Taciturn-duck

Plano do Projeto

# Introdução

O projeto visa construir um sistema de aconselhador de matricula, onde teremos dois papeis principais dentro do sistema sendo um, os funcionários da secretaria que deverão preencher os dados das disciplinas dos cursos e também as turmas para cada semestre, por sua vez o aluno preenche as disciplinas que ele já concluiu e o software irá gerar a grade sugerida para ele, aluno pode também escolher refazer a grade para ter outra opção de horários, e ainda pode colocar as disciplinas que não deseja cursar no semestre.

# Organização do Projeto

A equipe do projeto é composta por um Gerente de projeto, um analista, um projetista e desenvolvedor, e um testador.

# Praticas do Projeto

O projeto será desenvolvido na linguagem Ruby, suportando múltiplos acessos simultâneos, todos os membros, independente da função, estarão presente na programação do projeto e nas funções de gerencia do projeto.

# Marcos do Projeto

Para melhor entendimento, olhar no arquivo WorkItens

|  |  |  |  |
| --- | --- | --- | --- |
| **Iteração** | **Objetivos** | **Agenda** | **Velocidade estimada em horas** |
| I1 | Objetivos   1. Iniciação    1. Documentos de projeto    2. Levantamento de requisitos    3. Casos de uso definidos    4. Casos de teste | 10/09 | 70 |
| I2 | Objetivos   1. Primeira parte da Elaboração    1. Refinar e identificar requisitos    2. Desenvolver a arquitetura    3. UC05    4. UC06    5. Testar | 27/10\* | 100 |
| I3 | Objetivos   1. Segunda parte da Elaboração    1. Corrigir defeitos    2. Desenvolver o incremento    3. Testar    4. UC04\*    5. UC07\*\*    6. UC08\*\* | 26/11 | 70 |
| I4 | Objetivos   1. Construção   1.1 Documentação  1.2 UC01  1.3 UC2\*  1.4 UC3\*  1.5 Novos requisitos  1.6. Corrigir erros  1.7. Testar | 06/06/2016 | 500 |
| I5 | Objetivos  1. Transição  1.1. Corrigir e testar problemas  1.2 Preparação para Release | 08/08/2016 | 150 |

# \* Mudança no Plano do projeto.

# \*\* novos requisitos.

# Desenvolvimento

O software será construído a partir dos casos de uso e dos objetivos de cada iteração, sendo incremental cada iteração deve ter uma contribuição para o produto final.

# Lições Aprendidas

Não se aplica ao projeto, equipe composta por membros que nunca trabalharam juntos.