Fullstack - Introdução

Prof. Renan Ponick

Senac

Indicadores do saber ser

- **Participação** – Interage de forma contextualizada nas diversas atividades desenvolvidas em sala de aula.
- **Criatividade** – Demonstra criatividade na realização das atividades, respeitando os princípios éticos.
- **Responsabilidade** – Cumpre os prazos estabelecidos, respeitando as orientações metodológicas.
- **Espírito de equipe e cooperação** – Trabalha em equipe, interagindo em situações diversas, para atingir os objetivos comuns ao grupo.
- **Flexibilidade e respeito** – Respeita os diferentes pontos de vista, baseado em critérios éticos.

Indicadores

1. Analisa as ferramentas e práticas full-stack no ciclo de vida da aplicação
2. Codifica aplicações full-stack utilizando frameworks e bibliotecas
3. Codifica aplicações full-stack utilizando tecnologias avançadas
4. Identifica as tecnologias e estratégias de DevOps no desenvolvimento de aplicações

Habilidades

1. Identificar as ferramentas e práticas do desenvolvedor full-stack no ciclo de vida completo da aplicação
2. Desenvolver aplicações full-stack utilizando frameworks e bibliotecas
3. Desenvolver aplicações full-stack utilizando tecnologias avançadas
4. Diferenciar entre as tecnologias e práticas DevOps no desenvolvimento de sistemas

Bibliografia Básica

FOWLER, Martin. **Refatoração**: Aperfeiçoamento e Projeto. Bookman Editora, 2009.

HUNT, Andrew; THOMAS, David. **O Programador Pragmátic**o: de aprendiz a mestre. Bookman Editora, 2009.

MARTIN, Robert C. **Código limpo**: habilidades práticas do Agile software. Alta Books, 2019.

Bibliografia Complementar

KIM, G.; HUMBLE, J.; WILLIS, J.; DEBOIS, P. **Manual de DevOps**: como obter agilidade, confiabilidade e segurança em organizações tecnológicas. São Paulo: AltaBooks, 2018.

MARTIN, Robert C. **Arquitetura Limpa**: O guia do artesão para estrutura e design de software. Alta Books Editora, 2019.

OLIVEIRA, William. **O universo da programação** - Um guia de carreira em desenvolvimento de software. São Paulo: Casa do Código, 2018.

Agenda

22/07 - Introdução + Git configuração

23/07 - Git comandos e Github SSH

29/07 - Avaliação Git/GitHub

30/07 - CLAUDIA

Feedback Geral

Referente a última avaliação de QA

- Ótimo - 5
- Bom - 4
- Satisfatório -19

Combinados

Perguntem sempre que tiverem dúvidas;
Respeitem TODOS a sua volta;
USEM o tempo em sala para fazer os exercícios e tirar as dúvidas;
Meus slides são materiais de consulta (sim serão disponibilizados no Teams).

Combinados

As entregas deverão ocorrer conforme descrito na atividade. Só serão aceitos arquivos .zip ou .rar se for solicitado na tarefa para entrega de zip ou rar. Caso contrário as entregas serão do link do GitHub.

Combinados

Entregas na data e horário marcado.
QUALQUER entrega, após o horário, será considerado como **INSATISFATÓRIO**

Combinados

Entregas **SOMENTE** na tarefa do Teams, **QUALQUER** outra entrega, seja no chat do Teams, wtpp ou outros, não será considerada.

Combinados

Recuperações só serão realizadas nas datas marcadas, **APÓS** a realização da avaliação principal.

A **não entrega da avaliação** OU a **falta do aluno sem justificativa**, automaticamente colocará o aluno em recuperação;

A falta do aluno na recuperação, irá **ELIMINAR** a chance de outra recuperação.

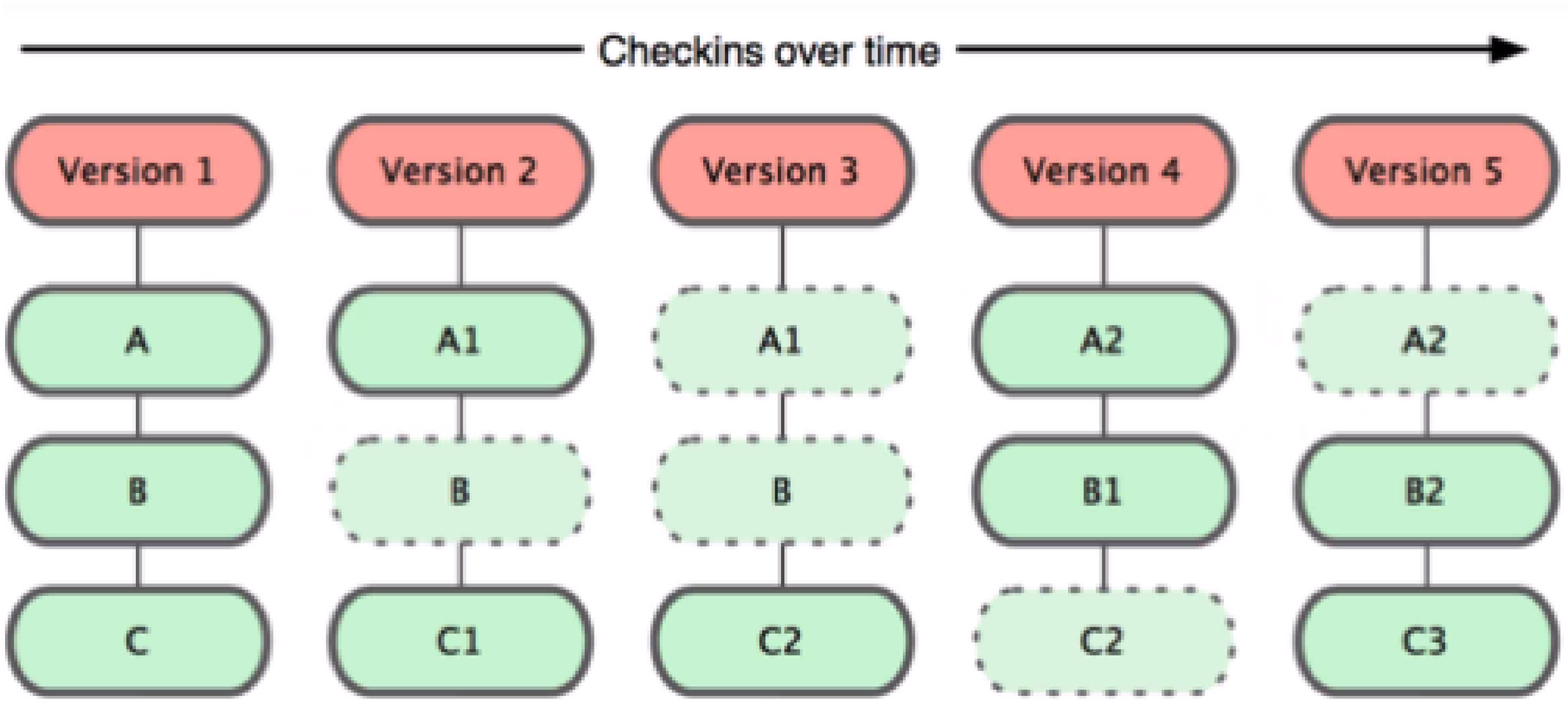
Dúvidas/reclamações?

Git

Git

O Git ferramenta de versionamento local, que salva o estado de cada arquivo na hora do versionamento e caso o arquivo não sofra a alteração ele cria um link simbólico para o arquivo não editado.

Git


Checkins over time
Version 1
Version 2
Version 3
Version 4
Version 5
A
A1
A1
A2
A2
B
B
B
B1
B2
C
C1
C2
C2
C3

Branchs

Branchs

Ramificações

É útil em situações nas quais você deseja adicionar um novo recurso ou corrigir um erro, gerando uma nova ramificação garantindo que o código instável não seja mesclado nos arquivos do projeto principal.

Equipe

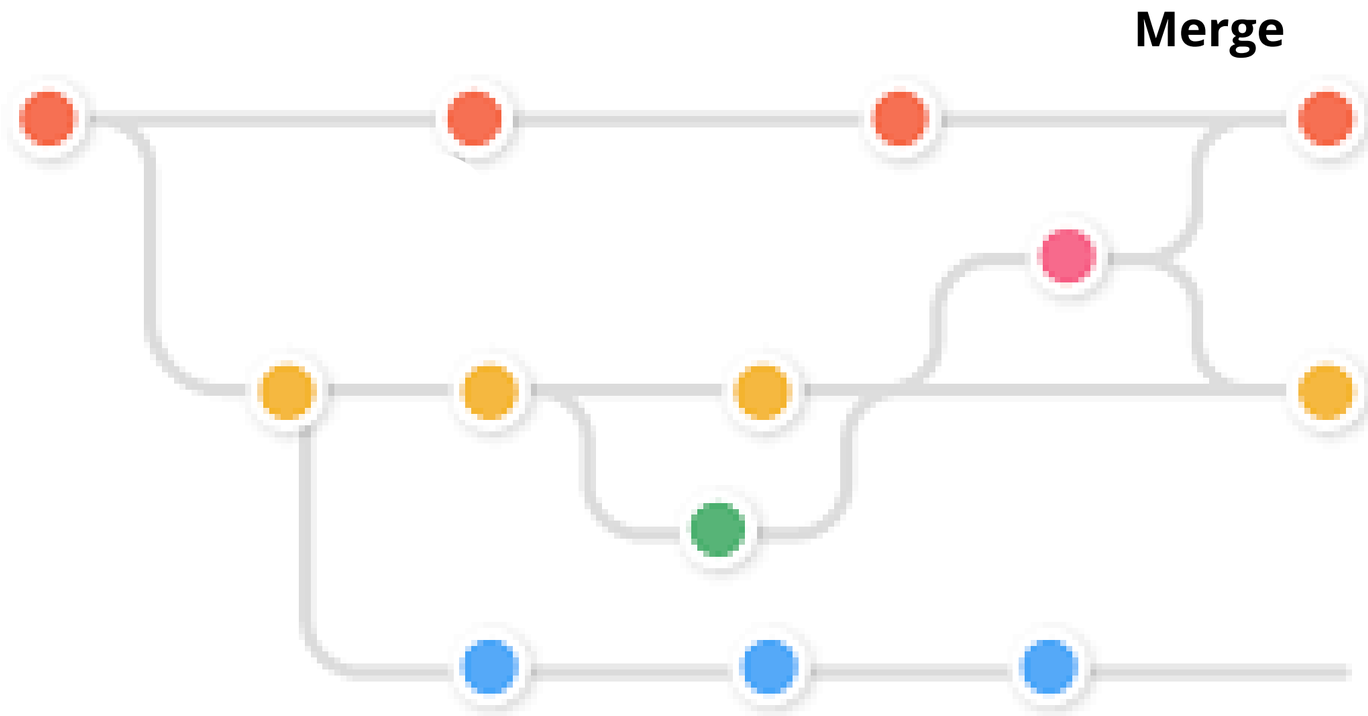
Main/Prod

Release/Hml

Dev

BugFix

Feature



Boas praticas

NUNCA COMMITAR NA MAIN/MASTER/PROD

Para isso abrimos branchs

Nomenclaturas

feat: Algo relacionado com features que você adicionar;
fix: Algo para corrigir;
docs: Algo relacionado a documentações, README e afins;
style: Algo relacionado com estilização;
refactor: Algo relacionado com refatoração(refazer);
perf: Algo relacionado a performance;
test: Algo com testes;
chore: Algo para coisas relacionados a build, configs e afins. Exemplo: Use esse, seja atualizando a versão do pacote ou instalando novas dependências.

Verificação

Abra o CMD, Terminal de Comando, Shell ou PowerShell
Digite git --version

```
C:\Users\Renan Ponick>git --version
git version 2.37.1.windows.1
```

Configuração

Configure também o seu git local, com os seguintes comandos:

```
>git config --global user.name "Seu Nome"

>git config --global user.email "seuEmail@gmail.com"
```

Até aqui tranquilo?

E o GitHub?

Dia a Dia

O desenvolvimento de software possui uma característica forte quanto ao desenvolvimento paralelo(em conjunto) do software pela equipe de desenvolvedores.

EX: Login, Cadastro, BD

E o GitHub?

Repositório remoto

GitHub - HTTP ou SSH?

Clonando com HTTPS

Vantagens:

Facilidade de Configuração: HTTPS é mais fácil de configurar inicialmente. Não requer a criação de chaves SSH e é mais direto para usuários novos.

Acesso Universal: Funciona em praticamente qualquer ambiente sem necessidade de configuração adicional, desde que o Git esteja instalado.

Desvantagens:

Autenticação Repetitiva: Cada vez que você faz uma operação que requer autenticação (como push ou pull), você precisa fornecer suas credenciais (a menos que você use um gerenciador de credenciais).

Menor Segurança: Armazenar senhas ou tokens de acesso em sistemas locais pode ser menos seguro em comparação com a utilização de chaves SSH

Clonando com SSH

Vantagens:

Autenticação Segura: SSH usa criptografia assimétrica, o que é mais seguro do que usar senhas. Uma chave privada SSH é utilizada localmente e a chave pública é adicionada ao GitHub.

Sem Necessidade de Senhas: Uma vez configurado, você não precisa fornecer suas credenciais a cada operação. Isso simplifica o processo e melhora a segurança.

Conexões Automáticas: Com SSH, uma vez que a chave está configurada, o Git usa automaticamente essa chave para autenticar todas as operações.

Desvantagens:

Configuração Inicial: Requer configuração inicial, incluindo a geração de chaves SSH e a adição da chave pública ao GitHub.

Gerenciamento de Chaves: Se você usar várias máquinas, precisará configurar as chaves SSH em cada uma delas.

Configurando - SSH

SSH - GitBash

Verificar se existe chave ssh.

ls -al ~/.ssh

Adicionar uma nova chave. (ID)

ssh-keygen -t **ed25519** -C "your_email@example.com"

Inicializar agente-ssh.

eval "$(ssh-agent -s)"

Adicionar chave ssh ao agente.

ssh-add ~/.ssh/id_ed25519

SSH - GitBash

Copiar chave ssh.

clip < ~/.ssh/id_ed25519.pub

Adicionar chave no github

Github -> Settings -> SSH and GPG keys -> New SSH key -> Colar

Coloque um título que identifique a chave

Ex: SENAC-SALA-206-PCXPTO

Testar conexão

ssh -T git@github.com

yes

Referência

https://docs.github.com/pt/authentication/connecting-to-github-with-ssh/about-ssh

Dúvidas?
Alguém não conseguiu configurar?

Recomendação

Agora com ambos, Git e Github, configurados recomendo criar um repositório no GitHub para suas anotações de estudo (independente a disciplina ou instituição).

Dessa forma conseguirá realizar anotações durante as aulas, inclusive deixar exemplos práticos no mesmo, sem se preocupar em perder os arquivos ou em ter muitos arquivos em locais diferentes

(o mesmo poderá ser privado caso não queira compartilhar o que esta aprendendo)

Exemplo: https://github.com/renanponick/notas-de-estudo

Atividade

Para que seu perfil como dev seja notado, é interessante ter um portfólio montado.

Com base nisso, crie um repositório no seu GitHub que sirva como seu portifólio pessoal. Recomendo que o mesmo tenha nome de “portifolio”. Dentro do mesmo recomendo criar um portifólio utilizando HTML e CSS básico.

Nele traga suas informações da mesma forma que traz em um curriculo *(Como ficará no GitHub cuide com dados sensíveis)