

# Online Survey G Feedback System

---

Submitted by:

Name : Johnson Kumar

U.I.D : 23bcs12654

## Project Description

The *Online Survey & Feedback System* is a web-based application designed to create, manage, and analyse surveys efficiently. It enables administrators to build customized surveys, participants to provide responses through a user-friendly portal, and stakeholders to evaluate results using interactive dashboards. Built using *Spring Boot* for backend services, *React.js* for the frontend interface, and *MongoDB* for scalable data storage, the system provides a dynamic and secure platform for conducting surveys across academic, corporate, and research domains.

## Key Features:

- Dynamic survey creation with multiple question types.
- Secure participant portal for submitting responses.
- Real-time aggregation and visualization of results.
- Role-based access for administrators, creators, and participants.

## Project Distribution into Modules

The project is divided into five core modules to ensure clarity, scalability, and effective development:

### 1. Frontend – Survey Builder & Participant Portal (React.js)

- Survey Builder UI:
  - Create dynamic forms (MCQs, text fields, ratings, etc.).
  - Add, edit, and arrange questions.
  - Preview surveys before publishing.
- Participant Portal UI:
  - Display available surveys.

- Provide responsive forms to submit answers.
- Ensure a clean and mobile-friendly design.

## 2. Frontend – Result Dashboard & Admin Panel (React.js)

- Result Dashboard:
  - Show aggregated results in visual charts/graphs (Chart.js/Recharts).
  - Enable filtering by date, survey type, or user group.
  - Provide export options (CSV, Excel, PDF).
- Admin Panel:
  - Manage surveys (publish/unpublish/delete).
  - Control user roles (Admin, Creator, Participant).
  - Monitor participation rates and activity.

## 3. Backend & Database (Spring Boot + MongoDB)

- API Development (Spring Boot):
  - CRUD APIs for surveys and responses.
  - Authentication APIs (login, signup).
  - Secure communication with JWT tokens.
- Database (MongoDB):
  - Store surveys in JSON format (flexible schema).
  - Store participant responses and link them to surveys.
  - Optimize queries for fast analytics.
- Integration:
  - Connect backend APIs with frontend React components.
  - Deploy backend and database on cloud platforms (Render, MongoDB Atlas).

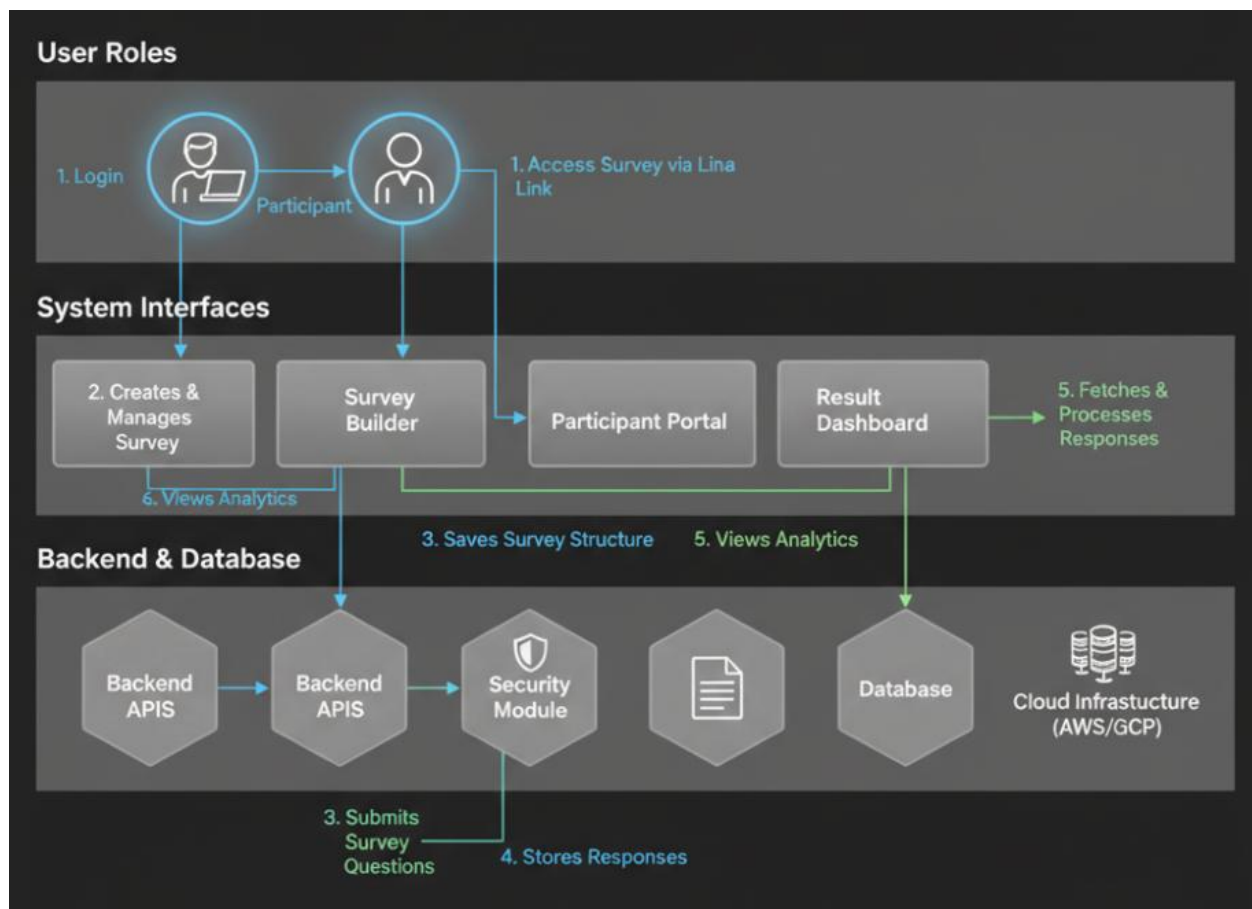
## 4. Security & Authentication Module

- Implement JWT-based authentication (login sessions).
- Role-based access control (Admin, Creator, Participant).
- Secure API endpoints with Spring Security.

- Input sanitization and validation to prevent attacks (e.g., SQL/NoSQL injection, XSS).

## 5. Integration & Deployment Module

- Connect frontend (React) with backend (Spring Boot APIs).
- Integrate backend with MongoDB Atlas (cloud-hosted DB).
- Add environment configs for development/production.
- Deploy application on platforms like Render / Vercel / Netlify + Heroku/Spring Boot server.



# Report on Module 2: Frontend – Result Dashboard & Admin Panel (React.js)

## Introduction

Module 2 of the *Online Survey & Feedback System* emphasizes the frontend development of the *Result Dashboard* and *Admin Panel*, providing administrators and stakeholders with comprehensive tools to monitor, manage, and analyse survey data. Utilizing *React.js*, the module delivers dynamic, interactive, and visually appealing interfaces that enable

The dashboard and admin panel are designed with performance, accessibility, and user experience as priorities, ensuring administrators can monitor and manage surveys effectively across multiple dimensions. Additional features include interactive visualizations, role-based access control, and exportable reports to facilitate both operational and strategic decisions.

## Objectives

- To provide administrators with a detailed interface for comprehensive monitoring of survey activity.
- To offer stakeholders clear and actionable insights into survey outcomes through interactive charts, graphs, and reports.
- To implement role-based access control for secure management of surveys, users, and system permissions.
- To enable seamless integration with backend APIs for real-time data retrieval, analytics, and reporting.
- To maintain a responsive and user-friendly interface that adapts to various devices and screen sizes.
- To create a foundation for advanced analytics and visualization features in future system expansions.
- To facilitate audit and compliance through detailed logs, export options, and secure data handling.

## Workflow

### Result Dashboard

#### 1. Data Visualization:

- Aggregates survey responses and displays them using charts, graphs, and tables.

- Users can filter and sort results by date, survey type, demographic groups, or custom parameters.
- Export features allow administrators to save survey data in CSV, Excel, or PDF formats for offline use and further analysis.
- Supports dynamic visualizations that adjust automatically as new responses are received.

## **2. Interactive Insights:**

- Hover and click capabilities enable detailed drill-downs into specific survey responses.
- Comparative analysis allows side-by-side evaluation of multiple surveys.
- Color-coded and annotated charts highlight trends, anomalies, and areas requiring attention.

## **3. Real-Time Monitoring:**

- Tracks participation rates, survey completion status, and response timelines.
- Provides alerts or notifications for low engagement, incomplete surveys, or irregular data patterns.
- Facilitates immediate decision-making by presenting actionable insights in real-time.

# **Admin Panel**

## **1. Survey Management:**

- Administrators can publish, unpublish, archive, or delete surveys.
- Edit survey details, question configurations, or scheduling without affecting active participants.
- Version control ensures that previous survey iterations are maintained for historical analysis.
- Supports scheduled deployments and automatic notifications to participants.

## **2. User & Role Management:**

- Assign roles such as Admin, Creator, or Participant to ensure proper access control.

- Monitor user activities, login history, and participation metrics for audit purposes.
- Enforce role-based restrictions on survey creation, editing, and data access to maintain security.

### 3. System Monitoring:

- View overall system analytics including active sessions, usage trends, and performance metrics.
- Detect bottlenecks or issues through dashboards designed for operational oversight.
- Maintain logs for troubleshooting and compliance reporting.

## Technology Stack

- **Frontend Framework:** React.js for responsive, component-based UI development.
- **Charting Libraries:** Chart.js, Recharts, or D3.js for interactive data visualization.
- **State Management:** React Context API to efficiently handle state across dashboards and admin components.
- **API Communication:** Axios or Fetch for secure and reliable connection to Spring Boot backend APIs.
- **Routing:** React Router to manage navigation between dashboard, admin panel, and other interface components.
- **Responsive Design:** Material-UI / Tailwind CSS for styling, adaptive layouts.
- **Data Security:** Input validation and secure API endpoints ensure data integrity and user privacy.

## Features Implemented

- Interactive and dynamic charts, tables, and graphs for visualizing survey results.
- Filtering, sorting, and aggregation functionalities for in-depth analysis.
- Role-based access control to enforce security and proper administrative operations.
- Export options for CSV, Excel, and PDF to facilitate reporting and offline analysis.
- Real-time updates to display the latest survey responses and participation metrics.
- Intuitive and responsive UI to support efficient navigation and task completion.
- Integration with backend APIs to enable CRUD operations on surveys, responses, and user management.

- Alerts and notifications to guide administrators towards key insights or required actions.

## UI Flow Diagram

### 1. Result Dashboard Flow:

- Dashboard Home → Select Survey → View Charts/Graphs → Apply Filters → Export Data → Track Trends → Generate Reports.

### 2. Admin Panel Flow:

- Admin Login → Dashboard Overview → Manage Surveys → Edit/Delete Surveys → Assign Roles → Monitor User Activity → View System Analytics → Address Issues.

## Expected Outcomes

- Enhanced ability to make data-driven decisions with visually interactive dashboards.
- Streamlined survey management process for administrators, including publishing, editing, and archiving surveys.
- Real-time monitoring and insights into participant engagement and survey completion trends.
- Secure and user-friendly admin interface, with clear role-based access control.
- Support for offline data analysis and reporting through export features.
- Foundation for advanced analytics, including trend prediction and cross-survey comparisons.
- Improved overall system efficiency, reliability, and user satisfaction.

## Conclusion

Module 2 provides a robust and comprehensive frontend for administrators and stakeholders, empowering them to manage surveys effectively, monitor participant engagement, and visualize results with clarity. Leveraging React.js and modern charting libraries, the module ensures a dynamic, responsive, and secure interface, forming a critical part of the *Online Survey & Feedback System*. With enhanced functionality, real-time updates, and advanced data visualization, Module 2 significantly strengthens the operational and analytical capabilities of the system.