

RAJIV GANDHI INSTITUTE OF TECHNOLOGY  
GOVERNMENT ENGINEERING COLLEGE  
KOTTAYAM-686 501



DEPARTMENT OF  
COMPUTER SCIENCE AND ENGINEERING

CSD 334 MINI PROJECT REPORT

**SkillVine - Activity Points Management System**

Submitted by

Alvin Varghese(KTE20CS008)

Anson Anthrayose Thomas(KTE20CS013)

Sreerag M(KTE20CS056)

Vignesh R Pillai(KTE20CS068)



APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY  
THIRUVANANTHAPURAM

JUNE 2023

**RAJIV GANDHI INSTITUTE OF TECHNOLOGY**  
**GOVERNMENT ENGINEERING COLLEGE**  
**KOTTAYAM-686 501**



**DEPARTMENT OF**  
**COMPUTER SCIENCE AND ENGINEERING**

**Vision of the Department**

To be a centre of excellence for nurturing the young minds to become innovative computing professionals for the empowerment of society.

**Mission of the Department**

- To offer a solid foundation in computing and technology for crafting competent professionals.
- To promote innovative and entrepreneurial skills of students by exposing them to the forefront of developments in the field of computing.
- To inculcate strong ethical values in the young minds to work with commitment for the progress of the nation.

RAJIV GANDHI INSTITUTE OF TECHNOLOGY  
GOVERNMENT ENGINEERING COLLEGE  
KOTTAYAM-686 501



DEPARTMENT OF  
COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

*This is to certify that this report entitled **SkillVine - Activity Points Management System** is an authentic report of the project done by the team consisting of **Alvin Varghese(KTE20CS008)**, **Anson Anthrayose Thomas(KTE20CS013)**, **Sreerag M(KTE20CS056)**, and **Vignesh R Pillai(KTE20CS068)** during the academic year **2022-23**, in partial fulfilment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science and Engineering of APJ Abdul Kalam Technological University, Thiruvananthapuram.*

GUIDE

COORDINATOR

HEAD OF THE DEPARTMENT

# Acknowledgement

Every successful project is the outcome of hard work and strong support and guidance given by a number of people. We sincerely appreciate the inspiration, support and guidance of all those people who have been instrumental in making this project a success.

We express our sincere gratitude to **Dr. Jalaja M. J., Former Principal** and **Dr. Prince A., Principal** for making the resources available at the right time without which this project would not have been a success.

We take this opportunity to express a deep sense of gratitude to **Prof. Kavitha N., Associate Professor Head, Department of Computer Science and Engineering.**

We also take this opportunity to express our profound gratitude and deep regards to our guide **Prof. Aswathy B, Asst. Professor** for her exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by her from time to time shall carry us a long way in the journey of life on which we are about to embark.

We also express our gratitude to Project Coordinators **Prof. Anil Kumar S** and **Prof. Nisha K.K** for the cordial support, valuable information and guidance, which helped us in completing this task through various stages.

Last, but not least, we thank **almighty, our parents** and **friends** for their constant encouragement without which this project would not have been possible.

Alvin Varghese  
Anson Anthrayose Thomas  
Sreerag M  
Vignesh R Pillai

# Declaration

We, the undersigned hereby declare that the project report entitled Project **SkillVine - Activity Points Management System**, submitted for partial fulfilment of the requirements for the award of the degree of Bachelor of Technology of the **APJ Abdul Kalam Technological University, Kerala** is a bonafide work done by us under the supervision of **Prof. Aswathy B.** This submission represents our idea in our own words where ideas or words of others have not been included; we have adequately and accurately cited and referenced the original sources. We have adhered to the ethics of academic honesty and integrity and have not misrepresented or fabricated any data, idea or on our submission. We understand that any violation of the above can result in disciplinary actions from the institute and/ or University and can evoke penal action from the sources which have not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed as the basis for awarding any degree, diploma or similar title of any other university.

Name of Students:

Signatures:

Alvin Varghese (Reg. No.: KTE20CS008)

Anson Anthrayose Thomas (Reg. No.: KTE20CS013)

Sreerag M (Reg. No.: KTE20CS056)

Vignesh R Pillai (Reg. No.: KTE20CS068)

Batch: 2020 - 2024

August 6, 2023

# Abstract

Our web-based Activity Points Management System is a much-needed solution that simplifies and enhances the way student for activity points allocation are handled. The system integrates and automates the existing process of recording, evaluating and awarding certificates. It saves time and resources for both students and teachers by eliminating paper-based certification and manual verification procedures. These features improve the quality and reliability of the certification process.

SkillVine is a platform that enables students to track their certificates and activity points. It also helps teachers to assess students' certificates, approve or decline certificate submissions, and create reports. SkillVine streamlines and simplifies the certification process for both students and teachers. It can be used from any device with an internet connection and aims to offer a convenient, efficient, and transparent solution for managing student activity points. Our system aims to provide a convenient, efficient and transparent way of managing student activity points.

# Contents

<b>Acknowledgement</b>	<b>i</b>
<b>Declaration</b>	<b>ii</b>
<b>Abstract</b>	<b>iii</b>
<b>List of Figures</b>	<b>ix</b>
<b>List of Tables</b>	<b>x</b>
<b>List of Algorithms</b>	<b>xi</b>
<b>List of Symbols and Abbreviations</b>	<b>xii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Objectives . . . . .	1
1.2 Motivation . . . . .	1
1.3 Scope of the Project . . . . .	2
1.4 Prerequisites for the Reader . . . . .	2
1.5 Organization of the Report . . . . .	2
<b>2 System Study and Requirement Engineering</b>	<b>3</b>
2.1 Existing Systems . . . . .	3
2.1.1 Manual Method . . . . .	3
2.1.2 Third-Party Educational Platforms . . . . .	3
2.1.3 Literature survey . . . . .	4
2.2 Gap Analysis . . . . .	4
2.3 Proposed System . . . . .	4
2.3.1 Problem Statement . . . . .	5
2.3.2 System Model . . . . .	6
2.4 Requirements Engineering . . . . .	6
2.4.1 Feasibility Study . . . . .	6
2.4.1.1 Economic Feasibility . . . . .	7
2.4.1.2 Technical Feasibility . . . . .	7
2.4.1.3 Behavioural Feasibility . . . . .	7
2.4.2 Requirements Elicitation . . . . .	7
2.4.3 Requirements Analysis . . . . .	8
2.4.4 Requirements Specification . . . . .	8
2.4.4.1 Functional Requirement . . . . .	9

2.4.4.2	Non Functional Requirement . . . . .	9
2.4.5	Requirements Validation . . . . .	10
2.5	Summary . . . . .	10
<b>3</b>	<b>Design</b>	<b>11</b>
3.1	System Design . . . . .	11
3.1.1	User Registration and Authentication Module . . . . .	11
3.1.2	Notifications Module . . . . .	12
3.1.3	Student Settings Module . . . . .	13
3.1.4	Certificate List Module . . . . .	14
3.1.5	Certificate Upload Module . . . . .	14
3.1.6	Certificate Evaluation and Approval Module . . . . .	15
3.1.7	Activity Point Allocation Module . . . . .	16
3.1.8	Report Generation Module . . . . .	17
3.2	Detailed Design . . . . .	18
3.2.1	OAuth2 Algorithm for Teachers and Students . . . . .	18
3.2.2	Certificate Management . . . . .	20
3.2.3	Score Management . . . . .	20
3.2.4	Report Generation . . . . .	21
3.3	Summary . . . . .	22
<b>4</b>	<b>Implementation</b>	<b>23</b>
4.1	Tools Used . . . . .	23
4.1.1	React Js . . . . .	23
4.1.2	Jspdf . . . . .	23
4.1.3	Axios . . . . .	24
4.1.4	Node Js . . . . .	24
4.1.5	Express Js . . . . .	24
4.1.6	MongoDB . . . . .	24
4.1.7	Amazon S3 . . . . .	25
4.2	Module Implementation . . . . .	25
4.2.1	User Registration and Authentication Module . . . . .	25
4.2.2	Notifications Module . . . . .	25
4.2.3	Student Settings Module . . . . .	25
4.2.4	Certificate List Module . . . . .	26
4.2.5	Certificate Upload Module . . . . .	26
4.2.6	Certificate Evaluation and Approval Module . . . . .	26
4.2.7	Activity Point Allocation Module . . . . .	26
4.2.8	Report Generation Module . . . . .	26



4.3	Summary . . . . .	27
<b>5</b>	<b>Testing</b>	<b>28</b>
5.1	Testing Strategies Used . . . . .	28
5.1.1	Unit Testing . . . . .	28
5.1.1.1	Black Box Testing . . . . .	28
5.1.2	Integration Testing . . . . .	29
5.1.2.1	Postman API Testing . . . . .	29
5.2	Testing Results . . . . .	29
5.2.1	Results of Black Box Testing . . . . .	29
5.2.2	Results of Integration Testing . . . . .	30
5.3	Summary . . . . .	30
<b>6</b>	<b>Results</b>	<b>31</b>
6.1	Login & Register pages . . . . .	31
6.2	Student Dashboard . . . . .	32
6.3	Student Certificates View . . . . .	32
6.4	Real Time Score Section . . . . .	33
6.5	Certificate Upload . . . . .	34
6.6	Teacher Dashboard . . . . .	34
6.7	Certificate Evaluation and Approval section . . . . .	35
6.8	Report Generation . . . . .	36
6.9	Summary . . . . .	37
<b>7</b>	<b>Conclusion</b>	<b>38</b>
7.1	Recommendations for Future Works . . . . .	38
7.2	Summary . . . . .	38
<b>A</b>	<b>Software Requirement Specification</b>	<b>39</b>
A.1	Introduction . . . . .	39
A.1.1	Purpose . . . . .	39
A.1.2	Document Conventions . . . . .	39
A.1.3	Intended Audience and Reading Suggestions . . . . .	39
A.1.4	Project Scope . . . . .	40
A.2	Overall Description . . . . .	40
A.2.1	Product Perspective . . . . .	40
A.2.2	Product Features . . . . .	40
A.2.3	User Classes and Characteristics . . . . .	41
A.2.4	Operating Environment . . . . .	41
A.2.5	Design and Implementation Constraints . . . . .	41

A.2.6	User Documentation . . . . .	42
A.2.7	Assumptions and Dependencies . . . . .	42
A.3	System Features . . . . .	42
A.3.1	Authentication of Students and Teachers . . . . .	42
A.3.1.1	Description and Priority . . . . .	42
A.3.1.2	Stimulus/Response Sequences . . . . .	42
A.3.1.3	Functional Requirements . . . . .	43
A.3.2	Dashboard for students and teachers . . . . .	43
A.3.2.1	Description and Priority . . . . .	43
A.3.2.2	Stimulus/Response Sequences . . . . .	43
A.3.2.3	Functional Requirements . . . . .	44
A.3.3	Uploading, Storing, & Editing certificates and their details. . . . .	44
A.3.3.1	Description and Priority . . . . .	44
A.3.3.2	Stimulus/Response Sequences . . . . .	44
A.3.3.3	Functional Requirements . . . . .	44
A.3.4	Viewing Status of Certificates and Grading them . . . . .	45
A.3.4.1	Description and Priority . . . . .	45
A.3.4.2	Stimulus/Response Sequences . . . . .	45
A.3.4.3	Functional Requirements . . . . .	45
A.3.5	Notification system for students and Teachers . . . . .	46
A.3.5.1	Description and Priority . . . . .	46
A.3.5.2	Stimulus/Response Sequences . . . . .	46
A.3.5.3	Functional Requirements . . . . .	46
A.3.6	Report Generation . . . . .	47
A.3.6.1	Description and Priority . . . . .	47
A.3.6.2	Stimulus/Response Sequences . . . . .	47
A.3.6.3	Functional Requirements . . . . .	48
A.4	External Interface Requirements . . . . .	48
A.4.1	User Interfaces . . . . .	48
A.4.2	Hardware Interfaces . . . . .	49
A.4.3	Software Interfaces . . . . .	49
A.4.4	Communications Interfaces . . . . .	50
A.5	Other Nonfunctional Requirements . . . . .	50
A.5.1	Performance Requirements . . . . .	50
A.5.2	Safety Requirements . . . . .	51
A.5.3	Security Requirements . . . . .	51
A.5.4	Software Quality Attributes . . . . .	52
A.6	Other Requirements . . . . .	53
A.6.1	User Documentation . . . . .	53

A.6.2	Database requirements . . . . .	53
A.6.3	Internationalization Requirements . . . . .	53
A.6.4	Legal Requirements . . . . .	54
A.6.5	Reuse Objectives . . . . .	54
<b>References</b>		<b>55</b>

# List of Figures

3.1	UML diagram - User registration and authentication module . . . . .	12
3.2	UML diagram -Notifications module . . . . .	13
3.3	UML diagram -Student settings module . . . . .	13
3.4	UML diagram - Certificate List module . . . . .	14
3.5	UML diagram - Certificate Upload module . . . . .	15
3.6	UML diagram - Certificate Evaluation & Approval module . . . . .	16
3.7	UML diagram - Activity Point Allocation module . . . . .	17
3.8	UML diagram - Report Generation module . . . . .	18
6.1	Login page and Register page for student and teacher . . . . .	31
6.2	Student dashboard with notification . . . . .	32
6.3	Student certificate view . . . . .	33
6.4	Real time score section . . . . .	33
6.5	Certificate upload and edit section . . . . .	34
6.6	Teacher dashboard . . . . .	35
6.7	The Certificate Evaluation and Approval section . . . . .	36
6.8	Report generation . . . . .	37

# List of Tables

2.1	Gap Analysis . . . . .	4
-----	------------------------	---

# List of Algorithms

1	Authentication and Authorization . . . . .	19
2	Certificate Management . . . . .	20
3	Score Management . . . . .	21
4	Report Generation . . . . .	21

# List of Symbols and Abbreviations

UML	Unified Modelling Language
API	Application Programming Interface
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
JSON	JavaScript Object Notation
SMTP	Simple Mail Transfer Protocol
SOC 2	System and Organization Controls 2
SRS	Software Requirements Specification
SSL/TLS	Secure Sockets Layer/Transport Layer Security
TBD	To Be Determined
UI	User Interface
UX	User Experience

# Chapter 1

## Introduction

Apart from technical knowledge and skills, universities have introduced activity points to nurture qualities like soft skills, leadership qualities, and team spirit in order to be successful as professionals. As a result, to comply with the guidelines, students and faculties have to go through the conventional activity point management system.

This project seeks to address this requirement by providing a platform that streamlines the process of managing students' certificates and activity points. Without failing the university guidelines, by leveraging web technologies, the project offers a user-friendly interface that ensures a rewarding experience for both students and faculties.

### 1.1 Objectives

This project aims to achieve the following objectives.

1. To simplify the process of certificate submission, viewing, and editing.
2. To provide real-time access to activity points and teacher remarks.
3. To provide a hassle free evaluation and report generation for the faculties.
4. To ensure a seamless user experience for students and teachers.

### 1.2 Motivation

The facts that motivated the choice of the project are the following.

1. Even though activity point management is now essential and mandatory, many academic institutions face challenges in manually handling, validating students' certificates, and calculating activity points. These manual processes often result in time-consuming tasks, potential errors, and a lack of transparency, impacting the overall academic experience.
2. The status of activity points will be available to the students only at the end of the procedure on the university website. Building a system that provides real-time status of the points will encourage students to actively participate in extracurricular activities and take charge of their academic journey.
3. Institutions and faculties stress more on basic academic procedures and give less time for the activity management procedure, which becomes a complex task when done manually.



## 1.3 Scope of the Project

- **Need:** The traditional manual methods for handling certificates and calculating activity points result in time-consuming and error-prone processes. This prompts the need to develop an automated and centralized platform.
- **Deliverables:** The project aims to deliver a user-friendly web application accessible to students and teachers, providing an effortless experience for managing student activity points. It will have a dedicated area for students to view their activity points, certificates, and remarks from teachers. Additionally, teachers will be able to efficiently mark and approve student certificates, access data, and generate reports.
- **Exclusions:** The system lacks the ability to automatically determine the authenticity of certificates. Instead, it relies on teachers to manually verify the certificates.

## 1.4 Prerequisites for the Reader

To understand the contents of this report, the reader is expected to meet the following prerequisites:

- While not mandatory, a basic understanding of software development concepts and technologies (such as front-end and back-end development) will facilitate comprehension of the technical aspects of the project.
- Familiarity with the typical activities, events, and certifications involved in college or university education.
- Some basic understanding of web applications, their functionalities, and user interfaces.
- An awareness of activity assigning guidelines prescribed by the university.

## 1.5 Organization of the Report

Chapter 2 deals with the Feasibility Study(section 2.4.1), explores the Existing Systems(section 2.1), continues to the Gap Analysis(section 2.2) and then proceeds to the Proposed System(section 2.3), and the Requirements Engineering(section 2.4). Chapter 3 deals with the design of the project. The System Design(section 3.1) which provides a basic overview of different modules in the system in 8 different sections. In section 3.2 of this chapter, we have the detailed design, where the User Authentication design in section 3.2.1, algorithms related to certificate management & score management in the first two Modules, in section 3.2.4 algorithm for report generation is given. Chapter 4 explores the Tools Used that are explained in section 4.1 and the Implementation of Modules that are shown in section 4.2(8 modules in total starting from 4.2.1). Chapter 5 describes the Testing Strategies Used in section 5.1 which includes Unit Testing in section 5.1.1 and Integration Testing in section 5.1.2 followed by the Testing Results in section 5.2 for each strategy given. Chapter 6 contains the results of the project. The report is summarised in Chapter 7.

# Chapter 2

## System Study and Requirement Engineering

Every project starts with some kind of reality check. More often, it is required to answer the following questions.

1. What is the purpose of the platform and what problem will it solve?
2. What is the impact of the project?
3. How much will it cost?

The answers to these questions can be understood easily by performing System Study through Literature Survey and Requirement Engineering.

### 2.1 Existing Systems

Existing systems are systems that are currently in place that are used to perform the task or satisfy most of the objectives that the project had set out to achieve. These systems are examined to see the different approaches and techniques that are in use now and their shortcomings. Analysis of existing systems gives an insight into how a new system can be implemented and the research, findings and results provide quick solutions to parts that are common. The following are the existing systems used to manage activity points.

#### 2.1.1 Manual Method

Teachers or administrators would maintain physical registers to record and validate certificates manually. Certificates kept as hard copies. Calculations are done manually, and teacher should have to attend different certificates of a single student, along the whole batch. While these systems may have served the purpose in the past, however they are overall time consuming and laborious series of tasks.

##### **Advantages:**

- Simple to implement.
- No dependency on technology or software.
- Easy to manage for smaller institutions with limited data

#### 2.1.2 Third-Party Educational Platforms

These are more generalised system which tracks academic related areas. They track academic grades, exam performances etc along with certificate management and point suggestions. However these systems can only suggest the estimated

points, which may not be correct as certificates are verified by Institution's teachers. They might come with subscription fees or limited customization options.

#### Advantages:

- May offer a range of additional educational tools and features.
- Regular updates and support from the platform provider.

### 2.1.3 Literature survey

A literature survey is a comprehensive study of the existing systems and their details related to the topic of the project. The literature review surveys scholarly articles, books, and other sources relevant to a particular area of the project. The review should enumerate, describe, summarize, objectively evaluate and clarify the existing system and it helps to form a set of requirements for the new system which we propose by analyzing the gaps in the existing system. Following is an existing system for undergoing Activity point management.

## 2.2 Gap Analysis

Some potential drawbacks were identified in each of the above-mentioned existing systems. They are discussed in the following.

Existing System	Analysis
Manual Method	Manual systems are time-consuming, labor-intensive, and prone to errors due to manual data entry and processing. Also the absence of real time data is also an issue. The physical involvement of the teacher and student is more and may become stressful in some situations. Also these may take up time which can be utilized for other academic procedures.
Third-Party Educational Platforms	Designed for a wider range of functions beyond activity points management. Also the points updated may not be accurate as the validation may or may not be conducted by institutional teachers. Also points may not be updated instantly. Limited flexibility and segregation of certificates by year, category, event. They might come with subscription fees or limited customization options.

Table 2.1: Gap Analysis

## 2.3 Proposed System

The system proposed in the project is a web application which can be accessed by both teachers and students. The system is a modern and comprehensive student activity point management platform that aims to address the limitations of existing systems and provides a secure solution for managing student certificates and activity points.

It automates the process of certificate management, eliminating the need for manual sorting and validation by teachers. Students can easily upload their certificates, edit certificate details, and view certificates accordingly. Students can instantly view their current activity points, remarks, and the points required to meet their course guidelines. Teachers can efficiently manage and update student certificates and points. The system offers an intuitive and user-friendly interface, making it easy for students and teachers to navigate and perform tasks efficiently. The dashboard provides a clear overview of activity points and certificates for students, while the teacher's dashboard allows batch-wise and individual certificate marking. It retrieves certificate marks as per University rules, also giving an option to reject certificates with a remark and generating personalized reports for every student and batch.

**Advantages:**

- **Efficient Certificate Management.**
- *Streamlined Teacher Workflow:* The system streamlines the teacher's workflow by providing batch-wise and individual certificate marking options.
- *User-Friendly Interface:* The platform offers an intuitive and user-friendly interface for both students and teachers.
- *Real-Time Point Tracking:* The system provides real-time access to students' activity points, allowing them to monitor their progress.
- *Scalability:* Can deal with certificates and data of students of entire department or college.
- *Time and Resource Savings:* It reduces the manual efforts involved in managing student activity points, saving time and resources for both students and teachers. (Eliminates physical order, sort, find, validate and grade)

**Disadvantages:**

- *Learning Curve:* Users, especially teachers and administrators, may need time to adapt to the new system.
- *Technical issue :* May encounter technical issues, such as server downtime, software bugs, or connectivity problems.

### 2.3.1 Problem Statement

For both students and teachers, the existing process of managing and verifying certificates and activity points is laborious and time-consuming. Our project intends to make the certification process a lot more efficient and transparent by developing a reliable and user-friendly system that automates it and also can assist in validation and the assigning points procedures.

## 2.3.2 System Model

- User Interface (UI): The UI serves as the front-end of the system, allowing users (students and teachers) to interact with the platform. It includes components such as login pages, dashboards, certificate forms, and report generation interfaces.
- Front-end: The front-end is developed using React and other web technologies. It communicates with the back-end through API calls to fetch data and update the system.
- Back-end: The back-end, powered by Node.js and Express, handles all server-side operations. It manages user authentication, processes API requests, communicates with the database, and applies business logic to mark certificates, generate reports, etc.
- Database: MongoDB is used as the database to store all relevant data, including student information, certificate details, teacher records, and activity points. The data in the database is structured based on the Mongoose schema.

## 2.4 Requirements Engineering

The process of gathering software requirements from clients and then analysing and documenting them is known as requirements engineering. The goal of requirement engineering is to develop and maintain a sophisticated and descriptive ‘System Requirements Specification’ document. It essentially involves the following five activities.

1. Feasibility Study
2. Requirements Elicitation
3. Requirements Analysis
4. Requirements Specification
5. Requirements Validation

### 2.4.1 Feasibility Study

Feasibility check is a procedure that identifies, describes and evaluates the proposed systems and selects the best system for the task under consideration. An estimate is made of whether the identified user needs may be satisfied using current software and hardware technologies. The study will decide if the proposed system will be cost-effective from a business point of view and if it can develop given existing budgetary constraints. The key considerations involved in the feasibility analysis are

- Economic Feasibility
- Technical Feasibility
- Behavioural Feasibility

### 2.4.1.1 Economic Feasibility

Economic analysis, or Cost/Benefit Analysis, is used to evaluate a system's effectiveness. The comparison results are found and changed if needed, improving accuracy at each phase of the system life cycle. If benefits outweigh costs, the system is designed and implemented. Otherwise, further justification or alteration is made.

The cost of the project is estimated by considering various factors, including infrastructure costs (server, hosting), and ongoing operational costs (maintenance, support). The platforms that the application is to be run in are devices that are typically possessed by students and teachers, like a personal computer (desktop or laptop or mobile).

### 2.4.1.2 Technical Feasibility

Technical Feasibility centres on the existing system and to what extent it can support the proposed additions. This involves financial considerations to be accommodated within technical enhancements. If the budget is a serious constraint, then the project of improving the current system will no longer be feasible. As the system utilizes hardware that is already in possession of the end users, no extra hardware is to be set up. The software and the assets used by it are created using tools that are well-tested or have gained a reputation for their reliability and hence the project is technically feasible.

As the system utilizes hardware that is already in possession of the end users, no extra hardware is to be set up. The software and the assets used by it are created using tools that are well-tested or have gained a reputation for their reliability and hence the project is technically feasible.

### 2.4.1.3 Behavioural Feasibility

This is also known as Operational Feasibility [5,6]. People are inherently resistant to change and computations are known to facilitate change. Therefore it is required to understand the special effort required to educate and train the users of the system when the proposed system is introduced.

Needs to identify the expectations, needs, and preferences of end-users. It evaluates the need for training and support for users during and after the system implementation. Since the software is run on devices already owned by them, an added layer of familiarity is provided.

## 2.4.2 Requirements Elicitation

The primary objective of the requirements elicitation task is to collect the requirements from the stakeholders. A stakeholder is a source of the requirements and is usually a person or a group of persons directly or indirectly concerned with the software.

1. *Interview:-* Formal or informal interviews with system stakeholders are part of most requirements engineering processes. In these interviews, the requirements engineering team puts questions to stakeholders about the system that they currently use and the system to be developed. Requirements are derived from the answers to these questions.

Informal interviews were held with teachers, with two main questions in mind. They are:

- (a) Are there problems with the current manual system?

(b) What features do you expect in an automated activity points management system?

Upon asking the teachers, it was noticed that the process is laborious, and there is less interaction with students as it is not appropriate to physically demand students every time they need to convey their remarks. Additionally, the teachers have to spend extra time going through each student's validated documents to prepare a report. Discussions with students also pointed out the areas where they are facing issues.

2. *Studying existing systems*:- The existing systems were studied as part of the literature survey and for each of these systems, we analysed the pros and cons. These details are given in section 2.1 with a gap analysis provided in section 2.2
3. *Ethnography and observations*:- Ethnography is an observational technique that can be used to understand operational processes and help derive requirements for software to support these processes. An analyst immerses themselves in the working environment where the system will be used. It helps discover implicit system requirements that reflect the actual ways that people work, rather than the formal processes defined by the organization.

Other than the interviews and discussions it was observed from the existing system that teachers found it difficult to operate as they are physically needed in the campus for the validation process. Also these may take up a large amount of valuable time which can be utilized for other academic purposes.

### 2.4.3 Requirements Analysis

Requirements Analysis is the process of defining the expectations of the users for an application that is to be built or modified. It involves all the tasks that are conducted to identify the needs of different stakeholders. Therefore requirements analysis means to analyze, document, validate and manage software or system requirements. We analyzed the data collected from the interview and the study of existing systems and valuable information was extracted from these data. This information led to the development of the app with the following requirements.

1. A remark system with notification. After a remark is done, the students get notified, providing the physical presence of student is not required to convey it.
2. A report generation system for student wise or even batch wise which can be accessed quickly. The teacher need not to create separate excel or pdf file as the system does so with single click.
3. The certificate and data are available at any time enabling teachers to work on it beyond college time, even at their home.

### 2.4.4 Requirements Specification

When writing a requirements specification, the primary goal is to deliver the best possible product, not a perfect specification. Requirements are categorized into functional and non-functional. Functional requirements describe the system's functionality, while non-functional requirements outline aspects such as speed and responsiveness. The essence

of a requirements specification is the requirements themselves. Several good definitions exist for writing a requirements specification, but they all share this common element.

#### 2.4.4.1 Functional Requirement

Functional requirements are product features that developers must implement to enable the users to achieve their goals. They define the basic system behavior under specific conditions. The functional requirements of the system are given below:

- A web app to store,manage and validate certificate and to manage its points attained.
- Email Authentication.
- Dashboard for students and teachers.
- Real-Time points updation.
- Certificate filtering, sorting, selection.
- Certificate marking, approving, grading for teachers.Notification system.
- Must have a Reporting system.
- Must be mobile responsive(users with laptop or desktop can also access the system)
- The GUI must be user friendly and modern.

#### 2.4.4.2 Non Functional Requirement

They impose constraints on the implementation of the functional requirements. Non-functional requirements can be classified into Interface Requirements, Communication Requirements, Hardware Requirements, Security Requirements, Software Quality Requirements and so on. **Interface Requirements**

- Interfaces must be easy to use and understand.
- Light Design and fast performance for seamless UI/UX.
- Menus must be easily accessible and easy to navigate through different functions.

##### **Software Quality Requirements**

- Any Operating system that supports a modern day browser is required.
- The system should contain minimal bugs and glitches.

##### **Hardware Requirements**

- The system should have minimal hardware requirements so that the system can be more accessible.
- A computer or laptop(RAM: 4 GB RAM or higher) or mobile
- Processor: Intel Core i3 or higher.
- A fast and reliable Internet connection.
- Printer (optional-to printout reports).



### **2.4.5 Requirements Validation**

Requirement validation ensures that the requirements are complete, consistent, and meet prescribed quality standards according to user requirements. An organized manual analysis of the project's needs was carried out, validating the system's ability to efficiently carry out point management procedures. Non-functional requirements like interface, software quality, software, and hardware requirements were also checked.

## **2.5 Summary**

In this chapter, we discussed the systems already in place to educate students on complexity classes, their drawbacks, as well as the system proposed as part of the project. We also discussed requirements engineering, which consists of the feasibility study, requirement elicitation, analysis, specification, and validation processes, during which we examined different methods for collecting requirements and the standard document specifying the requirements. The chapter ends with the set of functional and non-functional requirements of the project and its validation.

# Chapter 3

## Design

The design phase aims to plan a solution to meet the specified requirements, transitioning from the problem domain to the solution domain. It focuses on determining "how to satisfy the needs" identified earlier. The design heavily influences the software's quality, significantly impacting testing and maintenance. Upon completion, the design phase produces a detailed plan used for implementation, testing, and maintenance. The design activity is divided into two phases:

- System Design
- Detailed Design

### 3.1 System Design

System Design aims to identify the modules that should be present in the system, the specification of these modules, and how they interact with each other to produce the desired results. At the end of the system design, all the major data structures, file formats, output formats, and major modules in the system, along with their specifications, are decided.

We divide the system design into separate modules, each dealing with a different aspect of the game. They are:

1. User Registration and Authentication Module
2. Notifications Module
3. Student Settings Module
4. Certificate List Module
5. Certificate Upload Module
6. Certificate Evaluation and Approval Module
7. Activity Point Allocation Module
8. Report Generation Module

#### 3.1.1 User Registration and Authentication Module

The User Registration and Authentication Module is an essential component that enables users to log in as either students or teachers. Its primary goal is to streamline the authentication process by enabling users to create or access their personal accounts within the system.

1. **User Registration:** Students and teachers can register on the activity point management system through Google Auth2 protocol. Upon successful registration, the user's information is stored in the system's database.
2. **User Profile Management:** This module enables students to update their personal details, such as their KTU ID and Admission Number, if they are newly registered users.
3. **User Roles and Access Permissions:** This module manages and verifies the email address and, if it is not an institution email, the user is redirected to an error page. Depending on the selected login option, the email addresses are verified to ensure that they belong to a student or teacher, and then the user is redirected accordingly.

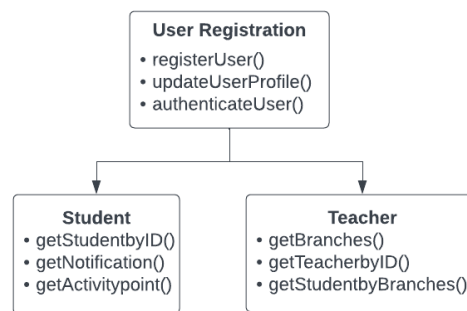


Figure 3.1: UML diagram - User registration and authentication module

### 3.1.2 Notifications Module

In the activity point management system, the Notifications Module is essential for keeping students up to date on the status of their uploaded certificates, whether they have been accepted or rejected.

1. **Notification Triggers:** The Notifications Module is integrated with the activity point management system and responds to specific events within the system, such as certificate approval and the allocation of activity points to users.
2. **Certificate Approval Notifications:** When a teacher approves a certificate, the Notifications Module automatically sends a notification to the student, informing them of the certificate's approval and the activity points they have earned.
3. **Notification History and Tracking:** The system may also keep a notification history for each student, enabling them to see previous notifications until it is cleared by the student.

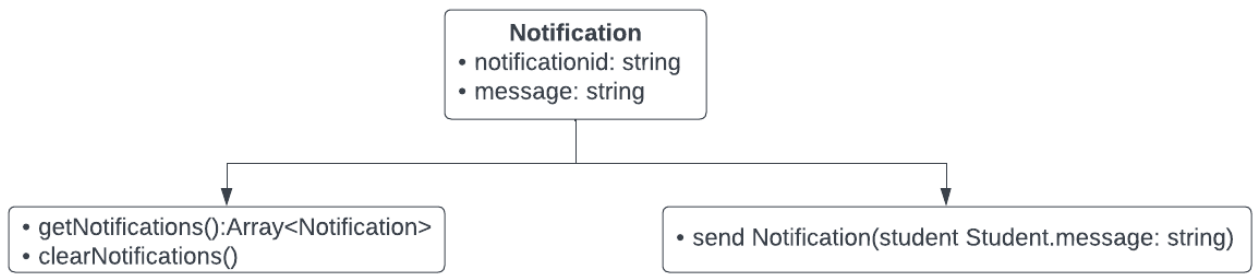


Figure 3.2: UML diagram -Notifications module

### 3.1.3 Student Settings Module

The Student Settings Module is a component that allows students to access their account settings. An explanation of how this module functions is provided below:

1. **Personal Information:** The Student Settings Module enables students to view their personal information, such as their KTU ID, Admission number, college, points etc. Students can access a user-friendly interface to modify their information whenever necessary. This ensures that the system always has up-to-date and accurate data for each student.
2. **Deleting Student Details:** The Student Settings module manages the account deletion feature for students, which allows them to delete all records related to their account from the entire database if requested.

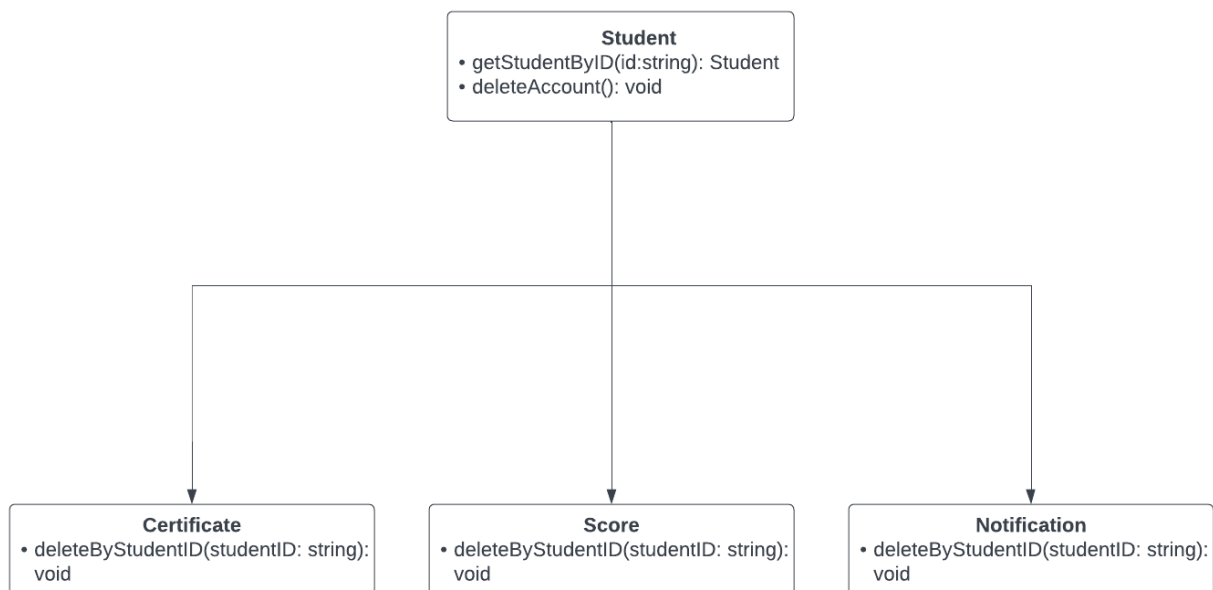


Figure 3.3: UML diagram -Student settings module

### 3.1.4 Certificate List Module

The Certificate List Module is a crucial component of the Activity Point Management System that helps students keep track of the various certificates they have uploaded. Its main function is to monitor the status of a student's certificates, whether they have been approved or rejected.

1. **Certificate Statuses:** This module provides a complete list of all certificates, along with their approval status and category details. Users can view more information about each certificate by opening it.
2. **Criteria and Guidelines:** Detailed criteria are provided for each certificate to help students understand the expectations and requirements for earning points. This ensures transparency and clarity in the evaluation process.

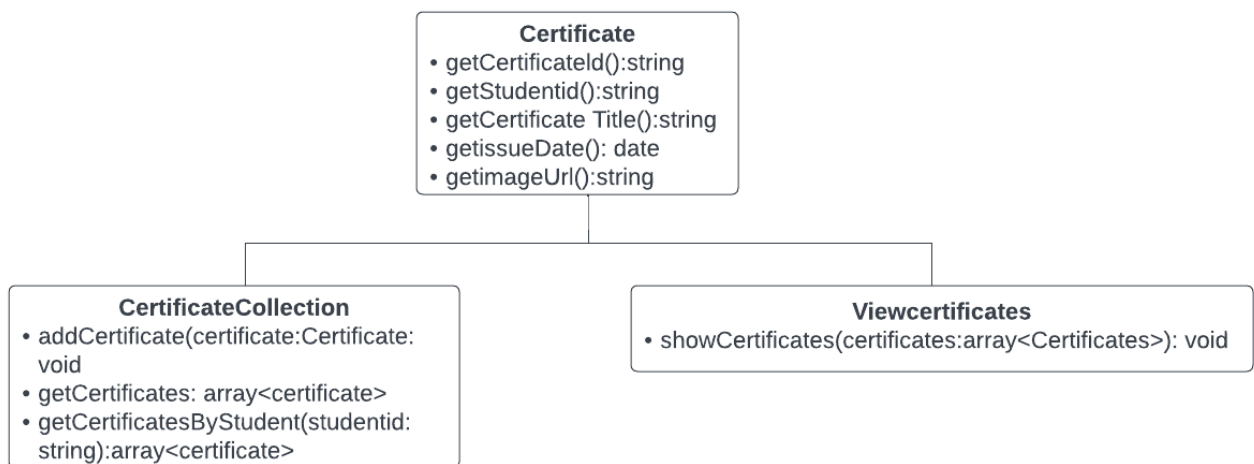


Figure 3.4: UML diagram - Certificate List module

### 3.1.5 Certificate Upload Module

The Certificate Upload Module is a feature that enables students to upload certificates for validation by teachers, allowing them to earn activity points. This module allows students to upload PDFs or images of their certificates and organize them by year.

- **Uploading Certificate:** Students can use the Certificate Upload Module to attach relevant certificates, such as scanned certificates, photos, reports, or other documentation that verifies their participation or achievements.
- **Verification:** Once the certificate is uploaded, the teacher will review the certificate provided by the student to ensure its authenticity.
- **Point Allocation:** After verification, if the certificate is deemed valid, the teacher will allocate the appropriate number of activity points to the student's profile.

- **Transparency and Accountability:** The system keeps track of all uploaded evidence and the point allocation history for complete transparency and accountability.

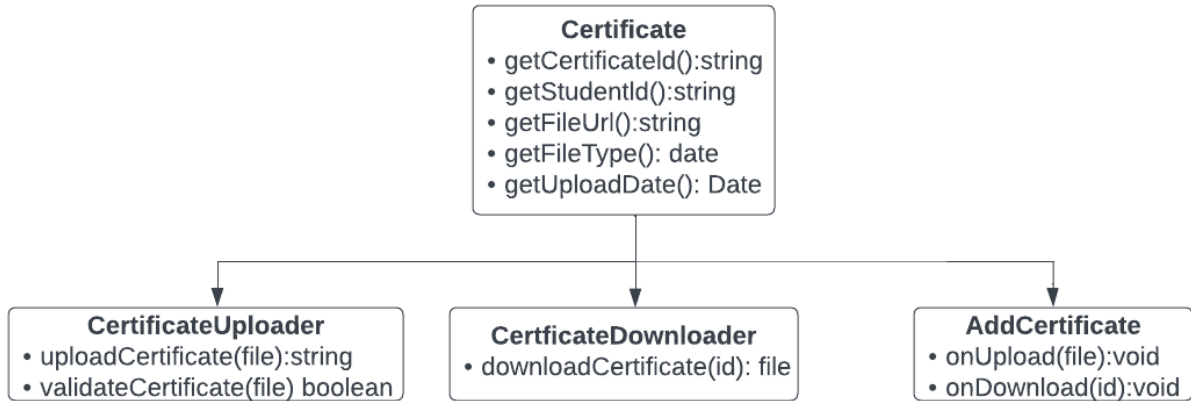


Figure 3.5: UML diagram - Certificate Upload module

### 3.1.6 Certificate Evaluation and Approval Module

The Certificate Evaluation and Approval Module in the Activity Point Management System streamlines the process of reviewing and approving certificates submitted by students. This system serves as a platform for storing and awarding activity points to users based on the certificates they earn.

- **Certificate Submission:** Students can submit certificates they have earned for various activities, such as participation in events, competitions, workshops, or any other achievements that are eligible for earning activity points.
- **Document Upload:** Students upload digital copies of their certificates along with relevant information like the name of the activity, date, organizer, and any additional notes if required.
- **Review and Approval:** Teachers or administrators with the necessary permissions have access to the submitted certificates. They can view the evidence provided by students to verify its authenticity and appropriateness. The module provides a user-friendly interface to facilitate easy evaluation.
- **Approval Decision:** Based on the evaluation, the teachers or administrators can approve or decline the certificates. Approved certificates will be considered valid for earning activity points.
- **Points Assignment:** The module automatically calculates and assigns activity points to each approved certificate based on predefined criteria. Different types of activities may have different point values associated with them, depending on their significance and impact.

- **User Profile Update:** The awarded activity points are added to the student's profile in the system. This update is reflected in their overall activity points tally.
- **Transparency and Communication:** The system ensures transparency by keeping a log of all certificate evaluation and approval activities. Students can also receive notifications about the status of their submissions, whether they have been approved or declined.
- **Feedback and Re-submission:** If a certificate is declined, the system allows administrators or teachers to provide feedback to students, specifying the reasons for the rejection. Students can then make necessary improvements or corrections and re-submit the certificate for re-evaluation.

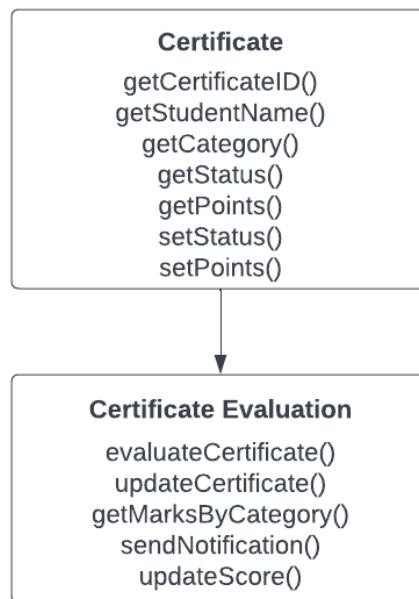


Figure 3.6: UML diagram - Certificate Evaluation & Approval module

### 3.1.7 Activity Point Allocation Module

The Activity Point Allocation Module is a vital component of the Activity Point Management System. Its primary purpose is to manage the activity points earned by students based on their approved certificates. The module performs several key functions:

- **Activity Point Calculation:** The module calculates the activity points earned by students based on the certificates they submit. Each certificate corresponds to a particular activity, and the module assigns a specific point value to each activity type.
- **Approval Process:** Once a student submits a certificate for an activity, it undergoes an approval process. The teacher or an authorized person reviews the certificate to ensure its validity and authenticity. If the certificate is genuine and meets the criteria for earning activity points, it is approved by the teacher.

- **Point Allocation:** After approval, the module allocates the appropriate number of activity points to the student's profile. This update is reflected in the student's activity point score.
- **Notification:** Upon successful approval and point allocation, the student is notified about the earned activity points with an internal notification system.

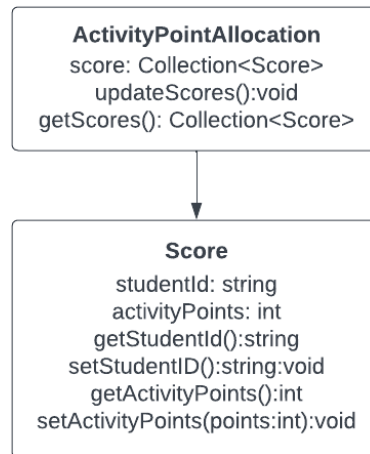


Figure 3.7: UML diagram - Activity Point Allocation module

The Activity Point Allocation Module streamlines the process of awarding activity points, minimizes errors, and ensures fairness and transparency. It automates the calculation and approval process, reducing manual effort and enhancing efficiency. The module allows for flexibility, enabling corrections when needed, ensuring an accurate representation of a student's activities and achievements.

### 3.1.8 Report Generation Module

The Report Generation Module is a crucial component of the Activity Point Management System, designed to generate comprehensive reports and analytics, offering valuable insights into the students' progress, achievements, and activity trends within the institution or educational setting.

#### Key Features of the Report Generation Module:

1. **Individual Student Reports:** The module allows teachers to access individual student reports. These reports provide a detailed breakdown of the student's earned activity points. It helps students understand their strengths and areas for improvement.
2. **Overall Achievements:** The module generates comprehensive reports that summarize the overall achievements of each student



3. **Certificates and Recognition:** The activity points are allotted based on the KTU grading schema, ensuring that the recognition is based on a standardized and widely accepted system.
4. **Export and Sharing:** Reports can be exported in PDF format for further analysis or sharing with stakeholders, such as parents, teachers, or school management.
5. **Data Security:** As with any system handling sensitive student information, data security and privacy are paramount. The module ensures that access to reports is restricted to authorized personnel only, such as teachers.

The Report Generation Module plays a crucial role in providing valuable insights and analytics to teachers. By offering individual reports, students can track their progress and make informed decisions about their activities, encouraging active participation.

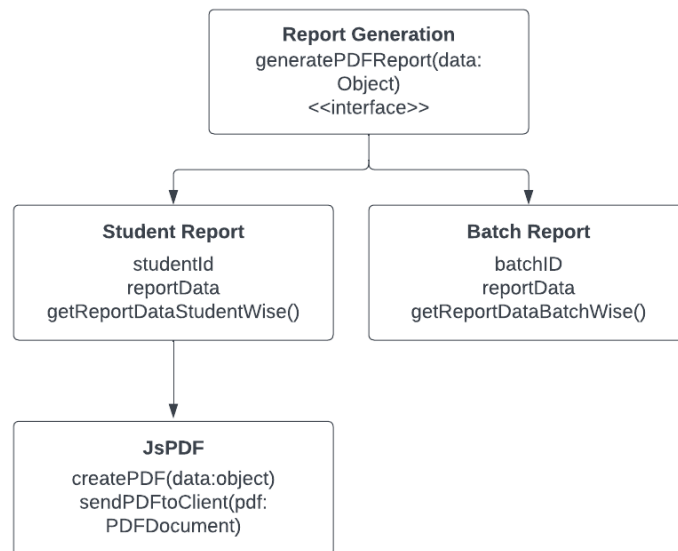


Figure 3.8: UML diagram - Report Generation module

## 3.2 Detailed Design

### 3.2.1 OAuth2 Algorithm for Teachers and Students

Our system authenticates teachers and students with OAuth2 protocol using two classes: `AuthTeacherController` and `AuthStudentController`. A method redirects users to the authentication URL, another fetches user info and verifies if they are a teacher or student. If valid, a JWT is generated and added to a cookie, redirecting the user. Another method generates an access token for authorized users, while a final one logs out users by removing their refresh token from the cookie.

**Algorithm 1:** Authentication and Authorization

**Input** : Student information from the authentication service.  
**Output**: Access token for the student or logout confirmation.

```

1 redirectAuth::
2   if the user is a teacher then
3     |   Redirect to the authentication URL for teachers.;
4   end
5   else if the user is a student then
6     |   Redirect to the authentication URL for students.;
7   end
8 callbackAuth::
9   Handle the callback from the authentication service and receive the user's
    information.;
10  if the user is a teacher then
11    |   Reject the authentication with an error message.;
12  end
13  else if the user is a student then
14    |   Check if the student already exists in the system.;
15    |   if the student is new then
16    |     |   Create a new student record with the provided information.;
17    |   end
18    |   Generate a refresh token for the student.;
19    |   Attach the refresh token to the response as a cookie.;
20  end
21 getAccessToken::
22   Input: Refresh token.;
23   if the refresh token is valid then
24     |   Generate a new access token for the student using the provided refresh token.;
25     |   Output: Access token for the student.;
26   end
27   else
28     |   Output an error message indicating an invalid refresh token.;
29   end
30 logoutUser::
31   Input: Refresh token.;
32   if the refresh token exists and is valid then
33     |   Clear the refresh token to log out the student.;
34     |   Output: Logout confirmation.;
35   end
36   else
37     |   Output an error message indicating an invalid or missing refresh token.;
38   end

```

### 3.2.2 Certificate Management

Our system has four functions for handling certificates: `getCertificateByStudentId`, `getCertificateById`, `markCertificate`, and `rejectCertificate`. The first two retrieve a student's certificates using their `studentId` or a specific certificate using its `certificateId`. The `markCertificate` function allows a user to mark a certificate with data such as points, category, and status, which is validated and updated. If any issues arise, appropriate errors are thrown. The `rejectCertificate` function allows a user to reject a certificate, setting its status to 'rejected' and points to 0. A notification is created to inform the student of the rejection.

Algorithm 2: Certificate Management
<pre> 1 <b>Algorithm</b> 2 <b>getCertificateByStudentId;</b> 3   <b>Input:</b> Student ID; 4   <b>Output:</b> Student's certificates for each year; 5   Retrieve the student's certificates for each year using the provided Student ID.; 6 <b>getCertificateById;</b> 7   <b>Input:</b> Certificate ID; 8   <b>Output:</b> Specific certificate; 9   Retrieve the specific certificate using the provided Certificate ID.; 10 <b>markCertificate;</b> 11   <b>Input:</b> Certificate ID, points, category, status; 12   <b>Output:</b> None; 13   Validate the provided data.; 14   Update the certificate using the provided services.; 15   <b>if</b> any issues arise during the update process <b>then</b> 16       Throw appropriate errors.; 17   <b>end</b> 18 <b>rejectCertificate;</b> 19   <b>Input:</b> Certificate ID; 20   <b>Output:</b> None; 21   Set the certificate's status to 'rejected' and its points to 0.; 22   Create a notification to inform the student about the rejection.;</pre>

### 3.2.3 Score Management

We have a `ScoreController` class with a `scoreServices` property initialized to a new instance of `ScoreServices`. The `getScoreAuthenticated` function extracts the user's id from the request, throws an error if not found, retrieves the user's score using the `scoreServices` property and user's id, and returns a response with the user's score. This algorithm describes creating a controller for handling scores using a `scoreServices` property and `getScoreAuthenticated` function to retrieve and return the authenticated user's score.

**Algorithm 3: Score Management**

```

1 Algorithm
2 ScoreController;;
3   A class with a scoreServices property initialized to a new instance of
   ScoreServices.;
4 getScoreAuthenticated;;
5   Input: None;
6   Output: User's score;
7   Extract the user's ID from the request.;
8   if the user's ID is not found then
9     |   Throw an error indicating that the user was not found.;
10  end
11  Retrieve the user's score using the scoreServices property and the user's ID.;
12  Return a response with the user's score.;

```

### 3.2.4 Report Generation

The ReportController class generates reports for students and has two main methods: getReportBatch and getReportStudent. The getReportBatch method takes a batch ID, retrieves a list of students in the batch, gets their scores using the reportService, and sends the report as a JSON response. The getReportStudent method takes a student ID, retrieves the student's information, gets their report using the reportService, and sends the report as a JSON response. These methods use several utility classes and services to handle errors, define interfaces, send responses, and interact with other services.

**Algorithm 4: Report Generation**

```

1 Algorithm
2 ReportController;;
3   A class responsible for generating reports for students.;
4 getReportBatch;;
5   Input: Batch ID;
6   Output: Report as a JSON response;
7   Takes in a batch ID as a parameter.;
8   Retrieves a list of students in the batch.;
9   Gets their scores using the reportService.;
10  Sends the report as a JSON response.;
11 getReportStudent;;
12  Input: Student ID;
13  Output: Report as a JSON response;
14  Takes in a student ID as a parameter.;
15  Retrieves the student's information.;
16  Gets their report using the reportService.;
17  Sends the report as a JSON response.;

```

## 3.3 Summary

This chapter discusses the design of the Activity Point Management System, composed of several modules, including User Registration and Authentication, Notifications, Student Settings, Certificate List, Certificate Upload, Certificate Evaluation and Approval, Activity Point Allocation, and Report Generation.

In the detailed design section, we provide an in-depth description of each module and its functions. We also discuss how each module interacts with other modules to achieve the overall goal of managing activity points effectively and efficiently.

# Chapter 4

## Implementation

The development or implementation stage is the stage where the development of the code and modules is performed. The development and integration of the units and creation of the project build are done according to the design documents' specifications. The goal of this phase is to translate the design of the proposed system into a working model. Since this phase affects the testing phase, the coding for the project is performed in such a way as to maintain simplicity and clarity.

### 4.1 Tools Used

A project of this scale requires a fair number of tools in order to implement all the required features. Below are the tools that were used for the development of this project:

#### 4.1.1 React Js

React js is a JavaScript library for building user interfaces. It allows to create reusable UI components and render them dynamically.

- It is efficient, declarative and flexible. We chose this tool because it simplifies the development of complex web-applications and provides a fast and interactive user experience.
- In React js, a component is a function that returns what should be rendered on the screen. A component can have props, which are inputs from the parent component, and state, which is the internal data that can change over time. It made the website's UI interactive and reactive.

#### 4.1.2 Jspdf

Jspdf is a JavaScript library for generating PDF documents. It enables to create PDF files from HTML elements, images, text, etc.

- It is lightweight, easy to use and supports various document styles.
- We chose this tool because it allows exporting the data from the application into a printable format and its easy to write syntax.

### 4.1.3 Axios

Axios is a JavaScript library for making HTTP requests. It supports both browser and node.js environments.

- We chose this tool because it simplifies the communication between the client and the server and handles errors gracefully.
- In axios, an HTTP request is made and handled using an instance of the Axios class that provides various methods and options like request and response.

### 4.1.4 Node Js

Node is a JavaScript runtime environment that executes JavaScript code outside the browser.

- We have chosen this tool because it enables to build scalable and performant server-side applications using JavaScript.
- We use modules which is a file or a folder that contains JavaScript code that can be imported and used by other files. Node has built-in modules such as fs, http, path, etc.

### 4.1.5 Express Js

Express is a web framework for node.js that provides various features such as routing, middleware, template engines, etc.

- It is a minimalist, fast and unopinionated framework. We chose this tool because it facilitates the development of RESTful APIs and web applications using node.js.
- Express routing defines how the app responds to client requests with specific URIs and HTTP methods. Middleware are functions that modify the request and response objects before reaching the final handler.

### 4.1.6 MongoDB

MongoDB is a NoSQL database that stores data in JSON-like documents. It is schemaless, scalable and flexible.

- We chose this tool because it suits the data model of our application and provides high performance and availability.
- In MongoDB, a database contains collections of documents. Collections group similar documents and don't enforce a schema. Documents are records of data with fields, and can have nested sub-documents or arrays. Each document can have a unique id field.

### 4.1.7 Amazon S3

Amazon s3 is a cloud storage service that offers storage and retrieval facilities for files or data.

- We chose this tool because it provides a secure and speedy solution for storing the pdf and image certificate files.
- In amazon S3, a storage system is composed of one or more buckets that contain objects.

## 4.2 Module Implementation

In this section of our project report, the implementation of each module is explained in detail.

### 4.2.1 User Registration and Authentication Module

This module manages the authentication process for users, including students and teachers, and their roles and access permissions. Express is used to create registration and login routes, MongoDB to store user details, Google OAuth2 protocol for authentication, and JSON Web Token to generate and verify tokens. Axios is used to make HTTP requests from the client to the server for registration and login. The module also checks if it's the user's first login and redirects them appropriately.

### 4.2.2 Notifications Module

This module is responsible for managing notifications and alerts within the system. It sends automated notifications to students about events such as certificate approval and points allocation. To implement this module, we have used MongoDB to store the notifications in a collection called 'notifications'. We have used Express to create the routes for adding, fetching, and clearing notifications. We have used Axios to make HTTP requests from the client to the server for notifications. Additionally, we have used React.js to create a component called 'Notification' that displays the notifications on the user interface.

### 4.2.3 Student Settings Module

This module allows students to view their account details. Students can view personal information and also delete their account and all its data. To implement this module, we have used React.js to create a component called 'Settings' that displays user data and an option to delete the user. We have used Axios to make HTTP requests from the client to the server for user details and the user delete action. Additionally, we have used Express to create a route to delete all details related to a student.



## 4.2.4 Certificate List Module

This module provides a list of available certificates. To implement this module, we have used MongoDB to store the certificates of all students in a collection called 'certificates'. We have used Express to create the routes for fetching certificates. Additionally, we have used Axios to make HTTP requests from the client to the server for certificates and to upload certificates. Furthermore, we have used React.js to create a component called 'View Certificates' that displays the certificates on the user interface for both students and teachers.

## 4.2.5 Certificate Upload Module

This module allows students to upload certificates in PDF or image formats to validate them for earning activity points. Amazon S3 is used to store the files in a 'certificates' bucket, and the Amazon S3 SDK, Node.js, Express, Axios, and React.js are used to upload, download, and manage the files. A 'AddCertificate' component allows students to upload files on the user interface and stream certificates when requested by the client.

## 4.2.6 Certificate Evaluation and Approval Module

This module facilitates the evaluation and approval process for submitted certificates by teacher users. Teachers review and approve certificates with recommended points or reject them according to university guidelines. MongoDB is used to store the status of each certificate in a 'certificates' collection, Express to create routes for updating certificate status and fetching certificate categories, and Axios to make HTTP requests for status updates. A 'Evaluate Certificate' component in React.js displays and modifies certificate status on the user interface.

## 4.2.7 Activity Point Allocation Module

This module calculates and manages the allocation of activity points to students based on the approved certificates. It ensures that the correct number of points is awarded for each validated activity. To implement this module, we have used MongoDB to store the points of each student in a collection called 'scores'. We have used Express to create the routes for updating and fetching scores. Additionally, we have used Axios to make HTTP requests from the client to the server for scores. Furthermore, we have used React.js to create a sidebar component that displays and updates scores on the user interface.

## 4.2.8 Report Generation Module

This module generates various reports related to activity points and certificates. The report generation is available both batch-wise and student-wise. To implement this module, we

have used jsPDF to generate PDF documents from the data using JavaScript. We have used Express to create the routes for generating and downloading data for reports. Additionally, we have used Axios to make HTTP requests from the client to the server for the data. Furthermore, we have used React.js to create a component called 'Report Generator' that allows teachers to generate and download reports on the user interface.

## 4.3 Summary

In this chapter, we discussed the implementation of the system using various tools and software packages. We provided detailed explanations of React.js, jsPDF, Axios, Node.js, Express, MongoDB, and Amazon S3. Each module in the project was then explained in simple language, detailing its implementation.

# Chapter 5

## Testing

Software Testing is a method to check whether the actual software product matches expected requirements and to ensure that the software product is defect-free. It involves the execution of software/system components using manual or automated tools to evaluate one or more properties of interest. The purpose of software testing is to identify errors, gaps, or missing requirements in contrast to actual requirements.

### 5.1 Testing Strategies Used

In this section, the strategies employed to test our project will be discussed. Testing played a critical role in ensuring the reliability and functionality of our web-based Activity Points Management System. The testing phase consisted of two essential methodologies: Unit Testing and Integration Testing.

#### 5.1.1 Unit Testing

Unit Testing formed the foundation of our testing strategy. This involved testing individual components of the system in isolation to ensure that each component was functioning correctly and met our requirements. The primary technique we used for unit testing was to write and execute test cases for each component, verifying that it produced the expected output for a given input. This allowed us to identify and fix any issues with the components before integrating them into the larger system.

##### 5.1.1.1 Black Box Testing

Black Box Testing was employed to verify the external behavior of the software components without examining their internal structure. Test cases were designed based solely on the system's specifications and requirements.

- Verifying the correct interaction and integration of different components.
- Conducted Black Box Testing with real data from students and teachers.
- Provided our application to a group of students and tested it using their actual activity points and certificates
- Verified the system's functionality and performance in a real-world scenario.

- Ensured that the system met the needs and expectations of its intended users.

## 5.1.2 Integration Testing

Integration Testing was a crucial phase in our testing process. During this phase, we evaluated the compatibility and communication among different integrated components of the system. This involved testing the interactions between components to ensure that they worked together seamlessly and produced the desired results. By conducting Integration Testing, we were able to identify and fix any issues with the integration of components, ensuring that the system functioned as a cohesive whole.

### 5.1.2.1 Postman API Testing

As part of Integration Testing, we conducted Postman API Testing to assess the functionality and performance of our system's API endpoints. Using the Postman tool, we interacted with the APIs and assessed their responses. The main aspects of Postman API Testing in our project include:

- Validating the correctness of API responses for various inputs and scenarios.
- Assessing the data integrity during data transfer between the client and the server.
- Ensuring the security of API endpoints against potential vulnerabilities.
- Verifying the performance and scalability of the API under different loads.
- Checking the compatibility of the API with different platforms and devices.
- We used Postman to send a variety of requests to our API endpoints, covering a range of inputs and scenarios.

## 5.2 Testing Results

### 5.2.1 Results of Black Box Testing

Our implementation of Black Box Testing for unit testing has been successful in verifying the external behavior of the software components without examining their internal structure. Test cases were designed based solely on the system's specifications and requirements, allowing evaluation of system from an end-user's perspective.

One of the key successes of our Black Box Testing was the ability for teachers to mark certificates and for students to upload them. The system also provided notifications to students for specific test cases, ensuring that they were kept informed about their progress. By deriving test cases solely from software specifications and user requirements, we were able to ensure that the system met the specified requirements and provided the expected

functionality.

Overall, our use of Black Box Testing for unit testing has been instrumental in delivering a robust and high-quality system that meets the needs of both students and teachers.

## 5.2.2 Results of Integration Testing

Our integration testing with Postman API Testing was successful in assessing the functionality and performance of our system's API endpoints. We were able to interact with the APIs through the Postman tool and evaluate their responses and validating them. Teacher-student authentication, certificate upload, marking certificates, retrieving category-wise scores from the database, and viewing certificates were among the key endpoints tested. We were able to find and fix problems with data integrity during client-to-server data transfers by using Postman API Testing. Overall, our use of Postman API Testing for integration testing has been essential in delivering a reliable and high-quality system that satisfies the needs of both students and teachers.

## 5.3 Summary

This chapter introduces software testing and its various strategies. Black Box Testing verifies external behavior without examining internal structure. Unit testing tests individual components in isolation. Integration testing tests how different components work together. These techniques were used to test our web-based Activity Points Management System.

# Chapter 6

## Results

### 6.1 Login & Register pages

One of the key results achieved through the development of SkillVine is the implementation of a secure authentication and authorization system for both students and teachers. This system uses OAuth 2.0, to enable users to securely authenticate with the system and grant access to their data without sharing their login credentials. This ensures that only authorized users can access the system and their data is protected at all times.

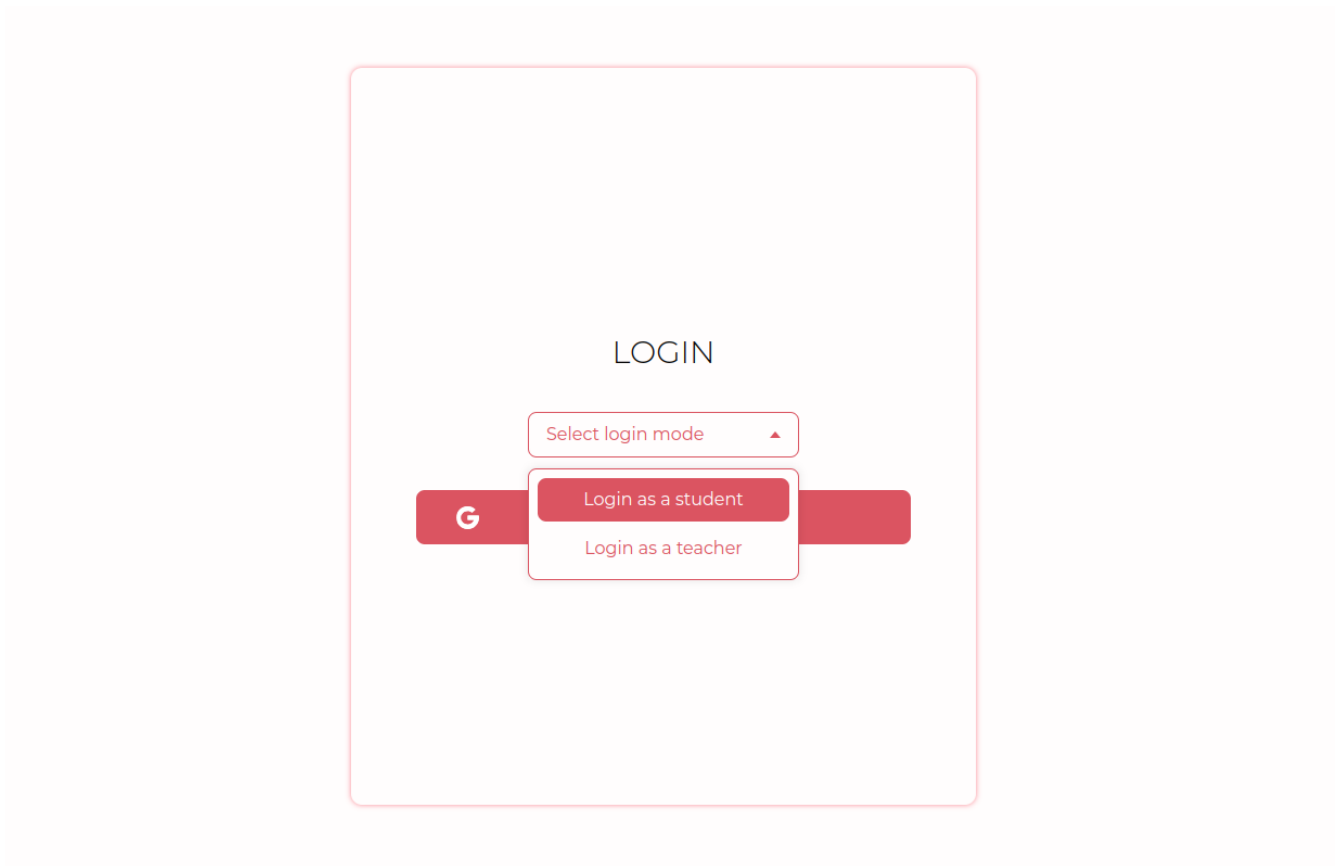


Figure 6.1: Login page and Register page for student and teacher

## 6.2 Student Dashboard

SkillVine also features an intuitive and user-friendly dashboard for viewing notifications and updates, providing a centralized location for students to view important information related to their activity points and certificates.

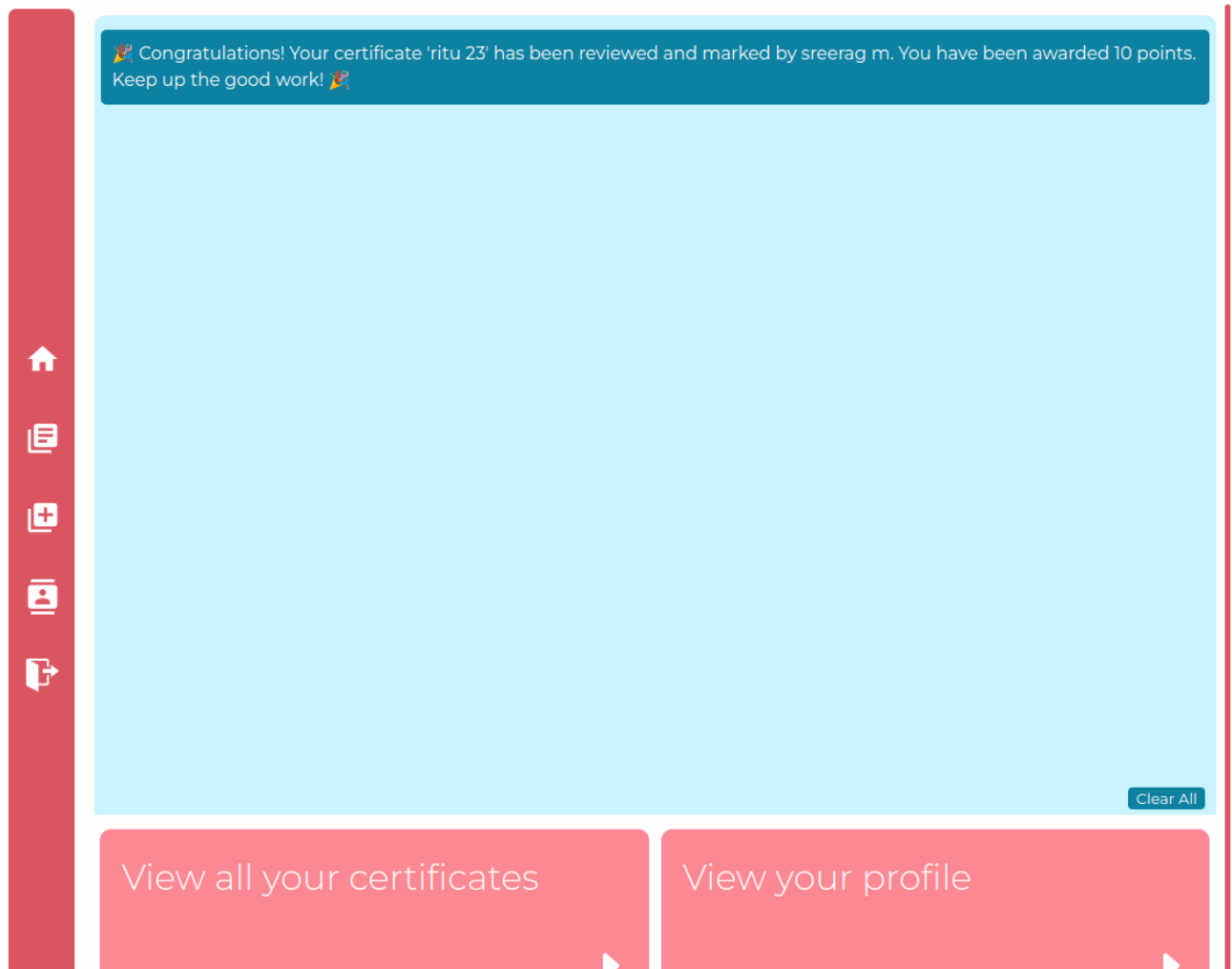


Figure 6.2: Student dashboard with notification

## 6.3 Student Certificates View

Additionally, SkillVine includes a centralized repository for storing, viewing, and managing certificates used for activity point allotment. This repository provides a single location for students to access and manage their certificates, streamlining the certification process and improving the overall user experience.

Name	Date	Activity	Level	Edit/Delete
<b>First Year</b>				
Internship OUT OF THE BOX ,OFB ACADEMY	2022-02-02	Industrial Training/Internship (atleast for 5 full days)	Level 1	
<b>Second Year</b>				
Internship Revertech IT Solutions Pvt. Ltd.	2022-02-01	Industrial Training/Internship (atleast for 5 full days)	Level 2	
Kelli'22 Poster Designing Competition	2022-05-17	Literary arts - First Prize	Level 1	
FUTSAL 3x3	2022-04-28	Sports - Participation	Level 2	
<b>Third Year</b>				
DSA mooc 2	2023-06-23	MOOC with final assessment certificate	Level 5	
DSA using Java	2023-06-14	MOOC with final assessment certificate	Level 4	
IV Spectrum Softtech Solutions (P) Ltd	2022-11-29	Industrial/Exhibition visits	Level 1	
<b>Fourth Year</b>				

Figure 6.3: Student certificate view

## 6.4 Real Time Score Section

Another key result achieved through the development of SkillVine is the implementation of a real-time tracking and monitoring system for activity points approved by teachers. This system provides transparency and accountability in the activity points allocation process, helping students stay motivated and engaged.

Name	Date	Activity	Level	Edit/Delete
<b>First Year</b>				
Internship OUT OF THE BOX ,OFB ACADEMY	2022-02-02	Industrial Training/Internship (atleast for 5 full days)	Level 1	
<b>Second Year</b>				
Internship Revertech IT Solutions Pvt. Ltd.	2022-02-01	Industrial Training/Internship (atleast for 5 full days)	Level 2	
Kelli'22 Poster Designing Competition	2022-05-17	Literary arts - First Prize	Level 1	
FUTSAL 3x3	2022-04-28	Sports - Participation	Level 2	
<b>Third Year</b>				
DSA mooc 2	2023-06-23	MOOC with final assessment certificate	Level 5	
DSA using Java	2023-06-14	MOOC with final assessment certificate	Level 4	
IV Spectrum Softtech Solutions (P) Ltd	2022-11-29	Industrial/Exhibition visits	Level 1	
<b>Fourth Year</b>				

No of activity points you have now:

65

No of activity points needed:

35

Total no. of activity points required:

100

X

Figure 6.4: Real time score section



## 6.5 Certificate Upload

One of the key results achieved through the development of SkillVine is the implementation of a comprehensive certificate management system with options to edit, delete, and upload certificates. This system provides a user-friendly interface for managing certificates, streamlining the certification process and improving the overall user experience.

The screenshot displays the 'Certificate Upload and Edit' interface. On the left is a red sidebar with icons for home, list, add, user, and share. The main form contains the following fields:

- Event Name:** A text input field containing 'test'.
- Date Issued:** A date picker field showing '16/07/2023'.
- Duration:** A text input field containing '45'.
- Remarks:** A text area containing 'dfsfsf'.

On the right side, there are four dropdown menus:

- Select year of study:** A dropdown menu.
- Select category:** A dropdown menu.
- Select event:** A dropdown menu.
- Select level:** A dropdown menu with a list of options:
  - \*Level I College Events
  - \*Level II Zonal Events
  - \*Level III State/ University Events
  - \*Level IV National Events
  - \*Level V International Events

Below these dropdowns is a file upload section with the text 'Upload certificate file' and a 'Choose file' button. To the right of the button, it says 'No file chosen'. At the bottom right is a large red button labeled 'Update certificate details'.

Figure 6.5: Certificate upload and edit section

## 6.6 Teacher Dashboard

SkillVine also features a dashboard for teachers, with options to view student batches, certificates uploaded by students, and batch-wise and student-wise reports. This dashboard provides a user-friendly interface for managing classes and tracking student progress.

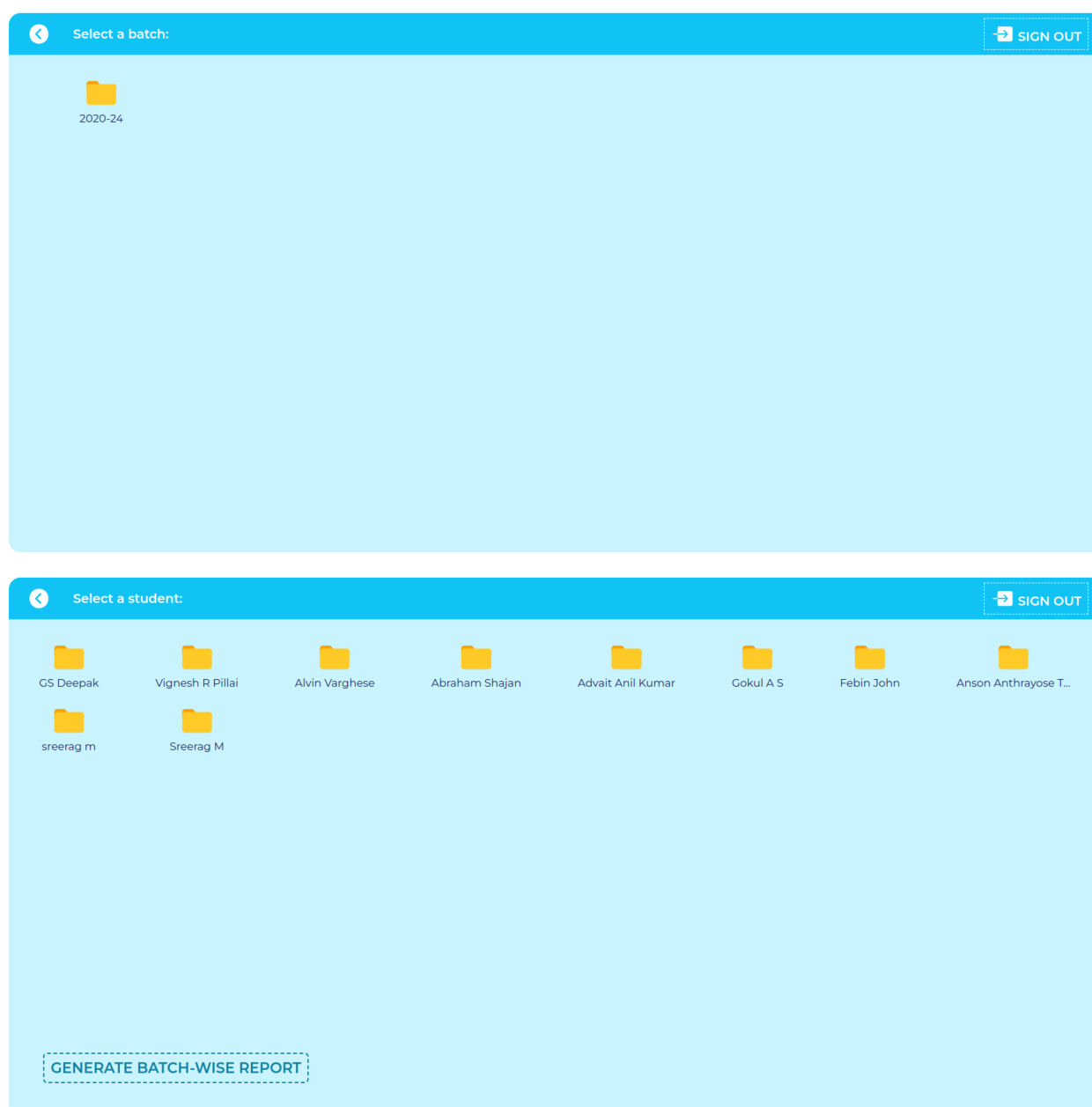


Figure 6.6: Teacher dashboard

## 6.7 Certificate Evaluation and Approval section

Additionally, SkillVine includes an advanced certificate assessment system with options to approve, decline, or edit certificate submissions. This system provides a user-friendly interface for managing and assessing students' certificates, streamlining the certification process and improving the overall user experience for teachers.

GO BACK

Certificate Details

Certificate Name:

Internship OUT OF THE BOX ,OFB ACADEMY

Date of event:

2022-02-02

Duration of event:

56

Description:

OUT OF THE BOX ,OFB ACADEMY

\*Level I College Events

\*Level II Zonal Events

\*Level III State/ University Events

\*Level IV National Events

\*Level V International Events

Certificate Link

Here is the link to the [Certificate PDF/Image!](#)

Category details

Currently Selected:

YEAR: First Year

CATEGORY: Professional Self Initiatives

ACTIVITY: Industrial Training/Internship (atleast for 5 full days) @ LEVEL: Level 1

Change year

Change category

CALCULATE

Certificate Marking

The points based on the categories are:

Enter remark

REJECT CERTIFICATE

ACCEPT CERTIFICATE

Figure 6.7: The Certificate Evaluation and Approval section

## 6.8 Report Generation

SkillVine also features an automated report generation system for tracking student progress and activity point allocation. This system provides a user-friendly interface for generating reports on student progress, streamlining the certification process and improving the overall user experience for teachers.

DELETE STUDENT

GENERATE REPORT

GO BACK

PENDING

APPROVED

REJECTED

REFRESH

Date

Activity

Level

Mark

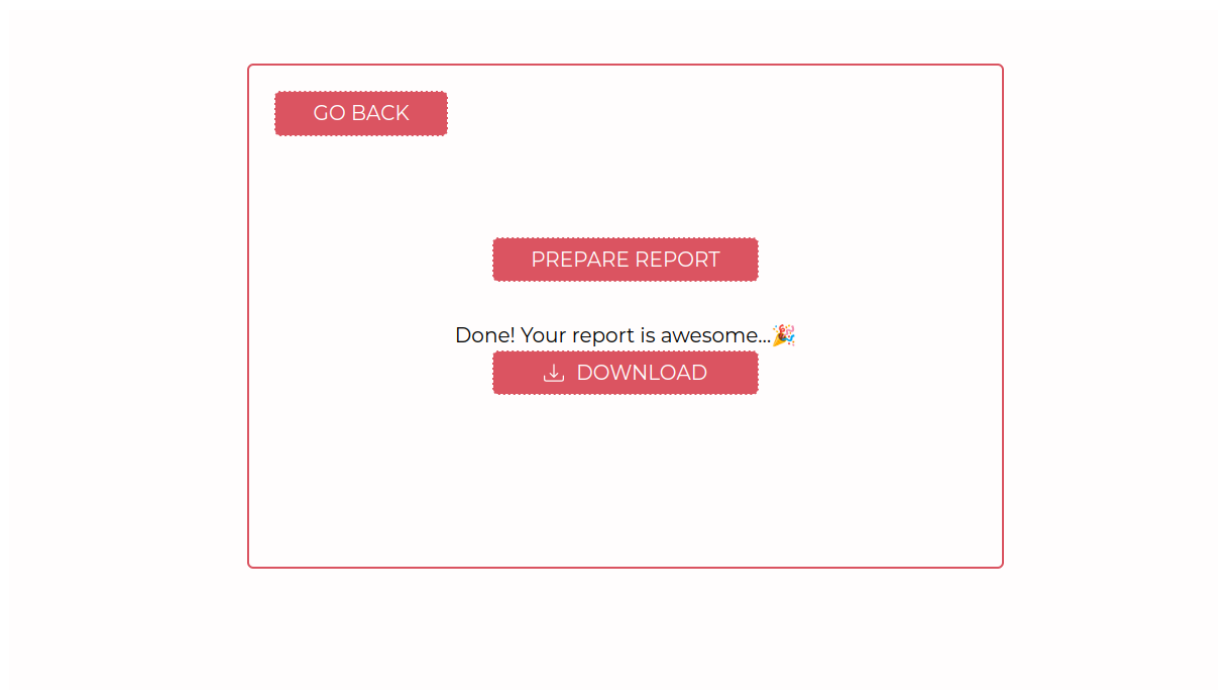


Figure 6.8: Report generation

## 6.9 Summary

The following objectives were successfully accomplished during the course of this project:

1. We have successfully created a web-based Activity Points Management System, SkillVine, that simplifies and enhances the way student activity points allocation is handled.
2. SkillVine offers several features for students, including secure authentication and authorization, real-time tracking of activity points approved by teachers, viewing notifications, a centralized repository for storing, viewing, and managing certificates used for activity point allotment, and a comprehensive certificate management system with options to edit, delete, and upload certificates.
3. SkillVine also offers several features for teachers, including options to view certificates uploaded by students, and batch-wise reports, a system with options to approve, decline, or edit certificate submissions, and an automated report generation system.

# Chapter 7

## Conclusion

### 7.1 Recommendations for Future Works

For performing future research on this work, we recommend the following.

1. Integration with other educational platforms and systems through the use of APIs and data exchange protocols to provide a seamless experience for students and teachers.
2. Development of a cross-platform mobile app using responsive design and mobile development frameworks to provide easy access to the platform on-the-go.
3. Addition of advanced analytics and reporting features using data visualization tools and machine learning algorithms to provide insights into student performance and progress.
4. Expansion of the platform's capabilities to support additional types of certificates and activities through modular design and flexible data structures, allowing for greater flexibility in the allocation of activity points.
5. Scaling the platform to a university level through the development of a centralized, cloud-based infrastructure and the implementation of advanced data management techniques and algorithms.

### 7.2 Summary

SkillVine is a web-based Activity Points Management System that simplifies and enhances the way student activity points allocation is handled. The platform integrates and automates the existing process of recording, evaluating, and awarding certificates, saving time and resources for both students and teachers. SkillVine offers a convenient, efficient, and transparent solution for managing student activity points, with features such as secure authentication, real-time tracking, intuitive dashboards, and advanced certificate assessment. Overall, SkillVine is a well-designed and innovative platform that has the potential to greatly improve the quality and reliability of the certification process. The system has been designed with both students and teachers in mind, providing a user-friendly interface and comprehensive features for managing certificates and activity points. Despite its many strengths, there is always room for improvement and future enhancements could include additional features such as audio effects or a save/load feature to further enhance the user experience. We conclude by recommending that further research and development be undertaken to continue to improve and enhance the SkillVine platform.

# Appendix A

## Software Requirement Specification

### A.1 Introduction

#### A.1.1 Purpose

This document provides a comprehensive overview of SkillVine - an Activity Points Manager, including its purpose, features, interfaces, capabilities, and operational and developmental constraints.

#### A.1.2 Document Conventions

In this SRS document, we have followed the following document conventions:

- Headings are written in Roman font.
- Body text is written in Times New Roman font.
- Emphasized text is formatted in italics.
- Important information is highlighted in bold. Abbreviations are defined when first used and a list of abbreviations is provided at the end of the document in Glossary.
- Bulleted lists are formatted with bullets (•) and indented text.
- Numbered lists are formatted with numbers and indented text.

#### A.1.3 Intended Audience and Reading Suggestions

The Software Requirements Specification (SRS) document is meant for the project evaluation team, project supervisor, and project guide who are responsible for evaluating the team's work.

To ensure that all project requirements have been addressed, it is recommended that the evaluation team carefully review this document. The SRS document contains a comprehensive overview of the project's scope, including functional and non-functional requirements, design and implementation constraints.

Technical terms and acronyms used throughout the document are defined in the Glossary (Appendix A). Additionally, the Issues List (Appendix B) provides a dynamic list of unresolved requirements issues such as TBDs, pending decisions, and conflicts that require resolution.

### A.1.4 Project Scope

The Activity Points Management System focuses on organising and automating the evaluation of activity points. Since a system will be in place for keeping track of each student's certificates and details, it is beneficial for both teachers and students. It has features like the ability to store student certificates, view activity point status in real-time, sort, and search the list of certificates for students. Additionally, allowing teachers to review and grade student-uploaded certificates.

## A.2 Overall Description

### A.2.1 Product Perspective

The Activity Points Management System is a one-of-a-kind solution and currently, there is no other system that performs the same task. The system can replace the current manual procedures for the task, as it is capable of adapting to this purpose.

Our system automates the entire process of grading and managing certificates. It can also offer a safe and convenient storage system for students to keep their certificates. The user-friendly interface makes it a breeze for both teachers and students to navigate. This can simplify the way teachers monitor each student's progress and number of activity points achieved.

### A.2.2 Product Features

The system contains the following major features:

- User authentication for students & teachers.
- Interactive dashboards for students & teachers.
- Adding, editing, & displaying certificates with details.
- Real-time viewing of activity points & certificate status
- Storage feature to organize & maintain records of all certificates.
- Notification system for updates to previously interacted certificates.
- Sorting, searching, & filtering of certificates for students & teachers.
- Grading or rejecting certificates with remarks for teachers.
- Automatic calculation & updating of activity points in the database.

### A.2.3 User Classes and Characteristics

The system will consist of two fundamental user roles - students and teachers.

1. Students: These users will upload certificates and their details and also have their own dashboards. These users are interested in knowing their activity points real time.
2. Teachers: These users can add, delete, update and view the students' certificates' details. They can also assign points to certificates uploaded by the students.

### A.2.4 Operating Environment

The activity points management application will be web-based and can be accessed through commonly used web browsers such as Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge. It will work seamlessly on any operating system that supports web browsers, including Windows, MacOS, Linux, iOS, and Android, and can be accessed through standard personal computers or mobile devices with internet connectivity. The application will be self-contained and won't require any additional software to function.

### A.2.5 Design and Implementation Constraints

1. Hardware Limitations:
  - The system should be designed to run on standard desktop or laptop hardware with reasonable specifications, as per industry standards.
  - The system should be able to handle a large number of concurrent users without crashing or significant performance degradation.
2. Technology Limitations:
  - The front-end of the system will be developed using ReactJS framework.
  - The back-end of the system will be developed using Express Js. & Node Js. run-time environment.
  - The database used will be MongoDB.
  - The system should adhere to RESTful API design principles.
  - The system should be compatible with major web browsers such as Google Chrome, Mozilla Firefox, and Safari.
3. Security Constraints:
  - The system should ensure the confidentiality, integrity, and availability of user data.
  - The system should incorporate security measures such as encryption, access control, and secure communication protocols.
  - The system should be tested for vulnerabilities and potential security threats before deployment.



## A.2.6 User Documentation

Along with the software, the following user documentation components will be delivered:

- User manual: A comprehensive guide to using the Activity Points Management System, including step-by-step instructions and screenshots. The user manual will be delivered in PDF format and will also be available online.

## A.2.7 Assumptions and Dependencies

- The implementation of the system relies on the hosting service and its underlying infrastructure.
- The functionality of the system depends on the availability and compatibility of the required software frameworks and libraries, such as React, Flask, and their dependencies.

# A.3 System Features

## A.3.1 Authentication of Students and Teachers

### A.3.1.1 Description and Priority

Description: This feature allows students and teachers to securely authenticate with the system using their login credentials. Moreover, the app has an email-based verification process. The students are directed to their dashboards and the teachers are instantly recognized as admin.

- Priority: High
- Benefit: 9 (The feature provides significant benefits by ensuring that only authorized students and teachers can access the system.)
- Penalty: 9 (The absence of this feature would pose serious security issues.)
- Cost: 7 (The cost of implementing this feature is moderate to high)
- Risk: 2 (The risk associated with implementing this feature is low)

### A.3.1.2 Stimulus/Response Sequences

- The student or teacher accesses the login page of the system.

- The user enters their login credentials (e.g., username and password).
- The system verifies the user and redirects to home page.

### A.3.1.3 Functional Requirements

- The system must provide a secure login page for students and teachers to enter their login credentials.
- The system must verify and authenticate users based on their login credentials.
- If the user enters incorrect credentials, error will be displayed.

## A.3.2 Dashboard for students and teachers

### A.3.2.1 Description and Priority

Description: This feature provides students with a dashboard displaying detailed information about their activity points, certificates, and profile information. Teachers, after authentication, can select a batch and a student to view their list of certificates.

- Priority: High
- Benefit: 8 (The feature provides significant benefits by giving students easy access to important information and for teachers, easy access to each student's profile)
- Penalty: 6 (Not having this feature would result in some inconvenience for students and teachers.)
- Cost: 5 (The cost of implementing this feature is moderate)
- Risk: 2 (The risk associated with implementing this feature is low)

### A.3.2.2 Stimulus/Response Sequences

- The user logs in to the system.
- The system displays the student's dashboard or teacher's space to select branch and student name respectively.
- The student can view detailed information about their activity points and certificates on the dashboard.
- The student can add or view certificates from the links in dashboard. The teacher can edit or mark the certificates of a student.

### A.3.2.3 Functional Requirements

- The system must provide a dashboard for students that displays detailed information about their activity points and certificates and a teacher's space to select branch and student.
- The system must display up-to-date information on the user's dashboard.
- The system must allow users to view their dashboard after logging in.

## A.3.3 Uploading, Storing, & Editing certificates and their details.

### A.3.3.1 Description and Priority

Description: This feature allows students to upload and store certificates for the events they participated. Moreover it allows students to edit the details of a certificate. It allows accurate tracking and recording of student progress.

- Priority: High
- Benefit: 9 (The feature provides significant benefits by ensuring that the main purpose of system is fulfilled.)
- Penalty: 9 (The absence of this feature would make the system functionless.)
- Cost: 8 (The cost of implementing this feature is moderate to high)
- Risk: 3 (The risk associated with implementing this feature is low)

### A.3.3.2 Stimulus/Response Sequences

- Students upload certificates, teachers grade and mark activities.
- Students can edit the details of a specific certificate they uploaded.

### A.3.3.3 Functional Requirements

- User and instructor interface for uploading, viewing, and storing certificates, system for integration with the overall grading system.

## A.3.4 Viewing Status of Certificates and Grading them

### A.3.4.1 Description and Priority

Description: This feature allows students to view the status of their certificates and provides teachers with a grading system for activity points.

- Priority: High
- Benefit: 9 (The feature provides significant benefits by enabling students to track their progress and understand the grading system used for activity points.)
- Penalty: 7 (The absence of this feature would pose significant issues for students who are unable to track their progress and understand the grading system used for activity points.)
- Cost: 6 (The cost of implementing this feature is moderate to high, depending on the complexity of the system.)
- Risk: 4 (The risk associated with implementing this feature is moderate, as there is a potential for technical issues or errors in displaying the information correctly.)

### A.3.4.2 Stimulus/Response Sequences

- The student logs in to their account.
- The student navigates to the activity point section of their account.
- The system displays the status of each certificate and the grading system used for awarding activity points.

### A.3.4.3 Functional Requirements

- The system must provide a user interface for accessing the activity point section of the student's account.
- The system must display the status of each certificate and the grading system used for awarding activity points.
- The certificate status and grading system must be accurate and updated in real-time.
- The system must be secure to ensure that only authorized users can access the activity point section.

## A.3.5 Notification system for students and Teachers

### A.3.5.1 Description and Priority

Description: This feature enables students to view the status of their certificates in the system and understand any changes made by teachers. Teachers can accept or reject certificates based on their evaluation, and students will be notified of these changes.

- Priority: High
- Benefit: 9 (The feature provides significant benefits by enabling students to track the progress of their certificates and understand any changes made by teachers.)
- Penalty: 8 (The absence of this feature would pose serious issues for students who are unable to track the status of their certificates and understand any changes made by teachers.)
- Cost: 7 (The cost of implementing this feature is moderate to high, depending on the complexity of the system.)
- Risk: 5 (The risk associated with implementing this feature is moderate, as there is a potential for technical issues or errors in displaying the information correctly or notifying students of changes.)

### A.3.5.2 Stimulus/Response Sequences

- The student accesses the system and views the status of their certificates.
- If a teacher rejects a certificate, the system will notify the student of the rejection and provide feedback from the teacher.
- If a teacher accepts a certificate, the system will notify the student of the acceptance and update the status of the certificate.
- The student can view the feedback from the teacher and make necessary changes to the certificate.

### A.3.5.3 Functional Requirements

- The system must allow students to view the status of their certificates and understand any changes made by teachers.
- The system must enable teachers to accept or reject certificates based on their evaluation.
- If a certificate is rejected, the system must notify the student and provide feedback from the teacher.

- If a certificate is accepted, the system must notify the student and update the status of the certificate.
- The system must allow students to view the feedback from the teacher and make necessary changes to the certificate.

## A.3.6 Report Generation

### A.3.6.1 Description and Priority

Description: This feature enables teachers to generate reports based on batch and student-wise criteria, which can be accessed and used at any time. The system generates customized reports that include relevant data on student performance, grades, attendance, and other criteria. Teachers can generate reports at the last minute for their immediate use, making it a convenient and efficient tool for assessing student progress.

- Priority: High
- Benefit: 9 (The feature provides significant benefits by enabling teachers to generate customized reports based on specific criteria and assess student performance efficiently and conveniently.)
- Penalty: 8 (The absence of this feature would pose serious issues for teachers who rely on reports to evaluate student progress and identify areas for improvement, especially in situations where immediate access to data is necessary.)
- Cost: 7 (The cost of implementing this feature is moderate to high, depending on the complexity of the system and the number of criteria that reports need to be generated for)
- Risk: 5 (The risk associated with implementing this feature is moderate, as there is a potential for technical issues or errors in generating accurate and relevant reports)

### A.3.6.2 Stimulus/Response Sequences

- The teacher accesses the report generation feature in the system.
- The teacher selects the batch or student-wise criteria for the report.
- The system generates a customized report that includes information on student performance, grades, attendance, and other relevant data.
- The report is immediately available for the teacher's use.

### A.3.6.3 Functional Requirements

- The system must provide a user interface for accessing the report generation feature.
- The system must allow the teacher to select batch or student-wise criteria for the report.
- The system must generate reports that include information on student performance, grades, attendance, and other relevant data.
- The reports must be customized based on the selected criteria and accurately reflect the student's performance.
- The system must ensure that the generated reports are immediately available for the teacher's use and accessible at any time.
- The system must ensure that only authorized users can access the report generation feature and the generated reports.

## A.4 External Interface Requirements

### A.4.1 User Interfaces

The user interface consists of the following components

#### **Login and Registration page:**

- Google Authentication - Users will need to input their institution email and password in order to access the system. Users who do not have an existing account will be redirected to a page to enter further details to register.

#### **Student dashboard page:**

- Navigation Bar - Users will be able to navigate areas of the system, such as the user's profile, detailed activity points information, certificates' list etc.
- Activity Points Summary- This displays a summary of the user's activity points, including their total points, points earned in different categories, and points required for certain rewards or achievements.
- Certificates- This displays a list of the user's certificates, including the certificate name, date earned, and any relevant details.

### Teacher dashboard page:

- Navigation Menu - This provides links to different areas of the system, to list students or the batches available, view certificates of a student, verify and remark sections etc.
- Summary- Status of certificate verification, number of certificates uploaded category wise.

### View and edit pages:

- For Students -To view their uploaded certificates, edit details, re-upload, view status and remarks .
- For Teachers-View, edit, verify and remark certificates based on the info provided.

## A.4.2 Hardware Interfaces

The hardware of the user's device will be primarily interfaced with by the software product. Different device types, such as desktops, laptops, tablets, and mobile phones, are supported by the system. Modern web browsers like Apple Safari, Mozilla Firefox, and Google Chrome will all work with the software.

Sending and receiving data through HTTP requests and responses will be necessary for the data and control interactions between the software and the hardware. The system will communicate with the back end server, which will store and handle the data, using a RESTful API. To enable user interaction with the system, the software will communicate with the hardware of the device, such as the display, keyboard, and mouse.

The software will use HTTPS for secure communication and JSON for data exchange in its communication protocols. Additionally, the software will use the user's device's local storage to save user preferences and cached information for offline use. The system will follow best practises for security, performance, and accessibility in addition to industry standards for web development.

## A.4.3 Software Interfaces

- Back end framework:The activity point manager will use the Express Js. (latest version) for the back end. development.
- Database: The app will use MongoDB (latest version) as the database management system.
- Data items: The app will receive data such as user details , different certificate data from storage of the back end, back end will also analysis different certificates based on their credentials and recommend the activity point for the certificates and it provides student details based on different batch.



- Services: The app will need services such as teacher student authentication , an external cloud service for storing different certificates, data processing and storage in database , and recommendation of activity point.
- Communications: The communication between the app and external APIs will be done using RESTful API calls. The communication between the app and the back end will be done using HTTP requests and responses.

#### A.4.4 Communications Interfaces

- The activity point manager app requires various communication functions to operate seamlessly .These communication functions include web browser ,network server communication protocols ,electronic forms and more
- The app will use the HTTP protocol for communication between the front end and back end components. REST-ful API calls will be used for communication between the app and external APIs.
- All the server client communication will follow standard formatting guidelines to ensure that they are understood and processed correctly. For example, requests and responses will follow the JSON format.
- The app will adhere to the relevant communication standards such as HTTP and SMTP for email communication. Communication security and encryption will be implemented using SSL/TLS protocols to secure data transfer between the app and the server.
- Data transfer rates and synchronization mechanisms will be optimized to ensure that the app operates efficiently, and data is synced across all devices in real-time. In addition, the app will be designed to handle network interruptions, so data can be transmitted and received without any data loss or corruption.
- Overall, the activity point manager app communication functions will be designed to ensure that teachers and students can interact with the app seamlessly and receive the desired results with minimum delays or errors.

### A.5 Other Nonfunctional Requirements

#### A.5.1 Performance Requirements

The following performance requirements must be met by the activity point management web app for students:

1. Response Time: The app must respond to user requests within 2 seconds. This includes all API calls, page loads, and data retrieval operations. The response time must not increase with an increase in the number of users.

2. Scalability: The app must be scalable to handle a large number of users. The system must be able to handle at least 1000 concurrent users without any decrease in performance.
3. Reliability: The app must be highly reliable and available at all times. The system must have an uptime of at least 99.9% with a maximum downtime of 5 minutes per month.
4. Data Retrieval Time: The app must be able to retrieve data related to activity points within 1 second. This includes certificate validation and activity point calculation.
5. Data Processing Time: The app must be able to process and store user data within 3 seconds. This includes certificates uploaded by students, activity points awarded by teachers, and any other relevant information.
6. Real-time updates: Any updates to activity points, certificates, or remarks provided by teachers must be reflected in real-time on the app. The app must be able to handle at least 10 updates per second without any decrease in performance.

## A.5.2 Safety Requirements

1. Data Security: The product must ensure that user data is protected from unauthorized access, modification, or deletion. The app shall use secure authentication mechanisms to prevent unauthorized access. Additionally, the app shall have proper access controls in place to ensure that only authorized personnel can access the app's data and functionality.
2. Data Privacy: The project must guarantee that user data, including personal data and activity certificates, is secure and private. The app must abide by all applicable data privacy laws and rules, including the California Consumer Privacy Act (CCPA) and the General Data Protection Regulation (GDPR). User information must be safely kept in the database and transmitted using encryption.
3. Performance under load: The product must be able to perform reliably and efficiently under heavy loads, such as during peak usage periods or high-volume data retrieval operations. The system must respond to user requests within a reasonable time frame, not exceeding 3 seconds.
4. Disaster recovery: The product must have a disaster recovery plan in place to ensure that data and functionality can be restored in the event of a disaster, such as a server failure or data breach.

## A.5.3 Security Requirements

1. The product must ensure the confidentiality and integrity of user data, including personal information, activity logs, and point balances.

2. All sensitive user data must be encrypted both at rest and in transit, using industry-standard encryption algorithms.
3. The product must implement secure communication protocols to protect user data during transmission, including SSL/TLS encryption and HTTP strict transport security.
4. The product must provide user authentication mechanisms that comply with relevant security standards, such as multi-factor authentication or password complexity requirements.
5. The product must undergo regular security testing and vulnerability assessments, including penetration testing and code review. Access controls must be implemented to ensure that only authorized users can view or modify sensitive data.
6. The product must provide users with transparency and control over their data, including the ability to view and delete their personal information.
7. The product must comply with all relevant data privacy and protection laws and regulations, including but not limited to GDPR, CCPA, and the Health Insurance Portability and Accountability Act (HIPAA) if applicable.
8. The product must ensure that any third-party services or APIs used also meet the same security and privacy requirements as the product itself.
9. The product must obtain any necessary security or privacy certifications or approvals required for its use or distribution, such as ISO 27001 or SOC 2 Type II.

#### A.5.4 Software Quality Attributes

1. Usability: The product shall be easy to learn and use, with a UI/UX that is intuitive and consistent. Users should be able to complete common tasks quickly and efficiently, with minimal errors.
2. Reliability: The product shall be dependable and consistently perform as expected. It shall have a low rate of failure, and any errors or malfunctions shall be easily identifiable and recoverable.
3. Maintainability: The product shall be easy to maintain and update, with well-documented code and clear instructions for making changes. Updates and bug fixes should be able to be implemented quickly and without disrupting other parts of the system.
4. Portability: The product shall be easily portable to different platforms and environments, with minimal changes required to adapt to new systems.
5. Interoperability: The product shall be able to integrate with other systems and platforms seamlessly, without requiring significant modifications or customizations.
6. Scalability: The product shall be able to handle increased usage and data storage requirements without a significant decrease in performance or stability.

7. Security: The product shall have strong security measures in place to protect user data and prevent unauthorized access or attacks.
8. Performance: The product shall be designed to have fast response times and minimal latency, with the ability to handle a high volume of user requests and data processing tasks.
9. Performance: The product shall be designed to have fast response times and minimal latency, with the ability to handle a high volume of user requests and data processing tasks.

## A.6 Other Requirements

### A.6.1 User Documentation

- User Documentation: Specify the user documentation that must be provided with the product, such as user manuals, online help, or training materials.

### A.6.2 Database requirements

- The database should be designed to efficiently store and retrieve activity point data.
- The database schema should be normalized to reduce redundancy and ensure data integrity.
- The database should be designed to handle a large number of concurrent users. The database should support backup and recovery procedures to ensure data availability in case of system failure.
- The database should be secured to prevent unauthorized access or modification of activity point data.
- The database should be optimized for fast query processing and report generation.
- The database should have sufficient capacity to handle future growth in activity point data.
- The database should support data import and export functionalities to allow easy data transfer between systems

### A.6.3 Internationalization Requirements

- The application must support different date and time formats based on the user's locale.

- The application must support English as the primary language. The application must also be designed to support additional languages in the future, may the regional language of the users.
- For there to be translation, all text that is displayed to the user, including labels, messages, and error messages, must be externalised to resource files.

## **A.6.4 Legal Requirements**

- Intellectual property rights: The product must not infringe on any third-party intellectual property rights, including patents, trademarks, and copyrights.
- The software must comply with all applicable export control laws and regulations, including those related to the export of encryption technology.

## **A.6.5 Reuse Objectives**

- Modular design: The project should be designed with modularity in mind, so that components can be easily reused in future projects.
- Standardization: The project should follow widely recognized standards for coding, documentation, and design, so that other developers can easily understand and reuse the artifacts.
- Open-source: Consider releasing the project as open-source software, with appropriate licensing, so that it can be easily reused and modified by other developers.
- Compatibility: The project should be designed to be compatible with other systems and technologies, so that its components can be easily integrated into future projects

# References

- [1] React. React – a javascript library for building user interfaces. <https://reactjs.org/>, 2023. Accessed on 2023-04-11.
- [2] Next.js. Next.js - the react framework. <https://nextjs.org/docs>, 2023. Accessed on 2023-04-19.
- [3] Amazon Simple Storage Service (Amazon S3) Cloud Storage — AWS. <https://docs.aws.amazon.com/s3>, 2023. Accessed on 2023-05-01.
- [4] Express. Express - node.js web application framework. <https://expressjs.com/>, 2023. Accessed on 2023-04-30.
- [5] Google Cloud Documentation | Google Cloud. <https://cloud.google.com/docs>, 2023. Accessed on 2023-04-11.
- [6] MongoDB. MongoDB - the most popular database for modern apps. <https://docs.mongodb.com/>, 2023. Accessed on 2023-05-12.
- [7] Node.js. Node.js - javascript runtime built on chrome's v8 javascript engine. <https://nodejs.org/en/docs/>, 2023. Accessed on 2023-04-21.
- [8] SRS Tech. <https://srs.tech/school-management-software/>, 2023.
- [9] KTU Academics. <https://ktu.edu.in/eu/att/attachments.htm?download=file&id=Bf1Ie64NN1CIbQPgxzMeiqaZaITiPNKhe24NqRV0reE\%3D&announcementId=m3wIwIqGuozHQ00dieVW5qK0b3nLceuS9z0dUcOhnN4\%3D&fileName=ACTIVITYPOINTS.pdf>, n.d.

# Index

Behavioural Feasibility, 6  
Black Box Testing, 28  
  
Detailed Design, 11, 18, 20, 21  
  
Economic Feasibility, 6  
  
Feasibility Check, 6  
Feasibility Study, 6  
  
Implementation, 23  
Integration Testing, 29  
  
Literature Survey, 4  
  
Postman API Testing, 29  
Proposed System, 4  
  
Requirements Analysis, 6, 8  
Requirements Elicitation, 6, 7  
Requirements Specification, 6, 8  
Requirements Validation, 6, 10  
  
System Design, 11  
  
Technical Feasibility, 6  
Testing, 28  
  
Unit Testing, 28