Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur

Students Kit

Objective

These guidelines are for students to adopt in order to make progress in the project. Below are the templates for the documents related to the project. These are just guidelines; the team can improve these.

Requirements Specification (RS)

Following is a template for the RS document. Some example requirements are entered in to it to show how to use the template. Make sure that you enter even the smallest/most trivial requirements also. That would help in validating the system 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 invoked. The welcome page should have a login feature.	
RS2	The system should have help screens	Essential	Help about the various features of the system should be provided in sufficient detail in a Q&A format.	The help section should include information relevant to managing criminal records.
RS3	The system should lock the login ID after 3 wrong attempts	Desirable	This feature will improve the robustness of the application.	Since the application is intended for law enforcement, this feature is not essential, but it can be implemented if time allows.
RS4	The system should implement client-side validation	Essential	Client-side validation will reduce unnecessary calls to the back end and minimize network traffic.	This can help improve the performance of the application.

Database Fields Specification

The **User ID** is the key of the user database. The **Case ID** is the key of the criminal record database. When a new user registers with the portal, they will receive a user ID. Their transactions will be tracked in the user database. The details of each criminal record (such as suspect information, charges, etc.) will be stored in the criminal record database. For the criminal record database, the different fields and the range of valid values are given in the following table. These values/fields are examples and can be modified by the team.

User_info

No.	Field Name	Range of valid values for the field	Remarks
1	User Id	1 to 1000	This is the key field of the database as it is unique for each user.
2	Username	Up to 15 characters in length.	Name of the user
3	Password	Up to 20 characters	User's encrypted password (stored securely)
4	Email	'Officer' or 'Admin'	Defines the access level of the user

Criminal_info

No.	Field Name	Range of valid values for the field	Remarks
1	Name	1 to 1000	This is the key field of the database as it is unique for each user.
2	Age	Up to 15 characters.	Name of the user
3	Gender	Up to 10 characters	User's encrypted password (stored securely)
4	Crime Type	Varchar(100)	Defines the type of the crimes committed
5	Crime Severity	ENUM('Low', 'Moderate', 'High')	ENUM('Low', 'Moderate', 'High')
6	Arrest Date	Date	Used for Record
7	Arrest Location	Varchar(100)	Can be any location
8	Officer In charge	Varchar(100)	Officer handling the case
9	Case Status	ENUM	Current status of the case
10	Description	TEXT	Detailed explanation of case
11	Prison Name	Varchar(100)	Palce where prisoner is kept
12	Release Date	Date	Optional if the case is closed
13	Image	Selected from the system	Image of the accused, URL stored in Vercel Blob

High Level/Detailed Design (HLD/DD)

Overview of the system

Provide a block diagram depicting where the database will be located, where the application will run, etc. Also, provide details about the database server that will be used.

Design Components

Split the system into its design components. In this case, the components would be user registration, criminal record management, case management, etc. For each of the components, provide information in the following format. The **User Verification** component is taken as an example.

Component one
User-verification

Purpose
This component will verify if the user who is trying to access the system is a valid user.

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 the user_id (which is the login) and the password from the user.
  Get_login_and_password();

  // Verify if this is a valid login (i.e., from 1 to 1000).
  If login_id_valid(user_id) {
    report_error('invalid login id');
    return false;
  }

  // Access the database entry for this
  if get_database_entry(user_id, database_entry) {
    // Get the encrypted password.
    Get_encrypted_password(user_id, password2);

    // Decrypt the password. The decrypted password is password3.
    Decrypt_password(password2, password3);

    // Compare the passwords.
```

```
    If compare_passwords(password1, password3) {
      // Enter the system.
      Enter_system();
    } else {
      // Password comparison failed.
      report_error('incorrect password. Try again.');
    }
  } else {
    // Unable to get the database entry
    report_error('invalid login');
  }
}
```

Component two
Criminal Record Management

Purpose
This component manages and stores information regarding criminal records, including suspects, charges, and case outcomes.
Pseudocode

```
Function manage_criminal_record(action, record) {
  If action == 'add' {
    // Add a new criminal record to the database
    insert_into_database(record);
  } Else If action == 'update' {
    // Update an existing criminal record
    update_database(record);
  } Else If action == 'delete' {
    // Delete a criminal record
    delete_from_database(record.id);
  }
}
```

Component three
Case Management

Purpose

This component tracks the status of cases and manages related information.

Pseudocode

```
Function manage_case(case_id, status) {
   // Update the status of a specific case
   update_case_status(case_id, status);
}
..
```

Test-Plan (TP)

The test plan is basically a list of test cases that need to be run on the system. Some test cases can be run independently for some components, while others require the whole system to be ready for execution. It is better to test each component as soon as it is ready before integrating them.
It is important to note that the test cases cover all aspects of the system (i.e., all the requirements stated in the RS document).

No.	Testcase Title	Description	Expected Outcome	The requirement in RS that is being tested	Result
1	Successful User Registration	The login to the system should be tried with the login assigned by the admin and the correct password.	Login should be successful, and the user should enter the system.	RS1	Passed
2	Unsuccessful User Verification due to wrong password	Login to the system with a wrong password.	Login should fail with an error 'Invalid Password'.	RS1	Passed
3	Unsuccessful User Verification due to invalid login id	Login to the system with an invalid login ID.	Login should fail with an error 'Invalid User ID'.	RS1	Passed
4	Verify User Lockout after 3 Failed Attempts	Attempt to login with an invalid password three times.	User ID should be locked, and an appropriate error message should be displayed.	RS3	Passed