

**Name** – Ritik raja

**Email ID** – [Ritik.rock121@gmail.com](mailto:Ritik.rock121@gmail.com)

**Title:** – Online Survey System Documentation in Java.

**Task Description** – This documentation provides an overview of the Online Survey System developed using Java. The system allows users to create surveys, distribute them, collect responses, and generate insightful reports. It leverages Object-Oriented Programming (OOP) concepts and Java Swing for the graphical user interface.

### **Project Details:**

**Authentication:** Users are required to provide valid credentials (username and password) to log in.

**Survey Creation:** Provided a user-friendly interface for creating surveys with various question types such as multiple choice, text input, and rating scales.

**Survey Distribution:** Enabled users to distribute surveys to targeted respondents via email invitations or a web interface.

Generated unique survey links for each respondent to track responses.

Stored survey distribution data in the database.

**Response Collection:** Allowed respondents to complete surveys online securely. Handled error cases such as duplicate responses or incomplete surveys.

**User Management:** Implemented user roles and permissions to control access to survey creation, response collection, and report generation functionalities.

**Scalability and Reliability:** Ensured the system is scalable and reliable to handle multiple users and surveys simultaneously.

**Deployment:** Deployed the Online Survey System on a server or hosting platform.

Ensured scalability, reliability, and security of the deployed application

## **Step Taken:-**

### Step 1: Create a Java Project

Begin by setting up a new Java project in your chosen Integrated Development Environment (IDE) or code editor.

### Step 2: Define Data Model

Created Java Classes to represent core entities of the survey system such as Survey, Question, Response, and user.

### Step 3: Create User Class

Defined appropriate properties and methods for each class to capture relevant data and behavior.

### Step 4: Create Database Management

Chose a relational database management system (RDBMS) like MySQL or SQLite. Created a database schema to store survey data, user information, and responses. Implemented database connectivity using JDBC to interact with the database from Java code.

### Step 5: Implement Survey Creation, Distribution, and Response Collection

Designed forms or dialogs for users to create surveys. Allowed users to add various types of questions to the survey. Created mechanisms to distribute surveys to respondents via email or a web interface. Designed interfaces for respondents to complete surveys online.

### Step 6: Implemented Report Generation

Analyzed survey responses to generate insightful reports. Used libraries like Apache POI or iText to create PDF or Excel reports. Displayed graphical representations of survey data using charting libraries like JFreeChart or Apache Commons Chart.

## Step 7: Implemented User Authentication and Authorization

Created login screens for users to authenticate themselves. Implemented authorization mechanisms to control access based on user roles and permissions.

## Step 8: Testing and Debugging

Conducted thorough testing to ensure the system functions as expected. Tested different scenarios, including edge cases and error conditions. Debugged any issues encountered during testing and fixed them promptly.

## **Deployment:-**

1. Deployed the Online Survey System on a server or hosting platform.
2. Ensured scalability, reliability, and security of the deployed application.
3. Monitored the system for performance and addressed any issues as they arose.

## **Conclusion:**

The Online Survey System provides a comprehensive solution for creating, distributing, and analyzing surveys. It is designed with scalability, reliability, and security in mind, ensuring a seamless experience for both administrators and respondents.