Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur

Students Kit

Objective

These guidelines are intended to help students progress through the development of the E-Post Office Web Application. Below are templates for various project documents, which serve as guidelines. Teams are encouraged to improve upon these templates.

Requirements Specification (RS)

This template captures all system requirements. Even the smallest details should be documented for easier validation during testing.

No.	Requirement	Essential / Desirable	Description of the Requirement	Remarks
RS1	The system should have a login	Essential	A Welcome Page should appear when the URL is accessed, with a login option.	
RS2	The system should have help screens	Essential	Help on various features should be provided in a Q&A format.	The leave policy should also be part of the help.
RS3	The system should lock the login after 3 attempts	Desirable	The system should lock the user if the wrong password is entered 3 times.	Optional for this project but can improve security.

Database Fields Specification

In this project, **Product ID** is the key for the products database, and **Customer ID** is the key for the customer database. When a customer registers, they receive a unique customer ID, and their transactions are tracked in the database.

No.	Field Name	Range of valid values for the field	Remarks
1	Product Id	1 to 1000	Unique identifier for each product.
2	Product Name	Up to 30 characters	Name of the product (e.g., stamps, letters).
3	Price	Numeric	Unit price of the product.

Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur

High Level/Detailed Design (HLD/DD)

Overview of the system

The system architecture consists of the database hosted on MongoDB, with the application running on Node.js/Express and the front-end built using React.js.

Design Components

The system is divided into key components. Below is a template for defining each component.

- **Component One: User-verification**

  Purpose
  This component checks if a user is valid and allows login.

  Pseudocode
  Pseudocode is written to get more clarity on the component so that the actual implementation is made easier. For the user-verification component:

```
bool verify_user(user_id, password1)
{
 Get user_id and password;
 If login_id_valid(user_id) then
   access_database_entry(user_id);
   Retrieve encrypted_password;
   Compare password1 with decrypted password;
   If matched, enter_system();
   Else, report error 'Incorrect password';
 Else, report error 'Invalid login id';
}
```

- **Component Two: Product Catalog**

  Purpose: Displays available stamps and letters and handles order placement.

- **Component Three: Order Management**

  Purpose: Tracks and manages user orders, including payment and shipment.

Test-Plan (TP)

The test plan is a list of test cases to ensure the system meets the requirements. Test each component individually before full integration.

No.	Testcase Title	Description	Expected Outcome	The requirement in RS that is being tested	Result
1	Successful User Registration	Attempt login with valid credentials.	User successfully logs in.	RS1	Passed
2	Unsuccessful User Verification due to wrong password	Attempt login with incorrect password.	Error message: 'Invalid Password'.	RS1	Passed
3	Unsuccessful User Verification due to invalid login id	Attempt login with invalid user ID.	Error message: 'Invalid user ID'.	RS1	Passed
4	Successful Product Catalog Display	Display products from the database.	Products are listed correctly.	RS4	Passed