A Midterm Progress Report

on

SURWAY: THE SMART WAY TO SURVEY

Submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY

(Computer Science Engineering)

SUBMITTED BY:

SAKSHAM KOHLI	SIMARPREET KAUR	TAVLEEN KAUR
2104176	2104193	2104209

UNDER THE GUIDANCE OF

PROF. DIANA NAGPAL

(October 2024)

GURU NANAK DEV ENGINEERING COLLEGE, LUDHIANA

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Introduction to the Project

In today's educational landscape, feedback collection is a critical component of institutional development and improvement. Feedback allows institutions to assess the effectiveness of their educational programs, administrative services, and overall campus environment. As educational institutions become increasingly complex and datadriven, the need for robust, flexible, and efficient feedback mechanisms has become essential. At Guru Nanak Dev Engineering College (GNDEC), Ludhiana, feedback has long been recognized as an important tool for monitoring and enhancing performance.

In response to the growing need for advanced feedback collection and analysis, the development of SURWAY – The Smart Way to Survey has been proposed. SURWAY is a survey creation and management platform designed to address the unique requirements of educational institutions, particularly GNDEC. It aims to streamline the process of gathering feedback from various stakeholders, including students, faculty, alumni, and administrative staff.

Traditionally, feedback has been collected through paper forms or static digital questionnaires, both of which are timeconsuming and often yield limited data. These conventional methods are not equipped to handle the dynamic needs of modern educational institutions, where realtime feedback, datadriven decisionmaking, and comprehensive analysis are vital. SURWAY aims to revolutionize the feedback process by offering a platform that is both flexible and powerful, ensuring that GNDEC can gather actionable data efficiently and effectively.

Built using the MERN stack (MongoDB, Express.js, React.js, and Node.js), SURWAY is a webbased platform that provides users with a seamless experience in creating, managing, and analyzing surveys. The platform is designed to be userfriendly, yet incorporates advanced features that allow for indepth customization, realtime data processing, and comprehensive reporting.

1.2 Objectives of the Project

The primary goal of SURWAY is to provide a modern, userfriendly survey creation tool that meets the diverse feedback needs of GNDEC. To achieve this, the project has been structured around three core objectives:

- **To develop a platform that allows users to create new survey forms using predefined components.**
- **To implement features for editing and managing existing survey forms.**
- **To implement essential data analysis and reporting features for survey results.**

The first objective is to build a platform that allows users to create new survey forms effortlessly. By offering a set of predefined components, SURWAY ensures that users can quickly design surveys without needing any technical expertise. These components include a variety of question types, such as multiplechoice questions, rating scales, matrix grids etc. Users can easily drag and drop components to customize their survey forms to meet specific feedback needs.

The second objective of the project is to implement robust features for editing and managing survey forms. SURWAY provides users with the ability to modify existing surveys, allowing for updates and improvements based on evolving feedback needs. For instance, faculty may wish to update course evaluation surveys midsemester to include new questions based on ongoing feedback, or administrators may need to modify alumni feedback forms to capture more specific data.

The third objective is to provide users with the tools necessary to analyze and report on survey results. SURWAY offers builtin data analysis features that allow users to process survey results in realtime, providing immediate insights into the collected data. These features include data visualization tools, such as graphs, charts, and tables, which make it easier for users to interpret large datasets and identify key trends.

1.3 Importance of SURWAY for GNDEC

The development of SURWAY is particularly significant for GNDEC as it addresses several limitations in the institution's current feedback collection system. Traditional methods of feedback collection, such as paperbased forms, are not only cumbersome but also prone to errors, delays, and low response rates. Even digital forms, while an improvement, often lack the flexibility and realtime data analysis needed for modern educational environments.

SURWAY offers a solution to these challenges by providing a platform that is both easy to use and highly customizable. The ability to design surveys using predefined components makes the process quick and efficient, while features like conditional logic and multipage surveys ensure that respondents only see relevant questions. This not only enhances the quality of the data collected but also improves response rates by making the survey experience more engaging for users.

Furthermore, the realtime data analysis capabilities of SURWAY allow administrators and faculty to act on feedback immediately, making it possible to implement improvements while the academic term is still ongoing. The platform's builtin reporting tools also make it easier to share insights with decisionmakers, ensuring that feedback data is used to drive positive changes at GNDEC.

SURWAY's responsive design ensures that surveys can be completed on any device, increasing accessibility and encouraging participation from a wider range of respondents. This is particularly important for GNDEC's alumni surveys, where respondents may be located offcampus and prefer to complete surveys on mobile devices.

SURWAY represents a significant step forward in GNDEC's efforts to modernize its feedback collection and analysis processes. By offering a platform that is both flexible and powerful, SURWAY ensures that the institution can gather, analyze, and act on feedback data efficiently and effectively. With features like customizable survey forms, realtime data processing, and comprehensive reporting, SURWAY is set to become an indispensable tool for improving the quality of education and services at GNDEC.

The project's objectives—creating a userfriendly survey platform, enabling survey management and editing, and providing advanced data analysis tools—are all geared towards making feedback a more integral part of decisionmaking at GNDEC. Ultimately, SURWAY will empower the institution to continue its tradition of excellence by ensuring that it remains responsive to the needs and experiences of its students, faculty, and alumni.

2. SYSTEM REQUIREMENTS

The successful development and deployment of SURWAY – The Smart Way to Survey require specific software and hardware resources. These resources are necessary to ensure that the system functions efficiently and can handle the anticipated workload, both during the development phase and after implementation.

2.1 Software Requirements

The following software components are essential for the development and functioning of the SURWAY platform:

2.1.1. Operating System:

Development: Windows 10/11, macOS, or any Linux distribution.

Deployment: any cloud service with Linuxbased hosting capabilities.

2.1.2. Development Frameworks & Libraries:

- Node.js: Used for serverside programming, handling requests, and performing backend operations.
- React.js: For building the user interface (UI) of SURWAY. It provides a dynamic and responsive UI for users to create and manage surveys.
- Express.js: A minimal and flexible Node.js web application framework for backend APIs and routing.
- MongoDB: NoSQL database used for storing user data, survey questions, and responses in a flexible documentoriented format.
- Mongoose: A MongoDB object data modeling (ODM) library for Node.js, simplifying data validation, queries, and business logic.

2.1.3. Development Tools:

- Visual Studio Code (VS Code): A source code editor for development with extensive language support and version control integration.
- Postman: Tools for testing and simulating API requests during development.
- Git: Version control software for tracking changes in the codebase and collaboration during development.

2.1.4. Web Browsers:

- Google Chrome, Firefox, Microsoft Edge: Required for testing the platform's crossbrowser compatibility and performance.

2.1.5. Backend & API Management:

- JSON Web Tokens (JWT): For managing user authentication and session control within the application.
- Nodemailer: For sending email notifications, such as survey invites, reminders, or result summaries.

2.1.6. Version Control & Project Management:

- GitHub/GitLab/Bitbucket: Platforms used for hosting and managing the project's source code, tracking issues, and managing development progress.

2.1.7. Hosting & Deployment:

- GNDEC College Server: The SURWAY platform will be hosted on GNDEC's internal servers, ensuring that the application is secure, accessible, and managed locally. By using the college server, we maintain complete control over the system, data privacy, and compliance with institutional policies.
- Nginx: The web server will be set up with Nginx to handle reverse proxy and serve static files efficiently. This ensures that the platform remains responsive and scalable as the number of users increases.

2.1.8. Testing Frameworks:

- Mocha/Chai: For testing API endpoints and validating backend functionalities.

2.2 Hardware Requirements

The hardware requirements for developing, testing, and deploying SURWAY are outlined below. These requirements ensure that the platform can handle the necessary computations, data storage, and user traffic.

2.2.1. Development Machines:

- Processor: Intel Core i5 or higher (or AMD equivalent).

- RAM: Minimum 8 GB (16 GB recommended for smooth multitasking and running virtual environments).
- Storage: At least 250 GB SSD (Solid State Drive) for faster read/write speeds during development.

2.2.2. Server Requirements (for hosting the platform):

- Processor: 2 vCPUs or higher for managing multiple concurrent users.
- RAM: Minimum 4 GB (8 GB recommended for handling higher traffic).
- Storage: Minimum 50 GB SSD for storing application data, user uploads, and survey responses.
- Network: Highspeed internet connection for seamless connectivity and faster data transfers.
- Cloud Server: For scalability, cloud services such as AWS or Azure with autoscaling capabilities are recommended to handle variable user load.

2.2.3. Client Devices (Endusers accessing SURWAY):

- Processor: Any modern processor (dualcore or higher).
- RAM: Minimum 4 GB for smooth operation when using the platform.
- Network: Reliable internet connection for accessing the platform, filling out surveys, and submitting feedback in real time.

3. SOFTWARE REQUIREMENT ANALYSIS

3.1 Problem Definition

Educational institutions like GNDEC often face challenges in collecting and analyzing feedback from students, faculty, and alumni. Traditional paperbased feedback mechanisms or limited digital tools are inefficient, timeconsuming, and prone to errors in data collection and processing. Moreover, institutions need a robust system to design custom survey forms, manage responses, and generate reports for informed decisionmaking.

SURWAY – The Smart Way to Survey is designed to overcome these challenges by providing a dynamic platform for creating, managing, and analyzing surveys. The platform aims to simplify survey creation, enhance the user experience, and offer essential data analytics to help improve decisionmaking and performance evaluation at GNDEC. SURWAY's key objective is to digitize the feedback system, making it more flexible and adaptive to institutional needs.

3.2 Modules and Their Functionalities

SURWAY is built with several key modules, each designed to serve a distinct purpose in the overall workflow of survey management. The primary modules include:

3.2.1 User Authentication & Authorization Module

Description: This module handles the security aspects of the platform by managing user login, registration, and rolebased access to different features.

Functionalities:

- User Registration: New users (students, faculty, alumni) can create accounts by providing necessary information such as email and password.
- Login: Registered users can log into the system using their credentials.
- Rolebased Access: Different roles (e.g.,survey creator, respondent) have different levels of access to the platform features.
- Session Management: Ensures that users remain logged in securely during their session and are logged out after a period of inactivity.

3.2.2 Survey Creation Module

Description: This is the core module where users (administrators or authorized staff) can create and customize survey forms using predefined components.

Functionalities:

- Question Types: Users can add multiple types of questions such as multiplechoice, dropdowns, checkboxes, rating scales, and openended questions.
- Survey Logic: Implement conditional branching, where the next question depends on the respondent's previous answer.
- DragandDrop Interface: Allows easy arrangement and reordering of questions within the survey.

3.2.3 Survey Management Module

Description: This module allows authorized users to manage existing surveys, including editing, copying, or deleting surveys.

Functionalities:

- Edit Survey: Modify existing surveys, add or remove questions, and change settings such as deadlines for submission.
- Activate/Deactivate Survey: Enable or disable surveys based on the feedback collection schedule.
- Clone Survey: Create a copy of an existing survey to make small modifications and reuse for new audiences.
- Set Survey Expiry: Define start and end dates for when a survey is open for responses.

3.2.4 Response Collection Module

Description: This module manages the process of distributing surveys to respondents and collecting their answers.

Functionalities:

- Survey Distribution: Share survey links via email or embed them on websites and portals.
- Anonymous Responses: Allow for anonymous feedback where user identity remains confidential, depending on the survey settings.
- Autosave Feature: Responses are autosaved to ensure no data is lost if a session is interrupted.
- Progress Tracking: Allows respondents to track their progress through multipage surveys.

3.2.5 Data Analysis & Reporting Module

Description: This module processes the collected survey responses and provides realtime reporting and analysis tools for decisionmaking.

Functionalities:

- Realtime Data Processing: View results as they come in, providing immediate insights.
- Data Visualization: Use charts, graphs, and tables to represent response data in a userfriendly format.
- Export Data: Export results to various formats like CSV, Excel, or PDF for further analysis.
- Generate Reports: Automatically generate summary reports, which include key statistics such as response rates, average scores, and respondent demographics.
- Custom Filters: Filter results by date range, specific respondents, or based on answers to certain questions.

4. SOFTWARE DESIGN

4.1 Creating a new survey form

The process of creating a new survey form begins with the user interacting with a Survey Creator Interface. The user selects predefined question types (e.g., multiplechoice, text entry) and customizes them according to their requirements. The system dynamically generates a JSON structure representing the survey's content, including questions, options, and logic. This JSON is then stored in the backend using API calls, allowing the survey to be saved, previewed, or edited later.

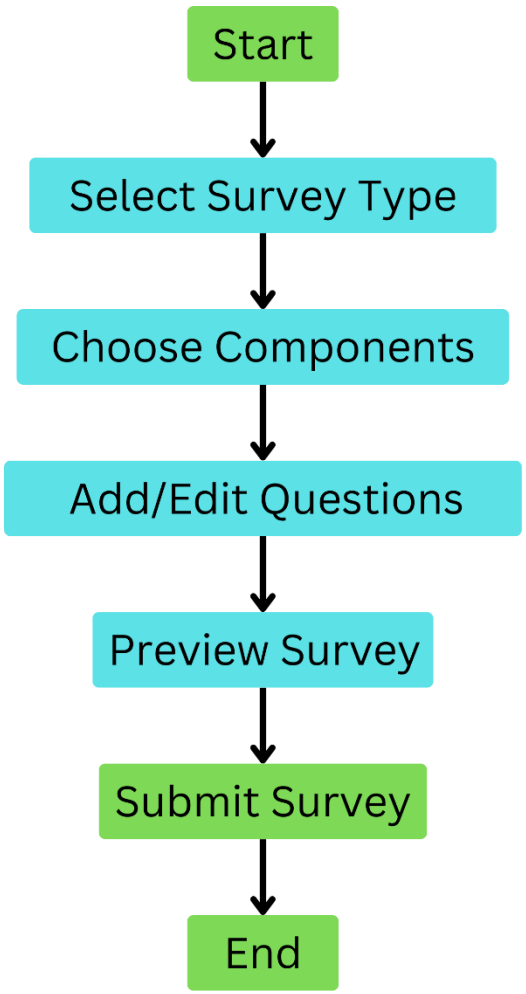

Fig 1.1 – *flow diagram of creating a new survey form.*

4.2 Editing and managing survey forms

Editing and managing existing survey forms involves retrieving the survey's JSON data from the database. The user can modify questions, add or remove sections, and update the survey logic. The modified JSON structure is autosaved at regular intervals or upon certain actions (like submitting a change). The system ensures that version control is maintained to avoid data loss and provide rollback options for previous survey versions.

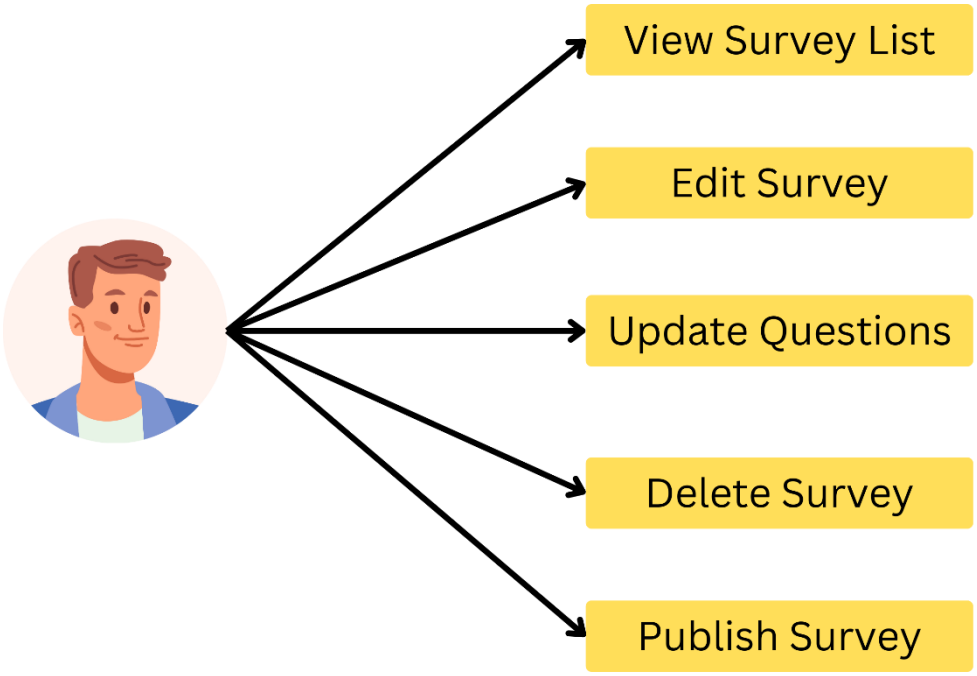

Fig 1.2 – *actionable objectives by the user.*

4.3 User registration

The user registration process starts with users submitting basic credentials like a username, email, and password. The password is hashed using secure algorithms before being saved to the database. After verifying that the email is unique, the user information is stored in the database with the isVerified flag set to false. The system then sends an OTP to the user's email for verification, and once verified, the user can access the platform's features.

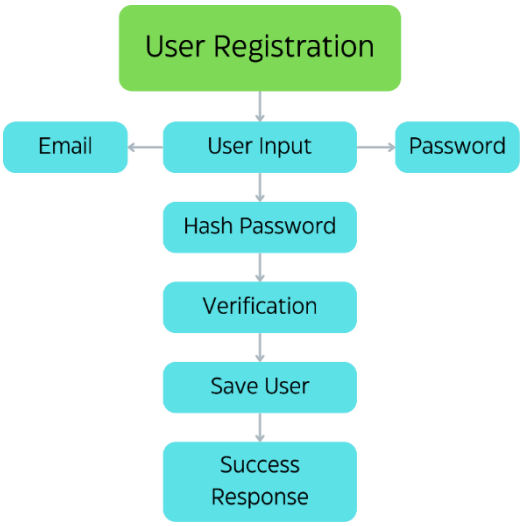

Fig 1.3 – flow diagram of user registration.

4.4 User login

During the login process, users provide their email and password. The system checks if the user exists in the database and compares the provided password with the stored hashed password. If the credentials are valid and the user is verified, a session is created and maintained using JWT (JSON Web Token), which is stored as a cookie for future authenticated requests. Unverified users are prompted to complete their email verification before gaining access.

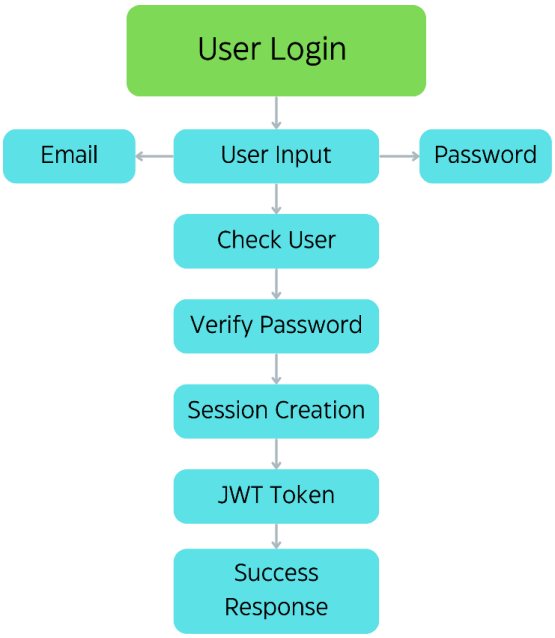

Fig 1.4 – flow diagram of user registration.

4.5 Survey creator instance

The Survey Creator Instance is the core tool used to design surveys. It initializes with a blank JSON structure and allows the user to add questions, customize their properties, and manage survey logic. The creator interface interacts with a backend API to automatically save changes as users build their surveys. This instance ensures flexibility, allowing users to preview the survey in realtime, while handling different question types, branching logic, and validations.

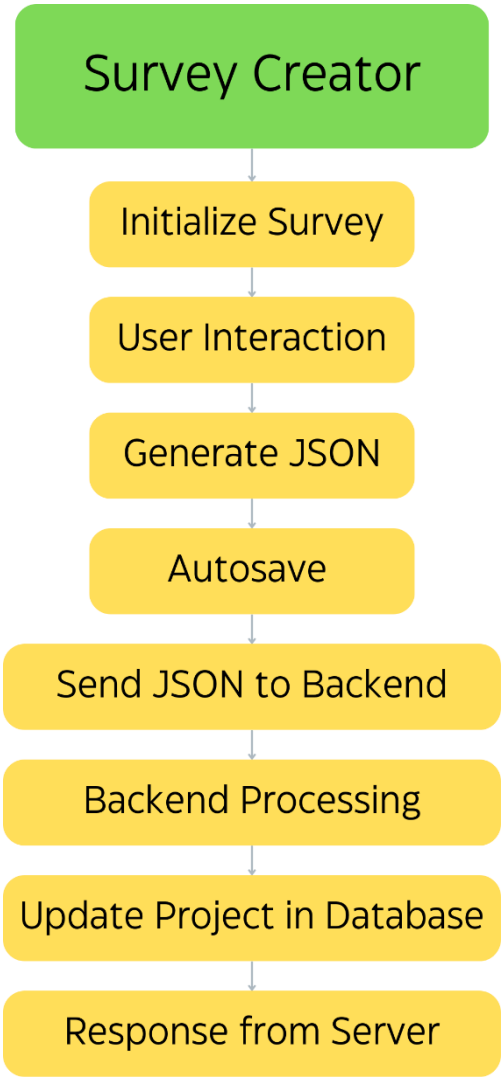

Fig 1.5 – flow diagram showing how form is saved in database.

5. TESTING MODULE

In order to ensure the successful implementation and operation of the SURWAY platform, rigorous testing is required throughout the development process. Testing will validate the platform's functionality, usability, and performance, ensuring that it meets the needs of its users in learning institutions. Given the platform’s role in handling academic surveys and realtime data analysis, it is crucial that each component of SURWAY is thoroughly tested to prevent issues that could hinder data collection, processing, and reporting. This proposal outlines the key testing techniques that will be employed, along with a list of relevant test cases designed to ensure the system operates smoothly and efficiently.

5.1 Unit Testing

- Purpose: To test individual components of the system in isolation.
- Scope: Each small part of the SURWAY platform, such as question components, survey form generation, data validation, and response handling, will be tested separately to ensure they function as intended.
- Tools: Jest, Mocha, or similar testing frameworks can be used to perform unit tests.

5.2 Integration Testing

- Purpose: To test how different modules interact with each other.
- Scope: Once individual components are verified, integration tests will focus on the interactions between various parts of the system, such as how the survey editor interacts with the data storage system or how survey results are passed to the reporting feature.
- Tools: Selenium, Cypress, or other frameworks capable of simulating realworld interactions between components.

5.3 Functional Testing

- Purpose: To ensure the system functions according to the specified requirements.
- Scope: Functional tests will focus on the userfacing features of the platform, ensuring that tasks like creating, editing, and submitting surveys work correctly. These tests will cover the full range of expected user interactions, checking for expected outputs based on predefined inputs.
- Tools: Manual testing or automation tools like TestComplete can be used to execute functional tests.

5.4 Usability Testing

- Purpose: To evaluate how userfriendly and intuitive the system is for its intended users (students, faculty members, administrators).
- Scope: Usability testing will involve getting actual users to interact with the system and gather feedback on ease of use, navigation, and user experience. The goal is to ensure that the platform is accessible and easy to operate for users of varying technical backgrounds.
- Tools: Tools like Crazy Egg or user testing sessions can be employed to analyze user behaviour.

5.5 Performance Testing

- Purpose: To assess the system's speed, stability, and scalability under normal and peak load conditions.
- Scope: This includes testing the platform with a large number of concurrent users (e.g., during a large academic survey) to ensure it can handle heavy traffic without slowing down or crashing. Response time, resource usage, and load time will be monitored.
- Tools: Apache JMeter, LoadRunner, or Gatling can be used to simulate user traffic and evaluate performance metrics.

5.6 Security Testing

- Purpose: To identify and mitigate security vulnerabilities that could jeopardize the system or its data.
- Scope: Security testing will cover areas such as data encryption, user authentication, and rolebased access control. Special attention will be given to ensuring that sensitive data (e.g., student feedback) is stored securely, and unauthorized access is prevented.
- Tools: OWASP ZAP, Burp Suite, or similar tools will be used to identify and fix security loopholes.

To ensure the SURWAY platform performs as expected under various conditions, a comprehensive set of test cases will be developed and executed. These test cases will cover the core functionalities of the system, such as creating and editing surveys, submitting responses, and analyzing data. Additionally, other important aspects such as security, performance under heavy loads, and user experience across different devices will also be tested. By using a range of test scenarios, the goal is to identify and resolve any potential issues, thereby ensuring a smooth, secure, and userfriendly platform that meets the requirements of both students and faculty members.

5.7 Create Survey

- Objective: Ensure that users can create a new survey by selecting predefined components.
- Test Steps:
  - Log in as an authorized user.
  - Open the "Create Survey" page.
  - Select various question types and arrange them in the survey form.
  - Click "Create Survey" and save.
- Expected Outcome: The system should successfully create and save the survey with all selected questions.

5.8 Edit Survey

- Objective: Verify that users can modify an existing survey and that the changes are saved.
- Test Steps:
  - Select an existing survey.
  - Change one or more questions or options.
  - Save the changes.
- Expected Outcome: The changes should reflect in the database, and the survey should update accordingly.

5.9 Submit Survey Response

- Objective: Ensure that users can successfully fill out and submit a survey.
- Test Steps:
  - Select an available survey.
  - Complete all required fields.
  - Submit the form.
- Expected Outcome: The survey should submit successfully, and the responses should be stored in the system.

5.10 Data Analysis

- Objective: Test the accuracy of the realtime data analysis feature.
- Test Steps:
  - Submit multiple responses for a survey.
  - View the analysis results in the admin panel.
- Expected Outcome: The system should correctly analyze the data and provide relevant insights (charts, graphs, or statistical summaries).

5.11 User Authentication

- Objective: Verify that users can log in based on their roles (admin, faculty, student).
- Test Steps:
  - Attempt login with various roles.
  - Check access permissions for each role.
- Expected Outcome: Users should have access to the appropriate sections of the platform based on their role.

5.12 Form Validation

- Objective: Ensure that all survey fields are properly validated.
- Test Steps:
  - Create a survey with mandatory fields.
  - Submit the form without filling out all required fields.
- Expected Outcome: The system should display error messages for missing or incorrectly formatted inputs and prevent submission.

5.13 Survey Response Storage

- Objective: Ensure survey responses are correctly stored in the database.
- Test Steps:
  - Submit responses for a survey.
  - Check the database or admin panel for stored responses.
- Expected Outcome: Responses should be accurately stored and retrievable.

5.14 Load Test

- Objective: Evaluate system performance under heavy load.
- Test Steps:
  - Simulate multiple users submitting surveys at the same time.
  - Monitor system performance (response time, load handling).
- Expected Outcome: The system should remain responsive and handle a large number of submissions without crashing.

5.15 Data Export

- Objective: Ensure that users can export survey results to external formats like CSV or Excel.
- Test Steps:
  - View survey results.
  - Select export options (CSV, Excel).

- Expected Outcome: The system should successfully export the data in the selected format without errors.

5.16 UI Consistency

- Objective: Ensure that the user interface is consistent across different devices and browsers.
- Test Steps:
  - Access the platform using different devices (desktop, tablet, mobile) and browsers (Chrome, Firefox, Edge).
  - Navigate through various sections.
- Expected Outcome: The UI should display correctly and maintain functionality across different environments.

6. PERFORMANCE OF THE PROJECT DEVELOPED

The SURWAY GNDEC project has made significant progress in its development, addressing the core objectives of survey creation, management, and analysis. The performance of the project can be evaluated based on several key factors, which include the platform's usability, responsiveness, scalability, data accuracy, and security. This chapter outlines the current performance metrics, challenges encountered, and improvements made throughout the development process.

6.1.Usability and User Experience

- **Intuitive User Interface:**
  - The survey creation interface is highly intuitive, allowing users to easily drag and drop predefined components to build surveys. This has significantly reduced the learning curve for new users, including faculty and administrative staff.
  - Users without technical expertise can effortlessly create complex surveys, making the platform accessible to a wide range of users.
- **Realtime Survey Preview:**
  - One of the key features is the real time survey preview option, which provides users with an immediate view of how the survey will look. This improves user satisfaction and reduces errors during survey creation.
- **Customization and Flexibility:**
  - The platform offers robust customization options, allowing users to create personalized surveys by modifying question types, layouts, and branching logic.
  - Conditional logic enhances user experience by ensuring that respondents only see relevant questions, improving the overall survey engagement.

6.2 Performance of Survey Creation Module

- **Dynamic Survey Generation:**
  - The system efficiently generates a JSON structure that reflects the entire survey, including questions, options, and logic. This JSON is stored in the backend, which ensures that survey data is well structured and easily retrievable for future editing and management.
- **Predefined Question Components:**
  - The drag and drop functionality for adding various question types (e.g., multiple choice, rating scales, text input) works seamlessly, with no significant delays or bugs encountered during testing.

- **Efficiency in Editing and Saving Surveys:**
  - The backend APIs handle survey creation and saving efficiently, allowing users to save their progress and return to surveys later for further editing. The use of APIs ensures a smooth interaction between the frontend and backend components.

6.3 Responsiveness and Compatibility

- **Device Compatibility:**
  - The SURWAY platform has been designed with responsive web design principles, ensuring that it functions smoothly across devices such as desktops, tablets, and smartphones.
  - Surveys created on the platform are fully mobile compatible, which is critical for alumni feedback, as many respondents are likely to complete surveys on mobile devices.
- **Responsive Loading Times:**
  - Throughout testing, the platform has demonstrated efficient loading times across different devices and browsers. Surveys load quickly without performance lag, even when populated with complex question logic or large datasets.

6.4 Scalability and Database Performance

- **Efficient Data Storage:**
  - MongoDB has been successfully integrated as the database, handling large sets of survey data and user credentials without noticeable performance issues. The document based model ensures scalability as the number of surveys and respondents increases.
- **API Performance:**
  - Backend performance is robust, with API calls completing within acceptable time limits. This ensures a fast user experience, particularly when saving surveys, collecting responses, or generating reports.
- **Handling Large Datasets:**
  - The platform has been tested to handle large numbers of respondents and survey responses. Even with hundreds of submissions, the system continues to perform efficiently without delays in data retrieval or reporting.

6.5 Security and Data Integrity

- **User Authentication & Session Management:**
  - Security is a key focus area, and the platform successfully implements user authentication using JWT (JSON Web Tokens). Passwords are hashed using strong algorithms, ensuring that sensitive data is protected.

- Session management works as expected, maintaining user login status across sessions while securely logging out inactive users after a set period.

- **Data Encryption:**
  - All sensitive data, such as user credentials and survey responses, is securely stored in the database using encryption protocols, ensuring the integrity and confidentiality of data.
- **OTP Verification:**
  - The email verification process using OTP (One Time Password) is fully functional, adding an additional layer of security for new users. This ensures that only verified users can access and create surveys.

6.6 Data Management and Performance

- **JSON Editor & Form Preview:**
  - Dynamic Form Creation: The JSON editor dynamically builds and updates the survey structure in real time, allowing users to easily customize questions and options.
  - Instant Form Preview: Users can preview the survey as they build it, with real time rendering that reflects changes instantly for a seamless survey creation experience.
- **Data Storage & Database Performance:**
  - Efficient Data Saving: Survey data, including questions and responses, is saved in the database through optimized API calls, ensuring quick storage and retrieval.
  - Data Integrity: The system ensures data integrity during form submissions, maintaining a robust backend for handling high volumes of data efficiently.

7. OUTPUT SCREENS

7.1 UI of User Authentication

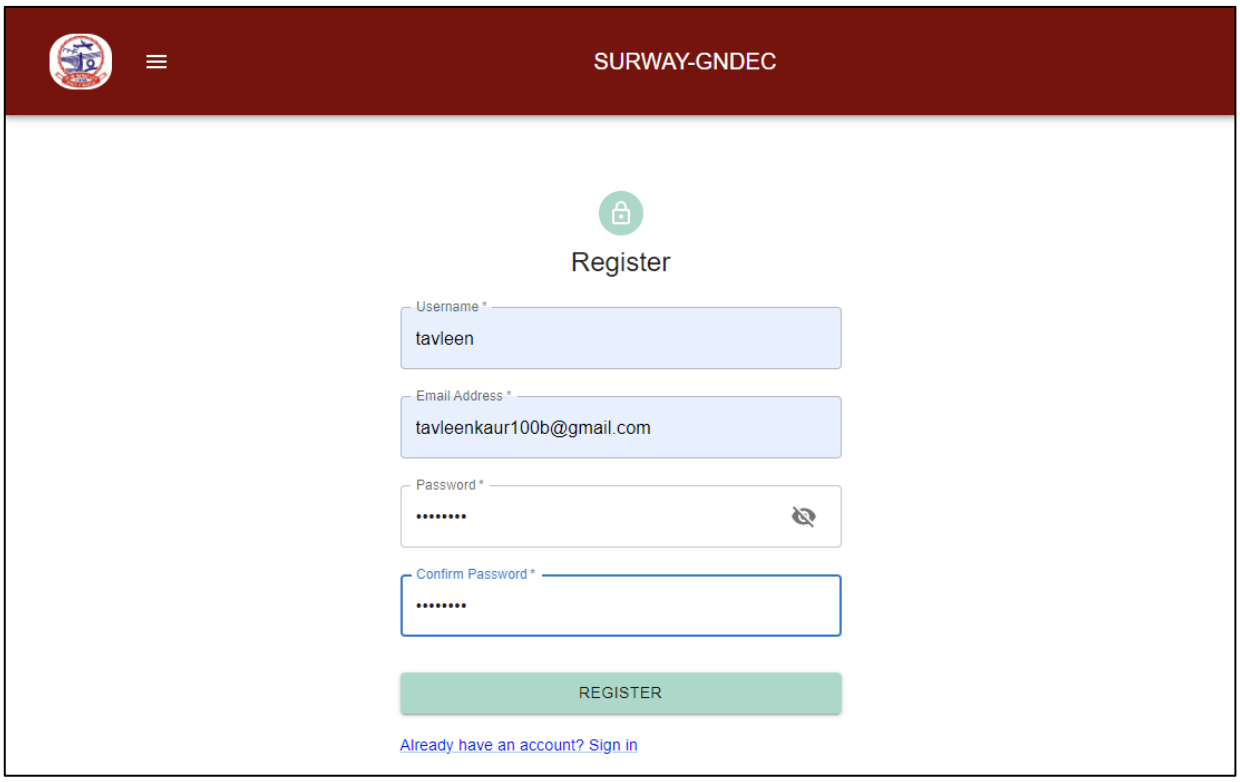

Fig. 7.1 – *Register Page*

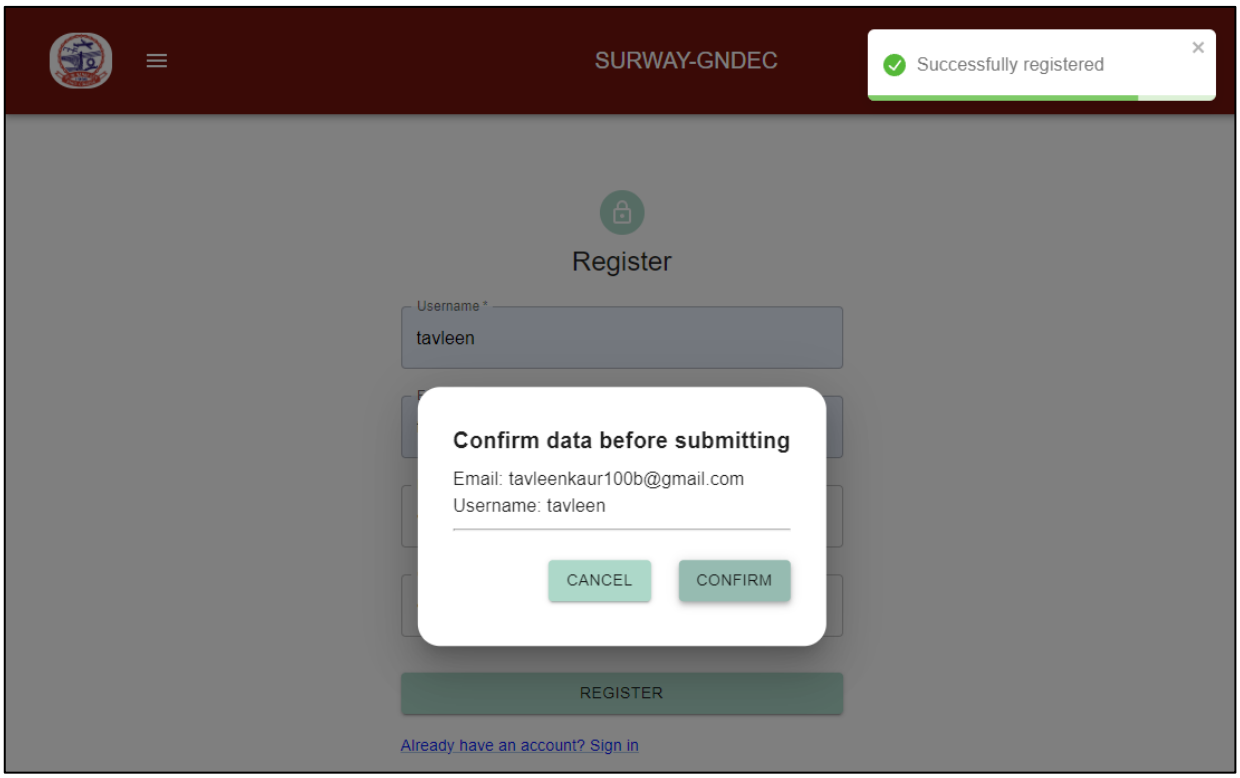

Fig. 7.2 – *Confirm Details before Submitting*

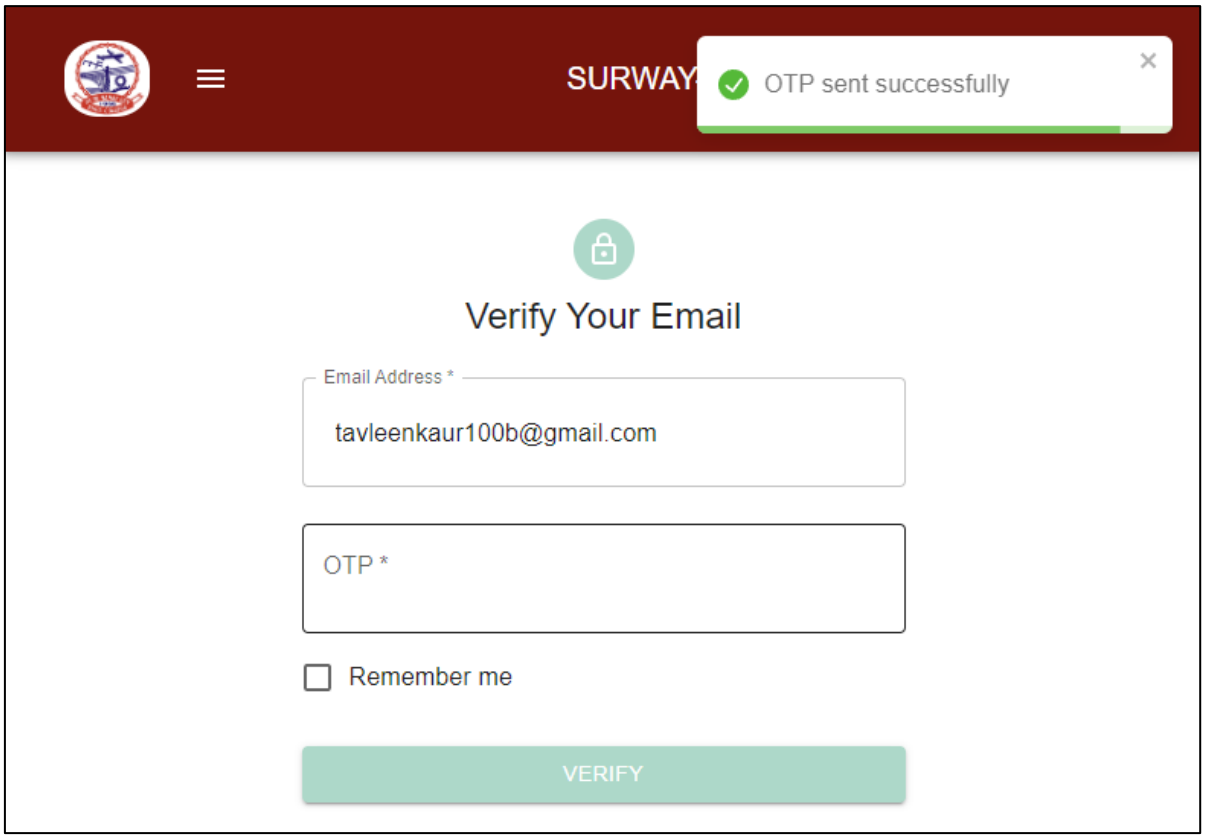

Fig. 7.3 – *OTP sent for Email Verification*

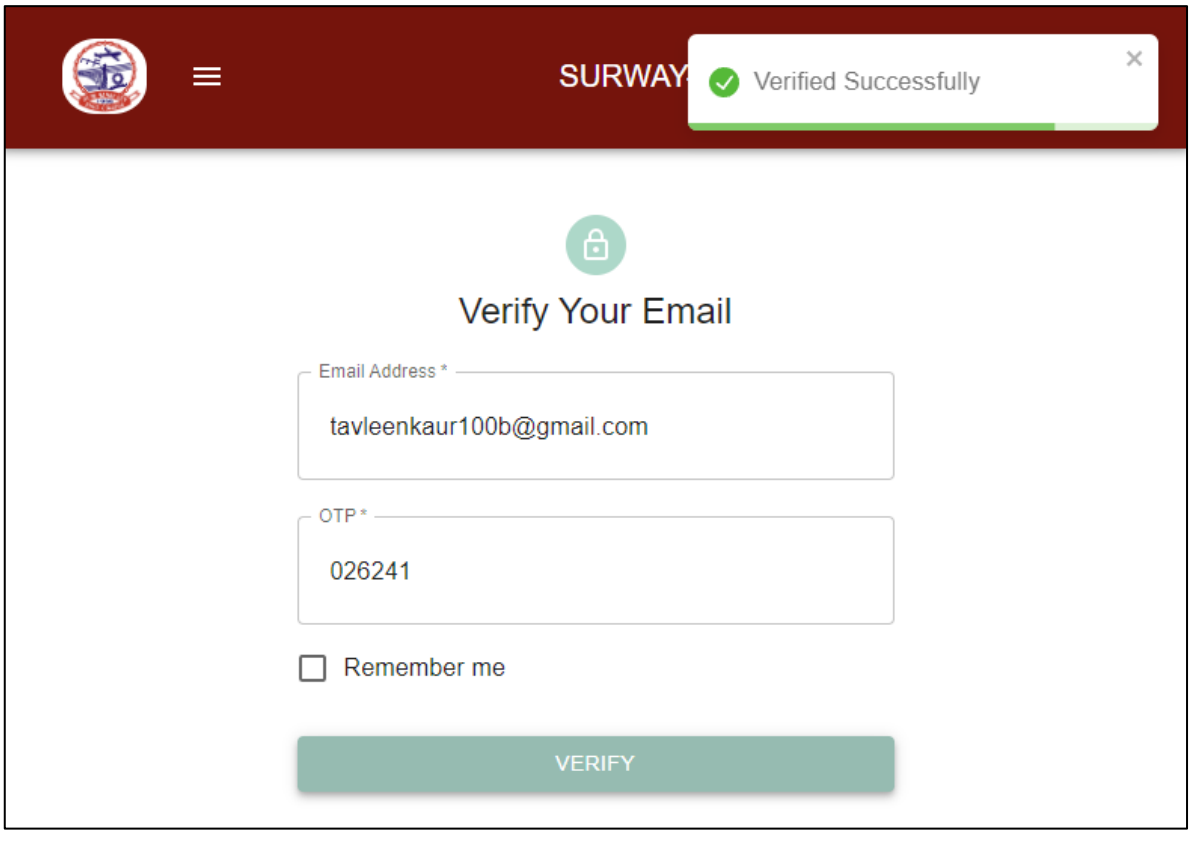

Fig. 7.4 – *Enter OTP for Email Verification*

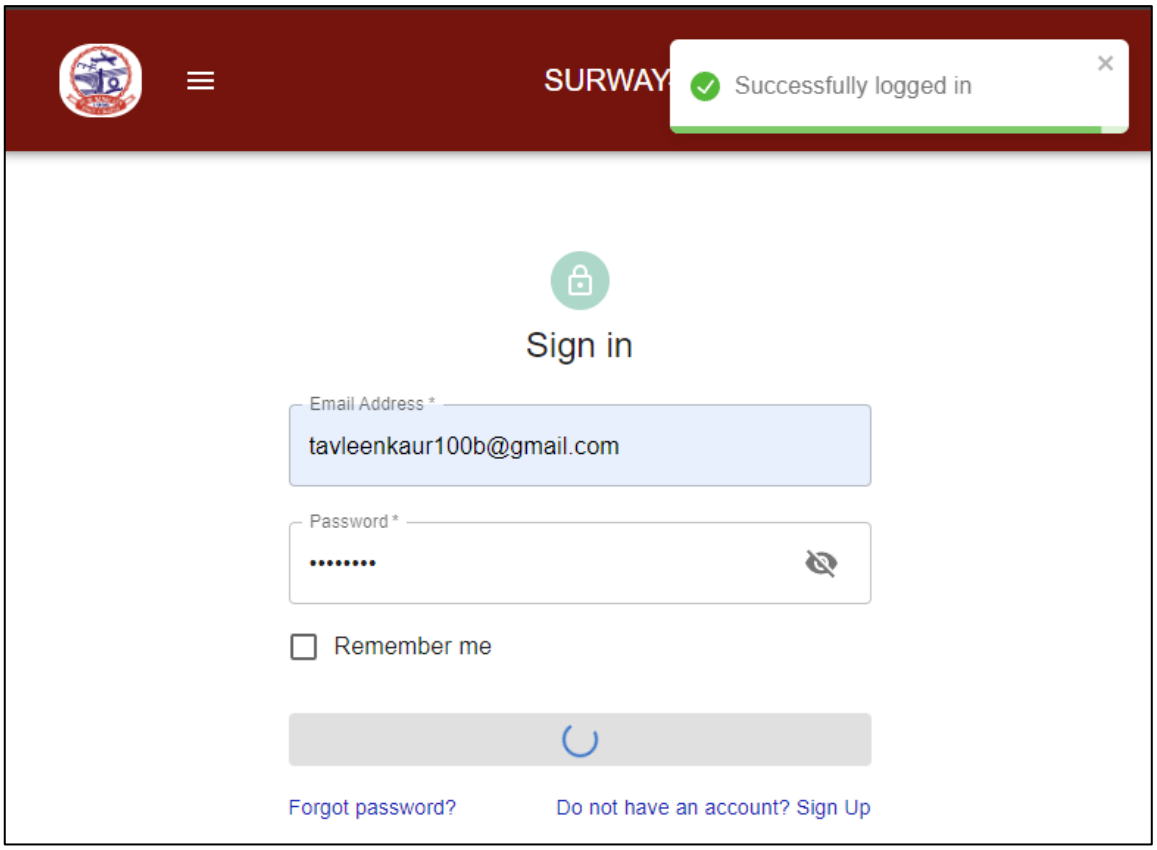

Fig. 7.5 – *Sign In Page*

7.2 User Interface of Projects Page (Home Page)

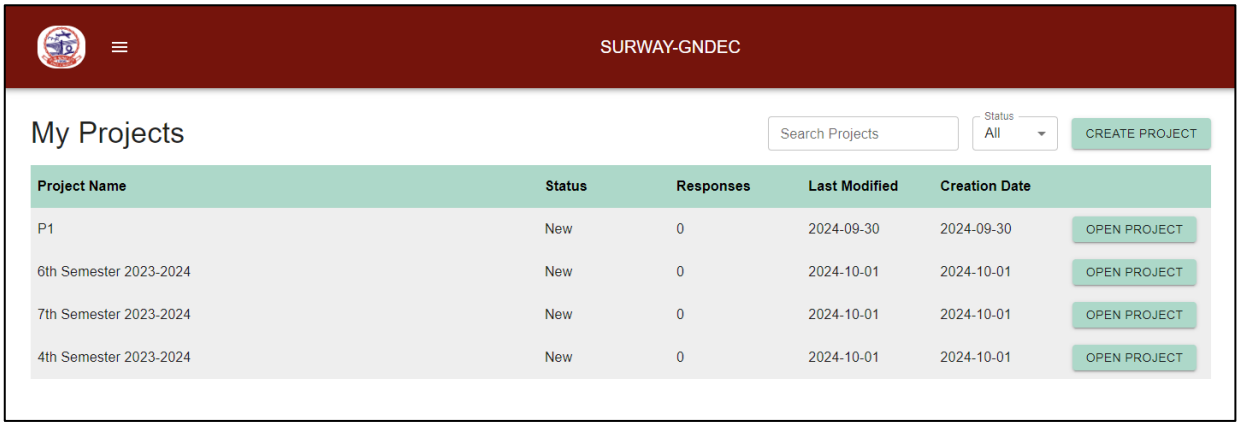

Fig. 7.6 – *Projects Page Interface (Home Page)*

7.3 User Interface of Survey Builder

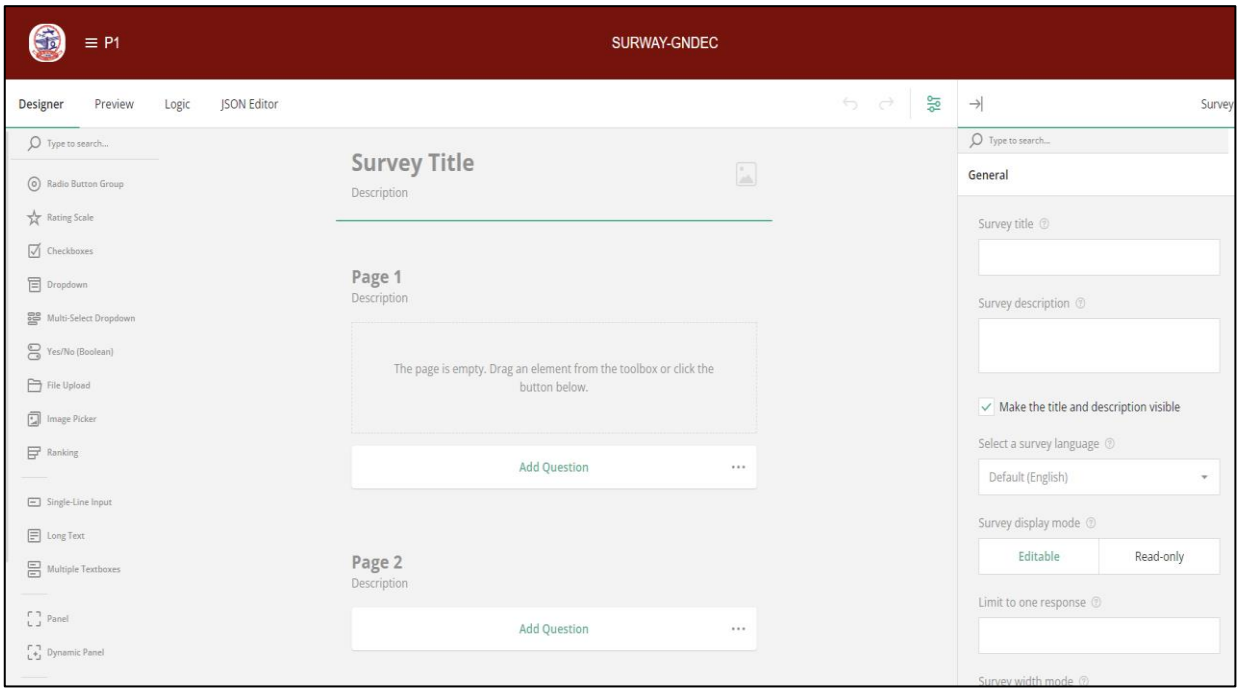

Fig. 7.7 – *UI of Survey Builder Page*

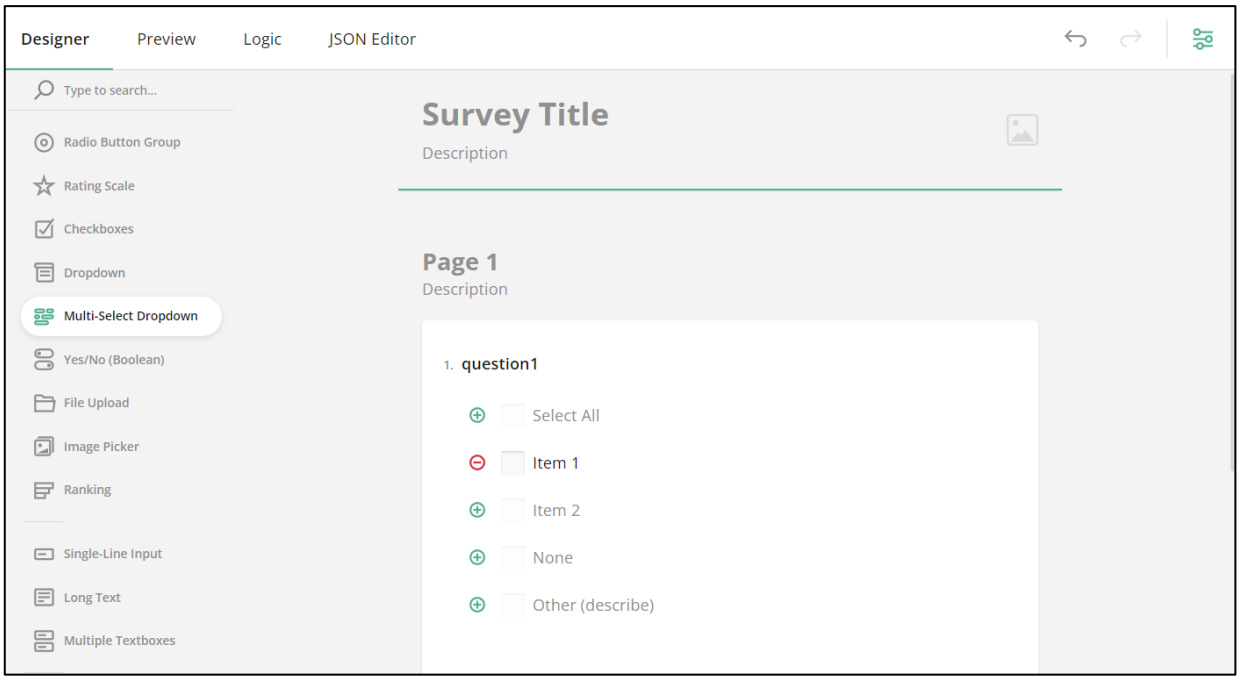

Fig. 7.8 – *Drag and Drop Menu (Sidebar)*

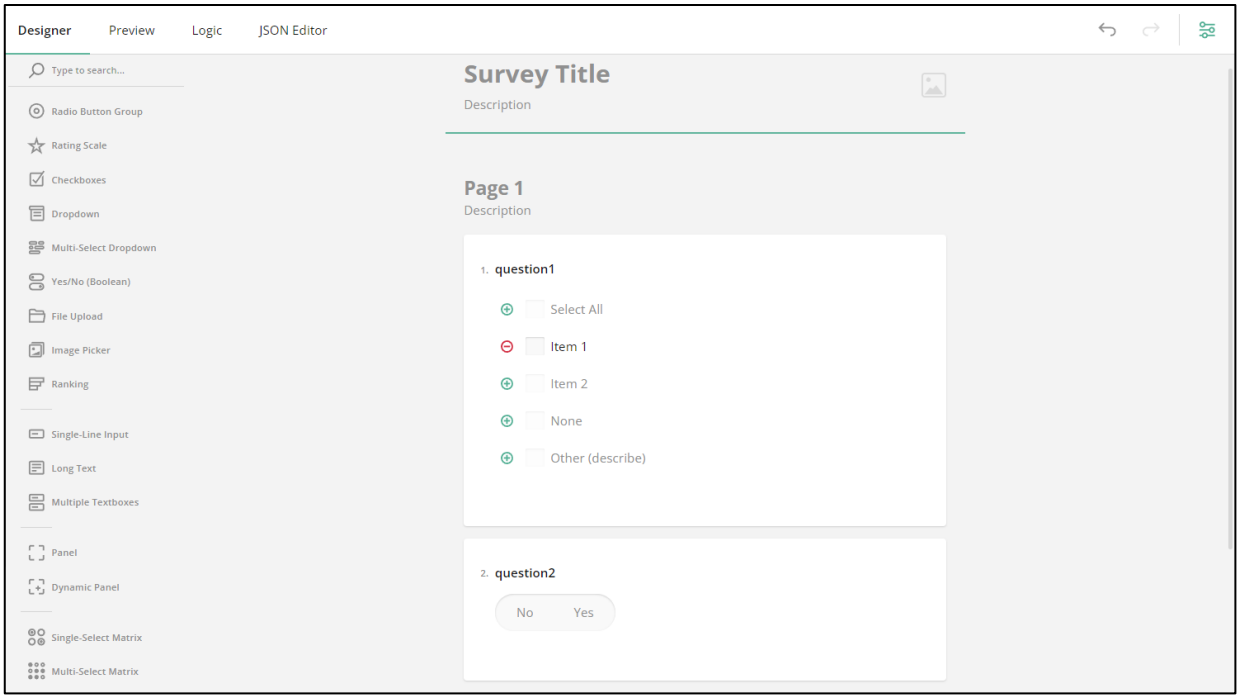

Fig. 7.9 – *An example of Survey Form*

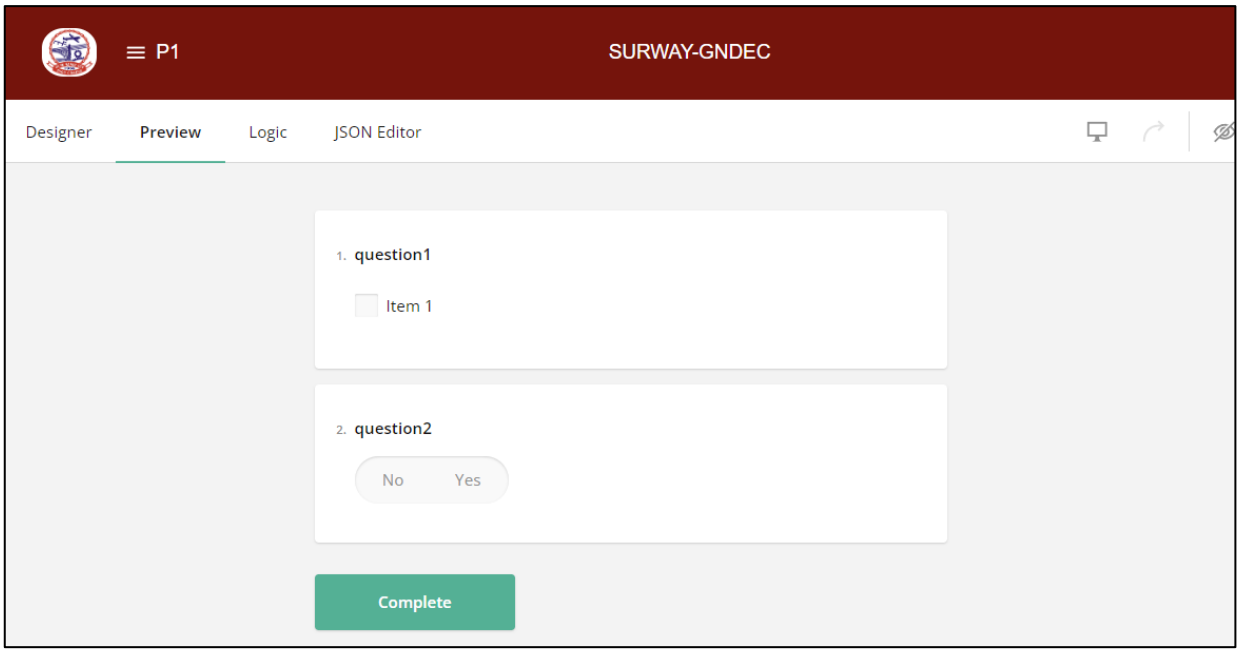

Fig. 7.10 – *Preview the Created Form*

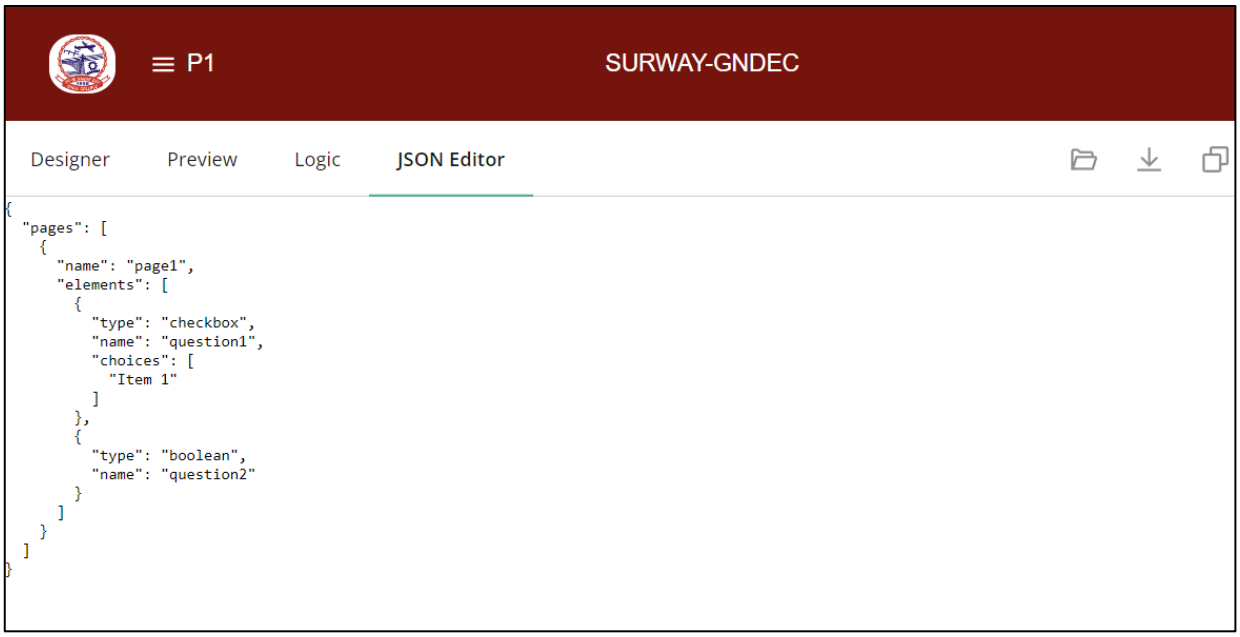

Fig. 7.11 – *JSON Editor of the Created Form*

8. REFERENCES

[1] L. Yan *et al.*, "Evidence-based multimodal learning analytics for feedback and reflection in collaborative learning," *British Journal of Educational Technology*, Jun. 2024, doi: 10.1111/bjet.13498.

[2] H. Yildiz Durak, "Impact of MLLA feedback system on learners' academic performance, engagement and behavioral patterns in online collaborative learning environments: A lag sequential analysis and Markov chain approach," *Educ Inf Technol (Dordr)*, Jul. 2024, doi: 10.1007/s10639024129119.

[3] P. Ren, L. Yang, and F. Luo, "Automatic scoring of student feedback for teaching evaluation based on aspectlevel sentiment analysis," *Educ Inf Technol (Dordr)*, vol. 28, no. 1, pp. 797–814, Jan. 2023, doi: 10.1007/s1063902211151z.

[4] M. V. Jongsma, D. J. Scholten, J. E. van MuijlwijkKoezen, and M. Meeter, "Online Versus Offline Peer Feedback in Higher Education: A MetaAnalysis," *Journal of Educational Computing Research*, vol. 61, no. 2, pp. 329–354, Apr. 2023, doi: 10.1177/07356331221114181.