

# Using Frameworks for Rapid Applications Development as Learning Object for Teaching Web Programming

Daniel S. Costa, Daniella O. Costa  
Instituto de Ciências Exatas e  
Tecnologia - Universidade Federal  
do Amazonas - UFAM  
Itacoatiara-AM, Brasil  
{danieldasilvaa08,  
daniella.ocosta}@gmail.com

Bruno Araújo Bonifácio  
Samsung Instituto de  
Desenvolvimento para a  
Informática da Amazônia - SIDIA  
Manaus-AM, Brasil  
bruno.ab@samsung.com

Bruno Pedraça de Souza  
Programa de Engenharia de  
Sistemas e Computação –  
PESC/COPPE  
Universidade Federal do Rio de  
Janeiro – UFRJ, RJ - Brasil  
bpsouza@cos.ufrj.br

Priscila Silva Fernandes  
Instituto Federal de Educação,  
Ciência e Tecnologia do Amazonas  
– Campus Distrito Industrial  
(IFAM/CMDI)  
Manaus-AM, Brasil  
priscila.fernandes@ifam.edu.br

**Abstract**— This paper describes the use of framework for rapid application development as learning object for teaching Web programming. From the research point of view, we aimed evaluate learning impact in web programming, by students without any previous experience in technology. From an industrial point of view, the goal was evaluating the possibility of use of framework for rapid application development as an efficient solution in real projects, focusing on productivity and code quality. The results show that novice developers are able to quickly learn the concepts and design patterns adopted in the frameworks, and after practicing them effectively in a short time. In addition to indications that this approach is more effective compared to teaching techniques that use traditional development.

**Keywords**— case study, frameworks; agile development education; experimental study, rails, groovy.

## I. INTRODUÇÃO

O processo de globalização da economia e o surgimento constante de novas tecnologias têm influenciado a indústria de desenvolvimento de *software* [1]. Enquanto a ciência e a tecnologia progredem e criam novos paradigmas para atender às exigências da sociedade e do mercado, a educação tenta adaptar-se aos resultados desse progresso para descobrir que rumos tomar e quais os melhores meios de preparar os novos profissionais [6]. Por essa razão, o meio acadêmico tem buscado alternativas para se adequar a estas mudanças [3].

Várias abordagens de ensino têm focado em problemas demandados pelo mercado, como forma de estimular o pensamento crítico, a habilidade de solução e a aprendizagem de conceitos fundamentais, da área de Computação, onde a produção de novas tecnologias acrescenta novos desafios no processo de desenvolvimento [4]. Os centros de ensino da área de Computação, especificamente da área de Engenharia de *Software*, têm buscado criar um ambiente no qual seja possível preparar os estudantes e profissionais para o mercado cada vez mais competitivo. O propósito é formar profissionais com habilidades para construção de aplicações de forma ágil, adotando padrões de desenvolvimento cada vez mais requisitados na indústria [11].

No entanto, grande parte dos estudantes da área ainda apresenta dificuldades para estabelecer padrões, no processo de produção de novas tecnologias, criando barreiras no desenvolvimento de *software* [3]. Nesse contexto, os *frameworks* para desenvolvimento rápido de aplicações têm se mostrado uma alternativa importante no processo de produção de aplicações, especialmente para aplicações *web*. As tecnologias utilizadas para a construção desses *frameworks* possibilitam o aumento da produtividade das equipes de desenvolvimento, maximizam o reuso de artefatos de *software*, estabelecem a padronização dos projetos e reduzem a possibilidade de erros [14].

Empresas como *Google*, *Netflix* e *Amazon* têm adotado os *frameworks* para desenvolvimento rápido dentro de suas plataformas, dado sua eficácia em comparação às abordagens de desenvolvimento tradicionais, sem o uso de qualquer *framework* [13]. Tais características dos *frameworks* podem favorecer o ensino e auxiliar na criação de novas habilidades aos estudantes da área, considerando aspectos técnicos e não técnicos relacionados ao desenvolvimento de *software* [4].

Destaca-se em comum entre abordagens de desenvolvimento utilizando *frameworks*, a possibilidade de reutilização de artefatos de *software* de alta granularidade e a caracterização de reuso de projeto e código. A adoção de *frameworks* rápidos de desenvolvimento de *software*, em conjunto ou isoladamente, pode reduzir relativamente o tempo de desenvolvimento de novas aplicações, assim como a quantidade de código a desenvolver, pois as obrigações propostas passam a ser divididas com os artefatos de *software* reutilizado no desenvolvimento de uma aplicação [18].

Além disso, vários estudos na literatura têm focado a atenção no uso desses *frameworks*, considerando a qualidade de código, padronização do projeto e a produtividade das equipes. Porém, poucos estudos, na academia e na indústria, consideram fatores externos para adoção efetiva dessas tecnologias, tais como o aprendizado e o valor de contribuição para o ensino de programação *web* [5].

O presente artigo apresenta a utilização de um *framework* para desenvolvimento rápido de aplicações, como objeto de aprendizagem para suporte ao ensino de programação *web*. Do

ponto de vista de pesquisa, o objetivo deste estudo foi avaliar os fatores tecnológicos e pessoais que podem impactar na aprendizagem em programação *web*, por profissionais sem qualquer experiência prévia na tecnologia utilizada. Do ponto de vista industrial, o objetivo foi avaliar a possibilidade de utilizar essa prática de desenvolvimento de *software*, como uma solução eficaz em projetos reais, focando em produtividade e qualidade de código. Para a condução da pesquisa foi utilizado o *framework Grails*.

Os resultados mostram indícios que desenvolvedores iniciantes são capazes de aprender rapidamente os conceitos de *web* e padrões de projetos de aplicações deste porte, que são adotados no *framework*, e, depois de praticá-los de forma efetiva em curto tempo. Os resultados mostram ainda que esta abordagem é mais eficaz em comparação com tecnologias de desenvolvimento tradicionais, como *PHP*, *HTML*, *CSS* e *JavaScript*.

O restante do artigo está organizado da seguinte forma: a Seção II apresenta a fundamentação teórica e trabalhos relacionados, que focam em *frameworks* para desenvolvimento rápido de aplicações. A Seção III apresenta a metodologia aplicada para a condução do estudo, o planejamento e os resultados obtidos. A Seção IV descreve as dificuldades e análise dos resultados em relação à experiência dos desenvolvedores. A Seção V mostra as considerações finais e lições aprendidas.

## II. FUNDAMENTAÇÃO TEÓRICA E TRABALHOS RELACIONADOS

### A. Fundamentação Teórica

A programação *web* compõe parte importante nas grades dos cursos de Computação, dada a importância desta área para a formação dos estudantes e demanda do mercado de trabalho [4]. Neste sentido, os conceitos e características do processo de desenvolvimento de aplicações *web* tornam-se elemento fundamental. Porém, o processo de desenvolvimento de aplicações deste porte é uma tarefa complexa e interdisciplinar, uma vez que a definição de métricas e os requisitos não-funcionais acabam tendo grande impacto na condução dos projetos de desenvolvimento, que acaba impactando no início do desenvolvimento das aplicações [7].

Por essa razão, as metodologias de ensino precisam focar nas mudanças tecnológicas como aliado ao ensino de tecnologias *web*, visando à preparação dos estudantes, criando novas habilidades nos aspectos técnicos e não técnicos relacionados ao desenvolvimento de sistemas de *software* [3]. Apesar da adoção de *frameworks* ser substancialmente nova, no processo de criação e na academia, várias destas têm criado iniciativas para desenvolvimento e utilização de *frameworks* deste porte, como forma de padronizar projetos e obter maior qualidade de código [9]. Na área de Tecnologia da Informação (TI), onde o aparecimento constante de novas tecnologias de *software* acaba influenciando no processo de desenvolvimento tradicional, o ensino também tem evoluído [3].

*Frameworks* exploram áreas importantes no desenvolvimento *web* e que são pertinentes para a evolução do desenvolvedor. Um dos aspectos mais explorados na

construção dos *frameworks* atuais tem sido o modelo arquitetural em que se destaca o MVC. A divisão em camadas do MVC permite organizar melhor a aplicação e facilita sua manutenção uma vez que cada camada pode ser modificada sem afetar diretamente as outras [19].

Nesse contexto, torna-se necessário conhecer as tecnologias e ferramentas que o mercado tem adotado, no processo de criação de sistemas interativos. Por isso, os *frameworks* para desenvolvimento rápido de aplicações têm importante destaque como iniciativa para aplicar no ambiente de ensino [15].

### B. Trabalhos Relacionados

Os *frameworks* para desenvolvimento rápido de aplicações surgiram como alternativa de maximizar a produtividade e melhorar a agilidade na produção de artefatos de *software*. O propósito foi diminuir o tempo de configuração e convenções dos projetos, considerando aspectos de qualidade como: usabilidade, segurança, testes, serviços, persistência de dados e apresentação [7]. Os *frameworks* integram um conjunto de pacotes de *software* que visam minimizar o tempo de configuração das aplicações, maximizando a produção do *software* com esforço reduzido se comparado ao desenvolvimento convencional [8].

Além disso, a principal vantagem do uso de *frameworks* para desenvolvimento rápido de aplicações é a sua arquitetura baseada em camadas, que utiliza o padrão MVC (Modelo, Visão, Controle), como forma de modularizar parte da produção de *software*. Esta arquitetura permite auxiliar na geração de componentes distintos e integrados com outras tecnologias *web* tais como: *JavaScript*, *HTML* e *CSS* [10]. A arquitetura em camadas dos *frameworks* simplifica o processo de desenvolvimento e aumentam a produtividade dos desenvolvedores destas aplicações.

Vários estudos na literatura têm focado a atenção na produtividade usando *frameworks* de desenvolvimento. [7] apresenta resultados que mostram que *frameworks* ágeis podem simplificar o processo de desenvolvimento, com um considerável incremento na produtividade. Os autores comparam quatro *frameworks* com diferentes abordagens para reutilização da linguagem Java. De acordo com os resultados aplicações o uso desses *frameworks* pode facilitar a produtividade, além de prover maior facilidade no aprendizado das linguagens de programação que são usadas. [7] relatam ainda a possibilidade de integração no processo de ensino, para auxiliar desenvolvedores a adquirem habilidades para aplicar em ambiente real.

[12] apresentam um comparativo da aprendizagem dos *frameworks*: *Rails*, *Grails*, *Django* e *CodeIgniter* baseado na experiência de uso e aprendizado para desenvolver uma aplicação *web*. Segundo o estudo, *frameworks* baseados na Máquina Virtual Java apresentam menor curva de aprendizagem. Os resultados do estudo, apresentado por [12] mostram que fatores como reutilização de código URLs amigáveis, utilização de *plugins* de autenticação e documentação pode aumentar substancialmente a produtividade. Além disso, os resultados mostram que a

estrutura dos *frameworks* chamada de *ready-to-use*, ou seja, pré-configuradas para uso, permite maior padronização das aplicações.

[16] apresentam uma análise sobre os benefícios e problemas associados ao uso de desenvolvimento rápido de aplicações para *web* em um curto período de tempo através de quatro estudos de casos relacionados ao setor de serviços financeiros. Cada estudo de caso possui um projeto com escopos diferentes, a partir do desenvolvimento dos projetos, foi realizada a análise sobre a produtividade e desempenho das tecnologias utilizadas. O objetivo do trabalho foi investigar o uso do desenvolvimento rápido de aplicações com *Groovy* e *Grails* para permitir que aplicações *web* sejam desenvolvidas dentro de um curto período de tempo. Dois casos da indústria estudos usando uma abordagem com *Groovy* e *Grails*. Outros dois estudos de caso são considerados e comparados: uma aplicação *web* desenvolvida com uma abordagem ágil usando Java e um desenvolvimento em paralelo da funcionalidade da aplicação *web* usando o *Groovy* e *Grails*.

O primeiro projeto (Projeto 1) foi desenvolvido para automatizar os processos CRUD manuais realizados por um grupo de dois desenvolvedores e três clientes com duração de projeto de três meses. O segundo projeto (Projeto 2) envolveu o desenvolvimento de uma aplicação *web* por dois desenvolvedores para três clientes com duração de projeto de 10 semanas. O terceiro projeto ágil (Projeto 3) envolveu a funcionalidade CRUD para criar relatórios financeiros. Consistia de um *front-end* Java que fazia conexão a um banco de dados. Havia três desenvolvedores, um cliente e a duração do projeto foram de cinco meses. Para o quarto projeto (Projeto 4), a funcionalidade para três as iterações do Projeto 3 foram desenvolvidas novamente usando uma abordagem ágil com *Groovy* e *Grails* para determinar se o mesma funcionalidade poderia ser implementada com *Groovy* e *Grails*, se o esforço de desenvolvimento medido em horas poderia ser reduzido usando essas tecnologias, se o tamanho, medido em linhas de código e portanto os esforços de manutenção da aplicação poderiam ser reduzidos e se a qualidade do código medida usando analisadores de código estáticos, poderia ser melhorado *Groovy* e *Grails*.

Os quatro projetos analisados demonstraram que a abordagem ágil com *Groovy* e *Grails* melhorou a produtividade, reduziu o esforço de desenvolvimento e permitiu a realização das aplicações dentro de um prazo relativamente curto. O desenvolvimento paralelo de três iterações do projeto com Java no *Groovy* e *Grails* foi bem-sucedido em mostrar que a mesma funcionalidade poderia ser desenvolvida em uma quantidade de tempo (horas) e com menos linhas de código e com qualidade comparável ou ligeiramente melhor [16].

[17] apresentam um estudo sobre problemas de desempenho enfrentados durante desenvolvimento de projetos implementados em tecnologias Java. Os desafios enfrentados durante o ciclo de vida do projeto e as ações de mitigação realizadas. Para a realização do estudo são comparadas três tecnologias Java e mostra como as melhorias são feitas através de análises estatísticas no tempo de resposta da aplicação. A comparação aconteceu entre projetos de mesmo escopo

desenvolvidos nos diferentes *frameworks*. O modelo base foi desenvolvido e aceito pelo cliente que solicitou o projeto. Depois da aceitação, o cliente colocou ordens de trabalho diferentes. Para cada ordem de serviço, a customização do modelo básico deveria ser feita pelos requisitos. Cronogramas foram definidos para toda e qualquer atividade, incluindo requisitos coleta, desenvolvimento e testes e um plano de trabalho foi feito no início do projeto.

Para realizar a comparação entre as tecnologias, uma equipe trabalhou com a tecnologia *Java/servelet* e a segunda trabalhou com o *Spring Framework*, ambas com os mesmos projetos, já havendo um projeto base feito em *Grails* para posterior comparação com as aplicações desenvolvidas nas outras tecnologias. Para a equipe um foi usado um servidor de banco de dados *Oracle* foi escolhido no lugar do *MySQL*. Um estudo comparativo sobre o tempo de execução da consulta foi realizado na definição do servidor e do banco de dados. Para a equipe dois foi utilizado o *Spring Framework* que fornece integração com o *Hibernate* em termos de gerenciamento de recursos, suporte à implementação do DAO e estratégias de transação e com servidor e banco de dados iguais ao da primeira equipe.

[17] afirmam que a comparação das três tecnologias, nomeadamente *Java/servelet*, *Spring framework* e *Grails* para desempenho nos levou ao resultado que *Grails* é uma plataforma de melhor desempenho para o projeto que assumiu. Coisas como feeds RSS e modelagem de domínio permitem um desenvolvimento mais rápido da aplicação, permitindo que o foco esteja no código funcional. O sistema através de várias otimizações mostrou uma melhoria geral de 84% no tempo de resposta e 93% no padrão desvios. Enquanto nas atividades mais recentes, o sistema não apresentou problemas de desempenho e os servidores funcionaram sem problemas, sem qualquer tempo de inatividade.

Segundo [9], o uso de convenções pré-estabelecidas sobre configuração, usadas pelos *frameworks* é a característica principal para aumentar a produtividade do processo. Além disso, o uso desta tecnologia pode diminuir radicalmente o tempo que o desenvolvedor gastaria para integrar e configurar todos os módulos de aplicações *web*.

O *Grails* é um *framework* de desenvolvimento baseado em *Java*, que utilizalinguagem orientada dinâmica, por meio da linguagem *Groovy* [12]. Devido às características do *Groovy*, a curva de aprendizado é menor em relação a outros *frameworks* [11]. No entanto, essa particularidade é equilibrada com a integração com tecnologias de persistência de dados e segurança, tornando-o uma alternativa para criação de aplicações de forma rápida.

Por essa razão, escolheu-se o *Grails*, para condução do estudo, no contexto de um projeto de desenvolvimento. Para isso, foi realizado um acompanhamento sobre os estudantes e profissionais envolvidos, desde o processo de ensino do *framework* até o desenvolvimento de aplicações reais. A próxima Seção descreve a condução do estudo realizado.

### III. ESTUDO DE CASO

Uma das disciplinas importantes na formação tecnológica dos cursos de Computação, é a programação *web* onde são apresentados ferramentas e padrões para o desenvolvimento de aplicações deste porte. Apesar do esforço para ensino desta disciplina, utilizando tecnologias tradicionais, como: *HTML*, *CSS*, *JavaScript* e *PHP*, há relatos de grande dificuldade de aplicação dos conceitos da área, podem impactar no processo de criação de aplicações *web*, tais como: padrão e design arquitetural, requisições dentre outros [4].

Diante da falta de utilização de modelos que facilitassem o uso das tecnologias aplicadas em sala de aula, acabou criando barreiras para aplicação das tecnologias e conceitos na construção de uma aplicação *web*, em contexto real. Diante desse cenário, foi proposta a utilização do *framework Grails* para desenvolvimento rápido de aplicações como ferramenta pedagógica na construção de aplicações *web*. A escolha foi em razão da curva de aprendizado e da facilidade de integração com outras tecnologias [12].

O estudo foi conduzido dentro de um projeto de capacitação, executado em paralelo a disciplina de programação *web*. Assim os estudantes, selecionados por conveniência no papel de desenvolvedores, foram apresentados aos conceitos de programação *web* e motivados a construir uma aplicação *web*, considerando duas abordagens: 1) Desenvolvimento tradicional: usando *PHP*, *HTML*, *JavaScript* e *CSS* e; 2) Desenvolvimento rápido: usando o *framework Grails*.

Vale salientar que os grupos receberam o mesmo treinamento, de ambas as abordagens. Como aspecto inovador, o objetivo foi comparar o uso do *Grails* para ensino de programação e o desenvolvimento usando tecnologias de desenvolvimento tradicionais, como *PHP*, *HTML*, *CSS* e *JavaScript*, em contexto de desenvolvimento de aplicação real.

A hipótese considerada é que o uso de *frameworks* ágeis pode auxiliar os desenvolvedores de *software* a disseminar o conhecimento sobre a tecnologia aprendida, bem como promover o surgimento e a maturação de novas habilidades. Neste sentido, os participantes foram encorajados a desenvolver a mesma aplicação, proposta na disciplina de programação *web*, mas utilizando o *framework Grails*.

#### A. Planejamento do Estudo

Para melhor compreensão do escopo abordado nessa pesquisa, o objetivo desse estudo foi descrito de acordo com o paradigma GQM – *Goal Question Metric* [2], que foi **analisar** a produtividade de desenvolvedores usando *framework Grails* **com o propósito de caracterizar em relação ao** tempo de implementação, a percepção do aprendizado e do valor de contribuição do *Grails*, **do ponto de vista** dos engenheiros de *software*, **no contexto de** ensino e produção de aplicações *web*.

Todos os participantes possuíam conhecimento prévio em tecnologias *web* e eram alunos da disciplina de programação *web*. O projeto foi realizado em conjunto a disciplina de programação *web*, no entanto usando o *framework* para ensino e aplicação dos conceitos da área. O objetivo final da disciplina e do projeto foi que com os conhecimentos

aprendidos os alunos desenvolvessem uma aplicação *web*. Ao final do estudo foi feita uma análise qualitativa para comparar a efetividade das duas abordagens de ensino.

Adicionalmente, foram elaborados documentos para garantir a segurança dos dados de cada aluno como: 1) Termo de Consentimento Livre e Esclarecido (TCLE): para garantir a confidencialidade dos dados e informações dos participantes; 2) Entregas e marcos: com as descrições do que cada problema deveria apresentar como parte da entrega em cada ao longo do curso e; 3) Questionários de experiência de desenvolvimento: para capturar as percepções de cada discente.

TABELA I – CARACTERIZAÇÃO DOS PARTICIPANTES DO ESTUDO.

Características	Questões	P	Exp.
Conhecimento em Tecnologias <i>web</i>	Não possuo nenhum conhecimento e/ou experiência em desenvolvimento de aplicações <i>web</i> ,	0	
	Tenho algumas noções de desenvolvimento de aplicação adquirida através de aula, livro e trabalhos acadêmicos	6	
	Tenho conhecimento em desenvolvimento de aplicação adquirida através de treinamentos profissionais	7	
Experiência em Desenvolvimento de Aplicações <i>web</i> , sem uso de <i>frameworks</i>	Não tenho nenhuma experiência no desenvolvimento de aplicações <i>web</i>	3	
	Tenho apenas experiência acadêmica no desenvolvimento de aplicações <i>web</i> adquirida através de aula, projeto (s) de pesquisa e/ou acadêmicos	5	
	Tenho anos de experiência no desenvolvimento de aplicações <i>web</i> , adquiridas em projetos profissionais	5	2 anos ou +
Conhecimento em <i>Frameworks</i> de desenvolvimento rápido	Não tenho nenhum conhecimento sobre os <i>frameworks</i> ágeis	8	8 anos ou +
	Tenho apenas conhecimento e noções acadêmicas no desenvolvimento de aplicações <i>web</i> adquiridas através de aula, projeto (s) de pesquisa e/ou acadêmicos	4	4 anos
	Tenho conhecimento em desenvolvimento de aplicação adquirida através de treinamentos profissionais	1	1 ano
Experiência em Desenvolvimento usando <i>Frameworks</i> de desenvolvimento Rápido	Não tenho nenhuma experiência em desenvolvimento, usando <i>frameworks</i> ágeis	1 0	
	Tenho apenas experiência acadêmica no desenvolvimento de aplicações usando <i>framework</i> ágil adquirida através de aula, projeto (s) de pesquisa e/ou acadêmicos	2	
	Tenho anos de experiência no desenvolvimento usando <i>framework</i> ágil	1	2 anos ou +
Legenda: P = Participante - Exp. = Grau de Experiência			

## B. Execução do Estudo

Para a execução do estudo, foi realizado um ciclo de treinamentos, com conceitos básicos, tanto para utilização da abordagem tradicional quanto para utilização do *framework Grails*. O treinamento foi ministrado durante duas semanas, com oito participantes, pelo mesmo professor para minimizar as ameaças à validade do estudo. No processo de ensino, foram aplicados conceitos preliminares de tecnologias *web*, linguagem *Groovy* e *MVC* aplicadas no *framework*. Para maior controle a disciplina aplicava os mesmos conceitos, porém usando tecnologia *PHP*.

Para contextualização do cenário real, cada participante foi inserido em demandas de desenvolvimento de aplicações que seriam usadas e implantadas no próprio Instituto de Ensino. Para comparação das duas abordagens foram divididos em dois grupos. Dos quais, quatro alunos desenvolveram, para comparação, duas funcionalidades usando *Grails* e outros de forma tradicional, usando *PHP*, *CSS*, *JavaScript*.

TABELA II – DINÂMICA PARA COMPARAÇÃO DAS ABORDAGENS DE ENSINO.

Abordagens	Grupo 1	Grupo 2	Tarefas
Framework Grails	X		Cadastro de usuários, Autenticação
Tradicional ( <i>PHP</i> , <i>HTML</i> , <i>CSS</i> , <i>JavaScript</i> )		X	
Framework Grails		X	Cadastro de usuários, Autenticação
Tradicional ( <i>PHP</i> , <i>HTML</i> , <i>CSS</i> , <i>JavaScript</i> )	X		

Cada projeto foi desenvolvido em pares, como forma de melhorar a qualidade de código produzida, por meio de revisões e testes. Após cada módulo do treinamento e entrega foi coletado dado dos participantes, por meio de questionário, onde foram definidas questões considerando as dimensões de Experiência de Aprendizado em Desenvolvimento [5]. Esta abordagem é similar ao conceito do ciclo de UX, no entanto focado no elemento desenvolvedor, no aprendizado de uma nova tecnologia. Assim, a análise é dúvida em três dimensões: percepção, afeição e expectativa no contato de uma nova tecnologia.

Segundo [5] a *dimensão de percepção* está relacionada aos fatores de infraestrutura percebidos pelo participante no processo de criação de *software*, estes podem ser ferramentas utilizadas, modelos de processos. Por sua vez a *dimensão afetiva* trata os fatores afetivos sociais, à percepção do artefato produzido, aos sentimentos com os envolvidos no processo. E a *dimensão de expectativa* está relacionada com o grau de alinhamento aos objetivos e perspectiva do participante em relação aos objetivos do projeto de desenvolvimento.

## IV. RESULTADOS OBTIDOS

Ao final, foram disponibilizadas oito aplicações, quatro realizadas usando o *framework Grails*, e quatro usando tecnologias tradicionais da disciplina. Para comparação foi

feita uma análise de tempo e maturidade da aplicação para as duas tarefas executadas. Assim, a partir de um protótipo de telas, contendo as funcionalidades da Tabela 2, foram elaboradas as funcionalidades. Os resultados mostram que o tempo usando *Grails* foi mais eficiente, conforme a Figura 1. Os resultados da caracterização e dos questionários estão disponíveis no *google forms*<sup>1</sup>.

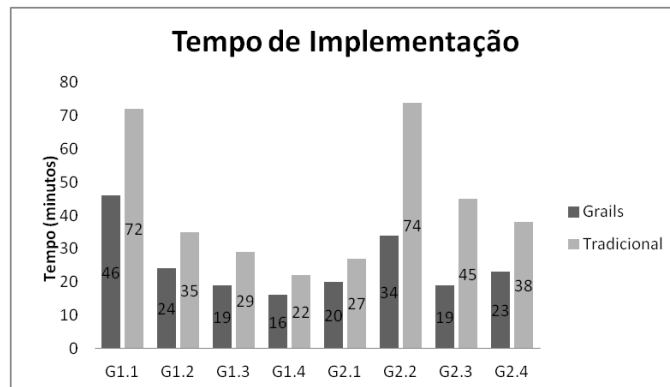


Fig. 1. Tempo médio de desenvolvimento das abordagens de ensino.

Considerando o tempo para desenvolvimento das funcionalidades de autenticação e cadastro de usuários, a abordagem usando *Grails* foi de média 25 minutos (G1), enquanto a abordagem usando tecnologias tradicionais foi de cerca de 42 minutos (G2). A Figura 1 mostra o desempenho individual por grupo e participante. Os quatro participantes do grupo 1 (G1), que utilizaram o *Grails*, obtiveram menor tempo, e maior efetividade na produção das atividades monitoradas. Por sua vez, os participantes do grupo 2 (G2), obtiveram maior tempo. Grande parte da demanda para esse resultado, se deve ao fato de ter sido realizada a implementação dos elementos do *front-end*, na camada de interface da aplicação. Aplicações em *Grails* possuem componentes prontos, o que permite maior eficiência na geração dos elementos funcionais em relação a abordagem que usa tecnologias sem *frameworks* para auxiliar a produção de aplicações.

Após coletar os dados quantitativos vistos na Figura 1, foram obtidos resultados qualitativos em relação a experiência dos participantes a partir de entrevistas e questionários aplicados aos mesmos. Dessa forma, para avaliar a experiência do aprendizado foram elaboradas questões relacionadas ao ciclo DX, categorizadas em: **1) Percepção sobre a infraestrutura:** consiste em estabelecer a infraestrutura e os objetos utilizados influenciam no aprendizado e utilização da nova tecnologia; **2) O sentimento sobre o trabalho realizado:** grau de motivação que a tecnologia utilizada trouxe para o aprendizado e produção de aplicações; **3) Percepção do valor do aprendizado:** consiste em avaliar a relevância da nova tecnologia para o desenvolvimento de aplicações *web*.

<sup>1</sup><https://docs.google.com/forms/d/1R4zR0aH6CL9P15sgv2ETcPtrU92h1BKvjNXNBtqE3cU/edit>

**Sobre a percepção da infraestrutura** cerca de 75% (06 participantes) dos participantes afirmaram satisfação positiva com a infraestrutura e recursos utilizados ao longo do treinamento, enquanto apenas 12,5% (01 participantes) afirmaram ter situação regular ou insatisfatória. Para justificar a escolha, um participante afirmou "*O laboratório apresenta máquinas com usuário compartilhado, isso tem prejudicado o andamento do projeto*".

**Em relação à tecnologia utilizada**, os resultados apontaram que 87% (07 participantes) dos participantes mostraram-se muito satisfeitos com a utilização do *framework* para desenvolvimento *web*, enquanto outros 12,5% (01 participantes) afirmaram estar parcialmente satisfeitos. Para coletar informações adicionais em relação a infraestrutura, foi realizada uma análise sobre a percepção do uso de objetos de ensino e recursos didático pedagógicos utilizados em sala. Os dados ainda mostram que 50% (04 participantes) concordaram com a didática e os objetos apresentados como forma de ensino ao longo do treinamento, enquanto 37% (03 participantes) consideraram boa a utilização dos recursos e 12,5% (01 participantes) considerou regular.

**Quanto ao valor de contribuição** os participantes afirmaram que o conteúdo agregou conhecimento. Cerca de 75% (06 participantes), afirmaram que o conteúdo abordado aprimorou seus conhecimentos. Os dados foram importantes para aprimorar a didática e a ordem dos conteúdos apresentados ao longo do processo de ensino, feito no curso executado.

Outro ponto identificado foi o de **valor de conhecimento**. Até mesmo os que tinham experiência com programação se sentiram confortáveis com os conceitos aplicados, como forma de aprimorar os conhecimentos já obtidos. Sobre a percepção de aprendizado a maior parte dos participantes conseguiu entender os conceitos aplicados para o paradigma *web*. Cerca de 87,5% (07 participantes) afirmaram que os conceitos aplicados foram excelentes para aprimorar o conhecimento que os mesmos já possuíam antes do curso, enquanto 12,5% (01 participantes) afirmaram que o conteúdo ministrado nas aulas foi bom, assim em totalidade não houve afirmações negativas.

Cada aplicação foi implantada nas respectivas instituições interessadas. Essa iniciativa permitiu a criação de um ecossistema de *software* do Instituto de Ensino e de outras cidades da região, localizadas na região do Médio Amazonas. Diante do conteúdo aplicado para introduzir os conceitos relevantes para construção das aplicações, foi possível identificar o quanto a experiência agregada à prática.

Esta abordagem teórico-prática pode ajudar aos participantes, na melhor absorção dos conteúdos e também no desenvolvimento de habilidades para resolução de problemas, além de qualificá-los para o mercado de trabalho. Apesar disso, foram enfrentadas dificuldades no decorrer deste projeto, mas que foram úteis para que fossem aplicadas melhorias durante o processo das três dimensões do ciclo DX, descritas abaixo.

#### A. Ameaças a Validade do Estudo

Nesta Seção são descritas as possíveis ameaças à validade deste estudo. Pesquisadores da área tem adotado em estudos experimentais maior controle sobre as ameaças que podem influenciar os resultados [3; 4; 6]. As ameaças são classificadas em quatro categorias descritas a seguir: validade interna, validade externa, validade de conclusão e validade de constructo.

*Validade interna*: este tipo de ameaça foi tratado quanto à instrumentação adequada para o estudo. Teve-se todo o cuidado na seleção e criação dos questionários, ferramentas de desenvolvimento e materiais adequados. Em relação a seleção dos participantes, essa seleção não foi realizada de maneira por conveniência, entre alunos de disciplina e participantes do projeto, o que faz representar uma ameaça ao estudo.

*Validade externa*: essa ameaça requer cuidado na escolha da amostragem e população do estudo, tempo e configuração do experimento. Em relação a população, essa ameaça foi cuidada na escolha dos participantes, apenas poderia participar do estudo quem estava dentro do escopo da pesquisa (alunos da disciplina e do projeto). Tem-se que assumir que o estudo foi realizado apenas numa unidade, o que não se pode dizer que a população é representativa. No entanto, o ambiente do experimento é compatível com o ambiente real.

*Validade de conclusão*: neste tipo de ameaça, os dados coletados foram tratados de modo que não haja generalização da pesquisa. Sabe-se que a amostra não é ampla e conclusiva, porém, o estudo mostra indícios que o uso dos *frameworks* ajuda bastante na aceitação de uma tecnologia como abordagem de ensino de programação *web*.

*Validade de Constructo*: Problemas relacionados à ameaça de generalizar os resultados obtidos do estudo à teoria que o sustenta. Quanto aos fatores humanos, os participantes não estavam envolvidos em outros tipos de estudos, além disso os pesquisadores utilizaram estudantes pré-formados, ou seja, poderiam representar o papel de desenvolvedores.

#### V. CONCLUSÕES E PERSPECTIVAS FUTURAS

Este artigo apresenta o uso de *frameworks* para desenvolvimento rápido de aplicações *web* como objeto de aprendizagem, no ensino de programação *web*. A experiência adquirida pelos alunos que participaram do projeto permitiu que os mesmos no decorrer da prática fossem identificando desafios e soluções durante a realização do projeto.

Além da possibilidade de visualizar como a experiência prática em um projeto real ajuda a ter um melhor entendimento das teorias, práticas, métodos, processos e ferramentas que envolvem a programação *web*. Os resultados qualitativos, feitos em entrevistas, durante o processo de ensino, o uso da abordagem se mostrou prática, misturando a aplicação dos conceitos teórico-práticos que tornaram possível a criação de aplicações *web*.

Como uma das principais limitações do estudo está a falta de tempo para realização do projeto. Assim limitando a apenas o uso de um *framework* específico e ainda a parte prática dos conceitos abordados nas aulas com curto prazo para

desenvolvimento. Além da falta de comparativo com as tecnologias sem o uso de *framework*.

Apesar disso, a grande parte dos alunos obteve êxito na disciplina ao qual o desenvolvimento da aplicação *web* se destinava. Outra contribuição advinda do curso foi o aproveitamento de alguns dos sistemas desenvolvidos para a execução do Trabalho de Conclusão de Curso. Diante disso, acredita-se que um estudo no contexto maior, de uma disciplina obrigatória possa auxiliar na identificação de dificuldades em relação à aplicação da metodologia de ensino.

No decorrer do projeto foi possível perceber as dificuldades enfrentadas e adaptar a metodologia de desenvolvimento, visando melhorar as habilidades dos participantes, potencializando as chances de os mesmos entrarem no mercado de trabalho. Espera-se, contribuir para disseminar o uso *frameworks* ágeis no processo de ensino e aprendizagem da programação *web*, buscando proporcionar melhor qualificação dos profissionais, por meio da teoria aliada à prática.

## VI. AGRADECIMENTOS

Os autores agradecem aos participantes que se disponibilizaram gentilmente em participar desse estudo, bem como a Universidade Federal do Amazonas por ceder a estrutura necessária.

## Referências

- [1] K. Bergner, A. Rausch, M. Sihling and T. Ternit, —DoSAM - Domain-Specific Software Architecture Comparison Model, In the Proc. of the International Conference on Quality of Software Architectures, pp. 4-20, 2005.
- [2] V. Basili, G. Caldiera, H. D. Rombach, “The Goal Question Metric Paradigm”, Encyclopedia of Software Engineering (Marcianik, J.J., editor) Volume 1, John Wiley & Sons, 1994, pp. 578 – 583.
- [3] L. Cavalcante, B. Bonifácio, P. S. Fernandes. Aprimorando aplicações Web por meio de avaliação de usabilidade: um estudo de caso em uma instituição de ensino superior. In: XV WWW/Internet e Computação Aplicada 2015, p. 11 – 18, Florianópolis, 2015.
- [4] H. Farias, Ferreira, R. Avaliando o Uso da Ferramenta Scratch para Ensino de Programação através de Análise Quantitativa e Qualitativa. In: XXVI Simpósio Brasileiro de Informática na Educação, Maceió (SBIE 2015), p. 947-956, 2015
- [5] F. Fagerholm, and J. Münch, “Developer experience: concept and definition”. In: Proceedings of the International Conference on Software and System Process, 73-77, 2012.
- [6] P. S. Fernandes, B. A. Bonifácio, T. Conte, A Usability Inspection Approach to Assist in the Software Development Process. In The Ninth International Conference on Software Engineering Advances, Nice, França, p. 651-657, 2014.
- [7] J. Ignácio, D.L. Casillas, and A. C. Iglesias, A Comparison Model for Agile Web Frameworks. In Euro American Conference on Telematics and Information Systems, EATIS’08. ACM New York, NY, p. 28-36, 2009.
- [8] E. Ishikawa and C.G. Ralha Ensino de Sistemas de Informação em Cursos de Computação: Relato de experiência com uso de abordagem prática em TIC. In XI Brazilian Symposium on Information System, Goiania, Brasil, p. 571-578, 2015.
- [9] A. N. Neto, A. P. Silva, I.I. Bittencourt, Uma análise do impacto da utilização de técnicas de gamificação como estratégia didática no aprendizado dos alunos. In XXVI Simpósio Brasileiro de Informática na Educação (SBIE 2015), p. 667 – 676, 2015.
- [10] F. P. Martins, Desenvolvimento de aplicações corporativas com Grails. Treinamento avançado em desenvolvimento de sistemas. Acesso em 05/02/2017. Disponível em: <http://fibodev.com.br>, 2012.
- [11] M. A. C. Meireles, e B. A. Bonifácio Uso de Métodos Ágeis e Aprendizagem Baseada em Problema no Ensino de Engenharia de Software: Um Relato de Experiência. In XXVI Simpósio Brasileiro de Informática na Educação, p. 180 – 189, 2015.
- [12] D. Díaz Clavijo. A practical comparison of agile web frameworks. Universidad de Las Palmas de Gran Canaria. 2014.
- [13] S. Schneider. Netflix, Amazon, Asgard, Groovy, & Grails: Elastically Scaling Apps for 40+ Million Internet TV Viewers. Disponível em: <https://content.pivotal.io/blog/netflix-amazon-asgard-groovy-grails-elastically-scaling-apps-for-40-million-internet-tv-viewers>. Acesso em: 27.05.2013.
- [14] A. Pereira, V. Cogo, A. Charao. Frameworks para Desenvolvimento Rápido de Aplicações Web: um Estudo de Caso com CakePHP e Django. In Proceedings of the Workshop de Software Livre. Porto Alegre, Rio Grande do Sul. 2009.
- [15] J. Martin. Rapid Application Development. Macmillan Publishing Co., Inc., 1991.
- [16] K. Malone, J. Griffith. A Case Study in the Use of Groovy and Grails. In: Proceedings of the 27th Annual ACM Symposium on Applied Computing. ACM, 2012. p. 1254-1255.
- [17] P. Dutta, V. Gupta, S. Rana. Performance Comparison on Java technologies - A Practical Approach. Journal Computer Science & Information Technology. DOI : 10.5121, 2013, pp. 350 – 357.
- [18] R. Silva. Suporte ao desenvolvimento e uso de frameworks e componentes. Tese de Doutorado. Universidade Federal do Rio Grande do Sul. 2000.
- [19] E. Da Silva, D. De Abreu. Um Framework de Componentes para o Desenvolvimento de Aplicações Web Robustas de Apoio à Educação. In: Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE). 2004. p. 158-167.