FeedEm food project.

FeedEm project is used for preorder your food in popular restaurants.

User roles

- Customer
- Seller
- Administrator

Usecases

BG-1: Reduce customers waiting time.

F1-1: Order will be ready when you get to the restaurant.

As customer i want to get my food as fast as possible.

This app allows you to preorder food and get it in desired time.

F1-2: No more waiting in queues.

FeedEm service has *no* waiting queues.

All you need is money and internet connection.

BG-2: Easiest way to process orders for customers.

F2-1: Don't need to pay for delivery service.

FeedEm made not for delivery, but for preorder.

Sellers goal is to make food in customer desired time. Customers will come to get it.

F2-2: Don't need to create your own web-app.

BurgerKing™ has its own app.

And it's cool, to order food, without queues. But writing your own app is very expencive. And we got solusion for you.

F2-3: Free advertisement.

If your food is good enough, new customers will found you.

More likes you get, more people will know about your food.

BG-3: Easily control FeedEm service.

F3-1: Get the summary of the system.

As an administrator you want to know what's going on in your system.

You can view various data in the administrator's panel or get the summary of it.

F3-2: View information about orders.

As an administrator you want to view all orders in the system and their status.

You can monitor them in the administrator's panel.

F3-3: Add or remove sellers.

As administrator you want to add or remove sellers (add new ones; delete ones that working poorly).

You can add or delete them in the administrator's panel.

Components

Models (M in MVC model)

See the ER diagram.

Implemented as Ruby on Rails models.

Data stored in PostgreSQL database.

- Customers
- Credit cards
- Sellers
- Merchandise
- Orders

Views (V in MVC model)

Implemented as Ruby on Rails views.

Uses vue js framework on client side.

- Front page
- Search page
- Merchandise page
- Order page
- User profile
- User login page
- User registration page
- Seller profile
- Seller login page
- Administrator panel

Controllers (C in MVC model)

Implemented as Ruby on Rails controllers.

- User controller
- Seller controller
- Order controller
- Merchandise controller
- Credit card controller