

**UNIVERSIDADE FEDERAL DE GOIÁS**  
**Instituto de Informática**  
Engenharia de Software

Fernando Henrique Callata  
Gustavo Batista  
Lucas Sampaio  
Pedro Basílio  
Saulo Calixto

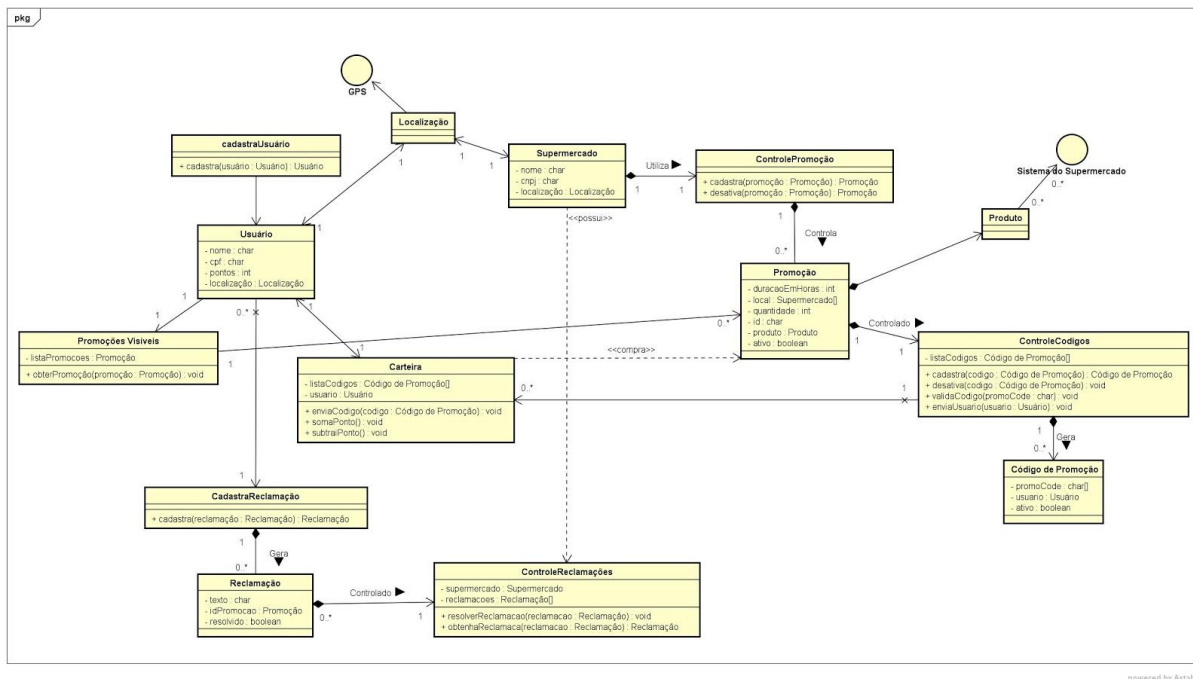
**PROJETO**

Goiânia, 20 de outubro de 2017.

## Controle do Documento

Versão	Data	Autor/Revisor	Descrição
0.1	14/10/2017	Saulo Calixto / Gustavo Batista / Fernando Henrique / Lucas Sampaio / Pedro Basílio	Versão inicial

## 1. Diagrama de Classes:



### 1.1 Descrição de Diagrama de Classes:

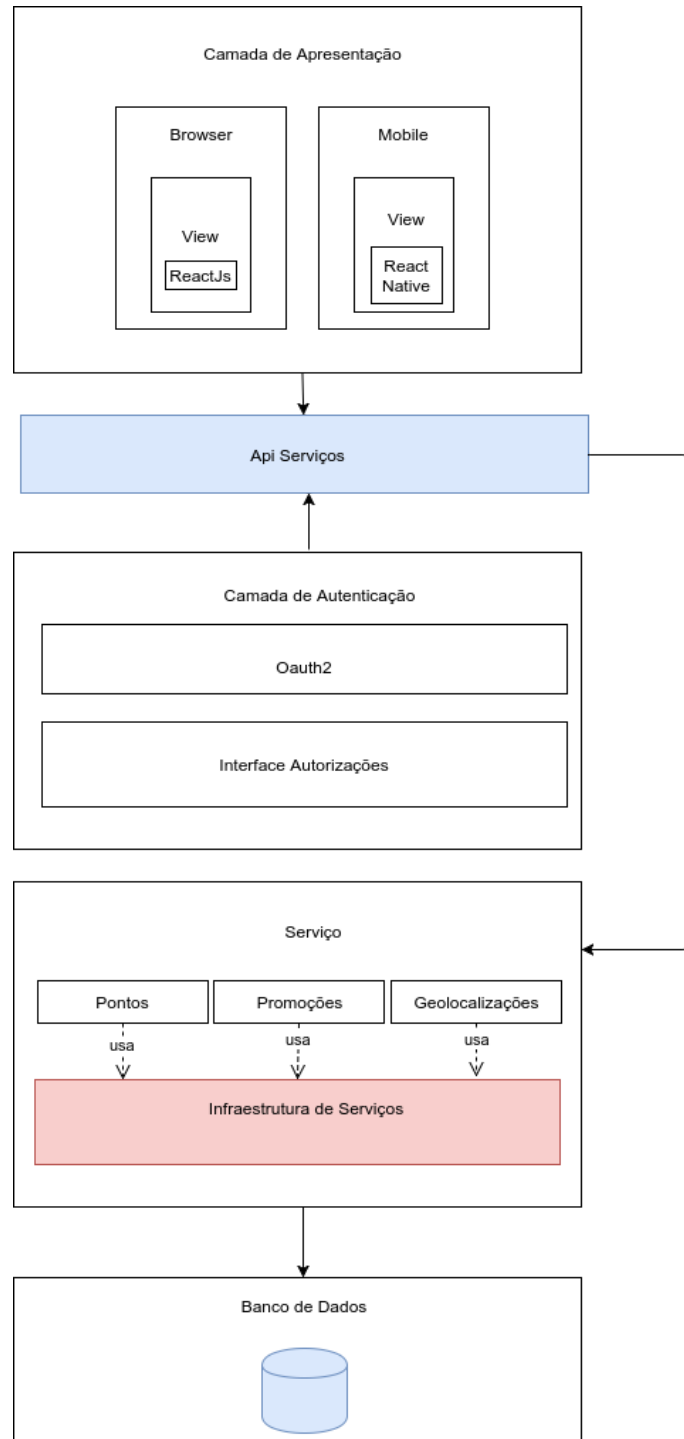
Foi definido um diagrama de classes de forma que se possa atender à arquitetura definida e os casos de uso definidos anteriormente. São representados as interfaces relevantes à solução do problema, como foi definido nos requisitos a necessidade de integração com o sistema de supermercado e a geolocalização dos usuários e promoções.

No modelo também está representado a solução proposta para a gerência dos códigos promocionais e pontos, conforme detalhada a necessidade de códigos únicos para cada usuário, recompensa pela utilização de códigos, entre outros definidos nos documentos de requisitos.

A solução escolhida foi considerada como a melhor presente para atender às necessidades de persistência das promoções e para respostas em tempo real e personalizadas ao usuário, assim como uma solução que apresente graus suficientes de coesão e acoplamento.

Para maiores detalhes sobre as classes, métodos e atributos definidos, visitar o [repositório do projeto](#).

## 2. Diagrama de Arquitetura:



### 2.1 Descrição de Diagrama de Arquitetura:

Utilizamos uma arquitetura dividida em camadas, apresentamos 5 camadas, a de apresentação, a de autenticação, a de serviço, banco de dados e a de api que

é a camada intermediária entre a camada de apresentação e serviço. Nesse modelo é identificado os componentes e suas dependências e os colocamos em camadas que possuem relação com aquele componente. É uma boa arquitetura porque nós podemos dividir cada parte da aplicação, facilitando a manutenibilidade.

A primeira camada, que seria a *Camada Apresentação* será visível para o usuário, é responsável pela interação entre os usuários e a aplicação. Temos assim duas apresentações possíveis, via browser e via smartphone. Definimos que usaremos o framework ReactJs para desenvolvimento das views.

Mais abaixo, a segunda camada é destinada à autenticação que irá definir o que será apresentado ao usuário, é a camada intermediária que possibilita que o usuário comum tenha acesso aos serviços apresentados pelo sistema. Dessa forma tem uma autenticação que irá disponibilizar certos serviços de acordo com o perfil de usuário.

A camada de API é o elo entre o serviço e a view, ao se autenticar o sistema de acordo com suas permissões irá disponibilizar os serviços que você tem acesso através da API.

A Camada de Serviço apresenta os serviços que o sistema irá oferecer. Pontuação e Promoções de acordo com o perfil de usuário terá funções diferentes. Por exemplo o supermercado poderá cadastrar promoções, já o cliente usufruirá delas. Há também a infraestrutura que terá ferramentas necessárias para realização dos serviços.

Por fim temos a camada de banco de dados responsável pela persistência dos dados que serão usados pelo serviço e apresentados para os clientes na camada de apresentação.