

API - Aprendizagem por Projetos Integrados

2023-2

Cliente:	Parceiro interno	
Período/Curso:	1º DSM (Desenvolvimento de Software Multiplataforma)	
Professor M2:	Arley Ferreira de Souza	arley.souza@fatec.sp.gov.br
Professor P2:	Antônio Egydio	antonio.graca@fatec.sp.gov.br
Cliente:	Antônio Egydio	
Kick off:	31/08/2023 às 19h15	
Tema do Semestre		
Aplicação Web de instruções para desenvolvimento de projetos ágeis (SCRUM) na área de TI		
Conhecimentos exigidos no semestre		
<ul style="list-style-type: none">• Documentação de Software (briefing, wireframes);• Controle de versão (GitHub);• Processo de Desenvolvimento Ágil (Scrum);• Lógica de programação (JavaScript);• Programação para Web (HTML e CSS);• Layout;• Comunicação formal falada e escrita (norma culta);• Tecnologia utilizáveis (treinamento) serão de plena responsabilidade do grupo desenvolvedor (tecnologias básicas para a confecção do projeto, poderão ser utilizadas mesmo que artesanalmente).		
Desafio (problema)		
<p>Desenvolver um sistema web que indique todos os processos e artefatos da metodologia ágil (Scrum), todos os processos deverão ter conceitos e fundamentos com referências, bem como a aplicação com exemplos práticos para a plena compreensão do usuário do sistema.</p> <p>Deverá criar um formulário para avaliações parciais e final (com totalização).</p> <p>O sistema deve primordialmente ter foco na disseminação dos processos para que os usuários possam replicar todos os passos em futuros desenvolvimentos.</p> <p>A prioridade secundária será a compreensão de habilidades (Skills) importantes para um profissional de tecnologia da informação.</p> <p>Todo o material de conhecimento publicado no sistema deverá ter forma indireta (Conhecimento adquirido e repassado pelos desenvolvedores).</p> <p>Bem como toda a documentação de desenvolvimento deverá ser lastreado no GitHub com todos os processos necessários para a fidelização do cliente.</p>		

Faculdade de Tecnologia Professor Francisco de Moura – FATEC Jacareí

Requisitos
<p>Requisitos Funcionais:</p> <ul style="list-style-type: none">• Apresentação via GitHub;• O sistema web deverá ser intuitivo e não ter poluições de informações, isso deverá ser uma preocupação constante dos desenvolvedores, sugestões de controles serão bem-vindas e bem avaliadas;• Criar um sistema de avaliação (Processo, Produto e Conhecimento em disciplina, SM., PO., TD);• Processo Scrum;• Burndown• Product e Sprint Backlog;• Sprint;• Sprint Planning;• Sprint Review;• Sprint Backlog;• Dailys;• Retrospective;• DOR – Definition of Ready;• DOD – Definition of Done;• Planning Poker;• Kanban;• Artefatos do Scrum;• MVP. <p>Requisitos Não Funcionais:</p> <ul style="list-style-type: none">• Documentação no GitHub;• Linguagem de programação JavaScript.• Linguagem de marcação HTML e estilos CSS.• Uso do framework Bootstrap. <p>Requisitos Não Funcionais:</p> <ul style="list-style-type: none">• JavaScript, HTML, CSS, SK, Bootstrap, Figma e Jira.

Requisitos avaliados:

1. O produto atingiu o proposto na sprint;
2. O produto está satisfatório;

Faculdade de Tecnologia Professor Francisco de Moura – FATEC Jacareí

3. O repositório no GitHub deve estar organizado em 2 repositórios, sendo um repositório para a documentação, um repositório para o código;
4. O repositório de documentação deverá ter:
 - a. Sobre o projeto: breve apresentação do problema a ser resolvido;
 - b. Entregas de sprints: cronograma de entregas com o link para as sprints;
 - c. Requisitos: requisitos funcionais e não funcionais com identificação de prioridade (média, baixa e alta);
 - d. User stories: detalhamento dos requisitos escrito no formato Ator, ação e motivo. Se necessário incluir épicos;
 - e. Product Backlog: lista priorizada de todos os requisitos, funcionalidades, melhorias e correções de bugs que compõem o escopo do produto a ser desenvolvido;
 - f. Sprint backlog;
 - g. MVP (Minimum Viable Product);
 - h. Gráfico burndow de cada sprint;
 - i. Print das telas explicando o que foi entregue;
 - j. Tecnologias: ferramentas, linguagens, bibliotecas e tecnologias foram usadas no projeto;
 - k. Equipe: nomes dos integrantes da equipe, função (product owner, scrum master ou dev team) de cada membro e link para o GitHub e LinkedIn de cada membro.

Exemplos de organização do API no GitHub:

<https://github.com/ProManage-FatecSJC/pro-manager-documentation>

<https://github.com/The-Bugger-Ducks/owl-partners-documentation>