



(An Autonomous Institute Affiliated to Savitribai Phule Pune University)

A Project Report on

Digital Gram Sahaayak - A Virtual System on Government Schemes

Submitted by,

Harshal Patil	(Exam Seat No. B228056)
Yash Dengre	(Exam Seat No. B228063)
Vishal Sule	(Exam Seat No. B228004)
Manoranjan Jena	(Exam Seat No. B224112)

Guided by,

Mrs. Kavitha S.

A Report submitted to MIT Academy of Engineering, Alandi(D), Pune,
An Autonomous Institute Affiliated to Savitribai Phule Pune University
in partial fulfillment of the requirements of

BACHELOR OF TECHNOLOGY in
Computer Engineering

School of Computer Engineering and Technology
MIT Academy of Engineering
(An Autonomous Institute Affiliated to Savitribai Phule Pune University)
Alandi (D), Pune – 412105

(2022–2023)



(An Autonomous Institute Affiliated to Savitribai Phule Pune University)

CERTIFICATE

It is hereby certified that the work which is being presented in the BTECH Project Report entitled "**Digital Gram Sahaayak - A Virtual System on Government Schemes**", in partial fulfillment of the requirements for the award of the Bachelor of Technology in Computer Engineering and submitted to the **School of Computer Engineering and Technology** of MIT Academy of Engineering, Alandi(D), Pune, Affiliated to Savitribai Phule Pune University (SPPU), Pune, is an authentic record of work carried out during Academic Year **2022–2023**, under the supervision of **Mrs. Kavitha S., School of Computer Engineering and Technology**.

Harshal Patil

(Exam Seat No. B228056)

Yash Dengre

(Exam Seat No. B228063)

Vishal Sule

(Exam Seat No. B228004)

Manoranjan Jena

(Exam Seat No. B224112)

**Mrs. Kavitha S.
Project Advisor**

**Mr. Pranav Shriram
Project Coordinator**

**Mrs. Ranjana Badre
Dean SCET**

Director/Dy. Director(AR)

External Examiner

DECLARATION

We the undersigned solemnly declare that the project report is based on our own work carried out during the course of our study under the supervision of **Mrs. Kavitha S..**

We assert the statements made and conclusions drawn are an outcome of our project work. We further certify that

1. The work contained in the report is original and has been done by us under the general supervision of our supervisor.
 2. The work has not been submitted to any other Institution for any other degree/diploma/certificate in this Institute/University or any other Institute/University of India or abroad.
 3. We have followed the guidelines provided by the Institute in writing the report.
 4. Whenever we have used materials (data, theoretical analysis, and text) from other sources, we have given due credit to them in the text of the report and giving their details in the references.

Harshal Patil (Exam Seat No. B228056)

Yash Dengre (Exam Seat No. B228063)

Vishal Sule (Exam Seat No. B228004)

Manoranjan Jena (Exam Seat No. B224112)

Abstract

For the development of Rural India, the Indian government's ministries come up with numerous lucrative programs known as schemes or 'Yojanas.' Thousands of crores of the government's budget have been set aside for these projects to bridge the gap between rural and urban India. Even though considerable improvements are discernible, the target for a developed nation is yet to be achieved. All of the efforts will be futile until these initiatives are made more widely known. The creation of a product or service is never the last step. Similar to how a product needs advertising, villagers must be made aware of the government's profitable initiatives through an easily accessible and identifiable means. Neither the internet nor any broadcasting service now has the capability of effectively disseminating precise information about these yojanas to assure adequate response and participation. This project denotes a virtual system to disintegrate this gap and bring significant changes in the lives of people of Rural India.

Acknowledgment

We want to express our gratitude towards our respected project advisor/guide Mrs. Kavitha S. for her constant encouragement and valuable guidance during the completion of this project work. We also want to express our gratitude to respected School Dean Mrs. Ranjana Badre for her continuous encouragement.

We would be failing in our duty if we do not thank all the other staff and faculty members for their experienced advice and evergreen co-operation.

Harshal Patil

Yash Dengre

Vishal Sule

Manoranjan Jena

Contents

Abstract	iv
Acknowledgement	v
1 Introduction	1
1.1 Background	1
1.2 Motivation	2
1.3 Problem Statement	3
1.4 Project Challenges	3
1.5 Observations	3
1.6 Objectives and Scope	4
2 Literature Review	5
2.1 Research Papers	5
2.2 Findings	7
3 Problem Definition and Scope	8
3.1 Problem statement	8
3.2 Goals and Objectives	8

3.3	Scope and Major Constraints	9
3.4	Hardware and Software Requirements	9
3.5	Expected Outcomes	10
4	System Requirement Specification	12
4.1	Overall Description	12
4.1.1	Product Perspective	12
4.1.2	Product Function	12
4.1.3	User Characteristics	13
4.2	Specific Requirements	13
4.2.1	User Requirements	13
4.2.2	External Interface Requirements	14
4.2.3	Functional Requirements	14
4.2.4	Non-Functional-Requirements	15
4.2.5	Performance Requirement	18
4.3	Project Planning	18
4.4	Modeling	20
4.4.1	System Architecture	20
4.4.2	UML Diagram	21
4.4.2.1	Block Diagram	21
4.4.2.2	Use Case diagram	22
4.4.2.3	Sequence diagram	23
4.4.2.4	Activity diagram	24
4.4.2.5	Class diagram	25

4.4.3	Data Flow Diagram	26
5	Methodology	28
5.1	Approach for implementation	28
5.2	System Workflow	29
5.2.1	Explanation	29
5.3	Algorithms	30
5.4	Features	31
5.5	Technical Specifications	32
6	Implementation	33
6.1	System Implementation	33
6.1.1	Logo	33
6.2	Experiment/Implementation	34
6.3	User Interface	40
6.4	Data Description	49
6.4.1	Database Implementation	50
6.5	Output	55
6.6	Standard Industry practice adopted	62
7	Result Analysis/Performance Evaluation	63
7.1	Result Analysis of Objective 1	63
7.2	Result Analysis of Objective 2	63
7.3	Result Analysis of Objective 3	64
8	Conclusion	65

8.1	Conclusion	65
8.2	Future Scope	65
	Appendices	67
	A Sudo Code	68
A.1	Keyword Search	68
A.2	Send user OTP during newsletter application	69
A.3	Unsubscribe link	70
	B ICMR Conference Acceptance Letter	71
	C Conference Participation	72
C.1	Harshal Patil	72
C.2	Yash Dengre	73
C.3	Vishal Sule	74
C.4	Manoranjan Jena	75
C.5	Mrs. Kavitha S.	76
	D Plagiarism Report of Text	77
	References	82

List of Figures

4.1	Architecture Diagram	20
4.2	Block Diagram	21
4.3	Use Case Diagram	22
4.4	Sequence Diagram	23
4.5	Activity Diagram	24
4.6	Class Diagram	25
4.7	Flow Diagram	26
6.1	Logo	33
6.2	Code 1	34
6.3	Code 2	35
6.4	Code 3	36
6.5	Code 4	37
6.6	Code 5	38
6.7	Code 6	39
6.8	Home	40
6.9	Schemes	41
6.10	Newsletter	42

6.11	Updates	43
6.12	Content	44
6.13	Feedback	45
6.14	Articles	46
6.15	Filter Schemes	47
6.16	Keyword search	48
6.17	Database 1	50
6.18	Database 2	51
6.19	Database 3	52
6.20	Database 4	53
6.21	Database 5	54
6.22	Output 1	55
6.23	Output 2	56
6.24	Output 3	57
6.25	Output 4	58
6.26	Output 5	59
6.27	Output 6	60
6.28	Output 7	61
A.1	Sudo Code 1	68
A.2	Sudo Code 2	69
A.3	Sudo Code 3	70
B.1	Acceptance Letter	71

C.1 ICMR Certificate	72
C.2 ICMR Certificate	73
C.3 ICMR Certificate	74
C.4 ICMR Certificate	75
C.5 ICMR Certificate	76

List of Tables

4.1	Non-Functional Requirements Table 1	15
4.2	Non-Functional Requirements Table 2	16
4.3	Non-Functional Requirements Table 3	17

Chapter 1

Introduction

1.1 Background

The ministries of the Government of India come up with various profitable programs called schemes or ‘Yojanas’ for the betterment of the people of India. These schemes have thousands of crores of budget set by the government to decrease the faulty gap between Rural and Urban India. Though significant changes can be easily determined, the goal for the developed nation is yet not met. Without creation of awareness of these programs along with proper application means, all the efforts decay.

The majority of financial provisions under schemes are allocated to rural areas, hence they are required to be appropriately structured and utilized. For the schemes to be actually beneficial, they should be able to reach the needy. This includes any broadcasting service over the internet or news channels that would promote such programs. After this stage, beneficiaries need to have the knowledge and facilities to actually avail the services. For the same, an easy understandable as well as accessible means must exist that encounters poor literacy, bad internet connectivity, and other such factors that may hamper the end goal.

Though many Self-Help Groups (SHGs) have shown a positive impact on this domain, considerably more efforts have to be made to increase the rate of proper utilization.

Currently, most of the processes in this domain are carried out physically which leads to lots of efforts and loss of time from both the government agents as well as the end-users.

An approach must be taken which provides a one-stop solution where the users can search for relevant information on government schemes as well as an easy way for their application. This paper demonstrates the same along with technicalities, functionalities and users' requirements.

1.2 Motivation

As a united nation, every Indian citizen needs and deserve equal rights and opportunities to improve their as well as their family's quality of life. As of now, around 68 Percent. of the entire Indian population resides in villages without proper health care, education facilities, traveling means, lack of infrastructure, etc. India, being its roots originating from the villages, needs to equally address the rural population and their well-being. To relevel the lives of such people is the main goal of many government ministries. For the same, many schemes are launched on a yearly basis.

According to statistics, between 1995 and 2018, approximately 4,000 000 farmers in India killed themselves. The main reasons are unpaid debts from banks and private lenders, unpredictable climate changes, low health infrastructure leading to unreasonable billing amounts, etc. To prevent such incidents and to promote the welfare of Rural India, the government launches various profitable programs like Aam Aadmi Bima Yojna (AABY) for health-related benefits, Pradhan Mantri Krishi Sinchai Yojna (PMKSY) for the sake of farmers, and many more. Villagers need to know where and how to avail such benefits by applying on government schemes, and what all documents and procedures are required for the same. Moreover, villagers need a means to know whether they are even eligible to apply at a particular scheme or not.

Currently, everything is going over the internet all over the world - whether it is about ordering food, booking traveling tickets, availing educational courses, and

much more. Similarly, a platform is required for the population of Rural India where individuals may quickly apply for government programs with a single tap and receive the all required info about them. Though the prerequisite of technical knowledge is very well required for villagers to get acquainted with this virtual system, small efforts may digitalize all the procedures and will lower human efforts and time.

1.3 Problem Statement

To design a System for people of Rural India and provide them with detailed information on Government Schemes.

1.4 Project Challenges

The main issues that are to be considered in the portal for villagers to be actually able to use such a website is how easily accessible the interface is, whether it is available in their regional languages and how fast is its response time considering remote areas with low internet connectivity.

1.5 Observations

1. We have observed that currently there is not a single platform that would appropriately provide information related to government schemes. A platform having this capability will lead to more awareness of such programs among Rural people.
2. Verification process takes too much physical effort. The government is currently trying to migrate all such procedures only (like passport applications).
3. A vast majority of rural India suffers due to the language barrier (as most portals are available in English).

1.6 Objectives and Scope

1. Supply users with various options to filter various government schemes on the portal.
 2. Prepare functionality for users to validate his/her documents and check foreligibility for schemes.
 3. Provide daily newsletters to keep users updated regarding the programs.
 4. Furnish the portal in the regional language (Hindi) to disintegrate the language barrier.
 5. Layout other functionalities like feedback, etc.
-

Chapter 2

Literature Review

2.1 Research Papers

- 1. Tejas R. Marawar , Swapnil P. Kale and Ketan I. Araspure , "E Governance", in IEEE International Conference on Data Storage and Data Engineering, 2010**

Government authorities worldwide are currently trying to deliver services online as much as possible considering the potential of growing internet facilities. Although failure is easily witnessable in today's scenario, manual services certainly need to be overridden by automated systems available online.

- 2. Samruddhi Khandare , Sushopti Gawade and Varsha Turkar , "Design and development of e farm with S.C.H.E.M.E. E.", in IEEE International Conference on Recent Innovations in Signal Processing and Embedded Systems, 2017**

Growing technology and internet facilities have resulted in information related to agricultural schemes online over various websites and mobile applications. But lower technical exposure and language barriers have hampered the goals of such efforts taken by government authorities as well as various private parties.

- 3. Amruta M. Deshpande and Vidyadheesh J. Pandurangi , "Measuring the effectiveness of Government Schemes using Machine learning algorithms", in IEEE International Conference on Applied and The-**

oretical Computing and Communication Technology, 2018

The Internet has made a huge impact on digital communication and information accessibility.

4. Sabyasachi Mohanty, Dhruba C.Panda , Amit K.Mishra

Sharing and analysis of information related to schemes is highly crucial inorder to increase acceptability.

5. R Raksha, Pradnya Patil, N Swaras, G L RajatV Rekha

227 million people are regular users of the internet in rural areas. The Digital India scheme has helped rural people a lot.

6. Preeti Soni, Kumari Khushboo, Arup Kumar Pal

Information and communication technology has upgraded the delivery of government services to the citizens. Consequently, it has been widely adopted by government agencies.

7. Kalyani Raghunathan, Neha Kumar, Amir Jilani, Alejandra Arrieta, Suman Chakrabarti, Agnes R. Quisumbing Purnima Menon

Various self-help groups (SHGs) have played an important role in awareness of government schemes, but they are considerably limited.

8. Vibhore Vardhan, Vivek Srinivasan, Snigdha Kar, Rajendran Narayanan, Pushpendra Singh, Dipanjan Chakraborty, Siddhartha Asthana, Amarjeet Singh, Aaditeshwar Seth

Beneficiaries' lack of understanding of their rights and obligations under government programs is decreasing participation. People are also having difficulty because of inadequate internet connection and poor literacy. At this time, there has not been any feedback gathering or system modification in response to needs.

2.2 Findings

We searched for various research papers relevant to this field and below provided are the points to be noted:

- The administration is quite interested in the prospect of offering services online.
 - The internet currently provides access to information on plans, but a huge The people of rural India lack education and access to it.
 - Digital communication has got a real boom nowadays and is highly effective for the spread of information.
 - Though portals on the internet have been provided (by the government as well), they are still not easy to operate or go through with less information present and no filtering.
 - Spreading detailed information in this domain is very essential.
 - People across India are not satisfied with the current arrangement of application on government schemes. Hence, they require an easy medium serving the same purpose.
-

Chapter 3

Problem Definition and Scope

3.1 Problem statement

To design a portal for people of Rural India and provide them with detailed information on Government Schemes.

3.2 Goals and Objectives

1. Supply users with various options to filter various government schemes on the portal.
2. Prepare functionality for users to validate his/her documents and check for eligibility for schemes.
3. Provide daily newsletters to keep users updated regarding the programs.
4. Furnish the portal in the regional language (Hindi) to disintegrate the language barrier.
5. Layout other functionalities like feedback, etc.

3.3 Scope and Major Constraints

Our project's implementation has to conquer many limitations as well as restrictions in various aspects. Considering most users are villagers and come from a poor educational background, the portal had to be designed with an interface which is extremely easy to use along with the contents available in regional language as well. These constraints had to be taken into consideration while developing the portal which itself added for more time and efforts.

Secondary Major Constraints:

1. Keeping data secure from any kind of threat. Users' will be providing their documents that shouldn't be lost at any cost.
2. Overall cloud deployment on the cloud will be expensive considering low latency and higher availability.
3. Proposed system should be actually accepted by the government to serve the people of Rural India.

3.4 Hardware and Software Requirements

Hardware Requirement

1. Computer: 4 GHz minimum, multi-core processor
2. Memory(RAM): At least 4 GB, preferably higher, and commensurate with concurrent usage
3. Hard Disk: Space At least 10 GB.

Software Requirement

1. Client OS: Windows, Mac, Linux
2. PC Web: Chrome-v89+ , Firefox-v91+ , Edge-v44+.

3. Server software: AWS
 4. HTML, CSS, JavaScript, Bootstrap, MySQL(AWS RDS), Node Js, Express Js.
 5. GitHub.
6. External Libraries:
- Twilio: This module helps in sending messages on WhatsApp on a specific number. Functionality in the project: Used Twilio to send OTP verification messages as well as newsletters.
 - Translate: Helps in converting a word from one language to another. When provided a word as a parameter along with input and output languages, an array of words in the output language is returned.
 - Nodemailer: Used to send mails to an account. In the project, we had used this module to send newsletters to the users' who had availed this functionality.
 - Ejs: This module is required for templating in NodeJS. To send the data from the nodejs index file over to be shown in the html after compute, this module is required.
 - Spellchecker: Gives corrections for spelling errors. Will be helpful in keyword searching if user inputs a word with incorrect spelling.

3.5 Expected Outcomes

According to the scope decided, we were expected to make a portal that would provide the facility to showcase various government schemes launched by the ministries of the government using various algorithms. Moreover, the portal had to be provided in the regional language as well along with various additional functionalities like newsletters, feedback, etc. The portal had to be designed considering low internet connectivity and higher technological and educational gaps. Outcomes or the result can be seen in the images provided.

As per the goals set, our portal must be able to successfully surpass major constraints and deliver the services in the domain of government schemes easily and efficiently which consider all kind of necessary factors like highly accessible interface, low internet connectivity, etc.

Chapter 4

System Requirement Specification

4.1 Overall Description

4.1.1 Product Perspective

As per the rural peoples' product perspective on this portal, their efforts, resources and time will be saved according to the facilities provided in the portal. For the users, this project will be able to tackle most of their problems while applying or trying to gain information related to government schemes. From their viewpoint, this will mark a great change in the domain and will necessarily grow in the future as well.

4.1.2 Product Function

1. Functionality to make searching for government schemes easily using three different ways - based on state and ministry, based on users' credentials and based on keyword search.
2. Functionality to provide daily newsletters.
3. Functionality to take users' feedback to make necessary updates in the future.

4.1.3 User Characteristics

1. User profile: These are the fields of a user profile. Only registered users are eligible for this option. It includes their gender, name, age, etc.
2. Verification Proof: A valid ID proof required
3. User documents for application: User is supposed to upload documents required depending on the scheme during application.
4. Secondary options' acknowledgement or information

4.2 Specific Requirements

4.2.1 User Requirements

These requirements specify what the end users or rural people expects from our project's portal to be able to do. The main 4 points are:

1. Easy and simple to operate: The interface design must be simple and accessible to all features.
2. Quick Response: Response for all client requests must be done quickly surpassing the constraint of low internet connectivity. Highly available and scalable architectures will be used along with a queue structure to achieve the same.
3. Effectively handling operational errors.
4. Customer support: User are provided with the feedback system where they can efficiently interact with the portal's support.

4.2.2 External Interface Requirements

1. Hardware Required: a. Computer: 4 GHz minimum, multi-core processor. b. Memory(RAM): At least 4 GB, preferably higher, and commensurate with concurrent usage c. Hard Disk: Space At least 10 GB.
2. Supported device types: Windows, Mac, Linux.
3. Web: Chrome-v89+ , Firefox-v91+ , Edge-v44+.
4. Server: AWS

4.2.3 Functional Requirements

1. Registration: The user needs to be register using their mobile number.
2. Login: For product use, the user needs to log in.
3. Information and documentation upload: The user needs to upload information along with necessary documents.
4. Documentation: For the document, Verification User needs to upload valid ID.
5. Scheme Search: Users' are provided with three different options to filter different schemes - based on state or ministry, based on keyword search and based on users' credentials.
6. Daily newsletter: Users will be provided the facility to get daily updates on mail or WhatsApp regarding government schemes.
7. Feedback: Users can also provide feedback so that necessary updates or patches can be implemented accordingly.

4.2.4 Non-Functional Requirements

Number.	Requirement's	Explanation
1.	Scalability	How easily it is to scale the entire application
2.	Reliability	whether the system is reliable or not i.e. even after any faults, how is it able to handle upcoming requests
3.	Regulatory	how compliant it is according to proposed limitations
4.	Maintainability	how easily can the system be maintained and updated

Table 4.1: Non-Functional Requirements Table 1

Number.	Requirement's	Explanation
1.	Serviceability	describes the quality of services provided by the system
2.	Utility	how useful the proposed system is in accordance with the domain
3.	Security	whether the system is secure to different kinds of threats
4.	Manageability	describes how easy it is to manage the entire system's application and infrastructure

Table 4.2: Non-Functional Requirements Table 2

Number.	Requirement's	Explanation
1.	Data integrity	whether the data is consistent and integrated at different levels
2.	Capacity	describes how many client requests it can handle at a time
3.	Availability	describes how low is the system's downtime
4.	Usability	describes the degree to which it can be used or later future scope to be added
5.	Interoperability	system's ability to use parts or functionalities of another system

Table 4.3: Non-Functional Requirements Table 3

4.2.5 Performance Requirement

1. Reliability: The dependability of the individual components determines the reliability of the program as a whole.
2. Availability: The user should always be able to access the system via a web browser, with the sole exception being any downtime experienced by the server that hosts the system. A substitute page will be shown in the event of a hardware malfunction or database corruption. The administrator should obtain database backups using the My-SQL server and store them in case of hardware failure or database damage.
3. Security:
 - 1. To preserve user privacy, passwords will be stored in the database protected.
 - 2. The system will be shielded against flaws like SQL injection attacks by AWS itself.
4. Maintainability: The database is maintained using My-SQL, and the website is hosted on an AWS server. A re-initialization of the software is advised in the event of a failure.
5. Portability: Because it is web-based, the program ought to work with various platforms. and My-SQL applications are essentially agnostic of the operating system they interact with. Any system running any web browser ought to be able to use the functionality of the program because the end-user portion is completely portable.

4.3 Project Planning

1. After the consideration of the domain, we searched on for what areas proper systems aren't present to tackle the goal or what issues can be conquered.
2. After the selection of the problem statement, we did a survey on what websites or portals were present that tried to solve the problem we decided to cover.

3. We did a functional requirement analysis to make sure all required features along with secondary ones are to be created in a proper manner.
4. Considerations or limitations were gathered according to the technological as well as educational gaps.
5. Technologies were selected based on various factors like low internet connectivity, better user interface, etc.
6. The actual implementation was done in coding different modules and then integrating them back together. For every module, various tests were also performed to check the outputs.
7. Once the final website was developed, it was then later deployed on the cloud again considering facts like high availability, low latency, etc.

4.4 Modeling

4.4.1 System Architecture

- The architecture diagram offered shows a technique for building constraints, interactions, and boundaries between components while also abstracting the general structure of the software system.

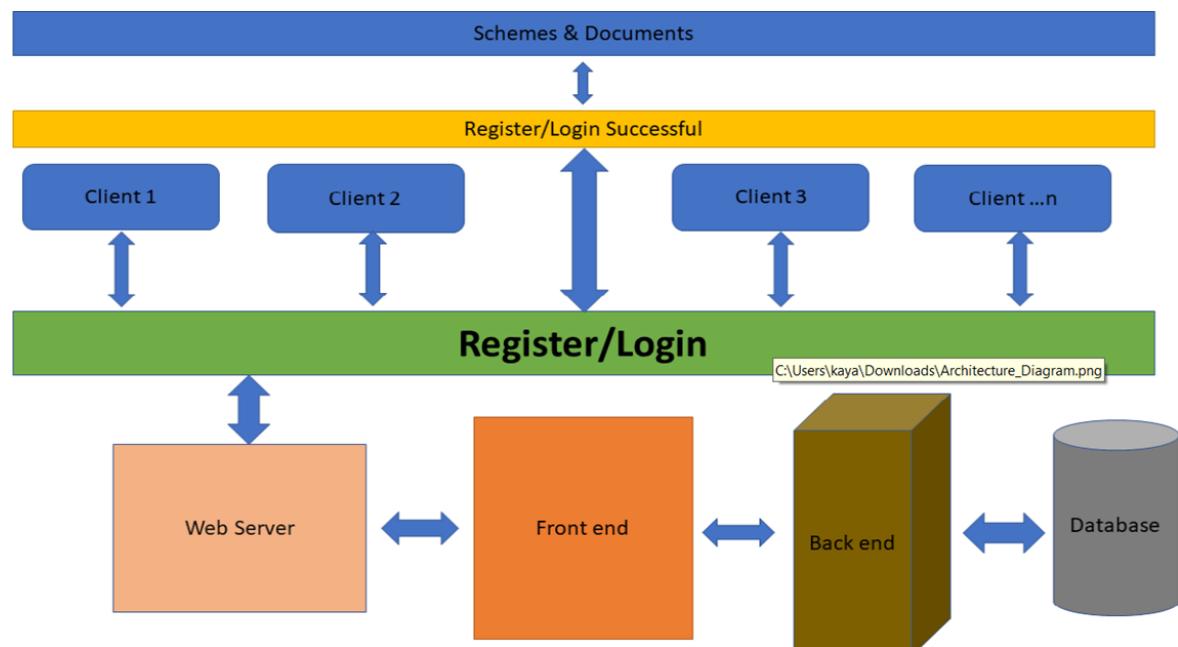


Figure 4.1: Architecture Diagram

4.4.2 UML Diagram

4.4.2.1 Block Diagram

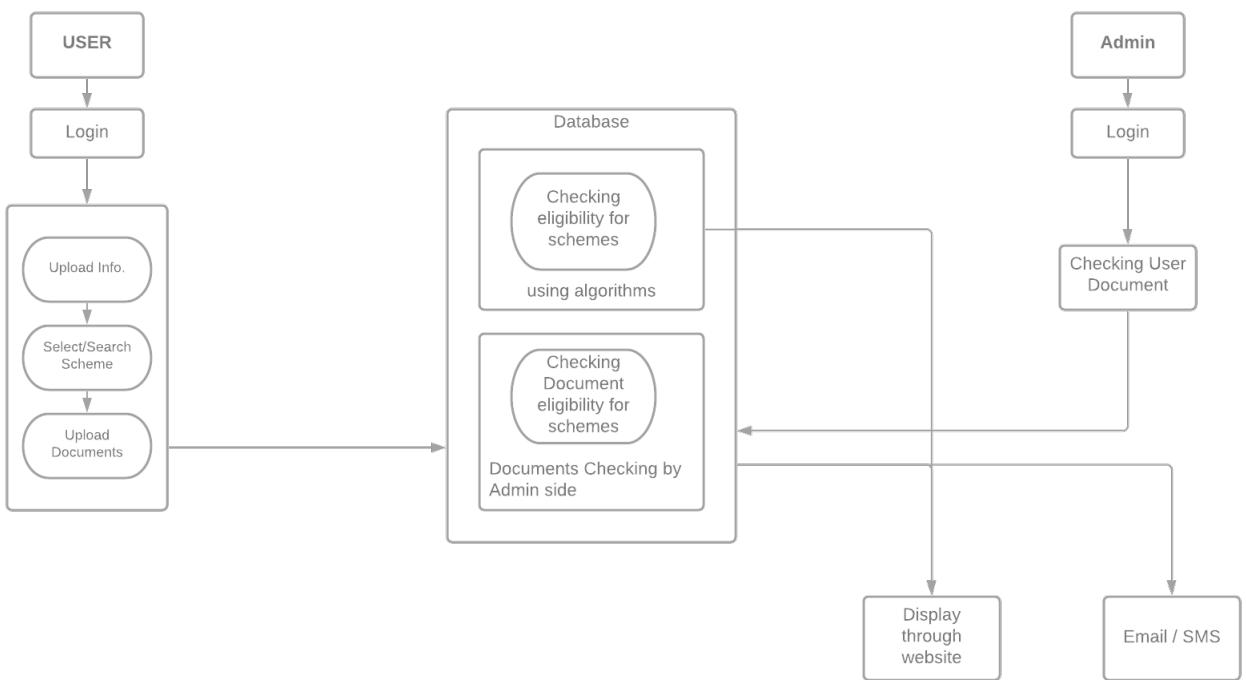


Figure 4.2: Block Diagram

4.4.2.2 Use Case diagram

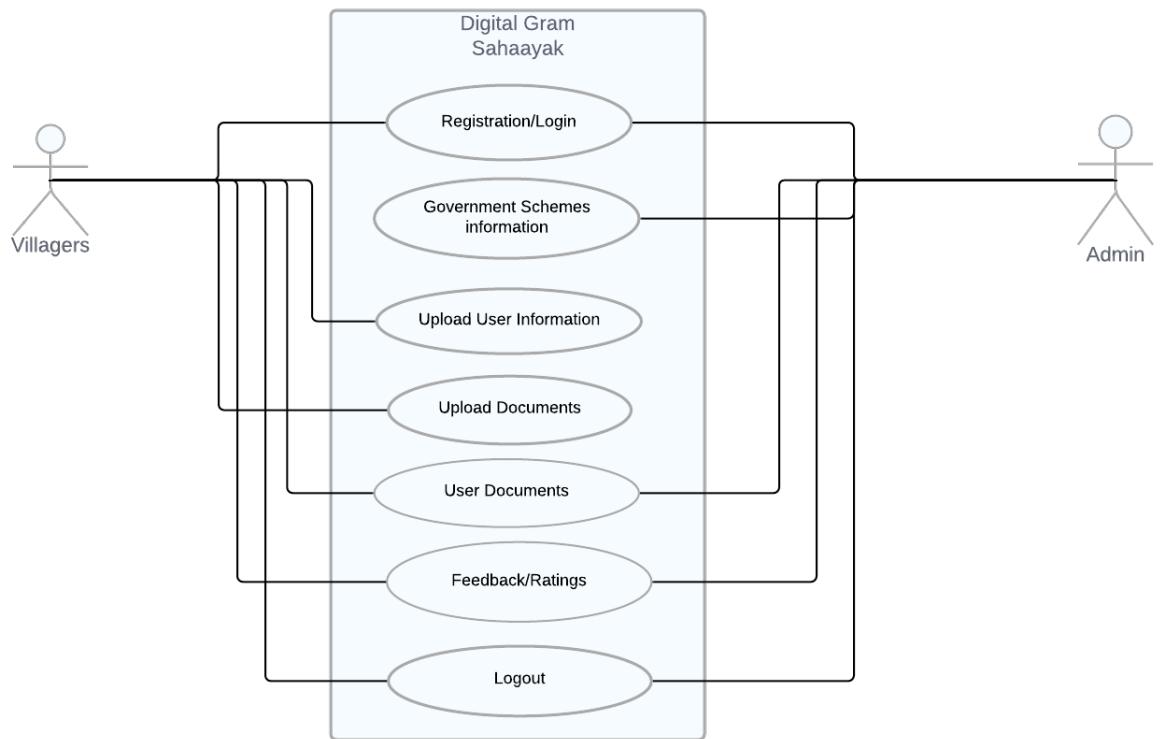


Figure 4.3: Use Case Diagram

4.4.2.3 Sequence diagram

- Sequence diagram has been provided to show object interactions arranged in time sequence.

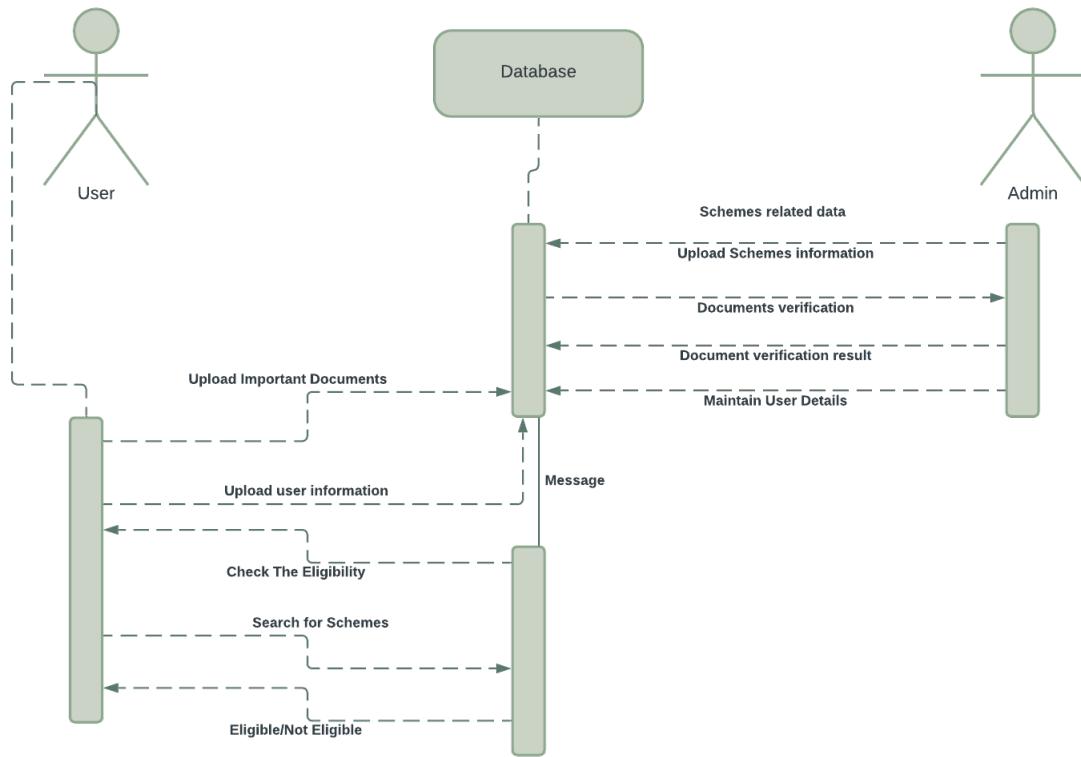


Figure 4.4: Sequence Diagram

4.4.2.4 Activity diagram

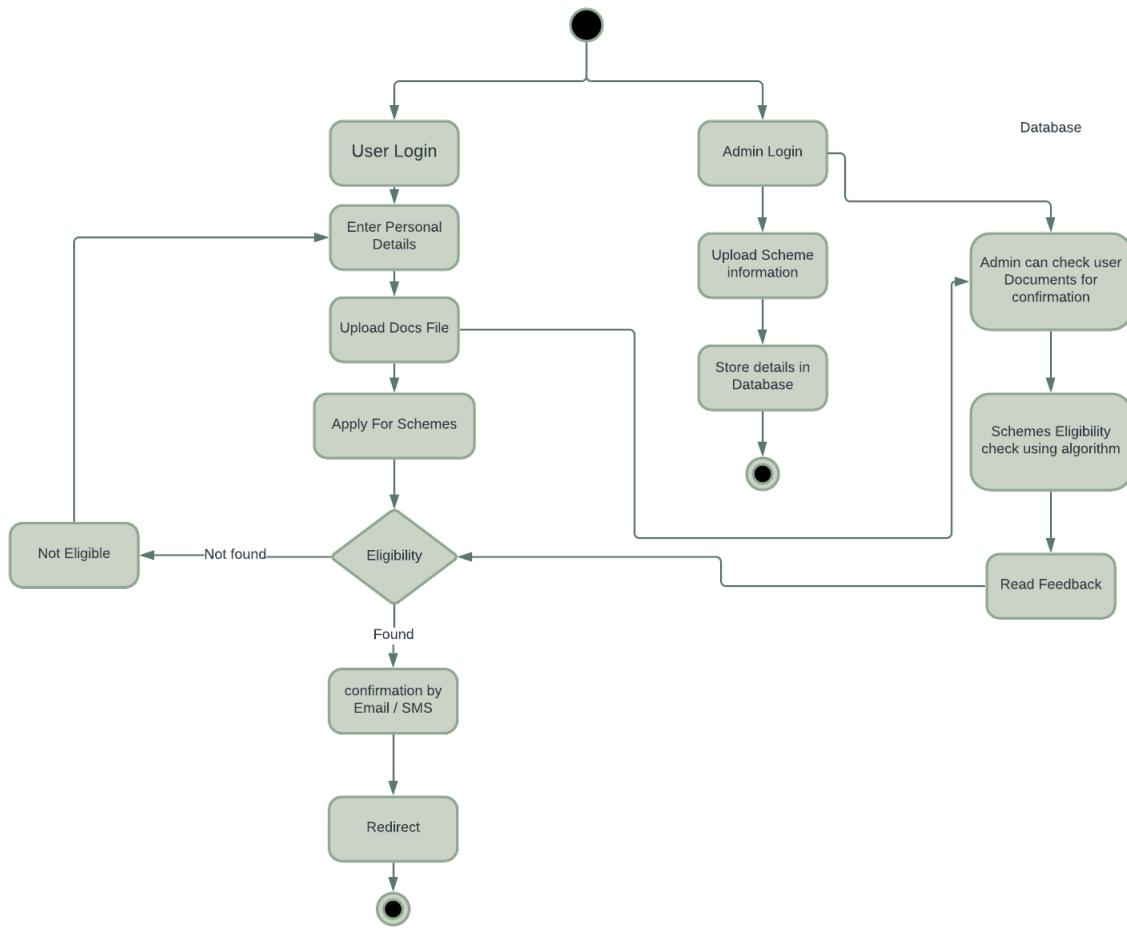


Figure 4.5: Activity Diagram

4.4.2.5 Class diagram

- Class diagram (static) has been provided to propose object-oriented systems for the project.

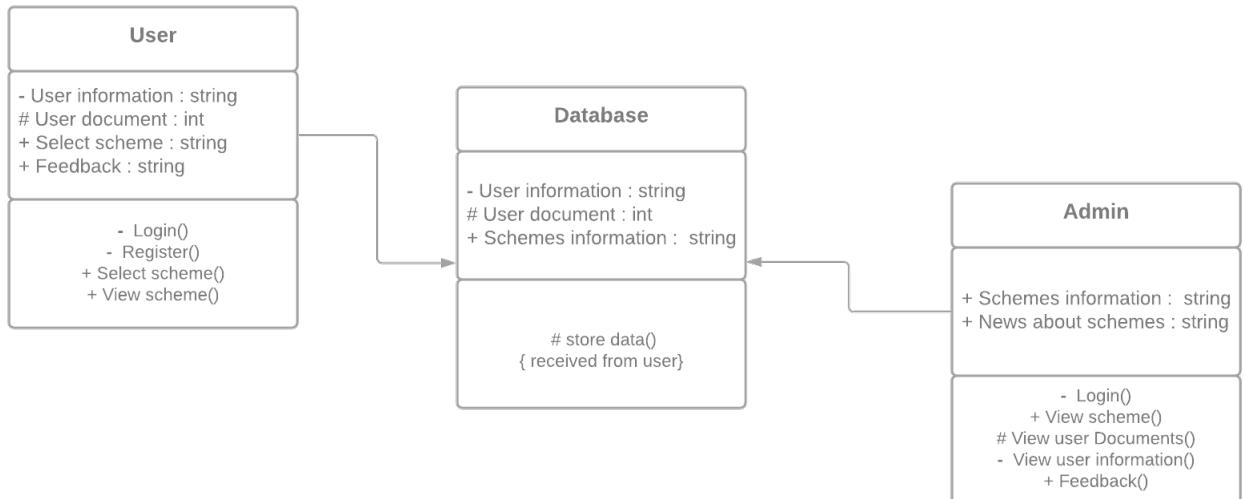


Figure 4.6: Class Diagram

4.4.3 Data Flow Diagram

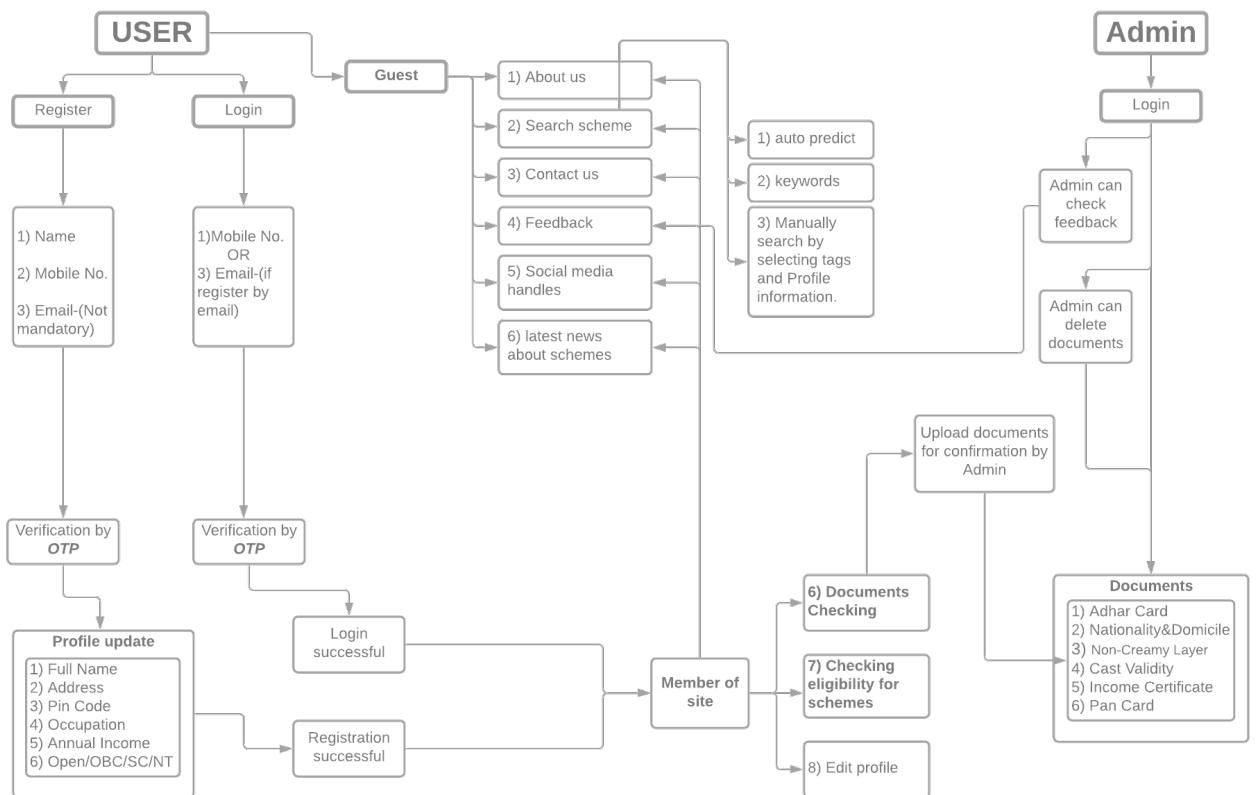


Figure 4.7: Flow Diagram

- Description of the above diagrams:

- Architecture: An architecture diagram has been supplied to provide a method for building constraints, interactions, and boundaries between components while also abstracting the overall structure of the software system.
- Block diagram has been provided to demonstrate main parts or functions present in the model.
- Use Case diagram has been provided to show the dynamic behaviour of the system. Encapsulation of model's functionality can also be witnessed.

- A sequence diagram has been provided to show object interactions arranged in a time sequence. It just shows the sequence of message exchanges between entities or objects that are interacting with each other.
 - Activity diagram is provided to show dynamic behaviour of the system along with the workflow of step-wise activities.
 - Class diagram (static) has been provided to propose object-oriented systems for the project.
 - Flow diagram has been provided to demonstrate the entire flow of all functionalities that a user can interact with in the system.
-

Chapter 5

Methodology

5.1 Approach for implementation

The beginning phase of the project is the conduction of proper research in the current available systems designed for the same purpose. But due to the major factors they lack, a new system proposal is being given in this paper. Two things are to be considered parallelly - the technical side as well as the functionalities directly relevant to the user.

Villagers with less prerequisites relevant to the new technical advancements need to get acquainted with the digital age. But, language barriers can be removed by providing the system in regional languages apart from English. Users may prefer to search specific schemes based on several parameters like ministry, state, gender, domain, etc. Several scheme filtration options are hence to be implemented. Alongside, application to scheme should also be made available making the platform a single-stop solution to every aspect of government scheme functionalities. Apart from the implementation of the main functionalities, some secondary facilities like saving documents, marking schemes, etc. are also to be made available. Real time feedback from the users are to be collected and considered for the necessary changes. Research must be conducted on a regular basis to evolve the platform accordingly.

Before beginning with the actual implementation, the current technical as well as educational status of Rural India is to be considered. With low internet range, the

system should be able to work properly without any lag or improper display of the content. To achieve this, technologies that are asynchronous (to load page content parallelly with higher speed) are used along with light weight front-end which itself is made easily accessible. Content needs to be delivered with low latency, hence to achieve the same, appropriate cloud facilities are to be used.

5.2 System Workflow

Features on the system are made easily accessible with a simple lightweight interface. Users may search schemes on the basis of various factors such as location, ministry, state as well as on the basis of keywords entered. Schemes on the basis of user's eligibility can also be filtered. Application on a particular scheme as well as storing documents require the users to be registered on the platform beforehand. User feedback, marking schemes, daily newsletters, and many more secondary features are kept for better user experience as well as for lavishing the end goal of this project.

5.2.1 Explanation

- Auto predict Schemes will be filtered automatically depending on inputs provided by the users like location, income, caste, gender, etc.
- Keyword Users can type keywords in the search bar and schemes will be shown accordingly.
- Manual searching Users can provide tags and ministry or state to filter out schemes.
- Users can check for eligibility as well by providing options depending on the scheme type.
- Users can get their documents like Aadhar cards, caste certificates, etc. verified on the portal itself.
- Villagers will be provided with options to either log in or register.

- To log in, the user either has to input an email id or mobile no. Further, through OTP, verification will be executed and the user will be logged in.
- To register, the user has to enter some input fields that include name, email id or mobile no. Further, through OTP, verification will be executed and the user will be redirected towards filling in other information like occupation, income, caste, gender, etc.
- Users can select language preferences from English or Hindi.
- Users will be provided various options to interact with like providing feedback, searching for schemes, social media handles, news related to schemes, etc.

5.3 Algorithms

1. Rendering content in different languages: Information to be displayed on the portal will be stored in several languages in the database. Depending on user's preference, data will be loaded dynamically or display properties will be manipulated to show content in the particular language (No application like Google Translator will be imported)
2. Keyword scheme filtration : Keywords entered by user will be searched directly in stored languages (with conversion) in the database to show schemes' details. Elastic search may also be used.
3. Generation of Unique ID for user: New ID for user will be generated by incrementing the previous ID by 1. For example, if last generated ID was '1a2z', new ID will be '1a3a'.
4. Deletion of documents: Through PLSQL, a program will be executed that would delete users' documents depending on the time-lapse.
5. Checking for eligibility: For particular schemes, eligibility will be checked on various parameters like income, caste, gender, etc. Conditionals will be used.
6. Verification through OTP: OTP verification will be implemented and will be

applicable only for a small duration and then repeated. Random 4-digit number generation will be executed and the value can be stored in a particular column or variable. contact. Text-To-Speech, a part of Natural Language Processing has been one of the challenging research fields.

7. Login: Villagers will be provided with options to either login or register.
8. Mobile: To log in, the user either has to input an email id or mobile no. Further, through OTP, verification will be executed and the user will be logged in.
9. Manual searching: Users can provide tags and ministry or state to filter out schemes.
10. Feedback: Users will be provided various options to interact with like providing feedback, searching for schemes, social media handles, news related to schemes, etc.

5.4 Features

1. Auto predict: Schemes will be filtered automatically depending on inputs provided by the users like location, income, caste, gender, etc.
2. Keyword Search: Users can type keywords in the search bar and schemes will be shown accordingly.
3. Manual searching: Users can provide tags and ministry or state to filter out schemes.
4. Users can check for eligibility as well by providing options depending on the scheme type.
5. Users can get their documents like Aadhar card of Harshal Patil, caste certificate, etc. verified on the portal itself.

5.5 Technical Specifications

The application system contains four major sections: Front End which is implemented with the help of HTML, CSS, JavaScript and Bootstrap accompanied by the Back End which is established using NodeJS and ExpressJS along with EJS Templating. For the database, MySQL Relational Database is used and the entire infrastructure of the application is deployed on AWS taking in consideration various performance enhancing factors like application load balancing, auto scaling, loose coupling with the help of queues, database replication by clustering, etc.

Chapter 6

Implementation

6.1 System Implementation

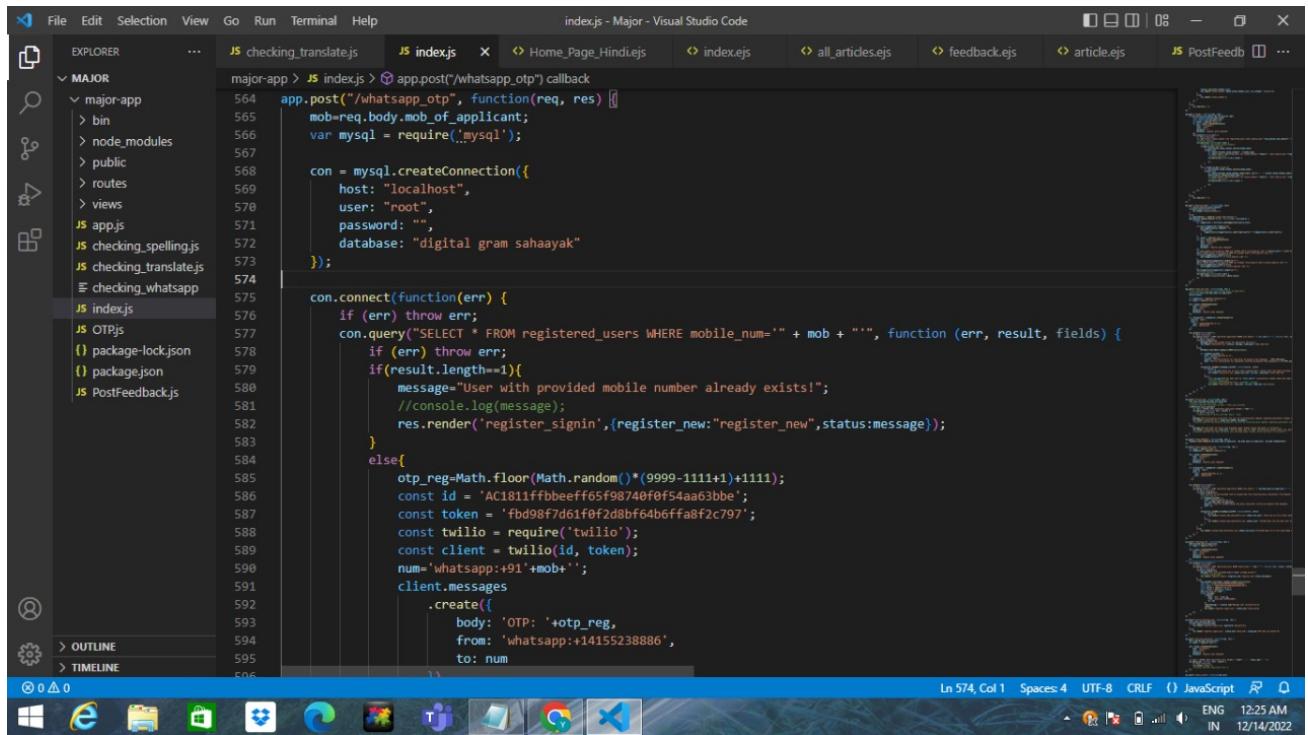
6.1.1 Logo



Figure 6.1: Logo

6.2 Experiment/Implementation

- Code 1



The screenshot shows the Visual Studio Code interface with the file `index.js` open. The code is written in JavaScript and performs several functions:

- It handles a POST request to `/whatsapp_otp` by extracting the mobile number from the request body and connecting to a MySQL database.
- If the mobile number already exists, it sends an OTP via Twilio to the registered phone number.
- If the mobile number does not exist, it generates a random OTP and stores it in the database.

```
index.js - Major - Visual Studio Code
File Edit Selection View Go Run Terminal Help
index.js > app.post('/whatsapp_otp', function(req, res) {
  mob=req.body.mob_of_applicant;
  var mysql = require('mysql');

  con = mysql.createConnection({
    host: "localhost",
    user: "root",
    password: "",
    database: "digital gram sahaayak"
  });

  con.connect(function(err) {
    if (err) throw err;
    con.query("SELECT * FROM registered_users WHERE mobile_num='"+mob+"'", function (err, result, fields) {
      if (err) throw err;
      if(result.length==1){
        message="User with provided mobile number already exists!";
        //console.log(message);
        res.render('register_signin',{register_new:"register_new",status:message});
      }
      else{
        otp_reg=Math.floor(Math.random()*(9999-1111+1)+1111);
        const id = 'AC1811ffbbbeeff65f98740f0f54aa63bbe';
        const token = 'fb9d98f7d61f0f2d8bf64b6ffa8f2c797';
        const twilio = require('twilio');
        const client = twilio(id, token);
        num='whatsapp:+91'+mob+'';
        client.messages
          .create({
            body: 'OTP: '+otp_reg,
            from: 'whatsapp:+14155238886',
            to: num
      
```

Figure 6.2: Code 1

- Code 2

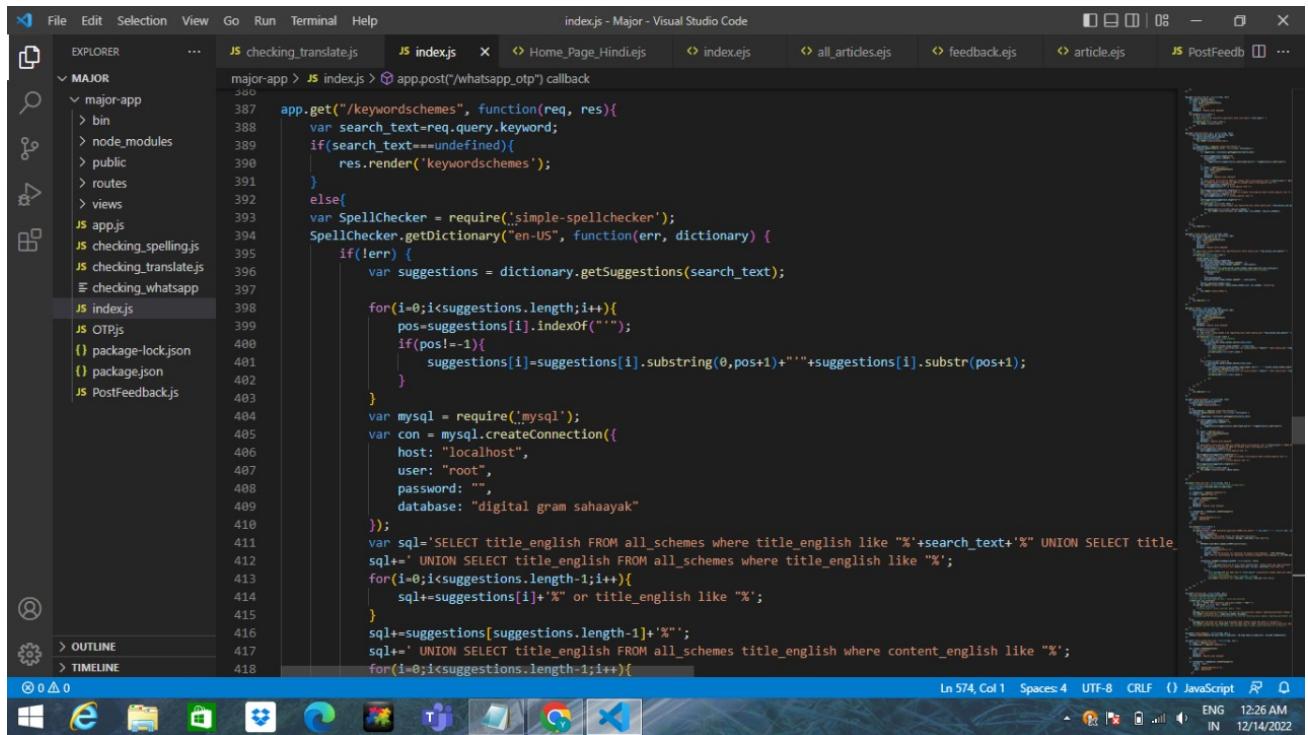
```

File Edit Selection View Go Run Terminal Help
index.js - Major - Visual Studio Code
EXPLORER ... JS checking_translate.js JS index.js < Home_Page_Hindi.ejs < index.ejs < all_articles.ejs < feedback.ejs < article.ejs JS PostFeedb ...
MAJOR
major-app
> bin
> node_modules
> public
> routes
> views
JS app.js
JS checking_spelling.js
JS checking_translate.js
JS checking_whatsapp
JS index.js
JS OTP.js
() package-lock.json
() package.json
JS PostFeedback.js
index.js
585     otp_reg=Math.floor(Math.random()*(9999-1111+1)+1111);
586     const id = 'AC1811ffbbbefff65f98740f0f54aa63bbe';
587     const token = 'fb98f7d61f0f2d8bf64b6ffa8f2c797';
588     const twilio = require('twilio');
589     const client = twilio(id, token);
590     num='whatsapp:+91'+mob+'';
591     client.messages
592       .create({
593         body: 'OTP: '+otp_reg,
594         from: 'whatsapp:+14155238886',
595         to: num
596       })
597       .then(message => console.log("Message sent successfully"))
598       .done();
599     res.render('register_signin.ejs', {check_otp:"check_otp"})
600   }
601   });
602 }
603 );
604
605 app.post("/verify_whatsapp_otp", function(req, res) {
606   user_input_otp=req.body.otp;
607   if(user_input_otp==otp_reg){
608     res.render('register_signin.ejs',{password:"password"});
609   }
610   else{
611     res.render('register_signin.ejs', {check_otp:"check_otp", wrong_otp:"OTP does not match!"})
612   }
613 });
614
615 app.post("/new_user_password", function(req, res) {
616   pass_reg=req.body.new_password;
617   var mysql = require('mysql')...
Ln 574, Col 1 Spaces: 4 UTF-8 CRLF () JavaScript ⌂
Windows Taskbar: File Explorer, Edge, File Manager, Start, Task View, Taskbar Icons, Taskbar Buttons, Taskbar Notifications, Taskbar Language, Taskbar Date and Time
ENGLISH 12:26 AM IN 12/14/2022

```

Figure 6.3: Code 2

- Code 3



```

File Edit Selection View Go Run Terminal Help
index.js - Major - Visual Studio Code
EXPLORER ... JS checking_translate.js JS index.js < Home_Page_Hindi.ejs < index.ejs < all_articles.ejs < feedback.ejs < article.ejs JS PostFeedb ...
MAJOR
major-app
> bin
> node_modules
> public
> routes
> views
JS app.js
JS checking_spelling.js
JS checking_translate.js
JS checking_whatsapp
JS index.js
JS OTP.js
() package-lock.json
() package.json
JS PostFeedback.js
major-app > JS index.js > app.post("/whatsapp_otp") callback
387 app.get("/keywordschemes", function(req, res){
388   var search_text=req.query.keyword;
389   if(search_text==undefined){
390     res.render('keywordschemes');
391   }
392   else{
393     var SpellChecker = require('simple-spellchecker');
394     SpellChecker.getDictionary("en-US", function(err, dictionary) {
395       if(!err) {
396         var suggestions = dictionary.getSuggestions(search_text);
397
398         for(i=0;i<suggestions.length;i++){
399           pos=suggestions[i].indexOf('');
400           if(pos!=-1){
401             suggestions[i]=suggestions[i].substring(0,pos+1)+" "+suggestions[i].substr(pos+1);
402           }
403         }
404         var mysql = require('mysql');
405         var con = mysql.createConnection({
406           host: "localhost",
407           user: "root",
408           password: "",
409           database: "digital gram sahaayak"
410         });
411         var sql='SELECT title_english FROM all_schemes where title_english like "%'+search_text+'%" UNION SELECT title_english
412         sql+=' UNION SELECT title_english FROM all_schemes where title_english like "%";
413         for(i=0;i<suggestions.length-1;i++){
414           sql+=suggestions[i]+"' or title_english like '%";
415         }
416         sql+=suggestions[suggestions.length-1]+"''";
417         sql+=' UNION SELECT title_english FROM all_schemes title_english where content_english like "%';
418         for(i=0;i<suggestions.length-1;i++){

```

Ln 574, Col 1 Spaces: 4 UTF-8 CRLF () JavaScript ENG 12:26 AM IN 12/14/2022

Figure 6.4: Code 3

- Code 4

The screenshot shows a Visual Studio Code interface with the following details:

- File Explorer:** Shows a tree view of files in the "MAJOR" folder, including "index.js", "OTP.js", "package-lock.json", "package.json", and "PostFeedback.js".
- Code Editor:** The main area displays a block of JavaScript code for "index.js". The code handles requests for schemes, connects to a MySQL database, and performs updates based on user input.
- Status Bar:** Shows the line number (Ln 574), column (Col 1), spaces (Spaces: 4), encoding (UTF-8), and file type (JavaScript).
- Taskbar:** Shows standard Windows taskbar icons for Start, File Explorer, Task View, and other applications.

```

index.js - Major - Visual Studio Code
File Edit Selection View Go Run Terminal Help
JS checking_translate.js JS index.js x Home_Page_Hindi.ejs index.ejs all_articles.ejs feedback.ejs article.ejs JS PostFeedb ...
EXPLORER ... MAJOR major-app > JS index.js > app.post("whatsapp_otp") callback
major-app > JS index.js > app.get("/scheme", function(req, res){
  if(req.session.user_mobile && req.session.reg){
    const scheme_name=req.query.name;
    const what_to_do=req.query.fun;
    var mysql = require('mysql');
    var con = mysql.createConnection({
      host: "localhost",
      user: "root",
      password: "",
      database: "digital gram sahaayak"
    });
    con.connect(function(err) {
      if (err) throw err;
      var sql="select saved_schemes from registered_users where mobile_num='"+req.session.user_mobile+"'";
      //console.log(sql);
      con.query(sql,function(err,rows) {
        //console.log(rows[0].saved_schemes);
        if(what_to_do=='save'){
          var pos=rows[0].saved_schemes.search(scheme_name);
          if(pos==1){
            var newstr=rows[0].saved_schemes+', '+scheme_name;
            var sql2="update registered_users set saved_schemes='"+newstr+"' where mobile_num='"+req.session.us
            //console.log(sql2);
            con.query(sql2,function(err,rows2) {
              });
            }
          }else if(what_to_do=='unsave'){
            var pos=rows[0].saved_schemes.search(scheme_name);
            if(pos!=1){
              var newstr=rows[0].saved_schemes.substring(0, pos-1) + '' + rows[0].saved_schemes.substring(pos);
            }
          }
        }
      });
    });
  }
}
  
```

Figure 6.5: Code 4

- Code 5

```
File Edit Selection View Go Run Terminal Help index.js - Major - Visual Studio Code

EXPLORER ... JS checking_translate.js JS index.js < Home_Page_Hindi.ejs < index.ejs < all_articles.ejs < feedback.ejs < article.ejs JS PostFeedb ...

MAJOR
major-app
> bin
> node_modules
> public
> routes
> views
JS app.js
JS checking_spelling.js
JS checking_translate.js
E checking_whatsapp
JS index.js
JS OTP.js
{ package-lock.json
{ package.json
JS PostFeedback.js

299
300 app.get("/saved_schemes", function(req, res){
301   if(req.session.user_mobile && req.session.reg){
302     var mysql = require('mysql');
303     var con = mysql.createConnection({
304       host: "localhost",
305       user: "root",
306       password: "",
307       database: "digital gram sahayak"
308     });
309     var sql="select saved_schemes from registered_users where mobile_num='"+req.session.user_mobile+"'";
310     //console.log(sql);
311     con.query(sql,function(err,rows) {
312       //rows[0].saved_schemes
313       saved_schemes_list=[];
314       if(rows[0].saved_schemes.length!=0){
315         var start_pos=rows[0].saved_schemes.indexOf(',');
316         var end_pos=rows[0].saved_schemes.indexOf(',',start_pos+1);
317         while(end_pos!=1){
318           saved_schemes_list.push(rows[0].saved_schemes.substring(start_pos+1,end_pos));
319           //console.log(rows[0].saved_schemes.substring(start_pos+1,end_pos));
320           if(end_pos==1){
321             break;
322           }
323           start_pos=end_pos;
324           end_pos=rows[0].saved_schemes.indexOf(',',start_pos+1);
325         }
326         console.log(saved_schemes_list);
327         res.render('saved_schemes',{data:saved_schemes_list, all_schemes: "present"});
328       }
329     else{
330       res.render('saved_schemes');
331     }
332   }
333 }

IN 12/14/2022
```

Figure 6.6: Code 5

- Code 6

The screenshot shows a Visual Studio Code interface. The title bar reads "index.js - Major - Visual Studio Code". The left sidebar has an "EXPLORER" tab open, showing a tree view of files and folders under "MAJOR", including "major-app", "bin", "node_modules", "public", "routes", "views", "app.js", "checking_spelling.js", "checking_translate.js", "checking_whatsapp.js", "index.js", "OTP.js", "package-lock.json", "package.json", and "PostFeedback.js". The main code editor tab is "index.js", showing lines 248 to 299 of the code. The code handles a POST request for "whatsapp_otp" and performs various operations like rendering templates, using a spell checker, and connecting to a MySQL database. The status bar at the bottom right shows "Ln 574, Col 1 Spaces: 4 UTF-8 CRLF JavaScript ENG 12:27 AM IN 12/14/2022".

```

index.js - Major - Visual Studio Code
File Edit Selection View Go Run Terminal Help
index.js - Major - Visual Studio Code
EXPLORER ... JS checking_translate.js JS index.js x Home_Page_Hindi.ejs index.ejs all_articles.ejs feedback.ejs article.ejs JS PostFeedb ...
MAJOR
major-app > JS index.js > app.post("/whatsapp_otp") callback
248     if(req.session.user_mobile && req.session.reg){
249         var search_text=req.query.keyword;
250         if(search_text==undefined){
251             res.render('keywordschemes_reg');
252         }
253     else{
254         var SpellChecker = require('simple-spellchecker');
255         SpellChecker.getDictionary("en-US", function(err, dictionary) {
256             if(!err) {
257                 var suggestions = dictionary.getSuggestions(search_text);
258
259                 for(i=0;i<suggestions.length;i++){
260                     pos=suggestions[i].indexOf("");
261                     if(pos!=i){
262                         suggestions[i]=suggestions[i].substr(0,pos+1)+" "+suggestions[i].substr(pos+1);
263                     }
264                 }
265             var mysql = require('mysql');
266             var con = mysql.createConnection({
267                 host: "localhost",
268                 user: "root",
269                 password: "",
270                 database: "digital gram sahaayak"
271             });
272             var sql='SELECT title_english FROM all_schemes where title_english like "%'+search_text+'%" UNION SELECT t';
273             sql+= ' UNION SELECT title_english FROM all_schemes where title_english like "%';
274             for(i=0;i<suggestions.length-1;i++){
275                 sql+=suggestions[i]+"' or title_english like '%";
276             }
277             sql+=suggestions[suggestions.length-1]+"''";
278             sql+= ' UNION SELECT title_english FROM all_schemes title_english where content_english like "%';
279             for(i=0;i<suggestions.length-1;i++){
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
}

```

Figure 6.7: Code 6

6.3 User Interface

- Home Page



Figure 6.8: Home

- Schemes

The screenshot shows the homepage of the Digital Gram Sahaayak website. At the top, there is a navigation bar with links for Home, About Us, Contact Us, Feedback, Register / Sign In, and Language / ભાષા. The main heading 'Explore Schemes' is displayed prominently. Below it, there is a sub-instruction 'Select any of the provided options below to filter specific schemes.' Three filtering options are shown in boxes: 'Filter schemes based on domain, ministry or state ►', 'Keyword Search ►', and 'Filter schemes based on your eligibility ►'. At the bottom of the page, a copyright notice reads 'Copyright © 2021 Digital Gram Sahaayak. All Rights Reserved'.

Figure 6.9: Schemes

- Newsletter

Apply for Newsletter

To receive daily updates from Digital Gram Sahaayak regarding government schemes and programs, kindly fill the provided form below.

Email *

SEND OTP

OTP received on Email *

VERIFY

Figure 6.10: Newsletter

- Updates

The screenshot shows a web-based news update interface. At the top, a teal header bar contains the text "Daily News and Updates". Below this, a sub-header asks "Want to receive daily updates and news regarding government schemes and programmes? [Receive updates on email](#)". A teal button labeled "View all articles ▶" is positioned below the sub-header. The main content area is divided into two sections: "Latest Updates" on the left and "Most Viewed" on the right. The "Latest Updates" section features a news item about the CBI closing a PMO case against Narendra Modi. The "Most Viewed" section features a news item about Narendra Modi addressing the 49th Conference of Governors in Delhi.

Daily News and Updates

Want to receive daily updates and news regarding government schemes and programmes? [Receive updates on email](#)

[View all articles ▶](#)

Latest Updates

[CBI closes PMO case against misuse of Narendra Modi's name to dupe unsuspecting citizens with fraudulent schemes](#)

[AYUSHMAN BHARAT SCHEME UNDERSCORES ISSUES WITH SC'S INTERIM ORDERS IN MARCH](#)

Most Viewed

['Help citizens get maximum benefits': Narendra Modi addresses 49th Conference of Governors in Delhi](#)

Figure 6.11: Updates

- Content

What's on Digital Gram Sahaayak?

Information Regularly Updated <p>All information on this portal gets monitored and updated regularly to prevent any issue.</p>	Present in English/Hindi <p>This portal is designed in both english and hindi languages to degrade the language barrier.</p>	Daily Newsletters <p>Articles related to government schemes and programmes gets published on the portal regularly to keep users updated.</p>
Filter Necessary Information <p>Different options are provided on the portal to filter out specific schemes - based on keyword, ministries or states and one's for which current user is eligible.</p>	Eligibility Checker <p>Eligibility status for a particular scheme can be checked based on various parameters like location, gender, income, etc.</p>	Verify Documents <p>Different documents can be verified on this portal which will aid at the time of application for schemes.</p>

Figure 6.12: Content

- Feedback

Feedback

Have any query or want to give feedback regarding design or content of the portal? Kindly fill the form provided below.

Email *

Name *

Feedback / Query *

Send Feedback

Figure 6.13: Feedback

- Articles

[Home](#)[About Us](#)[Contact Us](#)[Feedback](#)[Register / Sign In](#)Language / [भाषा ▾](#)

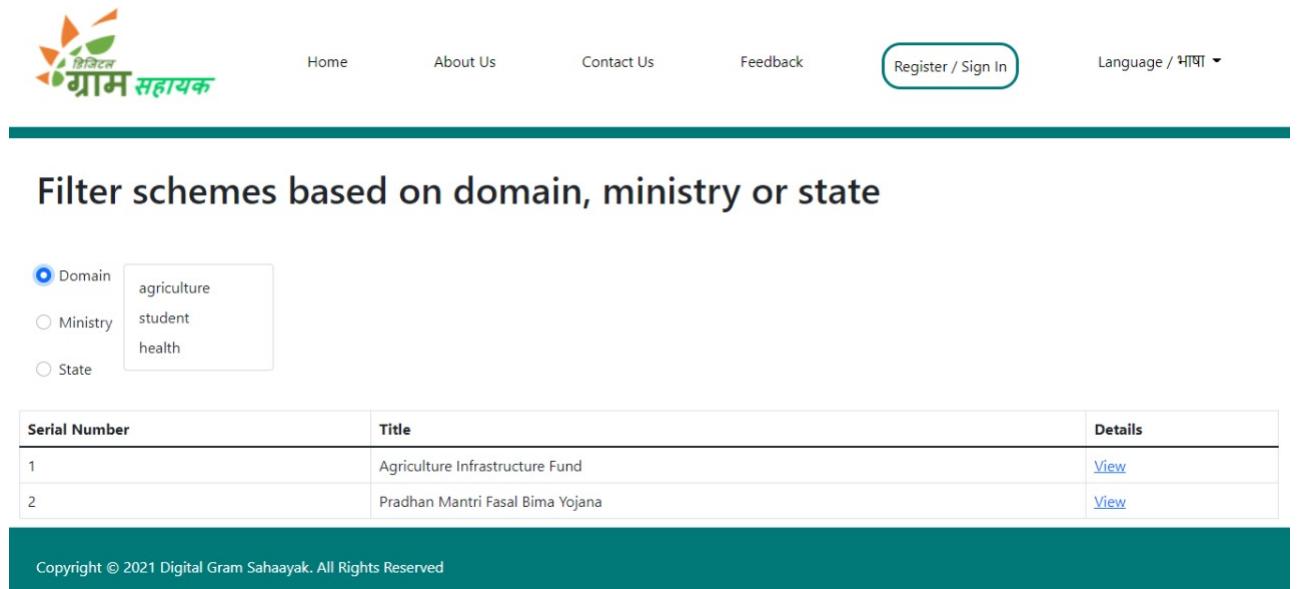
Articles

Serial Number	Title	Details
1	CBI closes PMO case against misuse of Narendra Modi's name to dupe unsuspecting citizens with fraudulent schemes	View
2	AYUSHMAN BHARAT SCHEME UNDERSCORES ISSUES WITH SC'S INTERIM ORDERS IN MARCH	View
3	'Help citizens get maximum benefits': Narendra Modi addresses 49th Conference of Governors in Delhi	View
4	Narendra Modi launches survey to assess public views on govt's performance as NDA completes 4 years	View

Copyright © 2021 Digital Gram Sahaayak. All Rights Reserved

Figure 6.14: Articles

- Filter Schemes



The screenshot shows the Digital Gram Sahaayak website interface. At the top, there is a navigation bar with links for Home, About Us, Contact Us, Feedback, Register / Sign In, and Language / माणा ▾. The logo 'ग्राम सहायक' (Gram Sahaayak) is on the left. Below the navigation bar, a section titled 'Filter schemes based on domain, ministry or state' is displayed. A radio button group allows users to filter by Domain (selected), Ministry, or State. Under 'Domain', options include agriculture, student, and health. A table lists two schemes: 'Agriculture Infrastructure Fund' and 'Pradhan Mantri Fasal Bima Yojana', each with a 'View' link. A copyright notice at the bottom states 'Copyright © 2021 Digital Gram Sahaayak. All Rights Reserved'.

Serial Number	Title	Details
1	Agriculture Infrastructure Fund	View
2	Pradhan Mantri Fasal Bima Yojana	View

Figure 6.15: Filter Schemes

- Keyword search

The screenshot shows the homepage of the Digital Gram Sahaayak website. At the top, there is a navigation bar with links for Home, About Us, Contact Us, Feedback, Register / Sign In (which is highlighted with a green rounded rectangle), and Language / ભાષા ▾. The main content area has a teal header "Keyword Scheme Filtration". Below it is a search bar with placeholder text "Search" and a blue "Search" button with a magnifying glass icon. A table follows, with columns for Serial Number, Title, and Details. The first row shows a serial number of 1 and a title of "Agriculture Infrastructure Fund", with a "View" link in the Details column. At the bottom of the page, a teal footer bar contains the text "Copyright © 2021 Digital Gram Sahaayak. All Rights Reserved".

Serial Number	Title	Details
1	Agriculture Infrastructure Fund	View

Figure 6.16: Keyword search

6.4 Data Description

There are two branches of information to be stored on the portal - users' as well as the portals' itself. Users' information include their name, contact number and password for registration. Additional data includes their documents like ration card, income certificates, verification cards (like PAN Card, license, etc.) for further applications on schemes. Their feedbacks, newsletter application option acknowledgement and other secondary data will also be present in the database. As the data can be stored in a structured manner easily, we have used a relational database - MySQL. Portal itself will store information of schemes in both English as well as in Hindi. Schemes' data like name, region, eligibility criteria, benefits, etc. will also be stored in a relational manner only. Administrators data like ID and password for required access will also be present.

6.4.1 Database Implementation

- AWS 1

The screenshot shows the Amazon RDS Management Console interface. The left sidebar lists various database management options like Dashboard, Databases, Query Editor, etc. The main content area is titled 'Connectivity & security' and contains three columns: 'Endpoint & port', 'Networking', and 'Security'. The 'Endpoint & port' section shows the endpoint as 'digital-gram-sahaayak.c6c3bvsclh4.us-east-1.rds.amazonaws.com' and the port as '3306'. The 'Networking' section shows the availability zone as 'us-east-1c' and the VPC as 'vpc-0b326c894bb847a7a'. The 'Security' section shows the VPC security group as 'default (sg-09f3f6b43bbcccd4ae)' which is marked as 'Active'. Other security details include publicly accessible status, subnet group, certificate authority, and certificate authority date.

Connectivity & security		
Endpoint & port	Networking	Security
Endpoint digital-gram-sahaayak.c6c3bvsclh4.us-east-1.rds.amazonaws.com	Availability Zone us-east-1c	VPC security groups default (sg-09f3f6b43bbcccd4ae) Active
Port 3306	VPC vpc-0b326c894bb847a7a	Publicly accessible No
	Subnet group default-vpc-0b326c894bb847a7a	Certificate authority rds-ca-2019
	Subnets subnet-0538ba4e12260d97c subnet-0aa408aeeff77ec871 subnet-0834619bc1c48c755 subnet-078e49670255ae735 subnet-006475cb9d3729cff subnet-06bd1ae7de63a2644	Certificate authority date August 22, 2024, 22:38 (UTC+05:30)
	Network type IPv4	

Figure 6.17: Database 1

- AWS 2

The screenshot shows the Amazon RDS Management Console interface. The top navigation bar includes tabs for WhatsApp, BTECH_Project_Report_Template, and RDS Management Console. The main title is "digital-gram-sahaayak". The left sidebar has a "Databases" section selected, listing options like Dashboard, Query Editor, Performance insights, Snapshots, Exports in Amazon S3, Automated backups, Reserved instances, Proxies, Subnet groups, Parameter groups, Option groups, Custom engine versions, Events, and Event subscriptions. The main content area displays the "Summary" tab for the database, showing details such as DB identifier (digital-gram-sahaayak), CPU usage (7.87%), Status (Backing-up), Class (db.t2.micro), Role (MySQL Community), Current activity (0 Connections), and Region & AZ (us-east-1c). Below the summary is a navigation bar with tabs: Connectivity & security (selected), Monitoring, Logs & events, Configuration, Maintenance & backups, and Tags. The "Connectivity & security" section shows details for Endpoint & port (Endpoint: digital-gram-...), Networking (Availability Zone: us-east-1c), and Security (VPC security groups: default (sa-...)). The bottom of the screen shows the Windows taskbar with various icons and system status information.

Figure 6.18: Database 2

- AWS 3

The screenshot shows the AWS RDS Management Console interface. On the left, there's a sidebar with various navigation options like Dashboard, Databases, Query Editor, etc. The main content area is titled 'Instance' and displays detailed configuration settings for a specific database instance.

Configuration	Instance class	Storage	Performance Insights
DB instance ID digital-gram-sahaayak	Instance class db.t2.micro	Encryption Not enabled	Performance Insights enabled Turned off
Engine version 8.0.28	vCPU 1	Storage type General Purpose SSD (gp2)	
DB name -	RAM 1 GB	Storage 19 GiB	
License model General Public License	Availability		
Option groups default:mysql-8-0 In sync	Master username admin	Provisioned IOPS -	
Amazon Resource Name (ARN) arn:aws:rds:us-east-1:541668482335:db:digital-gram-sahaayak	IAM DB authentication Not enabled	Storage throughput -	
Resource ID db-VH727RAF5ZP5HQBUIJNBYEH	Multi-AZ No	Storage autoscaling Disabled	
	Secondary Zone -		

At the bottom of the page, there are standard browser controls, a status bar showing the URL https://us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#, and a footer with copyright information and links to Privacy, Terms, and Cookie preferences.

Figure 6.19: Database 3

- Dataset 1

Table	Action	Rows	Type	Collation	Size	Overhead
agriculture		2	InnoDB	utf8mb4_general_ci	16.0 Kib	-
all_schemes		5	InnoDB	utf8mb4_general_ci	16.0 Kib	-
articles		4	InnoDB	utf8mb4_general_ci	80.0 Kib	-
domains		3	InnoDB	utf8mb4_general_ci	16.0 Kib	-
feedback		0	InnoDB	utf8mb4_general_ci	16.0 Kib	-
health		1	InnoDB	utf8mb4_general_ci	16.0 Kib	-
latest_updates		2	InnoDB	utf8mb4_general_ci	32.0 Kib	-
maharashtra		1	InnoDB	utf8mb4_general_ci	16.0 Kib	-
ministries		0	InnoDB	utf8mb4_general_ci	16.0 Kib	-
most_viewed		1	InnoDB	utf8mb4_general_ci	16.0 Kib	-
newsletter_applicants		0	InnoDB	utf8mb4_general_ci	16.0 Kib	-
registered_users		1	InnoDB	utf8mb4_general_ci	16.0 Kib	-
state_names		1	InnoDB	utf8mb4_general_ci	16.0 Kib	-
student		1	InnoDB	utf8mb4_general_ci	16.0 Kib	-
Sum		22	InnoDB	utf8mb4_general_ci	304.0 Kib	0 B

Figure 6.20: Database 4

- Dataset 2

The screenshot shows the phpMyAdmin interface for the 'digital gram sahaayak' database. The 'articles' table is selected. The table has four columns: 'title_english', 'title_hindi', 'content_english', and 'content_hindi'. The data in the table includes news items such as 'CBI closes PMO case against misuse of Narendra Modi...', 'AYUSHMAN BHARAT SCHEME UNDERSCORES ISSUES WITH SC...', and 'Help citizens get maximum benefits'. The content is displayed in both English and Hindi.

title_english	title_hindi	content_english	content_hindi
CBI closes PMO case against misuse of Narendra Modi...	फार्जी योजनाओं से नागरिकों को ठगने के लिए नरेंद्र म...	The move by the Prime Minister's Office (PMO) to i...	प्रधान मंत्री कार्रवाय (पीएमओ) द्वारा प्रधानमंत्री...
AYUSHMAN BHARAT SCHEME UNDERSCORES ISSUES WITH SC...	आयुष्मान भारत योजना मार्च में अनुसूचित जाति के अ...	The Ayushman Bharat National Health Protection Mis...	आयुष्मान भारत राष्ट्रीय स्वास्थ्य सुरक्षा मिशन (यो...
'Help citizens get maximum benefits'. Narendra Modi...	नागरिकों की अधिक से अधिक लाभ प्राप्त करने में मदद...	Addressing the opening session of the 49th Confere...	राष्ट्रपति भवन में राज्यपालों के 49वें सम्मेलन के ...
Narendra Modi launches survey to assess public vie...	एनडीएप के 4 चाल पूरे होने पर नरेंद्र मोदी ने सरकार...	Prime Minister Narendra Modi has launched a survey...	केंद्र सरकार के साथ-साथ राज्य सरकारों और स्थानीय न...

Figure 6.21: Database 5

6.5 Output

- Output 1

The screenshot shows the homepage of the Digital Gram Sahaayak website. At the top, there is a logo with the text 'डिजिटल ग्राम सहायक' and a stylized orange and green leaf icon. Below the logo is a navigation bar with links for 'Home', 'About Us', 'Contact Us', 'Feedback', 'Register / Sign In', and 'Language / ભાષા ▾'. A search bar labeled 'Search' is located below the navigation bar. The main content area has a title 'Keyword Scheme Filtration'. Below the title is a search bar with the placeholder 'Search' and a blue 'Search' button with a magnifying glass icon. A table follows, with columns for 'Serial Number', 'Title', and 'Details'. The first row of the table shows a serial number of 1 and a title of 'Agriculture Infrastructure Fund', with a 'View' link in the Details column. At the bottom of the page, a teal footer bar contains the text 'Copyright © 2021 Digital Gram Sahaayak. All Rights Reserved'.

Serial Number	Title	Details
1	Agriculture Infrastructure Fund	View

Figure 6.22: Output 1

- Output 2

The screenshot shows the homepage of the Digital Gram Sahaayak website. At the top, there is a navigation bar with links for Home, About Us, Contact Us, Feedback, Register / Sign In, and Language / ભાષા. The logo "ગ્રામ સહાયક" is visible on the left. Below the navigation bar, a section titled "Filter schemes based on domain, ministry or state" is displayed. A radio button group allows users to filter by Domain (selected), Ministry, or State. Under the "Domain" filter, options include agriculture, student, and health. A table lists two schemes: "Agriculture Infrastructure Fund" and "Pradhan Mantri Fasal Bima Yojana", each with a "View" link. At the bottom, a teal footer bar contains the copyright notice "Copyright © 2021 Digital Gram Sahaayak. All Rights Reserved".

Serial Number	Title	Details
1	Agriculture Infrastructure Fund	View
2	Pradhan Mantri Fasal Bima Yojana	View

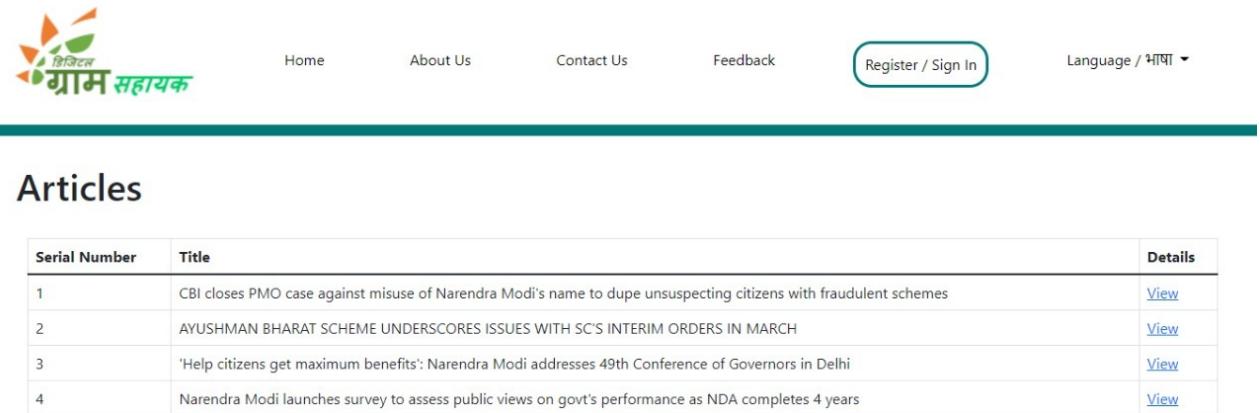
Figure 6.23: Output 2

- Output 3

The screenshot shows the homepage of the Digital Gram Sahaayak website. At the top, there is a navigation bar with links for Home, About Us, Contact Us, Feedback, Register / Sign In, and Language / ભાષા. The main heading "Explore Schemes" is displayed prominently. Below it, a sub-instruction "Select any of the provided options below to filter specific schemes." is followed by three buttons: "Filter schemes based on domain, ministry or state ►", "Keyword Search ►", and "Filter schemes based on your eligibility ►". A copyright notice at the bottom states "Copyright © 2021 Digital Gram Sahaayak. All Rights Reserved".

Figure 6.24: Output 3

- Output 4



The screenshot shows the homepage of the Digital Gram Sahaayak website. At the top, there is a navigation bar with links for Home, About Us, Contact Us, Feedback, Register / Sign In (which is highlighted with a blue border), and Language / ભાષા. Below the navigation bar, there is a logo for 'Digitized Gram Sahaayak' featuring a stylized orange and green design. The main content area is titled 'Articles' and contains a table with four rows of news items. Each row includes a serial number, the article title, and a 'View' link.

Serial Number	Title	Details
1	CBI closes PMO case against misuse of Narendra Modi's name to dupe unsuspecting citizens with fraudulent schemes	View
2	AYUSHMAN BHARAT SCHEME UNDERSCORES ISSUES WITH SC'S INTERIM ORDERS IN MARCH	View
3	'Help citizens get maximum benefits': Narendra Modi addresses 49th Conference of Governors in Delhi	View
4	Narendra Modi launches survey to assess public views on govt's performance as NDA completes 4 years	View

Copyright © 2021 Digital Gram Sahaayak. All Rights Reserved

Figure 6.25: Output 4

- Output 5

Apply for Newsletter

To receive daily updates from Digital Gram Sahaayak regarding government schemes and programs, kindly fill the provided form below.

Email *

SEND OTP

OTP received on Email *

VERIFY

Figure 6.26: Output 5

- Output 6

Feedback

Have any query or want to give feedback regarding design or content of the portal? Kindly fill the form provided below.

Email *

Name *

Feedback / Query *

Send Feedback

Figure 6.27: Output 6

- Output 7

The screenshot displays a section titled "What's on Digital Gram Sahaayak?" which lists six features:

- Information Regularly Updated**: All information on this portal gets monitored and updated regularly to prevent any issue.
- Present in English/Hindi**: This portal is designed in both english and hindi languages to degrade the language barrier.
- Daily Newsletters**: Articles related to government schemes and programmes gets published on the portal regularly to keep users updated.
- Filter Necessary Information**: Different options are provided on the portal to filter out specific schemes - based on keyword, ministries or states and one's for which current user is eligible.
- Eligibility Checker**: Eligibility status for a particular scheme can be checked based on various parameters like location, gender, income, etc.
- Verify Documents**: Different documents can be verified on this portal which will aid at the time of application for schemes.

Figure 6.28: Output 7

6.6 Standard Industry practice adopted

1. Unit Testing: Unit testing is not intended to find bugs. It is an explanation of what the code that is being tested should do. The code that is being evaluated carries out those predicted behaviours.
 2. Continuous Integration: In the technique of continuous integration, team members often integrate their work, typically once per day for each employee, resulting in many integrations per day. To find integration faults as fast as possible, each integration is checked via an automated build (which includes tests).
 3. Refactoring: To address the current known issue, code must be produced. Teams frequently get insight into the issues they are trying to solve, and ongoing code reworking and changes guarantee that the code base is always fulfilling the most recent demands of the company in the most effective way. Your regression testing should be automated to ensure that modifications do not impair already-existing functionality. Unit tests are crucial, for example.
-

Chapter 7

Result Analysis/Performance Evaluation

7.1 Result Analysis of Objective 1

After putting this system to the test, we will have the ideal qualifying standards for programs, enabling rural India to get the most from the support that the Indian government intends to provide. People can take advantage of programs designed to aid India's farmers and veligers thanks to this portal.

7.2 Result Analysis of Objective 2

The application for finding and suggesting eligible government schemes is analyzed and implemented using web technologies. The inclusion of AWS makes it fault resistant and the scalability of the application also rises by a huge degree. By educating the masses regarding the schemes provided by the government we can stop the harassment done by bad actors and corrupt officials and reduce the incidents where bad actors take advantages of innocent people because of the knowlegde gap of the system those people have. The software implementation is done with HTML, CSS, JavaScript and Express and for hosting the servers AWS is used.

7.3 Result Analysis of Objective 3

- From the project's implementation, we can conclude that our approach of providing scheme details' will definitely create a huge impact on Rural India.
 - Proving our portal in English, as well as Hindi, will disintegrate the language barrier. For the future scope, multiple regional languages may also be introduced.
 - We have normalized the data (tables in the database) which will ensure low latency and higher efficiency. Our algorithms for filtering schemes will work on any scale of the database.
-

Chapter 8

Conclusion

8.1 Conclusion

Though the government is constantly exerting efforts for the welfare of Rural India, not a single portal currently exists that would act as a one-stop solution for villagers to find relevant scheme information and apply to them digitally. To supply a remedy for the same, this paper proposes a system that would come up with features that would directly impact the current situation of searching and availing benefits of the profitable programmes launched by the government of India.

8.2 Future Scope

In the future, many more functionalities can be further added in our project to make it more usable and covering more aspects on the domain. Rather than just diverting the users to required pages, application forms can be directly made available on the portal itself. Entire customer helpline can also be created along with centres in districts across India to ensure smooth processing. Rather than having multiple portals by the government concerned with a single scheme, this project's idea can be integrated along with all the schemes available for the villagers. More regional languages like marathi, kannada, etc. can be added to make the portal more useful for users coming from poor educational backgrounds from different regions across India.

Entire application stages can be made online concerning with filling the required information, processing, verification, etc. Moreover, multiple algorithms or softwares can be integrated or implemented to provide personalized recommendations to the users.

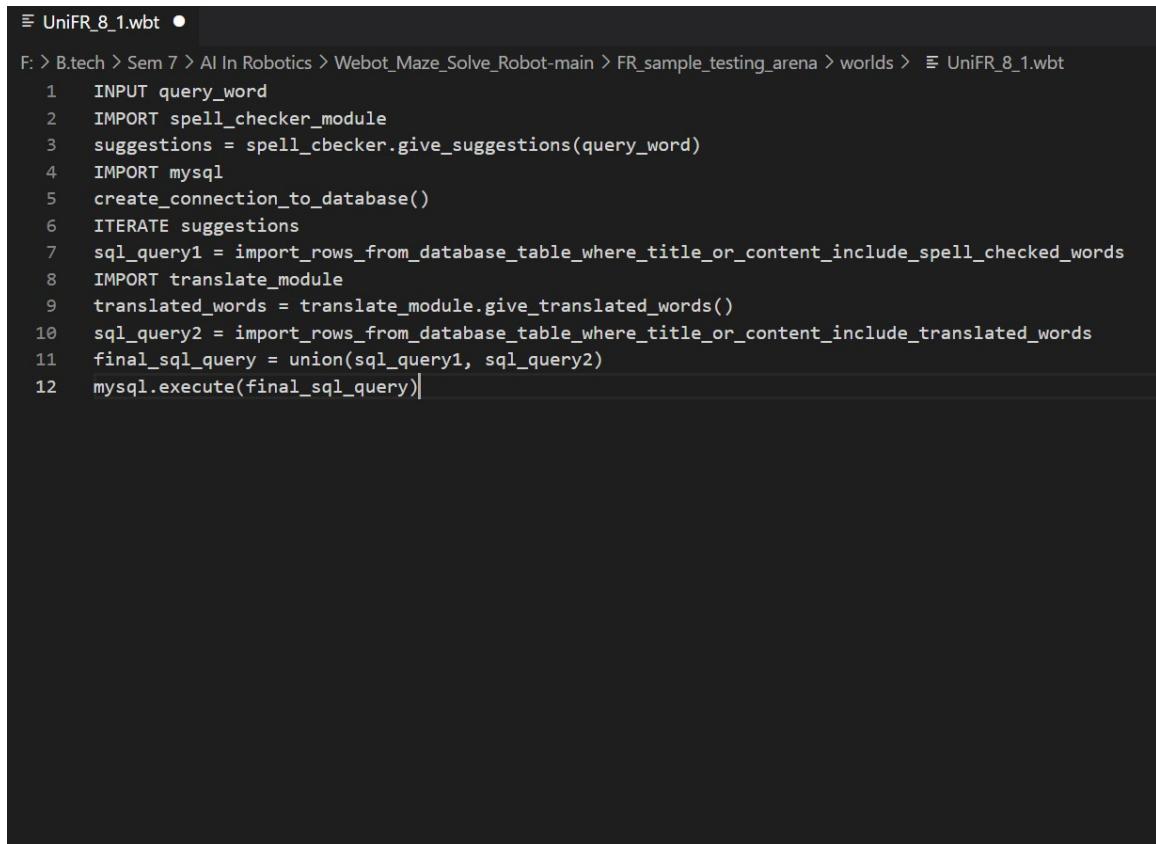
Appendices

Appendix A

Sudo Code

A.1 Keyword Search

- Sudo Code



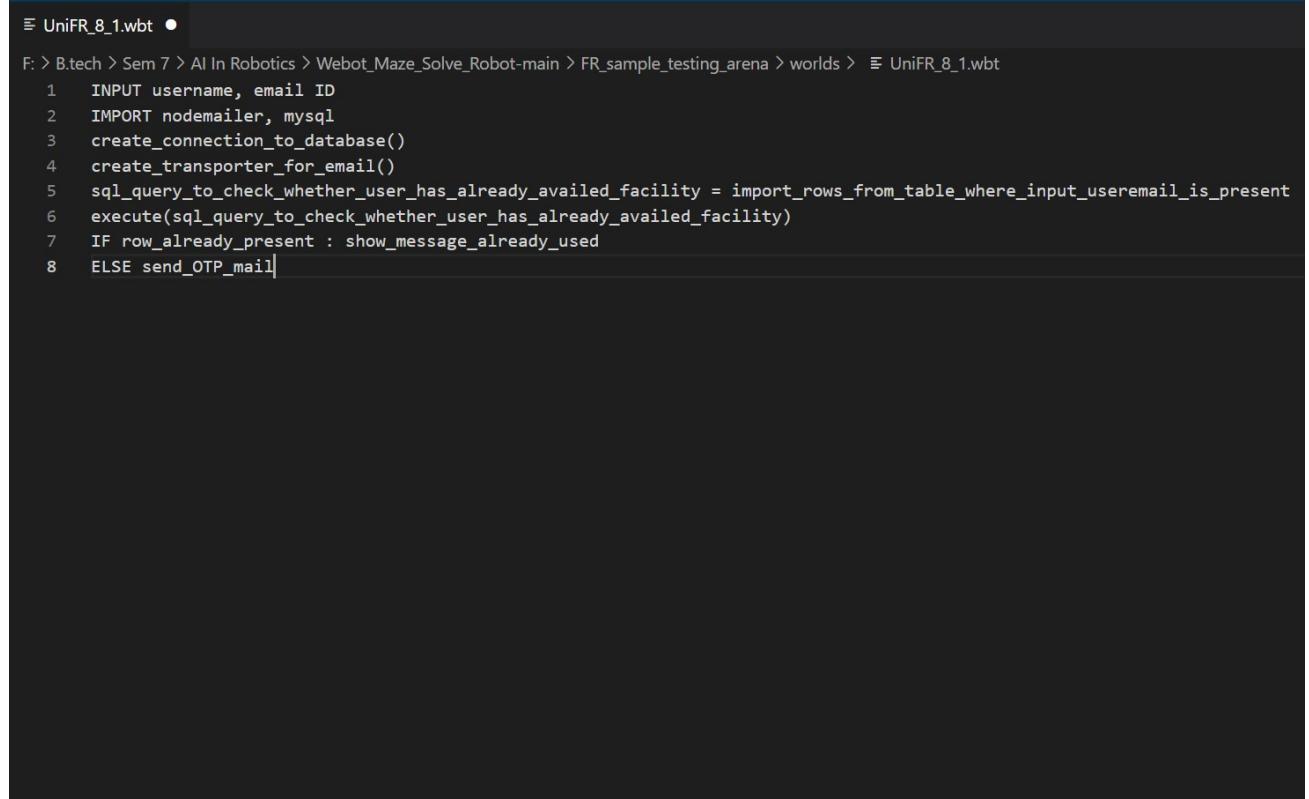
The screenshot shows a code editor window titled "UniFR_8_1.wbt". The code is a Python script for performing a keyword search. It imports modules for input handling, spell checking, MySQL database connection, and translation. It defines a function to get suggestions from a spell checker module, creates a MySQL connection, iterates over suggestions, and performs two SQL queries to find words that include either the checked or translated words. Finally, it executes the final SQL query.

```
UniFR_8_1.wbt •
F: > B.tech > Sem 7 > AI In Robotics > Webot_Maze_Solve_Robot-main > FR_sample_testing_arena > worlds > UniFR_8_1.wbt
 1 INPUT query_word
 2 IMPORT spell_checker_module
 3 suggestions = spell_cbecker.giveSuggestions(query_word)
 4 IMPORT mysql
 5 createConnectionToDatabase()
 6 ITERATE suggestions
 7 sql_query1 = import_rows_from_database_table_where_title_or_content_include_spell_checked_words
 8 IMPORT translate_module
 9 translated_words = translate_module.giveTranslatedWords()
10 sql_query2 = import_rows_from_database_table_where_title_or_content_include_translated_words
11 final_sql_query = union(sql_query1, sql_query2)
12 mysql.execute(final_sql_query)|
```

Figure A.1: Sudo Code 1

A.2 Send user OTP during newsletter application

- Sudo Code



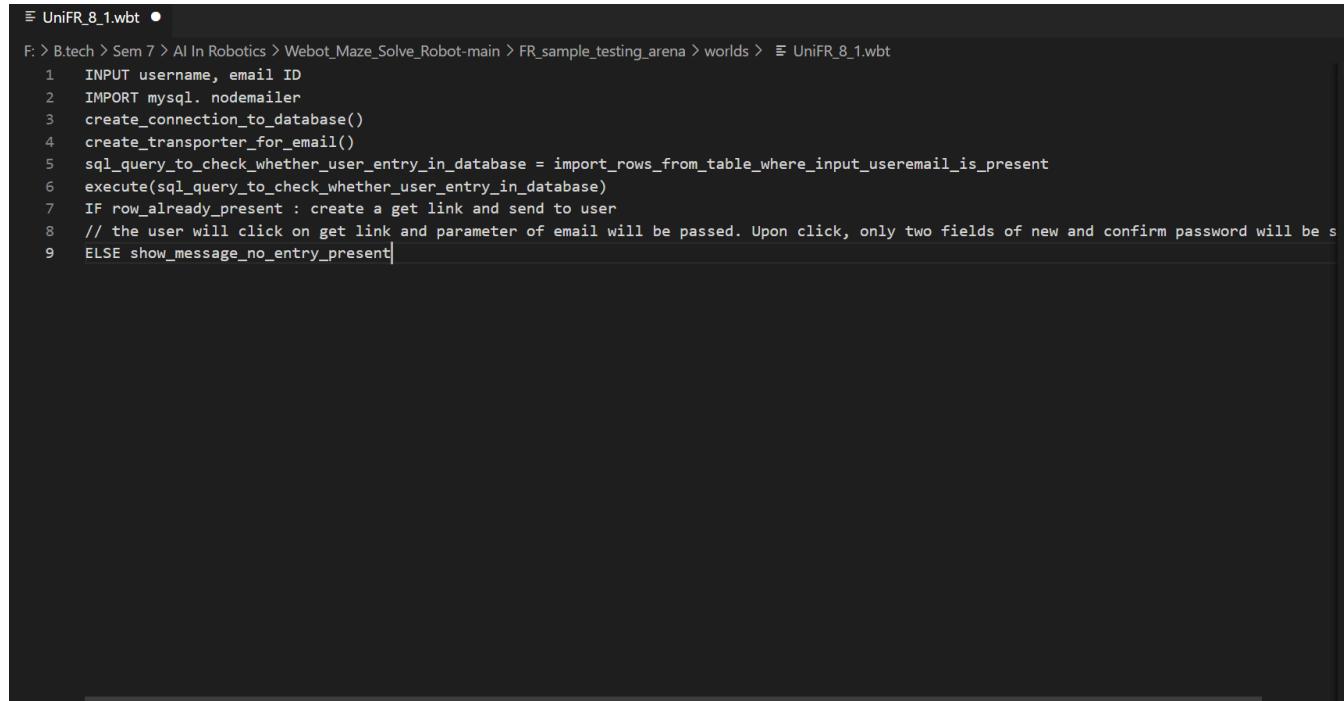
The screenshot shows a code editor window titled "UniFR_8_1.wbt". The code is written in a pseudo-code-like language used for Webots scripts. It includes imports for nodemailer and mysql, database connection creation, transporter creation, and an SQL query to check if a user's email is already present. It then executes the query and branches based on whether a row was found, showing a message or sending an OTP mail.

```
F: > B.tech > Sem 7 > AI In Robotics > Webot_Maze_Solve_Robot-main > FR_sample_testing_arena > worlds > UniFR_8_1.wbt
1 INPUT username, email ID
2 IMPORT nodemailer, mysql
3 create_connection_to_database()
4 create_transporter_for_email()
5 sql_query_to_check_whether_user_has_already_availed_facility = import_rows_from_table_where_input_useremail_is_present
6 execute(sql_query_to_check_whether_user_has_already_availed_facility)
7 IF row_already_present : show_message_already_used
8 ELSE send_OTP_mail|
```

Figure A.2: Sudo Code 2

A.3 Unsubscribe link

- Sudo Code

A screenshot of a software interface titled "UniFR_8.wbt". The title bar also shows the path: F: > B.tech > Sem 7 > AI In Robotics > Webot_Maze_Solve_Robot-main > FR_sample_testing_arena > worlds > UniFR_8.wbt. The main window contains a block of code labeled "Sudo Code 3".

```
UniFR_8.wbt •
F: > B.tech > Sem 7 > AI In Robotics > Webot_Maze_Solve_Robot-main > FR_sample_testing_arena > worlds > UniFR_8.wbt
1 INPUT username, email ID
2 IMPORT mysql. nodemailer
3 create_connection_to_database()
4 create_transporter_for_email()
5 sql_query_to_check_whether_user_entry_in_database = import_rows_from_table_where_input_useremail_is_present
6 execute(sql_query_to_check_whether_user_entry_in_database)
7 IF row_already_present : create a get link and send to user
8 // the user will click on get link and parameter of email will be passed. Upon click, only two fields of new and confirm password will be s
9 ELSE show_message_no_entry_present|
```

Figure A.3: Sudo Code 3

Appendix B

ICMR Conference Acceptance Letter



Figure B.1: Acceptance Letter

Appendix C

Conference Participation

C.1 Harshal Patil

- Certificate



Figure C.1: ICMR Certificate

C.2 Yash Dengre

- Certificate



Figure C.2: ICMR Certificate

C.3 Vishal Sule

- Certificate



Figure C.3: ICMR Certificate

C.4 Manoranjan Jena

- Certificate



Figure C.4: ICMR Certificate

C.5 Mrs. Kavitha S.

- Certificate



Figure C.5: ICMR Certificate

Appendix D

Plagiarism Report of Text

ks_g54

by A B

Submission date: 14-Dec-2022 06:45AM (UTC-0500)

Submission ID: 1946801107

File name: BTECH_Project_Report_Template__4Students__Copy.pdf (4.55M)

Word count: 7126

Character count: 39703



PRIMARY SOURCES

- | | | |
|---|---|-----|
| 1 | Submitted to MIT Academy of Engineering
Student Paper | 4% |
| 2 | www.coursehero.com
Internet Source | 2% |
| 3 | Submitted to Kent Institute of Business and Technology
Student Paper | 1% |
| 4 | Submitted to National College of Ireland
Student Paper | <1% |
| 5 | lup.lub.lu.se
Internet Source | <1% |
| 6 | Submitted to Sogang University
Student Paper | <1% |
| 7 | Submitted to University of Bahrain
Student Paper | <1% |
| 8 | pce.ac.in
Internet Source | <1% |
| 9 | Submitted to Liverpool John Moores University | <1% |

- 10 Vanathi Eniyal R, Sathya Priya K, Meeralakshmi M, Sivanantham S. "Securing Technology and Government Services Enrollment - Stage", 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), 2020
Publication <1 %
- 11 repository.unika.ac.id <1 %
Internet Source
- 12 Submitted to University of East London <1 %
Student Paper
- 13 docplayer.net <1 %
Internet Source
- 14 github.com <1 %
Internet Source
- 15 www.tdnsindia.org <1 %
Internet Source
- 16 www.slideshare.net <1 %
Internet Source
- 17 "[iCATccT 2018 Front Matter]", 2018 4th International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT), 2018 <1 %
Publication

Submitted to University of Greenwich

Student Paper

18

<1 %

19

commons.wikimedia.org

Internet Source

<1 %

20

dokumen.pub

Internet Source

<1 %

21

home.medewerker.uva.nl

Internet Source

<1 %

22

Di Wu. "Robust Latent Feature Learning for Incomplete Big Data", Springer Science and Business Media LLC, 2023

Publication

<1 %

23

ukdiss.com

Internet Source

<1 %

24

"Contents", 2017 International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE), 2017

Publication

<1 %

Exclude quotes On

Exclude matches Off

Exclude bibliography On

References

- Abhineet Anand, P. S. N., Tanishk Sharma (Ed.). (2017). *Using kiosks as information-delivery channels to apply for schemes and other government services for rural india.* International Journal Of Engineering And Computer Science.
- Deshpande, A. M., & Pandurangi, V. J. (Eds.). (2018). *Measuring the effectiveness of government schemes using machine larning algorithms.* IEEE International Conference on Applied and Theoretical Computing and Communication Technology.
- jagranjosh. (n.d.). Retrieved 2022-12-13, from <https://www.jagranjosh.com/>
- Ministry of rural development government of india. (n.d.-a). Retrieved 2022-12-13, from <https://rural.nic.in/>
- Ministry of rural development government of india. (n.d.-b). Retrieved 2022-12-13, from <https://rural.nic.in/>
- Naresh Saxena, J. F., Radhika Nayak (Ed.). (2002). *Reaching the poor: The influence of policy and administrative processes on the implementation of government poverty schemes in india.* Research Gate,.
- Neha Kumar, A. A. A. J. S. C. P. M. A. R. Q., Kalyani Raghunathan (Ed.). (2019). *Social networks, mobility, and political participation: The potential for women's self-help groups to improve access and use of public entitlement schemes in india.* New Delhi, India: International Food Policy Research Institut,.
- RAO, P. S. (Ed.). (2019). *Rural development schemes in india – a study.* INTERNATIONAL JOURNAL OF RESEARCH AND ANALYTICAL REVIEWS (IJRAR.ORG) ,.
- Samruddhi Khandare, S. G., & Turkar, V. (Eds.). (2017). , *design and development of e farm with s.c.h.e.m.e.e.* IEEE Internation Conference on Recent Innovations in Signal processing and Embedded Systems,.
- Scroll. (n.d.). Retrieved 2022-12-13, from <https://amp.scroll.in/article/1003219/>

nearly-400000-farmers-committed-suicide-in-india-between-1995-and-2018
-why

- Tejas R. Marawar, S. P. K., & I.Araspure, K. (2010). *E governance*. International Conference on Data Storage and Data Engineering.
- vikaspedia. (n.d.). Retrieved 2022-12-13, from <https://vikaspedia.in/>
- Vivek Srinivasan, S. K. S. A. R. N. P. S. D. C. A. P. S. A. S., Vibhore Vardhan (Ed.). (2013). *Airavat: an automated system to increase transparency and accountability in social welfare schemes in india*. New York, NY, United States: Association for Computing Machinery,.