

Versionamento de Projeto na Prática com Git e GitHub: Um Relato de Experiência do Curso Ofertado pelo Projeto LearningLab No Interior Cearense

Francisco Adam Andrade¹, Israely Lima¹, Letícia M. Puga¹,
Laura M. Puga¹, Aline H. A. Barros¹, Ana Livia N. Ramos¹,
Jacilane de H. Rabelo²

¹Projeto de pesquisa, extensão e desenvolvimento LearningLab – Universidade Federal do Ceará - Campus Russas (UFC - Campus de Russas)
Caixa Postal 62.900-000 – Russas – CE – Brazil

{adam.andrade, israelylima, leticiademirandapuga,
laurapuga, alinealvessr, livianascimento}¹@alu.ufc.br,
jacilane.rabelo²@ufc.br

Abstract. *Version control in software development is crucial to ensure quality and avoid errors in integration. With this purpose, the LearningLab project at the Universidade Federal do Ceará - Campus de Russas offered a course on Git and Github, aiming to empower computer science students. The course had 34 students participating. The results reveal that the majority of students intend to apply the concepts learned. Furthermore, students expressed satisfaction with the practical teaching methodology and the learning dynamics provided by the Kahoot! tool.*

Resumo. *O controle de versões no desenvolvimento de software é crucial para garantir qualidade e evitar erros na integração. Com esse propósito, o projeto LearningLab da Universidade Federal do Ceará - Campus Russas ofereceu um curso sobre Git e Github, visando capacitar estudantes de computação. O curso teve a participação de 34 alunos. Os resultados revelam que a maioria dos alunos pretende aplicar os conceitos aprendidos. Além disso, os alunos expressaram satisfação com a metodologia prática de ensino e a dinâmica de aprendizado proporcionada pela ferramenta Kahoot!.*

1. Introdução

O mercado da Tecnologia da Informação (TI) está em constante busca por profissionais qualificados [de Sousa Pereira and Saqui 2023]. O ensino de programação, por sua complexidade e novidade para alunos, é desafiador [Widthauper 2020]. Além disso, ferramentas importantes para a qualidade do desenvolvimento, como o controle de versão, são frequentemente negligenciadas nas disciplinas tradicionais. Nesse sentido, segundo [Gonçalves and Scaico 2019], a preparação de um futuro profissional enfrenta consideráveis obstáculos, sobretudo quando se depara com currículos desatualizados, que seguem práticas que já não são adequadas para atender às necessidades do mercado de trabalho. Nas instituições de ensino de computação, nota-se a falta de atualização dos conteúdos para atender às demandas do mercado de trabalho, o que resulta em uma lacuna na preparação dos estudantes para o cenário profissional [de Sousa et al. 2021]. Portanto, é preciso oferecer cursos sobre ferramentas que auxiliam o desenvolvimento, para

assim formar profissionais mais qualificados. Tais ferramentas facilitam o desenvolvimento, revisão e compartilhamento de código, contribuindo para aprimorar a eficiência e a qualidade do trabalho em equipes de desenvolvimento de *software*. Os autores, [Braught et al. 2023], destacam a importância do controle de versão na garantia de qualidade do *software*, assim com mais chances de atender as necessidades do cliente. A integração de módulos por diferentes desenvolvedores em empresas tradicionais pode causar problemas, tornando o controle de versão essencial para a recuperação de versões anteriores após quebras no código [Oliveira et al. 2021].

Um grande desafio no ensino de programação e nas ferramentas utilizadas na área é motivar os estudantes a se envolverem nas atividades e aplicarem os conceitos aprendidos em sala de aula de maneira eficaz na vida real [da Silva et al. 2021]. Uma abordagem usada para superar essa dificuldade, focando no ensino com base em casos práticos, onde o estudante desempenha um papel principal, é a abordagem de Aprendizagem Baseada em Projetos (ABPj) [Santiago et al. 2023]. Nesse contexto, o projeto *LearningLab* (Laboratório de Ensino, Pesquisa e Desenvolvimento de Tecnologias Alinhadas à Gestão do Conhecimento e Inovação em Processos de *Software*) da Universidade Federal do Ceará, Campus Russas (UFC-Russas) utiliza a abordagem (ABPj) para oferecer cursos que vão além da teoria, priorizando a prática. Esses cursos simulam projetos do mercado de trabalho, permitindo que os discentes desenvolvam suas habilidades interpessoais e estejam melhor preparados para enfrentar os desafios no cenário real.

Este trabalho tem como finalidade relatar as experiências e lições aprendidas no ensino da ferramenta *Git* e *GitHub*. De acordo com [Blischak et al. 2016], *Git* é um sistema para controle de versão distribuído, enquanto o *GitHub* é uma plataforma na nuvem que abriga o sistema de versão *Git*. O curso adotou uma metodologia que enfatizou a entrega do conteúdo de forma prática, acompanhando de uma avaliação final. A análise qualitativa dos dados do curso foi realizada por meio do Método *Grounded Theory* (GT) [Jarczyk et al. 2018], que proporcionou a identificação de aspectos favoráveis e que possibilitaram melhorar a execução dos cursos futuros. Já a análise quantitativa foi feita seguindo a métrica *Intrinsic Motivation Inventory* (IMI) [Choi et al. 2010], que possibilitou percepções valiosas sobre o nível de motivação dos alunos, ao aprender sobre *Git* e *GitHub*.

Além desta introdução, este trabalho segue a seguinte estrutura organizacional: na Seção 2, serão discutidos os trabalhos relacionados; na Seção 3, será detalhada a metodologia; na Seção 4, será mostrada a realização do curso; na seção 5, serão apresentados os resultados alcançados juntamente com as discussões pertinentes; na Seção 6, serão apresentados os desafios encontrados; e finalmente, na Seção 7, serão apresentadas as conclusões deste estudo, bem como a perspectiva para futuras pesquisas.

2. Trabalhos Relacionados

No contexto dos estudos relacionados, serão apresentadas duas pesquisas destacadas no âmbito do Congresso da Sociedade Brasileira de Computação (CSBC), com base nos trabalhos apresentados no *Workshop* sobre Educação em Computação (WEI). [Zhu and Wei 2019] analisam o problema de múltiplos nomes e *e-mails* (MNE) em repositórios *Git*. Os pesquisadores coletaram dados de 450 repositórios *Git* do *GitHub*, examinaram padrões e impactos do MNE, identificando que cerca de 6% dos desenvolvedores usaram MNE em mais de 60% dos repositórios, contribuindo com metade dos

commits. Eles descobriram padrões de MNE, como variação de caixa e recombinação de segmentos de nome e *e-mail*, e exploraram a relação entre MNE e idade/popularidade dos repositórios, bem como tempo de permanência dos desenvolvedores e número de *commits*. O *Git* e o *GitHub* desempenharam papéis essenciais neste estudo, fornecendo dados de histórico de *commits* e informações sobre os desenvolvedores. O uso do *Git* permitiu a extração detalhada de informações, incluindo nomes e endereços de *e-mail* dos desenvolvedores, enquanto o *GitHub* possibilitou o acesso aos repositórios necessários para a pesquisa. A capacidade de acessar esses dados foi fundamental para analisar padrões de MNE nos repositórios *Git*.

A pesquisa conduzida por [Silva et al. 2019], destaca a eficácia da ferramenta *Kahoot!* no ensino de conceitos de programação. O estudo foi realizado no Instituto Federal Sertão-PE, campus Petrolina-PE, em uma turma de jovens entre 13 a 14 anos, na qual ao todo participaram 30 alunos do curso. A ferramenta *Kahoot!*¹ foi empregada para reforçar os conhecimentos adquiridos ao longo do curso. Após cada aula, um jogo de perguntas e respostas no *Kahoot!*¹ foi aplicado, proporcionando maior envolvimento dos alunos nas atividades do curso. Posteriormente, os pesquisadores elaboraram um questionário de *feedback* para realizar análises quantitativas e qualitativas. A avaliação dos dados, vindo do *feedback* dos alunos, revelou um aumento na motivação e engajamento dos estudantes em relação aos conceitos de programação.

O estudo atual e o de [Silva et al. 2019] compartilham abordagens similares, mas se diferenciam na análise de *feedbacks*, utilizando *Grounded Theory* e interpretação intuitiva, respectivamente. Em comparação, o trabalho de [Zhu and Wei 2019] destaca a importância do controle de versões usando *Git* e *GitHub* na prática, enquanto o presente estudo amplia essa discussão para enfatizar a relevância geral de aprender e trabalhar com controle de versões.

3. Metodologia

A metodologia utilizada para implementar este estudo é dividida em seis passos: (1) Selecionar o tema de interesse do público a partir de relatos dos formulários de *feedback* e estudo de ferramentas mais atuais no mercado de trabalho; (2) Definir instrutores e planejar a ementa e materiais para o curso; (3) Realizar Treinamento Interno; (4) Divulgar curso nos cursos de Computação; (5) Realizar a seleção de alunos inscritos no curso com base nos pré-requisitos definidos para a participação no curso; (6) Realizar curso; (7) Coletar e Analisar dados de *feedback*. Essa metodologia tem como principal objetivo, compartilhar conhecimentos sobre *Git* e *Github*, de forma prática e objetiva para os alunos.

1 - Seleção do tema do curso: A equipe do projeto analisou os formulários de *feedback* de cursos anteriores para identificar os temas de interesse dos alunos. Considerando também a alta demanda por habilidades relacionadas ao *Git* e *Github* no mercado de trabalho, foi escolhido o conteúdo do curso.

2 - Definição de instrutores e planejamento de curso: Instrutores foram selecionados do projeto, considerando seu interesse. Após seleção, receberam treinamento para melhorar suas habilidades. Em conjunto com a coordenadora, desenvolveram a ementa, referenciando cursos anteriores, detalhando aulas, conteúdos e atividades. Também criaram materiais complementares, como vídeos e *slides*, com apoio dos monitores.

3 - Realização do Treinamento Interno: Foram realizadas reuniões para apoiar os ins-

trutores, fornecendo-lhes materiais necessários e conduzindo treinamentos internos. Essas sessões serviram para avaliar tanto os conteúdos quanto a didática empregada pelos instrutores.

4 - Divulgação do curso nos cursos de Computação: Após o planejamento inicial, a equipe do projeto *LearningLab* realizou reuniões para alinhar as estratégias de divulgação do curso. Durante essas sessões, foram discutidas ideias para criar modelos visuais para as redes sociais do projeto. Além disso, a equipe entrou em contato com as coordenações dos cursos de computação da Universidade Federal do Ceará - Campus Russas para promover a divulgação interna por meio de *e-mails* direcionados aos estudantes. O público-alvo do curso foi definido dentro do contexto acadêmico da universidade, com foco em alunos com conhecimento limitado ou nenhum sobre o tema.

5 - Realização da seleção de alunos inscritos no curso com base nos pré-requisitos definidos para a participação no curso: Para a seleção dos alunos foi considerado os seguintes requisitos: (a) disponibilidade para participar das aulas presenciais, que ocorriam à noite; (b) alunos que tenham pouco ou nenhum conhecimento sobre *Git* e *GitHub*; e (c) prioridade dada aos alunos iniciantes nos cursos de computação do campus.

6 - Realização do curso: O curso foi realizado à noite, em um formato que não conflitasse com as aulas dos alunos, durante um período de quatro dias, com cada aula tendo uma duração de duas horas. Embora o curso pudesse acomodar até 40 alunos, apenas 34 conseguiram concluir todas as etapas.

7 - Coleta e Análise dos *Feedbacks*: A avaliação do *feedback* dos alunos ocorreu por meio de um formulário no *Google Forms*, distribuído aos alunos na última aula por *e-mail* e *WhatsApp*. Os alunos foram informados sobre a confidencialidade dos dados conforme os Termos de Consentimento Livre e Esclarecido (TCLE). Os dados foram analisados qualitativamente com o método GT e quantitativamente com a abordagem IMI, buscando entender a motivação e autorregulação dos alunos durante o curso. Essas análises visaram identificar melhorias para futuros cursos do *LearningLab*.

4. Aplicação do curso

Nesta seção, será apresentada a implementação do curso, fornecendo detalhes sobre o perfil dos instrutores, o cronograma das aulas, os materiais utilizados e a condução do curso.

4.1. Perfil dos Instrutores, Cronograma, e Materiais

O planejamento do curso foi conduzido pelos membros do projeto. A líder do setor de cursos formou um grupo de instrutores no *whatsapp* para discutir os próximos passos, composto por cinco pessoas: (i) Instrutor 1: graduanda em Ciência da Computação, bolsista do Programa Institucional de Bolsas de Iniciação Científica (PIBIC) com interesse em qualidade de *software*; (ii) Instrutor 2: graduanda em Ciência da Computação, voluntária do PIBIC com foco em mobile; (iii) Instrutor 3: graduanda em Engenharia de *Software*, bolsista da Bolsa de Iniciação Acadêmica (BIA) com interesse em desenvolvimento *front-end* e segurança da informação; (iv) Instrutor 4: graduando em Engenharia de *Software*, envolvido em qualidade de *software* e análises qualitativas; (v) Instrutor 5: graduando em Engenharia de *Software*, com experiência em análise de desenvolvimento e como desenvolvedor *back-end* no projeto *LearningLab*. Após a seleção dos instrutores, reuniões de alinhamento foram realizadas para definir um desafio final que incentivasse os

alunos a alcançarem resultados satisfatórios. Uma análise na plataforma *Udemy* resultou na elaboração de um plano de ensino para o curso, baseado nas descobertas e experiências adquiridas durante o estudo. O roteiro inclui:

Aula 1 - Introdução ao *Git* e ao *GitHub* (2 horas, com teoria e prática intercaladas, ministradas pelas instrutoras 1, 2 e 3): O objetivo da aula foi conceituar as funcionalidades do *Git* e do *GitHub*, distingui-las e apresentar os principais comandos para realizar os primeiros *commits*. Também foi discutida em sala de aula a importância das ferramentas no mercado de trabalho, como instalar o *Git* e como fazer o primeiro *login* no *GitHub*. Ao final da aula, realizou-se uma atividade na plataforma educacional de aprendizado *online* usada principalmente para criar, jogar e compartilhar questionários interativos, *Kahoot!*¹.

Aula 2 - Configurações do *Git*, criação de repositórios e fluxo de trabalho básico (2 horas, com teoria e prática intercaladas, ministrada por todos os instrutores): a aula teve como propósito mostrar o processo de realização de *commits*, a criação de repositórios, o envio de código-fonte para o repositório e a compreensão do fluxo de trabalho. A atividade no *Kahoot!*¹ encerrou a aula.

Aula 3 - Remoção de arquivos e *branches* (2 horas, com teoria e prática intercaladas, ministrada por todos os instrutores): O objetivo da aula foi abordar como os arquivos podem ser ignorados e removidos do repositório, trabalhar com *branches* e controle de versionamento. Ao final da aula, realizou-se uma atividade na plataforma *Kahoot!*¹.

Aula 4 - Foram abordados os conceitos de funcionamento do *merge*, *commits* e *branches* (2 horas, com teoria e prática intercaladas, ministrada por todos os instrutores): O objetivo da aula foi explicar como funciona o trabalho com *merge*, histórico de *commits* e estratégias de *branches*. Ao final da aula, realizou-se a gamificação na plataforma *Kahoot!*¹.

Aula 5 - Desafio final (*online*, com prazo de três dias): O objetivo deste desafio foi testar os conhecimentos dos alunos adquiridos no curso. O desafio final consistiu no envio de *commits*, no trabalho com as *branches* e *merge*, além de realizar os comandos ensinados em sala de aula. O envio foi feito para os instrutores conseguissem avaliar.

Os materiais utilizados no curso foram os computadores do laboratório da UFC - Campus Russas para o progresso do curso. As demais instruções e materiais foram disponibilizados mediante *e-mail* e *WhatsApp*, os quais foram os principais meios de comunicação. A seguir, serão detalhados os tipos de materiais desenvolvidos para o curso. (i) *Slides*: durante as aulas, eram apresentados *slides* de maneira didática, contendo os comandos essenciais e suas respectivas funções. Todos os instrutores reforçaram esses comandos utilizando esse material e demonstravam na prática na plataforma *Visual Studio Code* (um editor de código-fonte desenvolvido pela *Microsoft*), como revisão, visando otimizar o aproveitamento de cada aula; (ii) Vídeos: também foram compartilhados vídeos de tutoriais visando auxiliar os alunos nas dúvidas sobre as aulas.

4.2. Condução do curso

As aulas tiveram uma duração de duas horas, realizadas no período noturno, e abrangiam a aplicação dos conteúdos de forma teórica e prática, incluindo uma revisão do conteúdo das aulas anteriores. Durante as aulas, os monitores supervisionavam e auxiliavam os

¹Plataforma Kahoot!: <https://kahoot.it/>

alunos na utilização das ferramentas *GitBash* (emulador de terminal para sistemas operacionais *Windows*) e *GitHub* para o armazenamento e versionamento de projetos. O uso do *WhatsApp* simplificou a comunicação entre instrutores e alunos. Além disso, foram aplicadas dinâmicas lúdicas utilizando a plataforma *Kahoot!*, com perguntas e respostas, para solidificar os conhecimentos adquiridos durante as aulas. Durante o curso, os participantes executaram um trabalho final, que consistiu na criação de um novo repositório. Nele, adicionaram colaboradores, clonaram o repositório em seus computadores, realizaram *commits* e enviaram alterações para o repositório remoto. A avaliação considerou três critérios principais: corretude, completude e qualidade do código, assegurando uma avaliação precisa e consistente do trabalho dos participantes.

5. Resultados e discussões

Com o propósito de obter um *feedback* abrangente, foram utilizados dois meios de análises para os dados coletados no decorrer do curso: análise qualitativa e análise quantitativa.

5.1. Análise Qualitativa dos Dados

As análises qualitativas buscam compreender a complexidade de um fenômeno [Junior et al. 2021]. Neste estudo, a análise foi baseada em informações coletadas por meio de um questionário elaborado pelos membros do projeto *LearningLab* no *Google Forms* e disponibilizado ao final do curso para coletar *feedback* dos alunos. Foi utilizado os procedimentos da *Grounded Theory*, uma abordagem que envolve a codificação dos dados, utilizando as próprias informações como códigos e categorias. Nessa técnica, o código representa a atividade que o pesquisador deseja analisar [Strauss and Corbin 1990]. Após a coleta dos *feedbacks*, as respostas dos alunos foram agrupadas por perguntas, e os dados foram processados utilizando a ferramenta ATLAS.ti. O projeto conta com um setor dedicado à realização de análises qualitativas e quantitativas, buscando identificar tanto os aspectos positivos quanto os negativos do curso. Essa abordagem visa aprimorar o curso em futuras iterações. Todos os participantes consentiram em fazer parte da pesquisa, assinando o Termo de Consentimento Livre e Esclarecido (TCLE). Para garantir confidencialidade, foram atribuídos nomes alfanuméricos aos alunos, identificados de P1 a P34. A seguir serão destacadas as principais informações extraídas dos *feedbacks* dos alunos.

5.1.1. Aspectos que mais gostou - Metodologia de ensino

A Figura 1 apresenta os pontos positivos a partir das respostas do formulário de *feedback*. Os códigos a seguir descrevem os aspectos que mais agradaram aos alunos em relação à metodologia de ensino: “*Ensino de acompanhamento prático*” e “*Metodologia de ensino excelente*”, conforme os alunos P5, P7, P12 e P24, respectivamente:

P5 - “*Os monitores eram bem atenciosos e dedicados para tirar todas as dúvidas não só durante as aulas mas também no grupo do whatsapp*”.

P7 - “*Gostei de que a maior parte do curso foi prática, isso, pelo menos para mim, ajuda muito em fixar o ensinamento*”.

P12 - “*Didática foi muito boa, ajudou muito os alunos que estavam com dúvida e fez exemplos de aplicações*”.

P24 - “*As dinâmicas, principalmente o jogo do Kahoot!*”.

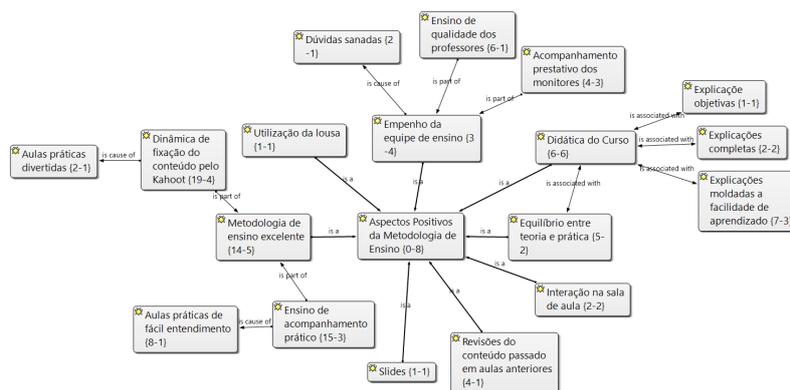


Figura 1. Rede de aspectos positivos da metodologia de ensino

5.1.2. Aspectos que menos gostou - Metodologia de ensino

O propósito da análise dos aspectos menos favoráveis da metodologia de ensino é detectar eventuais lacunas no plano de ensino, observadas pelos alunos. Nesta avaliação, os códigos adiante são aspectos do curso que foram identificados como pontos negativos da metodologia: “*Problemas técnicos do laboratório de informática*” e “*Pouco tempo de curso*”, assim foi relatado pelos alunos P11 e P12, respectivamente:

P11 - “*Sobre muitas vezes a aula ter que parar pra ajudar os que deram problema*”.

P12 - “*Somente mesmo a questão de tempo que não foi muito bem observada*”.

5.1.3. Dificuldades ao longo do curso

Através da análise dos depoimentos coletados no formulário de *feedback*, foi viável investigar os desafios que os estudantes do curso enfrentaram. Essas dificuldades foram identificadas pelos códigos “*Dificuldades com branches*”, “*Dificuldade em lembrar do conteúdo*” e “*Dificuldade com o git fetch*”, conforme relatado pelos alunos P20 e P3:

P20 - “*Criação de branches e seu funcionamento*”.

P3 - “*Tive dificuldade de lembrar dos comandos, da ordem... Porém, ao final do curso, com a prática, consegui entender bem*”.

5.1.4. Facilidades ao longo do curso

Para coletar informações sobre as facilidades que os alunos tiveram durante o curso, a seguinte pergunta foi formulada: “*Quais foram os conceitos que foram possíveis de entender? Fale um pouco sobre:*”. Com base nas respostas, pode-se observar que as facilidades que os estudantes tiveram foram: “*Comandos do Git*” e “*Configuração de uso do Git e Github*”, como pode observar nas respostas dos participantes P1 e P8:

P1 - “*Foram vários, alguns como o git init, que criamos um repositório, Git clone onde clonamos o repositório remoto, Git merge que fazemos uma mesclagem das branches com a main/master, Git commit em que comitamos um arquivo e escrevemos uma descrição do que foi modificado, a fim de quando outro desenvolvedor pegar o arquivo (...)*”.

P8 - “*Consegui entender bem sobre os comandos, como funciona o Github e o Git separadamente e juntamente também*”.

5.1.5. Incentivo a Permanência

Conforme as análises, os códigos que representam o fator determinante para que os alunos permanecessem no curso são: “*Dinâmica de fixação do conteúdo pelo Kahoot!1*” e “*Acompanhamento prático*”, como foram relatadas pelos alunos abaixo:

P1 - “*Sim, o que me motivou foi o modo como fizeram a introdução do curso no primeiro dia, me deixou entusiasmada com o que poderia vir nas próximas aulas e o modo de como todas as aulas foram bem explicativas e didáticas, eu realmente gostei bastante do curso*”.

P2 - “*As aulas em si foram bem divertidas e tiveram muitas práticas. Entretanto, se eu fosse apontar algo que me motivou, seriam os exercícios de fixação*”.

5.2. Análise Quantitativa dos Dados

Para obter uma melhor análise sobre os aspectos do curso, foram elaborados gráficos para examinar aspectos como tensão, escolha, competência e interesse dos alunos no curso. Os próximos tópicos abordarão esses aspectos, assim como descreve o conhecimento dos alunos relacionados aos assuntos apresentados no curso.

5.2.1. Análise - Nível de Conhecimento dos alunos

No que se refere ao nível de conhecimento dos alunos alcançado no curso, a sentença “*Lembrar o que é - ANTES*”, somente 15,7% dos alunos assinaram a opção concordo totalmente, enquanto em “*Lembrar o que é - DEPOIS*” 68,4% marcaram a opção concordo totalmente e 26,3% a opção concordo, diante dessa informação fica evidente o significativo aumento em relação a lembrar os assuntos abordados no curso. Por outro lado, a sentença “*Compreender como funciona - ANTES*”, observou-se que 21% dos discentes indicaram a opção concordo e 10,5% a opção concordo totalmente, já na pergunta “*Compreender como funciona - DEPOIS*”, 63,1% dos estudantes apontaram a opção concordo totalmente e 31,5% a opção concordo, assim pode-se dizer que houve aumento sobre a compreensão dos alunos relacionada aos conteúdos compartilhados. Já em relação à sentença “*Aplicar na prática - ANTES*”, 15,8% dos discentes sinalizaram a opção concordo e 10,7% a opção concordo totalmente, paralelamente para a sentença “*Aplicar na prática - DEPOIS*”, 36,8% dos alunos marcaram a opção concordo e 52,7% a opção concordo totalmente, sendo assim fica evidente que houve um aumento sobre a aplicação dos conteúdos vistos no curso. Desse modo pode-se deduzir que o nível de conhecimento adquirido no curso foi bastante satisfatório por parte de mais 70% dos alunos, assim levando à conclusão de que o curso correspondeu às expectativas estabelecidas pelo projeto *LearningLab*.

5.2.2. IMI (Inventário de Motivação Intrínseca)

Para a análise dos dados numéricos, foi adotado a abordagem IMI, que se concentra na avaliação da experiência subjetiva dos estudantes relacionada à motivação intrínseca e autorregulação [Ryan et al. 1991]. Foi utilizado o IMI para obter informações sobre o interesse dos alunos no trabalho final, se eles sentiram que escolheram realizar a tarefa por vontade própria ou por obrigação, como perceberam sua competência ao concluí-la e se experimentaram alguma tensão durante o processo. O questionário incluiu 22 perguntas, com respostas baseadas na escala *Likert*, variando de “*discordo totalmente*” a “*concordo totalmente*”.

Na Figura 2.a, é exposto o nível de **Interesse**, na qual os alunos avaliam sua

autonomia nas atividades do curso. Os dados refletem um alto grau de engajamento dos alunos em várias dimensões, com todas as afirmações indicando uma resposta positiva, como *"Enquanto realizava as entregas das práticas durante o curso, pensava no quanto gostei de realizá-las"*, *"Realizar as práticas do curso foi muito interessante"* e *"Realizar a entrega das práticas foi divertido"*, com mais de 70% das respostas marcadas como *"Concordo"* e *"Concordo totalmente"*. Por outro lado, na afirmação *"As entregas das práticas do curso foram muito chatas"*, mais de 70% dos alunos selecionaram *"Discordo"* e *"Discordo totalmente"*, destacando o alto nível de interesse dos alunos nas atividades do curso.

Na Figura 2.b apresenta os dados das **Competências** dos alunos no curso, através da análise de seu desempenho e realização das atividades executadas. Com base no gráfico, pode-se concluir que, mais do que a média dos estudantes se sentiram capazes e qualificados, em entregar as atividades propostas no curso, como mostra o gráfico, onde o acúmulo de mais 50% dos alunos, em quase todas as sentenças, exceto nas *"Acho que fui muito bem nas práticas, em comparação aos outros alunos"* e *"Eu acho que fui muito bem em realizar a entrega final do curso"*. A Figura 2.c mostra os dados relacionados à **Tensão**, ou seja, como o discente se sente ao realizar as atividades oferecidas durante o curso. Com base nos gráficos, pode-se notar que na sentença *"Eu me senti tenso enquanto fazia as entregas das práticas do curso"*, 32% dos alunos assinalaram as respostas *"Concordo"* e *"Concordo totalmente"*, por outro lado, 67,4% dos estudantes discordaram da afirmação *"Eu me senti pressionado enquanto fazia as entregas das práticas do curso"*. Já nas sentenças *"Eu me senti relaxado enquanto fazia as entregas das práticas do curso"* e *"Não me senti nada nervoso ao fazer as práticas que o curso proporciona"*, cerca de mais de 60% dos alunos assinalaram as opções *"Concordo"* e *"Concordo totalmente"*. Pode-se concluir que os alunos não se sentiram pressionados em realizar as atividades do curso. Os dados da figura 2.d tem o objetivo de analisar a perspectiva do aluno em

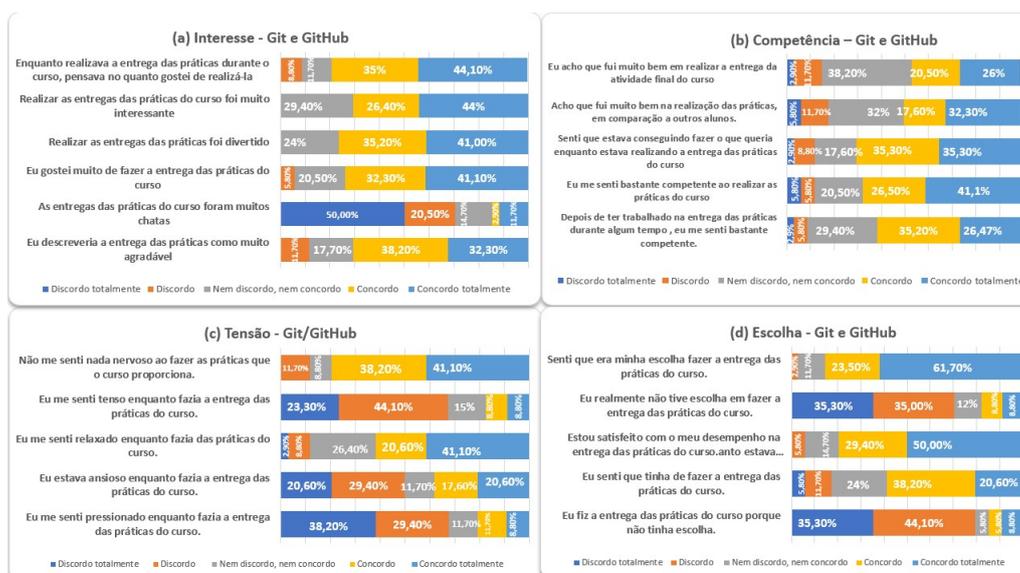


Figura 2. Gráficos de interesse, competência, escolha e tensão

relação à **Escolha Percebida**, ou seja, identificar se os alunos foram obrigados ou fizeram as atividades do curso por conta própria. Em resumo, de acordo com acúmulo de respos-

tas “*Discordo*” e “*Discordo totalmente*” na sentença “*Eu fiz as práticas do curso porque não tinha escolha*” e “*Eu realmente não tive escolha em fazer as práticas do curso*”, em conjunto com a aglomeração das respostas em “*Concordo*” e “*Concordo totalmente*” na sentença “*Senti que era minha escolha fazer as entregas das práticas do curso*”, fica evidente que mais de 60% dos alunos teve opção de escolha durante a execução das práticas do curso.

6. Desafios Encontrados

No decorrer das atividades do curso, desde o planejamento até a realização, foram identificados alguns desafios, que incluem: (a) Preparação dos instrutores em relação ao conteúdo do curso: os instrutores possuíam diferentes níveis de conhecimento em relação ao tema do curso. Portanto, tornou-se necessário que todos os instrutores estudassem o mesmo conteúdo especificamente, a fim de garantir uma uniformidade no conhecimento a ser ministrado; (b) Familiaridade dos alunos com o tema do curso: visando atender ao público com conhecimento básico e intermediário, isso implicou no nivelamento das aulas, uma vez que havia alunos que não sabiam nada sobre o tema. Isso representou um desafio, pois os instrutores precisavam adaptar o conteúdo para atender tanto aqueles que não tinham conhecimento sobre o assunto, quanto os alunos que já entendiam um pouco do tema; (c) Realização do projeto final: um número reduzido de estudantes concluiu a etapa final do projeto, na qual tinha como objetivo permiti-los praticar de uma só vez, tudo o que já haviam aprendido nas aulas do curso. Observa-se que os alunos enfrentaram consideráveis dificuldades, possivelmente por muitos deles terem pouca familiaridade com o *Git* e o *GitHub*. Como resultado, do total de 34 alunos que receberam certificado de participação, 25 estudantes não conseguiram submeter o projeto final, mesmo após terem tido um prazo de uma semana para fazê-lo.

7. Conclusão e trabalhos futuros

Os alunos precisam se dedicar ao aprendizado, e os tutores devem ser pacientes, aplicando uma metodologia eficaz. Neste contexto, este artigo descreve como o projeto *LearningLab* adota uma abordagem participativa para promover interação e compartilhamento de conhecimento entre alunos e tutores. Com base na análise qualitativa realizada, pode-se concluir que o aspecto do curso que mais agradou aos alunos foi “*Dinâmica de fixação do conteúdo pelo Kahoot!*”, juntamente com “*Ensino de acompanhamento prático*”. Enquanto o que menos agradou foi “*Pouco tempo de curso*” e “*Explicações rápidas*”. Além disso, a análise quantitativa revelou que 85% dos alunos se comprometeram a realizar a entrega do trabalho final, porém muitos não conseguiram. Por outro lado, é possível afirmar que o curso atingiu seu objetivo, de ofertar um curso de maneira interativa, mantendo os alunos engajados nas aulas, consoante as análises feitas.

Para trabalhos futuros, planeja-se repetir o curso de *Git* e *GitHub* com estudantes de Engenharia de *Software* e Ciência da Computação da UFC - Campus Russas. A ênfase estará nas melhorias identificadas nas análises anteriores para avaliar a relevância do curso na graduação. Também, considera-se a possibilidade de disponibilizar o curso *online*, ampliando seu alcance além da comunidade da instituição.

Referências

Blischak, J. D., Davenport, E. R., and Wilson, G. (2016). A quick introduction to version control with git and github. *PLoS computational biology*, 12(1):e1004668.

- Brought, G., Jackson, S., and Wurst, K. R. (2023). Gitkit: Teaching git and github/gitlab workflow in an authentic context. *Journal of Computing Sciences in Colleges*, 38(8):22–23.
- Choi, J., Mogami, T., and Medalia, A. (2010). Intrinsic motivation inventory: an adapted measure for schizophrenia research. *Schizophrenia bulletin*, 36(5):966–976.
- da Silva, T. R., Barros, I. S., Sousa, L. K. D. S., Sá, A. L. D., Silva, A. F. M., Araujo, M. C. S., and da Silva Aranha, E. H. (2021). Um mapeamento sistemático sobre o ensino e aprendizagem de programação. *Revista Novas Tecnologias na Educação*, 19(1):156–165.
- de Sousa, R. R., da Silva Coutinho, J. C., Nascimento, S. M., and Leite, F. T. (2021). Investigando as dificuldades e perspectivas sobre um curso de engenharia de software de dois ciclos: Um survey com a visão discente. In *Anais do Simpósio Brasileiro de Educação em Computação*, pages 55–65. SBC.
- de Sousa Pereira, L. and Saqui, D. (2023). Análise sobre a importância da realização de iniciação científica entre alunos e egressos em ti. *15º JORNADA CIENTÍFICA E TECNOLÓGICA E 12º SIMPÓSIO DE PÓS-GRADUAÇÃO DO IFSULDEMINAS*, 15(3).
- Gonçalves, A. and Scaico, P. (2019). O desafio de professores da licenciatura em computação ao usarem práticas de aprendizagem ativa: Um estudo de caso na universidade federal da paraíba. In *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*, volume 8, page 91.
- Jarczyk, O., Jaroszewicz, S., Wierzbicki, A., Pawlak, K., and Jankowski-Lorek, M. (2018). Surgical teams on github: Modeling performance of github project development processes. *Information and Software Technology*, 100:32–46.
- Junior, E. B. L., de Oliveira, G. S., dos Santos, A. C. O., and Schnekenberg, G. F. (2021). Análise documental como percurso metodológico na pesquisa qualitativa. *Cadernos da FUCAMP*, 20(44).
- Oliveira, C., Tostes, R., and Sirqueira, T. F. M. (2021). Git viewer: uma plataforma para análise de dados do github. *ANALECTA-Centro Universitário Academia*, 6(3).
- Ryan, R. M., Koestner, R., and Deci, E. L. (1991). Ego-involved persistence: When free-choice behavior is not intrinsically motivated. *Motivation and emotion*, 15(3):185–205.
- Santiago, C. P., Menezes, J. W. M., and de Aquino, F. J. A. (2023). Proposta e avaliação de uma metodologia de aprendizagem baseada em projetos em disciplinas de engenharia de software através de uma sequência didática. *Revista Brasileira de Informática na Educação*, 31(1):31–59.
- Silva, J., Oliveira, F., and Martins, D. (2019). Kahoot! como instrumento potencializador na participação e engajamento dos alunos na aprendizagem de conceitos de programação. In *Anais do XXVII Workshop sobre Educação em Computação*, pages 41–50. SBC.
- Strauss, A. and Corbin, J. (1990). *Basics of qualitative research*. Sage publications.
- Widthauer, M. R. (2020). O uso de jogos educacionais para a aprendizagem de programação no ensino fundamental.

Zhu, J. and Wei, J. (2019). An empirical study of multiple names and email addresses in oss version control repositories. In *2019 IEEE/ACM 16th International Conference on Mining Software Repositories (MSR)*, pages 409–420. IEEE.