

# Zarooriyaat Platform

UNLOCKING E-COMMERCE FOR  
EVERYONE



# Problem Statement

**01**

## **Problem 01**

Barrier to Entry for New  
Buisnesses

**02**

## **Problem 02**

Increased Financial Pressure

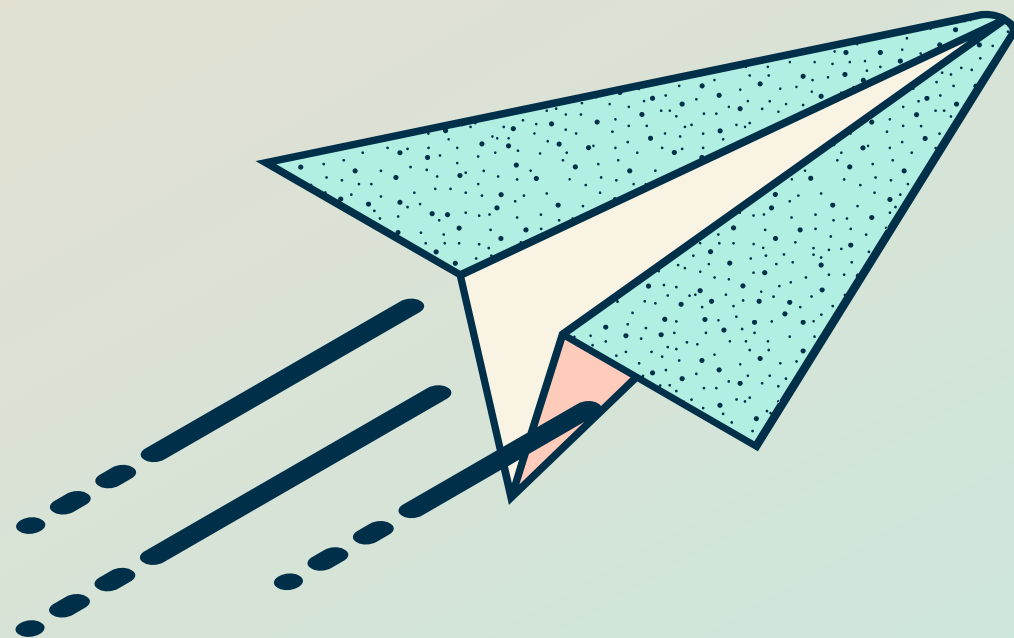
**03**

## **Problem 03**

Reduced Flexibility for  
Business Operations



# Zarooriyaat Platform



**01**

## **Solution 01**

No Upfront Costs

---

**02**

## **Solution 02**

Performance-Based Fees

---

**03**

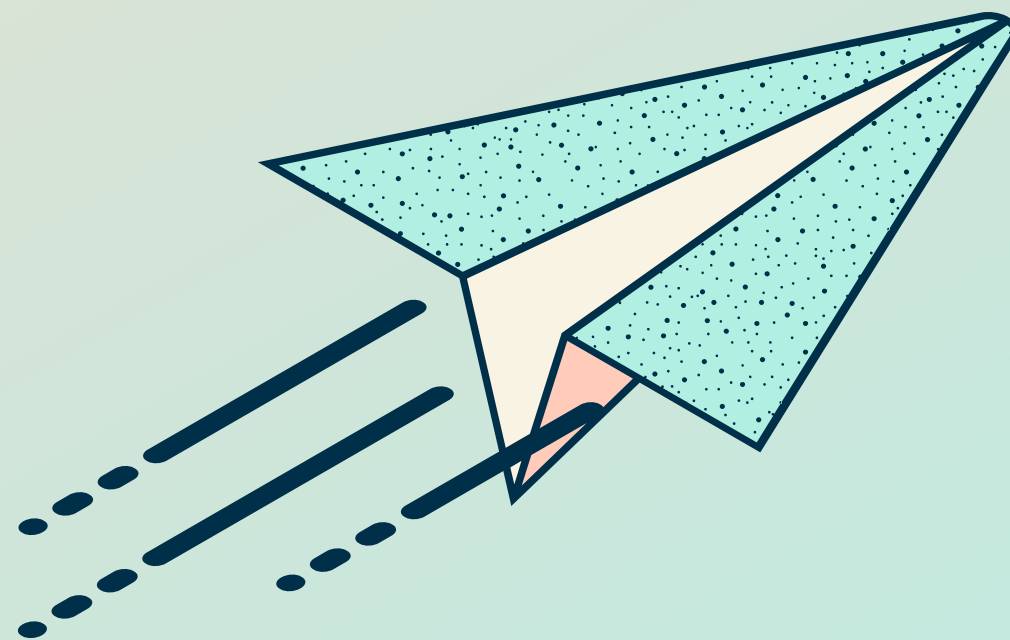
## **Solution 03**

Scalability Support



# System Architecture

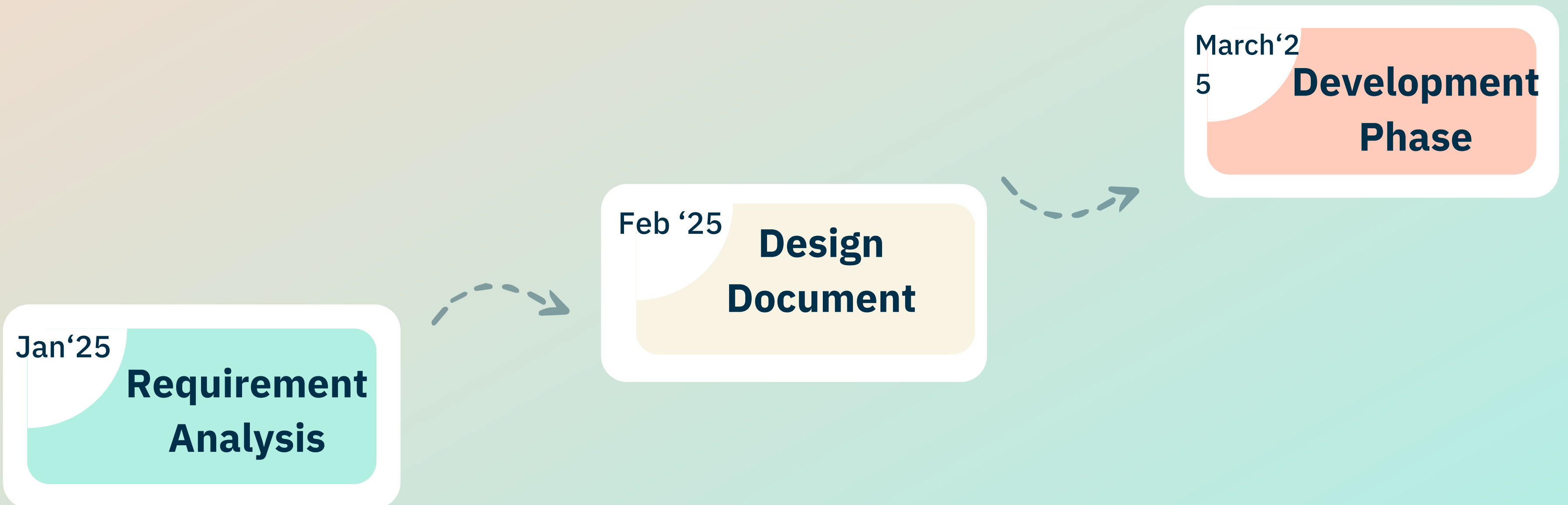
---





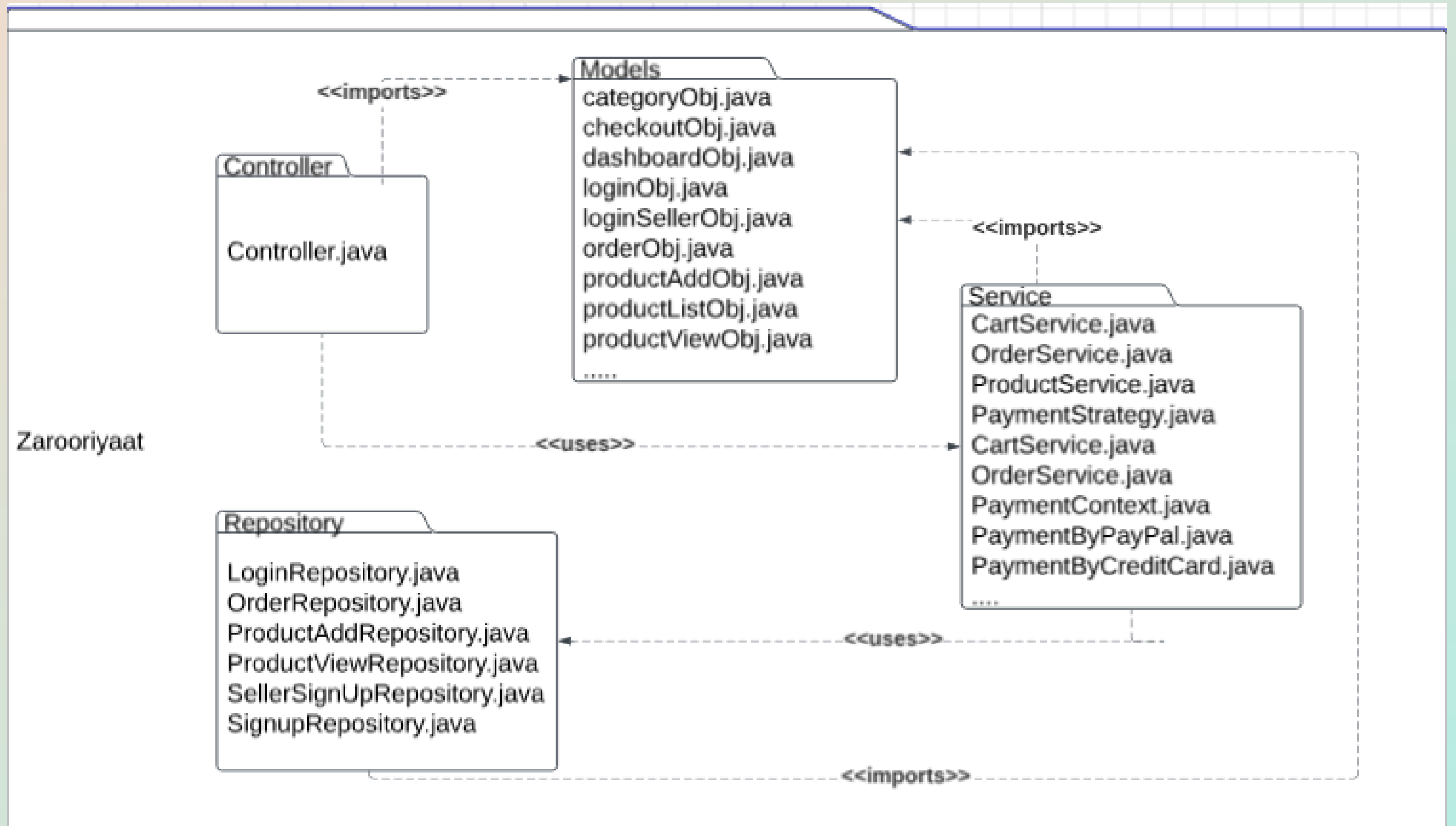
# Zarooriyaat Development

Using Scrum (Agile), we iteratively progressed through requirement analysis, design, and development, enabling adaptability and continuous improvement.



# Package Diagram

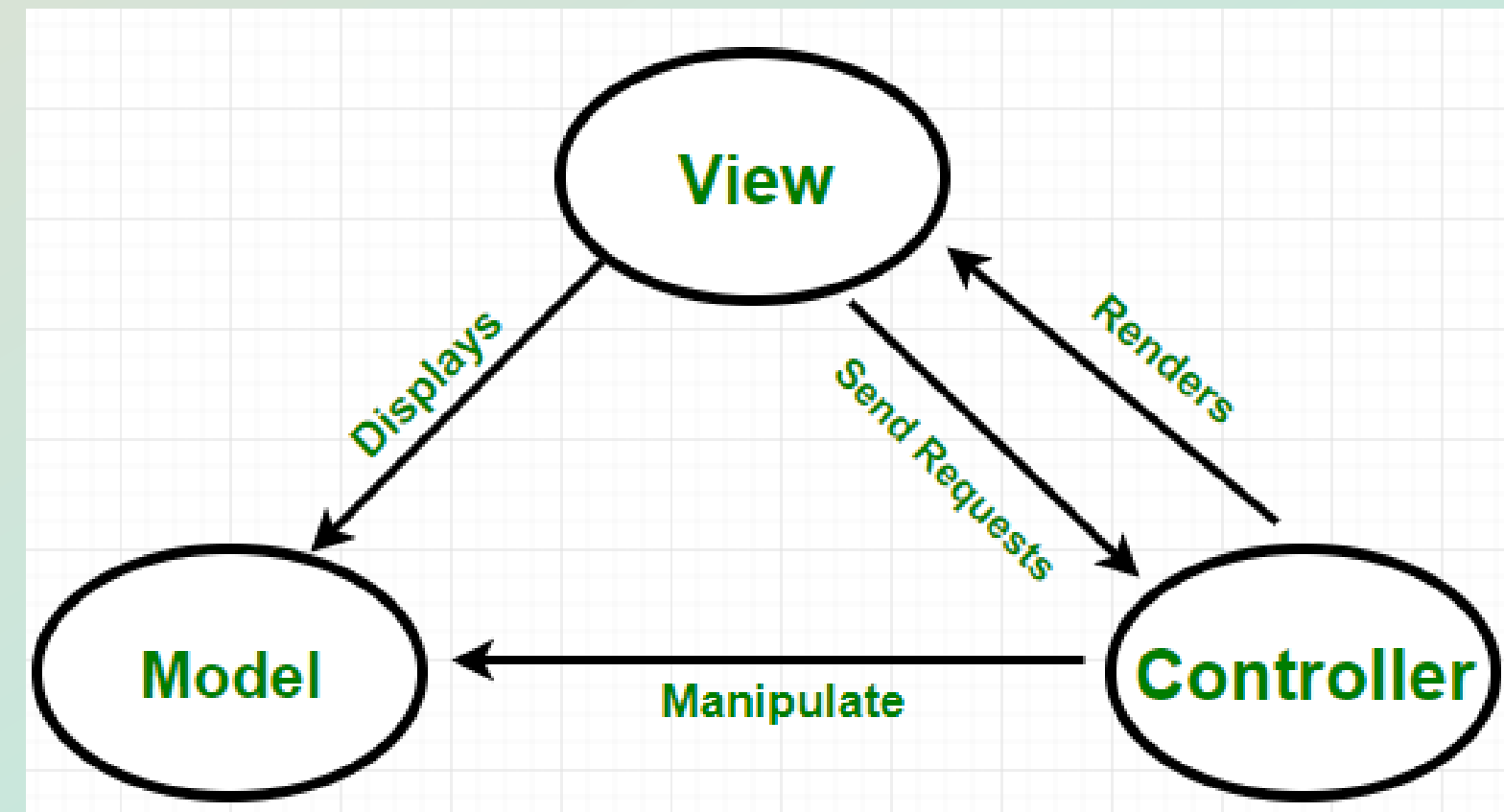
An abstract view  
of all the packages  
involved



# Architectural Style

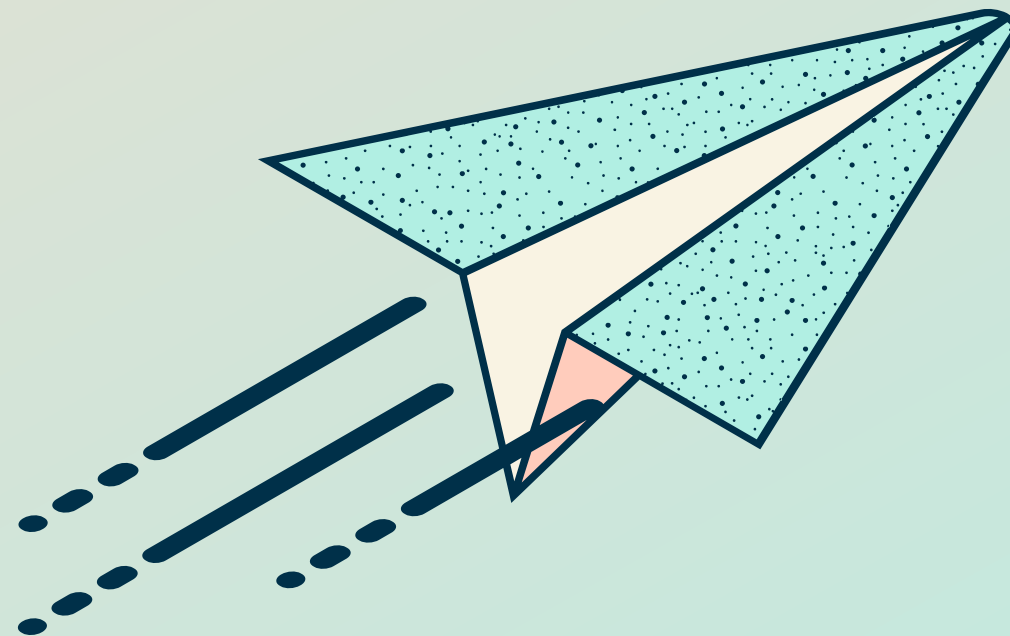
## MVC Architecture Highlights

- **Model:** Handles business logic and data (Service and Repository layers in Spring Boot)
- **View:** Displays data to users (HTML/CSS frontend)
- **Controller:** Receives user requests and delegates to appropriate services.
- **Separation of Concerns:** Each layer manages a distinct responsibility
- **Easier Maintenance and Scalability:** Clear structure for future updates



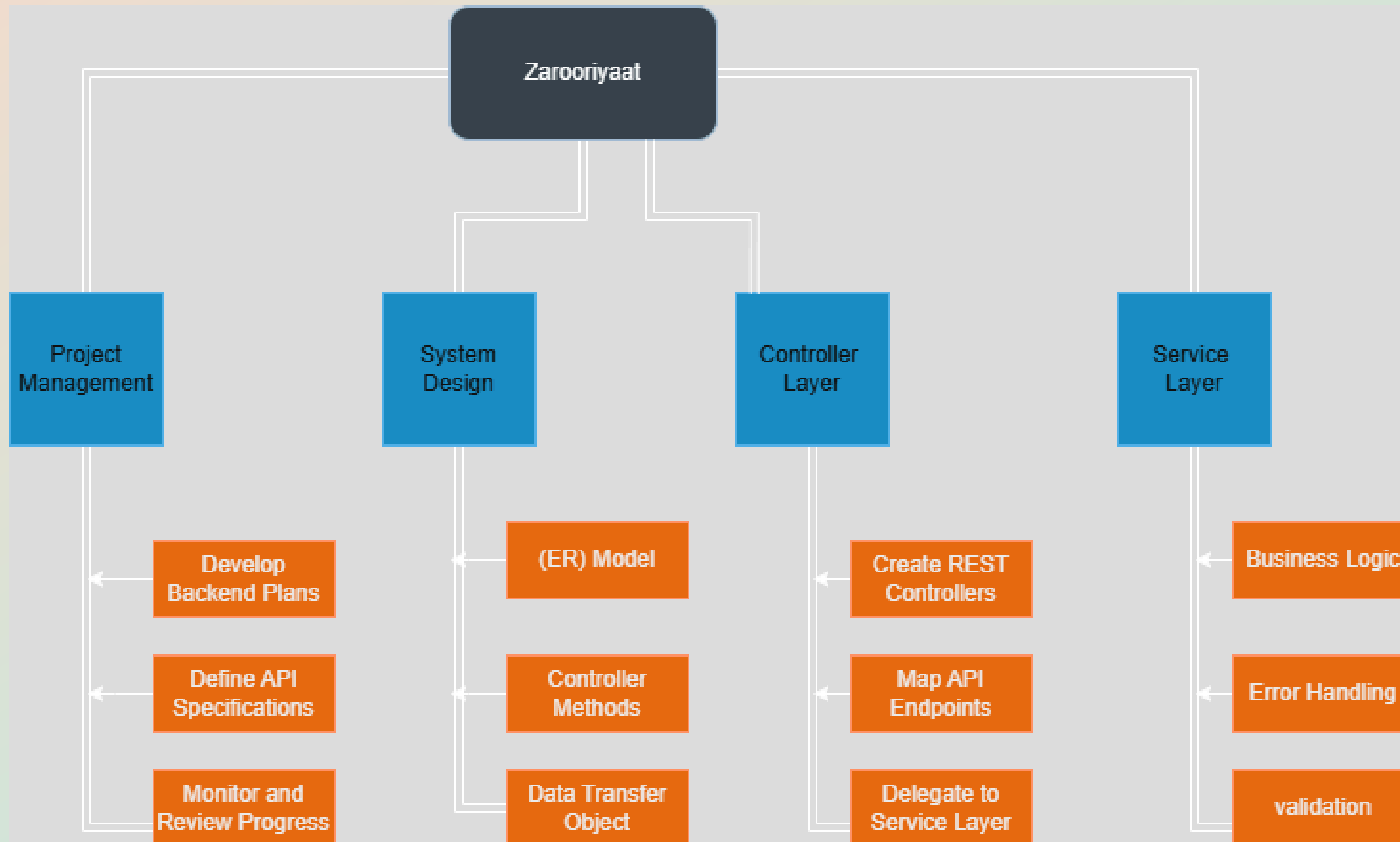


# Project Planner





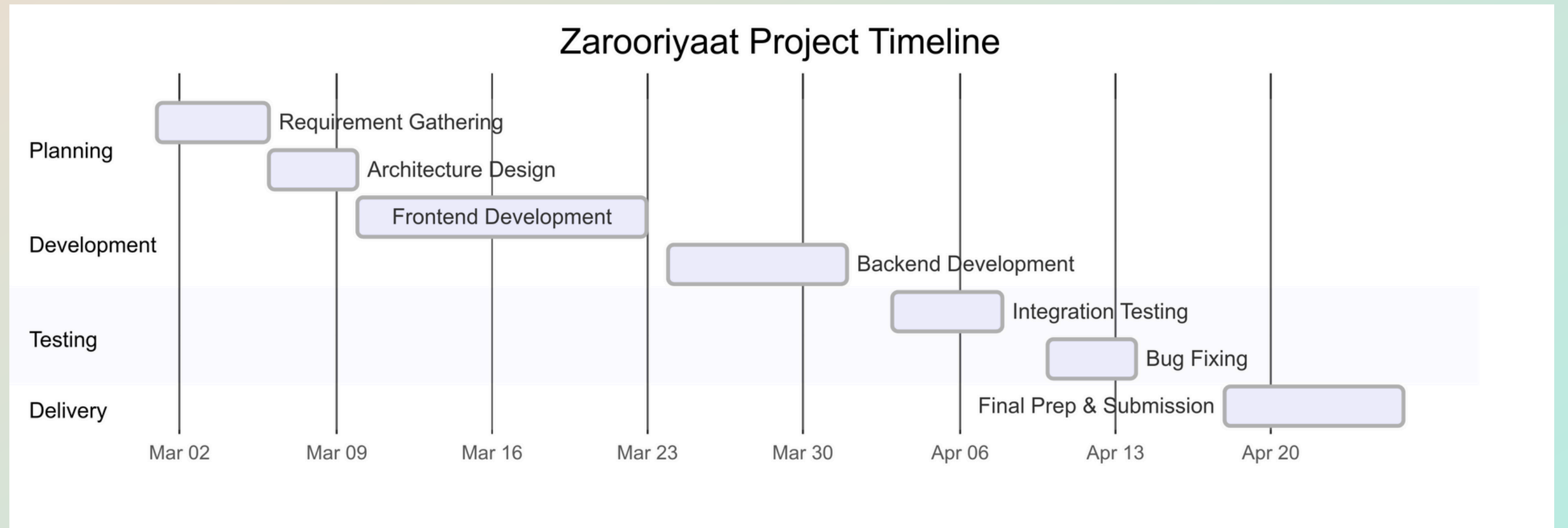
# Work Breakdown Structure



# Gantt Chart

## Zarooriyaat Project Timeline

- Y-axis lists the tasks.
- X-axis shows the timeline in days.
- Each bar spans from the task's start day for its duration.



# Sprint Planning

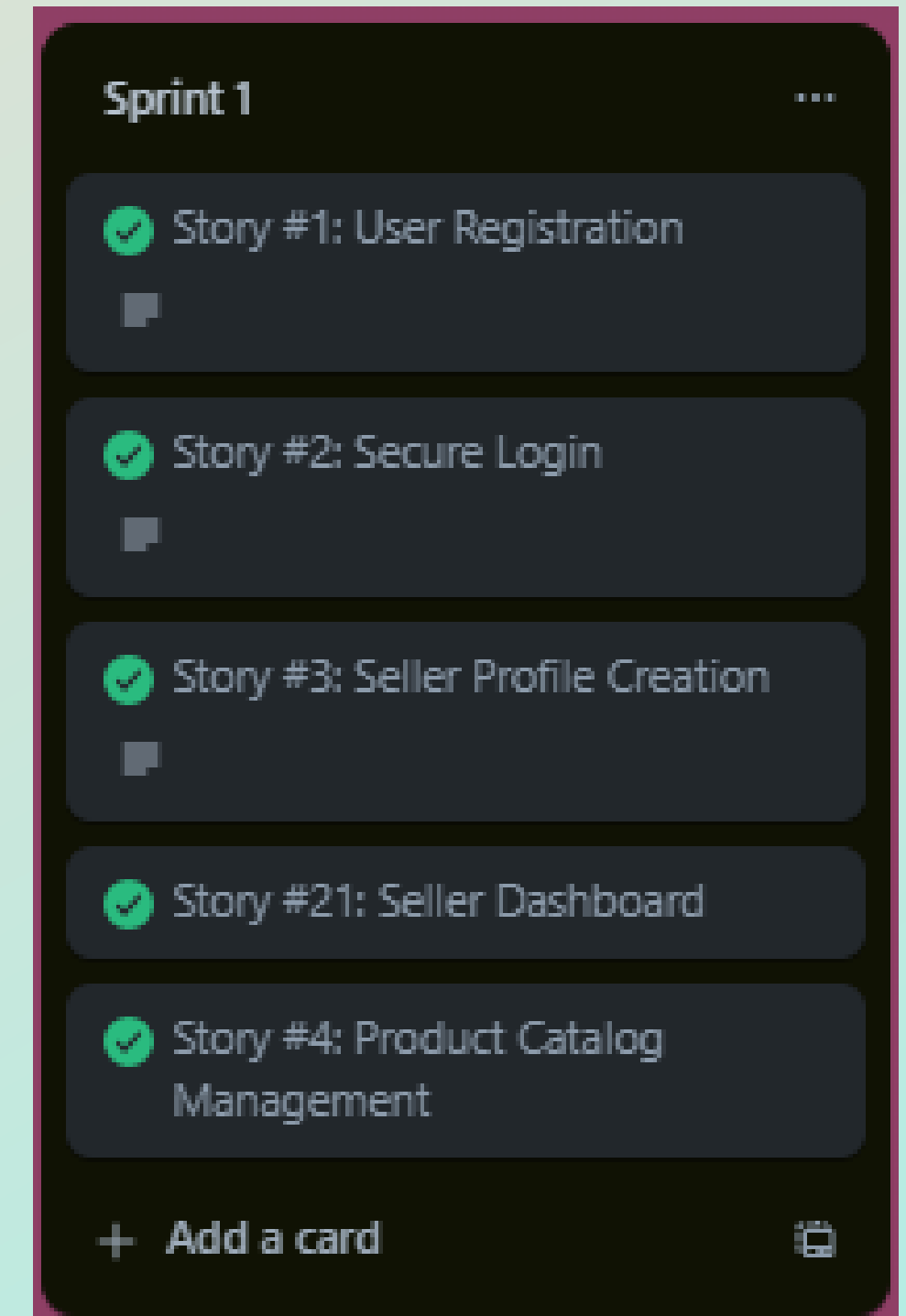
The screenshot displays a Trello workspace named 'Zarooriyaat'. The top navigation bar includes 'Workspaces', 'Recent', 'Starred', 'Templates', and a 'Create' button. A search bar and user avatars are on the right. The board is set to 'Board' view and features five columns: 'Product Backlog (All Useases)', 'Sprint 1', 'Sprint 2', 'Doing', and 'Done'. Each column contains task cards, with 'Sprint 1' and 'Done' cards marked with green checkmarks. A horizontal scrollbar is visible at the bottom.

Product Backlog (All Useases)	Sprint 1	Sprint 2	Doing	Done
Story #18: Read Customer Reviews	✓ Story #1: User Registration	Story #6: Wishlist	Story #6: Wishlist	✓ Story #1: User Registration
Story #19: Low Stock Alerts	✓ Story #2: Secure Login	Story #7: View Product Details	Story #7: View Product Details	✓ Story #2: Secure Login
Story #20: Real-time Shipment Tracking	✓ Story #3: Seller Profile Creation	Story #8: Shopping Cart	+ Add a card	✓ Story #3: Seller Profile Cre
Story #21: Seller Dashboard	✓ Story #4: Manage Profile	Story #9: Secure Checkout		✓ Story #4: Manage Profile
Story #23: Returns and Refunds	✓ Story #5: Product Search	Story #10: Order Tracking		✓ Story #5: Product Search
Story #24: Social Media Integration	+ Add a card			+ Add a card
Story #25: Mobile-Friendly Interface				
Story #26: Multi-Vendor Support				
+ Add a card				

# Sprint 1 Backlog

The following user stories were selected for Sprint 1:

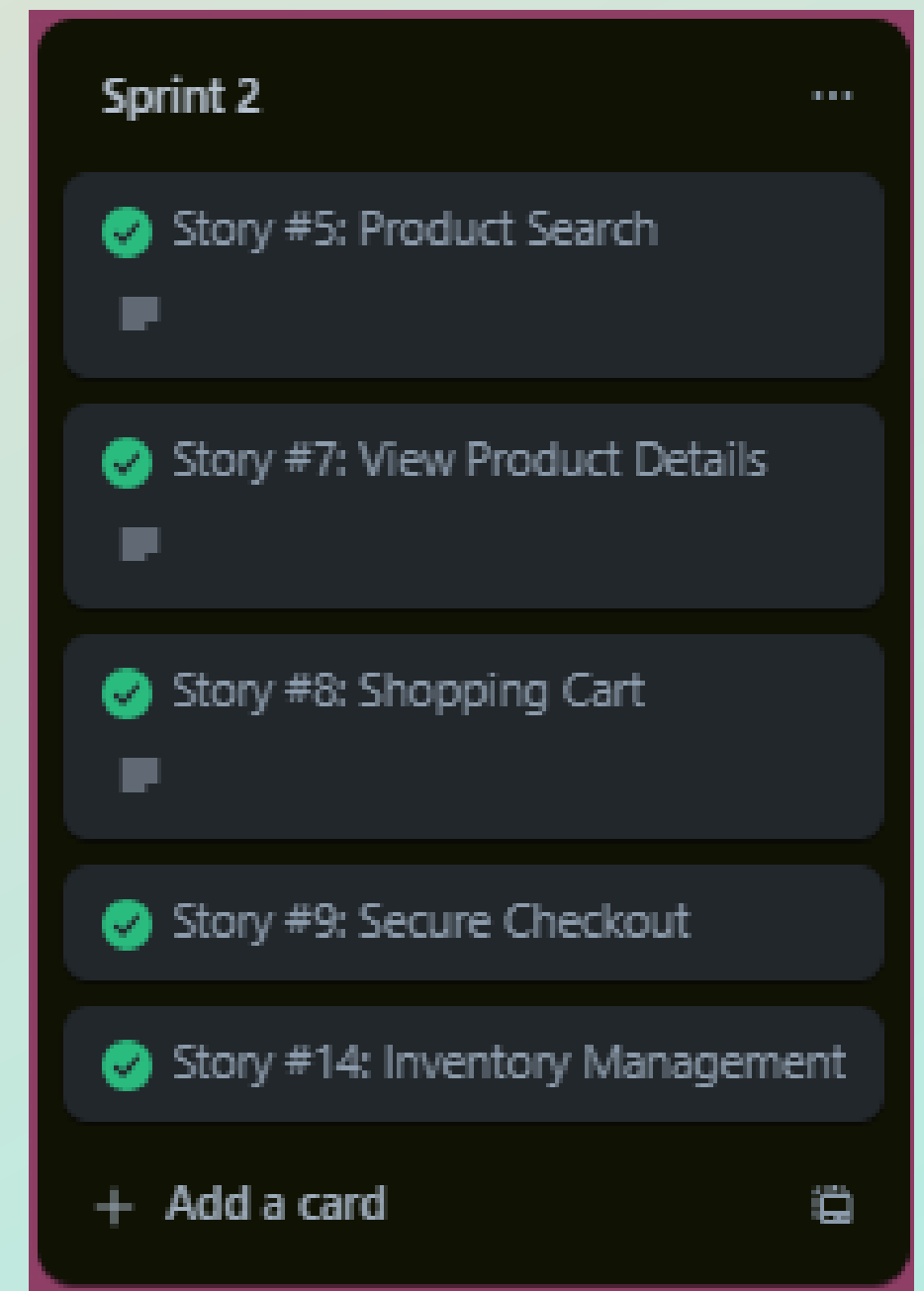
1. Story #1: User Registration (High Priority)
2. Story #2: Secure Login (High Priority)
3. Story #3: Seller Profile Creation (High Priority)
4. Story #21: Seller Dashboard (Medium Priority)
5. Story #4: Product Catalog Management (High Priority)



# Sprint 2 Backlog

Sprint 2 focused on enhancing the buyer experience:

1. Story #5: Product Search (High Priority)
2. Story #7: View Product Details (High Priority)
3. Story #8: Shopping Cart(High Priority)
4. Story #9: Secure Checkout (High Priority)
5. Story #14: Inventory Managment (Medium Priority)



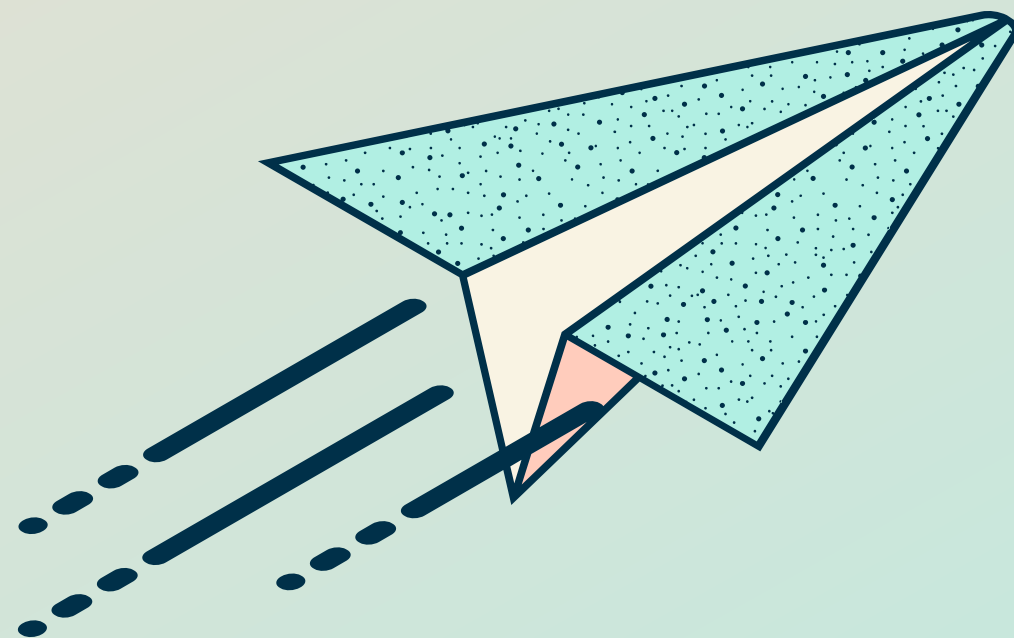
# Sprint 3 Backlog

Sprint 3 finalized:

1. Story #15: Order Management (High Priority)
  2. Story #16: Product Reviews(MediumPriority)
  3. Story #12: Track Order(MediumPriority)
- and so on.



# Important User Stories



# User Story 1

## **Product Catalog Management (Seller Side)**

As a seller, I want to manage my product catalog, So that I can add, edit, or remove products as needed.

### **Pre-conditions:**

- The seller must have a verified account.

### **Post-conditions:**

- The catalog updates reflect in the product listings.
- Inventory levels are maintained.



# User Story 2

## **Secure Checkout**

As a buyer, I want to purchase items securely, So that I can complete my transactions without issues.

### **Pre-conditions:**

- The user must have items in the cart.
- The user must provide valid payment details.

### **Post-conditions:**

- Payment is securely processed.
- An order confirmation message is displayed.

# User Story 3

## **Sales Analytics**

As a seller, I want to view sales analytics, So that I can track my business performance.

**Pre-conditions:** The seller must have sales data.

**Post-conditions:** The system generates reports and trends

# User Story 4

## **Product Feedback**

As a buyer, I want to leave reviews and ratings, So that I can share my experience with other users.

**Pre-conditions:** The user must have purchased the product.

**Post-conditions:** The review appears on the product page.

# User Story 5

## **Manage Profile**

As a registered user, I want to update or delete my profile information, So that I can keep my account up to date or remove my presence from the platform.

### **Pre-conditions:**

- The user must be logged in.

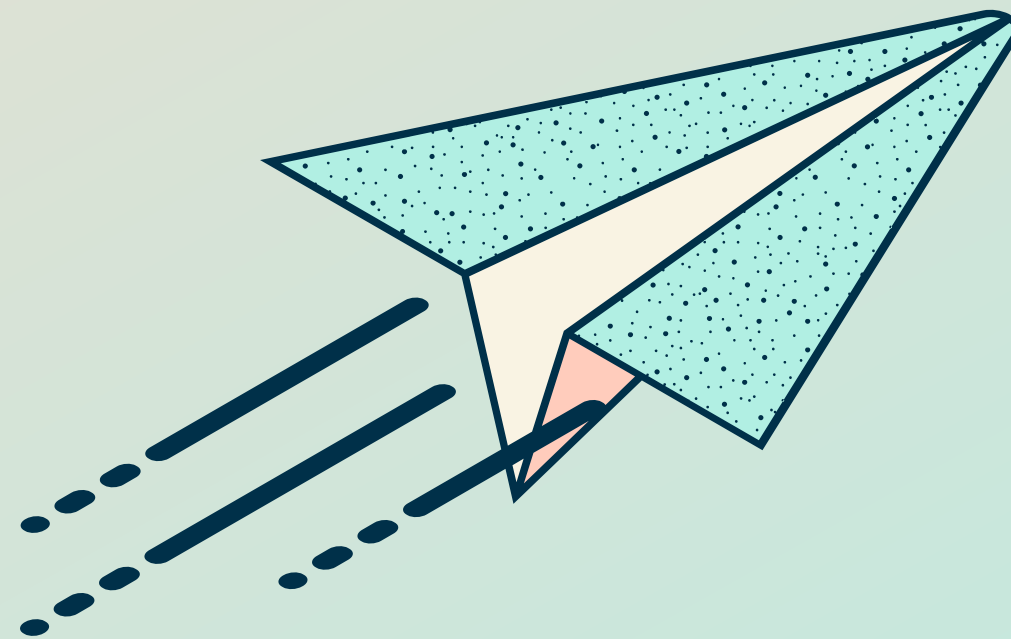
### **Post-conditions:**

- The profile information is updated.
- Account deletion is permanent and irreversible.



# Blackbox Testing

Testing without peeking at source—verifying inputs  $\leftrightarrow$  expected outputs





# Blackbox Testing

- **Boundary Value Analysis**
  - Username/password length: 1 char, min length, max length + 1.
- **Functional Test Cases**
  - “Add item to cart” with zero quantity → expect validation error.
  - “Checkout” with no Payment → expect payment failure message.
- **Usability & UI Tests**
  - Links/buttons render and navigate correctly on click.



# Blackbox Testing

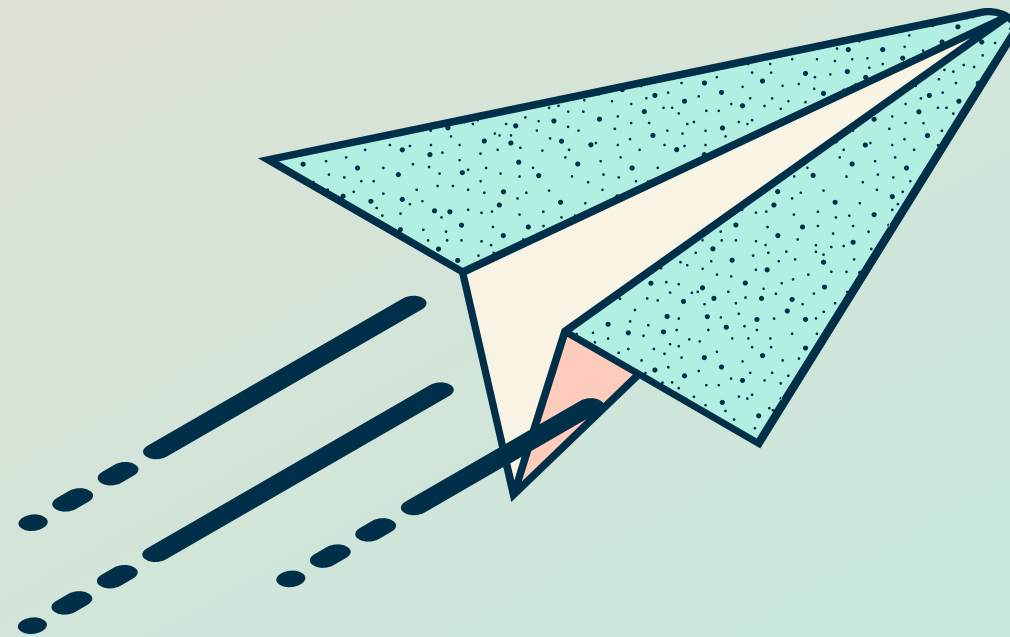
```
@Test
void testSubmitReview_ValidInputs() {
    // Test Case 1: Valid product ID and rating
    assertTrue(reviewService.submitReview(validProductId, rating:5));
    assertEquals(expected:1, reviewRepository.findByProductId(validProductId).size());
}

@Test
void testSubmitReview_InvalidProductId() {
    // Test Case 2: Invalid product ID
    assertFalse(reviewService.submitReview(invalidProductId, rating:5));
    assertEquals(expected:0, reviewRepository.findByProductId(invalidProductId).size());
}
```



# Whitebox Testing

Tests that inspect internal code—covering branches, paths, and logic.







# Whitebox Testing

- **Unit Tests (JUnit)**
  - Services: e.g. ProductServiceTest covering addProduct() etc.
- **Code Coverage**
  - Aimed for 80%+ on core modules (controllers, services).
- **Path & Branch Coverage**
  - Tested both “if success” and “if failure” branches in checkout() logic.



# Whitebox Testing

## Sample J-Unit Snippet:

```
@Test
void whenValidUser_thenAuthenticate() {
    User u = authService.login("test@x.com", "pass123");
    assertNotNull(u);
    assertEquals("test@x.com", u.getEmail());
}
```



# Whitebox Testing

```
[INFO] Tests run: 5, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.
[INFO] Running com.example.zarooriyaat.SignupEntityTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.
[INFO] Running com.example.zarooriyaat.ZarooriyaatApplicationTests
2025-05-08T10:59:21.182+05:00 INFO 4228 --- [zarooriyaat] [           ma
on classes for test class [com.example.zarooriyaat.ZarooriyaatApplication
inal, nested classes annotated with @Configuration.
2025-05-08T10:59:21.187+05:00 INFO 4228 --- [zarooriyaat] [           ma
xample.zarooriyaat.ZarooriyaatApplication for test class com.example.zarooriyaat.ZarooriyaatApplicationTests
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.
```



# Further Improvements

## **Payment Gateway Integration**

- Add real online payment support using Stripe or PayPal instead of dummy checkouts

## **Dark Mode & Accessibility Features**

- Add theme toggling, keyboard navigation, and screen-reader-friendly design.

## **OTP Verification & Email Notifications**

- Send OTP for login/registration and notify users of order status or promotions.



# Conclusion

- **Key Takeaways:** Sprint planning and development in patterns like MVC.
- **Team Reflection:** Agile methodology and Trello/GitHub kept us aligned.

# Meet the Team



**Mahad Rehman**

Scrum Master

**Hashim Awan**

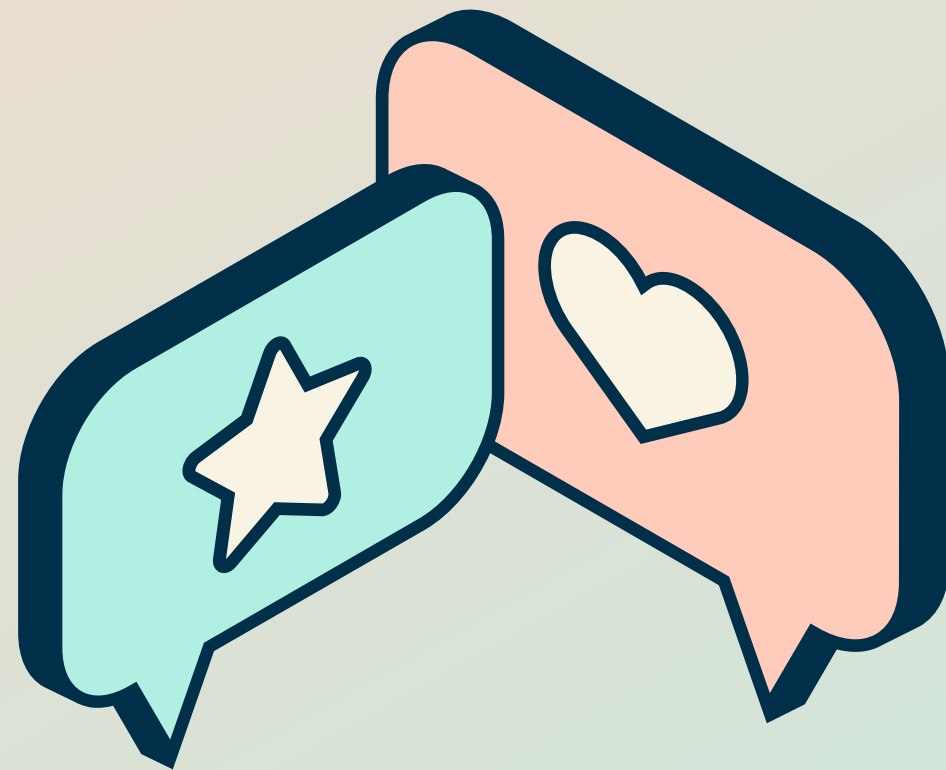
Senior Developer



**Masab Hammad**

Senior Developer

# THANK YOU!



## Email Address

zarooriyaat@gmail.com



## Phone Number

+92 333 9131317



## Mailing Address

AK. Brohi Road, H11/1