

Versionamento de código:

Professor Rafael;

Nas nossas aulas de versionamento de código o professor Rafael nos instruiu em como utilizar essa prática em nossos futuros trabalhos. Onde podemos assertivamente e gradativamente utilizar esses conhecimentos para garantir um bom desempenho em nossas futuras áreas profissionais.

Oque aprendemos em aula:

O Professor Rafael nos ensinou diversos assuntos referente a o Versionamento, dentre eles estão:

- **Base para a criação de apps, além de instruir o que é necessário para o processo.**
- **Nos foi ensinado o que são versões “Alfa e Beta”, e oque significa os números depois delas na criação de apps e jogos.**
- **Aprendemos sobre a “Necessidade de Versionar”, e qual a importância do controle de versão para ajudar em nossos trabalhos e atividades.**
- **Também aprendemos sobre os “Gits” e “repositórios”, para o controle de versão de nossas atividades, além de nos apresentar o “GitHub” site para armazenar seguramente nossos trabalhos.**
- **Utilizamos em aula o “Visual Studio Code”, ferramenta utilizada para a criação de web sites ou também para aprender sobre programação.**
- **Entendemos o que são as “Branchs” e como são usadas e para que servem na área do versionamento.**
- **O que são “Commits” e como ele interage com o repositório.**

Criação de Apps: O que praticamos em aula.

Nas nossas aulas de Versionamento, praticamos como criar um App, mais precisamente um App de locomoção, que visa o transporte de pessoas com certas deficiências motoras e físicas.

Processo na criação do conceito desse App:

1. Na criação do App, iniciamos decidindo a ideia e o nome:

Nome: DeficPort.

Ideia Central: O objetivo central do App seria como um melhor meio de transporte para pessoas com dificuldade de locomoção, onde elas poderiam encontrar caronas mais especializadas para suas condições. Além de tornar o atendimento mais especializados, os usuários que precisarem com urgência (seja qual for), não iram precisar esperar em uma possível fila de espera.

2. Depois decidimos como e onde o App funcionaria: OBJ

Em quais plataformas o App funcionaria: Decidimos que ele funcionaria, tanto para Android quanto IOS. Para assim abranger o máximo de pessoas o quanto possível, optamos por algo mais portátil que são os celulares, para o App sempre estar na mão de quem precisa.

Como o App funciona: O App registrará suas informações para poder saber como melhor atendê-lo. Assim um motorista parceiro adequado para seu transporte virá a você, e o levará ao seu destino. Isso com um simples toque no smartphone.

Versões Alfa e Beta: O que nos foi ensinado.

Na criação do App, nos estipulamos que a versão primaria do “DeficPort” seria “V.1.0.0” ou melhor, a versão Alfa, e esses são os significados por trás desses nomes e números:

Versão, o que são:

Esses números e nomes são o que separa as “versões” do App, em alfa, beta e a versão final, além das mudanças criadas no meio do projeto:

Versão “V.1.0.0”: Esses números sugerem que o sistema está na versão “Alfa”. Nessa versão já contém a ideia inicial e tudo mais, mas contém muitos bugs e falta de segurança. O primeiro dígito pode mudar para 2, virando a versão “Beta” uma atualização da “Alfa”.

Versão “V.1.1.0”: Uma nova versão do sistema. Ao alterar o segundo número significa que houve uma atualização, onde o sistema pode ter ganhado mais uma funcionalidade.

Versão “V.1.1.1”: Já essa alteração, que é o último dígito, simboliza uma correção de bugs encontrados, e a implementação de novas funções de segurança para o App.

Github, Branchs & Commit: Controladores de Versão.

Tudo sobre os controladores de versão, para que servem e como vão ser usados para ajudar e facilitar o trabalho do profissional de Versionamento de Códigos, na qual refere-se ao processo de manter um registro de alterações feitas em um arquivo ou conjunto de arquivos ao longo do tempo. Com esse controle, você pode rastrear ou, até mesmo, reverter o arquivo para um estado anterior, se necessário, e essas ferramentas são usadas no processo.

GitHub: O Git é um sistema de controle de versão distribuído que armazena informações em uma estrutura de dados chamada repositório, onde os usuários podem ver tanto as versões iniciais quanto as versões mudadas até por outros usuários. Assim temos total controle pelo acesso aos seus trabalhos, além de salvar também deixa seguro todos os seus trabalhos, onde apenas criador tem acesso total.

Branchs: Uma branch no Git é essencialmente uma linha separada e independente de desenvolvimento. Utilizar Branches no Git permite que os desenvolvedores trabalhem em diferentes funcionalidades simultaneamente sem interferir no trabalho dos outros. Além disso, entender como colaborar com outros desenvolvedores é essencial em ambientes de equipe.

Commits: Os commits são alterações feitas em um repositório de código durante o desenvolvimento de um projeto. Cada commit é uma marcação de uma alteração feita no código, por exemplo, correção de bugs, adição de novas funcionalidades e etc. Serve para os usuários saberem quais foram as alterações feitas, além de as salvar, podemos dizer que foi feita uma nova versão em commit como uma ramificação do projeto inicial excelente para.

Repositórios:

Um repositório é usado como um local onde o código fonte de um projeto é guardado e gerenciado, é usado para controlar diferentes versões do código ao longo do tempo. E de conhecimento que o "GitHub" é um site que torna isso facilmente possível.

é usado para controlar diferentes versões do código ao longo do tempo de forma digital centralizado que os desenvolvedores usam para fazer o gerenciamento de alterações no código-fonte de uma aplicação, que é o caso do "GitHub" um site que torna esse armazenamento possível sendo uma das plataformas mais utilizadas hoje para armazenar alterações ou códigos de forma segura e autoral.

NOMES: João Victor Guimarães Moura/ Eric Queiroz/ Augustus/ Ezequiel da Cruz: