

Documentação do Projeto – Sprint 1 – v1.0

Totem System – Grupo 9

2ADSB / 2022

Objetivo:

A Totem System S.A tem como objetivo monitorar totens de autoatendimento em estações de metrô com foco na cidade de São Paulo; entregando dados em tempo real através de nossa plataforma online, contendo informações sobre todos os totens cadastrados, seu respectivo funcionamento, informações sobre o cpu, memória, tempo de resposta, sobrecarga, interação com usuário (touchscreen, funcionamento de teclado), sistema de pagamento e outros.

Ainda: será disponibilizada uma dashboard interativa para tomada de decisão rápida por parte da empresa responsável.

Justificativa:

Máquinas de autoatendimento são uma realidade hoje em dia; seja para pagamento de estacionamento, para fastfoods, para sistemas bancários ou para compra de passagens de metrô - o foco deste projeto. A verdade é que essas máquinas facilitam muito a vida do usuário final, evitando filas longas e dando autonomia para o cliente.

Esses equipamentos podem ser acessados a qualquer momento, sem depender de uma pessoa atrás de uma cabine para fazer a interação. E a medida que a tecnologia avança e os componentes eletrônicos ficam mais baratos e acessíveis, é fácil pensar que não vai demorar muito até que estes equipamentos dominem o mercado – em diferentes setores da sociedade, como citado.

Uma vez estabelecido esse cenário nos grandes centros urbanos como a cidade de São Paulo, é comum pensar nos eventuais problemas que podem acontecer caso um equipamento deste apresente defeito. Um funcionário é capaz de se adaptar a diferentes situações e buscar soluções; uma máquina apenas executa o que está em sua programação, e uma vez apresentado um defeito, o equipamento nada pode fazer por si mesmo, depende de que alguém vá reparar seu erro.

Considerando este ponto, faz-se necessário um constante monitoramento dos totens de autoatendimento em diferentes pontos da cidade. É necessário verificar diversos itens, como conexão com internet, integridade do CPU, memória RAM, funcionamento do touchscreen e outros. A capacidade de monitorar totens de autoatendimento em tempo real dá a liberdade e a segurança para as empresas que a implementam que seus clientes finais raramente ficarão de mãos atadas; sem contar que seria possível evitar o prejuízo financeiro gerado por uma máquina apresentando defeito.

Inovação:

Implementar, junto com o sistema de monitoramento, uma dashboard interativa, que além de mostrar os dados de cada totem de autoatendimento, será capaz de demonstrar de forma gráfica e rápida quais totens apresentam problemas, quais totens estão sem funcionamento e quais precisam de atenção. Além, é claro, de uma maior interação do administrador do sistema com o suporte, onde ambos terão certa autonomia e maior agilidade para executar suas tarefas de forma independente.

Planejamento do Projeto:

1. Fazer a Ideação do projeto, com seu StoryBoard, Lean Canvas, User Storyes etc;
2. Fazer a pesquisa e buscar encontrar a inovação do projeto;
3. Fazer a modelagem de dados do projeto;
4. Criar o site institucional do projeto;
5. Estruturar o Planner;
6. Iniciar o GitHub como repositório online e criar conta na Azure;
7. Fazer desenho de solução técnica e de negócio;
8. Fazer apresentação da ideia do projeto para a banca;

Escopo:

- Ambiente Azure + Github;
- Site estático institucional;
- Diagrama de Banco de Dados;
- Plano de resposta-Lições Aprendidas;
- StoryBoard;
- Miniprojeto Java (Individual);
- Histórias do Usuário (User Story), Requisitos;
- Cliente Linux (WSL);
- Planner;
- Definição de Negócio;
- Visita (virtual);
- Inovação;
- Proto-Persona;
- Slide de Apresentação;

Marcos do Projeto:

- (15/03/2022) - Entrega da primeira Sprint com foco na Ideação;
- (21/03/2022) - Entrega da subsprint 1;
- (28/03/2022) - Entrega da subsprint 2;

- (04/04/2022) - Entrega da subsprint 3;
- (12/04/2022) - Entrega da subsprint 4;

Restrições do Projeto:

- Limite da entrega da primeira sprint até a data de 12/04/2022;
- Uso do banco de dados relacional MySQL;
- Uso do Planner;
- Uso de site estático em HTML/CSS/JS;
- Uso da linguagem de programação Java;
- Equipe restrita ao máximo de 6 (seis) membros;
- Uso da ferramenta de versionamento git / github;

Premissas do Projeto:

- Notebooks com internet Wi-Fi e demais tecnologias imbutidas, como IDE para Java, VSCode, etc;
- Auxílio de professores e monitores na realização do projeto, assim como o tempo de resposta de dúvidas dentro do prazo;
- Disponibilidade do edifício Digital Building das 8h AM até 22h PM de segunda à sexta, assim como as demais dependências do mesmo (salas e biblioteca) para uso da equipe;
- Equipamento de apresentação (data show) assim como demais cabos (HDMI) disponibilizados pelas salas da SPTech;
- Funcionamento adequado das plataformas Moodle e Outlook, evitando o impedimento da comunicação com os professores;
- Equipe com disponibilidade diária de pelo menos duas horas extras, extra-classe, para a execução do projeto;

Equipe Envolvida:

- Um PROJECT OWNER, pessoa responsável por lidar com o cliente, desenvolver a documentação do projeto, o backlog do projeto e acompanhar, junto com o cliente, o andamento do projeto, sugerindo mudanças junto ao cliente. É um cargo rotacionado a cada subsprint;
- Um SCRUM MASTER, pessoa responsável por liderar a equipe, organizar as reuniões diárias (daily), cobrar a equipe dos entregáveis da sprint e saber se a equipe está apta ou não para cumprir os requisitos. É um cargo rotacionado a cada subsprint;
- A equipe, composta de 3 a 4 membros, responsável por entregar as diferentes partes do projeto, comparecendo diariamente a reunião e dando feedback ao líder de equipe. A equipe pode ser dividida em algumas partes, como:

- Front-End: responsável por toda a parte de design de projeto, tanto para o portal, quanto calculadora e elementos de apresentação de produto;
- Back-End: responsável por toda a parte que envolve código-fonte, de página, de projeto e estrutura;
- DBA: Responsável pelo banco de dados do projeto, assim como os modelos relacionais e sua estruturação lógica;
- QA: Responsável por testar e verificar a qualidade do software, encontrando e resolvendo bugs.

Arquitetura:

- Banco de dados relacional MySQL;
- Protótipo do site escrito em tecnologia web básica (HTML, CSS, JavaScript);
- Notebooks DELL com OS Ubuntu ou HP com OS Windows 10;
- Software de gestão de projetos Planner;
- Software de prototipagem Figma ou Canvas;
- Equipamentos periféricos como mouse, pendrive (flashdrive) e celulares (opcionais);
- Pacote Office preferencialmente com Word e Powerpoint para realização de apresentações e escrita de documentação. Disponível na nuvem;
- IDE Netbeans para programar em Java;
- Sistema de controle de versão Git assim como seu repositório na nuvem;