**Fiap – Faculdade de Informática e Administração Paulista**

Daniel Sanchez Melero rm85109

Eric Luiz Campos Pessoa rm86405

Giovanna de Mello Leiva rm85817

Henrique Neves Lago rm84549

Kaue Augusto Miranda Santos rm85707

Larissa Alves da Silva rm86351

**Challenge Sprint 2**

**Arquitetura e Serviços a serem utilizados**

São Paulo – SP

Abril/2021

Arquitetura Escolhida

O projeto de imersão gastronômica Cardapp foi idealizado para o desafio da Plusoft com o tema de “Como melhorar a experiência do usuário utilizando IoT”. Pensamos em um aplicativo capaz de armazenar cardápios digitais de diversos estabelecimentos que sejam acessados através da leitura de um QR Code. Com o Cardapp o usuário poderá realizar atividades exclusivas, como sugerir uma música para tocar no local e alterar a iluminação da mesa, assim personalizando sua experiência.

Consideramos entre duas opções de WEB Services qual seria a melhor para esse projeto. O protocolo REST/API (Application Program Interface) acabou sendo a melhor opção em relação ao SOAP (Simple Object Access Protocol). Ambos têm seus prós, mas os contras do segundo nos fizeram escolher o primeiro.

O SOAP é um protocolo bem padronizado e restrito. Ele possui um contrato em arquivo xml, o WSDL (Web Services Description Language), que descreve os serviços que oferecidos, além de toda a regra para utilizá-los. A comunicação tanto de requisição quanto de resposta deve seguir a estrutura certa do contrato, envelopando os dados, contento um cabeçalho que vai descrever as regras da comunicação e no corpo a mensagem específica que se deseja mandar. É um protocolo bem rígido em relação às estruturas das mensagens, o que, apesar de ser bom por garantir uma maior segurança e tratamento de erros, é ruim por deixar amarrado de certa forma.

Já o protocolo REST é bem menos restritivo. Não existe um contrato de padronização da mensagem, como o WSDL. É possível apenas criar uma página da API que faz uma descrição do que está disponível e quais são os dados que podem ser enviados ou recebidos. Alterações podem ser feitas através das quatro operações básicas de persistência de dados – Create, Read, Update e Delete – utilizando os verbos HTTP **POST**, **GET**, **PUT** (atualiza o objeto todo), **PATCH** (atualiza apenas o que é editado) e **DELETE**. E mesmo não sendo um protocolo tão restritivo ainda garante a segurança dos dados.

Assim, escolhemos o protocolo REST por ser menos limitado, mais abrangente na forma que podemos manipular e por não termos que nos preocupar tanto com padronizações, podendo fazer a comunicação da forma que acharmos melhor.

Serviços REST que Serão Utilizados

O Cardapp contará com uma aplicação WEB para que os estabelecimentos consigam gerenciar as informações sobre eles e os cardápios, além de poderem visualizar as avaliações feitas no aplicativo mobile pelos clientes. Criaremos os serviços abaixo para realizar essas ações:

**- allRating** – utilizando o verbo **GET** de recuperação de objetos, serão recuperadas todas as avaliações realizadas pelos clientes após a finalização dos pedidos.

**- createItem(ItemCardápio ic)** – utilizando o verbo **POST** de criação de objetos, será criado um item no cardápio do estabelecimento após o gerente preencher um formulário com os dados necessários.

**- editItem(int codigo)** – utilizando o verbo **PUT** de atualização de objetos, um item do cardápio será recuperado para ser editado pelo gerente.

**- editEstab(int codigo)** – utilizando o verbo **PUT** de atualização de objetos, as informações do estabelecimento serão recuperadas e poderão ser editadas pelo gerente.

**- editManager(int codigo)** – utilizando o verbo **PUT** de atualização de objetos, as informações do gerente serão recuperadas e poderão ser editadas pelo administrador do sistema.

**- deleteItem(int codigo)** – utilizando o verbo **DELETE** de exclusão de objetos, será possível deletar um item do cardápio do estabelecimento.

**- deleteEstab(int codigo)** – utilizando o verbo **DELETE** de exclusão de objetos, será possível deletar um estabelecimento do sistema.