User Stories

Customer User Stories

1. As a customer, I want to be able to browse the cafe's menu, see item details, and add items to my cart. 🤯 🍔 🛒
2. As a customer, I want to be able to place an order, track it, and earn loyalty rewards. 🤑 🔢 👀

Staff User Stories

1. As a staff member, I want to be able to manage orders, view sales analytics, and update the menu. 💼 📊 📝

Manager User Stories

1. As a manager, I want to be able to view detailed sales reports, manage staff permissions, and receive email notifications for new orders. 📊 💼 📧
2. As a manager, I want to be able to analyze sales data and understand customer behavior to make informed decisions. 📈 🤯 📊

Product Backlog

1. Project Setup, Basic Models, and Authentication
   - Create a detailed project plan outlining the tasks that need to be completed in each section. 📝🧑‍💻📒
   - Set up a Trello board or another task management tool to track progress and assign tasks. 📌👀🗓
   - Choose a suitable template for the project and create the necessary template structure. 🎨🖌📐
   - Write the necessary template pages for the cafe website, including the home page, menu page, cart page, checkout page, and staff panel page. 📄🍔🛒
   - Create a new Django project and app. 🆕🧑‍💻🐍
   - Configure the app to use PostgreSQL as the database. 🗄🔧🐘
   - Initialize a Git repository and commit your initial project files. 📦📝🧑‍💻
   - Create an ERD (Entity Relationship Diagram) for your data model. 🗃🤔📊
   - Create the necessary models for cafe items and categories. 🍔🍦🍩
   - Create models for staff, customer orders, and order items. 📋📦📝
   - Define relationships between the models. 🔗🔍📊
   - Implement Django Form Objects, Django Sessions, and Django Users and Authentication. 🔑👥🧑‍💻
   - Create a custom backend authentication for Django using phone numbers. 🔢🔑📞

2. Online Menu and Staff Panel
   - Implement the online menu with categories and items. 🍔🍦🍩
   - Allow customers to view items, search, and see details. 🔍👀🗺
   - Handle cart functionality using cookies, and track orders using sessions. 🛒🍪📝
   - Implement Django Class Views and Django Test. 🐍🧑‍💻🧪
   - Create a staff panel with authorization, allowing staff to see and manage orders. 💼👀📊
   - Implement manager-specific features using Django permissions in views and templates. 🧑‍💼🔒👀

3. Analytics Feature
   - Add analytics tracking for customer behavior, such as most popular items, peak business hours, and customer demographic data. 🗺🧐📊
   - Implement an analytics dashboard for the manager view that displays key sales metrics and trends. 📊🧑‍💼🗺

- Enable the manager to generate and download detailed sales reports. 📊 👨‍💼 🖊️
4. Deployment and Finishing Touches
  - Ensure the website is live, functional, and accessible to customers. ✳️ 💻 👀
  - Implement Django Generic Views. 🐍 👨‍💻 📄
  - Add statistics and insights for the manager. 📊 👨‍💼 🔍
  - Improve the user interface and user experience. 💻 🤓 ✋
  - Implement frontend frameworks and libraries. 🎨 💻 🗂️
  - Add animations, transitions, and interactive elements.

Sprint Planning

Sprint 1 (Week 1)

- Authentication and Basic Models
  - Create a Django project and app using a template 🐍 💻
  - Configure the app to use PostgreSQL as the database 🐘 🗄️
  - Create an ERD (Entity Relationship Diagram) for the data model 📊 📈
  - Create the necessary models for cafe items and categories 🍔 🥤 🍩
  - Create models for staff, customer orders, and order items 💼 🛒 👨‍🍳
  - Define relationships between the models 🤝 💕
  - Implement Django Form Objects, Django Sessions, and Django Users and Authentication 🔐 👥
  - Create a custom backend authentication for Django using phone numbers 🔢 📞
  - Implement basic frontend design using HTML, CSS, and Bootstrap 🎨 💻

Sprint 2 (Week 2)

- Online Menu and Staff Panel
  - Implement the online menu with categories and items 🍔 🥤 🍩
  - Allow customers to view items, search, and see details 🧐 🔍 👀
  - Handle cart functionality using cookies, and track orders using sessions 🛒 🍪
  - Implement Django Class Views and Django Test 🐍 🧽
  - Create a staff panel with authorization, allowing staff to see and manage orders 💼 👨‍🍳 🖊️
  - Implement manager-specific features using Django permissions in views and templates 👨‍💼 👀 🖊️

o Write unit tests for the online menu and staff panel that cover more than 95% of the code 🧪 ☑

Sprint 3 (Week 3)

- Final Features and Frontend Magic
    o Implement Django Generic Views 🐍 🔍
    o Add statistics and insights for the manager 📊 📈 👨‍💼
    o Improve the user interface and user experience 🎨 🤩 👍
    o Polish the frontend using HTML, CSS, and Bootstrap 🎨 🖌 💻
    o Implement JavaScript and DOM manipulation 🕹 🎮
    o Optimize the website for different devices and screen sizes 📱 💻 🖥
    o Write unit tests for final features that cover more than 95% of the code 🧪 ☑

🚀 **Tasks with High Priority:**

To create the best MVP for the cafe project, we need to prioritize our tasks based on their importance. Here are the high priority tasks that we need to complete first:

- 🖥 Create a Django project and app using a template
- 🗄 Configure the app to use PostgreSQL as the database
- 🗂 Create an ERD (Entity Relationship Diagram) for the data model
- 🍔 Create the necessary models for cafe items and categories
- 📋 Create models for staff, customer orders, and order items
- 🔗 Define relationships between the models
- 🔑 Implement Django Form Objects, Django Sessions, and Django Users and Authentication
- ⌨ Create a custom backend authentication for Django using phone numbers
- ✋ Implement the online menu with categories and items
- 🔍 Allow customers to view items, search, and see details
- 🛒 Handle cart functionality using cookies, and track orders using sessions
- 💼 Create a staff panel with authorization, allowing staff to see and manage orders
- 👨‍💼 Implement manager-specific features using Django permissions in views and templates
- 🐍 Implement Django Generic Views
- 🧪 Write unit tests for final features that cover more than 95% of the code

We can use a table to visualize the tasks and their status:

Task	Priority	Status
Create a Django project and app using a template	High	In Progress
Configure the app to use PostgreSQL as the database	High	Not Started
Create an ERD (Entity Relationship Diagram) for the data model	High	Not Started
Create the necessary models for cafe items and categories	High	Not Started
Create models for staff, customer orders, and order items	High	Not Started
Define relationships between the models	High	Not Started
Implement Django Form Objects, Django Sessions, and Django Users and Authentication	High	Not Started
Create a custom backend authentication for Django using phone numbers	High	Not Started
Implement the online menu with categories and items	High	Not Started
Allow customers to view items, search, and see details	High	Not Started
Handle cart functionality using cookies, and track orders using sessions	High	Not Started
Create a staff panel with authorization, allowing staff to see and manage orders	High	Not Started
Implement manager-specific features using Django permissions in views and templates	High	Not Started
Implement Django Generic Views	High	Not Started
Write unit tests for final features that cover more than 95% of the code	High	Not Started

Let's get started on these tasks and create an amazing cafe project! 🍩 🍔 👨‍💻