

DESENVOLVENDO UM OBJETO DE APRENDIZAGEM VIRTUAL PARA O ENSINO DA PROGRAMAÇÃO *WEB* *MOBILE* COM O *DESIGN THINKING*

Geraldo Gomes da Cruz Júnior¹, Rafaella Leandra Souza do Nascimento¹

¹ Departamento de Estatística e Informática - Universidade Federal Rural de Pernambuco (UFRPE) - Recife, PE - Brazil

{geraldoj8, rafaellalsn}@gmail.com

Abstract. *The practice of programming helps in logical thinking and creativity, as well as facilitate the learning of subjects such as Portuguese and Mathematics. It focused on the relevance that there is around the schedule and the deficit in practical teaching of this study is to analyze the ecosystem of virtual learning objects (VLOs) programming and develop a VLO for teaching web programming with HTML5, CSS3 and Javascript. The VLO developed was planned from its ideation steps following the methodology of Design Thinking, so to arrive at a product with innovative features and focus on the needs of its members. We came to the Learning Mobile Apps, an idealized system to be multiplatform, collaborative and innovative.*

Resumo. *A prática da programação ajuda no raciocínio lógico e na criatividade, além de facilitar o aprendizado de disciplinas como o português e a matemática. É focado na relevância que há em torno do ensino da programação e no déficit na prática desta que este estudo visa analisar o ecossistema dos objetos de aprendizagem virtuais (OAVs) de programação e desenvolver um OAV para o ensino da programação web com HTML5, CSS3 e Javascript. O OAV desenvolvido foi planejado desde suas etapas de ideação seguindo as metodologias do Design Thinking, para que se chegasse a um produto com características inovadoras e de foco nas necessidades de seus usuários. Chegou-se ao Learning Mobile Apps, um sistema idealizado para ser multiplataforma, colaborativo e inovador.*

1. Introdução

Objetos de Aprendizagem Virtual, segundo Wiley (2000) podem ser compreendidos como “qualquer recurso digital que possa ser reutilizado para o suporte ao ensino”. Conforme Kay e Knaack (2007) objetos de aprendizagem são todas as ferramentas interativas baseadas na *web* que apoiam o aprendizado de conceitos específicos incrementando, ampliando, ou guiando o processo cognitivo dos aprendizes. Os OA podem ser criados em qualquer mídia ou formato, podendo ser simples como uma animação ou uma apresentação de slides ou complexos como uma simulação [MACÊDO et al, 2007]. Ou seja, pode envolver textos, animações, vídeos, áudios, imagens, aplicações e páginas Web, que combinadas, se destinam a apoiar o ambiente de aprendizado.

O uso de um OAV pode reduzir o tempo de desenvolvimento de uma atividade, diminuir a necessidade supervisão nestas tarefas, assim como os custos. Eles podem ser usados tanto para apoiar a aprendizagem presencial quanto à distância.

Diversos problemas podem ser citados no processo de ensino e aprendizagem de algoritmos e programação que culminam em desmotivação, reprovação ou evasão [Rapkiewicz 2006]. Entre estes, mencionados por Rodrigues (2002), pode-se destacar: i) dificuldade de adaptação dos alunos desenvolverem raciocínio lógico quando estão acostumados a decorar o conteúdo; ii) falta de motivação do aluno gerada pelo despreparo e o desânimo quando ele acredita que a disciplina constitui um obstáculo extremamente difícil de ser superado. Além destes motivos, pode ser apontado como fator limitante o ensino instrucionista: segundo Borges (2000), o modo tradicional não consegue facilmente motivar os alunos a se interessar pela disciplina, entre outras razões, pois não é clara para estes a importância de certos conteúdos para sua formação.

Existe hoje dificuldade na iniciação da programação *web*, tanto por se tratar da programação em si, como exclama Rapkiewicz, Rodrigues e Borges, como também pelas constantes mudanças nessa tecnologia e em seus padrões. Niskier (1985) destaca que os professores se veem às voltas com as necessidades de transmitir uma quantidade de conhecimento em constante progressão que envelhecem rapidamente, até mesmo em função do desenvolvimento científico-tecnológico, obrigando-os a uma permanente atualização.

Nas escolas, falta o incentivo e a viabilidade, são poucas instituições de ensino que incluem o ensino da programação em suas atividades. Mas este problema não é aplicado apenas aos alunos de ensino fundamental ou ensino médio. Os alunos de graduação também enfrentam dificuldade na hora de aprender, muitas vezes pela metodologia de ensino aplicada pelo professor, que não é interativa, levando-os a estudar por conta própria.

No entanto, nem sempre estudar por meios próprios é uma tarefa simples. Há uma busca por informações relevantes que resolvam problemas específicos e por ferramentas que possam auxiliar em seu aprendizado de distintas tecnologias, sendo difícil localizar uma centralização destas informações.

O estudo de linguagens de programação, combinado com a prática do desenvolvimento destas, ajuda no raciocínio lógico e na criatividade, além de facilitar o aprendizado de disciplinas como o português e a matemática (Revista Educação, 2014).

Com foco na relevância que há em torno do ensino da programação, este estudo visa analisar o ecossistema de objetos de aprendizagem virtuais (OAV) de programação e desenvolver um OAV para o ensino da programação *web* utilizando recursos como a linguagem, o *html5*, o *css3* e o *javascript*.

O *Learning Mobile Apps* foi o sistema desenvolvido, focado em auxiliar na resolução da problemática do ensino da programação para distintos públicos e servindo como um acervo para resolução de dúvidas. O OAV desenvolvido procurou trazer como diferencial a reunião de fontes conceituadas de estudos, com apontamentos para terceiros, jogos interativos, tutoriais, vídeos, colaboração e comunicação entre usuários do sistema e tudo isso em uma ferramenta gratuita e em português.

O projeto tem como área de estudo a Educação, e tem como área específica o ensino da programação *web* em instituições de ensino, pois percebe-se a necessidade de resolver problemas relacionadas a este seguimento. Para isto, todo o desenvolvido do

OAV foi apoiado pelas metodologias do *Design Thinking*, para que se chegasse a uma solução ratificada de conceitos de inovação e foco nas necessidades dos usuários.

Segundo Vianna et al (2012) o uso do *Design Thinking* busca formular questionamentos através da apreensão ou compreensão dos fenômenos, ou seja, são formuladas perguntas a serem respondidas a partir das informações coletadas durante a observação do universo que permeia o problema. Assim, ao pensar de maneira abductiva, a solução não é derivada do problema: ela se encaixa nele. E é com esta filosofia que este estudo foi desenvolvido.

2. Metodologia

O desenvolvimento do projeto foi guiado pelas principais etapas do *Design Thinking* (Figura 1), passando inicialmente pela Imersão (Entender e Observar), Análise e Síntese (Definir), depois pela Ideação (Idealizar) até se chegar a Prototipação (Prototipar e Testar).

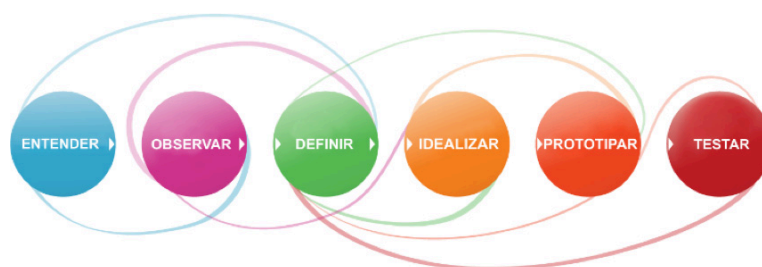


Figura 1. Fases do Design Thinking

Para o desenvolvimento do projeto também foram utilizadas técnicas dos padrões de desenvolvimento ágil de software, para a construção de um Produto Mínimo Viável (MVP). O termo Desenvolvimento Ágil identifica metodologias de desenvolvimento que adotam os princípios do Manifesto Ágil (2001), tendo como alguns princípios os seguintes: i. Indivíduos e interação entre eles mais que processos e ferramentas; ii. Software em funcionamento mais que documentação abrangente; iii. Colaboração com o cliente mais que negociação de contratos; iv. Responder a mudanças mais que seguir um plano;

Uma característica das metodologias ágeis é que elas se adaptam a novos cenários durante o desenvolvimento do projeto, ao invés de tentar analisar previamente tudo o que pode ou não acontecer no decorrer do desenvolvimento.

2.1. Imersão

A primeira fase do processo de *Design Thinking* é chamada Imersão. Segundo Vianna et al (2012) nesse momento a equipe de projeto aproxima-se do contexto do problema, tanto do ponto de vista da empresa (o cliente) quanto do usuário final (o cliente do cliente).

A Imersão pode ser dividida em duas etapas: Preliminar e em Profundidade. A primeira tem como objetivo o reenquadramento e o entendimento inicial do problema, enquanto a segunda destina-se à identificação de necessidades e oportunidades que irão nortear a geração de soluções na fase seguinte do projeto, a de Ideação. Durante as etapas Preliminares foram utilizadas as técnicas de Pesquisa *desk* e Pesquisa exploratória, a partir daí na fase de Profundidade utilizou-se as técnicas de Um dia na vida e Sombra.

A Pesquisa *desk* é uma busca de informações sobre o tema do projeto em fontes diversas. O nome *desk* origina-se de *desktop*, e é utilizado porque a maior parte da

pesquisa secundária realizada atualmente tem com base referências seguras da internet [Vianna 2012]. Com esta pesquisa foram levantados dados que mostraram a significância do aprendizado das técnicas de programação, bem como o déficit no ensino desta. Também foi feita uma verificação dos OAV's já existentes no meio ensino da programação.

A Pesquisa exploratória é a pesquisa de campo preliminar que auxilia a equipe no entendimento do contexto a ser trabalhado e fornece insumos para a definição dos perfis de usuários, atores e ambientes ou momentos do ciclo de vida do produto/serviço que serão explorados na Imersão em Profundidade [Vianna 2012]. Nesta fase, foram utilizados questionários *online* e entrevistas em grupo e individuais com estudantes e interessados da área, buscando observar a necessidade e os pontos-chaves a serem atacados e começar o desenvolvimento da ideia para se chegar a solução.

A técnica de Um dia na vida é uma simulação, por parte do pesquisador, da vida de uma pessoa ou situação estudada. Ou seja, membros da equipe de projeto assumem o papel do usuário e passam um período de tempo agindo sob um diferente ponto de vista e interagindo com os contextos e pessoas com os quais se estaria confrontado no dia a dia [Vianna 2012]. Nesta fase, a equipe assistiu algumas aulas de programação na Universidade Federal Rural de Pernambuco (UFRPE) e na Universidade Federal de Pernambuco (UFPE), buscando aprender o assunto ensinado e observar possíveis dificuldades.

A técnica da Sombra é o acompanhamento do usuário ao longo de um período de tempo que inclua sua interação com o produto ou serviço que está sendo analisado. Enquanto “sombra”, o pesquisador não deve interferir na ação do usuário, apenas observá-lo [Vianna 2012]. Nesta fase os membros da equipe assistiram aulas da Introdução a Programação no primeiro período da UFRPE para observar o aprendizado e o comportamento dos alunos.

2.2. Análise e síntese

Após as etapas de levantamento de dados da fase de Imersão, os próximos passos são análise e síntese das informações coletadas. Para tal, os *insights* são organizados de maneira a obter-se padrões e a criar desafios que auxiliem na compreensão do problema. Nesta etapa, foi criado um Mapa conceitual com as ideias levantadas da etapa de Imersão e definida as Personas do projeto.

O Mapa Conceitual é uma visualização gráfica, construída para simplificar e organizar visualmente dados complexos de campo, em diferentes níveis de profundidade e abstração. Seu objetivo é ilustrar os elos entre os dados e, assim, permitir que novos significados sejam extraídos das informações levantadas nas etapas iniciais da fase de Imersão, principalmente a partir das associações entre elas [Vianna 2012]. Nesta fase, todos os conhecimentos e ideias levantados durante a etapa de Imersão foram colocados em *post-its* onde pode-se traçar o mapa conceitual do projeto.

Personas são arquétipos, personagens ficticiais, concebidos a partir da síntese de comportamentos observados entre consumidores com perfis extremos. Representam as motivações, desejos, expectativas e necessidades, reunindo características significativas de um grupo mais abrangente [Vianna 2012]. Foram definidas 5 Personas sendo, dois estudantes e dois professores, da graduação e do ensino médio, e um interessado em saber mais sobre práticas de programação.

2.2. Ideação

Esta fase tem como intuito gerar ideias inovadoras para o tema do projeto e, para isso, utilizam-se as ferramentas de síntese criadas na fase de análise para estimular a criatividade e gerar soluções que estejam de acordo com o contexto do assunto trabalhado. Nesta fase, a equipe fez uso da técnica do *brainstorming*, onde foram levantadas diversas ideias, por cada integrante, que pudessem vir a solucionar o problema alvo.

Brainstorming é uma técnica para estimular a geração de um grande número de ideias em um curto espaço de tempo. Geralmente realizado em grupo, é um processo criativo conduzido por um moderador, responsável por deixar os participantes à vontade e estimular a criatividade sem deixar que o grupo perca o foco [Vianna 2012]. Após a realização do *Brainstorming* de ideias, houve uma discussão entre a equipe do projeto para se chegar a uma definição de qual ideia, ou conjunto de ideias, seria ideal.

2.3. Prototipação

A Prototipação tem como função auxiliar a validação das ideias geradas e, apesar de ser apresentada como uma das últimas fases do processo de *Design Thinking*, pode ocorrer ao longo do projeto em paralelo com a Imersão e a Ideação [Vianna 2012].

O protótipo é a tangibilização de uma ideia, a passagem do abstrato para o físico de forma a representar a realidade - mesmo que simplificada - e propiciar validações. É um instrumento de aprendizado sob dois aspectos: ótica da equipe mais o ponto de vista do usuário [Vianna 2012].

Protótipos em papel são representações de interfaces gráficas com diferentes níveis de fidelidade, desde um *wireframe* desenhado à mão em pequenos pedaços de papel, para representar esquematicamente as telas de um aplicativo de celular, até uma embalagem de sabonete com detalhes finais de texto e cores. Um protótipo em papel pode começar de maneira simplificada e ganhar complexidade ao longo das iterações com o usuário ou com a equipe [Vianna 2012]. Nesta fase, foi elaborado um protótipo em papel da solução escolhida na etapa de *Brainstorming*, e este protótipo posteriormente foi levado para validação de alguns possíveis usuários do sistema.

2.4 MVP

Os empresários são frequentemente confrontados com recursos limitados em sua busca para comercializar novas tecnologias. O MVP é uma forma de arranque simples, a qual pode ser aplicada a um organização, independentemente de seu tamanho ou ambiente [Moogk 2012]. Tomando por bases estes princípios, juntamente com a prática ágil do *Scrum*, se desenvolveu o *Learning Mobile Apps*. O método *Scrum* segue os princípios do Manifesto Ágil (Manifesto Ágil, 2001) e tem como pai três de seus signatários: Mike Beedle, Ken Schwaber e Jeff Sutherland. Segundo Schwaber e Beedle (2002), ele tem como objetivo definir um processo de desenvolvimento de projetos focado nas pessoas da equipe.

O MVP do *Learning Mobile Apps*, foi desenvolvido tomando por base o protótipo em papel, validado por possíveis usuários do sistema. Com o MVP desenvolvido, foi disponibilizado o acesso para ele via *internet*, para que mais usuários pudessem testar suas funcionalidades e para se obter mais *feedbacks*. O uso da técnica de MVP possibilitou a obtenção rápida de um produto de qualidade para mais testes com usuários. O uso de MVPs traz aprendizagem acelerada para ajudar a reduzir a incerteza que acompanha a

comercialização de projetos, o que leva novas tecnologias ao mercado mais rapidamente [Moogk 2012].

Para o desenvolvimento do MVP foi utilizada a linguagem de marcação *html5*, a de estilo *css3* e a de programação *javascript*. Estes padrões foram escolhidos para garantir que o acesso e experiência de uso do sistema fosse realizada de forma agradável e responsiva. A adaptabilidade é uma questão importante no desenvolvimento de aplicações para múltiplas plataformas. Assegurar esta capacidade às interfaces tem sido um grande desafio. Neste sentido, o *W3C (World Wide Web Consortium)* tem empregado um grande esforço para prover esta característica à web e assegurar a independência de dispositivos de suas aplicações [Neto 2004].

3. Resultados

Na busca por respostas e feedbacks dos usuários, e visando continuamente aperfeiçoar o projeto lançado, a equipe inferiu sobre os dados obtidos durante as fases do *Design Thinking*.

3.1. Imersão

Na fase de Imersão, durante a Pesquisa Desk foi realizado um levantamento de informações que comprovassem a importância do ensino e aprendizado da programação, bem como dados do cenário deste ensino no Brasil e no mundo. Também foram levantados aspectos que comprovassem a relevância das tecnologias utilizadas durante o desenvolvimento do projeto. Alguns resultados desta pesquisa inicial podem ser encontrados durante a leitura deste estudo.

Na Pesquisa *desk* ainda foram levantados outros OAVs, ou semelhantes, relacionados a programação e ao ensino desta. Alguns dos estudados foram os seguintes:

- *Codyhouse*: É uma biblioteca de *html*, *css* e *javascript*. Além dos conteúdos, disponibiliza curso, no entanto é pago. O site é em inglês e é direcionado para o público com experiência na área (<https://codyhouse.co/>).
- *Code combat*: É um jogo *multiplayer* que tem a proposta de ensinar a programar com diversão. O jogo é gratuito para os primeiros cinco níveis. É necessário que um professor faça uma assinatura e após isso, adicione os alunos. Também oferece assinatura para professores para fins de avaliação de alunos (<http://br.codecombat.com/>).
- *Codecademy*: É um *site* que possui uma plataforma interativa *online* e oferece aulas gratuitas de programação em linguagens como *jQuery*, *Javascript*, *Python*, *Ruby*, *PHP*, bem como as linguagens de marcação, incluindo *HTML* e *CSS* (<https://www.codecademy.com/pt>).
- Computação na escola: A Iniciativa Computação na Escola é dedicada a aumentar o ensino da computação nas escolas. Seu site possui tutoriais para alunos, apoio a professores e técnicos de informática com material didático, piloto em escola e divulgação de eventos (<http://www.computacaonaescola.ufsc.br/>).

Posteriormente, o trabalho seguiu com a realização da pesquisa exploratória por meio de questionários online (<https://goo.gl/tBBTef>) com estudantes e interessados da área. Os resultados podem ser vistos em (<https://goo.gl/TfaIwW>).

Na etapa da Imersão em profundidade, utilizou-se as técnicas de Um dia na vida e de Sombra, onde os membros da equipe foram assistir aulas de Programação na UFRPE e na UFPE, e pode-se perceber que durante as aulas a maioria dos alunos prestavam atenção, mas alguns perdiam o foco para jogos e redes sociais. A maioria dos alunos comentavam que aprendiam mais em casa pesquisando os assuntos na *internet* e que desconheciam a existência de OAVs que pudessem ajudar em seus aprendizados. Outro ponto importante observado foi, os alunos sempre prestavam mais atenção as aulas quando o professor fazia uso de alguma brincadeira ou piada.

3.2. Análise e síntese

A equipe criou um mapa conceitual com *post-its* de tudo aquilo que foi levantado durante a etapa de Imersão, com isto, pode-se dividir os artefatos em grupos. Com o mapa criado pode-se definir Personas que simbolizaram os principais *stakeholders* do projeto, foram criadas 5 (cinco) personas, dois estudantes e dois professores, da graduação e do ensino médio, e um interessado em saber mais sobre práticas de programação, ambos com idade, nome e características específicas.

3.3. Ideação

Todo o processo de ideação seguiu as metodologias para a realização de um *Brainstorming*, ou seja, gerando um processo de discussão de ideias livres com foco em pivotar nos conceitos iniciais aos quais se deseja solucionar e atacar.

As ideias foram surgindo de forma aleatória, sem filtros para complexidade de implementação ou de se seriam possíveis ou não de serem realizadas. Frente a todas as ideias lançadas foi construída uma lista com estas, e os integrantes em conjunto discutiram sobre qual seria a solução ideal para atacar a problemática em discussão. Dentre as soluções apontadas pelos integrantes durante a etapa de *brainstorming* destacam-se: Jogo para o ensino da programação; Um *blog* com discussões sobre o tema; Um objeto de aprendizagem virtual; Um site que reúna assuntos sobre o tema; Tutoriais explicativos; Jogos clássicos que remetam ao ensino da programação; Um canal para diferentes fontes de aprendizados de terceiros; Um sistema de *IDE online* para programação.

Frente as ideias levantadas, aos resultados de soluções já existentes na área e ao desafio que buscasse atacar, a equipe entrou em consenso quanto a construção de uma ferramenta que deve servir como base de auxílio ao aprendizado da programação, com um pouco de cada ideia discutida.

Foi planejado o desenvolvimento de um sistema onde em cada um de seus módulos se possa atacar um dos problemas levantados, como o aprendizado da programação para diferentes públicos, conteúdo gratuito e em português, centralização de informações, espaços para interação e dúvidas, dentre outros. Para isto, se esboçou o desenvolvimento de módulos de jogos clássicos voltados para o ensino, indicadores de tutoriais, explicações sobre o tema e canais para a solução de dúvidas, enfim, uma aplicação *web* modularizada que visará ao aprendizado facilitado da programação e o encaminhamento de seus usuários para locais onde possam tirar suas dúvidas.

3.4. Prototipação

Com a definição tomada a partir do *Brainstorming* da etapa de Ideação, foi desenvolvido o protótipo em papel (Figura 2) desta ideia e o mesmo foi levado para testes com alguns possíveis usuários do sistema. Abaixo segue a descrição e os resultados destes testes.

Primeiro Teste

Perfil: Interessado em aprender a programar (Estudante de administração)

Resultados: Positivos: Achou a ferramenta interessante e dinâmica. Disse que a utilizaria para começar seus aprendizados na área da programação; Negativos: Acha que deveria existir alguma espécie de glossário, ou algo explicando termos específicos utilizados no sistema que podem confundir um usuário iniciante.

Segundo Teste

Perfil: Desenvolvedora *web* do NTI UFPE

Resultados: Positivos: O protótipo traz margem para a construção de uma ferramenta poderosa de aprendizado e ponte de aprendizado para terceiros, possibilitando o aprendizado facilitado de uma nova tendência tecnológica através da própria tecnologia. Negativos: Acha que seria interessante existência de mais jogos e a possibilidade de interações diretas com as postagens.

Terceiro Teste

Perfil: Aluna da Cadeira de laboratório de programação de Sistemas de Informação da UFRPE

Resultados: Positivos: Achou a solução bem construída e com muitas possibilidades para o aprendizado. Gostou da possibilidade de poder resolver dúvidas com distintos profissionais da área. Negativos: Acha que se o *quiz* tiver muitas questões se torna cansativo; Gostaria da opção de “Retornar” em todas as páginas do sistema; Ter a opção de refazer o *quiz*; Na área de “Contato”, não mostra a mensagem de confirmação de que o *email* foi enviado.

Com a observação destes testes chegou-se as seguintes conclusões: 1. É necessária a construção de um espaço comum para interação de usuários (fóruns); 2. Criar perfis de usuários e gamificar o sistema; 3. Espaço de projetos colaborativos.

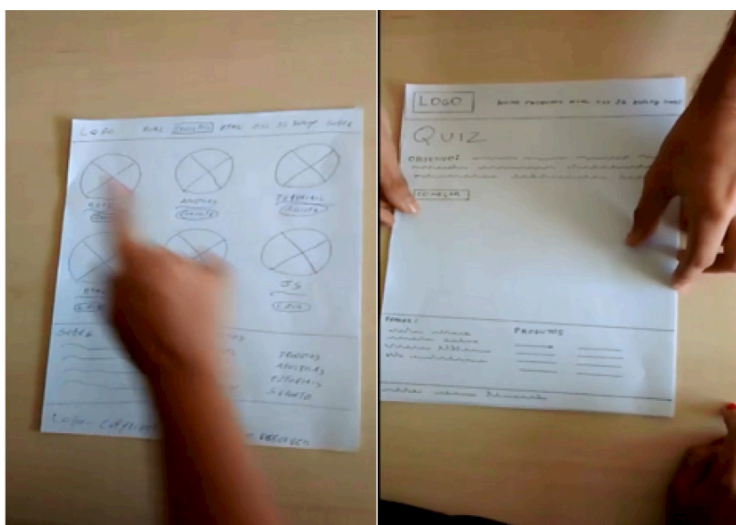


Figura 2. Protótipo em papel

3.5. MVC

Frente aos *feedbacks* obtidos do Protótipo em papel, foi desenvolvido o MVC do *Learning Mobile Apps* (Figura 3) (<http://www.learningmobileapps.esy.es/>), uma aplicação web que une os aspectos dos OAVs com os dos sistemas colaborativos. Estudos sobre aspectos colaborativos em empresas, sistemas, dentre outros, são importantes para se conhecer melhor estas instituições e assim poder adaptar e melhorar as relações existentes nestas sociedades [Cruz-Júnior 2015]. Com o processo de desenvolvimento seguindo as metodologias do *Design Thinking*, pode-se identificar a necessidade de uma ferramenta que não apenas auxiliasse no ensino da programação, mas que unisse estas características a tópicos do aprendizado colaborativo, onde fosse possível postar dúvidas, sugestões, pesquisar em sites terceiros, enfim, que possibilitasse a liberdade do aprendizado.



Figura 3. *Learning Mobile Apps*

O MVC do *Learning Mobile Apps* conta tanto com o aspecto do aprendizado inicial da programação através da reunião de tutoriais, jogos, vídeos e livros, como também possibilita que o usuário transite entre estas funcionalidades com facilidade, garantindo uma liberdade e indicação para um ensino mais agradável.

Os módulos implementados no MVC são:

- Produtos: Neste módulo encontra-se todos os materiais disponíveis no OAV, como apostilas, *quiz*, tutoriais, além de acesso a materiais envolvendo a programação *web* (*javascript*, *css*, *html*).
- *Quiz*: Este módulo apresenta conceitos básicos de tecnologias *web* através de um *quiz* com imagens, animações e *feedbacks*.
- Apostilas: Nesta sessão, os alunos poderão encontrar apostilas sobre *javascript*, *css*, *html*, entre outras, de forma a adquirir mais conhecimentos.
- Tutoriais: Neste módulo encontram-se os melhores tutoriais referentes ao tema, para que os estudantes tenham um melhor suporte no processo de aprendizagem.
- Contato: Neste módulo os usuários poderão enviar mensagens de dúvida e sugestão aos administradores do projeto.
- Sobre: Esta sessão estão contidas informações sobre o projeto e a equipe desenvolvedora.

A interface foi um ponto chave trabalhado durante a construção do projeto, por isso se optou por utilizar HTML5 e CSS3, com isso o design do MVC ficou responsivo e pode ser acessado de celulares com uma experiência agradável.

Segundo Nielsen (2012), usabilidade é uma medida de qualidade que avalia o quão fácil é utilizar uma interface na visão do usuário, seja computadorizada ou não. Dessa forma, quanto maior a usabilidade de uma interface, mais fácil será de utilizá-la, por isso são essenciais no desenvolvimento de *software* [Ferreira 2014]. Neste sentido, o Learning Mobile Apps foi desenvolvido seguindo este critério de usabilidade, para que os usuários tivessem uma boa experiência ao utilizar a ferramenta em suas atividades.

Com o MVC concluído, houve a etapa de disponibilização deste para testes (Figura 4), no qual mais uma vez o objetivo foi captar *feedbacks* de usuários-chaves. Espera-se sempre atuar neste ciclo de desenvolvimento e captação de *feedbacks* para a garantia de uma ferramenta inovadora e de foco nas necessidades de seus usuários.

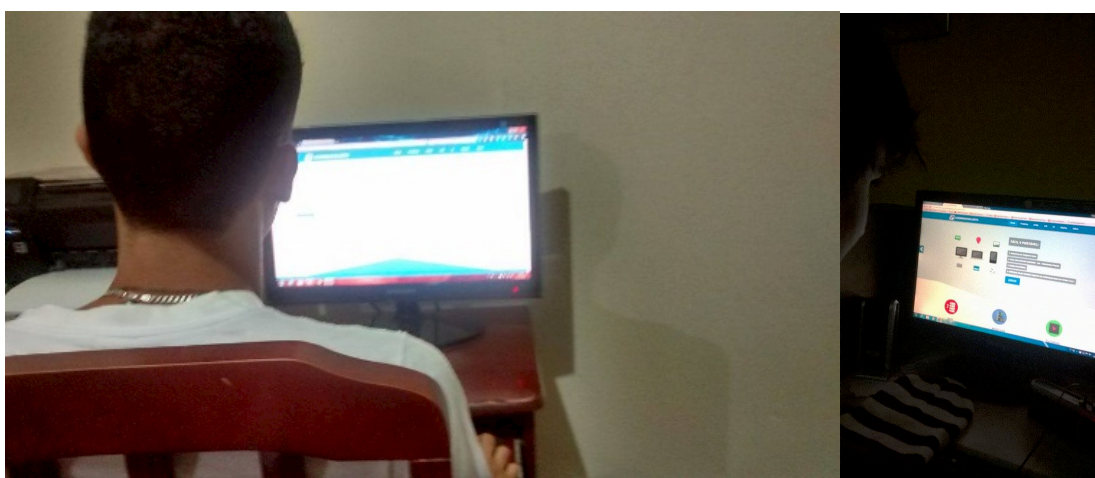


Figura 4 - Usuários validando MVP

4. Trabalhos futuros

O desenvolvimento de uma solução inicial do *Learning Mobile Apps* já foi muito satisfatório, mas espera-se incrementar aos poucos, mais jogos e aspectos colaborativos e cada vez mais indicar outras possibilidades de aprendizado interativo para os usuários e centralizar mais dados para consultas e indicações.

A proposta é dar continuidade ao projeto seguindo a metodologia *do Design Thinking*, para que o *Learning Mobile Apps* chegue a ser uma ferramenta de referência no ensino da programação, e seja adotado por instituições de ensino e por pessoas interessadas em aprender. Pretende-se manter a aproximação dos usuários ao desenvolvimento do projeto para sempre se ter *feedbacks* do que foi desenvolvido.

Espera-se portabilizar o *Learning Mobile Apps* para *mobile*, pois, a aplicações móveis são tendência e facilitarão o uso de alguns usuários. De acordo com o levantamento realizado pela agência *We Are Social*, através do relatório intitulado *Digital, Social and Mobile 2015*, no Brasil a quantidade de telefones celulares já ultrapassa o total de habitantes. São cerca de 204 milhões de pessoas no país e 276 milhões de conexões móveis, isso representa um total de 35% a mais de dispositivos em relação aos habitantes [Kemp 2015].

Outra proposta é criar um sistema de ranking para alunos que fizeram trabalhos ou jogos durante o mês, atribuindo pontuação na tentativa de incentivar os alunos a

manterem o uso da plataforma por meio da competição e ajudar os professores que quiserem acompanhar o progresso dos alunos por meio das pontuações.

Por fim, espera-se implementar uma ferramenta com tutoriais que ajudem os alunos no aprendizado do inglês. Há o incentivo aos alunos com a iniciativa gratuita e em português, mas também pretende-se auxiliar no ensino de uma segunda língua que ajude os usuários a prosseguir o aprendizado em programação em geral.

4. Conclusão

Este estudo pode mostrar aspectos distintos do uso de Objetos de Aprendizagem Virtual para o ensino da programação. Foi discutido o uso de uma metodologia para o desenvolvimento de um OAV inovador e que tivesse suas funcionalidades focadas nas necessidades dos usuários. Pode-se discutir sobre outros OAVs já existentes na área e apontar a relevância do ensino/aprendizagem da programação. Porém, o grande foco foi o desenvolvimento da aplicação *web Learning Mobile Apps* (<http://www.learningmobileapps.esy.es/>) e a importância educacional que esta ferramenta pode trazer para a sociedade.

O sistema desenvolvido oferece a reunião de fontes seguras para estudos, jogos interativos, tutoriais e vídeos, tudo gratuito, em português e responsivo. Além destas garantias, também é uma fonte de interação entre alunos que estão estudando programação, pois oferece comunicação entre usuários, os quais podem trocar experiências de aprendizado e aumentar a rede de amigos.

O *Learning Mobile Apps* não é um sistema que foi criado para competir ou substituir OAVs já existentes, mas sim para somar. Através da metodologia do *Design Thinking* pode-se perceber lacunas relatadas por usuários durante o processo de aprendizagem de técnicas de programação. Logo o ponto chave que a aplicação traz para a comunidade não é apenas os seus módulos de OAV, mas a base de sistema colaborativo que a ferramenta possui, possibilitando a troca de informações facilitada entre seus usuários.

Referências

- Borges, M. A. F. 2002. Avaliação de uma Metodologia Alternativa para a Aprendizagem de Programação. VIII Workshop de Educação em Computação – WEI 2000. Curitiba, PR.
- Bruna M. Ferreira et al. 2014. UsabiliCity: um jogo de apoio ao ensino de propriedades de usabilidade de software através de analogias. In Anais do CBIE 2014 – Congresso Brasileiro de Informática na Educação e XXV Simpósio Brasileiro de Informática na Educação (SBIE). <http://www.br-ie.org/pub/index.php/sbie/article/view/3075/2583>
- David A. Wiley. 2000. Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. <http://reusability.org/read/chapters/wiley.doc>.
- Geraldo Gomes da Cruz Júnior et al. 2015. Um estudo para implantação de uma rede social à luz do modelo 3C e das ontologias de colaboração: o caso CirCor. XII Simpósio Brasileiro de Sistemas Colaborativos (SBSC 2015). Salvador - BA, Brazil
- Jakob Nielsen. 2012. Usability 101: Introduction to Usability. Retrieved June 09, 2015. <http://www.nngroup.com/articles/usability-101-introduction-to-usability>.

- Kemp, S. (2015) “Digital, Social and Mobile in 2015”, <http://wearesocial.net/blog/2015/01/digital-social-mobile-worldwide-2015/>, Junho.
- Laécio Nobre de Macêdo. de et al. 2007. Desenvolvendo o pensamento proporcional com o uso de um objeto de aprendizagem. In: PRATA, C. L.; NASCIMENTO, A. C. de A. (Org.). *Objetos de aprendizagem: uma proposta de recurso pedagógico* Brasília: MEC, SEED, 2007.
- Luciana Alvarez. 2014. Revista Educação. Ensino de programação é aposta de colégios em todo o mundo. <http://revistaeducacao.uol.com.br/textos/211/aposta-no-futuroo-ensino-de-programacao-tem-se-espalhado-como-330266-1.asp>
- Manifesto Ágil. 2001. Disponível em: <<http://www.manifestoagil.com.br/>>. Acesso em: fev. 2016.
- Marina Kuzuyabu. 2014. Revista Educação. Linguagem de programação desde cedo. <http://revistaeducacao.com.br/textos/206/linguagem-de-programacao-desde-cedo-313309-1.asp>.
- Maurício Vianna et al. 2012 *Design thinking: inovação em negócios*. Rio de Janeiro. MJV Press, 2012. 162p.: il. ; 24 cm.
- Moogk, D. R. 2012. Minimum viable product and the importance of experimentation in technology startups. *Technology Innovation Management Review*, 2(3), 23.
- Neto, M. A. C., & Leite, J. C. 2004. Uma proposta para o desenvolvimento de interfaces de usuário multi-plataforma com tecnologia Web. *IHC 2004*
- Niskier, A. 1985. *Nova Educação: entre o coração e a máquina*. Rio de Janeiro: Bloch Editoras.p.215.
- Rapkiewicz, C. E., Falkembach, G., Seixas, L., dos Santos Rosa, N., da Cunha, V. V., & Klemann, M. (2006). Estratégias pedagógicas no ensino de algoritmos e programação associadas ao uso de jogos educacionais.*RENOTE*, 4(2).
- Robin H. Kay and Liesel Knaack. 2007. Evaluating the learning in learning objects. *Open Learning*, 22(1), 5-28.
- Rodrigues Junior, M. C. 2002. *Como Ensinar Programação? Informática – Boletim Informativo Ano I nº 01*, ULBRA, Canoas, RS.
- SCHWABER, K.; BEEDLE, M. 2002. *Agile software development with SCRUM*. Prentice Hall.