

Reunião de Orientação 04

Igor Pires dos Santos

igor.pires@ice.ufjf.br

Orientador: Rafael Bonfim



Programa de Pós-Graduação em Modelagem Computacional
Universidade Federal de Juiz de Fora

13 de novembro de 2020

Introdução

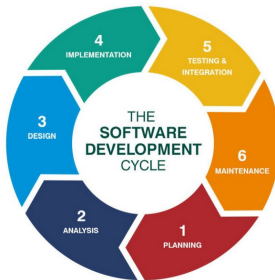
► Proposta Passada

- Outubro: Realização do Toefl iBT, Adicionar possibilidade de se executar o IGU 1.0 via linha de comando, Finalizar Repositório de imagem com o IGU 1.0 Instalado, Definir os experimentos relevantes com CCO (Ye, Pressão e Fluxo?)(Quais parâmetros ?), Definir se um sistema de CI/CD será aplicado e QUAL e COMO.
- Novembro: Finalização da Graduação, Rodar os experimentos relevantes com CCO utilizando o IGU 1.0 , escrever (e plotar) estes resultados.
- Dezembro: Finalizar parte Gráfica IGU 2.0, finalizar sistema de CI/CD, re-analisar funcionalidades do IGU 2.0 à adicionar e começar dissertação.
- Janeiro: Começar IGU 2.0 e sua parte distribuída e escrever dissertação.

► **Proposta Atualizada**

- Outubro: Realização do Toefl iBT, Adicionar possibilidade de se executar o IGU via linha de comando (InGU), Definir os experimentos relevantes com CCO
- Novembro: Finalização da Graduação, Rodar os experimentos relevantes com CCO utilizando o IGU , escrever (e plotar) estes resultados.
- Dezembro: Finalizar parte Gráfica IGU atualizada, finalizar sistema de CI/CD, re-analisar funcionalidades do IGU à adicionar e começar dissertação.
- Janeiro: Começar parte distribuída e escrever dissertação.

► O ciclo de vida de um software



Synotive

- ▶ Pensando na situação atual em que possua o código do IGU e em paralelo possuo esse código do "IGU 2.0", toda modificação que eu faria no código anterior se torna antiquada, porque na primeira versão as conexões entre os objetos (gráficos árvores arteriais, basicamente os objetos inteligentes) eram estáticas.
- ▶ Ainda na primeira versão todos os objetos eram também obrigatoriamente gráficos. Enfim, diversas mudanças novas no código acarretariam em soluções antiquadas no novo contexto do IGU 2.0 que apresentaria uma solução mais simples e unificada. Possibilitando ainda a parametrização de todas as variáveis (uma vez que o arquivo VTK é lido e pode ser manipulado, escalado e itera), podendo ainda ter ou não o fator gráfico e ainda trocar o método iterativo em tempo de execução.

- ▶ Pensando nisso e nas atividades conseguintes eu juntei os dois projetos de forma separada, dentro do mesmo repositório agora possuirei 3 compiláveis (nomes sugestivos):
- ▶ IGU: Iterador Gráfico Universal, IGU "antigo" com todo o seu código e parte gráfica.
- ▶ InGU: Iterador não-Gráfico Universal, programa console com toda a arquitetura IGU atualizada.
- ▶ PERIGU: PERimeter for IGU: Programa console simples que alterna entre as versões do programa (opcional, simplifica na hora de executar comandos do console).

- ▶ InGU + IGU
- ▶ (mostrar no Qt)