“ Newsletter Generator ”

Minor Project Report

Submitted in partial fulfilment of the requirements of the subject Minor Project

by

Hiral Shah (TY-B 49)
Jai Shah (TY-B 50)
Sarvesh Soni(TY-B 65)

Supervisor

Prof. Nisha Vanjari

Department of Computer Engineering

K.J. Somaiya Institute of Engineering and Information Technology

Ayurvihar, Sion Mumbai-400022

2021-22

This is to certify that the project entitled ***"NEWSLETTER GENERATOR"*** *is a bona fide work of Hiral Shah, Jai Shah and Sarvesh Soni submitted as mini project in the subject of* ***Minor Project*** *in* ***"Computer Engineering".***

_____

Prof. Nisha Vanjari

(Project Guide)

DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. we also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. we understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Hiral Shah _____

Jai Shah _____

Sarvesh Soni _____

Date:

ACKNOWLEDGEMENT

Before presenting out our seminar work entitled "**NEWSLETTER GENERATOR**" we would like to convey our sincere thanks to many people who guided us throughout the course for this seminar work. First, we would like to express our sincere thanks to our beloved Principal **DR. SURESH K. UKARANDE** for providing various facilities to carry out this report.

We would like to express our sincere thanks to **Prof. Nisha Vanjari** for her guidance, encouragement, co-operation and suggestions given to us at progressing stages of report.

Finally, we would like to thank our **H.O.D. Dr. SARITA AMBADEKAR** and all teaching, non-teaching staff of the college and friends for their moral support rendered during the course of the report work and for their direct and indirect involvement in the completion of our report work, which made our endeavour fruitful.

Hiral Shah

Jai Shah

Sarvesh Soni

ABSTRACT

In this project we are developing a website for a departmental Newsletter Generator. Here the faculties can upload information, statistics, images of the respective fields. We have created an admin panel where the admin can delete, insert and update the information that is to be put in the newsletter.

This system makes it very convenient for faculties to upload all the information that is to be put in the departmental newsletter.

INDEX

6. CONCLUSION

REFERENCES

CHAPTER 1

INTRODUCTION

1.1 Introduction:

Newsletter Generator is a Web based application that is designed to collect all necessary information that is to be put in the departmental newsletter.
This project aims at creating a portal for all faculty members to upload and submit data for various fields in the newsletter. The admin can make the changes in this data and also upload additional data by themselves and finally generate a newsletter.

The basic aim is to provide ease for the faculties to upload all information, statistics and images in one place and the admin can generate newsletter from the portal.

1.2 Problem Introduction:

Design a web-based Newsletter Generator having following objectives:
1. User can upload text-based and image-based achievements of faculty members and students of Computer Department.
2. Admin should be able to access and edit the data according to the requirements.
3. Newsletter should be published in the format of word document after all the changes are made by the admin.

CHAPTER 2

REQUIREMENT SPECIFICATION

2.1 INTRODUCTION:

To be used efficiently, all computer software needs certain hardware components or the other software resources to be present on a computer. These pre-requisites are known as(computer) system requirements and are often used as a guideline as opposed to an absolute rule. Most software defines two sets of system requirements: minimum and recommended. With increasing demand for higher processing power and resources in newer versions of software, system requirements tend to increase over time. Industry analysts suggest that this trend plays a bigger part in driving upgrades to existing computer systems than technological advancements.

2.2 HARDWARE REQUIREMENTS:

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware. A hardware requirements list is often accompanied by a hardware compatibility list (HCL), especially in case of operating systems. An HCL lists tested, compatibility and sometimes incompatible hardware devices for a particular operating system or application. The following sub-sections discuss the various aspects of hardware requirements.

HARDWARE REQUIREMENTS FOR PRESENT PROJECT:

PROCESSOR: Intel Pentium dual core or above.

RAM: 1 GB

HARD DISK: 160 GB

Network Interface.

2.3 SOFTWARE REQUIREMENTS:

Software Requirements deal with defining software resource requirements and pre-requisites that need to be installed on a computer to provide optimal functioning of an application. These requirements or pre-requisites are generally not included in the software installation package and need to be installed separately before the software is installed.

SOFTWARE REQUIREMENTS FOR OUR PROJECT:

OPERATING SYSTEM: Windows XP and above, Ubuntu v12.04 and above.

FRONT END: HTML, CSS, JavaScript, jQuery.

SERVER-SIDE SCRIPT: Django, Python

DATABASE: PostgreSQL

CHAPTER 3

ANALYSIS

3.1 PROPOSED SYSTEM:

Website for Newsletter Generator is created where all the faculty members of department can add details about highlights, achievements, etc. of students and faculties. It is easy and convenient to use this site.

3.2 FEASIBILITY STUDY:

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are:

3.2.1 Economic Feasibility

Users just have to login to the website and upload data. As this web-based application is primarily an academic project and most of the features would be incorporated using freeware so as such there wouldn't be any economic cost associated with it.

3.2.2 Technical Feasibility

The technical feasibility assessment meets with the expected needs of the proposed system. It has evaluated that hardware and software meets the need of the proposed system. The assessment based on the project of online testing consist of an interactive interface between student and teachers reveals the following outline design of system requirements:

->HTML
->CSS
->JavaScript
->jQuery
->Python
->PostgreSQL

To deal with requirements to handle completion of the project we are having strong resource of knowledge over the required technologies among our group members. Furthermore, these technologies are being thought in depth in WT tutorials to overcome any of the difficulties. Also, the technologies required are economically and legally feasible for implementation purpose.

3.2.3 Operational Feasibility

Website for newsletter generation is used to upload content for the newsletter. It is very convenient to use. All the requirements for the newsletter can be found here.

3.3 SOFTWARE SPECIFICATION

HTML:

The HyperText Markup Language, or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

CSS:

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript. CSS is designed to enable the separation of presentation and content, including layout, colours, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file which reduces complexity and repetition in the structural content as well as enabling the .css file to be cached to improve the page load speed between the pages that share the file and its formatting.

JavaScript:

JavaScript, often abbreviated as JS, is a programming language that conforms to the ECMAScript specification. JavaScript is high-level, often just-in-time compiled, and multi-paradigm. It has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions. Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. Over 97% of websites use it client-side for web page behaviour, often incorporating third-party libraries. Most web browsers have a dedicated JavaScript engine to execute the code on the user's device.

Python:

Python is an interpreted high-level general-purpose programming language. Its design philosophy emphasizes code readability with its use of significant indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects. Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented and functional programming. It is often described as a "batteries included" language due to its comprehensive standard library. Python consistently ranks as one of the most popular programming languages.

Django:

Django is a Python-based free and open-source web framework that follows the model–template–views (MTV) architectural pattern. It is maintained by the Django Software Foundation (DSF), an independent organization established in the US as a 501(c)(3) non-profit. Django's primary goal is to ease the creation of complex, database-driven websites. The framework emphasizes reusability and "pluggability" of components, less code, low coupling, rapid development, and the principle of don't repeat yourself. Python is used throughout, even

for settings, files, and data models. Django also provides an optional administrative create, read, update and delete interface that is generated dynamically through introspection and configured via admin models.

PostgreSQL:

PostgreSQL also known as Postgres, is a free and open-source relational database management system (RDBMS) emphasizing extensibility and SQL compliance. It was originally named POSTGRES, referring to its origins as a successor to the Ingres database developed at the University of California, Berkeley. In 1996, the project was renamed to PostgreSQL to reflect its support for SQL. After a review in 2007, the development team decided to keep the name PostgreSQL and the alias Postgres. PostgreSQL features transactions with Atomicity, Consistency, Isolation, Durability (ACID) properties, automatically updatable views, materialized views, triggers, foreign keys, and stored procedures.[17] It is designed to handle a range of workloads, from single machines to data warehouses or Web services with many concurrent users. It is the default database for macOS Server and is also available for Windows, Linux, FreeBSD, and OpenBSD.

CHAPTER 4
IMPLEMENTATION

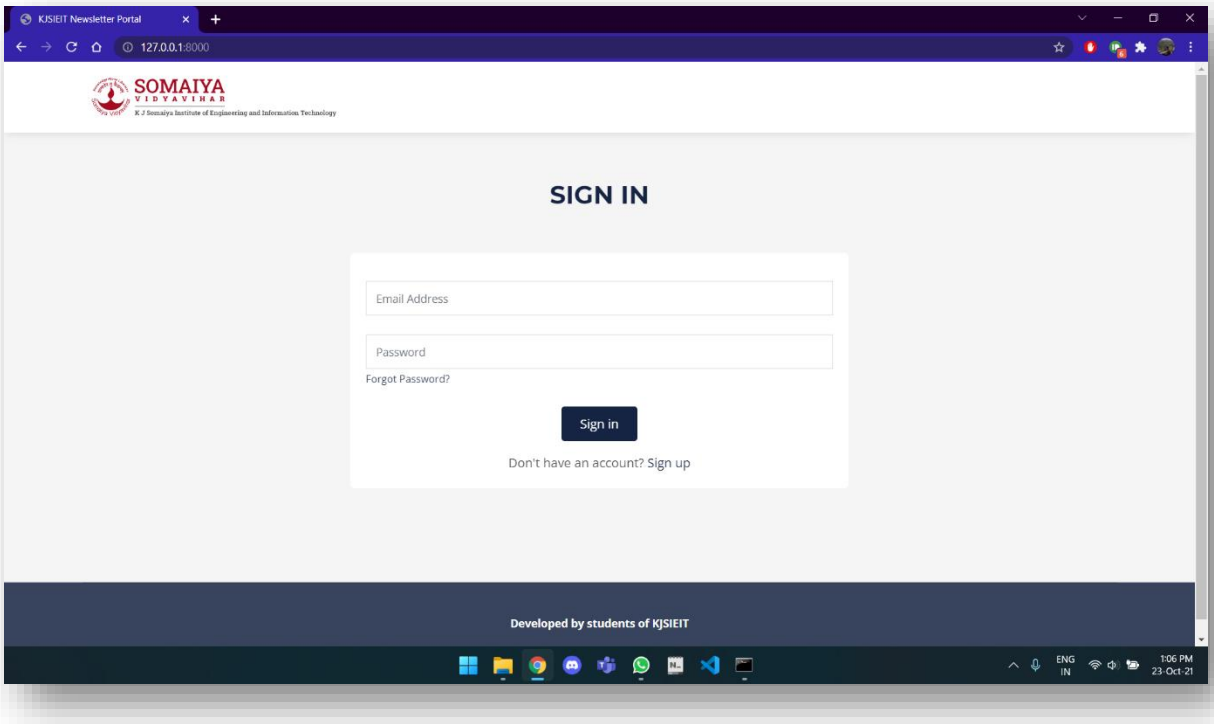
4.1 Introduction:

The implementation stage involves careful planning, investigation of the existing system and its constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

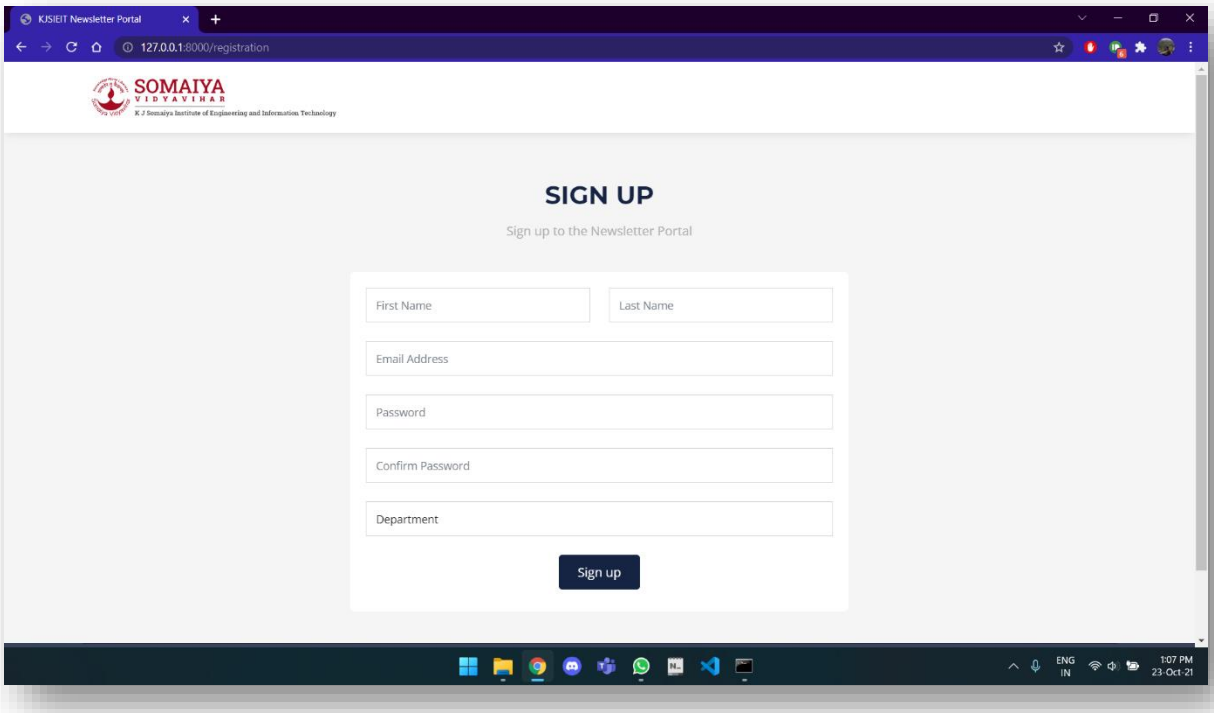
Since our project is the first of its kind, we have designed it with all necessary requirements for now. The website will be deployed for testing. After passing all the test cases, it will be open to all faculties for use. Gradually we'll be adding more features to the website as and when required.

Link to GitHub Repository: https://github.com/jai-cs/Newsletter-Generator

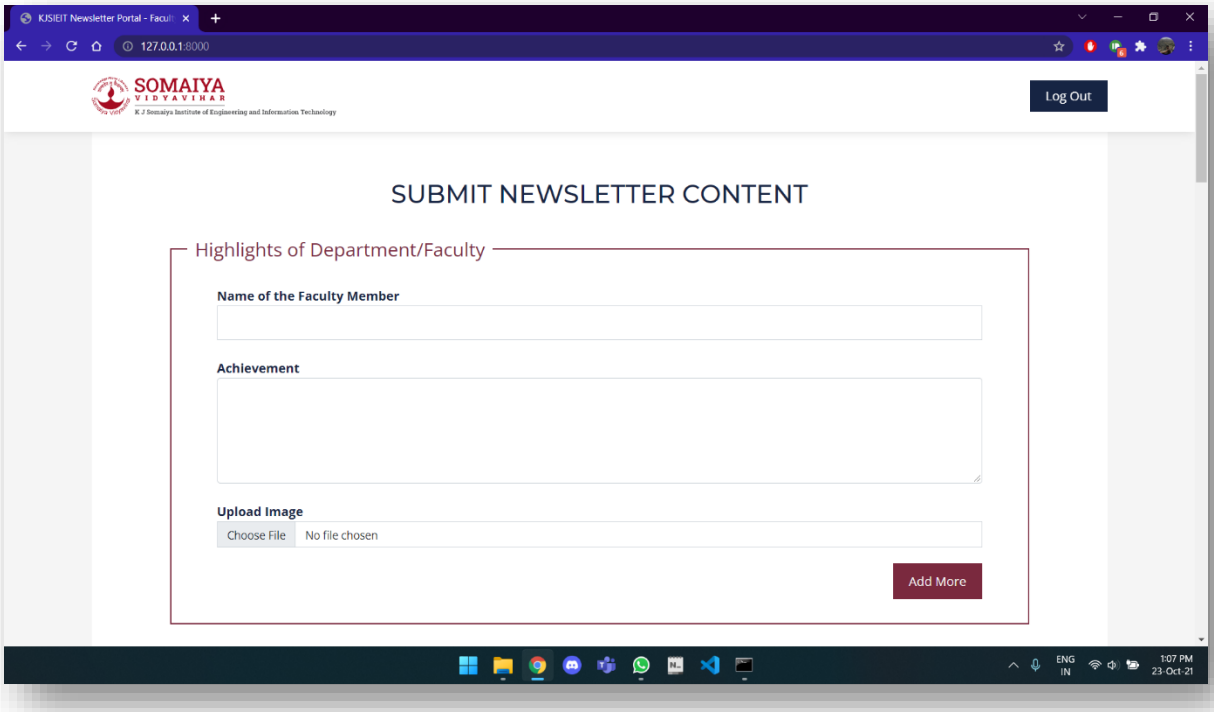
Implementation Screenshots



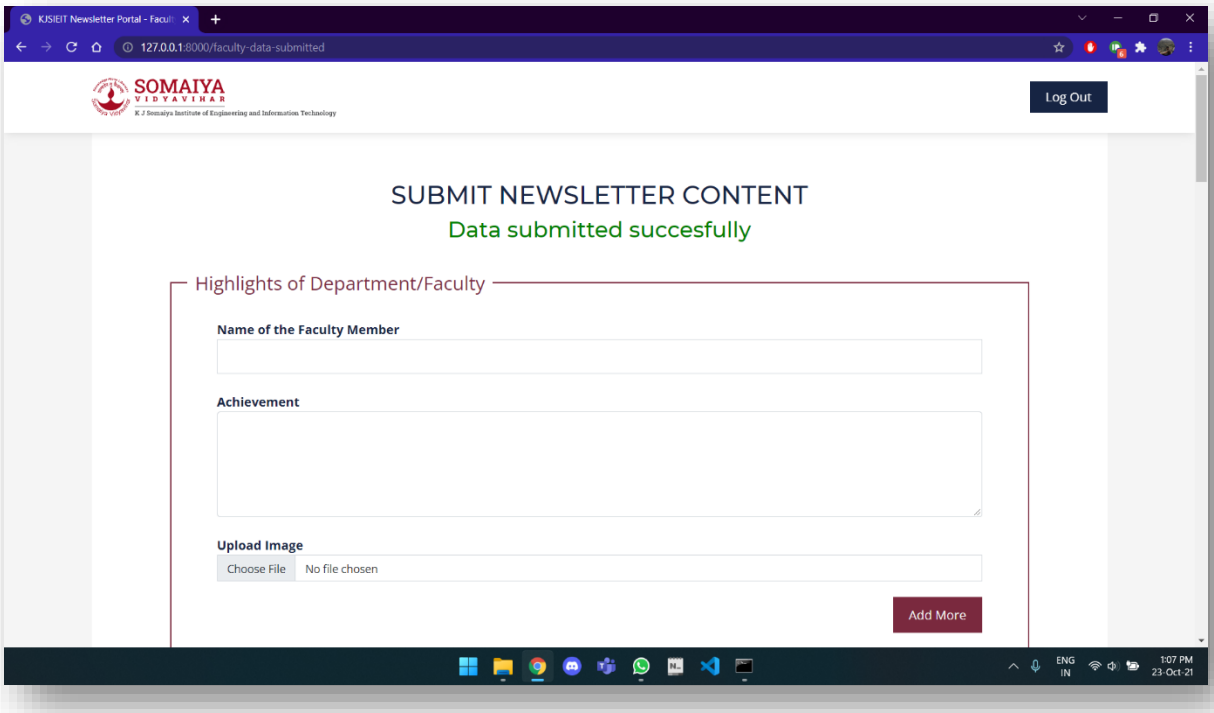
Sign in page



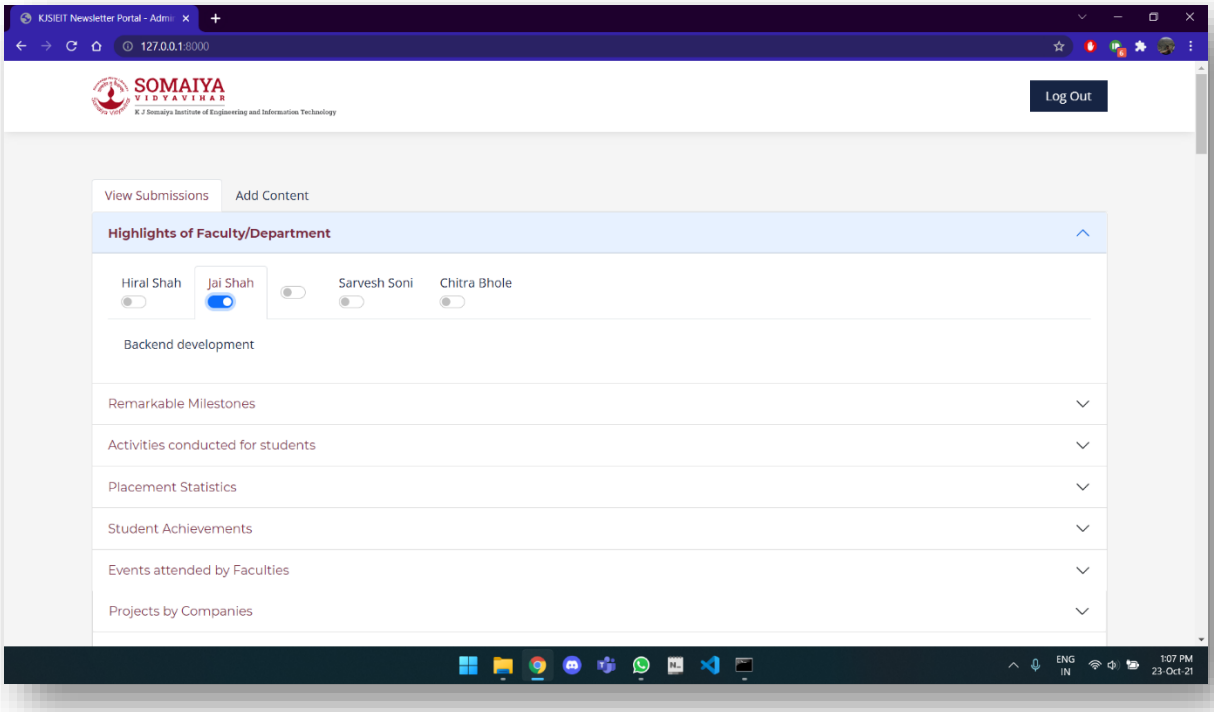
Sign Up page



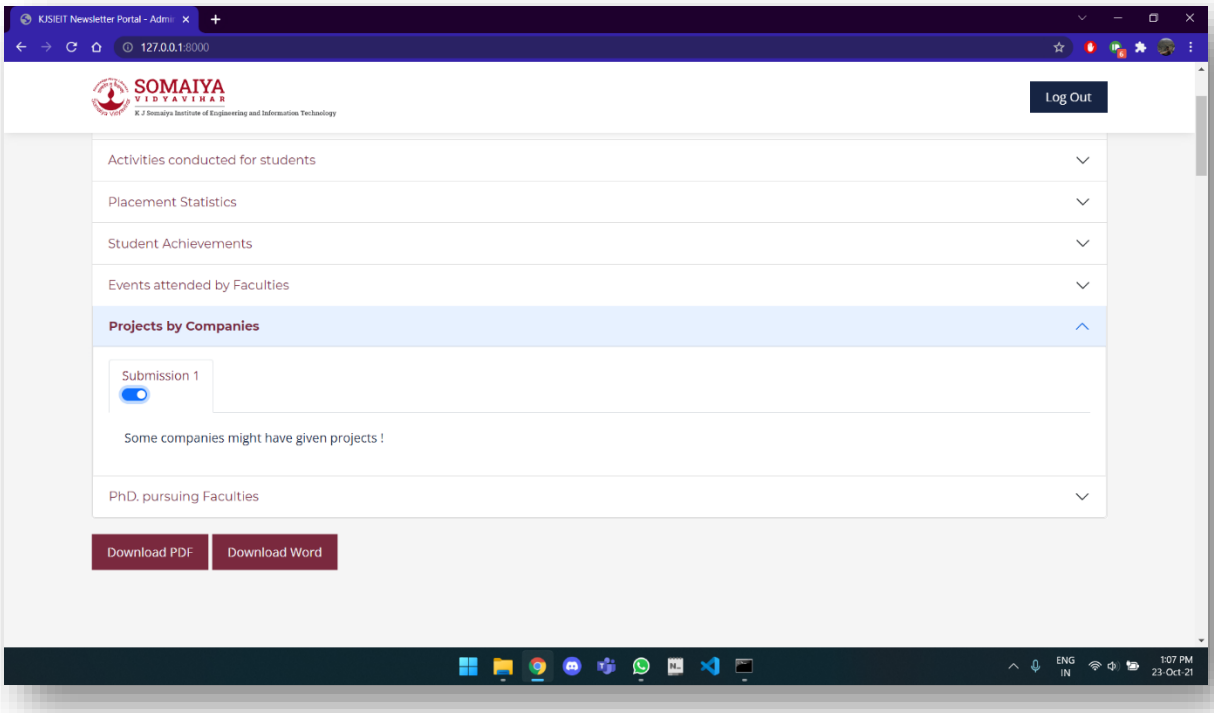
Faculty page



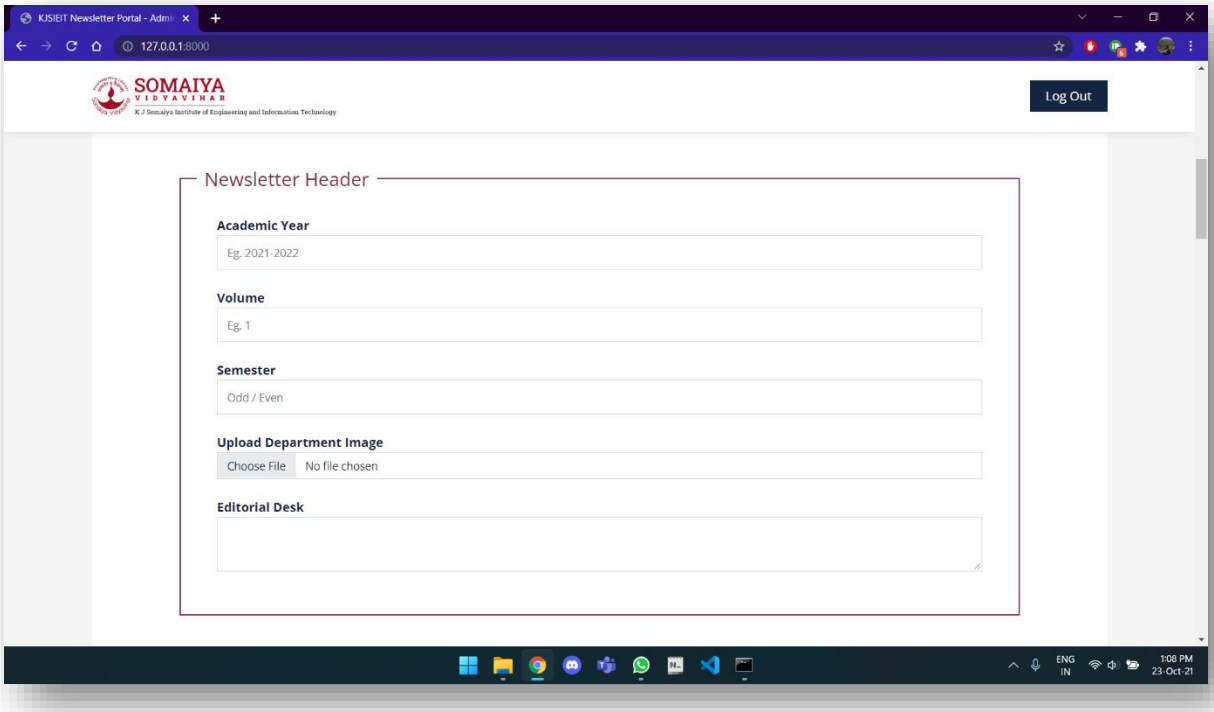
Data Submitted to database



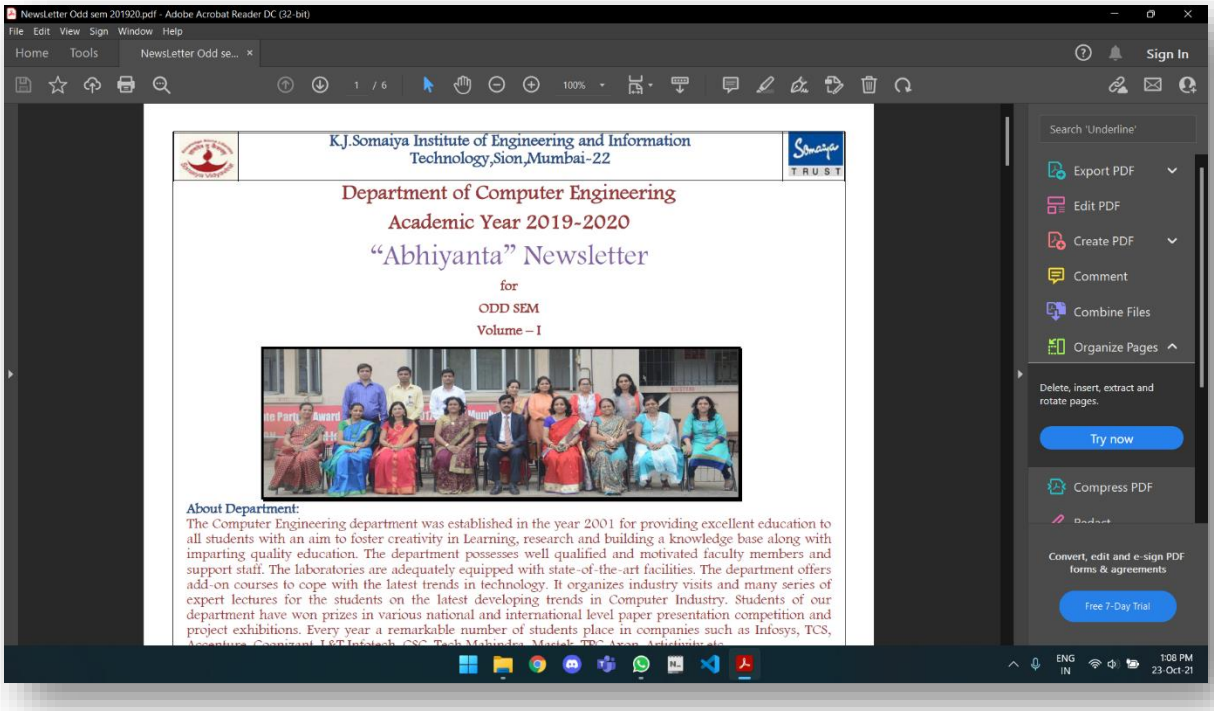
Admin page



View of information for admins

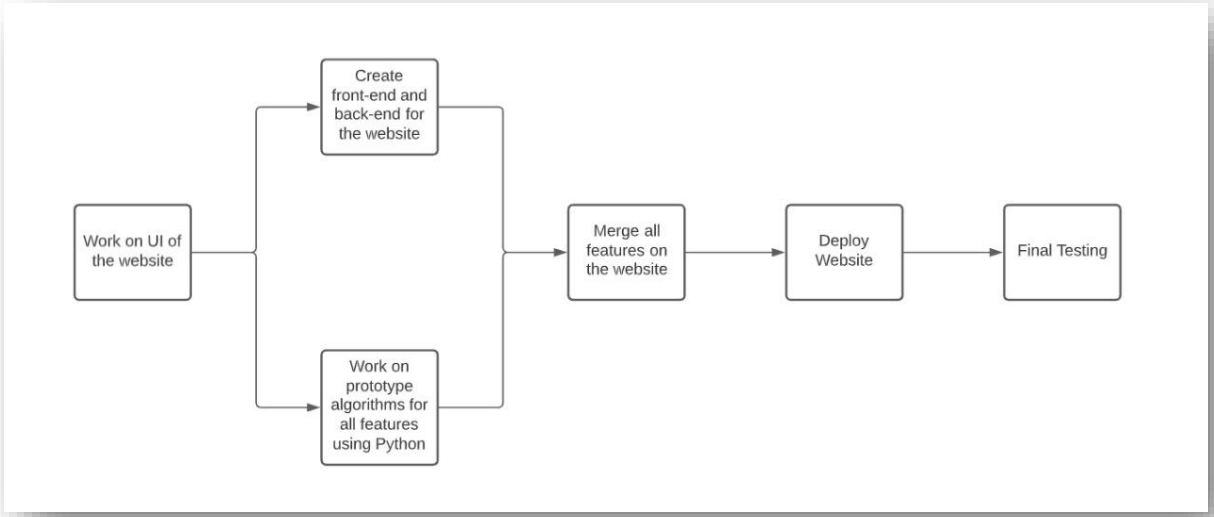


Info filling option for admin



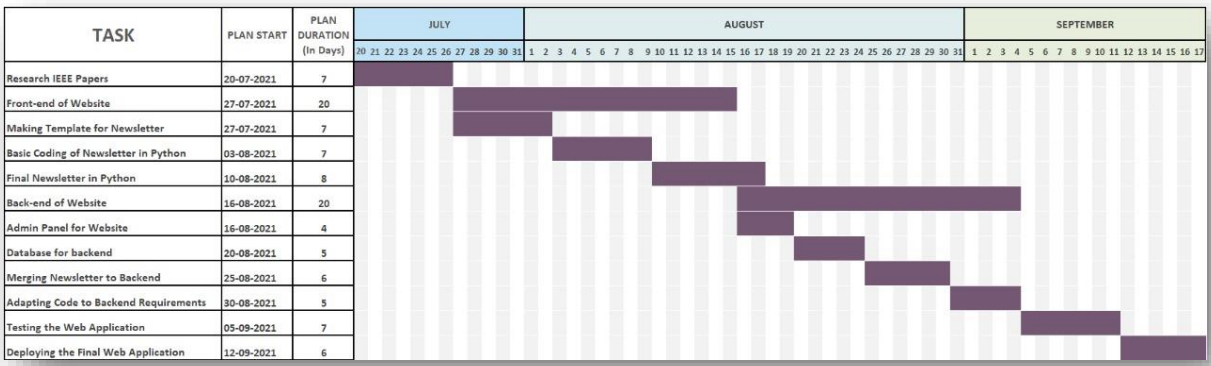
Resultant Output

Software Engineering Analysis

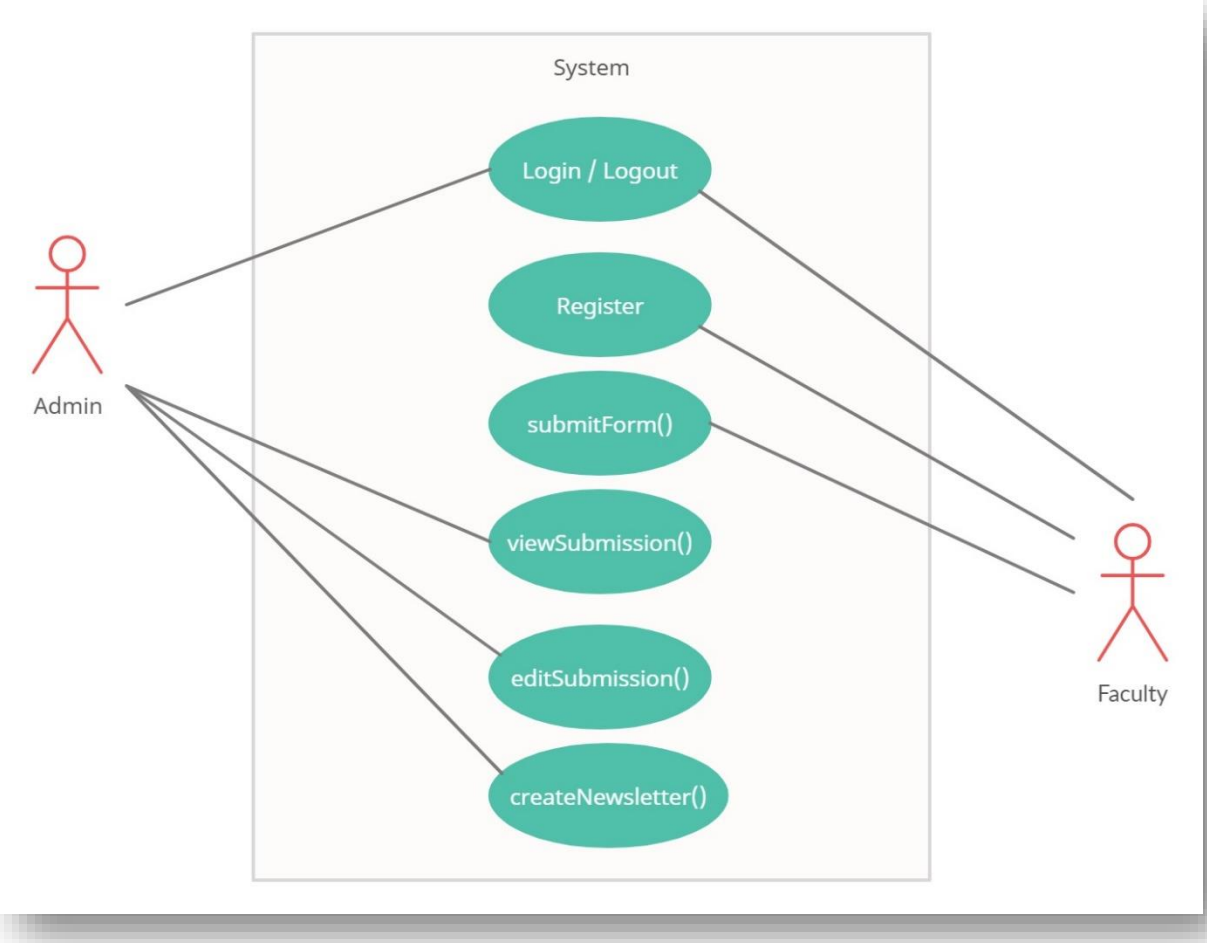


Flowchart

TASK	PLAN START	PLAN DURATION (In Days)
Research IEEE Papers	20-07-2021	7
Front-end of Website	27-07-2021	20
Making Template for Newsletter	27-07-2021	7
Basic Coding of Newsletter in Python	03-08-2021	7
Final Newsletter in Python	10-08-2021	8
Back-end of Website	16-08-2021	20
Admin Panel for Website	16-08-2021	4
Database for backend	20-08-2021	5
Merging Newsletter to Backend	25-08-2021	6
Adapting Code to Backend Requirements	30-08-2021	5
Testing the Web Application	05-09-2021	7
Deploying the Final Web Application	12-09-2021	6



Gantt Chart



UML Diagram

CHAPTER 5

TESTING

5.1 INTRODUCTION TO SYSTEM TESTING:

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

5.2 TYPES OF TESTING:

1. **Unit testing:**
   Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

2. **Integration testing:**
   Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

3. **Functional test:**
   Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

   Functional testing is centered on the following items:
   Valid Input: identified classes of valid input must be accepted.
   Invalid Input: identified classes of invalid input must be rejected.
   Functions: identified functions must be exercised.
   Output: identified classes of application outputs must be exercised.
   Systems/Procedures: interfacing systems or procedures must be invoked.

   Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for

testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

4. **System Test:**
   System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

5. **White Box Testing:**
   White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

6. **Black Box Testing:**
   Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box you cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works.

7. **Unit Testing:**
   Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

8. **Integration Testing:**
   Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

9. **Acceptance Testing:**
   User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

5.3 Testing of Project

Test strategy and approach
Field testing will be performed manually and functional tests will be written in detail.

Test objectives
- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

Test Table:

Sr No.	Test Case	Input	Output
1	Email ID	Invalid Format	OK
		Correct Format	OK
		Empty	OK
2	Password	String	OK
		Empty	OK
3	First Name	String	OK
		Empty	OK
4	Last Name	String	OK
		Empty	OK
5	Image	Uploaded	OK
		Empty	OK

Test Results:

All the test cases mentioned above passed successfully. No defects encountered.

CHAPTER 6

CONCLUSION

We have made a project that aims that generating a newsletter for the department. All the necessary information is collected here. The data can be edited and updated as per the requirements and will publish a newsletter in format of a word document. Thus, newsletter is generated conveniently and efficiently.

CHAPTER 7

REFERENCES

We have referred the following websites:

- https://jqueryvalidation.org/documentation/
- https://getbootstrap.com/docs/5.0/getting-started/introduction/
- https://www.jsdelivr.com/package/npm/documentation-theme-griest