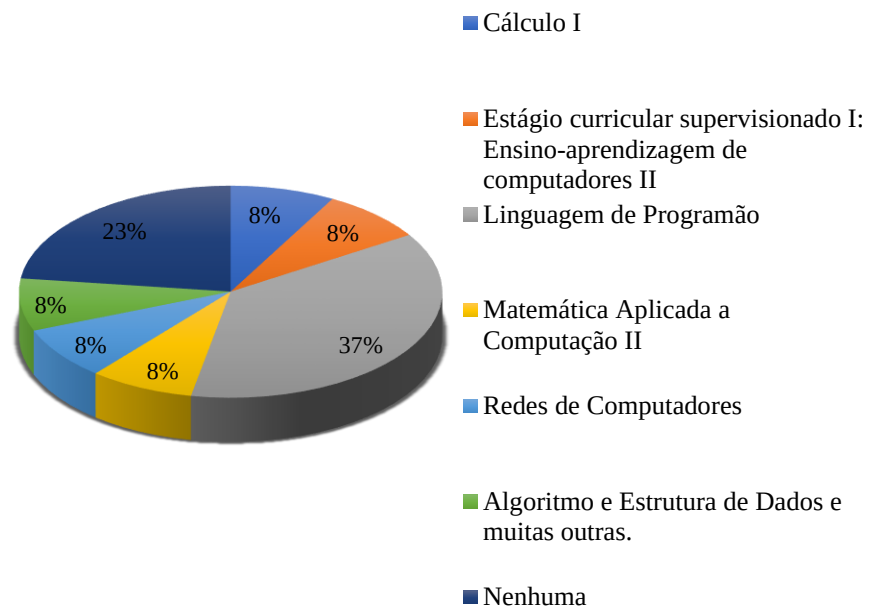
Figura 8 - Você foi reprovado em disciplina de programação?



Fonte: Autoria própria (2017).

No que diz respeito, a qual disciplina ficou reprovado, 8% ficou reprovado em cálculo I, 8% em estágio curricular supervisionado I: ensino-aprendizagem de computadores II, 37% em linguagem de programação, 8% matemática aplicada a computação II, 8% redes de computadores, enquanto que 23% não ficou reprovado em nenhuma disciplina. Isso é preocupante e, provavelmente, é uma das principais causas das desmotivações dos estudantes e que ocasiona na desistência de cursos na área de tecnologia que envolvam alguma disciplina de programação, como pode ser visto na Figura 9.

Figura 9 - Quais disciplinas ficou reprovado?



Fonte: Autoria própria (2017).

Diante do que foi exposto a partir dos dados coletados em pesquisa com alunos do curso de Licenciatura em Computação a Distância, foi iniciado o projeto de um sistema web que atuasse no foco deste problema fornecendo um suporte extra aos alunos com dificuldades na prática de programação.

5 DESENVOLVIMENTO DA APLICAÇÃO

O objetivo do desenvolvimento dessa aplicação online, é auxiliar o ensino-aprendizagem de disciplinas de algoritmos/programação. Assim, facilitando as atividades dos educadores e estudantes. Os docentes têm como atividades o cadastramento de exercícios computacionais, postagem de conteúdo (vídeo, apostilas e slides), acessar as atividades dos estudantes e tirar dúvidas via chat/fórum. Por outro lado, os discentes têm como tarefas responder problemas computacionais, assistir vídeos, baixar apostila, slide e comunicar-se com seus docentes para tirar dúvidas via chat/fórum. Além disso, cabe ao administrador do sistema o cadastro de professores, alunos, turmas e disciplinas.

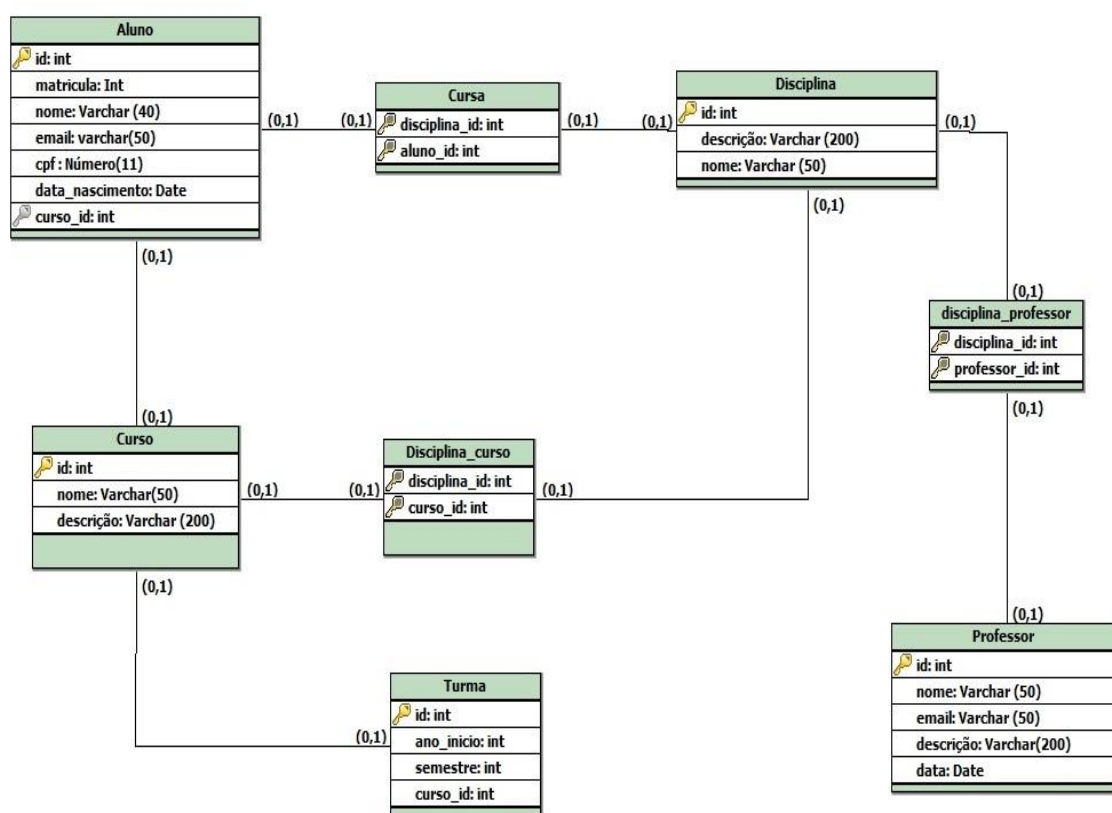
Dessa forma, os principais requisitos do sistema, são cadastro de usuários, postagem de material, fórum, bate-papo, automatizar a submissão e correção de exercícios.

Assim iniciou-se o projeto com a modelagem dos diagramas de UML – Linguagem de Modelagem Unificada (*Unified Modeling Language*), foram modelados inicialmente dois tipos, casos de uso e de classes. No que diz respeito ao diagrama de casos de uso ele define o que cada usuário irá fazer o uso da aplicação. Já o diagrama de classes determina as classes existentes e

os relacionamentos entre as mesmas. Dessa forma, utilizou-se a ferramenta Astah para criação dos modelos de diagramas.

Porém, em seguida teve início a fase de modelagem relacional do banco de dados em que foi definido o relacionamento existente entre cada entidade, aluno, disciplina, curso, professor e turma. Para isso escolheu-se o banco de dados relacional Mysql por ser gratuito, compatível com distintas linguagens de programação, ser multiplataforma e eficaz para aplicações web. O modelo relacional do banco de dados é apresentado na Figura 10.

Figura 10 - - Modelo lógico do banco de dados do sistema de suporte a aprendizagem de programação



Fonte: Autoria própria (2017).

Além disso, para o desenvolvimento e a implementação do sistema foi escolhida a linguagem de programação PHP, que foi produzida por Ramus Lerdorf, no ano de 1994, inicialmente foi definida como Personal Home Page, porém posteriormente foi redefinida para PHP (Hypertext Preprocessor). Tinha como propósito a criação de páginas proativas para controlar o acesso de currículos na internet. Por outro lado, na sua concepção foi adicionado características de outras linguagens como C, Java e Perl.

A escolha dessa linguagem aconteceu por causa de suas particularidades de ser usada para desenvolvimento web, código aberto, portabilidade, multiparadigma, interpretada e porque grande maioria dos bancos de dados suportam a linguagem PHP.

Ainda convém lembrar que foi usado o framework Laravel baseado na linguagem de programação PHP, pelo fato dos frameworks facilitarem a vida do desenvolvedor e também por possuírem inúmeros recursos e funcionalidades que torna o desenvolvimento mais rápido e eficaz.

É necessário frisar, por outro lado, que a parte de implementação foi realizada apenas o sistema de cadastros e uma página para postar e mostrar vídeos diretamente do You Tube. Na Figura 11, é possível observar a tela de login que os usuários encontraram.

Figura 11 - Login do sistema.



Login

E-Mail

Senha

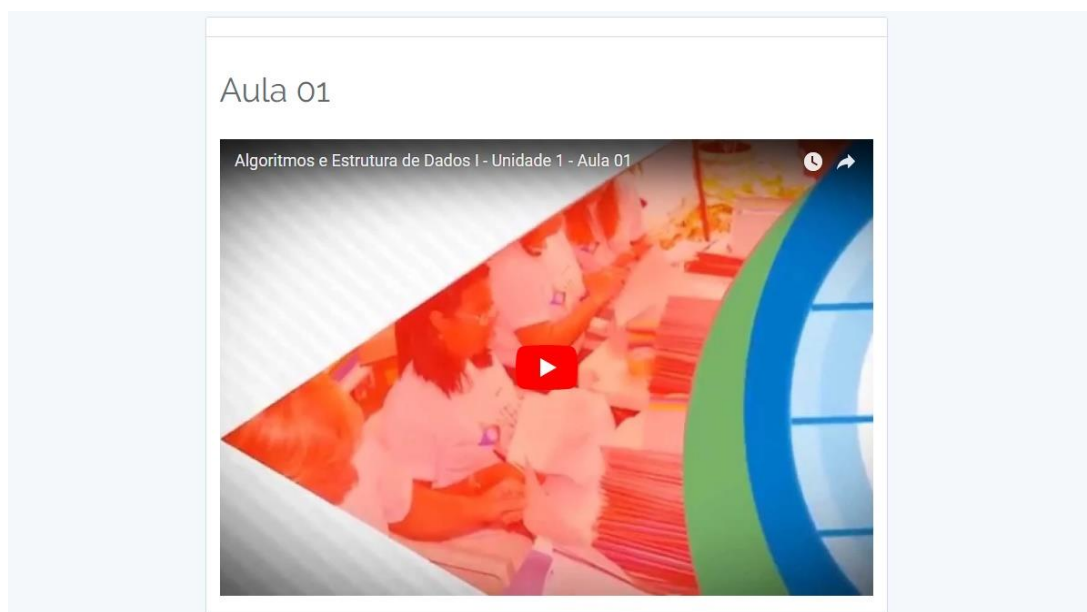
☐ Lembre de mim

[Login](#) [Esqueceu sua senha?](#)

Fonte: Autoria própria (2017).

A Figura 12, mostra o ambiente onde os alunos poderão ter acesso aos vídeos que foram postados por seus docentes.

Figura 12 - Pagina de Vídeos.



Fonte: Autoria própria (2017).

6 RELATO DE EXPERIÊNCIA

A fim de averiguar e comprovar se a ferramenta de suporte na aprendizagem de programação e/ou suas características são favorável para aquisição de conhecimento em disciplinas de algoritmos e/ou linguagem de programação, julgou-se imprescindível a realização de testes e elaboração e aplicação de um formulário com nove questões, para com cinco estudantes do oitavo período do curso de Licenciatura em Computação da Universidade Federal Rural do Semi-Árido – UFERSA.

Contudo, o pressuposto dessa etapa de aquisição de dados buscou reconhecer, se o sistema possui uma interface amigável, facilita a aprendizagem, se a automação compromete a aprendizagem, usaria com seus futuros discentes, a automação melhora a atividade do professor, o feedback rápido melhora o processo de ensino aprendizagem, disponibilizar conteúdos facilita a aprendizagem dos estudantes, se tivesse utilizado uma aplicação similar teria tido menos dificuldades em programar, quais funcionalidades você sugeria para ser inserida na aplicação.

É possível observar a disposição de percentagem do *feedback* da primeira indagação, referente as funcionalidades do sistema apresentado aos alunos do Curso de Licenciatura em Computação à Distância. No que concerne, se o sistema possui uma interface amigável, 100% acredita que sim, o sistema dispõe de uma interface inteiramente amigável para os estudantes.

Com relação, se o sistema facilita a aprendizagem, pode-se observar que 100% dos discentes que responderam o questionário acham que sim. Isso mostra que os estudos e principalmente o desenvolvimento de ferramentas que se preocupe na parte pratica das disciplinas de programação, tem surtido efeito e proporcionado para os estudantes uma maneira de superar suas dificuldades.

No que tange, se a automação pode comprometer a aprendizagem, 20% acredita que sim e 80% considera que não. Indicando assim, que a automação de exercícios práticos de programação não afeta a aprendizagem dos discentes.

Em relação se eles usariam com seus futuros discentes, 100% confessam que sim, demonstrando que o auxílio de uma aplicação para submissão e automatização de exercícios de programação é essencial tanto para os discentes como para os docentes.

Além disso, no que se refere a automação de algumas tarefas melhorar a atividade do professor, 100% dos alunos questionados afirmam que melhora as tarefas dos docentes, porque se o professor tiver uma turma muito numerosa, o mesmo terá um grande trabalho na correção de atividades manuais. Assim, uma aplicação de submissão e automação possibilitaria uma diminuição de tempo para correção das tarefas e um rápido feedback.

No que tange a um feedback rápido melhorar o processo de ensino-aprendizagem, 100% dos estudantes ratificaram que sim, porque quanto mais rápido o discente obtiver uma resposta, melhor será o processo de ensino-aprendizagem, porque o mesmo vai ter um maior tempo para corrigir erros e dúvidas.

Quanto a disponibilização de conteúdos facilitar a aprendizagem dos estudantes, 100% responderam que sim, isso porque quando o professor disponibiliza distintos tipos de conteúdo ao aluno, seja ele vídeo, apostila, artigos e slides, ele terá diferentes formas de aprender, seja lendo as apostilas ou assistindo vídeos.

Com relação a se tivessem utilizado uma aplicação similar teria tido menos dificuldades em programar, 100% confirmou que sim, porque com a aplicação iriam ter um feedback mais rápido e poderiam praticar mais exercícios de programação.

Ainda convém lembrar, no concerne a quais funcionalidades sugeria para ser inserida na aplicação, pode-se identificar que, 20% não respondeu, 20% logar com contas externas (Facebook e Gmail), 40% Chat (Bate-papo) e 20% jogos educativos. São funcionalidades relevantes em uma plataforma para o ensino-aprendizagem, login com contas externas facilita o cadastro dos mesmos, os chats é uma maneira de acontecer a comunicação e os jogos educativos.

Levando em conta as informações colhidas percebe-se que, o sistema proposto é uma ferramenta imprescindível tanto para os docentes como para os estudantes, e que precisam ser incorporadas funcionalidades que facilitem o feedback e dessa forma melhore cada vez mais o processo de ensino-aprendizagem. Porém, por causa que amostra de dados ter sido muito pequena, é imprescindível que futuros trabalhos, investiguem com um grupo maior de pessoas e principalmente com um questionário mais detalhado.

7 CONSIDERAÇÕES FINAIS

O presente trabalho tem como finalidade a construção de uma ferramenta web, baseada em juiz online para contornar os obstáculos encontrados por alunos na aprendizagem de programação e também ajudar professores no ensino das disciplinas de programação, no ensino presencial e principalmente na educação a distância, por possibilitar com que os estudantes tenha uma resposta instantânea, outra preocupação existente é propiciar com que os docentes acompanha as maiores dificuldades encontrada por seus alunos.

Objetiva-se incorporar na ferramenta técnicas mais inteligentes no processo de comunicação, submissão, correção, além da automação na correção de códigos pretende-se possibilitar com que os docentes comentem os códigos feito por seus alunos. Ainda convém

lembra, que desejasse adicionar método prático de programação colaborativa, possibilitar com que os professores possam postar conteúdo e integrar a ferramenta ao Moodle.

Dessa forma para novos projetos/pesquisas recomendasse a criação de protótipos para dispositivos moveis como smartphone e/ou tablet, visto que grande maioria das pessoas dispões desses aparelhos.

REFERÊNCIAS

- ALVES, Fábio P.; JAQUES, Patrícia. **Um ambiente virtual com feedback personalizado para apoio a disciplina de programação.** In: Anais dos Workshops do Congresso Brasileiro de Informática na Educação. 2014. p.51.
- CAMPOS, Cassio P.; FERREIRA, Carlos E. **BOCA:** um sistema de apoio a competições de programação. In: Workshop de Educação em Computação. 2004. p. 885-895.
- CHAVES, José Osvaldo et al. **MOJO:** uma ferramenta para auxiliar o professor em disciplinas de programação. In: Congresso Brasileiro de Ensino Superior a Distância, Belém, PA. 2013.
- COSTA, Vânia Medianeira Flores et al. **Educação a distância x educação presencial: como os alunos percebem as diferentes características.** In: CONGRESSO BRASILEIRO DE EDUCAÇÃO SUPERIOR A DISTÂNCIA. 2014. p. 2088-2102.
- GIL, A. C. **Como elaborar projetos de pesquisa.** 4. ed. São Paulo: Atlas, 2007.
- KURNIA, A.; LIM, A.; cheang, B. **Online judge.** Computers & Education, v. 36, n. 4, p. 299-315, 2001.
- LESTA, L.; YACEF, K. **An intelligent teaching assistant system for logic.** In: International Conference on Intelligent Tutoring Systems. Springer, Berlin, Heidelberg, 2002. p. 421-431.
- NASCIMENTO, F.L.M; SILVA, J.B.O; lucena, L. **Inspetor online.** II congresso sobre tecnologia na educação. Mamanguape, Paraíba. 2017.
- PRIETCH, S. S., & PAZETO, T. A. **Estudo sobre a Evasão em um Curso de Licenciatura em Informática e Considerações para Melhorias.** WEIBASE, Maceió/AL. 2010.
- RAABE, A. L. A., & Silva, J. D. **Um ambiente para atendimento as dificuldades de aprendizagem de algoritmos.** In XIII Workshop de Educação em Computação (WEI'2005). São Leopoldo, RS, Brasil. 2015.
- SANTOS, Joanna CS; RIBEIRO, Admilson RL. **JOnline:** proposta preliminar de um juiz online didático para o ensino de programação. In: Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE). 2011.
- WELLING, Luke; THOMSON, Laura. **PHP e MySQL Desenvolvimento Web.** 3. ed. Elsevier, 2005. 712 p.