

SURWAY - THE SMART WAY TO SURVEY

MAJOR PROJECT REPORT

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE
AWARD OF THE DEGREE OF

BACHELOR OF TECHNOLOGY
(Computer Science and Engineering)



Submitted By :

Saksham Kohli(2104176)
Simarpreet Kaur (2104193)
Tavleen Kaur (2104209)

Submitted To:

Prof. Diana Nagpal
Assistant Professor

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GURU NANAK DEV ENGINEERING COLLEGE
LUDHIANA, 141006
(November, 2024)**

ABSTRACT

SURWAY-GNDEC is a complete survey management system created in an attempt to improve and address obstacles related, especially to feedback collection in educational institutions such as GNDEC Ludhiana. The project seeks to develop a strong system for designing, disseminating and collating survey responses which would assist the administrators, faculty and students with the realization of valuable feedback for academic and institutional improvement.

At the core of SURWAY-GNDEC is a user-centered application which provides administrators with the essential building blocks to create highly customized surveys that solicit feedback, for instance, course evaluation surveys, or Alumni surveys. The system comes with various question types, as well as logic and branching features that provide a rich and pleasing experience allowing administrators to source for specific and useful information in a controlled fashion.

SURWAY-GNDEC also implements advanced analytical capabilities to support its users in making data informed decisions and visualization techniques to enhance interpretation of information. The platform has been designed in such a way that its users do not only respond to surveys but also have the capability of analyzing the trends in responses in order to provide recommendations to enhance the quality of education rendered at and the services offered by GNDEC.

It is implemented on top of the MERN stack which ensures decent scaling and performance while maintaining security. Its design is so user friendly that people with different levels of relevant skills can use it which ultimately increases the user-friendliness and engagement. In addition, the platform has a comprehensive data management and protection system to ensure safety of user information.

Overall, the SURWAY-GNDEC is a very advanced and ingenious system for collecting surveys and feedbacks from the users of the institution, which helps in developing the institution and supports the mission of GNDEC to provide quality education and manage the institution effectively.

ACKNOWLEDGEMENT

We are highly grateful to the Dr. Sehijpal Singh Khangura, Principal, Guru Nanak Dev Engineering College (GNDEC), Ludhiana, for providing this opportunity to carry out the major project work.

The constant guidance and encouragement received from Dr. Kiran Jyoti, H.O.D. CSE Department, GNDEC Ludhiana has been of great help in carrying out the project work and is acknowledged with reverential thanks.

We would like to express a deep sense of gratitude and thanks profusely to Prof. Diana Nagpal without her wise counsel and able guidance, it would have been impossible to complete the project in this manner.

We express gratitude to other faculty members of computer science and engineering department of GNDEC for their intellectual support throughout the course of this work.

Saksham Kohli (2104176)

Simarpreet Kaur (2104193)

Tavleen Kaur (2104209)

LIST OF FIGURES

Fig No.	Figure Description	Page No.
2.1	Agile methodology used	26
3.1	Flow diagram of creating a new survey form.	35
3.2	Flow diagram of user registration.	36
3.3	Flow diagram of Surway	37
4.1	Test Cases	52
5.1	Landing Page of Surway	60
5.2	Sign Up Page of Surway	61
5.3	Login Page of Surway	62
5.4	Verification Page	63
5.5	An example of Survey Builder Form	64
5.6	Drag and Drop Menu (Sidebar)	65

5.7	An example of Survey Form	65
5.8	Preview the Created Form	66
5.9	JSON Editor of the Created Form	67
5.10	Results and Analysis	68
5.11	MongoDB Atlas Database	69
5.12	Database Collections	69
5.13	Response and Feedback Forms Data (Part1)	71
5.14	Response and Feedback Forms Data (Part2)	72
5.15	Response and Feedback Forms Data (Part3)	72
5.16	Project Section Data Storage	73

TABLE OF CONTENTS

Contents	Page No.
<i>Abstract</i>	<i>i</i>
<i>Acknowledgement</i>	<i>ii</i>
<i>List of Figures</i>	<i>iii-iv</i>
<i>Table of Contents</i>	<i>v-vi</i>
Chapter 1: Introduction	1-11
1.1 Introduction to Project	1-4
1.2 Project Category	4-5
1.3 Problem Formulation	5-6
1.4 Identification/Recognition of Need	6-7
1.5 Existing System	8
1.6 Objectives	9
1.7 Proposed System	10
1.8 Unique features of the proposed system	10-11
Chapter 2: Requirement Analysis and System Specification	12-26
2.1 Feasibility study	12-18
2.2 Software Requirement Specification Document	18-23
2.3 SDLC model	23-26
Chapter 3: System Design	27-41
3.1 Design Approach (Object oriented)	27-31
3.2 Detail Design	33-37
3.3 User-Interface Design Diagram	37-39
3.4 Methodology	39-41
Chapter 4: Implementation and Testing	42-53

4.1 Introduction to Languages, IDE's, Tools and Technologies used for Project work	42-44
4.2 Algorithm/Pseudocode used	44-48
4.3 Testing Techniques	48-51
4.4 Test Cases designed for the project work	51-53
Chapter 5: Results and Discussions	54-74
5.1 User Interface Representation	54-56
5.1.1 Brief Description of Various Modules of the system	57-59
5.2 Snapshots of system with brief detail of each and discussion.	59-68
5.3 Back Ends Representation with brief description	69-74
Chapter 6: Conclusion and Future Scope	75-79
6.1 Conclusion	75-76
6.2 Future Scope	77-79
References	80

CHAPTER – 1 INTRODUCTION

1.1 Introduction to Project

In the dynamic domain of education, academic institutions are always looking for ways to improve their services, educational provision, and operational efficiency. An important part of this effort is through the collection of feedback—a process that allows stakeholders the opportunity to voice their views and provide invaluable insights to propel the advancement and growth of an institution. Feedback is both an evaluation of the effectiveness of different programs and a very helpful tool for further facilitations. In addition, the level of complexity and information-related features of modern educational organizations rises incredibly, and therefore there was indeed quite a large shift in the methods and expectations of gathering feedback.

Guru Nanak Dev Engineering College (GNDEC), Ludhiana has always recognized the importance of feedback towards enhancing academic and administrative affairs. Traditionally, feedback collection at GNDEC was done through a mode of paper surveys or fixed digital questionnaires. Although these methods were capable of performing their desired task, they also faced various challenges. The process of analyzing the feedback was time-consuming, prone to human error and often could not provide the kind of insights needed for proper decision-making. Moreover, feedback mechanisms usually do not have real-time analysis capabilities, which is a very important aspect for incorporating instantaneous changes based on the most current feedback. These limitations highlight a great need for an automated, comprehensive, and flexible system designed to support specific needs for feedback collection at an institution like the GNDEC.

This need gives rise to **SURWAY: The Smart Way to Survey**-an innovative platform for the management of a survey strictly tailored to meet the feedback needs of GNDEC. The concept for SURWAY is that of a comprehensive, adaptive, and user-friendly platform for gathering and

analyzing feedback from students, faculty, alumni, and staff. Built with the MERN stack: MongoDB, Express.js, React.js, and Node.js, SURWAY brings together flexibility characteristic of modern web applications and solidity required in an academic environment. The platform is easy to use for creating and managing customized surveys, analyzing results immediately, and generating highly informative reports that can be used for the improvement of educational programs as well as administrative functions.

With SURWAY, the answers can certainly go beyond the normal ways of survey development by offering technical attributes that allow not just more in-depth insights but even greater flexibility. It's flexible enough and suited to accommodate a range of question formats, from multiple-choice items and rating scales to matrix grids and open-ended questions, offering many ways of getting all-inclusive feedback. Using the drag-and-drop functionality, SURWAY would enable creating surveys in a very swift and intuitive manner, requiring minimal technical knowledge. More so, an approachable interface integrated with robust backend functionalities places SURWAY as an effective and easily accessible instrument for managing feedback.

Thus, it allows the real-time processing of feedback outcome through data-centric architecture, providing administrators, faculty, and other users with the ability to quickly respond to emerging trends and insights. This particular aspect is of essential value in educational contexts, because the ability to adapt and respond to the feedback received during an active academic term can make a huge difference in the quality of the learning experience. Through real-time analysis, SURWAY enables GNDEC to make prompt modifications and enhancements, guaranteeing that the feedback data stays pertinent and usable.

More importantly, it is to the institution that SURWAY will bring about greater significance. As an institution committed to excellence, GNDEC aims at creating an environment responsive and supportive to its stakeholders. A feedback system through easy real-time data analysis, easy

customization of surveys, and extensive reporting will therefore enable GNDEC to respond accordingly in order to continually improve its educational and administrative processes. This is very important especially considering the institution's diverse stakeholder population that includes current students, faculty, alumni, and administrative personnel. Each of these cohorts has different opinions and expectations for providing feedback. SURWAY's flexible model ensures that the needs of each of these groups are addressed efficiently with survey components specifically tailored to meet their individual feedback expectations.

Beyond increasing the quality of feedback, SURWAY aims to make the collection process even enjoyable for its users. Access is provided through many types of computers to conduct the survey on various devices, thus making it easier to carry out surveys. This is particularly helpful in engaging alumni or other respondents who are off-campus and perhaps like to complete their surveys while on the move. Through its provision of a cohesive, mobile-optimized interface, SURWAY promotes higher engagement and interaction, thereby producing more holistic and representative data.

SURWAY shares the commitment of GNDEC to sustainability activities. Getting away from paper surveys and into an online platform allows the institution to greatly reduce its contribution to environmental destruction. Moreover, a digital system for a survey minimizes data loss, providing for safe record-keeping of records for feedback that may be retrieved easily. Centralizing the management of data of feedback chases away suspicions over results, thus further focusing on the goal of yielding an excellent experience in education by GNDEC.

This platform required ease of scalability, flexibility, and performance, thus motivating the development of SURWAY in MERN technologies; that is, MongoDB, Express.js, React.js, and Node.js. The MERN stack represents an integrated framework of strong but flexible development for web applications. MongoDB is a NoSQL database, allowing flexible data storage,

accommodating a variety of feedback forms and responses effectively. The combination of Express.js and Node.js provides a powerful backend infrastructure that facilitates rapid data processing and effectively manages concurrent requests, thereby rendering real-time analysis achievable. Concurrently, React.js contributes to a dynamic and interactive front-end experience, empowering users to navigate surveys effortlessly while offering a responsive interface for participants.

With the complexity of the educational landscape and their growing demand for data, GNDEC's forward-thinking strategy in collecting feedback through the SURWAY sets a standard for similar organizations. This platform should not only enhance GNDEC's capability to collect and interpret feedback but also create a nimble framework that may change with the shifting needs of the institution as time passes. It would represent a progressive initiative at integrating feedback into the fundamental processes of institutional decision-making in such a way that GNDEC remains responsive to community needs and committed to continuous improvement.

SURWAY is a modern and critical tool for GNDEC as well, serving to provide a better way of getting practical views from the stakeholder regarding the institution. It guides the collection, administration, as well as the analysis of feedbacks that will help GNDEC achieve its goal of creating an ideal learning environment, maintaining high standards of education, and building a strong, active community. With this platform, GNDEC stands ready to make decisions that are data-informed and aligned with what its students, faculty, and alumni require and aspire for to enhance the quality of education and services offered.

1.2 Project Category – Institute-Based

SURWAY is an Institute-Based Project as it is being tailored specifically to improve the mechanisms of Guru Nanak Dev Engineering College (GNDEC) for feedback collection and analysis. The platform, thus addresses the feedback needs of the different students, faculty,

alumni, and staff in the institution. This initiative stands in line with what the institution intends to do, especially towards educational quality elevation, supporting data-informed decision making, and strengthening stakeholder participation through more effective mechanisms for feedback.

The creation of the SURWAY platform is driven by the specific demands of GNDEC regarding scalability, adaptability, and the capacity for real-time feedback assessment. Although the platform has the potential for application in various educational institutions, it has been specifically designed to meet the feedback system needs of GNDEC at this stage. This initiative will not only facilitate the optimization of internal procedures but also offer the necessary flexibility to adapt to the evolving requirements of the institution.

The system is designed to be scalable, meaning it will satisfactorily meet the current requirements of GNDEC but be easily expandable or changeable as and when future requirements or other similar organizations arise. Again, using modern technology like the MERN stack in SURWAY has protected it from technological advancement, thus ensuring it continues to function and serve for a long time.

1.3 Problem Formulation

The existing feedback mechanism at GNDEC was suffering from different problems that hindered its effectiveness and efficiency towards actionable data delivery. Traditionally, Qualtrics was used at GNDEC, which is a paid survey tool, and had advantages but also posed great limitations on the institution to not use their advanced features such as customized question formats and live analytics so that GNDEC could receive more detailed and actionable information.

Moreover, the process of creating and distributing questionnaires relied on extensive paperwork efforts and much time intake. Every year, the feedback coordinator had to start a new account,

thus ensuring the workflow was fragmented and somewhat confusing. The inherent limitations in the free version of Qualtrics, which included restrictions both on the number of surveys that could be generated and the volume of data that could be retained, led to inadequate or constrained feedback acquisition on many occasions, ultimately influencing the overall caliber of insights that GNDEC could extract from data.

The major problems with the existing system were:

- **Restricted Access to Functionality:** The complimentary edition of Qualtrics prevented GNDEC from utilizing sophisticated features, which constrained the customization of surveys and the analysis of data.
- **Creating a new account manually:** For the feedback coordinator, a new account had to be started every year, increasing unnecessary administrative burden.
- **Lack of Instant Feedback:** In the absence of more advanced data processing capabilities, the analysis of feedback experienced delays, which adversely affected the institution's ability to respond to feedback in a timely manner.
- **High Price:** Since it is paid software, it was not worthy of the money; besides, the free version had limited features that did not justify the money.
- **Fractured System:** Many accounts and the manual process created an in-efficient and fragmented mechanism for collecting feedback, which had to be continuously monitored.

Accordingly, SURWAY was proposed as a panacea for these limitations and to provide a more efficient, economically viable, and adaptable means of collecting feedback.

1.4 Identification/Recognition of Need

The need for a better and more robust feedback mechanism was realized with a handful of critical observations:

- **Existing Process Inefficiencies:** The existing system of Qualtrics required manual interventions, renewal of accounts at times, and limited access to features that together themselves prove to be a barrier to the process of feedback and are inconvenient for designers, overseers, and evaluators of surveys.
- **Limited capabilities in Real-Time Data Analysis:** Complex functionalities unavailable in Qualtrics meant that real-time analysis of feedback could not be done, which delayed the institution's realization of feedback and timely improvement.
- **Limit of the free version of Qualtrics:** The free version of Qualtrics had imposed certain limitations on some of the important features, including conditional logic, customized question types, and real-time analytics tools that further restricted the capability to develop effective surveys and derive meaningful insights from the responses.
- **Paid Version Highly Expensive:** This paid version was hugely priced, while on the free version, limitations existed. With such limits to the free version, value for money obtained by the institution was far from maximized.
- **Stakeholder Engagement Increase:** The growth in the number of stakeholders, including students, faculty, staff, and alumni, also necessitated a more agile and scalable process that would handle large volumes of various types of surveys and feedback.

With further modernization of GNDEC and emphasizing the need for a data-driven decision-making process, there is an open call for creating an interface that meets the following goals:

- Facilitate the straightforward development and administration of surveys.
- Provide online real-time data analysis and reporting features.
- Support tailoring of surveys to suit varied needs in diverse departments.
- Support mobile-friendly surveys to encourage greater participation from a diverse range of respondents.
- Shows cost-effectiveness and free from the constraints prevailing in this existing system.

1.5 Existing System

The current mechanism of GNDEC used Qualtrics software, which is a broadly applied tool for surveys. Still, the proprietary software from which GNDEC gathered data regarding its feedback did not exist. As emphasized on the previous page, this institution utilized the free Qualtrics software, which, as noticed, entails quite a few disadvantages:

- **Limitations of Features:** Though Qualtrics is a robust survey tool, the free version was only available without access to critical functionalities like complex analytics, custom reporting capabilities, or multi-page survey designs. This limited the full capability of the platform for complex and interactive analysis of feedback.
- **Recurring Account Setup:** The coordinator for feedback had to open an account every year; this system was fractured and ineffective. The strenuous process created unnecessary administrative work and caused inconsistencies in allotments and analytics of surveys over diverse academic years.
- **Limited Survey Capability:** The limitations in the free version meant that the number of surveys that could be created and distributed simultaneously were limited; data storage space was also limited. This made it challenging for GNDEC as it needed multiple distributed surveys to run simultaneously in different departments.
- **Lack of Customization:** The existing feedback system lacked much open space for customization, mostly in the area of customized needs for departments. Absence of features like Conditional questions, question branching and multi-page survey restricted how much the surveys can be customized to fit the needs of various stakeholders.

The current system has imposed several constraints in terms of features and functionalities, which have resulted in administrative problems that made the process for collecting feedback less efficient. Constraints involving Qualtrics emphasize the need for a more flexible, extensible, and more cost-efficient alternative like SURWAY.

1.6 Objectives

The prime objective of SURWAY is to provide an up-to-date, user-friendly instrument for survey development that caters to the different needs of feedback as required by GNDEC. The project has therefore been conceptualized 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 target is to have an easy interface to create new survey forms. In SURWAY, by providing predefined elements, it is ensured that any user can quickly create surveys without requiring any technical skills. These comprise multiple-choice questions, rating scales and other different types of matrices. Users can easily drag the elements and drop them into any position to design their survey form so that it answers the specific requirements for feedback gathering.

The secondary purpose is to create robust functionalities for editing and managing survey forms. SURWAY enables users to alter existing surveys, hence allowing for redesign and improvement based on changing needs in required feedback. For example, the faculty might want to alter the evaluation surveys for courses mid-semester with additional questions derived from continuous feedback, while administration might want to alter the student feedback forms of alumni for a more specific requirement of data required.

The third goal is providing users with the necessary functionality to analyze and report on data from surveys. SURWAY has built-in data analytic capabilities that allow the user real-time review of the results, thus providing timely insight into information collected. This includes capability in the form of data graphics tools such as graphs, tables, and charts to interpret lengthy datasets and pick out interesting trends.

1.7 Proposed System

Since the proposed system is specifically towards custom-built, web-based surveying using SURWAY, with the entire (custom-built) platform customized for GNDEC, unlike the existing system (Qualtrics), which proposes several improvements:

- **User-Friendliness:** The platform will allow individuals to construct surveys through an intuitive drag-and-drop interface, thereby obviating the requirement for specialized technical expertise.
- **Customization:** The platform is designed to introduce adjustable survey elements, including multiple-choice inquiries, rating scales, and text input fields, hence enabling users to change surveys based on particular requirements.
- **Real-time data processing:** SURWAY is made to process the feedback in real time, thus providing timely insight into gathered data. This enables quick decision-making and agile response.
- **Advanced analytics:** These include in-module functionalities such as interactive charts, graphs, and reports to enable users in the examination of feedback in greater depth and easy identification of important trends.
- **Mobile Compatibility:** The platform is responsive, ensuring that the surveys are accessible and functional on desktop as well as mobile, which most probably will increase participation rates, especially from alumni.
- **Economically Viable:** SURWAY will be free to the institution, unlike Qualtrics, with no reliance on pricey paid versions or subscription-based models.

1.8 Unique Features of the Proposed System

Some of the features in SURWAY that distinguish it from other systems include:

- **User-centric interface:** The website is designed intuitively and lavishly simple so that an amateur can really come to design and administer surveys, though it hardly considers the high profile geek.
- **Instantaneous Feedback:** The capability to evaluate and interpret feedback data as it is received will furnish decision-makers with the necessary insights promptly, eliminating the need to rely on manual analysis.
- **Personalizable Elements:** Different question types can be added, including applying conditional logic to make dynamic changes in the survey based on user inputs.
- **Integration with GNDEC's Current Systems:** SURWAY has been developed to facilitate smooth integration with the current systems at GNDEC, thereby enabling straightforward data import and export as required.
- **Scalability:** The system can scale up to meet the growing needs of the institution-be it more surveys, more users, or more advanced data analysis.

CHAPTER 2 REQUIREMENT ANALYSIS AND SYSTEM SPECIFICATION

2.1 Feasibility Study

A feasibility study is very essential to assess the practicality of the SURWAY platform. The different facets of feasibility are discussed at length as below:

2.1.1 Technical Feasibility:

Technical feasibility of SURWAY is exercised through referring to the technology stack, required resources and system architecture in developing and implementing the platform. A study like this will assure GNDEC that it has appropriate technological infrastructure and skills for the effective deployment.

Technical Support :

- **MERN Stack** MongoDB, Express.js, React.js, Node.js: This uses the whole MERN stack for its development, which provides a more modern scalable and efficient way of creating interactive, responsive web applications. Its popularity and robustness make it suited for dynamic data handling and also for user-friendly interfaces in applications.
- **Database (MongoDB):** MongoDB represents a NoSQL database that is particularly well-suited for the management of unstructured data, a prevalent characteristic observed in feedback surveys. Its inherent flexibility in organizing documents and collections facilitates the storage and retrieval of survey responses, thereby improving both performance and efficiency.
- **Frontend (React.js):** Since React.js enables the development of a very responsive user interface for users to have an interactive experience with the application, a component-based architecture promotes modular development that allows the easy altering or

enhancing of specific parts.

- **Backend (Node.js & Express.js):** Node.js provides the server environment, while Express.js facilitates efficient routing and handling of requests. Together, they provide a scalable solution capable of handling high volumes of survey data and requests.

Cross-Platform Compatibility:

It is made to be browser-based, thus supporting the computer, any tablet, or smartphone of any kind, so students, faculty, and administrators can take the advantage of using the system from anywhere.

Safeguards Procedure:

- **User Authentication:** SURWAY employs JWT authentication, thus ensuring safety through security as only registered members can access certain sections.
- **Access Control:** Role-based access control enables graded levels of access to the system, thereby distinguishing administrators from faculty members and students.

Scalability:

The MERN stack facilitates substantial scalability, thereby guaranteeing that the platform is capable of managing a rising influx of users and data as time progresses. The adaptability of MongoDB further permits dynamic scaling, which simplifies the process of accommodating GNDEC's evolving requirements.

Maintenance and Support:

The open-source nature that comes along with the MERN stack gives straightforward maintenance and accessible resources. With GNDEC having access to constant updates from the community, they know that it'll be constantly enhanced and supported, ensuring that SURWAY

remains up-to-date and running.

In conclusion, technical feasibility analysis verifies that the suggested technology stack, infrastructure, and security protocols are adequate for meeting the set goals of SURWAY.

2.1.2 Economic Feasibility

Economic feasibility evaluates the cost-efficiency associated with the development and eventual maintenance of the SURWAY platform so that it delivers a financially viable solution for GNDEC.

Development Cost:

- **Initial Development:** An investment in skilled developers-for MongoDB, Express.js, React.js, and Node.js-would be required if the development of SURWAY were on the MERN stack.
- **Infrastructure:** Since the platform will be hosted on GNDEC's existing servers or a cloud service, infrastructure costs are limited only to server setup, storage, and any needed subscription of a cloud service, such as AWS or Azure, for hosting. MongoDB's NoSQL database is also cost-effective and efficient which means database management expenses are reduced to a minimum.

Maintenance Costs:

- **Routine Maintenance and error correction:** MERN stack is enhanced by an active open-source community, GNDEC can draw upon available resources to minimize its maintenance cost.

Training and Operating Costs:

- **Staff Training:** Faculty and admin. staff will require very little training since the

interface is user-friendly. Major training session will be an internal; it should be supplemented with tutorials on "how to design a survey, how to analyze outcome results."

- **User Support:** Any technical support that an institute's user may require can be made available to such users. Since SURWAY is customized for GNDEC, institute can deliver focused effective support at low expense.

Money Saving:

- **No External Subscription Fees:** Through the creation of an internally developed platform, GNDEC circumvents the ongoing subscription expenses linked to utilizing a service such as Qualtrics. This initiative signifies a considerable financial saving, particularly over an extended period.
- **Efficiency:** SURWAY has its own set templates and reusable survey features. They reduce time and efforts in the administration of surveys. This efficiency reduces 'man-hours' and other associated administrative costs.

In conclusion, the analysis of economic feasibility reveals that SURWAY represents a financially viable solution, providing considerable savings over the long term.

2.1.3 Operational Feasibility

Operational feasibility evaluates if SURWAY meets the goals, infrastructure, and end-user requirements of GNDEC. This implies that the platform will be practically implementable and sustainable within the institution.

Aligns with Organisational Objectives:

- SURWAY actively helps GNDEC achieve its vision for the campus: building greater institutional openness, responsiveness, and student involvement. It allows GNDEC to

gather direct feedback to inform decision-making on instruction, curricula, and campus facilities.

- The feedback system of the platform promotes transparent dialogue among students and administrators, enabling GNDEC to undertake decisions based on data that correspond with its objective of achieving educational excellence.

Acceptance and Engagement of Users:

- **Accessibility for Students:** The user-friendly interface of SURWAY, developed using React.js, is made easy-to-navigate for the student. Also, the cross-device accessibility of the platform further enhances the engagement levels.
- **Faculty and Administrators Acceptance:** Training sessions could be conducted to acquaint the faculty and administrators with the functionalities of the platform, which includes building and managing the surveys, response interpretation, and producing reports. As a proprietary system customised for GNDEC, it is expected that the users would literally fall in place.

Operational Workflow Integration:

- SURWAY is engineered to harmoniously integrate with GNDEC's existing feedback workflows. The incorporation of survey templates, automated data storage, and real-time analytics minimizes the necessity for manual data entry and processing, thereby optimizing the feedback management procedure.
- Customizable roles and permissions enable administrators to control access to surveys and results, ensuring that the feedback process remains organized and efficient.

Ease of Maintenance:

- The modular characteristics of the MERN stack facilitate the processes of updating,

enhancing, and diagnosing issues within the platform. As the internal technical team at GNDEC gains proficiency with the platform, they will be capable of managing maintenance activities independently, thereby ensuring sustained operational efficiency.

- Enhanced by the open-source community, SURWAY ensures compatibility with new technologies, protection against vulnerabilities, and the benefits of the latest features.

Scalability for Future Needs:

- SURWAY's architecture allows GNDEC to expand its functionality and accommodate increased user demands over time. The platform can evolve to include additional feedback tools or adapt to changing survey requirements, ensuring long-term relevance.
- Because as GNDEC evolves, SURWAY should be adapted to engage other stakeholder groups than students; thus, alumni and industry partners outside the institution will be added to the portfolio of feedback the institution collects.

Stakeholder Benefits:

Through an adjustable platform that gives a learner an opportunity to express their point of view, advantages are derived in terms of greater engagement and satisfaction with their learning process.

- **Faculty:** Faculty gain insights into student satisfaction and course effectiveness, enabling them to refine their teaching methods.
- **Administrators** make timely and quality decisions to boost the reputation and operational effectiveness of GNDEC using real-time data and reporting instruments.

In a nutshell, the feasibility review shows that SURWAY aligns well with GNDEC's operational objectives, user requirements, and institutional processes. The platform's intuitive design and integration capability with existing systems and flexibility guarantee the smooth incorporation

into the operations of GNDEC as it leads to a sustainable and significant solution.

2.2 Software Requirement Specification (SRS) Document

2.2.1. Purpose

The primary objective of the survey management system is to enhance the efficiency of the collection, organization, and examination of diverse surveys. It accommodates various survey formats, enabling the modification of questions and response options for purposes of data gathering, analysis, and feedback. This system is developed to function as a comprehensive instrument for organizations, enabling them to evaluate user feedback, make data-driven decisions, and promote enhancements in multiple domains.

2.2.2. Scope

The system supports different formats of survey management: text-based questions, rating scales, and matrix-type responses. It generally covers a wide range of functions in creating, deploying, collecting, and reporting responses. The system is flexible and can be used in different domains- instructional, health, corporate feedback, and market research-as this allows its users the creation of their specific survey formats and scales.

2.2.3. Target Audience

The target group includes:

- **Survey Developers:** People who create and set up questionnaires.
- **Survey Respondents:** People who complete the surveys.
- **Faculty/Administrators:** Personnel who analyze survey results, determine the response patterns, and produce reports.

2.2.4. Summary

It supports developing surveys that allow incorporation of flexible features and support different types and formats of questions. Whereas the backend manages users and surveys, the frontend provides a user-friendly interface to the administrator as well as to the participants. The data gathered from the surveys can be either exported or analyzed internally, thus facilitating the generation of actionable insights.

Software Specifications

2.2.5. Data Requirements:

- **User Data:** Users information (creator of surveys, respondents, and administrators)-ID, roles, access level, and contact information.
- **Survey Data:** Each survey will contain an ID, questions, response type, choices, and timestamps for tracking completion.
- **Response Data:** All the survey responses of respondent ID, survey ID, timestamps, and answers will be saved in order to preserve data integrity and facilitate further analysis.

2.2.6. Functional Requirements:

- **Survey Development and Controls:** The system must provide the functionality to add, edit, delete, or duplicate surveys. It should also allow for various types of questions: multiple choice, text input, and matrix.
- **Survey Deployment:** Users should be able to share surveys through individual links, embedded forms, or QR codes, and distribute them privately or publicly.
- **Collect the data:** The system gathers responses instantaneously, guaranteeing that data is stored securely within a database and is accessible for subsequent analysis or reporting.
- **User Authentication and Authorization End:** It supports multi-role access control

where only authorized users can create surveys, view responses, or even generate reports.

- **Survey Status Monitoring:** Track survey status as 'draft,' 'active,' 'closed,' etc so the administrator knows which are active and which are completed.

2.2.7. Performance Requirements

- **Scalability:** The system must be capable of supporting a large number of concurrent users and handling high volumes of data, thus enabling large organizations to execute several surveys at the same time.
- **Latency Period:** The survey interface should have minimal delay while in use. It is anticipated that loading survey questions will be within 1-2 seconds and submission of responses.
- **Data Processing Speed:** Datasets with fewer than 10,000 responses should generate analytical reports within less than 10 seconds, with progressive loading for larger datasets.

2.2.8. Dependability Requirements:

- **Reliability:** The system should thus provide an availability rate of at least 99.9% for the polls to be always reachable, without interrupting data collection.
- **Backup and Recovery:** Prevent data from getting lost by establishing regular backups, preferably on a daily basis. A recovery protocol in case servers fail or data gets corrupted should also be established.
- **Fault Tolerance:** The system shall handle exceptions gracefully where data relevant to the user will not get lost in case of unexpected errors.

2.2.9. Maintainability Requirements:

- **Modularity:** The architecture of the system ought to be modular, facilitating

straightforward maintenance, updates, and scalability. Each module (such as survey creation, response collection, and reporting) must be self-sufficient and capable of being modified independently.

- **Code Documentation:** The codebase should be well-documented with inline comments and a separate API documentation for developers to assist in future updates.
- **Version Control:** Implement a version control system-for example, Git-so changes may be tracked and rolled back if necessary.

2.2.10. Security Requirements:

- **User Authentication:** Multi-factor authentication should be enabled for administrators and other high-privilege roles to prevent unauthorized access.
- **Access Regulation:** Role-based access control should ensure users can only perform actions permitted by their role. Respondents, for example, should only access surveys assigned to them, while admins have broader access.
- **Audit logging:** Log all changes and access made to the survey data including creation of a survey, editing, and deletion activities for high security tracking and monitoring.

2.2.11. Look and feel requirements:

- **User Interface Design:** The system should have a modern and accessible user interface, comprising accurately applied color schemes, typography as well as icons, for maximizing usability and access.
- **Accessibility:** The design must adhere to accessibility standards, including WCAG 2.1, to facilitate navigation of the interface and successful completion of surveys for users with disabilities.
- **Adaptive Design:** The interface must be optimized for various screen dimensions to guarantee usability on mobile devices, tablets, and desktop computers.

2.2.12. User Interface:

- **Dashboard:** An administrative dashboard would provide the view of active, drafts, and completed surveys, including the responses from recent surveys and analytical insights.
- **Survey Builder:** There should be a drag-and-drop survey builder interface to enable users to add and arrange question types, set survey logic, and preview the survey before its deployment.
- **Interface for reporting and analytics:** A response summary interface to view and generate reports together with the ability to filter certain subsections of survey data.

2.2.13. Constraints, Assumptions and Dependencies:

Constraints:

- **Technical Constraints:** In a particular technology stack will deploy (e.g., MongoDB, Express.js, React, Node.js). The deployment significantly limits the integration of technologies not compatible with that stack.
- **Legal and Regulatory Constraints:** The data must be safe during data transfer and should adhere to norms like GDPR.
- **Financial Constraints:** It does have a fixed budget for development, so in turn, restricts the usage of third-party APIs and resources.

Assumptions:

- Availability of internet for users to respond to surveys.
- The administrators will have rudimentary training to design surveys and interpret analytics reports.
- All users, including respondents, will utilize updated browsers that support HTML5.

Dependencies:

- **Database Management:** The system relies on a database service that stores survey configurations and response data to be, for example, MongoDB.
- **Authentication Service:** The system makes use of an authentication service for secure login along with role-based access control.
- **Server and Hosting Architecture:** The system needs stable hosting and server support with the ability to easily upscale for high-traffic periods.
- **External Libraries:** Third-party library dependencies; for example, SurveyJS library or UI/Widget components would require periodic updates and checks for compatibility.

The comprehensive SRS framework is intended to establish a solid basis for the creation of a survey management system, encompassing critical requirements while facilitating adaptability in subsequent phases of development.

2.3 Software Development Life Cycle (SDLC) Model to be Used

The development of the Feedback Management System (SURWAY) will employ the Agile Software Development Life Cycle (SDLC). The iterative and adaptable nature of Agile is particularly appropriate for this initiative, as it anticipates that input from students, faculty, and administration will influence the system's ongoing functional development.

2.3.1 Key Characteristics of Agile Methodology

The Agile methodology is characterized by core principles that align well with SURWAY's requirements:

- **Incremental and Iterative Development:** Agile breaks the project into small, manageable sprints of size 1–4 weeks where functional parts develop incrementally. This

ensures early and continuous delivery, which lets SURWAY features be deployed and tested more frequently.

- **Adaptability in Change:** Agile methodology is scoped to accommodate change, an important characteristic of SURWAY. It should be easy to introduce new functionalities or change parts of the survey as inputs accumulate continuously.
- **Collaborative Approach:** The Agile methodology supports intensive cooperation among developers and stakeholders, ensuring continuous involvement of students, faculty, and administrative representatives so that the features of SURWAY meet the anticipations of its users.
- **Continuous Improvement and Customer Focus:** The potential of Agile is promoting regular testing and refinement to achieve high levels of quality with user satisfaction. SURWAY functionalities shall be iteratively tested for usability, relevance, and functionality.
- **Quick Delivery and Enhanced Quality:** Agile focuses on delivering working components with each sprint, so that SURWAY functionalities can be released quicker and useful features come earlier in the development process.

2.3.2 Phases of Agile SDLC for SURWAY

Successive phases of the Agile process have been customized for SURWAY operations, balancing the needs for structured workflow with flexibility.

- **Requirement Gathering:** The team meets at the start of each sprint to refine the requirements, based on input directly gathered from students and faculty and with a focus on top features.
- 2 **Design and Planning:** Features in focus, like surveys creation and response tracking functionalities, have wireframes and high-level layouts of designs made in each sprint.

- 3 **Development:** Planned features in a sprint module are developed. SURWAY starts with the most critical functionalities like question creation and distribution of available surveys. Advanced features come later in the sprints.
- 4 **Testing:** After development, rigorous testing identifies any issues. Testing will ensure SURWAY's components, such as survey submissions and analytics, function correctly and meet usability standards.
- 5 **Deployment and Review:** The team implements functional features at the end of every sprint, and collects feedback for subsequent modifications. This process ensures that SURWAY's usability and pertinence are continued to be elevated.
- 6 **Retrospective:** Each sprint ends with a review of successes, challenges, and improvements that can help refine the development process for future sprints.
- 7 **Ongoing Maintenance:** Even after initial deployment, Agile's iterative cycle keeps SURWAY up-to-date, incorporating feedback and refining features as user needs evolve.

2.3.3. Rationale for Choosing Agile:

The Agile methodology is well-suited for SURWAY due to its flexibility, user-focused approach, and iterative delivery. The need for adapting to evolving requirements in student feedback surveys and the importance of early feature delivery make Agile an ideal choice. Agile's incremental structure allows us to deploy features rapidly, gather feedback continuously, and incorporate adjustments throughout development, ensuring SURWAY aligns with user needs and improves consistently with each iteration.



Fig. 2.1 – Agile methodology used

CHAPTER – 3 SYSTEM DESIGN

3.1 Design Approach

3.1.1 User Management Module

This module forms the basis for access and security with respect to user. It aims to secure a friendlier interface for user registration, login, and account management.

- **User Registration and Authentication:**
 - Registration: The users provide their mandatory details such as email, user name, and password to create accounts.
 - Email Verification through OTP : An OTP is mailed for authenticating the identity of the user before the account activation. This increases the security level and also brings about the reduction in the presence of spam or unwanted accounts.
 - JWT Authentication: The use of JSON Web Tokens is done to ensure a secure handling of sessions while providing stateless authentication. JWTs are signatures. On the server-side, verification of the signature is done which ensures that the integrity of the token .
- **Session Management:**
 - JWT Refresh and Access Tokens: Utilize a short-lived access token in tandem with a long-lived refresh token as a mechanism for ensuring securely interactive user sessions. The backend is the one that creates and verifies these tokens to enable smooth flow and minimize instances of illicit access.
 - Token Expiration and Renewal: Once tokens expire, users are logged out automatically or challenged to renew their session by utilizing the refresh token .
- **Security Enhancements:**
 - Password Hashing: Keep passwords hashed in the database for added protection against data breaches-in this case, using bcrypt, to mention one example.
 - Validation Checks: Enforce validations for emails and passwords to prevent common

security threats from weak or format invalidated emails

3.1.2 Survey Creator Interface

The Survey Creator Interface is the designing tool used to interactively create the structure of the surveys. Using this interface, users make a series of forms of customized questions in different question types and configurations.

- **Dynamic Adding of Questions:**

- Predefined Question Types: It provides the user with a set of predefined question types, like multiple choice, dropdown, matrix, text entry, and rating scales.
- Branching and Conditional Logic: Complex survey flows can be created by setting conditions on questions. Some question will only appear if a prior question was answered in a certain way.
- Drag-and-Drop Interface: A user-friendly drag and drop interface for easily adding, reshuffling, and configuring questions, hence making the task of creating surveys intuitive.

- **Real-time Preview and Save Functionality:**

- Live Preview: With preview functionality, users can have their survey built out, viewing a sample of the survey from an end-users' perspective, enabling review of the structure and potential errors before finalization
- Auto save and manual save features for user control over updating any changes in storage

- **Include full customizable settings options**

- Elements of Survey Design: Basic styling and layout choices such as font and color selections, along with section dividers are intended to allow a user to further control the look of their survey.
- Question Validation: Users can specify validation rules, such as required fields, minimum/maximum text length or allowed numeric ranges, to keep the data coming in top notch.

3.1.3 Survey Data Management

This module manages the storage and processing of survey structures, allowing for an effective storage and retrieval of forms and their modification.

- **Storage of Survey Structure:**

- The entire form, questions, options, and any kind of branching logic will be stored as a structured JSON object.
- Survey Versioning: Any modifications made to a survey using the Survey Versioning feature are created as a new version instead of overwriting the original. This ensures that, in case users need to go back, they can revert back to previous versions.

- **Data Integrity and Autosave:**

- Autosave Feature: Autosave at regular intervals ensures no changes are lost, and on closing, users are prompted to either save or discard changes, preventing unintended overwrites.
- Database Schema: Consider a document-oriented database (such as MongoDB), whose documents are where surveys are stored in JSON; that way, the survey structure updates do not carry pains of relational dependencies.

3.1.4 Survey Preview and Execution Module

Preview and Execution module allows full previewing of the surveys before publishing and is the portal through which end-users will be answering the survey.

- **Survey Preview:**

- Full Preview Mode: The final survey, exactly how the respondent will see it. This is useful for error catching, a final review of logic, and ensuring all the layout is correct.
- Conditional Logic Testing: Test by giving a series of survey questions to other users, allowing them to test the branch and skip logic so you can ensure the right questions are being displayed based on their answers.

- **Survey Execution:**

- Frontend Optimization for User Experience: Responsive design; users can use it on desktop, tablet, and mobile.
- Autosave for long Surveys. For long surveys, autosave answers after some time to avoid an interruption in the session if it happens.
- Error Handling and Validation: Presents obvious errors if the validation criteria are not met. It will guide users to fill in required fields and adhere to input constraints.

3.1.5 Result Analysis and Reporting Module

This module deals with processing the collected survey data to provide insights through means of data visualization and export options.

- **Data Aggregation and Filtering:**

- Customizable Filtering: The results obtained from the respondents may be filtered based on demographic or answer criteria. This makes it easier to segment and analyze the specific respondent groups.
- Aggregation and Summary: Automatically calculates the mean, sum, and percentages in case the data collected are numerical or ratings

- **Visualizations and Data Representation:**

- Charts Types: It provides a variety of charts like bar charts, pie charts, line graphs, and matrix tables with appropriate visualization of the data of surveys.
- Customisable Views: Its customization allows the viewers to choose which data points and metrics to view along with further drilling down into subsets of specific data.

- **Export Functionality:**

- Export Options: You can export data in CSV, Excel or PDF for further analysis or to share with others.
- Graphical Export: You can export charts and tables into image files or PDF documents, so you can embed the results of a survey in presentations or reports.

3.1.6 API and Data Storage Module

This module acts as a bridging platform between the frontend and the backend, performing data storage and retrieval operations along with the preservation of integrity in data.

- **CRUD Operations:**
 - Survey and User Data Management: It provides secure endpoints regarding the creation, reading, updation, and deletion of surveys and user profiles.
 - Rate Limiting/Throttling: Prevents abuse by limiting the number of requests coming from individual users or IP addresses, which avoids overwhelming the server.
- **Data Consistency and Transaction Management:**
 - Database Transactions: If it is supported by the database, then use transactions to ensure atomic operations, especially when updating correlated records, such as the surveys and responses.
 - Data Validations and Sanitization: Validate all incoming data so that it does not contain erroneous values; sanitize inputs to prevent many security-related vulnerabilities, including SQL injections and cross-site scripting (XSS).
- **Error Handling and Logging:**
 - Error Responses and Logging: Proper HTTP status codes are provided for every type of error. It is much easier to debug and track errors if server-side errors are logged.

3.2 Detail Design

3.2.1 Use Case diagram:

- **Registered User:** A user who has created an account and logged into the system. They can create, edit, and manage surveys and view survey reports.
- **Admin:** A superuser with permissions to manage registered users, oversee surveys, and access usage metrics. They may also have access to more advanced analytics tools.

- **Database System:** The backend storage system, which stores user data, survey data, responses, and other related records.

3.2.2 Use Cases:

- **User Registration and Login:**
 - Register: Allows users to create an account by providing their email, username, and password.
 - Login: Authenticates a user based on their credentials.
 - Email Verification: An OTP (One-Time Password) is sent to verify the email address, ensuring account authenticity.
 - Password Reset: Allows users to reset their password via an email link or OTP, ensuring secure recovery in case of a forgotten password.
- **Survey Creation and Editing:**
 - Create Survey: Enables users to initiate a new survey, setting up questions and defining configurations.
 - Edit Survey: Users can modify existing surveys, changing questions, or adjusting survey logic.
 - Preview Survey: Allows users to see how the survey will appear to respondents, ensuring that formatting and logic are correct.
 - Save Survey: Saves the current state of a survey, allowing users to continue editing or exit without losing progress.
 - Delete Survey: Permanently removes a survey and associated data.
- **Survey Management:**
 - Retrieve Survey Data: Fetches stored survey structures or survey responses for viewing or further editing.
 - Update Survey Data: Updates specific survey records, such as changing questions, options, or settings.

- Autosave: Automatically saves progress periodically to avoid data loss.
- Version Rollback: Allows users to revert to previous survey versions, helpful when accidental changes are made or if there is a need to return to an older survey structure.
- Apply Survey Logic: Allows conditional logic to be configured, enabling specific questions to appear or skip based on respondent answers.
- **Survey Execution and Analysis:**
 - Collect Responses: Records responses as users complete the survey, validating entries based on predefined rules.
 - Store Responses: Safely saves each completed survey's responses in the database for future retrieval.
 - Aggregate Data: Aggregates responses for reporting and analysis, such as calculating averages or identifying trends.
 - Generate Reports: Produces visualizations and summary reports based on collected data.
 - Export Data: Allows users to export survey results in formats such as CSV, Excel, or PDF for offline analysis.

3.2.3 Flow Chart

Survey Creation Flow Chart

This flow chart depicts the process from creating a survey to saving or publishing it.

- **User Initiates Survey Creation:**
 - The user clicks the "Create Survey" button, starting a new survey.
- **System Generates Dynamic Survey Structure:**
 - The system displays an interface for adding various question types (e.g., multiple choice, text, rating).
 - As the user interacts, a JSON structure representing the survey is generated

dynamically in real-time, capturing questions, options, and logic.

- **User Configures Survey Logic:**

- The user sets conditions and branching logic if needed (e.g., displaying Question 5 only if Question 3 is answered "Yes").

- **Save and Preview Options:**

- Save Survey: The user can save the survey progress, initiating an API call to store the survey JSON in the database.
- Preview Survey: The user previews the survey to confirm layout and functionality.

- **User Publishes Survey:**

- When satisfied, the user clicks "Publish." The survey's status updates to active, making it accessible to respondents.

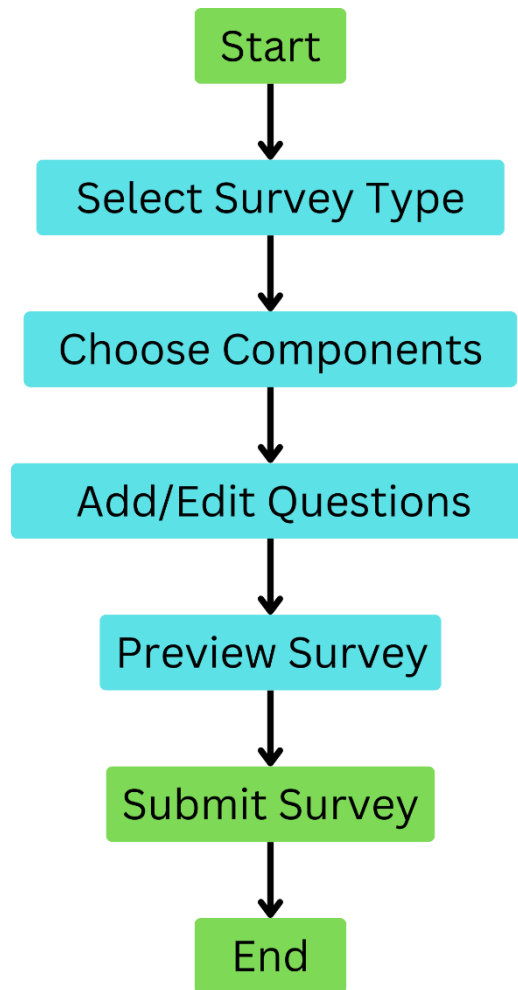


Fig 3.1 – flow diagram of creating a new survey form.

User Registration Flow Chart

This flow chart outlines the steps from a user entering credentials to successful registration and account activation.

- **User Enters Registration Details:**
 - The user provides their email, username, and password on the registration page.
- **System Validates Uniqueness:**
 - The system checks that the email and username are unique in the database.
- **System Sends OTP:**
 - If unique, the system generates an OTP and emails it to the provided address for

verification.

- **User Enters OTP:**

- The user enters the OTP received in their email for confirmation.

- **System Confirms and Registers User:**

- If the OTP is valid, the system finalizes the registration, saving the user's data to the database.
- Account Activation: The user's account is marked as active, allowing access to the system.

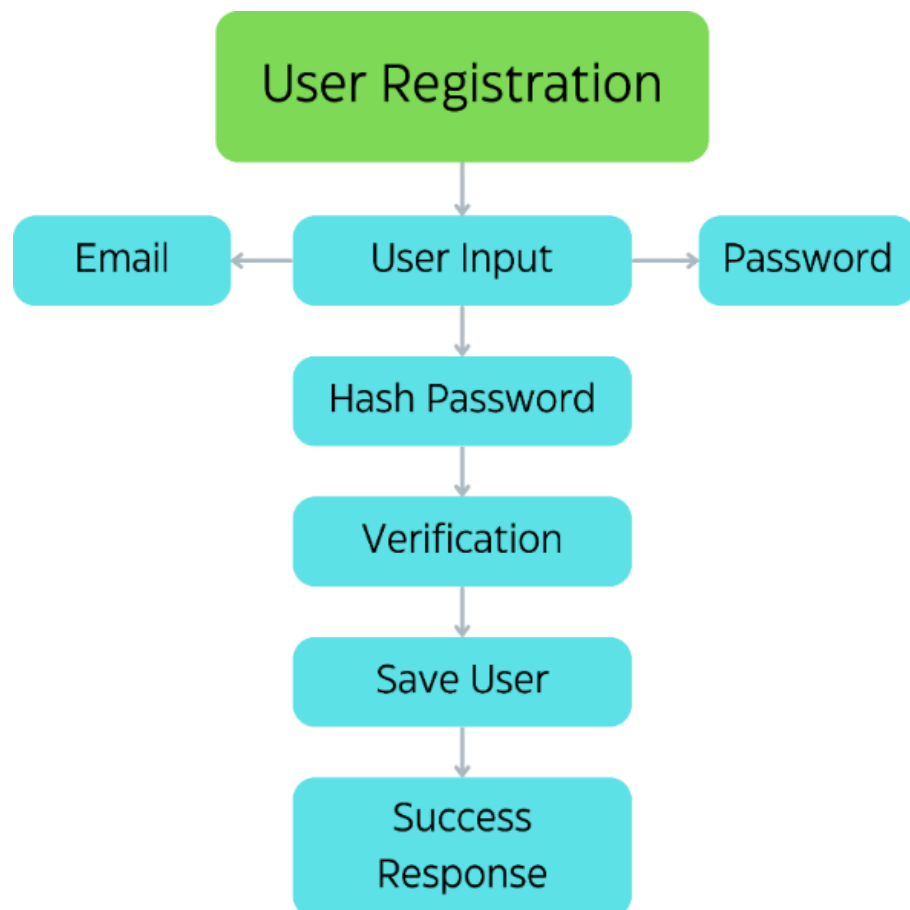


Fig 3.2 – flow diagram of user registration.

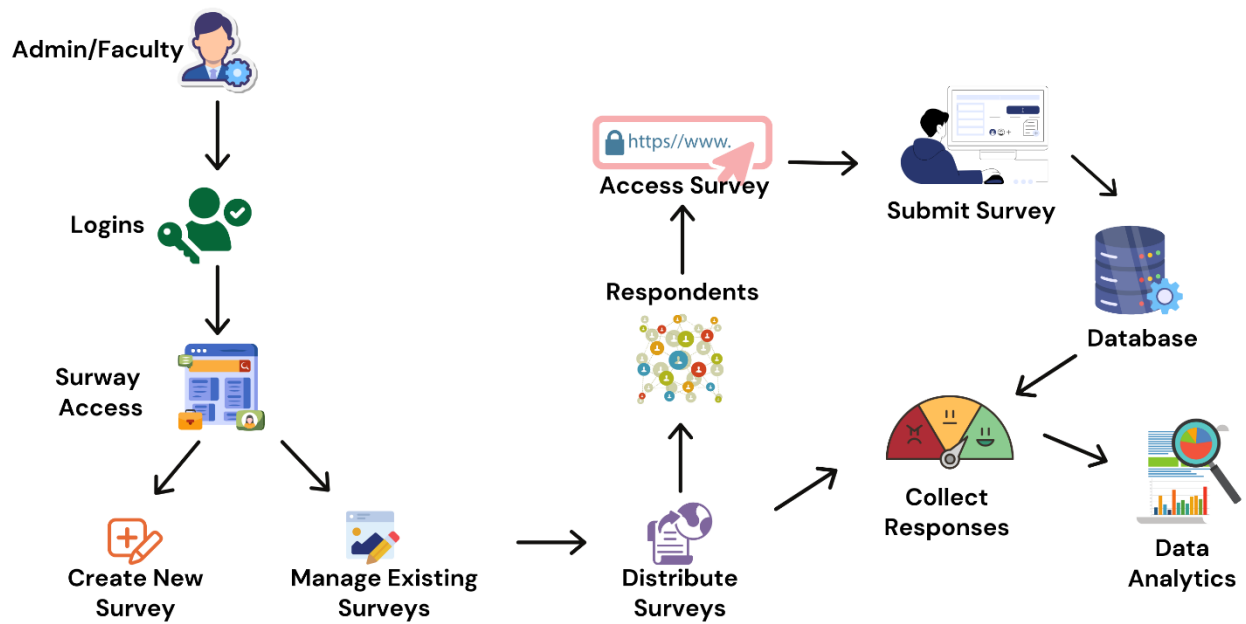


Fig 3.3 – flow diagram of Surway

3.3 User Interface Design:

3.3.1 Frontend Technologies:

React:

- **Component-Based Structure:** React's component-based architecture is ideal for building modular UI elements that can be reused across the application.
- **Virtual DOM:** React's virtual DOM enables efficient rendering, allowing the survey creator interface to respond quickly to user interactions without full page reloads.
- **Hooks for State and Side Effects:** React hooks, such as `useState` and `useEffect`, provide a streamlined way to manage component states, side effects, and lifecycle events.

SurveyJS Library:

- **Survey Form Elements:** SurveyJS provides a wide array of pre-built question types (e.g., multiple choice, dropdown, matrix, text input), making it easier to implement complex survey forms.
- **JSON-based Survey Schema:** SurveyJS stores surveys as JSON structures, compatible

with MongoDB and other JSON-centric databases. It allows easy modifications and storage of survey structures.

- **Built-in Survey Preview and Execution:** The library supports live previews, so users can see exactly how their survey will look to respondents. This functionality enables real-time adjustments to questions, layout, and logic.

Tailwind CSS:

- **Responsive Design:** Tailwind CSS offer pre-designed UI components and responsive grids. It provides an extensive library of components with minimal configuration, and allows for more customization by applying utility classes directly in HTML, ideal for creating a unique, custom look.
- **Consistent Styling:** Using a consistent style framework ensures a cohesive user experience across various devices, making the survey interface responsive and accessible.
- **Customizability:** Tailwind CSS allows custom styling while keeping CSS manageable, which is particularly useful for designing a branded and polished interface.

Backend Technologies:

Node.js with Express:

- **RESTful API Development:** Express enables building RESTful APIs that handle requests for survey data management, authentication, and result retrieval.
- **Middleware Support:** Middleware in Express (e.g., for authentication, input validation, and error handling) enhances security and data integrity.
- **Scalable Architecture:** Node.js's event-driven, non-blocking I/O model makes it ideal for handling multiple API requests concurrently
- **Integration with MongoDB:** MongoDB is a natural fit for Node.js, as both use JSON-compatible data formats.

MongoDB:

- **JSON Document Storage:** MongoDB stores survey structures, responses, and user data

as JSON documents

- **Schema Flexibility:** Since surveys may vary in structure, MongoDB's flexible schema enables easy storage.
- **Efficient Data Retrieval:** MongoDB's querying capabilities allow efficient retrieval of data, supporting aggregation queries for generating survey reports and result analysis.
- **Indexing and Scaling:** Indexes on commonly queried fields (e.g., user ID, survey ID) improve performance.

3. JWT (JSON Web Tokens)

- **Stateless Authentication:** JWTs enable stateless authentication, meaning the server does not need to store session data.
- **Secure User Sessions:** JWTs can be signed and verified with a secret key, preventing tampering.
- **Token Expiry and Refresh:** JWTs can have a short expiration time with a longer-lived refresh token, requiring users to re-authenticate periodically

3.4 Methodology

The methodology employed in the development of Surway encompasses several sequential steps aimed at achieving the project objectives effectively and efficiently. Each step contributes to the iterative refinement of the system, ensuring the delivery of a robust and user-centric solution.

The methodology consists of the following key phases:

3.4.1 Requirement Analysis:

- **Survey Creation:** Ability for users to create and customize surveys with multiple question types (e.g., multiple-choice, text, matrix).
- **Survey Customization:** Support for branching logic, validation rules, and real-time survey previews.

- **Data Collection:** Storage of survey responses with features for data aggregation, filtering, and export.
- **User Authentication and Management:** Secure login, registration, email verification, password reset, and role-based access control (admin and standard users).
- **Data Analysis and Reporting:** Visualizations of collected data (e.g., charts, tables) to represent survey results for easy interpretation.

3.4.2 Design Phase:

- **Survey Creator:** Interactive UI for adding questions, configuring survey logic, and previewing.
- **Survey Execution Interface:** Simple, user-friendly interface for respondents to answer surveys.
- **Admin Dashboard:** Interface for monitoring user activity, managing surveys, and accessing system logs.

3.4.3 Development Phase:

- **Frontend Development:** Implement SurveyJS for survey question types, real-time previews, and JSON-based survey structures.
- **Backend Development:** Develop RESTful APIs in **Express** for handling survey CRUD operations, user management, response collection, and data analysis.
- **Database Development:** Design collections in **MongoDB** for user data, survey structures, and responses.
- **Deliverables:** Details endpoints, expected request/response formats, and any authentication requirements.

3.4.4 Testing Phase:

- **Unit Testing:** Test API endpoints using **Mocha** or **Jest**, ensuring each endpoint performs as expected with accurate data handling and validation.

- **Integration Testing:** Test interactions between modules, such as frontend-backend communication and backend-database operations, ensuring data flows correctly.
- **End-to-End Testing:** Test entire user journeys, from registration to survey creation, completion, and result analysis, using tools like **Cypress** or **Selenium**.
- **Performance and Security Testing:** Measure response times and system stability under heavy load, ensuring it handles large datasets and multiple users.

3.4.5 Deployment Phase:

- **Deployment Planning:** Deploy the complete system in a staging environment for final tests, simulating real-world conditions.
- **Deployment Execution:** Set up continuous integration and deployment pipelines using tools like **GitHub Actions** or **Jenkins** to automate deployment and testing.
- **Post Deployment Testing:** Outlines steps for deployment, rollback procedures, and monitoring configuration.

3.4.6 Maintenance and Enhancement:

- **Monitoring and Maintenance:** Continuously monitor system logs, resolve any errors or bugs, and maintain system health to ensure a stable user experience.
- **Continuous Improvement:** Collect feedback to identify new features or improvements, such as additional question types, better data visualizations, or advanced survey logic.
- **Feature Enhancement:** Periodically release new versions with enhanced features and update the documentation to reflect changes, keeping users informed.

CHAPTER – 4 IMPLEMENTATION & TESTING

4.1 Introduction to Languages, IDEs, Tools, and Technologies Used for Project Work

The **SURWAY**, or Feedback Management System, was built using the MERN stack, combined with SurveyJS and Material-UI (MUI) in order to deliver an adaptable, responsive, and scalable solution. The following section outlines the primary technologies used, described in their functionalities and how they have contributed to the project.

4.1.1. Languages and Frameworks

MERN Stack:

- **MongoDB:** A NoSQL database designed for the storage of survey data, responses, and user information in a flexible format akin to JSON, facilitating efficient data storage and retrieval.
- **Express.js:** The backend web application framework for Node.js, managing server requests and defining routes. It supports handling HTTP requests, responses, and routing, making data processing seamless and efficient.
- **React.js:** This front-end JavaScript library is used for dynamic and responsive user interface development. A component-oriented architecture of React allowed creating modular components of surveys, thus enhancing a richer user experience through smooth interaction and quick reloading.
- **Node.js:** This enables JavaScript to be run on the server-side. The same script language is both at the front and back ends, making it easier to build and ensuring compatibility.

SurveyJS:

SurveyJS constitutes a dedicated library aimed at streamlining the process of survey development and oversight. This library incorporates a range of customizable question formats, validation mechanisms, and functionalities for collecting responses, thereby minimizing the time required

for the development of survey-oriented interfaces. Furthermore, SurveyJS has been integrated with React components to enhance user experience intuitively while simultaneously permitting developers to dynamically regulate survey functionality.

Material-UI (MUI):

An acclaimed framework for React user interfaces, MUI, was used to enhance the aesthetic appeal and usability of the interface of this project. It comes with pre-built, ready-to-use React components that ensure a sophisticated and modern look. MUI has made uniform implementation of design principles easier and more accessible, such as responsiveness, a friendly navigation system, and accessible controls across multiple devices.

4.1.2. Tools and IDEs

- **Visual Studio Code (VS Code):**

This project had the major Integrated Development Environment (IDE) as VS Code, which supported extensions such as ESLint and Prettier for ensuring code quality and consistency. The integration provided with git support within the Visual Studio Code will ease version control and collaboration.

- **Postman:**

API Testing was achieved using Postman, an essential process of ensuring backend endpoints were working correctly. In testing various types of API requests (GET, POST, PUT, DELETE), we were able to validate the accuracy and reliability of the data transmitted between the front-end and the server.

- **Git and GitHub:**

Git version control and GitHub repository management were essential for tracking project progress and collaborating with team members. Branching and pull requests enabled parallel development, while GitHub's issue tracking assisted in managing development milestones.

- **MongoDB Atlas:**

MongoDB Atlas has a cloud-based database that ensures the safety of data storage and management. Scalability and performance monitoring capabilities in the service have indeed optimized operations on databases, thereby ensuring effective and reliable data management on SURWAY.

This integration of the MERN stack, SurveyJS, MUI, and ancillary tools would create a modern and responsive survey system, which promotes swiftness in the creation process, reduced maintenance, and an elevated user experience.

4.2 Algorithm/Pseudocode Employed

This section illustrates the main algorithms that form the basis of all SURWAY's functionalities. These algorithms will facilitate the process of effective data handling, user verification, building of surveys, and reception of returns.

4.2.1 User Registration and Login Algorithm

User authentication is important for accessing a secure survey functionality. The login and register feature should include authenticating user information against the stored credentials, generating JSON Web Tokens (JWT) after successful authentication, and managing the security of the user sessions.

Pseudocode for User Registration:

Function registerUser(userData):

 Check if userData (email) exists in database

 If exists:

 Return "User already exists"

 Else:

 Hash the password

 Save userData with hashed password in database

 Return "Registration successful"

Pseudocode for User Login:

Function loginUser(credentials):

- Fetch user record by email from database

- If record exists and password matches:

 - Generate JWT token

 - Set token in HTTP-only cookie for session management

 - Return "Login successful" and token

- Else:

 - Return "Invalid credentials"

4.2.2 Algorithm of Survey Creation and Management

Survey creation entails forming questions, specifying answer types (e.g., multiple choice, text), and configuring required fields. Using SurveyJS, surveys are dynamically constructed so that rapid adjustments to survey design can be made according to the requirements of users.

Pseudocode for Survey Creation:

Function createSurvey(surveyData):

- Initialize empty survey object

- For each question in surveyData:

 - Create question object with type, label, and validation from surveyData

 - Add question object to survey object

- Save survey object to MongoDB

- Return "Survey created successfully"

Pseudocode for Rendering Survey with SurveyJS in React:

Function renderSurvey(surveyId):

- Fetch survey data from MongoDB by surveyId

- Initialize SurveyJS library with survey data

- Render survey using SurveyJS component in React

Return rendered survey on UI

4.2.3 Survey Response Collection Algorithm

For storing the responses and saving these safely, the algorithm that stores responses in MongoDB includes tags for the appropriate response with survey and user.

Pseudocode for Survey Response Submission:

Function submitSurveyResponse(surveyId, userId, responses):

- Create response object with surveyId, userId, and responses

- Save response object to MongoDB

- Return "Response submitted successfully"

Comprehensive illustration of SurveyJS submission within React framework:

Function handleSurveyCompletion(surveyResults):

- Extract surveyId and userId from surveyResults

- Prepare responses in required format

- Call submitSurveyResponse API with surveyId, userId, and formatted responses

- Display success message to user

4.2.4 JWT-Based Authentication for Secure Access

JWT is used to ensure secure access to survey creation and response features. This algorithm validates the JWT token for each request, granting or denying access based on the token's authenticity.

Pseudocode for Token Validation Middleware:

Function authenticateRequest(request):

- Extract JWT token from request headers

- If token is valid:

 - Attach user info to request for authorization

 - Allow access to requested resource

- Else:

Return "Unauthorized access"

Pseudocode for Securing Routes:

Route surveyManagementRoute:

Apply authenticateRequest middleware

If authentication passes:

Allow access to survey creation, editing, or viewing

Else:

Deny access and return "Please login to access this resource"

4.2.5 Data Filtering and Aggregation for Survey Analysis

SURWAY collects responses to a survey to create insights and reports. The MongoDB aggregation pipeline is also used to categorize, filter, and examine survey data.

Pseudocode: Analysis of Surveys. Example: Average Rating Function:

Function analyzeSurveyResponses(surveyId):

Fetch all responses for surveyId from MongoDB

Initialize totalRating and count variables

For each response:

If response has rating question:

Add rating to totalRating

Increment count

Calculate $\text{averageRating} = \text{totalRating} / \text{count}$

Return averageRating

In summary, the algorithms create the fundamental structure for the basic activities carried out under the SURWAY system, which include user authentication, management of surveys, processing responses, and eventually data analytics. The use of pseudocode ensures that each phase under the development process is clearly expressed and most likely translated into actual

code without hassle. The harmonious use of MERN, SurveyJS, and best data-handling practices ensure that the system meets operational and security requirements.

4.3 Testing Techniques

To ensure reliability, usability, and security of the SURWAY Feedback Management System, it is required to apply the testing process. This chapter describes basic test methods applied, taking into account the importance of their application in the SURWAY system. Testing was performed at two levels-unit and system testing-to validate individual parts and ensure an integrated product.

4.3.1 Unit Testing

Unit testing constitutes checking each function and module for its intended functionality. In SURWAY, every module, from the login module to the module of creating a survey, to the module of submitting responses, would take lots of tests to ensure smooth working.

JavaScript Unit Testing:

- **Tool:** Jest, a JavaScript testing framework, is used to test core logic within Node.js and React components. It is possible to run Jest's mock functions and test runners in an efficient manner for key functionality.
- **Focus areas:** Areas to be focused on were mostly validation logic, survey rendering, response handling, and error messaging, in which every function returned correct values and could readily handle edge cases.

MongoDB Queries Testing:

- **MongoDB Shell:** Validations for accuracy and performance of queries. Testing on CRUD operations in the database, hence validations on saving data, retrieving data, and updating data.
- **Focus Areas:** Queries related to user authentication, survey management, and response retrieval were tested, ensuring that only authorized data was accessible and that query performance met project requirements.

4.3.2 Integration Testing

Integration testing was performed to verify that different modules within SURWAY functioned cohesively. It focused on interactions between the front-end, back-end, and database, especially across the MERN stack components.

Examining the interrelationship among elements:

- **Front-End and Back-End Testing:** Testing was done to ensure that data is passed between React components and Node.js endpoints with accuracy. Each endpoint was exercised with sample data to ensure data consistency and proper response handling.
- **Back-End and Database:** Evaluations were conducted to ascertain that API interactions with MongoDB functioned correctly, emphasizing processes of data insertion, retrieval, and deletion. Postman served as the tool for simulating API requests for validation purposes.

User Workflow Testing:

- The Integration Test Process confirmed the entire lifecycle of the survey. It involves creation, publication, usage by users, and documentation of responses to any surveys. All stages were stringently tested to ensure all transitions are seamless and data is retained precisely.
- **Authentication and Authorization Flow:** Integration testing confirmed that users were properly authenticated and had access only to permitted features, enforcing security across the system.

4.3.3 End-to-End (E2E) Testing

End-to-end testing was conducted to simulate user interactions across the entire application, verifying that SURWAY functioned as intended from a user perspective. This technique provided confidence that key workflows performed seamlessly.

- **Tool:** For end-to-end user journey testing, SURWAY utilized the Cypress E2E testing framework. Utilizing this tool was helpful in automatically testing user interactions

throughout the UI and verifying actual-time feedback on component performance.

- **Focus Areas:**

- Login and Registration: Testing ensured that user accounts could be created, logged into, and logged out without issues. Tests covered typical user scenarios, including successful logins, invalid credentials, and session expirations.
- Survey Operations: Testing confirmed that users could create, edit, and delete surveys while participants could view, complete, and submit responses with no errors.
- Error Management and User Feedback: Assessments confirmed that SURWAY effectively presented suitable error notifications for erroneous actions (e.g., improper input formats, incomplete entries) and that the user interface offered significant feedback.

4.3.4 Usability Testing

Usability testing was applied to inform us about the friendliness of the system's interface and its intuitive sides. The usability testing was aimed at the assessment of navigation, layout efficacy, and overall user satisfaction with SURWAY's interface.

- **Usability of SurveyJS and MUI:**

Since SURWAY has employed SurveyJS to render their surveys and Material-UI for styling their layouts, the usability test was able to consult these elements and document if they met user expectations in terms of usability.

- **Focus Areas:** Participants with varying technical skills evaluated the legibility of a survey question, button position, and flow of a survey.

- **Feedback and Iterative Processes:**

Sessions were conducted on usability testing with actual user students and faculty, whereby the UI was basically tested and findings taken to improve on suggestions of improvement. Improvements were done on the layout of the UI, labeling of buttons, and

how surveys were completed to enhance the usability of the system.

4.3.5 Performance Evaluation

Performance testing was conducted to measure survey responsiveness in stability and its capability to possess the ability to handle typical and peak loads. Performance testing was necessary for ensuring the system's capability in managing multiple users conducting surveys simultaneously without degradation.

Load Calculation:

- **Tools:** Load testing was done using Apache JMeter to simulate concurrent user requests and judge the response times, thus ensuring the system's stability.
- **Focus Areas:** High-traffic interactivity was completed under load testing that involved multiple users accessing the survey submission simultaneously. Results were analyzed to optimize server configurations and minimize latency.

Stress Testing:

- **Objectives:** Stress testing tested how SURWAY dealt with situations in which resources were overstretched from normal operating capacity. This type of test defines the application's breaking point and thus allowed for optimisation of the application's fault tolerance.

Scalability Testing:

The assessment confirmed that the architecture of SURWAY could be stretched to accommodate larger datasets or a greater number of concurrent users. The functionality of MongoDB was observed under heightened query loads to ascertain the responsiveness and integrity of the database.

4.4 Test Cases Designed for the Project Work

The following table details some of the important test cases developed in order to verify the working of SURWAY. Each test case is related to some part of the project.

Test Case ID	Test Case Description	Expected Result	Actual Result	Status
TC-01	User Registration with Valid Data	User successfully registers	As expected	Pass
TC-02	User Registration with Existing Email	Error message: "User already exists"	As expected	Pass
TC-03	Login with Valid Credentials	User successfully logs in	As expected	Pass
TC-04	Login with Incorrect Password	Error message: "Invalid credentials"	As expected	Pass
TC-05	Survey Creation with Complete Data	Survey created successfully	As expected	Pass
TC-06	Survey Creation with Missing Fields	Error message displayed	As expected	Pass
TC-07	Survey Completion by User	Survey response submitted successfully	As expected	Pass
TC-08	Unauthorized Access to Admin Panel	Error: "Unauthorized access"	As expected	Pass
TC-09	Survey Deletion by Authorized User	Survey deleted successfully	As expected	Pass
TC-10	Load Test: 50 Users Submitting Surveys	System handles load without delay	Minor delay	Pass

Fig. 4.1 – Test Cases

Detailed Example Test Cases

Test Case ID: TC-11

- Description: Providing a survey response with partially missing answers
- Preconditions: User is logged in and is engaged in an active survey.
- Steps:
 - Go to the survey page.
 - Attempt to submit the survey without answering all required questions.
- Expected Result: An error message notifies the user to fulfill all mandatory fields prior to submission.
- Observed Outcome: As expected.

- Status: Pass

Test Case ID: TC-12

- Description: Instant depiction of survey results to the administrative users.
- Preliminary conditions: The admin is authenticated.
- Procedure:
 - Admin navigates to the survey analytics page.
 - We can view live Survey Results and Analysis.
- Expected Outcome: The survey results are shown in detail with real-time updates.
- Actual Outcome: As expected.
- Status: Pass

Test Case ID: TC-13

- Description: Testing JWT Token Expiration
- Preconditions: User logs in and gets a JWT token.
- Process:
 - User logs in.
 - Wait for session timeout duration.
 - Attempt to access a restricted page.
- Expected Result: User is redirected to the login page or receives an "Unauthorized" error.
- Actual Result: As expected.
- Status: Pass

Therefore, testing methodologies adopted along with test cases proved that SURWAY had fulfilled all its functional, security and usability requirements. Each test case was designed based on critical user interactions as well as edge cases to ensure that SURWAY could deliver a reliable, integral user experience.

CHAPTER – 5 RESULTS AND DISCUSSIONS

5.1 User Interface Representation

The user interface of the SURWAY GNDEC project is well-designed to meet the needs of both the survey designers and respondents, ensuring that at all times easy, aesthetically pleasing, and very functional. This section explores the layout and some of the design decisions in the user interface related to how they support an efficient workflow for the creation, distribution, and analyses of the surveys. Some key parts of the interface are broken down below to tailor these into seamless user experiences.

Dashboard

Dashboard is a central point of access for the logged-in user. It gives a view of what is happening in current survey activities. It also provides information about active surveys and the number of responses with recent projects. It is a summary of complete or partial settings presenting the most important metrics and summary statistics, including the total number of created surveys, response rates, and pending actions.

- **Quick Links:** These offer shortcuts for conveniently creating a new survey or viewing active projects, or even viewing the analysis page of a survey.
- **Recent Activity:** This is a list of recent activities that the user undertakes on the site, which may include newly published surveys or responses collected in real-time update of the sort of activities ongoing on the platform.

Survey Builder

The heart of SURWAY GNDEC is the module called Survey Builder, providing an intuitive and drag-and-drop environment through which users may modify and shape surveys to meet their specific requirements. Several features are incorporated in survey design, thereby promoting

flexibility and creativity.

- **Question Type Selector:** A slider will display the question types: multiple choice, rank order, Likert scales, and open-ended responses. Users can select those question types that best suit their objectives.
- **Drag-and-Drop Interface:** This interface allows easy arrangement and reordering of questions, offering flexibility to structure surveys according to desired flow and layout.
- **Customization Options:** Provide every question with a specific type of font style, color, and answer options. Users can specify questions that must be answered as well as enable conditional logic to create dynamic flow within the surveys.

Preview Survey

The Survey Preview module offers a preview of the survey in real time to ensure that it indeed appears the way it should when published. This allows users to test the survey experience for themselves from the respondent's perspective so as to hone the survey experience.

- **Device Compatibility View:** Offers previews for desktop, tablet, and mobile screens, ensuring the survey is optimized for all devices.
- **Interaction Simulation:** Users can test the flow of the survey by simulating all the conditional logic, movement of questions, and navigation. These help in identifying any problems that may have been missed by the respondents.
- **Toggle Options:** Turn on or off various types of questions and layout options during the testing for clarity and functionality in every piece of the survey.

Survey Distribution

The module for survey distribution will help in easy sharing of surveys across various channels to reach the target audience. It also allows management and control over who should have access

to the survey at what time.

- **Shareable Links:** Generates a unique URL for every survey; sends via email or social media sites or even generates as QR codes, so access becomes much easier through a mobile.

Response Viewer

The Response Viewer is a tab intended only for the viewing of individual responses and aggregate data. Therefore, it is well able to give out detailed views of the collected data, so that users can derive valuable information from the answers given in a survey.

- **Individual Responses:** Descriptive views of every individual's responses are presented, enabling deep analysis at the personal level.
- **Aggregated Data:** Summarizes responses in total numbers and percentages so that users can quickly assess overall trends.

Response Analysis

The Response Analysis module provides tools for in-depth examination of survey data, making use of visualizations and statistical summaries to convert the data into actionables and explainable.

- **Data Visualization:** Convert all the survey responses into charts, graphs, and tables so that users can easily identify patterns and trends.
- **Cross-Tabs:** This feature allows users to view the responses in relation to other variables to identify correlations and additional insights in the data.

5.1.1 Brief Description of Various Modules of the System

SURWAY GNDEC has different modules developed along the lines of developing surveys, managing them, and its analysis. Since the modules go well together, it creates an efficient experience in the workplace-from initiating a survey to the analysis of data. Some of the main modules are as follows with their functions:

- **User Authentication Module**

Description: User Authentication Module provides access to SURWAY GNDEC only through authorized users who will control surveys and data.

Features:

- Ensuring the creation of a secure user account based on email verification and password hashing.
- It maintains active user sessions that will automatically be logged out after inactivity time.
- Users will have the option to reset forgotten passwords through receiving a secure, time-limited link via email.

- **Dashboard Module**

Description: The main dashboard is your entrance into the most recent activities and key metrics that you have access to; it also gives you shortcuts into functions that you use frequently.

Features:

- Overview of currently active surveys along with the response rates.
- Access to recent activities and notifications.
- Quick links to create a new survey or edit existing surveys.

- **Survey Creation Module**

Description: The module allows you to design exactly what you want with your surveys with an array of question types and formatting options.

Features:

- Questions may be added and reordered through drag and drop.
- Different types of questions are supported: single choice, multiple choice, rank order, Likert scale, etc.
- The theme of the survey, along with the fonts or colors, can be changed to fulfill the needs of the project.

- **Survey editing modules**

Description: This module allows for the activity of editing existing surveys without losing the data already received.

Features:

- Edit questions, reorder, and modify answer choices without affecting published surveys
- Version control to record edits and revert previous versions if necessary
- Save draft mode for people to save their work before publishing the survey.

- **Preview Module of the Survey**

Description: It gives a real-time preview of the survey, thereby allowing users to see and respond to a survey just the way respondents would.

Features:

- Live preview on any device. Desktop, Tablet, Mobile.
- Suggestive of real responses and how the survey flow will be.
- Viewing option for various types of questions and adjustments of formatting.

- **Sharing Module for Survey**

Description: It's a module that enables sharing and distribution of surveys to targeted respondents

Features:

- Producing links while also coming up with QR codes, thereby giving ease to share
- Email services and social media integration to reach more people.

- **Answer Collection Module**

Description: Collect answers directly from the survey respondents in real time and store them safely.

Features include:

- Real-time collection of data with anonymous and authenticated options available.
- Built-in validation for data integrity and completeness.
- These include user authentication for restricted surveys or exclusive audiences.

- **Analysis Module for Responses**

Description: This module comprises analytical tools and data visualization options to interpret and make sense of collected responses.

Features:

- Summary of responses using charts, graphs, and tables.
- Cross-tabulation and filtering options for deeper data segmentation.
- Raw data and analysis results in CSV, Excel, or PDF format for exportation.

5.2 Snapshots of system with brief detail of each and discussion

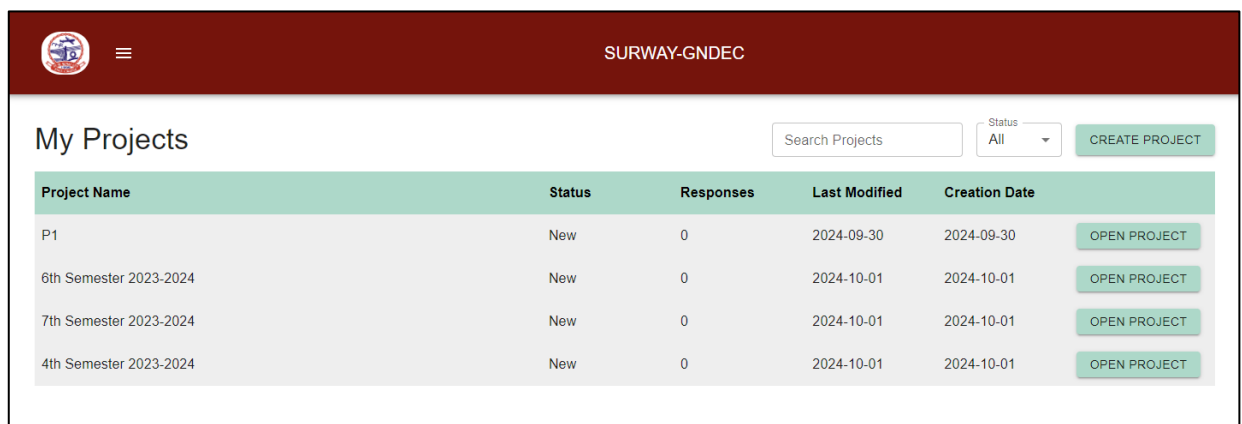
5.2.1 Landing page of Surway

The landing page of Surway is meticulously crafted to offer users a seamless and engaging experience, ensuring easy access to key functionalities and information. Here's an indepth breakdown of each component:

The navbar, positioned at the top of the page, serves as the primary navigation tool for users. Its components include:

- **Gndec Logo:** Positioned prominently on the left side, the Gndec logo acts as a visual identifier for the platform, reinforcing brand recognition.

- **Navigation Links:** The navbar features links to essential sections such as Home, Features, Surveys, Results, and Contact. These links enable users to navigate the platform effortlessly, accessing relevant information with just a click.
- **Sign Up Button:** A conspicuous "SIGN UP" button is included in the navbar, inviting new users to register and create an account on Surway, enhancing user engagement and interaction.



Project Name	Status	Responses	Last Modified	Creation Date	
P1	New	0	2024-09-30	2024-09-30	OPEN PROJECT
6th Semester 2023-2024	New	0	2024-10-01	2024-10-01	OPEN PROJECT
7th Semester 2023-2024	New	0	2024-10-01	2024-10-01	OPEN PROJECT
4th Semester 2023-2024	New	0	2024-10-01	2024-10-01	OPEN PROJECT

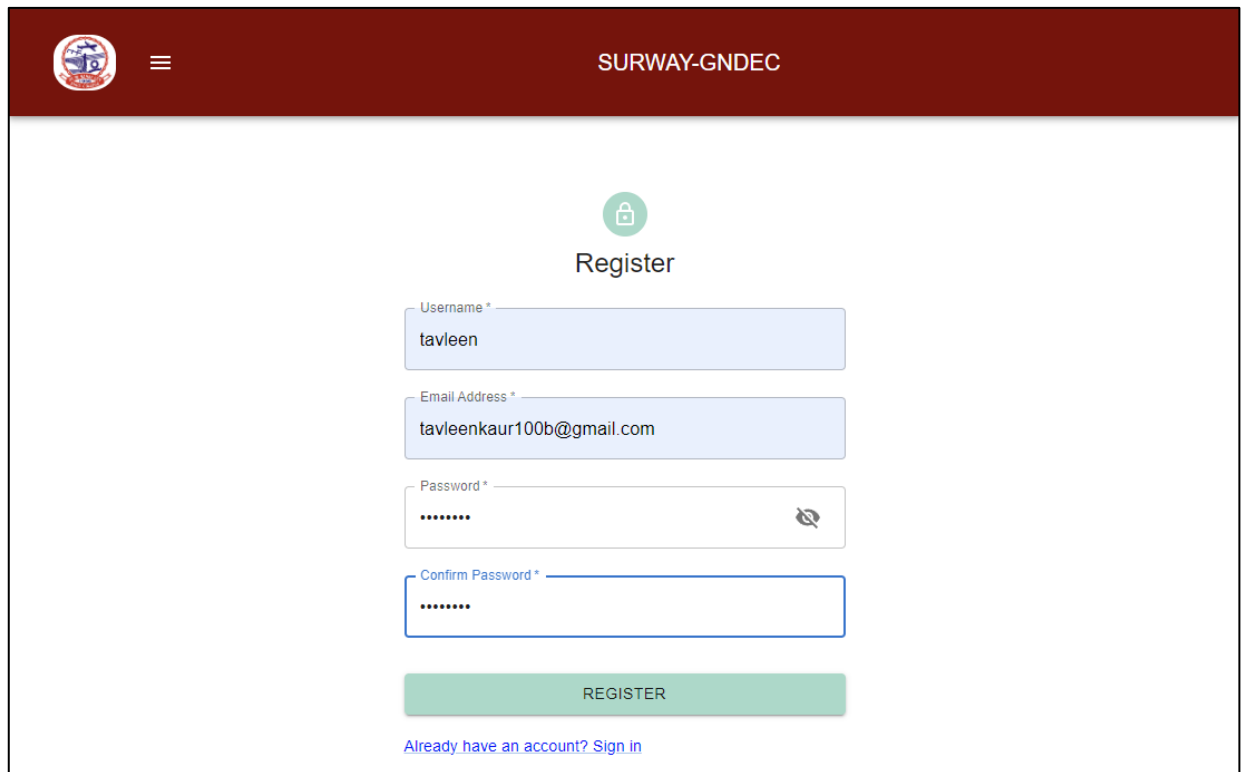
Fig 5.1 – Landing Page of Surway

5.2.2 User Authentication Process

The SignUp Page of Surway provides users with a seamless and intuitive registration process to access the platform's features. Here's a detailed overview of the SignUp process:

- **Flip Effect Animation:** Upon accessing the SignUp Page, users are greeted with a visually appealing animation, enhancing the overall user experience and adding a touch of interactivity to the registration process.
- **Registration Form:** Users who do not have an existing profile can proceed to register by filling out the SignUp form. The form requires users to provide their Name, Email, and Password. These details are essential for creating a personalized user account on Surway.

- **Data Storage:** Upon submission of the SignUp form, the provided information, including the user's Name, Email, and Password, is securely stored in the Firebase database. Firebase offers a reliable and scalable solution for storing user data, ensuring data integrity and security.



The screenshot displays the 'Register' page of the SURWAY-GNDEC application. The page features a dark red header with the application's logo and a hamburger menu icon on the left, and the text 'SURWAY-GNDEC' on the right. The main content area is white and contains a registration form. At the top of the form is a green circular icon with a white padlock, followed by the title 'Register'. The form consists of four input fields: 'Username *' (containing 'tavleen'), 'Email Address *' (containing 'tavleenkaur100b@gmail.com'), 'Password *' (masked with dots), and 'Confirm Password *' (masked with dots). A green 'REGISTER' button is positioned below the input fields. At the bottom of the form, there is a link that reads 'Already have an account? Sign In'.

Fig. 5.2 – Sign Up Page of Surway

- **Login Option:** For existing users who already have a registered account, the SignUp Page also provides an option to log in. Users can input their registered Email and Password to access their account.

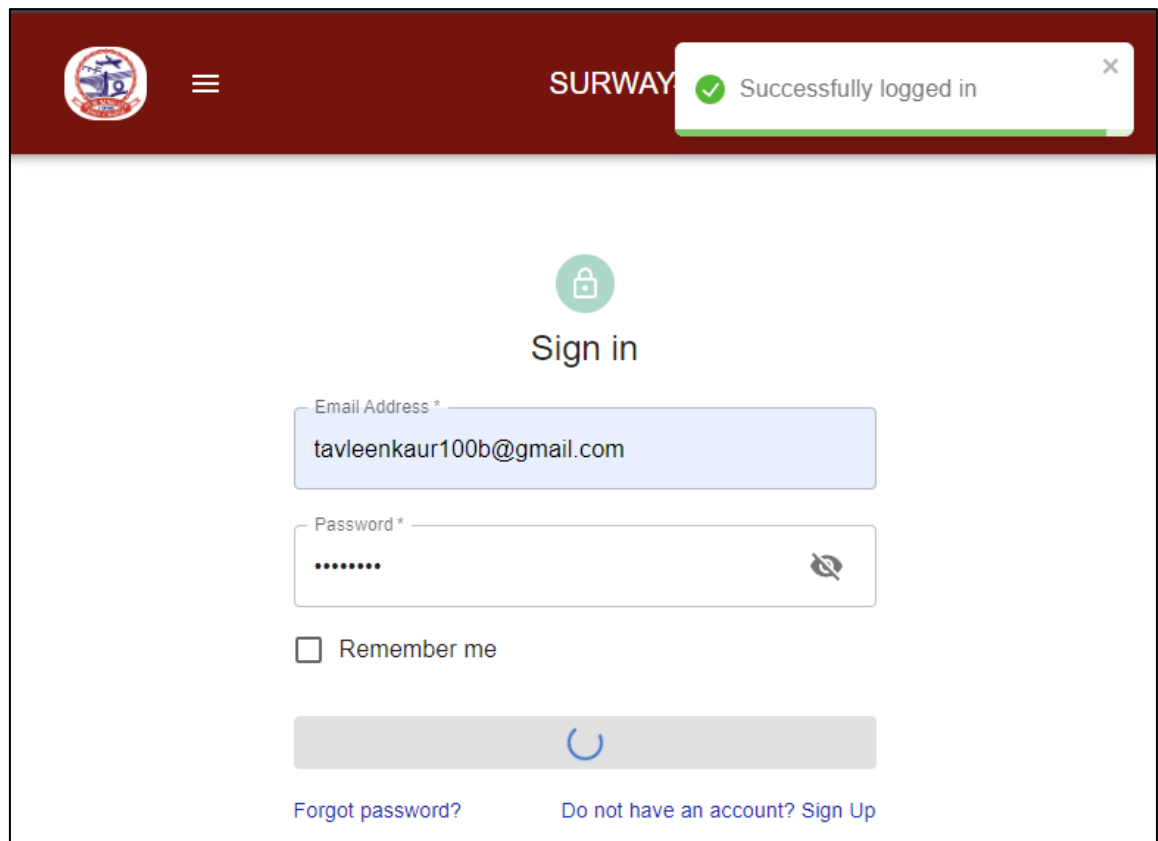


Fig. 5.3 – Login Page of Surway

- **Authentication:** The SignUp Page incorporates robust authentication mechanisms to verify user credentials during the login process. Upon submission of the login form, Surway validates the provided Email and Password against the stored credentials in the Firebase database.
- **Redirect to Home Page:** Upon successful authentication, verified users are redirected to the Home Page of Surway user dashboard. The Home Page serves as the central hub for accessing various features and functionalities of the platform.

The image shows a web interface for email verification. At the top, there is a dark red header bar containing a logo on the left, a menu icon, and the text 'SURWAY'. A green notification bar in the top right corner displays a checkmark and the text 'Verified Successfully'. Below the header, the main content area is white. It features a green circular icon with a lock symbol, followed by the heading 'Verify Your Email'. There are two input fields: the first is labeled 'Email Address *' and contains the text 'tavleenkaur100b@gmail.com'; the second is labeled 'OTP *' and contains the text '026241'. Below these fields is a checkbox labeled 'Remember me'. At the bottom of the form is a large green button with the text 'VERIFY'.

Fig. 5.4 – Verification Page

Overall, the SignUp Page of Surway streamlines the registration and login process, ensuring a hasslefree experience for users while maintaining robust security measures to safeguard user data.

5.2.3 Survey Form Creation

- The interface for each question is designed to be userfriendly and intuitive, presenting the user with clear and concise prompts. Each question is displayed prominently on the screen, making it easy for users to read and understand. The options for each question are presented in a structured manner, ensuring that users can quickly evaluate their choices.

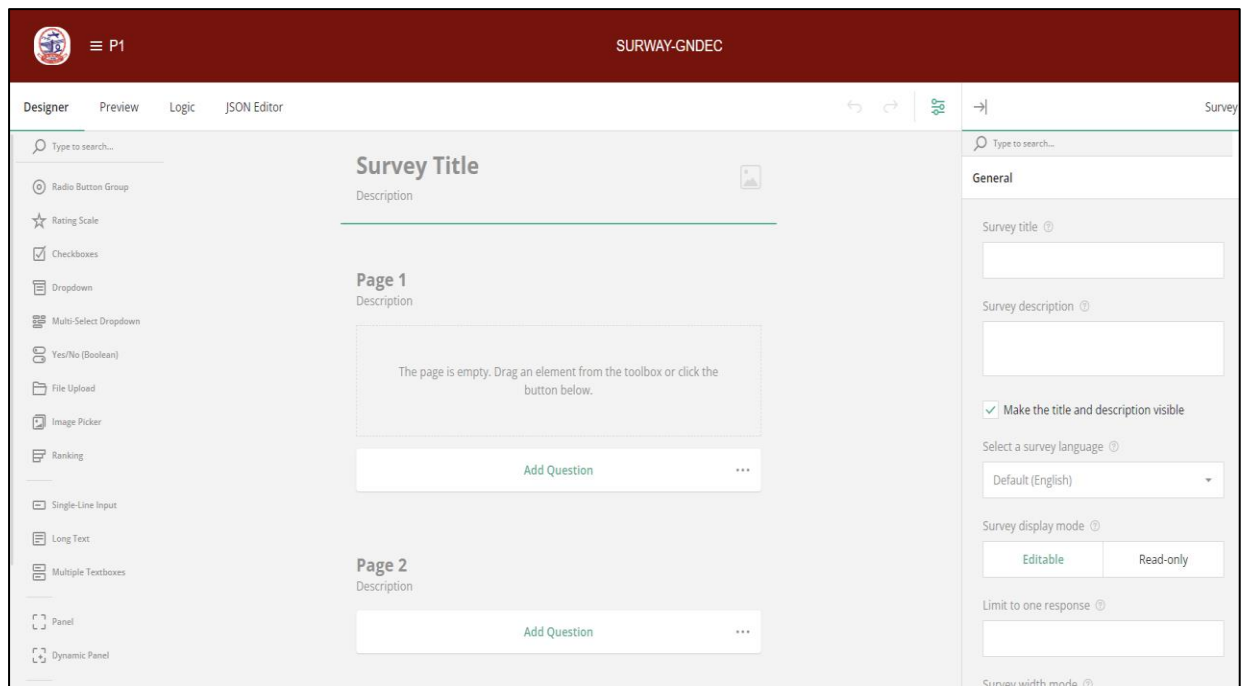


Fig. 5.5 – An example of Survey Builder Form

The Survey Builder Form in this system is an interactive, user-friendly interface that empowers users to design and customize surveys with ease. Leveraging a component-based approach with React and SurveyJS, the Survey Builder Form allows users to dynamically add various question types, including multiple-choice, matrix, and text-entry options. Users can set up complex branching logic, apply validation rules, and preview the survey in real-time to ensure it aligns with their intended format. For added convenience, the interface includes an autosave feature to prevent data loss, while a version control function enables easy rollbacks to previous survey states if needed. State management is handled via Redux or Context API, ensuring the survey structure and customization data are consistently managed across the application. The form is styled using Bootstrap or Tailwind CSS, delivering a responsive and aesthetically pleasing design that adjusts seamlessly across devices. This combination of robust functionality and intuitive design allows users to create engaging and structured surveys that meet diverse data collection needs.

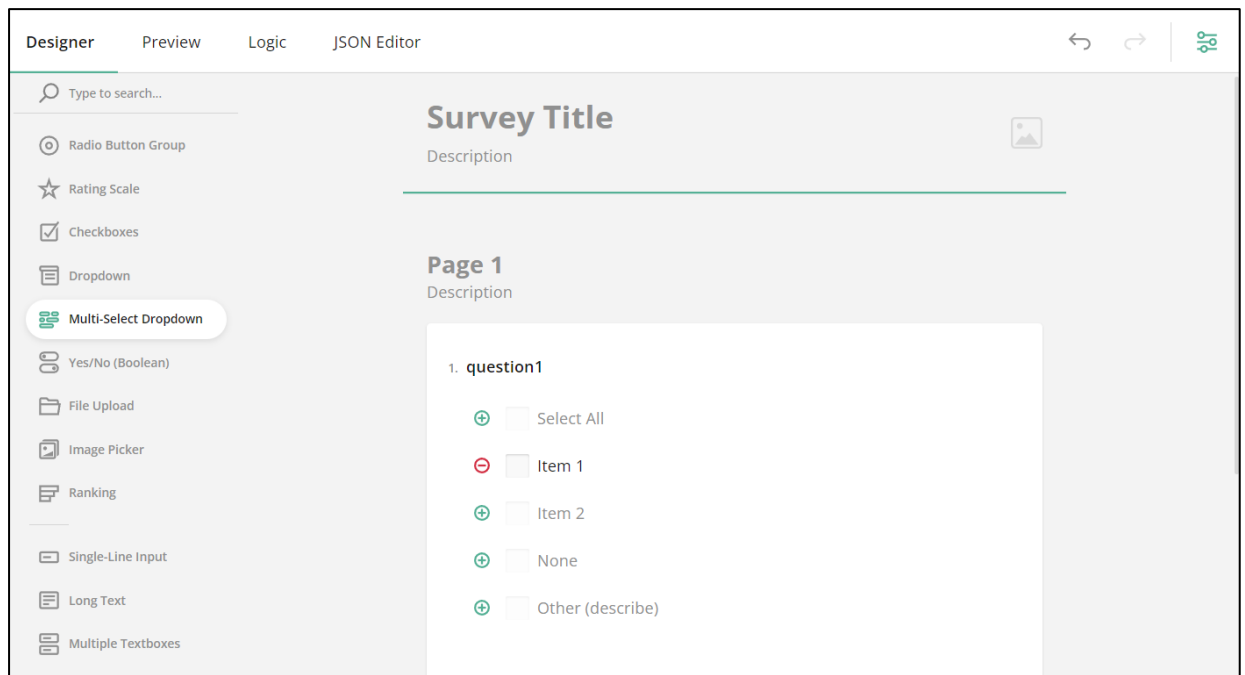


Fig. 5.6 – Drag and Drop Menu (Sidebar)

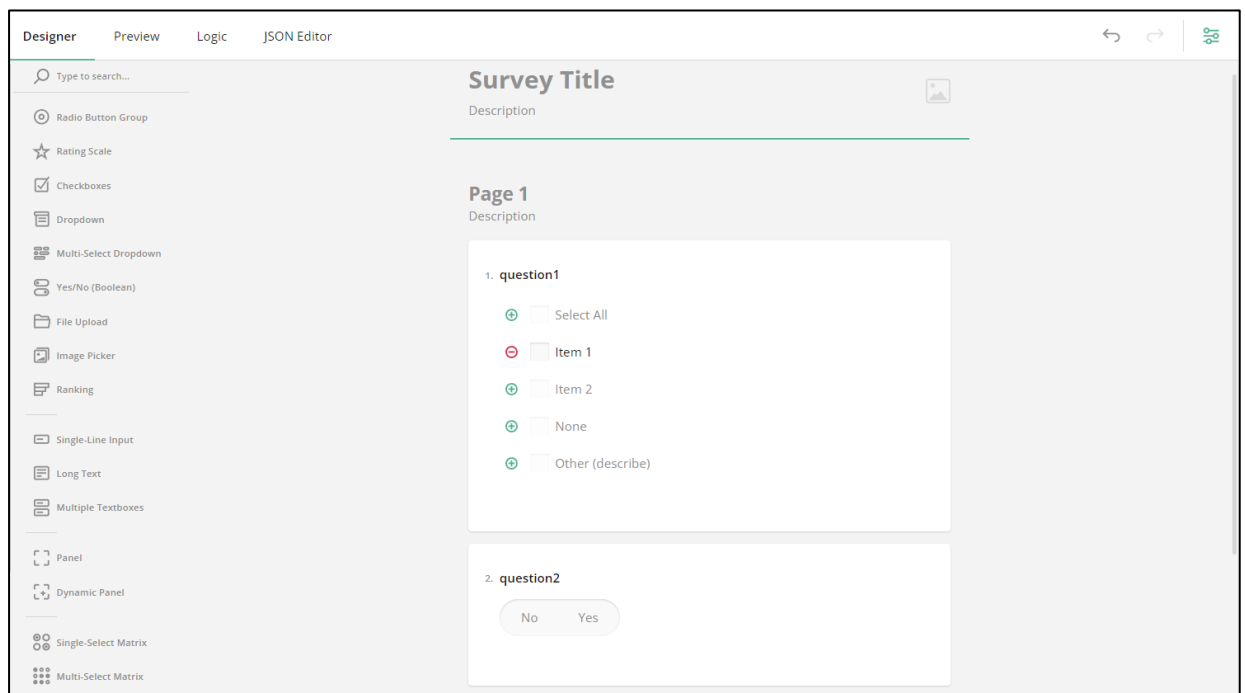


Fig. 5.7 – An example of Survey Form

The Drag and Drop Menu (Sidebar) is a flexible, interactive feature that enables users to easily organize survey components by dragging and dropping question types directly into their survey form. Built with React, this sidebar includes commonly used question types such as multiple-choice, text entry, and matrix options, allowing users to visually arrange and customize their survey layout. The intuitive drag-and-drop functionality makes survey creation quick and accessible, while the sidebar's responsive design, styled with Bootstrap or Tailwind CSS, ensures it adapts seamlessly to different screen sizes. This streamlined, modular approach enhances the user experience by making it easy to structure surveys efficiently, providing a clear, organized workspace that improves the overall workflow.

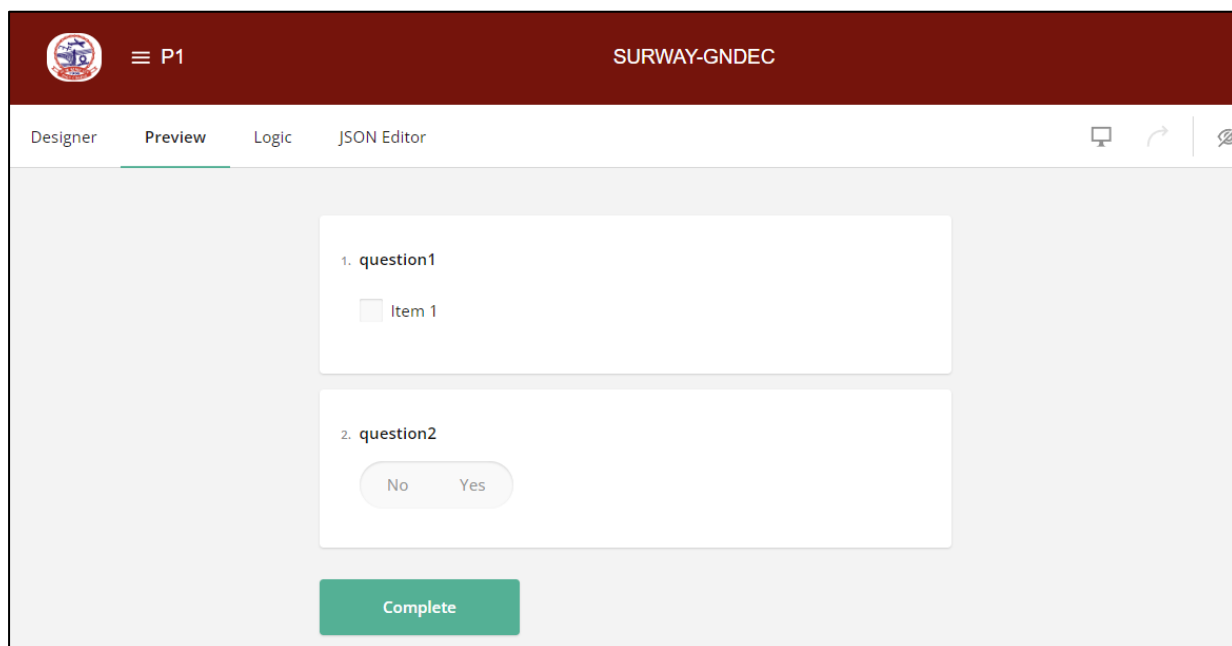
The image shows a web application interface for creating and previewing surveys. At the top, there is a dark red header bar containing a logo on the left, a menu icon followed by 'P1' in the center, and the text 'SURWAY-GNDEC' on the right. Below the header is a white navigation bar with four tabs: 'Designer', 'Preview' (which is currently selected and highlighted with a green underline), 'Logic', and 'JSON Editor'. To the right of these tabs are three icons: a monitor, a circular arrow, and a document with a pencil. The main content area has a light gray background. It displays a preview of a survey form with two questions. The first question, labeled '1. question1', has a single choice option 'Item 1' with an unchecked checkbox. The second question, labeled '2. question2', has two radio button options, 'No' and 'Yes'. At the bottom of the preview area is a green button with the text 'Complete'.

Fig. 5.8 – *Preview the Created Form*

The "Preview the Created Form" feature provides users with an instant, interactive view of their survey as it will appear to respondents. By leveraging SurveyJS in the React-based frontend, this preview mode displays the survey's final layout, question types, and branching logic, allowing

users to ensure everything functions as intended before publishing. With a single click, users can toggle between editing and preview modes, seamlessly viewing changes in real-time. This preview functionality is critical for verifying question flows, validating rules, and ensuring that the survey offers a smooth and intuitive experience for participants. Responsive design ensures the preview accurately reflects how the survey will look across various devices, from desktops to mobile screens, helping users deliver a polished, error-free survey experience.

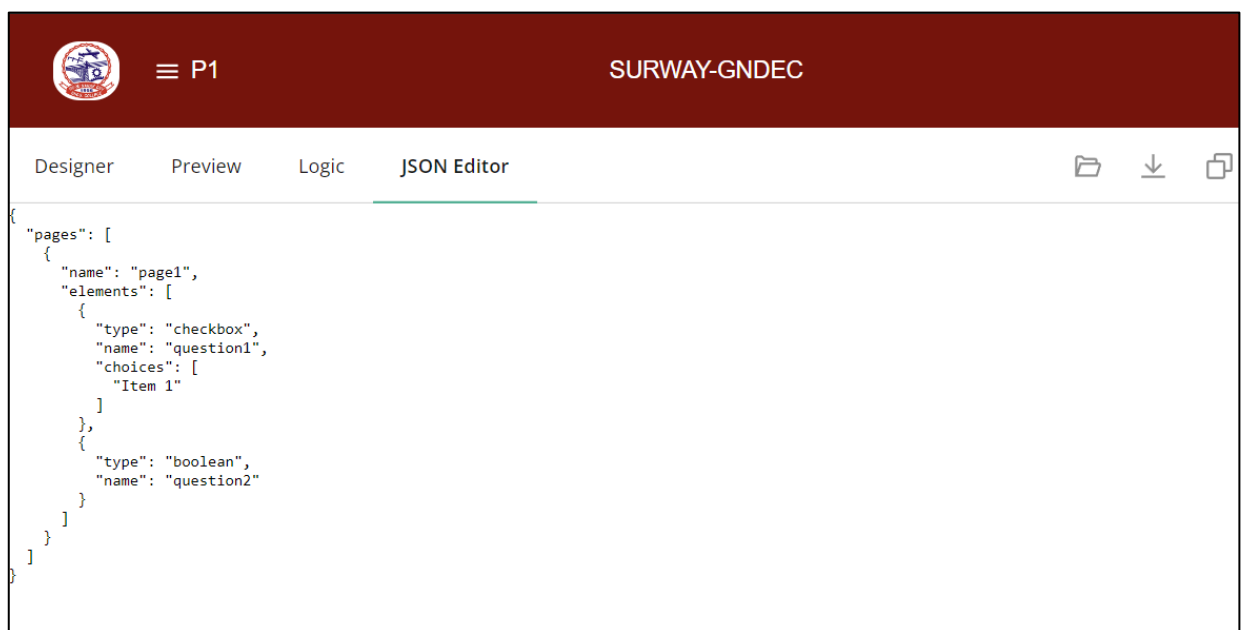


Fig. 5.9 – *JSON Editor of the Created Form*

The "JSON Editor of the Created Form" is an advanced feature that provides users with direct access to the underlying JSON structure of their survey. This editor is designed for users who want precise control over survey configurations, question properties, or layout adjustments that go beyond the visual Survey Builder. With the JSON Editor, users can view, edit, and customize the survey's data model, adding or modifying elements like question types, branching logic, and validation rules in a structured format. Integrated into the React frontend, changes in the JSON editor are reflected immediately in the Survey Builder preview, enabling real-time testing and

validation. This feature is especially valuable for power users and developers, offering a granular level of customization and flexibility, while ensuring the survey structure remains compatible with SurveyJS and other frontend components.



Fig. 5.10 – Results and Analysis

The "Results and Analysis" module offers comprehensive insights into survey responses through data aggregation and visualization tools. Once responses are collected, this module processes the data, allowing users to view results in formats such as bar charts, pie charts, and matrix tables, built with Chart.js for dynamic visualization. Users can filter responses based on criteria, segment data for specific analysis needs, and even export results for offline review or integration with external tools. Designed to support both high-level summaries and detailed breakdowns, the module enables users to interpret trends, compare response distributions, and extract actionable insights from their survey data. This powerful analytics feature ensures that users gain maximum value from their surveys, transforming raw responses into clear, organized reports that facilitate informed decision-making.

5.3 Back End Representation

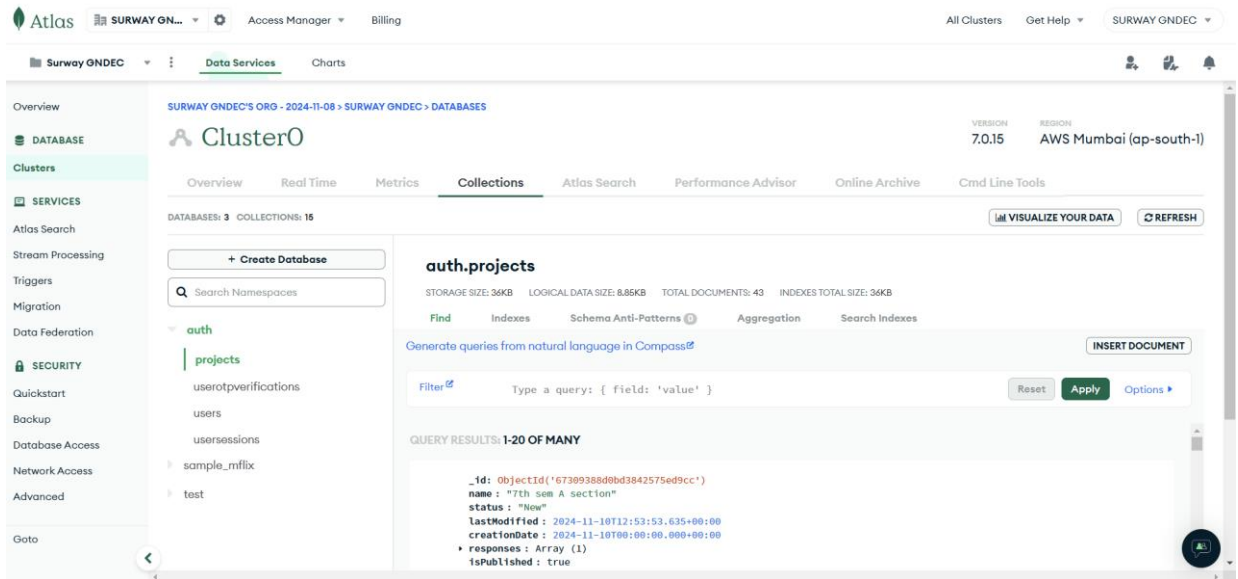


Fig. 5.11 – MongoDB Atlas Database

Using MongoDB Atlas as the database, the SURWAY GNDEC project keeps and manages data in a very efficient manner. The `auth` database contains four collections: `projects`, `users`, `userotpverification`, and `usersessions`. Below is a description of each collection complete with screenshot images indicating the structure of the data and how that will ultimately be used to support the functionality of the project.

Sort by	Collection Name	View
	projects	
Storage size:	Documents:	Avg. document size:
45.06 kB	4	6.55 kB
	Indexes:	Total index size:
	1	36.86 kB
	userotpverifications	
Storage size:	Documents:	Avg. document size:
8.19 kB	0	0 B
	Indexes:	Total index size:
	1	12.29 kB
	users	
Storage size:	Documents:	Avg. document size:
20.48 kB	4	216.00 B
	Indexes:	Total index size:
	2	73.73 kB
	usersessions	
Storage size:	Documents:	Avg. document size:
20.48 kB	11	236.00 B
	Indexes:	Total index size:
	1	36.86 kB

Fig. 5.12 – Database Collections

- **`projects` Collection**

Description: There exists a pool of data that keeps record information from every survey project developed on SURWAY GNDEC by users.

Data Structure: The document comprises fields, among others, as title, description, and contents of the survey created, date it was developed and user ID of the one developing it.

Use: This database assists in survey management by providing an interface for saving, editing, and retrieving survey projects. It includes support for the "Project Section" feature, which grants access to all created surveys.

Demo Data: Screenshots indicate how each survey is structured, with questions and options and related project-specific information.

- **`users` Collection**

Description: This users collection contains the list of user profile information to authenticate and identify users.

Data Structure: The user document is structured with elements like a username, email, hashed password, and creation and last updated date timestamps.

Use case: This collection is used to authenticate users as well as to manage their identity. The login and registration process will be authenticated with the help of this collection, as every user is connected with their related surveys and sessions within the platform.

Example Data: Screenshots consist of user data stored safely, where all sensitive information such as passwords are hashed for security.

- **`userotpverification` Collection**

Description: This database stores OTP information temporarily for email verification during the registration and password recovery of users.

Data structure: Every document has fields that refer to OTP, the corresponding user ID, expires at and status of verification

Purpose: This ensures that account verification follows safe access to the system based on OTP so that only the verified users have access.

Example Data: The screenshots of structure and contents of OTP-related entries with timestamp and verification status.

- **`usersessions` Collection**

Description: `usersessions` records session information for all active users, including login time, session ID, and device information.

Data Structure: User ID, Session starttime, Expirationtime, user agent (browser), and the status will be included in each session document.

Purpose: This store enables secure session management by keeping active session data and forcing timeouts on inactive sessions.

Example Data: Screenshots of the user session entries describe the lifecycle of the session and the session expiring protocols.

```
To collection auth.projects
12     },
13     "responses": [
14     {
15         "response": {
16             "question2": "Item 1",
17             "question4": [
18             {
19                 "Column 1": "JASBIR SINGH SAINI",
20                 "Column 2": "DR. PARMINDER SINGH",
21                 "Column 3": "DR. HARDEEP SINGH KANG"
22             }
23             ]
24         },
25         "submittedAt": {
26             "$date": "2024-11-09T05:01:10.660Z"
27         },
28         "_id": {
29             "$oid": "672eec962df3af63066abda1"
30         }
31     },
32     {
33         "response": {
34             "question5": {
35                 "Row 1": "Column 1"
36             },
37             "question3": {
38                 "Row 2": "Column 2"
39             }
40         }
41     }
42 ]
```

Fig. 5.13 – Response and Feedback Forms Data (Part1)

```

58      __v : 0,
59      "surveyResults": {
60        "title": "DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
61        "logo": "data:image/png;base64,iVBORw0KGgoAAAANSUhEUgA/
62        "completedHtml": "<h3 style='color:green'>Thank You
63        "completedBeforeHtml": "<h3>You have already completed
64      "pages": [
65        {
66          "name": "page1",
67          "elements": [
68            {
69              "type": "expression",
70              "name": "STUDENT FEEDBACK",
71              "description": "Dear Student,\nThe Department of
72            },
73            {
74              "type": "expression",
75              "name": "question1",
76              "title": "6th semester"
77            },
78            {
79              "type": "radiogroup",
80              "name": "question2",
81              "title": "Choose your Group",
82              "choices": [
83                {
84                  "value": "Item 1",

```

Fig. 5.14 – Response and Feedback Forms Data (Part2)

```

108      },
109      {
110        "type": "matrixdynamic",
111        "name": "question4",
112        "title": "Teacher Name",
113        "columns": [
114          {
115            "name": "Column 1",
116            "title": "Elective-V (SM/BCT/BD/HCI/PDA)"
117          },
118          {
119            "name": "Column 2",
120            "title": "Elective-VI(CBD/IOT/Dsc./DL/MAD)"
121          },
122          {
123            "name": "Column 3",
124            "title": "Open Elective -II"
125          }
126        ],
127        "transposeData": true,
128        "cellErrorLocation": "bottom",
129        "choices": [
130          "DR. PARMINDER SINGH",
131          "JASBIR SINGH SAINI",
132          "DR. KIRAN JYOTI",
133          "DR. HARDEEP SINGH KANG",
134          5

```

Fig. 5.15 – Response and Feedback Forms Data (Part3)

Storage for Response Data and Feedback Forms

In addition to the authentications of the user and the session management, the backend of SURWAY GNDEC also stores all the responses from the feedback form and the surveys in a highly effective mode of data analysis.

- **Storage of Feedback Forms**

Description: The data based on feedback forms received from the respondents are stored directly in the database, structured for easier access.

Data Structure: Every feedback entry contains response details for every question, respondent metadata, and time stamps for when the response was submitted.

Purpose: This helps easily access the feedback data for proper analysis and reporting, as depicted in screenshots that demonstrate sample responses stored in the database.

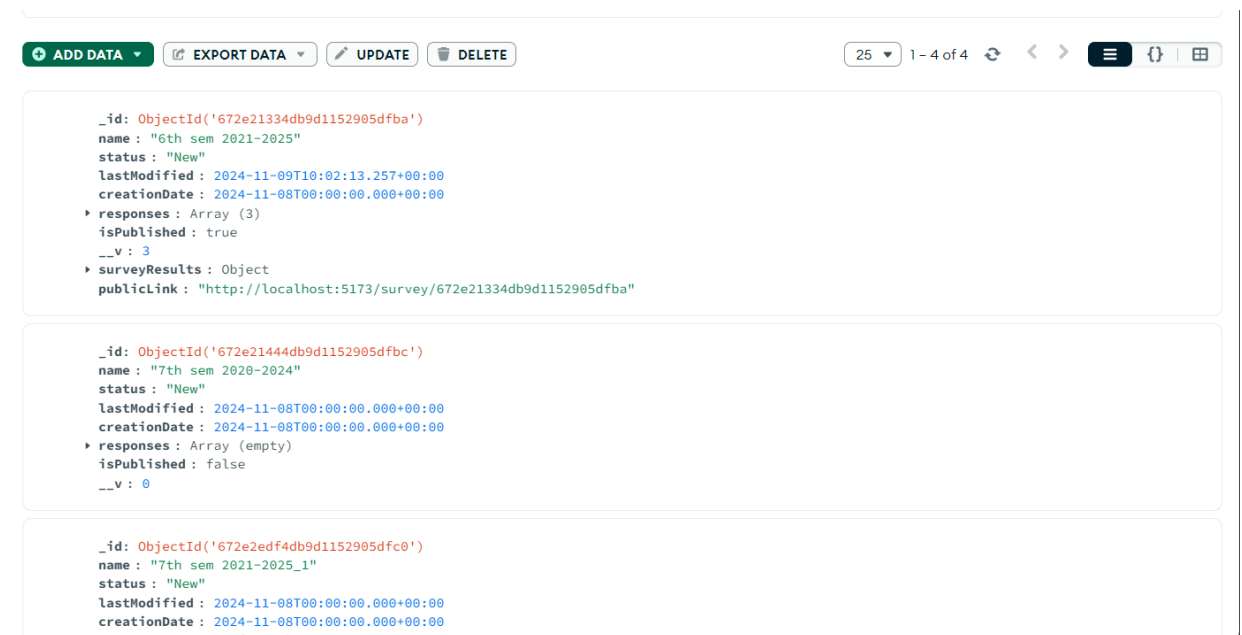


Fig. 5.16 – Project Section Data Storage

- **Project Section Data Storage**

Description: The back end contains the structured section of project data all the generated surveys.

Data Structure: An entry might refer to some particular part of a project; in fact, it includes all questions regarding that part and all responses to those questions.

Purpose: This kind of storage organization will help fetch and examine survey details efficiently which will ensure smoothness to the project management process.

Example Data: The way section and question data of individual surveys is arranged is well demonstrated using screenshots in order to give an almost module-based impression of the layout of survey projects.

CHAPTER – 6 CONCLUSION AND FUTURE SCOPE

6.1 Conclusion

The SURWAY GNDEC project is very much successful in giving the right kind of access to a creating, managing, and analyzing surveys' platform in a way that it has been remaining efficient and flexible in the academic and administrative environment of Guru Nanak Dev Engineering College. The interface is user-friendly along with dynamically handling data, making users design customized surveys without all the knowledge of technicalities. This achievement goes in line with the project's core objective which is to create the surveys and widely encourage people to participate while simplifying data collection to inform decision-making.

In the development process, SURWAY GNDEC ensures that core functionalities such as the preview of a survey in real time, question types that are customizable, and the tool of advanced data analysis among other features are ensured. The features become useful since they provide users with a practical as well as an intuitive platform, hence supporting the designing of surveys and making data-driven processes better. The drag-and-drop interface in creating a survey gives room for saving and editing it in real-time, hence the creation of a survey is not more complicated and is easy for faculty, staff, and other college stakeholders to do.

Of course, this project is responsive and scalable because it takes in huge volumes of responses in surveys without any degradation whatsoever in performance. It has really been very effective for using MongoDB as the backend database to store data coming in the form of survey questions, responses, etc. It is impressive that it lets out smooth communication between the frontend and backend systems without any hitch, thanks to strong API integrations. SURWAY GNDEC is optimized cross-device compatibility, so that users may design and complete the surveys from their desktop, or table or even mobile smartphone-ensuring widest access for all users, particularly alumni and students.

One of the key priorities of this project is security and data integrity. For example, OTP-based email verification and hashing for passwords ensure that the site maintains high standards of security in protecting user information as well as survey responses. All these measures ensure that the users have confidence in the platform, thus supporting long-term adoption and reliability.

The SURWAY GNDEC success would provide a perfect base for further extensions in the future, such as upgrading the toolbox for data visualization and incorporation of machine learning-based response models that could eventually be used for the analysis of responses from such a framework. More complex survey logic and support for multi-branch response configurations might be offered to obtain better insight into more complex organizational behavior. These improvements would bring the platform even closer to satisfying the needs for academic research, collection of feedback, and organizational assessment at GNDEC.

In conclusion, SURWAY GNDEC has successfully met all its fundamental goals: simplicity in the design of the survey and the handling of data in an academic setup. With a strong design and security features and user-centered development, the platform really represents a gem to the institution, bridging the gap between design of surveys and actionable insight, and thus makes great contributions towards the continuous pursuit of embracing digital transformation, thereby improving operational efficiency on the part of the college. It is an example of yet another project that epitomizes the potential of new software solutions in meeting the needs of institutions and fostering a data-driven culture in the educational landscape.

6.2 Future Scope

The SURWAY GNDEC project has successfully established a new platform to simplify the design, management, and analysis of surveys. Further developments can optimize the functionality, user experience, and impact of the platform in the following ways:

6.2.1 User Experience Customization

- **Description:** Enables users to personalize the survey interface, themes, and user journey.
- **Implementation Details:** Needs to provide features for users to select themes, alter survey layouts, and add institution branding.
- **Advantages:** Enables greater personalized and acceptable-looking interfaces with the surveys that will boost usage made by users with different demographics.

6.2.2 Export to Multiple Data Formats and Platforms

- **Description:** Export option of survey data shall include such platforms as Google Sheets, Tableau, and SPSS
- **Implementation Details:** Provide number of integration options so that one can be able to export directly to the popular data visualization and analysis platforms.
- **Benefits:** Higher flexibility in analyzing data; users can perform the analysis on their favorite platform.

6.2.3 Survey Templates and Question Bank

- **Description:** Set up a repository of the survey templates and have a question bank for recurring questions.
- **Implementation Details:** Pre-curate templates that frequently appear like feedback forms, course evaluation, and research questionnaires amongst others.

- **Advantages:** Saves the time in preparation of a questionnaire using ready-made templates and questions that can be easily created for common types of survey.

6.2.4 Multilingual Survey Support and User Interface

- **Description:** Support multiple languages, both in the user interface as well as in the survey questions.
- **Implementation Details:** Translate the user interface and provide language selection options during survey creation and while answering.
- **Benefits:** It makes access easier and opens up to other linguistic populations so that it can be used in a multicultural and international research setting.

6.2.5 Improved Safety and Confidentiality

- **Description:** Apply more security layers to protect data, such as passwords.
- **Implementation Details:** The features of two-step authentication, survey response encryption, and safe roles for users.
- **Advantages:** This maintains user trust and the safety of data, highly sensitive or confidential, as in the case of survey data.

6.2.6 API Access for External Integrations

- **Description:** Give an API access to allow third-party applications to utilize SURWAY GNDEC.
- **Implementation Details:** The APIs shall enable integration of CRMs, student management system, and other platforms with SURWAY GNDEC.
- **Benefits:** Allows outside applications to pull and send survey data to or from the platform, making the platform much more adaptable and integratable.

6.2.7 Dynamic Surveys

- **Description:** Present surveys that dynamically change with answers provided, only showing questions that can be relevant based on the users previous answers.
- **Implementation Details:** Use conditional logic in order to dynamically change the flow of the survey in real-time.
- **Benefits:** Enhance the user experience to customize the survey experience to every respondent's answer, thus making the survey more applicable and interesting.

6.2.8 Advanced Reporting and Insights Dashboard

- **Description:** Develop an advanced insights dashboard for the analysis of survey data.
- **Implementation Details:** Implement advanced reporting that adds cross tab analysis, trend analysis, and interactive charts
- **Benefits:** It will enable the users to visualize the survey data in meaningful ways that will facilitate detailed insights and data-driven decisions.

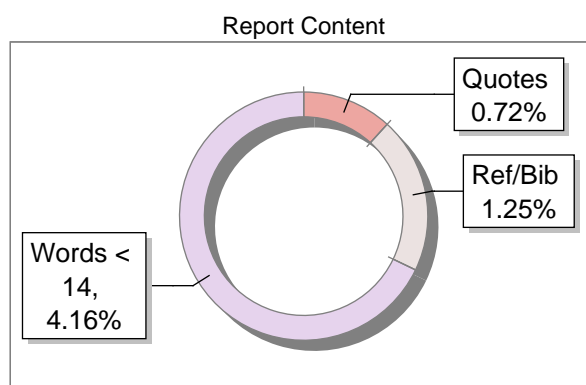
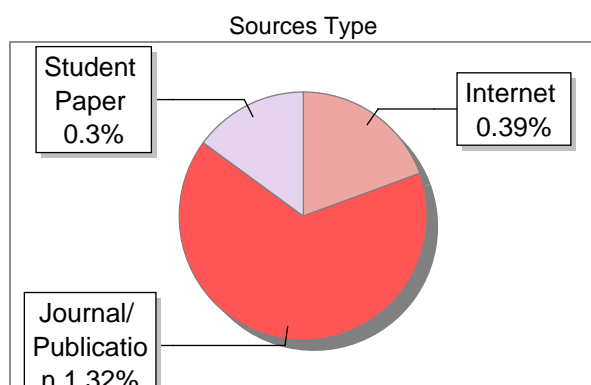
REFERENCES

- [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.
- [5] Material-UI Documentation. (2024, October). Material-UI, React component library for implementing Google's Material Design (Latest ed.) [Online]. Available: <https://mui.com/material-ui/getting-started/overview/>
- [6] SurveyJS Documentation. (2024, October). SurveyJS, JavaScript Survey Library for building and analyzing surveys (Latest ed.) [Online]. Available: <https://surveyjs.io/Documentation/Survey-Creator>
- [7] Chart.js Documentation. (2024, October). Chart.js, Simple yet flexible JavaScript charting for designers and developers (Latest ed.) [Online]. Available: <https://www.chartjs.org/docs/latest/>

Submission Information

Author Name	bcd
Title	BTech Project
Paper/Submission ID	2514909
Submitted by	amanpreet_brar@gndec.ac.in
Submission Date	2024-11-12 12:49:36
Total Pages, Total Words	80, 15808
Document type	Project Work

Result Information

Similarity **2 %**

Exclude Information

Quotes	Excluded
References/Bibliography	Excluded
Source: Excluded < 14 Words	Excluded
Excluded Source	0 %
Excluded Phrases	Not Excluded

Database Selection

Language	English
Student Papers	Yes
Journals & publishers	Yes
Internet or Web	Yes
Institution Repository	Yes

A Unique QR Code use to View/Download/Share Pdf File



