

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO GRANDE DO SUL

ESCOLA POLITÉCNICA

CURSO DE ENGENHARIA DE SOFTWARE

CAETANO KICH TAFFE

DANIELLA MORAES SANTOS

GABRIEL ARRUDA PAIM

GABRIEL LEÃO DOMINGUES

GUSTAVO DE FREITAS FIDÉLIS

LUCA WOLFFENBÜTTEL BOHNENBERGER

LUCAS DA PAZ OLIVEIRA

MATEUS GRANJA DE ANDRADE MADALOSSO TIEPPO

RODRIGO MIOTTO SLONGO

WILLIAM DE OLIVEIRA KLEIN

PRÁTICA COM GIT EM TIMES

PORTO ALEGRE

2024

CAETANO KICH TAFTE (*CaetanoKich*)

DANIELLA MORAES SANTOS (*MSDaniella*)

GABRIEL ARRUDA PAIM (*GabrielArrudaPaim*)

GABRIEL LEÃO DOMINGUES (*DominguesGabriel*)

GUSTAVO DE FREITAS FIDÉLIS (*gustavofid*)

LUCA WOLFFENBÜTTEL BOHNENBERGER (*LucaWBohnenberger*)

LUCAS DA PAZ OLIVEIRA (*xLucaspx*)

MATEUS GRANJA DE ANDRADE MADALOSSO TIEPPO (*Mateus-Tieppo*)

RODRIGO MIOTTO SLONGO (*Slongo11*)

WILLIAM DE OLIVEIRA KLEIN (*William-Olv-Kln*)

PRÁTICA COM GIT EM TIMES

Trabalho apresentado no curso de Engenharia de Software da Pontifícia Universidade Católica do Rio Grande do Sul, referente à cadeira de Gerenciamento de Configuração de Software.

Professor: Daniel Antônio Callegari

PORTO ALEGRE

2024

SUMÁRIO

I.	INTRODUÇÃO.....	4
II.	DESENVOLVIMENTO	5
2.1	Processo de <i>branching</i> e fluxo de trabalho.....	5
2.2	Contribuições.....	9
III.	CONCLUSÃO.....	11
3.1	Autoavaliações individuais.....	11
3.1.1	Caetano (CaetanoKich).....	11
3.1.2	Daniella (MSDaniella).....	11
3.1.3	Gabriel Paim (GabrielArrudaPaim).....	11
3.1.4	Gabriel Domingues (DominguesGabriel)	12
3.1.5	Gustavo (Gustavofid).....	12
3.1.6	Luca (LucaWBohnenberger)	12
3.1.7	Lucas (xLucaspx).....	13
3.1.8	Mateus (Mateus-Tieppo).....	14
3.1.9	Rodrigo (Slongo11).....	14
3.1.10	William (William-Olv-Kln).....	14
3.2	Autoavaliação do grupo.....	15

I. INTRODUÇÃO

Este trabalho tem como objetivo a prática da ferramenta *Git* em times de alunos do curso de Engenharia de *Software*. Foi solicitada a criação de um repositório no *GitHub* para o desenvolvimento de um sistema de gerenciamento de itens entre jogadores; o sistema foi implementado em *Java*, utilizando o *console* para interação com o usuário e sem persistência de dados. Para conveniência, o sistema inicia com dados já preenchidos; estes dados são mantidos em um arquivo chamado *seeder.txt* no diretório *resources* e são lidos, interpretados e armazenados na memória no início de cada execução. Mais informações sobre o preenchimento dos dados podem ser encontradas no arquivo [readme.md](#) no repositório do projeto.

O repositório remoto deste trabalho pode ser acessado no endereço <https://github.com/xLucaspx/gcs-t1>.

II. DESENVOLVIMENTO

De modo geral, o sistema possui opções de cadastro e *login*; conforme solicitado, é possível se autenticar e alternar entre usuários sem a necessidade de utilizar credenciais de acesso. Na tela inicial, são apresentadas informações do usuário autenticado (nome e *e-mail*) e notificações de troca, *i.e.*, quantas propostas recebidas e quantas propostas realizadas estão em aberto no momento.

A aplicação permite que o usuário liste seus itens (em ordem alfabética), cadastre um novo item ou exclua um já existente; também é possível listar os itens disponíveis, *i.e.*, itens que pertencem a outros jogadores, por ordem de preço (do menor valor para o maior). O usuário pode, ainda, filtrar itens buscando por *ID*, nome, descrição, categoria ou preço; destes critérios, a busca por *ID* retornará um ou nenhum item e a busca por preço receberá o valor mínimo e o valor máximo, retornando todos os itens com preço neste intervalo (incluso). As demais buscas retornam todos os itens cujo atributo correspondente contém o valor buscado.

Para abrir uma proposta de troca, informa-se o *ID* do item oferecido e o *ID* do item solicitado; após, o sistema mostra o resultado para o usuário e solicita a confirmação. Pode-se listar as propostas recebidas, propostas recebidas em aberto, propostas realizadas e propostas realizadas em aberto. O usuário pode, também, aceitar ou recusar propostas recebidas em aberto e cancelar propostas realizadas em aberto.

Jogadores autenticados conseguem visualizar informações do sistema (*e.g.*: quantidade de usuários cadastrados, quantidade de itens cadastrados, valor total e preço médio de itens, quantidade de propostas no sistema *etc.*). Por fim, é possível realizar *logout* e se autenticar novamente ou encerrar o sistema; nesta última opção, os dados armazenados e modificados durante a execução são perdidos.

É realizado um tratamento básico de exceções para impedir que a aplicação quebre com alguma entrada inválida. Para inserção de valores numéricos não-inteiros é utilizado o ponto decimal, não a vírgula.

2.1 Processo de *branching* e fluxo de trabalho

Anteriormente ao desenvolvimento do sistema, foram verificados o detalhamento e as funcionalidades desejadas do sistema – conforme constam na especificação do trabalho – e criou-se um diagrama de classes utilizando a ferramenta *Astah*. Neste diagrama foram detalhadas as classes iniciais do projeto, seus métodos, atributos e relacionamentos. Durante os

avanços do desenvolvimento avançou, o diagrama foi atualizado conforme a versão mais atual do sistema.

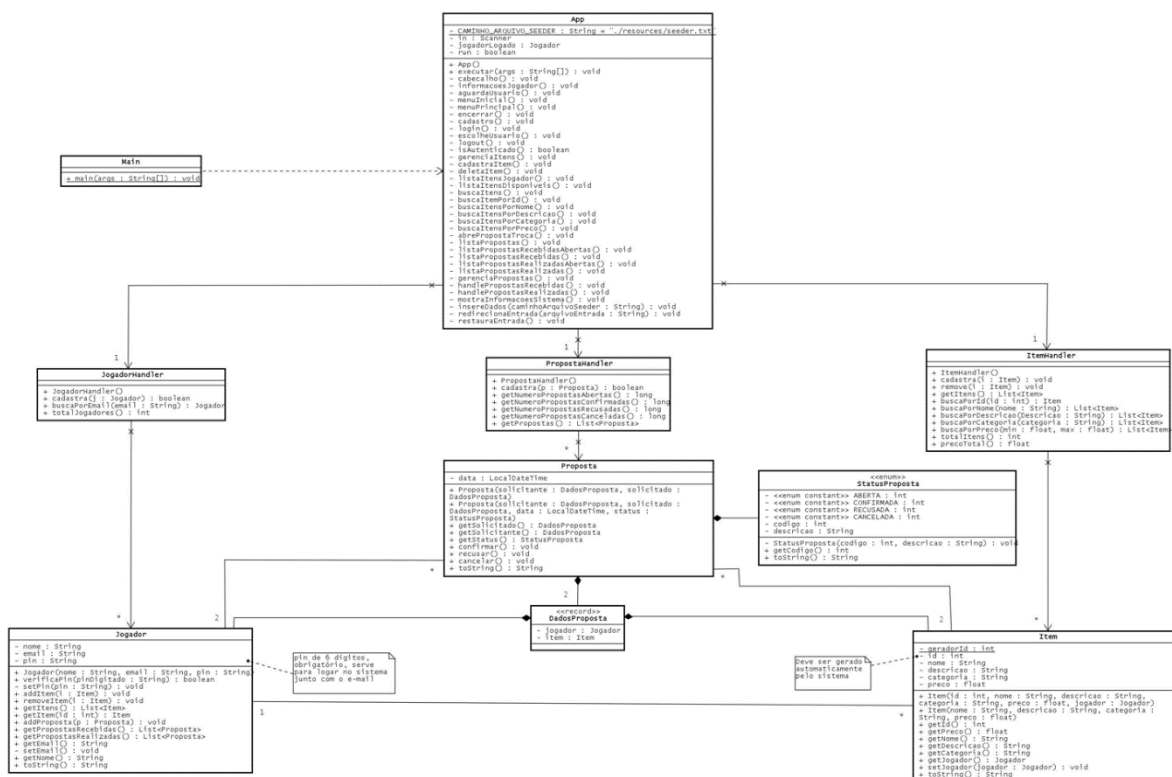


Figura 1: Diagrama de classes do projeto na versão 1.1

Com o diagrama ilustrando as “peças” do sistema, foram criadas diferentes tarefas para dividir o processo de desenvolvimento. Para manutenção das tarefas e organização do fluxo de trabalho entre os membros do grupo utilizou-se o *Trello* como ferramenta; nele, cada tarefa tornou-se um *card* e foram criadas diferentes colunas, como “A fazer”, “Em andamento” *etc.* Cada membro do grupo criou uma etiqueta própria com seu nome para a distribuição das tarefas, que eram movidas entre as colunas conforme o desenvolvimento, terminando – finalmente – na coluna de tarefas concluídas. Para algumas tarefas também foram criadas, além dos *cards* no *Trello*, *issues* no repositório do projeto.

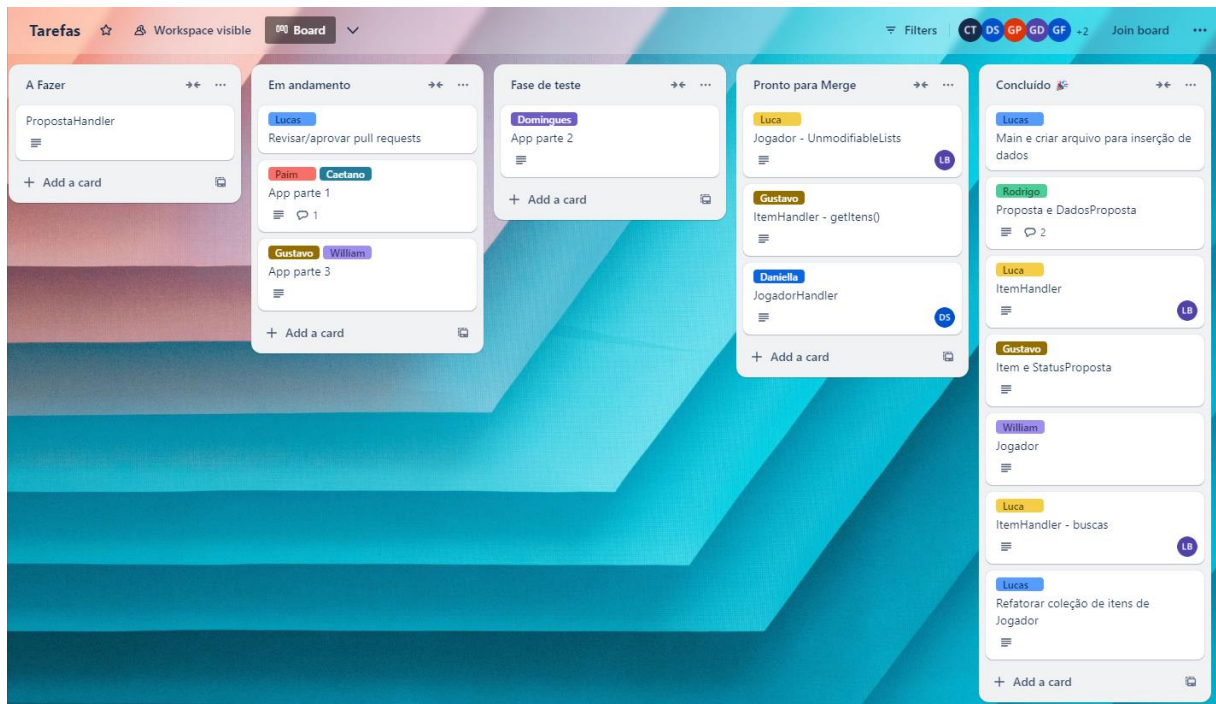


Figura 2: Captura de tela do quadro Trello do projeto durante o desenvolvimento

Com o repositório e a estrutura básica do projeto criadas, desenvolveu-se o processo de *branching* a ser utilizado: foi criada uma ramificação (*branch*) chamada *dev* contendo a estrutura e documentação iniciais do projeto. Cada integrante deveria clonar o repositório em sua máquina, mover-se para a *branch dev* e, a partir daí, criar uma ramificação para trabalhar em uma atividade específica. Ao concluir sua atividade, necessitaria subir sua *branch* para o repositório remoto e abrir um *pull request* para a *branch dev*, descrevendo todas as adições e alterações realizadas.

Durante a revisão do *pull request* por algum integrante, poderiam ser solicitadas mudanças a ser realizadas pelo autor do *pull request*. Após aprovação, o conteúdo do *pull request* seria incorporado à *branch dev*. A partir daí, para desenvolver uma nova funcionalidade repete-se o processo, garantindo antes que a *branch dev* esteja atualizada com o comando *pull*.

Neste projeto, ficou definido que só seria realizada liberação de versão após concluir as tarefas dispostas no quadro do Trello; nesta situação, com a *branch dev* contendo todas as funcionalidades desejadas, seria criada uma nova *branch* de *release* para ajustes finais, uma *tag* de versão um *pull request* para a *branch* principal do projeto (*main*), os dois últimos a partir da *branch* de *release*.

No eventual caso de o projeto necessitar de uma correção urgente e/ou de grande importância, podem ser criadas *branches* de *hotfix* para corrigir o problema. *Pull requests* abertos a partir destas ramificações devem ser incorporados diretamente à *branch* principal.

Convencionou-se da seguinte maneira a nomenclatura das *branches* do projeto:

- **Feature:** para o desenvolvimento de novas *features*/funcionalidades será utilizado o prefixo “feature/”;
- **Documentação:** para atividades relacionadas à documentação será utilizado o prefixo “docs/”;
- **Bugfix:** para a correção de *bugs* e defeitos no código será utilizado o prefixo “bugfix/”;
- **Hotfix:** para correções críticas, diretamente na *branch* principal, será utilizado o prefixo “hotfix/”;
- **Release:** para cada versão da aplicação, antes de fazer a integração com a *branch* principal será utilizada uma ramificação com o prefixo “release/”.

Abaixo, capturas de tela da aba *Insights/Network* do repositório demonstrando o fluxo das ramificações criadas e seus respectivos *merges*:

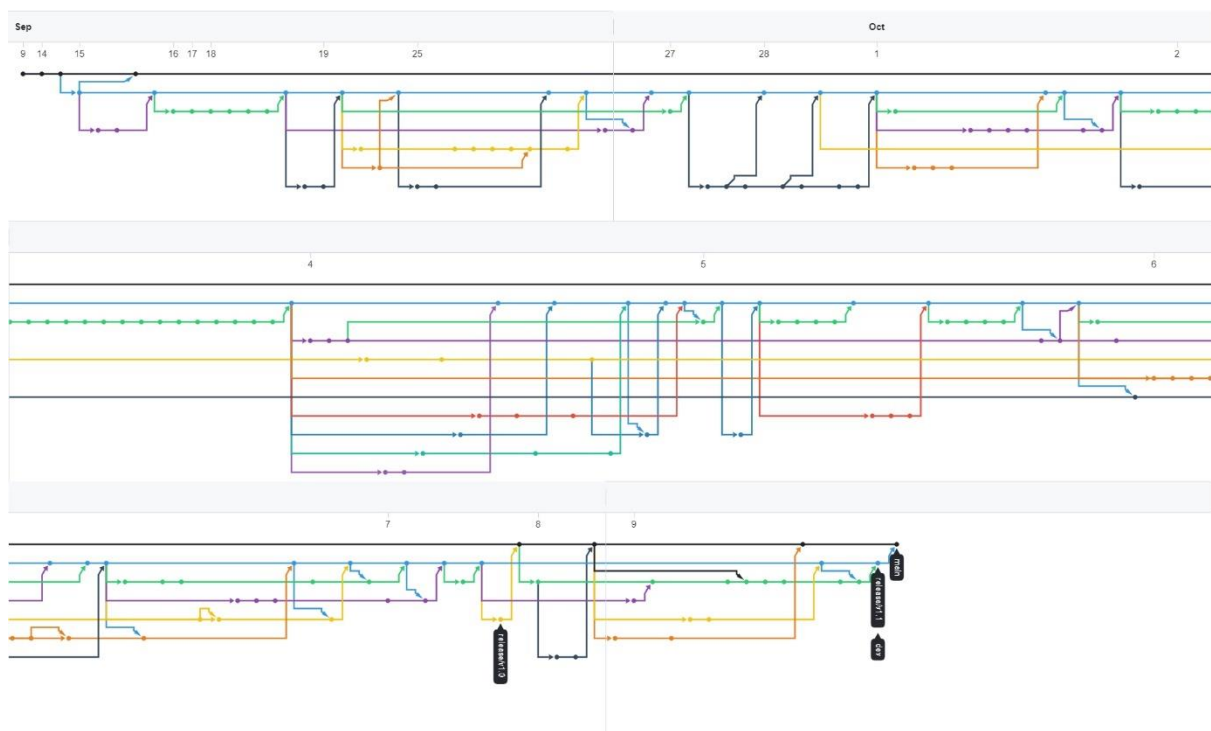


Figura 3: Capturas de tela do gráfico gerado pelo GitHub demonstrando o fluxo de trabalho no repositório

2.2 Contribuições

Nesta seção constam gráficos de *commits* ao longo do projeto – geral e individuais – e a tabela de contribuições contendo o nome de cada membro do grupo, seu usuário do *GitHub* e a lista de *pull requests* abertos por esse integrante.

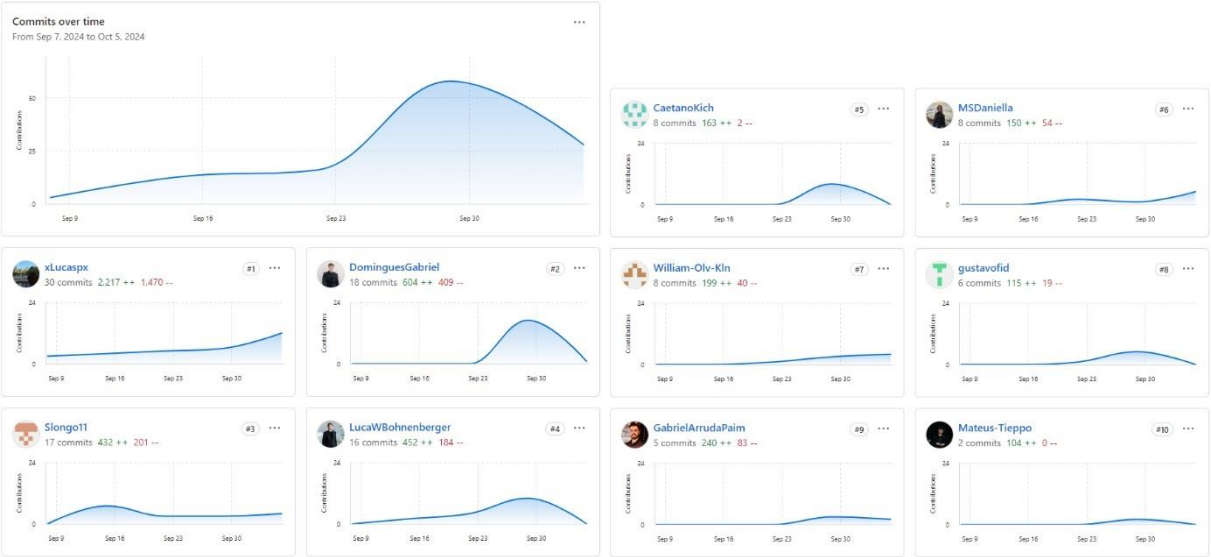


Figura 4: Gráficos de contribuições ao longo do tempo, gerados automaticamente pelo *GitHub*

Contribuições	
Nome (Usuário <i>GitHub</i>)	<i>Pull Requests</i>
Caetano (CaetanoKich)	<ul style="list-style-type: none">➤ #27 Métodos cadastro, login, logout e isAutenticado na App;➤ #30 Adição de mais métodos na classe App.
Daniella. (MSDaniella)	<ul style="list-style-type: none">➤ #6 Feature/jogador handler;➤ #24 add docs;➤ #34 Feature/busca itens.
Gabriel Paim (GabrielArrudaPaim)	<ul style="list-style-type: none">➤ #23 Feature/app1 pt1;➤ #36 reestruturação no método executar e no menu;➤ #39 Mudança do método listaItensJogador, e implementação dos métodos cadastraItens e listaItensDisponiveis;
Gabriel Domingues. (DominguesGabriel)	<ul style="list-style-type: none">➤ #19 Feature/app;➤ #43 Adicionando documentação no método precoTotal da classe ItemHandler.
Gustavo (Gustavofid)	<ul style="list-style-type: none">➤ #8 Feature/item;➤ #18 Adição do método .get na classe ItemHandler;

Contribuições	
Nome (Usuário <i>GitHub</i>)	<i>Pull Requests</i>
	<ul style="list-style-type: none"> ➤ #20 Feature/app3 pt2; ➤ #31 Feature/app3 pt2.
Luca (LucaWBohnenberger)	<ul style="list-style-type: none"> ➤ #5 Feature/itemhandler; ➤ #7 Feature/item handler; ➤ #16 Bugfix/jogador; ➤ #17 Bugfix/itemhandler; ➤ #21 add metodos de busca;
Lucas (xLucaspx)	<ul style="list-style-type: none"> ➤ #1 Add seeder file and base methods for reading data; ➤ #2 Creating basic project structure; ➤ #12 Correções gerais no estado atual do projeto; ➤ #26 Transformando coleção de itens do jogador em mapa; ➤ #28 Melhorias em coleções e buscas; ➤ #29 Refatorações: toString e streams; ➤ #40 Versão 1.0 do sistema; ➤ #42 Hotfix: métodos escolheUsuario e deletaItem; ➤ #44 Hotfix: Tratamento de exceções; ➤ #45 Funcionalidades adicionais; ➤ #46 Atualizando documentação e diagrama de classes; ➤ #47 Versão 1.1 do sistema.
Mateus (Mateus-Tieppo)	<ul style="list-style-type: none"> ➤ #32 Feature/proposta handler; ➤ #33 implementação da listaPropostas.
Rodrigo. (Slongo11)	<ul style="list-style-type: none"> ➤ #3 Adicionando classe proposta e propostas; ➤ #4 Validando a entrada para atualizar Status da proposta; ➤ #10 Feito correção da classe e atualizado diagrama; ➤ #12 Docs/proposta; ➤ #22 Feito o getStatus da proposta; ➤ #35 bugFix/duplicacao app; ➤ #48 Atualizando documentação.
William (william-0lv-k1n)	<ul style="list-style-type: none"> ➤ #9 add class Jogador; ➤ #25 Feature/app3 pt1; ➤ #37 Bugfix/propostas app.

III. CONCLUSÃO

3.1 Autoavaliações individuais

3.1.1 Caetano (CaetanoKich)

Durante o desenvolvimento do projeto da disciplina, consegui colocar em prática os conhecimentos da ferramenta *Git* aprendidos durante as aulas e pude experimentar como é trabalhar em um grupo grande de desenvolvimento de software. No início do projeto tive algumas dificuldades com os comandos do *Git* e a ordem com que deviam ser executados. Porém, através dos materiais de estudo e com o auxílio dos meus colegas de grupo, consegui sanar as minhas dúvidas e, com isso, contribuir para o projeto. Ao final do trabalho, vejo uma grande evolução minha referente ao uso da ferramenta *Git* e sobre os conhecimentos acerca de versionamento de código, evolução essa gerada, principalmente pela utilização destes recursos de forma prática durante o desenvolvimento do trabalho. Com isso, atribuo a mim mesmo uma nota 9,5, pois tive uma boa contribuição para o trabalho, além de conseguir superar minhas dificuldades e aprimorar meu conhecimento.

3.1.2 Daniella (MSDaniella)

Durante o desenvolvimento do projeto, consegui aprimorar meus conhecimentos em *Git*, aplicando na prática o que foi aprendido nas aulas de Gerenciamento de Configuração de Software. No começo, tive bastante dificuldade em entender o que precisava ser feito e como realizar as tarefas. Mesmo compreendendo os conceitos apresentados em aula, na prática me sentia bastante perdida. Felizmente, pude contar com a ajuda dos meus colegas, que sempre me auxiliaram quando surgiam dúvidas sobre o que deveria ser feito. Acredito que minha participação tenha uma nota 9, pois, apesar de ter contribuído da melhor forma que consegui, sempre há espaço para melhorias.

3.1.3 Gabriel Paim (GabrielArrudaPaim)

Durante o desenvolvimento do projeto botei em prática diversos conhecimentos adquiridos durante as aulas de gerenciamento de configuração de software. Pude me aperfeiçoar em diversos aspectos durante todo o processo de desenvolvimento do projeto, poder experimentar como é trabalhar no desenvolvimento de algum projeto em conjunto com uma equipe me fez aprender bastante e apreciar ainda mais a área em que quero trabalhar. Mesmo com o grupo bem organizado tive um pouco de dificuldade de entender o funcionamento na prática do *Git*, mas após estudar um pouco consegui entender como utilizá-lo de forma eficiente,

e daí em diante não tive mais dificuldades. Acredito que mereço nota 9,5 pois me dediquei ao projeto e tive uma boa contribuição ao projeto, mas sempre há espaço para melhorar.

3.1.4 Gabriel Domingues (*DominguesGabriel*)

Durante o desenvolvimento do trabalho na disciplina de gerenciamento de configuração de software, percebi uma grande evolução na minha aprendizagem. Ao realizar o projeto, consegui aplicar os conhecimentos teóricos adquiridos de forma prática. No início, tive algumas dificuldades com o controle de versões utilizando *Git* e *GitHub*, mas nada que me impedisse de contribuir com o grupo. Com o tempo, meu entendimento sobre essas ferramentas se ampliou significativamente, especialmente após concluir minhas tarefas. A experiência de trabalhar com uma equipe relativamente grande, sem conhecer alguns dos integrantes, tornou o desafio mais difícil, mas também mais enriquecedor, pois pude sentir na prática como é colaborar em um projeto onde todos buscam o mesmo objetivo, sem qualquer tipo de relação além da profissional.

Apreendi bastante com meus colegas ao longo do processo, o que foi essencial para meu crescimento. Além disso, melhorei minhas habilidades de trabalho em equipe, o que me deixou mais preparado para futuras colaborações. Hoje, sinto-me mais confiante para utilizar o *Git* como ferramenta de controle de versões e contribuir de maneira eficaz em projetos coletivos.

Por conta dessa grande evolução no decorrer do trabalho, atribuo a mim mesmo, na autoavaliação nota 10.

3.1.5 Gustavo (*Gustavofid*)

Através do desenvolvimento deste projeto, aprendi bastante sobre o uso da ferramenta *Git* e, principalmente, a trabalhar em equipe, consultando meus colegas sempre que surgia algum problema na execução das minhas tarefas. No início, tive bastante dificuldade para entender o que precisava ser feito ou corrigido no trabalho, mas, com a prática constante e as conversas com os outros integrantes, consegui sanar minhas dúvidas e começar a contribuir para o projeto. Porém, apesar de ter ajudado meus colegas nas atividades, acredito que poderia ter contribuído um pouco mais. Assim, atribuo a mim uma nota 9,5 pela minha participação.

3.1.6 Luca (*LucaWBohnenberger*)

Conheço *Git* há cerca de 6 meses e já o utilizo com frequência no meu dia a dia para projetos pessoais, juntamente com o *GitHub*. Neste projeto, aprendi a usar alguns comandos novos, como *git switch -C*. Achei o projeto interessante e gostei de trabalhar em equipe. Além

de programar, ajudei os outros membros do grupo com suas partes do trabalho, também criei um *Trello* para auxiliar a equipe na divisão de tarefas. Em resumo, acredito que, embora meu desempenho não tenha sido perfeito, fiz o meu melhor e, por isso, me atribuo a nota 10.

3.1.7 Lucas (*xLucaspx*)

Conheço a ferramenta *Git* há, pelo menos, 2 anos; na maioria das vezes utilizo-a, juntamente com a plataforma *GitHub*, para versionamento e organização de projetos pessoais e/ou de estudos. No meu parecer, o maior desafio desta atividade não é o uso da ferramenta, nem sequer o código, mas o trabalho em equipe. Sendo o membro mais experiente do grupo, esforcei-me para auxiliar ao máximo os companheiros de equipe com alguma dificuldade; no início do projeto, em consenso com os outros integrantes, desenvolvi o diagrama de classes e a estrutura inicial do projeto. Auxiliei, também, na divisão de tarefas e escrevi a primeira documentação do projeto explicando como realizar o *clone* do repositório, navegar entre ramificações e criar *branches* seguindo a convenção definida.

No que se refere à implementação, houve algumas discrepâncias que não pude deixar de notar, mas que não considero graves, principalmente para um grupo grande de alunos com pouca ou nenhuma experiência em versionamento. Para citar exemplos, refiro-me às mensagens de *commits* e descrições de *pull requests*; algumas ficaram em inglês, outras em português, algumas muito curtas e não descritivas, outras longas demais *etc.* Por mais que sejam detalhes, e não erros crassos, sinto que poderiam ter sido corrigidos e evitados.

Em suma, fiz do meu objetivo neste trabalho nortear o projeto, tentando gerir a equipe e solucionar os problemas conforme ocorriam. Visivelmente fiquei com mais linhas de código, porém julgo que isso se deve, sobretudo, a dois fatores: resolução de conflitos e correção de formatação; a escrita da documentação também aumentou significativamente o número de linhas modificadas. Creio que minha maior contribuição, porém, foi o auxílio prestado aos colegas; da metade para o final do projeto, repassei a revisão e aprovação de *pull requests* para os membros mais ativos do grupo, auxiliando apenas na resolução de conflitos e fazendo os ajustes finais no código. Fiz o possível para que todos participassem e estive disponível o tempo todo para prestar auxílio; seguindo a máxima de que sempre é possível melhorar, atribuo-me nota 9 na conclusão deste trabalho e espero ter contribuído para o desenvolvimento dos colegas.

3.1.8 Mateus (*Mateus-Tieppo*)

Durante o desenvolvimento deste projeto, consegui colocar em prática os conhecimentos que adquiri nas aulas de Gerenciamento de Configuração de Software, especialmente no uso do *Git*. No início, senti uma certa dificuldade para entender como aplicar os conceitos que aprendi, mas, com a ajuda dos meus colegas e a prática, fui melhorando. Um dos maiores desafios para mim foi entender o fluxo de trabalho no *Git* e como utilizá-lo em equipe, mas com o tempo me senti mais confortável. Contribuí nas tarefas do grupo, e mesmo quando enfrentava dificuldades, sempre busquei aprender e melhorar. Tive o apoio dos meus colegas, que foram essenciais para o meu desenvolvimento, e com isso fui capaz de contribuir de forma significativa no projeto. Hoje, me sinto mais confiante para trabalhar com controle de versão e colaborar em projetos coletivos. Acredito que minha participação foi positiva, e por isso me atribuo uma nota 9. Sei que ainda posso melhorar em alguns aspectos, como otimizar minha organização de tempo e aprender mais sobre boas práticas de *commits*, mas estou satisfeito com o progresso que fiz até aqui.

3.1.9 Rodrigo (*Slongo11*)

Durante o desenvolvimento deste trabalho, enfrentei várias dificuldades e cometi alguns erros, mas, ao longo do processo, consegui compreender as razões pelas quais as coisas não saíram como o esperado e, mais importante, aprendi muitas coisas novas. Um exemplo disso foi o uso da ferramenta *Git*, onde aprendi a criar repositórios remotos e, acima de tudo, o valor do trabalho em equipe no papel de "desenvolvedor". Posso afirmar que contribuí em cada etapa do projeto, auxiliando os colegas que enfrentavam as mesmas dificuldades pelas quais eu já havia passado. Fiquei maravilhado ao conhecer essa ferramenta, que certamente utilizarei em minha carreira profissional, especialmente para o versionamento de código. Além disso, fui um dos mais ativos na fase inicial do desenvolvimento, tomando as primeiras iniciativas. Por esse motivo, acredito que mereço uma nota 10 na minha avaliação individual pela contribuição ao projeto em equipe.

3.1.10 William (*William-Olv-Kln*)

No início do projeto, tive dificuldades em usar o *Git* para gerenciar e versionar o código, mas, com o apoio dos meus colegas de equipe e do professor, que sempre estiveram disponíveis para me ajudar, consegui aprender os comandos principais e entender a importância do *Git* para desenvolvedores. Ao longo do trabalho, não só aprendi a usar essa ferramenta, como também desenvolvi habilidades de trabalho em equipe, o que acelerou e facilitou o desenvolvimento do

projeto ao agregar essas competências. A experiência me ensinou a colaborar e a gerenciar tarefas em grupo de maneira mais eficiente. Contribuí ativamente com todas as atividades que me foram designadas e estive sempre disposto a ajudar meus colegas. Antes, eu não tinha conhecimento de *Git*, mas agora, aplico-o em todos os projetos em equipe. Por esse progresso, considero meu desempenho excelente e me atribuo a nota 10.

3.2 Autoavaliação do grupo

As contribuições a nível de código de cada membro do grupo estão dispostas acima, na seção 2.2, porém é válido ressaltar que também houve contribuições significativas fora deste escopo. Diagrama de classes, quadro *Trello* e documentação (*readme*) do projeto foram ferramentas utilizadas, principalmente, visando nortear os membros menos experientes do grupo durante o desenvolvimento do projeto.

Colegas com algum grau de dificuldade foram auxiliados por outros integrantes em diversos momentos, fosse com relação ao código ou à ferramenta *Git*. Neste aspecto, as maiores dificuldades foram encontradas pelos colegas menos experientes no uso do *Git*; com a explicação inserida na documentação do projeto, a maior parte das dúvidas com relação a isso foi sanada, e as que persistiram foram respondidas por algum integrante mais experiente ou que já havia passado por situação semelhante. Algumas dúvidas sobre código foram respondidas em comunicação direta entre os membros do grupo, enquanto outras questões foram resolvidas diretamente nos *pull requests*, com revisões e solicitações de alteração. Tendo em mente que o objetivo do trabalho é a prática de versionamento, e não o desenvolvimento em si, é visível que todos – inclusive os que já conheciam e/ou utilizavam a ferramenta – adquiriram um grau maior de conhecimento e experiência.

A utilização básica do *Git* não é algo considerado difícil de se aprender, porém exige prática. A utilização organizada e eficiente da ferramenta exige mais conhecimento e ainda mais prática. Da forma como o fluxo de trabalho foi estruturado, com utilização de *branches* separadas para cada atividade, seguindo padrões de nomenclatura, e com utilização de *pull requests* para integrar códigos desenvolvidos por cada membro, com certeza ofereceu uma visão rica das possibilidades da ferramenta e elucidou formas de organização e funcionamento da mesma. É consenso entre os membros do grupo que todos saem deste trabalho melhor preparados para a utilização de ferramentas de versionamento de código em projetos grandes ou pequenos. Em consoante com o que foi apresentado neste documento, os participantes concordam em atribuir ao grupo a nota geral de 9,5 pelo aprendizado adquirido.