

DOCUMENTAÇÃO

Objetivo:

Utilizar os conhecimentos adquiridos no projeto anterior para abstração de um novo problema voltado ao monitoramento de dados de CPU em totens de estações de trem.

Site desenvolvido em HTML, CSS, JavaScript e Java para conexão com o monitoramento de totens. O monitor terá acesso as dashboards que fornecerá as informações compiladas sobre o funcionamento de todos os totens, com indicadores de uso para comparação, previsão de problemas e tomada de decisão. A dashboard será conectada diretamente ao funcionamento da CPU do sistema através do Shell e seus dados serão guardados no BD para utilização em prol de um funcionamento efetivo para todos os âmbitos, desde usuário, até o próprio gerente do metrô.

A inovação escolhida para o projeto é o sorteio de produtos baseado no usuário que utiliza os totens que tenham nossa aplicação. Através de um QR Code que o redirecionará para uma aplicação web conectada ao BD, aumentando assim seu contador conforme a quantidade de vezes que ele utiliza o totem da estação. A partir de uma determinada quantidade, o usuário poderá participar de sorteios de prêmios, que ao ser realizado, enviará ao usuário via slack uma notificação informando o ganho do produto e sua data prevista de entrega. Incentivando assim o uso dos totens que tenham a nossa tecnologia, e garantindo um funcionamento melhor do transporte público.

Planejamento:

- Scrum para abordagem do tema;
- Utilizar da plataforma de versionamento para backup e desenvolvimento do projeto da equipe (Github);
- Criação do protótipo das telas do site a fim de desenvolver o front-end mais rapidamente e homologar a aplicação com o cliente antes de desenvolver cada tela;
- Planejar e organizar a entrega dos requisitos da Sprint (Backlog) e utilizar da ferramenta de planejamento Planner para definição de atividades e entregas da semana;
- Desenvolver a regra de negócio consistente para sustentação do site e segurança dos usuários;
- Desenvolver um site com login e cadastro em JavaScript conectado ao BD, garantindo acesso a usuários específicos e melhor funcionamento do produto;
- Configurar ambiente na nuvem para armazenar a solução (Azure);
- Coletar os dados de funcionamento da CPU de totens do metrô via Shell realizando comandos para captação de dados da CPU da estação;
- Utilizar de API's para cadastro e consulta de dados a fim de conectar diferentes tecnologias e melhorar a integração do sistema;
- Criar uma dashboard para interação do monitor com o funcionamento das máquinas;
- Fornecer indicadores de uso dos totens para melhor visualização do responsável pelas máquinas;
- Envio de notificações para o usuário da aplicação com base em Slack, fornecendo guias prático sobre procedimento de suporte e dados de comparação sobre cada máquina daquela estação em específico;
- Utilização de uma API de cadastro e acesso voltada a inovação em específico. Onde o usuário do totem lerá um QR Code que o redirecionará para o cadastro ao sorteio e, caso já cadastrado, irá armazenar mais um ponto no cadastro do usuário em específico e, a partir de determinado ponto concorrerá a prêmios pelo uso de totens que contenham nossa tecnologia.

Escopo:

- **Scrum:**
 - Abordagem Scrum baseada em reuniões periódicas para atualização de informações sobre andamento do projeto e tomadas de decisão;
- **Github:**
 - Utilização da plataforma de versionamento de código baseada em branches individuais para subida de arquivos beta e organização do projeto;
- **Protótipo de telas do site:**
 - Desenvolvimento do protótipo de telas do site através do Figma, definindo cores base do site e abordagem dinâmica de navegação;
- **Backlog:**
 - Desenvolvimento de planilha de Backlog utilizando Excel, tabela criada com a classificação, ordem de execução e dificuldade da tarefa;
- **Regra de negócio:**
 - Elaboração da regra de negócio do site pautada na navegação do monitor dos totens do metrô.
 - Cada monitor terá seu próprio acesso através do login do site que terá seus dados contabilizados no sistema pelo gerente do metrô.
 - Cada monitor terá acesso à sua própria dashboard, e cada gerente terá acesso a uma página para cadastrar o monitor de sua estação.
 - O monitor terá visualização de todos os totens que ele é responsável;
 - Os totens do metrô terão seus dados de funcionamento atualizados constantemente no banco de dados;
- **Tela de cadastro e login:**
 - Tela de cadastro e login conectado ao site estático. Utilização da plataforma Visual Studio Code para integração dos códigos e utilização de Java para desenvolvimento das telas de acesso do gerente do metrô;
- **Azure:**
 - Desenvolvimento do ambiente Azure para Banco de Dados na nuvem. A máquina deverá possuir o MVP necessário para sustentação das necessidades do usuário;
- **Dados da CPU:**

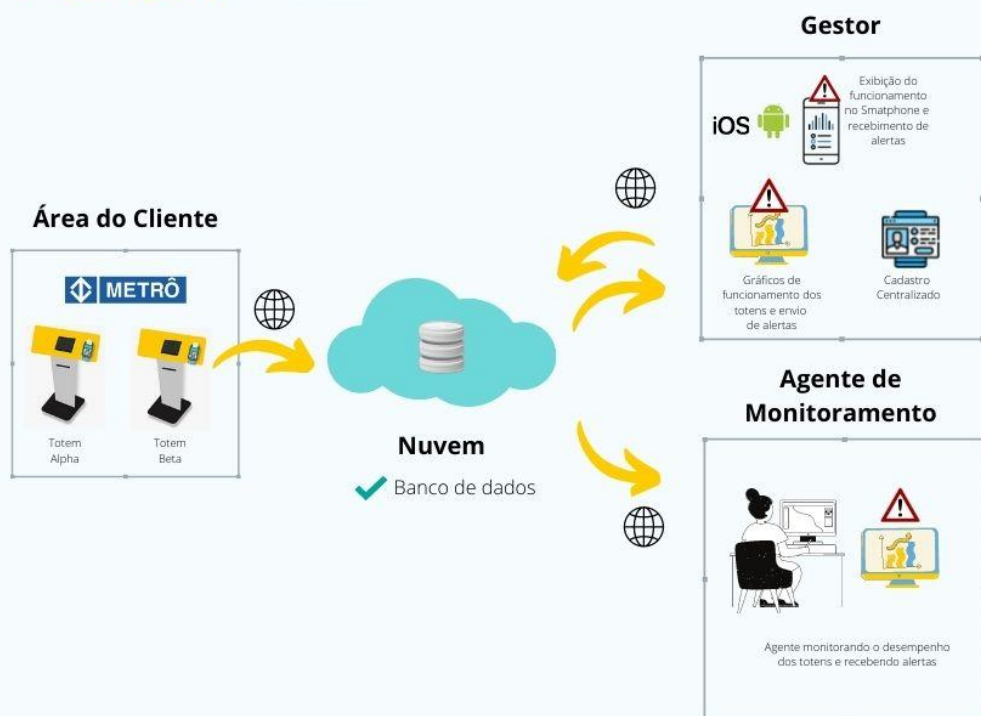
- Coleta de dados da CPU de totens pautada em PowerShell e cmdlets que farão a consulta do funcionamento de cada máquina;
- **API's:**
 - Utilização de API's de cadastro e login de usuário com base no Banco de Dados criado, bem como a regra de negócio imposta. A API será do tipo *rest* e terá o funcionamento ativado para o usuário poder navegar pelas funcionalidades do site;
- **Dashboard:**
 - O monitor terá acesso as dashboards depois de seu login, onde será atualizado em tempo real com os dados necessários para verificação dos totens de metrô. Os gráficos que serão apresentados precisam ser de fácil leitura e possuir indicadores para avisar em casos de anormalidade dos totens;
- **Slack:**
 - A partir da utilização da ferramenta Slack iremos interagir com o agente do metrô, enviando notificações sobre o uso dos totens, bem como totens mais utilizados, guia prático em caso de parada do funcionamento.
 - A ferramenta irá servir de princípio para a inovação, enviando notificação informando o usuário ganhador do sorteio, com dados sobre o produto sorteado e data prevista de entrega.
- **Inovação:**
 - Durante o monitoramento o sistema analisa quais totens não estão sendo utilizados a um certo período de tempo, se o aumento de tempo continuar o sistema irá notificar o usuário por meio de alertas sobre qual totem está em desuso. sendo que cada período de tempo possui um nível de alerta diferente baseado por cores. verde(30 min), amarelo(1h), laranja(2h) e vermelho(3h ou mais).

ARQUITETURA:



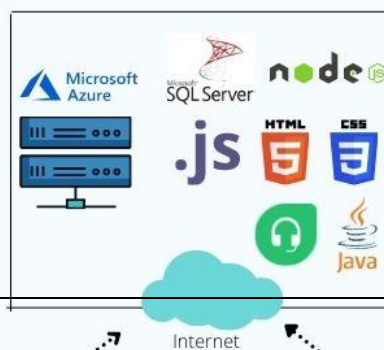
HLD TotemHub

Desenho de solução



LLD TotemHub

Desenho de solução



Requisitos:

Essenciais:

- Configuração do repositório Git;
- Abordagem Scrum do projeto;
- Utilização da ferramenta de planejamento;
- Definição do projeto;
- Ideação de inovação do projeto;
- Definição do Backlog;
- Definição da Regra de Negócio;
- Desenvolvimento do DER;
- Desenvolvimento do LeanUX;
- Definição UserStory e StoryBoard;
- Utilização de Java para cadastro e login;
- Versão 1 do site estático;
- Configuração do ambiente Azure;
- Criação do Plano de Resposta;
- Definição da Proto-Persona;

Importantes:

- Desenvolvimento do protótipo de telas;
- Desenvolvimento da documentação do projeto;
- Consulta constante com especialistas;
- Elaboração da apresentação;

- Visita para entendimento real do serviço a ser inovado;

Desejáveis:

- Padronização de todos os elementos utilizados no processo;
- Sistema de recompensa ao usuário de totem via QR Code;

Sustentação:

- Envio de guia de procedimentos para caso de parada de máquinas para o gerente do metrô;
- Abertura de chamado redirecionado para equipe de suporte da aplicação;
- Backup periódico dos dados do cliente em dispositivos móveis guardados com segurança.

Premissas:

- Cliente possui acesso à internet (50 megas);
- Cliente abrirá chamado em caso de dúvida ou parada da solução;
- Equipe de suporte atuará de prontidão para resolução de bugs;
- Totens receberão a manutenção adequada para suporte da solução.
- Monitor terá acesso aos totens somente depois de ter sido cadastrado no site pelo gerente;
- Monitor ficará responsável pelo funcionamento externo da solução (hardware dos totens);

Restrições:

- Cliente necessita da aplicação em Java e JavaScript;
- Banco de dados podem ser MySQL ou SQL Server;

- O ambiente necessita estar alocado no Azure;
- Plataforma de versionamento necessita ser Github;
- Ferramenta de planejamento necessita ser Planner;
- Prazo de entrega da Sprint 1 prevista para 08/09.

Marcos do projeto:

- Desenvolvimento da prototipação do site;
- Planejamento do projeto e abordagem da solução;
- Ideação do projeto;
- Levantamento de requisitos;
- Product Backlog;
- Criação e configuração do repositório Git do projeto;
- Criação do LeanUX, UserStories e StoryBoard;
- Criação da documentação do projeto;
- Criação da arquitetura de solução (HLD / LLD);
- Criação DER e Regra de Negócios da solução;
- Desenvolvimento de páginas do site;
- Desenvolvimento dashboard;
- Definição de falas da Sprint;
- Criação da apresentação.

Pautas para futuras reuniões:

- Homologação da primeira versão de protótipos e documentação do site; [26/08]
- Discussão sobre regra de negócio da solução; [26/08]
- Discussão para decisão de próximas atividades a serem feitas; [29/08]

- Criação v1 do DER / definição v1 Regra de Negócio; [31/08]
- Homologação do protótipo de telas; [01/09]
- Homologação v1 dashboard e arquitetura da solução (HLD / LLD); [02/09]
- Estruturação da apresentação; [04/09]
- Finalização dos entregáveis sprint 1; [05/09]
- Ensaio da apresentação 1; [05/09]
- Ensaio da apresentação 2; [06/09]
- Ensaio da apresentação 3; [07/09]
- Apresentação Sprint 1. [09/09]