

Problem Statement

Product Dissection for top leading Platforms

Welcome to this case study on dissecting and designing products for top leading platforms. In this case study, you will delve into the intriguing world of schema design for a prominent platform of your choice. Your task is to choose a top leading platform, research its features, and meticulously craft a schema design that encapsulates the essence of its functionality. By focusing on key entities, attributes, and relationships, you will gain invaluable insights into how data architecture drives the platform's effectiveness.

Step 1: Choose a Leading Platform

Select a leading platform of your choice, which could span various domains such as social media, e-commerce, finance, or any other industry. This choice will form the foundation of your exploration into its schema design.

Step 2: Research:

Thoroughly research the platform you have selected. Investigate its core features, functionalities, and user interactions. Identify the top features that define its user experience and contribute significantly to its popularity.

Step 3: Product Dissection and Real World Problems solved by the platform

In this step, you will meticulously analyse the platform's standout features and how they provide innovative solutions to real-world challenges. By identifying key functionalities that resonate with users, you'll unravel how the platform effectively addresses problems and enhances user experiences. This dissection will serve as the foundation for understanding how the schema design aligns with the platform's core objectives.

Step 4: Case Study on the real world problems and approach to solving them

In this pivotal step, you will expand on the real-world challenges uncovered in Step 3 through a comprehensive case study. Delve into specific instances where users encountered difficulties and showcase how the platform's unique features provided effective solutions. By dissecting the approach taken by the platform to overcome these challenges, you'll gain a deeper appreciation for the platform's user-centric design philosophy and how it shapes the schema design.

Step 5: Schema Design Based on Top Features

Based on the features you have identified, craft a schema design that reflects the platform's data structure. Focus on the key entities, attributes, and relationships that underpin the chosen features. Your schema should capture the essence of how the platform organises and utilises its data.

Step 6: Rationale Behind the Design

While creating the schema design, consider the rationale behind the platform's choices. Reflect on why certain entities and relationships were chosen and how they align with the platform's goals. This will help you understand the strategic decisions driving the schema's architecture.

Step 7: Create an ER Diagram

Utilise tools like the Miro platform or similar applications to create an illustrative Entity-Relationship (ER) diagram. This diagram should vividly depict the entities, attributes, and relationships present within your schema design. The ER diagram will serve as a visual representation of your insights.

Step 8: Presentation of Findings

Present your findings in a clear and concise manner. Showcase your understanding of how the schema design impacts the platform's functionality and user experience. Explain how your chosen features are integrated into the schema and how the schema's structure supports the platform's objectives.

Task Details:

1. **Answer Submission:** Your submission should include well-structured solutions for all provided questions related to product schema designs.
2. **Video Creation:** Create an informative and engaging video where you thoroughly explain the Case Study.
3. **Depth and Clarity:** Ensure your solutions are detailed and showcase your understanding of product schema design principles. Similarly, in the video, provide clear explanations that are easy to understand for a wide audience.
4. **Creativity Encouraged:** You are welcome to utilise visuals, diagrams, or creative elements to enhance the clarity and impact of your explanations.

Note:

1. Duplicate this document and proceed to write your solutions and prepare your video.
2. Include the video link in this document before final submission.

Best of luck in completing this project and showcasing your prowess in dissecting and designing product schema for leading platforms! **For reference, we have also conducted a case study on Instagram, which you can find below. This case study will provide you with valuable insights into how schema design plays a pivotal role in shaping the functionality and success of a prominent platform.**

The Zomato logo is displayed in a bold, italicized, red font. The letters are lowercase, and the 'o' is stylized with a dot. The logo is centered at the bottom of the page.

Product Dissection for Zomato

Company Overview:

Zomato was founded by Deepinder Goyal and Pankaj Chaddah in the year 2008 providing food at your doorstep with just a click. Zomato has a user friendly interface, providing fast delivery service from multiple restaurants, it has emerged as a top leading platform in food delivery apps.

Product Dissection and Real-World Problems Solved by Zomato:

Zomato, the popular food delivery app, addresses numerous real-world problems with its comprehensive platform. It revolutionizes the dining experience by offering unparalleled convenience, allowing users to effortlessly browse a wide range of restaurants, view menus, and place orders from the comfort of their homes. By providing access to an extensive array of cuisines and dishes, Zomato tackles the issue of limited options present in traditional dining scenarios, especially in areas with fewer restaurant choices.

Moreover, its integration of user-generated reviews and ratings empowers users to make informed decisions about where to order from, mitigating the uncertainty associated with trying new eateries. The app's customization options cater to individual preferences and dietary restrictions, solving the problem of limited flexibility in ordering food.

Additionally, Zomato's frequent discounts and promotions make dining out or ordering in more affordable, while its real-time delivery tracking feature provides users with peace of mind regarding the status of their orders. Furthermore, Zomato serves as a platform for culinary discovery and exploration, enabling users to uncover new restaurants and cuisines based on their preferences and location.

Overall, Zomato streamlines the food ordering and dining experience, offering convenience, variety, transparency, and affordability, thereby solving several real-world problems associated with traditional food delivery and dining services.

Case Study: Real-World Problems and Zomato's Innovative Solutions

Zomato, a popular food delivery app, solves several real-world problems related to food delivery, dining, and discovering new restaurants. Let's dissect the product and explore some of the problems it addresses:

Problem 1: Food Ordering Experience

Real-World Challenge: Traditional methods of ordering food often lack the convenience and efficiency desired by modern consumers. Phone calls can be time-consuming and prone to errors, leading to a disconnect between customers and the food delivery process.

Zomato's Solution:

Zomato revolutionizes the food ordering experience by providing a convenient and user-friendly platform for browsing menus, placing orders, and tracking deliveries in real-time. By streamlining the process through a mobile app, Zomato bridges the gap between customers and restaurants, enhancing communication and reducing the disconnect often experienced in traditional food ordering methods.

Problem 2: Limited Restaurant Options and Information

Real-World Challenge: Customers often face the dilemma of choosing from a limited selection of restaurants in their area, with little information available about each option's menu, pricing, and reviews.

Zomato's Solution:

Zomato addresses this challenge by offering a comprehensive database of restaurants with detailed menus, pricing information, user reviews, and ratings. This information empowers customers to make informed decisions about where to order from expanding their choices and ensuring a satisfying dining experience.

Problem 3: Food Waste Reduction

Real-World Challenge: Restaurants often face the issue of food wastage due to overproduction or unsold food stock

Zomato's Solution:

Zomato introduced the "Feeding India" initiative a program aimed at reducing food wastage by partnering with restaurants to donate food to those in need. Restaurants can use the zomato app to identify excess food stock, which is then collected and distributed to local

charities and shelters. This initiative not only helps to minimize food wastage but also supports people in need.

Problem 4: Lack of personalization and customization options

Real-World Challenge: Users who have some restrictions in diet due to health issues, some users who are either vegan or eat only gluten free, they may struggle in finding restaurants as per their choices.

Zomato's Solution:

Zomato has search and filtering options that allow users to find restaurants where they can simply use a filter for vegan, gluten free, spicy, kids-choice or any other specifications as per their need as they have tied up with several restaurants. Users can also customize what they have ordered. This ensures that each and every user will have rich experience in ordering food no matter what preference they have.

Conclusion:

Zomato, a leading food delivery and restaurant discovery platform, has successfully transformed the dining and food delivery experience through innovative technology and extensive market reach. Despite facing intense competition and regulatory challenges, it has maintained a strong user base and expanded its services globally. Zomato's future growth will depend on its ability to sustain customer loyalty, enhance operational efficiency, and adapt to changing market dynamics.

Top Features of Zomato:

- 1. Restaurant Discovery:** In zomato users can explore a wide range of restaurants allowing them to discover new dining options based on cuisine, location, ratings, and reviews.
- 2. Menu Information:** Users can explore detailed menus of restaurants listed on zomato, including prices, photos, descriptions, help users choose dishes according to their preferences.
- 3. Reviews and Ratings:** Zomato provide user-generated reviews and ratings for restaurants helping users in making informed decisions about where to eat, order food based on their experiences.
- 4. Online Ordering:** Zomato allows users to place orders directly through the app Seamless ordering process for food delivery from a vast network of restaurants.
- 5. Food Tracking:** Users can track real-time of food orders from the restaurant to their location.
- 6. Deals and Discounts:** Zomato frequently notifies about special offers, discounts, and deals available at restaurants.

Schema Description:

The schema for Zomato involves multiple entities that represent different aspects of the platform. These entities include Users, Orders, Restaurants, Payment, Review, Delivery Partner, and more. Each entity has specific attributes that describe its properties and relationships with other entities.

- **User Entity:**

Users are at the core of Zomato. The user entity contains information about each user:

- **(Primary Key):** A unique identifier for each user.
- **Full_Name:** The user's full name as displayed on their profile.
- **Email:** The user's email address for account-related communication.
- **Phone_Number:** This is the phone number of the user
- **Joining_Date:** The date when the user joined Zomato
- **Membership:** Does the user have a membership or not
- **Rating:** Rating of user

Restaurants Entity:

Restaurants on Zomato are places where users can order food. Each restaurant listing provides information about that particular establishment:

- **RestaurantId(Primary Key):** A unique identifier for each restaurant
- **Name:** Name of the restaurant
- **Address:** Address of the restaurant
- **Phone_Number:** Contact number of restaurant

Partners Entity:

Delivery partners are the individuals responsible for delivering orders from restaurants to users:

- **PartnerID (Primary Key):** A unique identifier for each delivery partner.
- **Partner_Name:** Name of the Delivery partner
- **Phone_Number:** contact number of delivery partner.
- **Email:** Email ID of the delivery partner.
- **Partner_Rating:** Rating of the delivery partner (1 to 5)
- **Delivery_Area:** The geographic area or region covered by the delivery partner for order deliveries.
- **Licence_no:** Driving licence number of the delivery partner.

Order Entity:

Order placed by users:

- OrderID (Primary Key): A unique identifier for each Order.
- UserID (Foreign Key referencing User Entity): User who made the order.
- RestaurantID (Foreign Key referencing Restaurant Entity): Identifies the restaurant from which the order was placed.
- Order_Date: Date and time at which the order was placed
- Order_Status: Status of the Order (Delivered or Cancelled)
- PartnerID (Foreign Key referencing Partner Entity): Identifies the delivery partner responsible for delivering the order.
- Delivery_Date : The date and time when the order was delivered.
- AddressID (Foreign Key referencing Address Entity): Identifies the address from which the order should be delivered.
- Total_amount: The total amount of the order, including taxes and delivery charges.
- Coupon_code: This attribute stores any coupon codes applied to the order, providing discounts or special offers
- Discount_amount: The amount after the discount by using a coupon.
- ReviewID (Foreign key referencing Review Entity): Links the order to any review left by the user for the restaurant after the order is completed.

Payment Entity:

Payments represent the financial transactions associated with orders on Zomato:

- PaymentID (Primary Key): A unique identifier for each payment done.
- OrderID (Foreign Key referencing Order Entity): The user who is being followed.
- Payment_Mode: Mode of Payment chosen by the user.
- Payment_Status: Status of the payment such as pending, processed or completed.
- Payment_Date: The date and time when the payment was made.
- Amount: The amount of payment made.

Menu Entity:

Menu available at the restaurant:

- ItemID (Primary Key): A unique identifier for each menu item.
- RestaurantID (Foreign Key): Links the menu item to the restaurant it belongs to.
- Item_Name: The name of the menu item.
- Description: A brief description of the menu item, including ingredients or special features.
- Price: The price of the menu item.
- Category: The category or type of the menu item, such as appetiser, main course, dessert, etc.
- Availability: Indicates whether the menu item is currently available for order.

Address Entity:

Address of the users from which order is made:

- AddressID (Primary Key): A unique identifier for each address.

- UserID (Foreign Key referencing user entity): Links the address to the user who placed the order.
- Address: The complete address, including street address, city, state, postal code, and country.

Review Entity:

Reviews represent the feedback provided by users about their dining experience at a restaurant:

- ReviewID (Primary Key): A unique identifier for each review made by the user.
- UserID (Foreign Key referencing Order Entity): user who made the review.
- RestaurantID (Foreign Key referencing Restaurant Entity): identifies the restaurant being reviewed.
- Rating: Rating given by the user to restaurants a scale of (1 to 5)
- Comments: Any additional comments or feedback provided by the user about their experience.
- Review_Date: The Date at which Review is being made.

Relationships are:

Here are the relationships between the tables in the Zomato database design:

- ☐ Users — Orders: A user can place multiple orders, and each order is placed by a single user. This is a one-to-many relationship.
- ☐ Restaurants — Orders: A restaurant can have multiple orders placed for it, and each order is placed at a single restaurant. This is a one-to-many relationship.
- ☐ .Orders — Payment: Each order will have a single payment associated with it, and each payment is for a single order. This is a one-to-one relationship.

- ☐ Orders — Partners: Each order will have a single delivery partner associated with it, and each delivery partner is assigned to multiple orders. This is a one-to-many relationship.
- ☐ Users — Rating: A user can rate multiple restaurants, and each restaurant can be rated by multiple users. This is a many-to-many relationship.
- ☐ Restaurants — Rating: A restaurant can be rated multiple times by different users, and each rating belongs to a single restaurant. This is a one-to-many relationship.
- ☐ Users — Address: A user can have multiple delivery addresses, and each address belongs to a single user. This is a one-to-many relationship.
- ☐ Restaurants — Menu: A restaurant can have multiple menu items, and each menu item belongs to a single restaurant. This is a one-to-many relationship.

ER Diagram:

Let's construct an ER diagram that vividly portrays the relationships and attributes of the entities within the Zomato schema. This ER diagram will serve as a visual representation, shedding light on the pivotal components of Zomato's data model. By employing this diagram, you'll gain a clearer grasp of the intricate interactions and connections that define the platform's dynamics.

