

# HU Edge – Python Fast API

## Training Materials

Please go through the following courses.

FastAPI: <https://deloittedevelopment.udemy.com/course/fastapi-course-python/>

Official doc: <https://fastapi.tiangolo.com/learn/>

## Backend system for Pizza Delivery

### Project Description:

Develop a backend system for Pizza Delivery for a standalone store using Python FastAPI.

### Must have features:

- The system should have user authentication and authorization for APIs based on roles.

The following are the descriptions of the roles of users.

#### Customer:

- Should be able to get the list of Pizzas.
- Add to cart.
- Checkout from cart.
- Access previous orders.

#### Delivery Partner:

- Can update the status of the delivery.
- Can add comment to the delivery

#### Admin:

- Can add/update/delete Pizzas.

### Good to have:

- Customer ratings for individual Pizzas.
- Delivery partner rating.
- Different modes of payment.
- Filter pizzas based on type.
- Update user details.

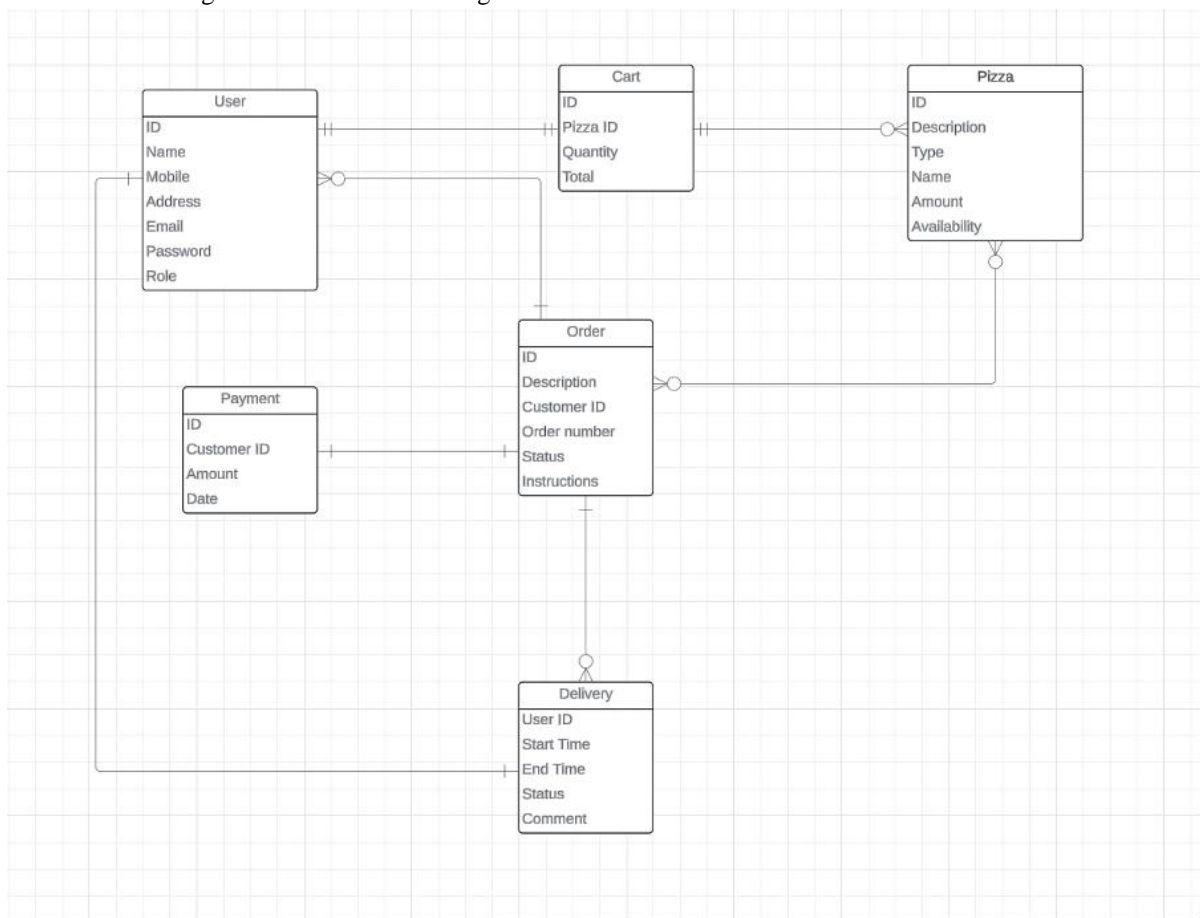
## Standards:

The following standards will earn extra points :)

- Good usage of branches and commits to segregate different features.
- Unit tests coverage should be >65%.
- Comments to enhance the readability of the code.

## Sample DB Design:

- This is a reference DB design, there are some intentional discrepancies with some fields. But this can be a good start for the DB design.



## Milestones:

1. Initial Setup for Project and configure DB
  - a. Complete the dev-env setup with the relevant framework and packages
  - b. Configure DB with required tables and relationships
  - c. List the APIs required and request and response models for them.
2. Create placeholder APIs
  - a. Create all APIs definitions with their respective Request and response Models  
Logic for individual APIs can be part of the next Milestones.
  - b. Group API routes based on the feature/role
3. User authentication and authorization
  - a. Implement APIs for user authentication and session management.
  - b. Implement authorization or role-based access to hit APIs
4. Admin APIs to populate Pizzas
  - a. Develop APIs that can be used by admin users.
  - b. They should be able to add new Pizza options, update the details and availability and delete them as well.
  - c. Create API to update status of the order.
5. Customer APIs
  - a. Create APIs to Add and remove Pizzas to the cart.
  - b. Create API to clear cart.
  - c. Create API to get the cart details and the cart value.
  - d. Create API to check out the cart.  
As part of the API - Create a new order, assign a delivery partner, update payment details.  
(You can simulate as if you are reaching out to a payment gateway if there is an online payment method)
  - e. Create API to view previous order details.
6. Delivery Partner APIs
  - a. Create an API to update the status of the delivery.
  - b. When closing the delivery, the delivery partner can add comments to the delivery.