

KUBA, Heitor José; LARA, Silvana Maria Affonso de. Desenvolvimento *front-end* de ferramenta de apoio educacional. In: WORKSHOP DE INOVAÇÃO, PESQUISA, ENSINO E EXTENSÃO, 2., 2016, São Carlos, SP. *Anais...* São Carlos, SP: IFSP, 2016. p. 116-120. ISSN 2525-9377.

DESENVOLVIMENTO *FRONT-END* DE FERRAMENTA DE APOIO EDUCACIONAL

HEITOR JOSÉ KUBA¹, SILVANA MARIA AFFONSO DE LARA¹

¹ Instituto Federal de Educação, Ciência e Tecnologia de São Paulo campus São Carlos, São Carlos, Brasil.

RESUMO: A importância da tecnologia e da internet torna-se cada vez maior. Em sintonia com o avanço tecnológico, surgem novas demandas, oportunidades e soluções nas mais diversas áreas, entre elas a educação. Neste contexto, pode-se observar a necessidade crescente por novas ferramentas que auxiliem no processo de ensino e aprendizagem. Este trabalho descreve a metodologia, as dificuldades e as tecnologias envolvidas no desenvolvimento *front-end* de uma ferramenta de apoio educacional. A aplicação tem como objetivo a otimização da interação entre professor e alunos durante uma aula expositiva por meio de um sistema de postagem e avaliação de dúvidas e respostas em tempo real.

PALAVRAS-CHAVE: *Front-End*. *Web*. Educação. Ensino.

ABSTRACT: The importance of technology and internet has been growing increasingly. As technology advances, new needs, opportunities and solutions emerge in several areas, including education. In this context, it is possible to detect the growing need for new tools that help in the process of teaching and learning. This article describes the methodology, the difficulties and the technologies involved in the front-end development of an educational supporting tool. The application aims to optimize the interaction between teacher and students during a lecture through the use of a real-time system that allows users to post and evaluate doubts and answers.

KEYWORDS: Front-End. Web. Education. Teaching.

INTRODUÇÃO

O uso de sistemas de informação e da internet tem sido cada vez mais presente nas mais diversas áreas. Essas ferramentas modificam e otimizam constantemente a forma como realizamos as mais variadas atividades e tarefas do dia-a-dia. O uso das tecnologias é cada vez mais recomendado como aliado dos professores nas atividades pedagógicas pois os alunos devem estar atualizados em relação às novas tecnologias da informação e devem se instrumentalizar para as demandas sociais presentes e futuras (RODRIGUES e ELIA, 2015). Neste contexto, pode-se observar o surgimento constante de novas necessidades e o aumento da demanda por novos recursos tecnológicos. Portanto, faz-se necessário o desenvolvimento de novas ferramentas que apresentem melhorias e novas dinâmicas no processo de ensino e aprendizagem.

O presente trabalho descreve o processo de desenvolvimento *front-end* de uma aplicação *Web* que disponibiliza um canal de comunicação entre os participantes de uma aula expositiva. O sistema oferece novas formas de interação em tempo real entre professor e

alunos de forma dinâmica e intuitiva por meio de um mecanismo de postagem e avaliação de dúvidas e respostas. Ao utilizar os recursos da ferramenta, o professor pode traçar um panorama geral da recepção do conteúdo durante o tempo de aula e os alunos podem expor suas dúvidas de forma imediata e até mesmo anônima. O módulo *front-end* apresentado neste trabalho é responsável por consumir e apresentar os dados de um *Web Service REST*, além de disponibilizar uma interface simples e responsiva para a interação do usuário.

O projeto tem como principal público-alvo as instituições de ensino, professores, palestrantes e alunos que possam vir a se beneficiar das funcionalidades oferecidas pela aplicação desenvolvida. Os requisitos básicos do sistema foram idealizados no início do projeto pelos próprios colaboradores, tomando-se como base os principais problemas e dificuldades enfrentados pelos participantes das aulas ou palestras. Entretanto, conforme a aplicação é utilizada durante testes realizados em aula, novas necessidades têm sido observadas por alunos e professores e novos requisitos têm sido adicionados ao projeto.

MATERIAL E MÉTODOS

A metodologia adotada no gerenciamento do projeto foi baseada na metodologia ágil *SCRUM* (SCRUM ALLIANCE, 2016), portanto a aplicação é desenvolvida de forma incremental durante ciclos de atividades chamados de *sprints*. Durante todo o processo de desenvolvimento, são realizadas reuniões periódicas entre 7 e 15 dias para avaliar as *sprints* realizadas e planejar as próximas atividades a serem cumpridas. A cada *sprint* uma nova versão do sistema é disponibilizada.

Para a implementação do *front-end* foram utilizados os recursos do HTML, do CSS e da linguagem de programação Javascript, o pré-processador de folhas de estilo SASS e os *frameworks* Materialize e AngularJS que serão descritos contextualmente na seção seguinte.

O código-fonte da aplicação é gerenciado e armazenado por meio do sistema de controle de versão Git (GIT, 2016) que permite que todo o histórico de desenvolvimento seja registrado e armazenado de forma segura e organizada.

RESULTADOS E DISCUSSÃO

Este trabalho dá continuidade a um projeto fomentado pelo CNPq realizado através da parceria entre a empresa Siena Idea Ltda e o IFSP - Campus São Carlos. Portanto, inicia-se em um cenário em que o *Web Service* e parte do módulo *front-end* estavam parcialmente implementados por outros desenvolvedores que integram o projeto. Assim, para que o processo de desenvolvimento tivesse prosseguimento, foi necessária a compreensão da estrutura, do código e da lógica da aplicação para que a implementação de melhorias e novas funcionalidades pudesse continuar de forma coerente. Entretanto, foi constatada grande dificuldade na manutenção do código preexistente e no desenvolvimento das novas interfaces de forma padronizada. Entre os problemas encontrados, pode-se citar a difícil compreensão de extensos arquivos CSS e dos métodos Javascript responsáveis pelo acesso aos recursos do *Web Service*, além da dificuldade de construção de novas páginas com o mesmo padrão visual. Dessa forma, optou-se por pela implementação total do módulo *front-end*. A construção da nova versão abrange a troca de *frameworks* CSS, uma total reestilização visual, a reescrita dos controladores que conectam as *views* com os dados do *Web Service* e uma nova forma de realizar o roteamento entre as páginas da aplicação.

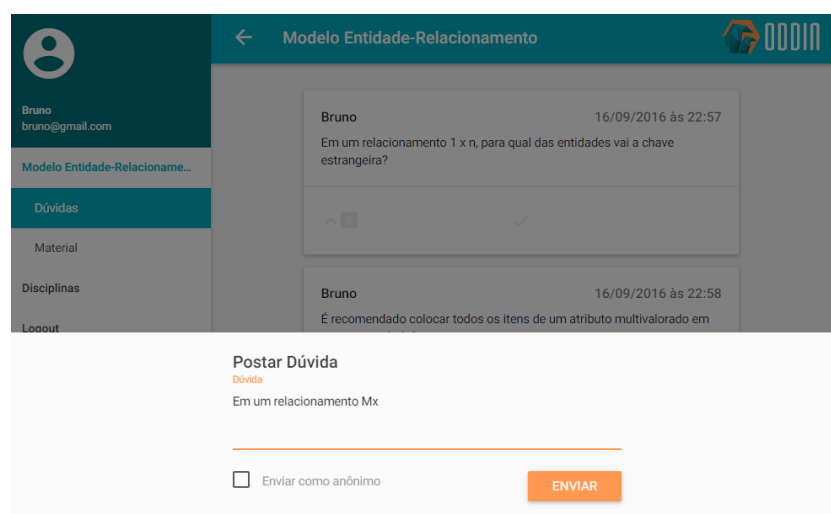
O desenvolvimento de aplicações *front-end* ou *client-side* é realizado em 3 camadas: A Camada de Informação, responsável por definir a estrutura e as informações a serem exibidas; a Camada de Formatação, responsável pela estilização e composição das telas e a Camada de Comportamento, responsável por recursos dinâmicos e interativos da interface.

A primeira etapa do desenvolvimento contemplou a definição da Camada de Informação. Neste processo, foram definidos os componentes que as telas e interfaces

deveriam possuir para que o usuário pudesse visualizar e interagir com o sistema. Esta etapa compreende basicamente a escrita de documentos HTML. Em etapas mais avançadas, os documentos HTML tornaram-se muito extensos, portanto optou-se por utilizar o *Template Engine Pug* (PUG, 2016) para reduzir e simplificar sintaxe do HTML. A etapa seguinte compreendeu a implementação da Camada de Formatação. Nesta etapa foi definido como os elementos seriam apresentados visualmente ao usuário por meio da escrita de folhas de estilo CSS. Para otimizar o processo, optou-se por utilizar um *framework* CSS. Essas ferramentas oferecem recursos como sistemas de *grid*, sistemas de responsividade e componentes previamente estilizados. O *framework* escolhido para este projeto foi o Materialize (MATERIALIZE, 2016), visto que ele oferece uma grande quantidade de recursos e por ser baseado no padrão de linguagem visual definido pela Google, o Material Design. Este padrão define uma série de instruções e parâmetros visuais a serem seguidos para a construção de um Design com alto padrão de usabilidade (MATERIAL DESIGN, 2016). Para facilitar a escrita de arquivos CSS, foi utilizada a ferramenta SASS, um pré-processador CSS que estende os recursos da linguagem. O SASS disponibiliza recursos como variáveis, funções e estruturas de decisão e repetição que permitem a escrita de códigos de estilização mais claros, modularizados e com menor índice de redundância. A estrutura e a aparência da aplicação podem ser vistas na Figura 1.

A etapa seguinte consistiu na definição dos comportamentos e da comunicação com o *Web Service*. A camada de comportamento é implementada através da manipulação dos elementos HTML por meio da linguagem Javascript. Grande parte dos comportamentos da aplicação já estavam definidos pelo uso dos componentes oferecidos pelo Materialize. Portanto, o maior desafio dessa etapa foi a realização de requisições ao *Web Service* e a apresentação e manipulação dos dados a serem recebidos. Para facilitar esse processo, foram utilizados os recursos oferecidos pelo AngularJS (ANGULAR JS, 2016), um *framework* JavaScript que tem como objetivo a simplificação do desenvolvimento de aplicações *Web*. O *framework* oferece mecanismos que facilitam e orientam o trabalho do desenvolvedor. Ao programar dentro da estrutura proposta pelo AngularJS e ao utilizar seus recursos e funcionalidades, o programador é capaz de construir rapidamente uma aplicação mais legível, expressiva, de fácil manutenção e preparada para testes.

Figura 1 – Tela de envio de dúvidas da aula e menu de funcionalidades



Fonte: Elaborado pelos autores.

A comunicação com o *Web Service* foi realizada por meio da troca de dados em formato JSON (IETF, 2014). Os dados e telas de visualização são conectados por meio do uso de controladores que mediam essa comunicação. A aplicação foi desenvolvida em um modelo SPA (*Single Page Application*). Uma SPA oferece uma experiência do usuário mais fluida e agradável pois não requer que o servidor renderize e envie uma nova página a cada requisição, possibilitando que toda a aplicação esteja contida no navegador *Web*. Desta forma, o roteamento entre as páginas e os recursos de navegação são de responsabilidade do cliente.

Um dos grandes desafios encontrados no decorrer do projeto foi o de solucionar as limitações impostas pelo uso de *frameworks*. Embora sejam capazes de aumentar drasticamente a produtividade e facilitar o trabalho do desenvolvedor, a abstração de processos oferecidas pelos *frameworks* podem dificultar a compreensão de mecanismos lógicos responsáveis por diversas funcionalidades da aplicação e conseqüentemente, sua manutenção. Ao oferecerem estruturas de desenvolvimento prontas, os *frameworks* podem se tornar inflexíveis em alguns casos, dificultando consideravelmente a implementação de funcionalidades que não foram previstas em suas definições.

CONCLUSÕES

A aplicação encontra-se em estágio avançado de desenvolvimento. As funcionalidades principais estão implementadas e no início da fase de testes com usuários. O presente trabalho apresentou as etapas, métodos e tecnologias utilizadas no desenvolvimento de uma aplicação *front-end*, bem como os problemas e desafios envolvidos no processo. A aplicação é parte de um sistema mais amplo, cujo funcionamento é resultado da interação entre seus módulos. Portanto, sua efetividade depende do esforço conjunto de todos os colaboradores. Espera-se dar continuidade ao trabalho por meio da disponibilização de novos recursos e a otimização constante das funcionalidades existentes, conforme forem surgindo novas demandas.

AGRADECIMENTOS

À orientadora deste projeto, pela confiança, colaboração e pelos conselhos. À Siena Idea Ltda, por ter oferecido apoio e infraestrutura para o desenvolvimento da aplicação e ao CNPq, pelo apoio financeiro (Processo: 184208/2015-7).

REFERÊNCIAS

ANGULAR JS. Disponível em: <<https://angularjs.org>>. Acesso em: 15 set. 2016.

GIT. Disponível em: <<https://git-scm.com>>. Acesso em: 9 set. 2016.

IETF. **The JavaScript Object Notation (JSON) Data Interchange Format**. Disponível em: <<https://tools.ietf.org/html/rfc7159>>. Acesso em: 15 set. 2016.

MATERIAL DESIGN. Disponível em: <<https://material.google.com>>. Acesso em: 11 set. 2016.

MATERIALIZE. Disponível em: <<http://materializecss.com>>. Acesso em: 11 set. 2016.

PUG. Disponível em: <<https://pugjs.org>>. Acesso em: 9 set. 2016.

RODRIGUES, Cristina A; ELIA, Marcos F. Atividades Extraclasse com base no currículo mínimo para a língua inglesa usando uma Rede Social. **Revista Brasileira de Informática na**

Educação, v.23, n.1, p.148-160, 2015. Disponível em: <<http://www.br-ie.org/pub/index.php/rbie/article/view/2823/2855>>. Acesso em: 16 set 2016.

SCRUM ALLIANCE. **Learn About Scrum**. Disponível em: <<https://www.scrumalliance.org/why-scrum>>. Acesso em: 9 set. 2016.