

Universidade Federal Fluminense

**Professor:**

Leonardo Gresta Paulino Murta

**Engenharia de Software II**

**Relatório Final**

**Alunos:**

Gabriela Pinheiro

Karina Lemos

Thiago Nascimento

Tiago Lacerda

Valesca Moura

Niterói, Rio de Janeiro

1º semestre de 2022

## 1. Dificuldades encontradas:

- Elaboração da WBS

Na etapa inicial do projeto houve muita dificuldade para a elaboração da WBS. Isso ocorreu em virtude de não entendermos muito bem quantos elementos deveriam estar presentes nela para e o quanto isso iria afetar a gestão do projeto para o desenvolvimento dos próximos documentos. Como consequência, gastamos um tempo muito maior do que o esperado para terminar, além de precisarmos refazer duas vezes a mais para no fim chegarmos no resultado desejado.

- Desenvolvimento do gráfico de Gantt

Em relação ao planejamento do cronograma de desenvolvimento, e especificamente sobre o Gráfico de Gantt, a falta de uma ferramenta adequada para a criação dessa documentação, tornou toda a sua execução muito mais complexa do que realmente deveria. Para o nosso caso, foi utilizado uma aplicação web para a construção do gráfico (<https://www.onlinegantt.com/#/gantt>) porém ao utilizarmos, percebemos inúmeros problemas lógicos da ferramenta (não fazia a adaptação à mudança de uma atividade com as suas dependências por exemplo), além de que não colocava por conta própria o caminho crítico, forçando a gente a realizar esse processo manualmente em uma outra ferramenta de edição de imagem, além disso o formato de exportação das imagens não funcionou de modo adequado, gerando mais problemas para a finalização.

- Monitoramento e controle do projeto

Em virtude de não termos utilizado uma ferramenta como o Trello para separarmos melhor as atividades que estão sendo realizadas em cada sprint, as atividades que já foram feitas em cada sprint e o tempo que foi gasto para cada uma delas, o cálculo dos valores de EV e AC se tornaram confusos ao longo do tempo e com isso a análise de valor agregado. Sendo assim, muitas atividades não tiveram seu tempo gasto medido corretamente, resultando em valores que não representam fielmente a realidade.

- Testes

Para a execução dos testes unitários, não conhecermos direito o funcionamento da biblioteca *unittest*, tornou a sua utilização um pouco confusa para utilizar o mock de um método que pertencia a uma classe diferente da classe que estava sendo testada, sendo utilizado o *patch* no lugar do mock em si para resolver o problema.

- Heroku

Dificuldade em debugar erros que aconteceram apenas no Deploy da aplicação e não eram visíveis na execução local.

## 2. Decisões Tomadas

- Aplicação WEB ao invés de Aplicativo Mobile
- Unittest ao invés de Pytest
- Usar Trunk Based Development
  - Minimizar esforço de merge tendo branches com curto ciclo de vida;
  - Deploy contínuo e automático;
  - Necessidade de um revisor na aprovação de um PR.
- Usar Heroku
  - Plataforma gratuita e de fácil integração ao GitHub actions.
- Para a Inteligência Artificial foi escolhido o algoritmo de busca min max, onde ele irá buscar uma jogada que gere a melhor pontuação possível se baseando no valor da peça e sua posição no tabuleiro. Buscando melhorar o desempenho, foi implementado o sistema de poda alfa beta, que poda a árvore se baseando nos valores previamente encontrados.

## 3. Listagem da participação

Para esse projeto, foi decidido que cada pessoa seria o responsável primário por alguma parte do trabalho, porém sempre teria pelo menos mais alguém presente que também soubesse como está o andamento, podendo ajudar caso seja necessário.

Além do que será mostrado a seguir, algumas atividades foram feitas totalmente em conjunto por todos, como:

- Grupo
  - Construção WBS e do gráfico de Gantt;
  - Cronograma de Desenvolvimento;
  - Relatório final.
  
- Gabriela Pinheiro:
  - IA.
  - Testes unitários para a IA.
  
- Karina Lemos:
  - Frontend;
  - Escopo do projeto e do produto;
  - Análise de riscos;
  - Plano de testes;
  - Testes de sistema.
  
- Thiago Nascimento:
  - Serviços para a integração entre o Backend e o Frontend;
  - Testes unitários para Serviços para a integração entre o Backend e o Frontend;
  - Estimativa de esforço, custo e orçamento;
  - Monitoramento e Controle do projeto.
  
- Tiago Lacerda:
  - Máquina de regras.
  - Testes unitários para a máquina de regras.
  
- Valesca Moura:
  - Infraestrutura;
    - Configuração Github;
    - Configuração Heroku;

- Configuração Django.
- Frontend
  - Requisições jQuery;
  - Funções javascript.
- Integração frontend-backend;
- Backend
  - Controllers;
  - Integração serviços-controllers.