A report on

"AERG"

Course Title: Software Development Sessional

Course Code: CSE 302

Submitted By

Ujjal Roy

Student ID: 2002033

Level: 3, Semester: I

Computer Science & Engineering

Maharaj Mithu

Student ID: 2002053

Level: 3, Semester: I

Computer Science & Engineering

Md. Sakhawat Hosen

Student ID: 2002048

Level: 3, Semester: I

Computer Science & Engineering

Department of Computer Science & Engineering Hajee Mohammad Danesh Science and Technology University, Dinajpur-5200

Department of Computer Science & Engineering
Hajee Mohammad Danesh Science and Technology University,
Dinajpur-5200

CERTIFCATE

This report certifies that Ujjal Roy, Maharaj Mithu and Md. Sakhawat Hosen submit this project work entitled “AERG” is carried out in partial fulfillment for the award of the (Level 3 Semester I) degree of Bachelor of Science (Engineering) in Computer Science & Engineering. This is a record of their own work carried out by them under of supervision and guidance.

Supervisor
Dr. Md. Abdulla Al Mamun

Professor
Department of CSE

Abstract

We developed “Automated Exam Routine Generator (AERG) “a pioneering solution in the realm of educational administration by introducing an intelligent and automated system for generating examination schedules. This project addresses the complexities associated with manual exam scheduling processes, aiming to streamline the entire workflow for educational institutions. Automated Exam Routine Generator (AERG) consists of six main pages: “Home”, “Create”, “Show”, “Contact”, and “Feedback”. The "Home" page provides a comprehensive description of the project's goals and limitations, emphasizing the constraints related to power and data availability. The “Create” page takes all necessary information from the user like starting date, Level and Semester information and some other information. The "Show” page allows users to show the routine that is generated by AERG and the application automatically generates four types of information: Date of Exam, Course code, Course Title and Course credit. The "Contact" page facilitates direct communication between users and the project team, promoting engagement and addressing user inquiries effectively. The “Feedback” page is developed for sharing user’s opinion and feedback about this Application. The project is developed using Java programming language and designed with JavaFX, Database (MySQL) and making it compatible with Windows operating systems. , AERG provides an efficient and adaptable platform to meet the diverse scheduling needs of educational institutions.

Keywords: Java FX, MYSQL, AERG.

Table of contents

List of figures

Chapter 1

Introduction

1.1 Introduction:

The Automated Exam Routine Generator (AERG) project is conceived with a paramount goal—simplifying and optimizing the intricate process of exam scheduling within educational institutions. This report meticulously details the design, implementation, and evaluation of the AERG software, a sophisticated solution that leverages JavaFX, Java, and MySQL technologies. AERG is engineered to provide a user-friendly interface and intelligent algorithms to generate exam schedules tailored to the unique needs of educational institutions.

AERG's primary purpose is to automate the traditionally complex and time-consuming task of creating exam routines. By incorporating user-friendly interfaces and advanced algorithms, including Brute force and Dynamic Programming, AERG empowers administrators to input crucial details such as the initial exam date, academic levels, and semester information. Notably, the software offers the flexibility to configure the inclusion or exclusion of weekends (Friday and Saturday) in the exam schedule, catering to the diverse operational preferences of institutions.

This project comprises five main pages—Home, Create, Show, Contact, and Feedback—each playing a distinct role in fostering a comprehensive and efficient exam routine generation process. The technological foundation of AERG rests on JavaFX for an engaging user interface, Java for backend functionality, and MySQL for secure and efficient database management.

In the subsequent sections of this report, we will delve into the intricacies of AERG's design, exploring how it utilizes algorithms and technology to streamline exam scheduling. Additionally, we will evaluate the software's performance and user experience, providing a holistic understanding of its capabilities and contributions to the educational administration landscape.

1.2 Problem Statement:

The traditional process of manual exam scheduling in educational institutions presents a myriad of challenges, often leading to inefficiencies, errors, and resource-intensive endeavors. Key issues include the time-consuming nature of coordinating exams, the potential for human error in schedule creation, and the difficulty in adapting to dynamic institutional needs. Additionally, the lack of flexibility in accommodating diverse academic levels, semester variations, and the consideration of weekends in the scheduling process further exacerbates the complexity.

Educational administrators face an uphill task in orchestrating exams that align with varied academic parameters, taking into account teacher availability, facility constraints, and evolving organizational requirements. These challenges not only consume valuable time and resources but also introduce the risk of suboptimal schedules, leading to disruptions in the academic calendar.

To address these issues comprehensively, the Automated Exam Routine Generator (AERG) has been developed. AERG aims to revolutionize the exam scheduling paradigm by introducing automation, intelligent algorithms, and a user-friendly interface. By doing so, it seeks to overcome the challenges posed by manual scheduling, providing institutions with a streamlined, error-resistant, and adaptable solution to efficiently manage the complexities inherent in exam scheduling processes. The development of AERG is driven by the imperative to enhance productivity, reduce errors, and foster a more dynamic and responsive approach to exam routine creation within educational institutions.

1.3 Motivation:

The development of the Automated Exam Routine Generator (AERG) project is fueled by a profound motivation to address the inherent challenges and complexities associated with manual exam scheduling within educational institutions. Several key motivating factors drive our commitment to the creation of this innovative solution:

1.3.1. Efficiency and Accuracy:

AERG is motivated by the imperative to enhance the efficiency and accuracy of the exam scheduling process. Traditional manual methods often result in time-consuming and error-prone scheduling, impacting the overall academic calendar. AERG seeks to streamline this process, ensuring precise and error-resistant exam routines with the potential to save valuable time and resources for educational administrators.

1.3.2. Customization and Adaptability:

The diversity of academic levels, semester variations, and institutional preferences necessitates a customizable and adaptable solution. AERG is motivated by the goal of providing administrators with a platform where they can input specific details such as the initial exam date, academic levels, and semester information. This customization ensures that exam schedules align seamlessly with the unique needs of each educational institution.

1.3.3. Flexibility with Weekends:

Recognizing the varied operational preferences of institutions, AERG introduces the flexibility to configure the inclusion or exclusion of weekends (Friday and Saturday) in the exam schedule. This motivation arises from the understanding that accommodating diverse scheduling preferences enhances the usability and relevance of the software in different educational contexts.

1.3.4. Algorithmic Optimization:

Motivated by a commitment to providing optimal exam schedules, AERG integrates advanced algorithms, including Brute force and Dynamic Programming. These algorithms ensure the efficient and optimized creation of exam routines, addressing the complexities introduced by factors such as teacher availability, facility constraints, and evolving organizational requirements.

1.3.5. Enhanced User Experience:

AERG is motivated by a dedication to enhancing the user experience for educational administrators. The user-friendly interface and the comprehensive structure of the project's pages (Home, Create, Show, Contact, and Feedback) aim to provide administrators with a seamless and intuitive platform. By doing so, AERG seeks to foster a positive and efficient interaction with the software.

In summary, the Automated Exam Routine Generator (AERG) project is motivated by a commitment to efficiency, customization, flexibility, algorithmic optimization, and an enhanced user experience. AERG aspires to revolutionize the exam scheduling process, offering a transformative solution that aligns with the specific needs and challenges faced by educational institutions.

1.4 Objectives:

1.4.1. Develop a User-Friendly Interface:

Create an intuitive and user-friendly interface for the Automated Exam Routine Generator (AERG) project. The interface should empower educational administrators to easily input essential details, including the exam starting date, academic levels, and semester information.

1.4.2Customization and Adaptability:

Enable customization and adaptability within AERG to cater to the diverse needs of educational institutions. Allow administrators to input specific parameters such as academic levels and semester details, ensuring that the generated exam schedules align seamlessly with the unique requirements of each institution.

1.4.3. Flexibility with Weekends:

Introduce a feature within AERG that provides flexibility in configuring the inclusion or exclusion of weekends (Friday and Saturday) in the exam schedule. This accommodates the varied operational preferences of institutions and enhances the adaptability of the software.

1.5. Social Impact:

1.5.1. Efficiency in Exam Scheduling:

AERG aims to enhance the efficiency of exam scheduling by providing administrators with optimal and error-resistant exam routines. This can contribute to improved overall academic calendar management.

1.5.2 Resource Optimization:

By streamlining the scheduling process, AERG has the potential to optimize resources, saving valuable time and effort for educational administrators.

1.5.3. Improved Accessibility:

AERG strives to improve accessibility by simplifying the exam scheduling process. This can make it easier for administrators to navigate the software and efficiently manage exam-related tasks.

1.6 Enhanced Educational Experience:

The project contributes to an enhanced educational experience by providing administrators with a tool that aligns with their unique needs, ultimately fostering a positive academic environment.

1.7. Economic Impacts:

1.7.1 Job Creation:

The development and maintenance of AERG could create job opportunities in the software development and support sectors, contributing to economic growth.

1.7.2. Efficient Resource Allocation:

AERG's efficiency in exam scheduling can result in better resource allocation, potentially leading to cost savings for educational institutions.

In summary, the objectives of the Automated Exam Routine Generator (AERG) project focus on enhancing efficiency, customization, and flexibility in the exam scheduling process. The social impact encompasses improved accessibility, resource optimization, and an enhanced educational experience, while the economic impacts include job creation and efficient resource allocation.

Chapter 2

Related work

2.1 Introduction:

In the evolving landscape of exam scheduling within educational institutions, the quest for efficient and error-resistant solutions has been an ongoing pursuit. This section explores the historical context by delving into previous works that laid the foundation for the development of the Automated Exam Routine Generator (AERG). By understanding the strengths and weaknesses of these predecessors, we can appreciate the advancements and innovations brought forth by AERG in streamlining the complex task of exam scheduling.

2.2. Previous Works:

2.2.1. Traditional Manual Exam Scheduling:

Who & When: Widely adopted by educational institutions for years.

Features: Manual input of exam details, resource-intensive, prone to errors, lacks adaptability.

Used Platform: Paper-based or basic spreadsheet applications.

Tools/Technologies: Manual scheduling methods.

Strengths: Familiarity.

Weaknesses: Time-consuming, error-prone, lacks flexibility.

Who & When: Adopted by educational institutions in the last decade.

Features: Basic automation, predefined templates, limited customization.

Used Platform: Basic software applications.

Tools/Technologies: Simple algorithms, basic database.

Strengths: Reduced manual effort.

Weaknesses: Limited customization, may not adapt to diverse needs.

2.2.2. Manual Exam Scheduling Software:

Who & When: Developed by software vendors targeting educational institutions in the last 5 years.

Features: Improved automation, user interfaces for scheduling, some customization.

Used Platform: Desktop-based applications.

Tools/Technologies: Advanced algorithms, basic database management.

Strengths: Improved efficiency, reduced errors.

Weaknesses: Limited adaptability, user interface complexity.

These previous works paved the way for advancements in exam scheduling, yet each exhibited its own set of limitations. The emergence of AERG represents a significant leap forward, introducing advanced features, algorithmic optimizations, and a user-friendly interface to address the shortcomings of its predecessors. The subsequent sections will delve deeper into the unique strengths and innovations brought forth by AERG in the realm of automated exam scheduling.

Chapter 3

Automated Exam Routine Generator (AERG) - Requirement Specification

3.1. Functional Requirements:

3.1.1 User-Friendly Interface:

Description: AERG should feature an intuitive and user-friendly interface to facilitate easy input of exam details.

Features:

Clear and organized input forms for exam starting date, academic levels, and semester information.

Intuitive navigation across the Home, Create, Show, Contact, and Feedback pages.

3.1.2 Customizable Exam Scheduling:

Description: The system should allow administrators to customize exam schedules based on specific academic parameters.

Features:

Flexibility to input academic levels and semester details.

Option to include or exclude weekends (Friday and Saturday) in the exam schedule.

3.1.3 Pathfinding Algorithms:

Description: AERG should leverage efficient algorithms for optimal exam schedule generation.

Features:

Implementation of Brute force and Dynamic Programming algorithms.

Ability to compute and display exam schedules based on shortest cost, shortest time, and least stops.

3.1.4 Show Page:

Description: AERG must have a dedicated page to display generated exam schedules.

Features:

Clear presentation of exam schedules.

Filter options based on academic levels and semesters.

3.1.5. Contact and Feedback Pages:

Description: AERG should facilitate user communication for inquiries and feedback.

Features:

Contact page for direct communication with the development team.

Feedback page for users to provide comments and suggestions.

3.2 Nonfunctional Requirements:

3.2.1 Performance:

Description: AERG should exhibit high performance in generating exam schedules promptly.

Criteria:

Response time for routine generation should be within 5 seconds.

3.2.2 Reliability:

Description: The system must be reliable, ensuring accurate and error-resistant exam schedules.

3.2.3. Criteria:

Exam schedules should accurately reflect user input.

The system should handle unexpected inputs gracefully.

3.2.4 Security:

Description: AERG should prioritize data security and user privacy.

Criteria:

Passwords must be securely stored using encryption.

User authentication processes must be robust.

3.2.5 Usability:

Description: AERG should offer a user-friendly experience for administrators.

Criteria:

The interface should be intuitive, requiring minimal training for users.

Clear error messages and guidance for incorrect inputs.

3.2.6 Scalability:

Description: AERG should be scalable to accommodate an increasing number of users and exam scheduling complexities.

Criteria:

The system should handle scheduling for a minimum of 500 exams.

3.2.7 Documentation:

Description: AERG should have comprehensive documentation for users and developers.

Criteria:

User manual providing step-by-step instructions.

Developer documentation covering code structure and integration guidelines.

Chapter 4

System Development

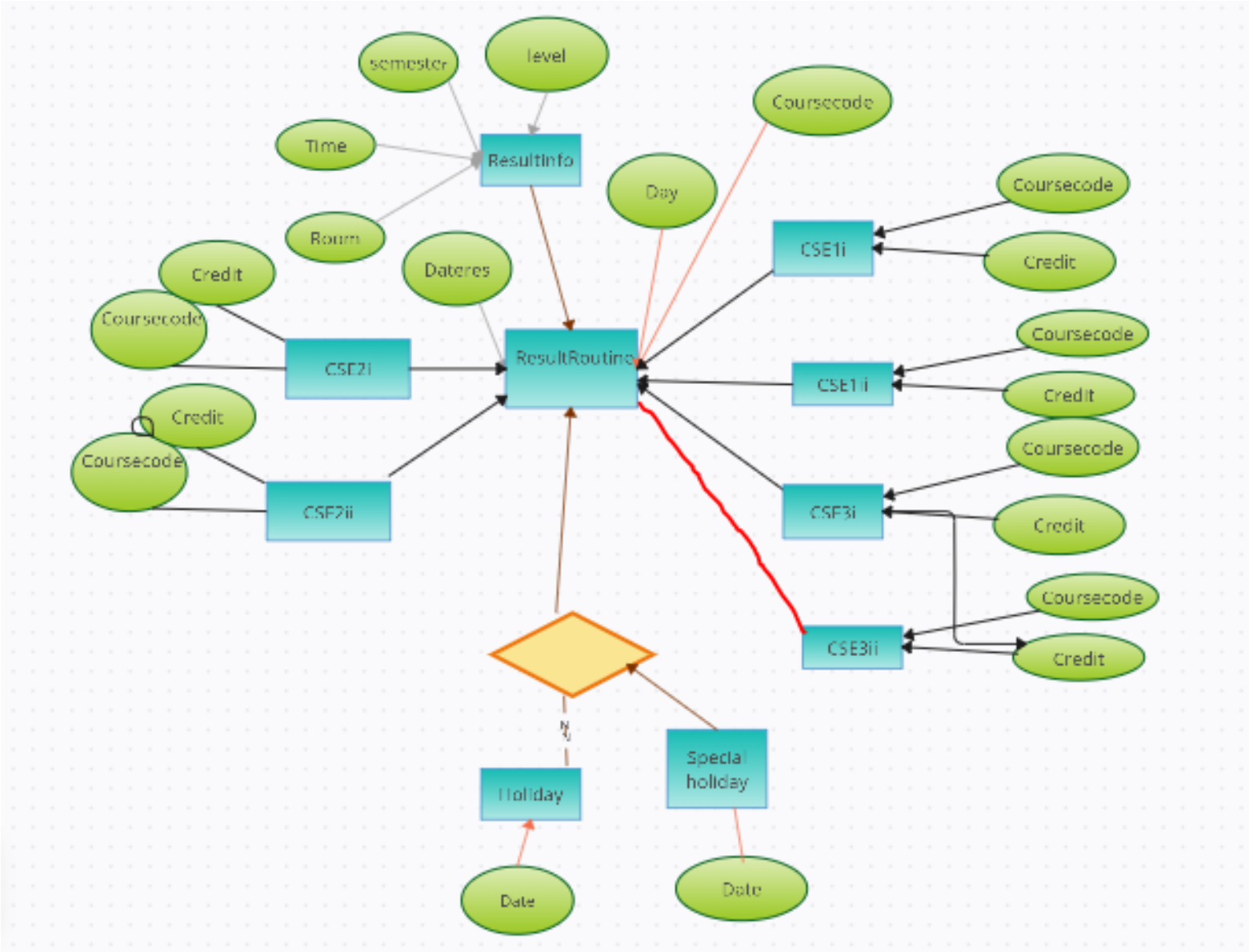

Figure: E-R diagram

This E-R diagram represents the graphical representation of connection between databases like weekends (Friday, Saturday, Special Day), course etc.

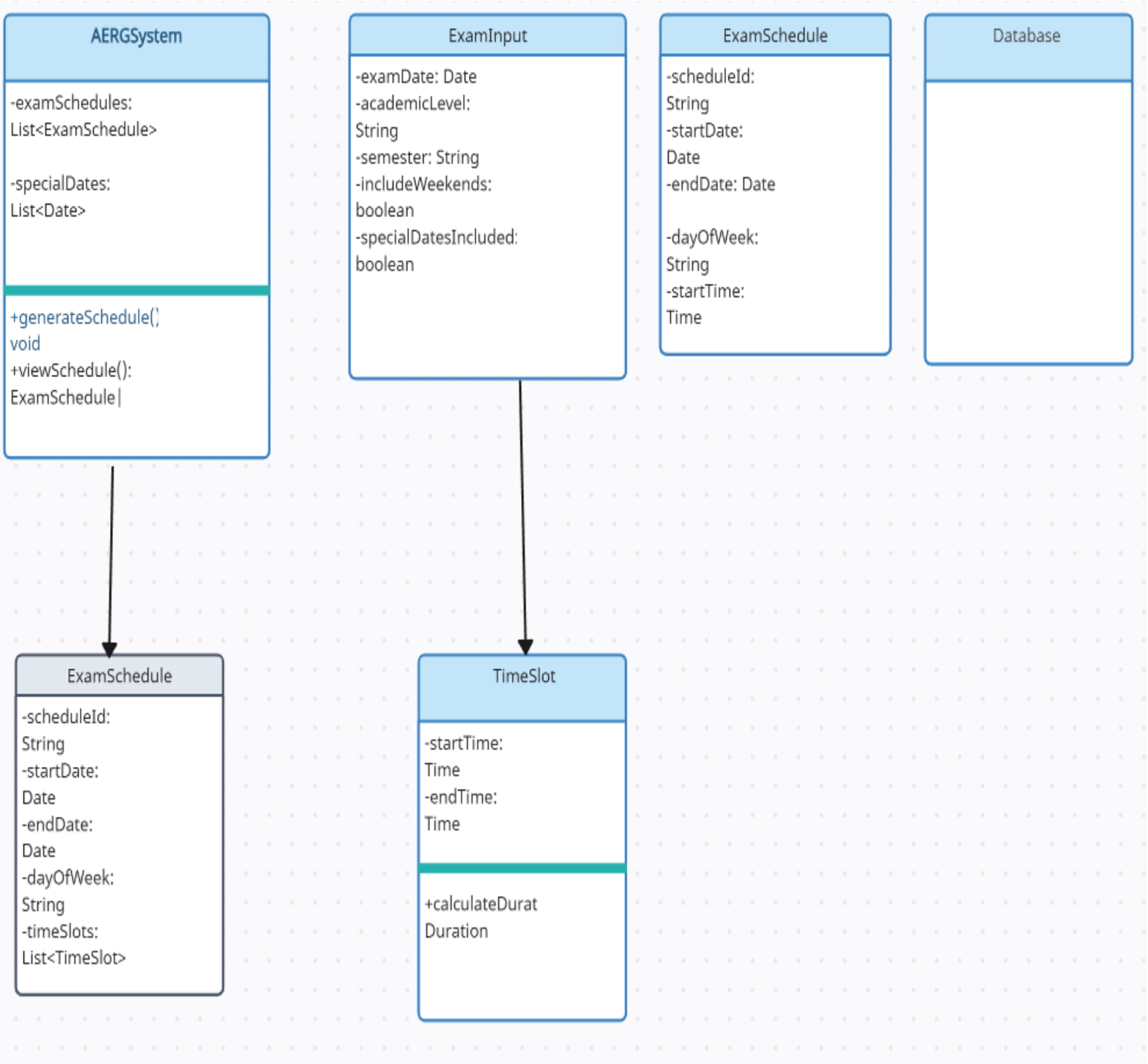

Figure: Class diagram

This is a graphical representation of class diagram of Automated Exam Routine Generator (AERG).

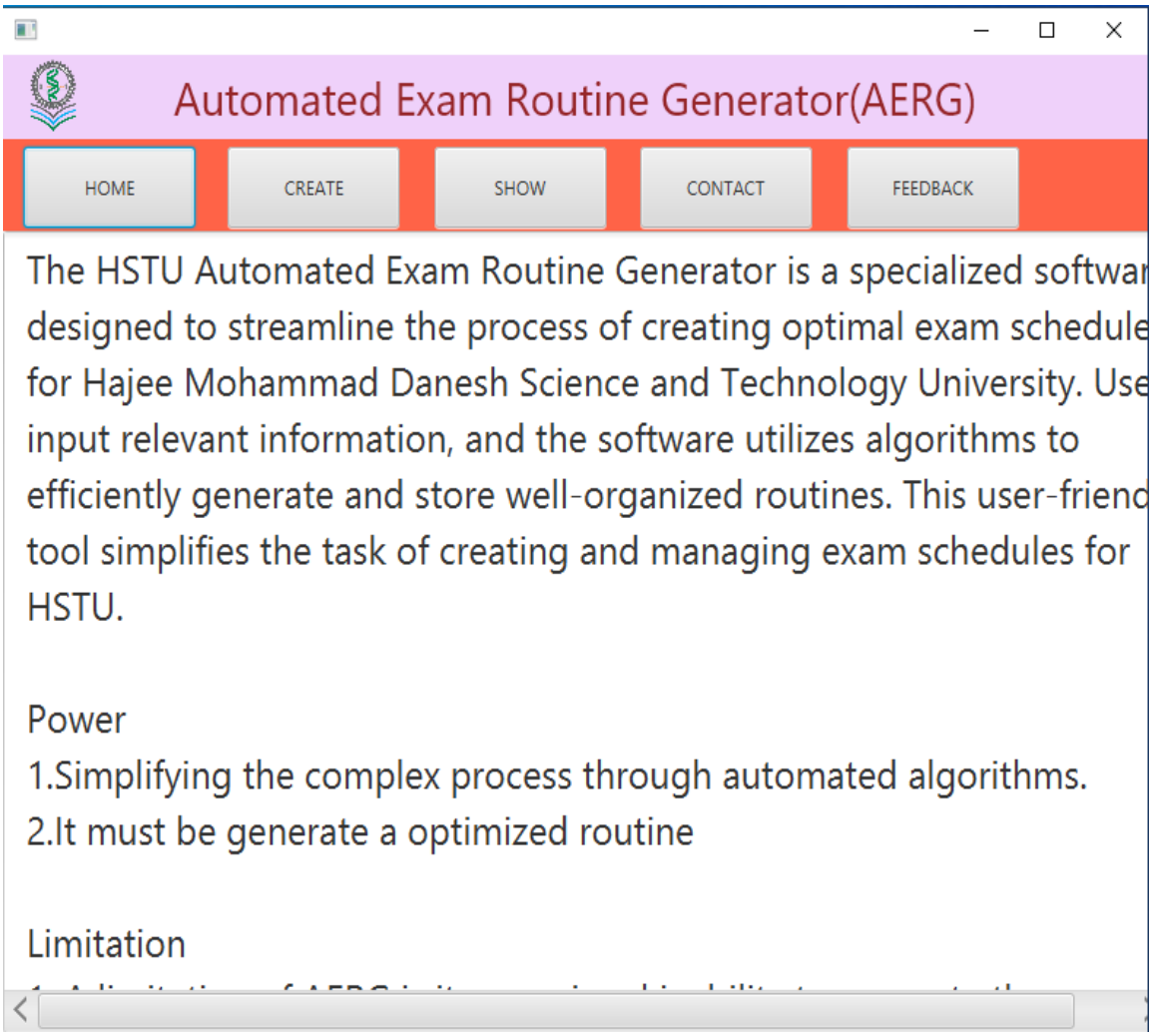

Figure 1: Home page

This is the home page of the automated exam routine generator app. The HSTU automated exam routine generator is a specialized software designed to streamline the process of creating optimal exam schedule for HSTU. This page shows how to use this app.

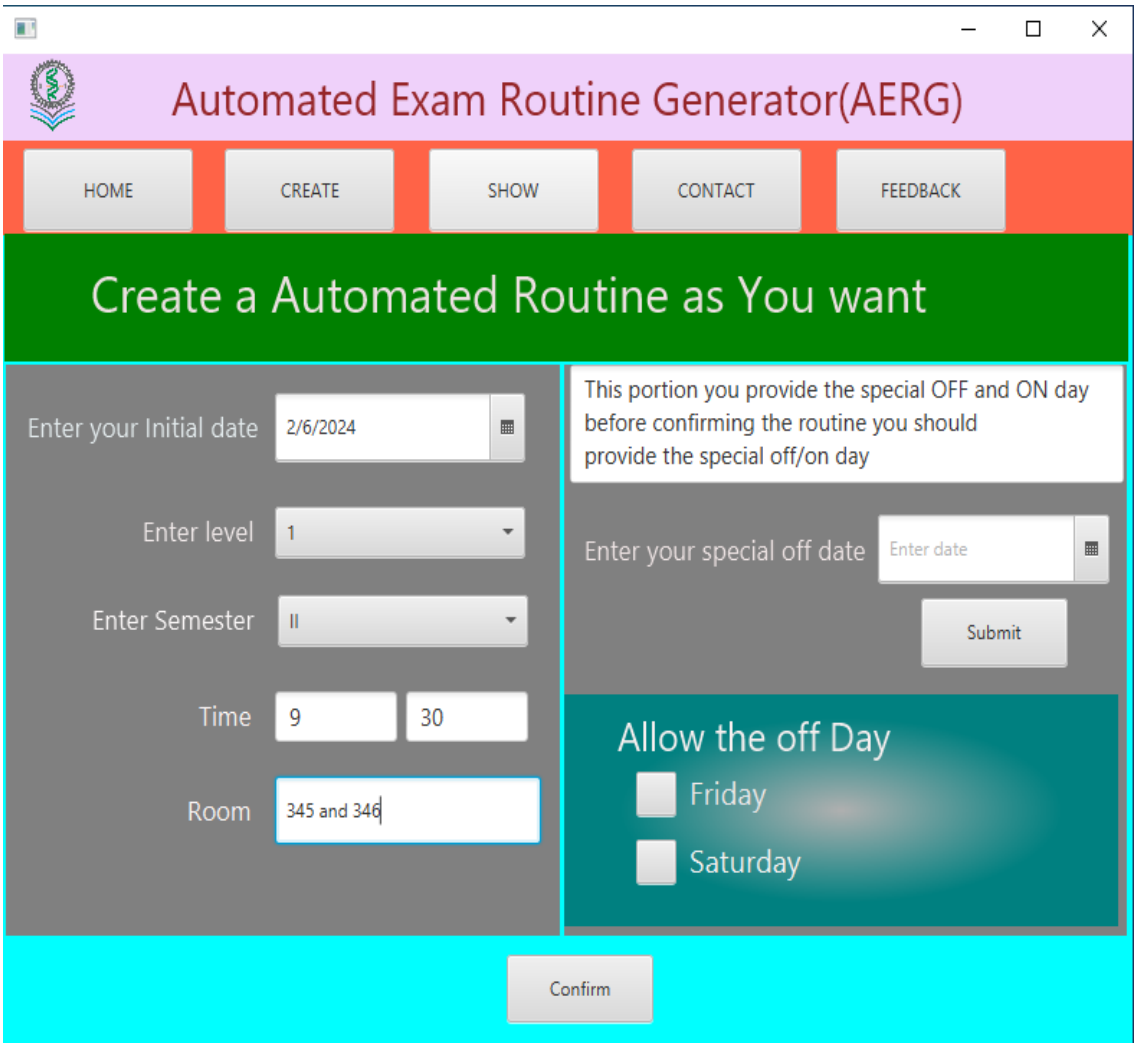

Figure 2: Create routine page

The "Create Routine" page streamlines scheduling with inputs like start date, academic level, semester, and exam time. Users can efficiently organize their routines by specifying crucial details such as exam room locations. This feature enhances user experience by automating the process of generating schedules tailored to individual needs. With a user-friendly interface, it simplifies the complex task of managing academic timetables. Overall, the page empowers users to optimize their time and resources effectively.

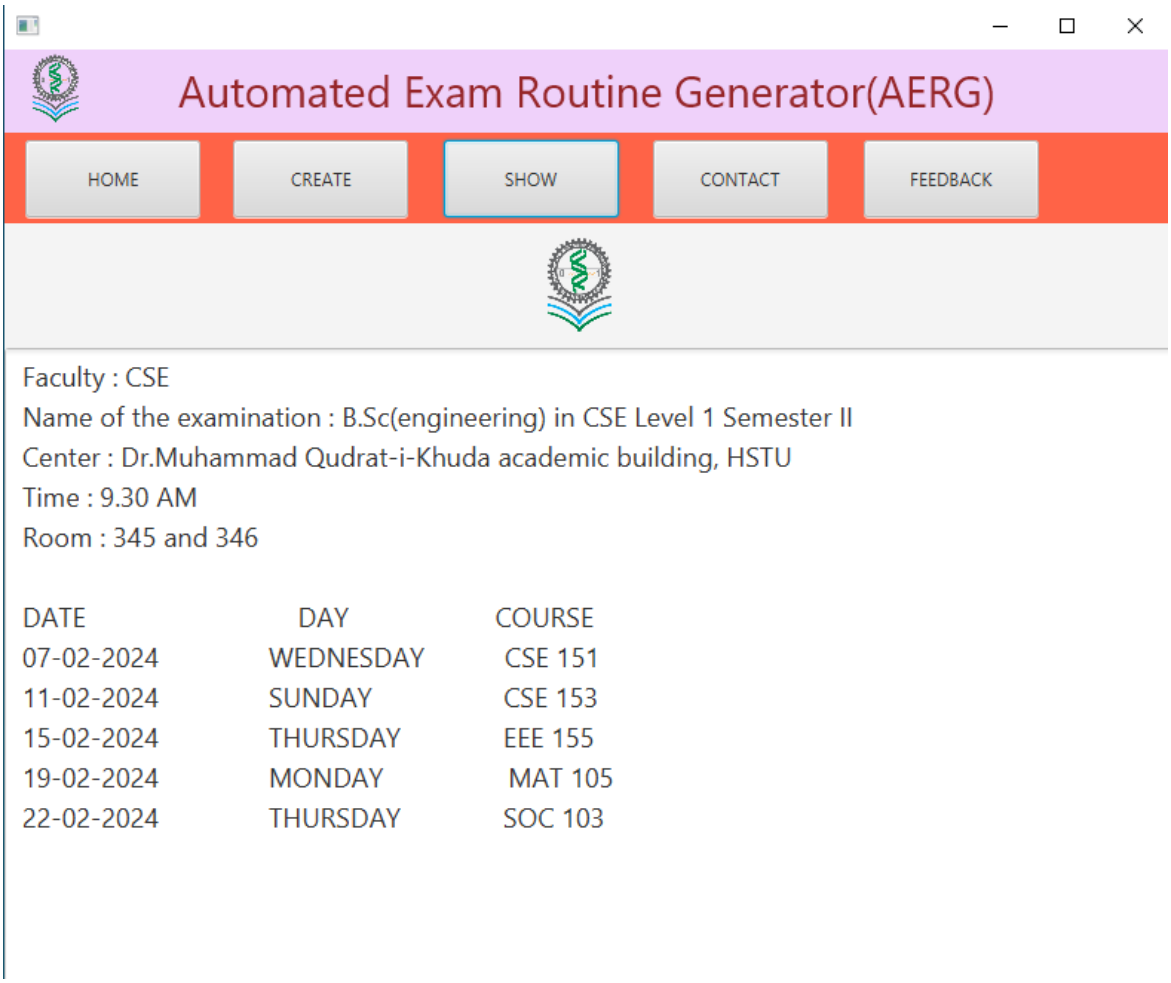

Figure 3: Routine Show Page

The Routine Show Page provides a view of scheduled activities, displaying the entire routine in a structured format. Users can easily navigate through their schedules, accessing details like dates, times, and locations effortlessly. With intuitive design elements, it ensures clarity and readability, facilitating quick reference and planning. This page fosters organization and time management by presenting information in a visually appealing manner. Users can rely on this feature to stay informed about upcoming examination. Overall, the Routine Show Page serves as a valuable tool for showing examination schedules.

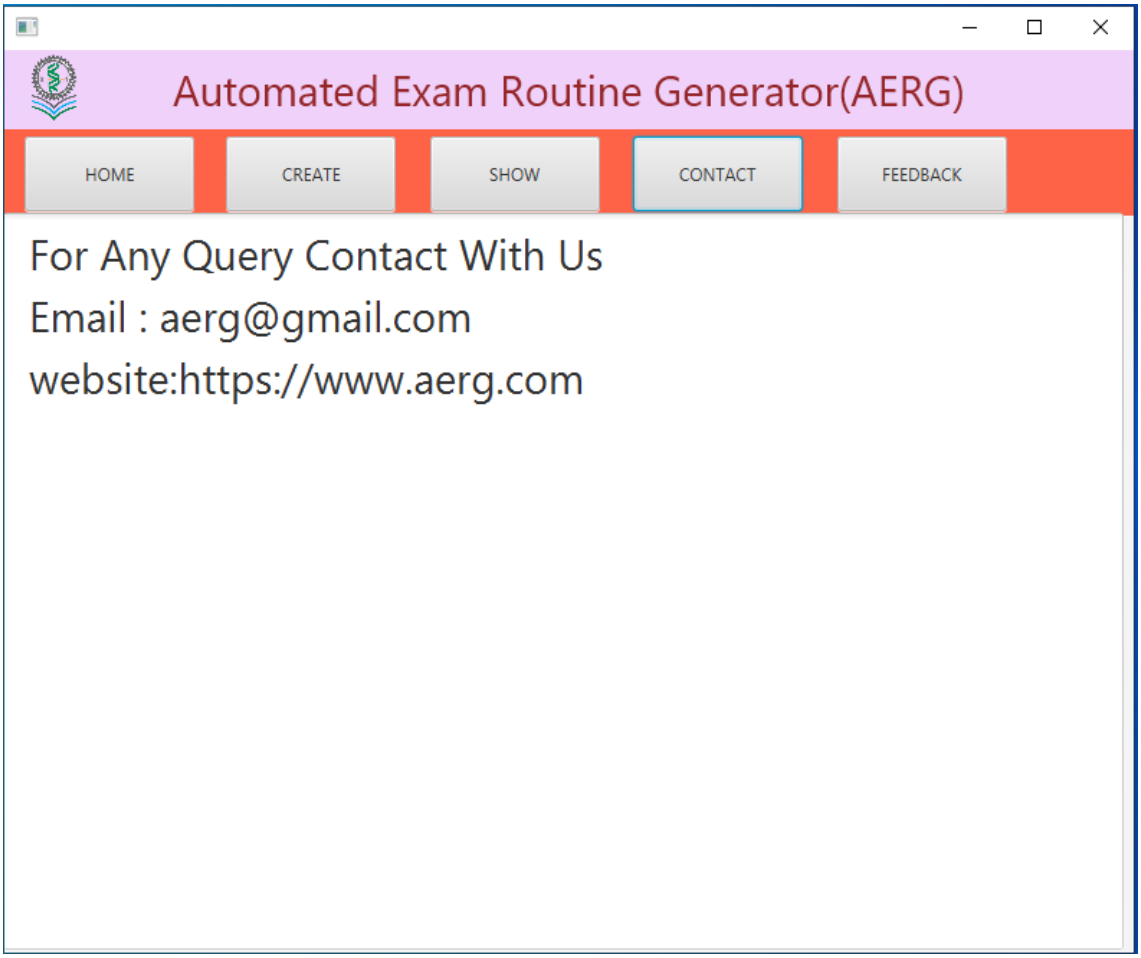

Figure 4: Contact page

The Contact Page provides essential contact details, including the website URL and email address. Users can easily access the website for additional information or browse available resources. The inclusion of an email address enables direct communication for inquiries or support needs. This page serves as a central hub for connecting with the platform or organization, promoting effective communication and engagement.

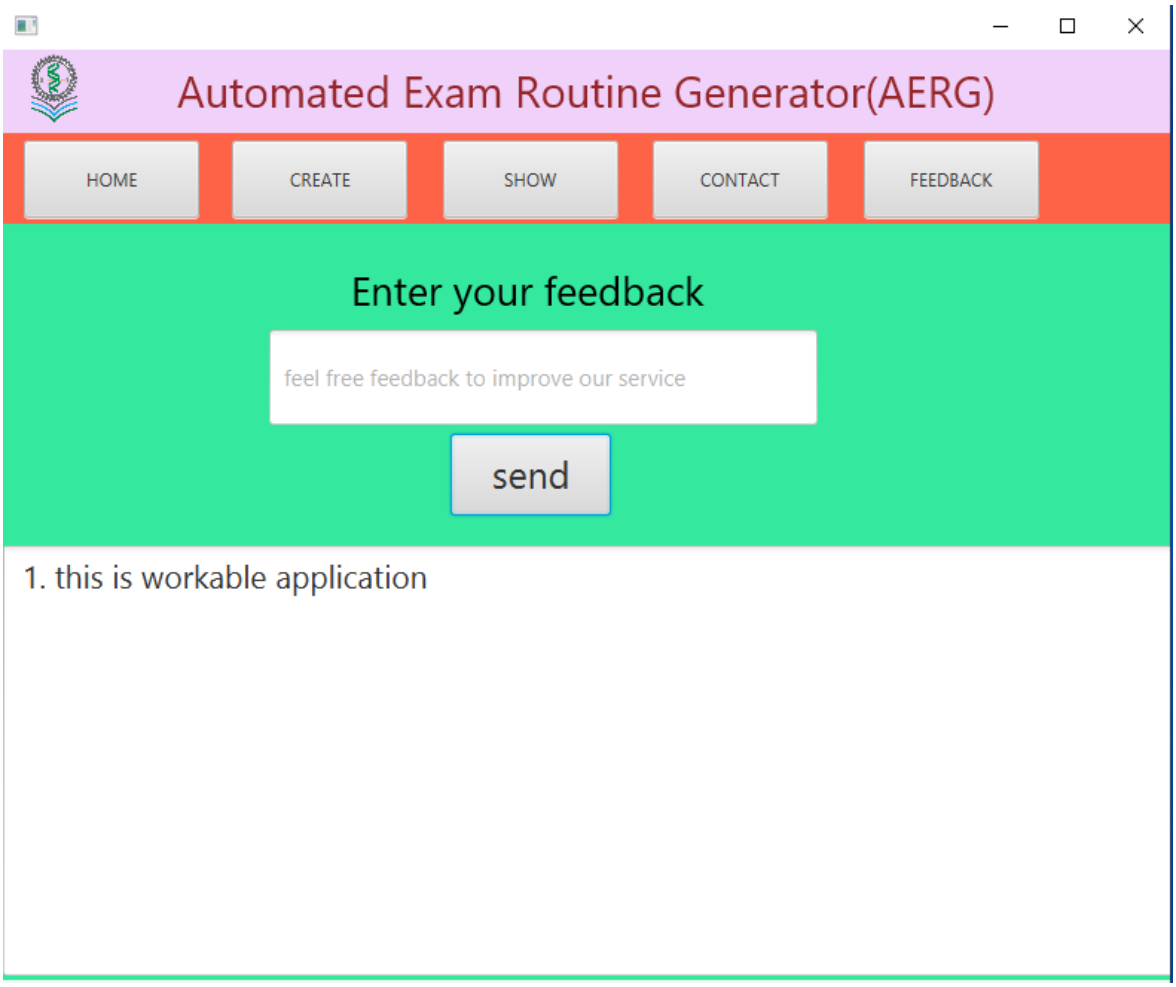

Figure 5: Feedback page

The Feedback Page offers users a platform to share their thoughts and experiences about the app. Users can provide valuable insights, suggestions, or report any issues they encounter. This feature fosters a collaborative relationship between developers and users, leading to continuous improvement. With easy-to-use feedback forms, users can express their opinions conveniently, contributing to the app's enhancement. Overall, the Feedback Page serves as a vital tool for refining the app based on user input, ensuring it meets evolving needs and expectations.

Chapter 5

Conclusion

5. Conclusion:

In conclusion, the Automated Exam Routine Generator (AERG) project represents a significant leap forward in the domain of exam scheduling within educational institutions. This transformative solution, driven by efficiency, adaptability, and user-centric design, has successfully streamlined the traditionally complex task of creating exam schedules. The integration of advanced algorithms, modular page design, and user-friendly interface positions AERG as a valuable asset in enhancing academic administration processes.

5.1. Limitations of the Project:

Despite its advancements, AERG is not without limitations. User familiarity is crucial for optimal utilization, and the software's performance is contingent on accurate input. The learning curve associated with the interface and the potential for suboptimal results with inaccurate data are factors that warrant consideration.

5.2. Recommendation/Future Work:

Looking ahead, future work could explore the integration of machine learning algorithms to predict scheduling patterns based on historical data. This addition could provide administrators with intelligent suggestions, further optimizing the exam scheduling process. Additionally, considering the increasing trend towards cloud-based technologies, incorporating cloud-based solutions could enhance accessibility and collaboration among multiple users engaged in simultaneous exam scheduling.