

TIMETABLE MANAGEMENT SYSTEM

A final year project documentation submitted to the Department of Computer Science University of Turbat (Kech) Baluchistan in partial fulfillment of the requirement degree of Bachelor of Science in Computer Science (BSCS)

Session 2021-2024

Submitted By:

1. **Bakhshullah Wahid**
2. **Abdul Wahab**
3. **Robina**
4. **Rahat Manzoor**

Supervised By:

Mr. Mamoon Rasheed
Department of IT





CERTIFICATE

This is certified that the work presented in this project report on “**Time Table Management System**” in partial fulfillment of the requirement degree of **Bachelor of Science in Computer Science (BSCS) Session 2021-2024** is entirely written by the following students themselves under the supervision of **Mr. Mamoon Rasheed** Department of IT, University of Turbat

Group Members,

1. Bakhshullah Wahid
2. Abdul Wahab
3. Robina
4. Rabat Manzoor

INTERNAL ADVISOR

Mr. Mamoon Rasheed
IT Department
University of Turbat

EXTERNAL ADVISOR

CHAIRMAN

Mr. Rashid Ali
Assistant Professor
Department of Computer Science
University of Turbat

DEDICATION

One of the biggest dedication of this project is wholeheartedly dedicated to our beloved Parents, who have been our source of inspiration and gave us strength when we thought of giving up, who continually provide their moral, spiritual, and emotional support, and encouragement. Secondly, it is dedicated to our team members whose efforts, struggles, and beliefs in one another were quite praiseworthy and appreciable in the complete process, and; finally, dedicated to our supervisor **Mr. Mamoon Rasheed**.

ACKNOWLEDGEMENT

It was not achievable for us however we have never forgotten one thing Allah is our creator, who gave us constrained capacity to make anything in this world with his approval, “so, nothing is inconceivable”. Along these lines of reasoning, we began our voyage to build up this task, it was unrealistic without the assistance of people we were going to refer with, By the dent of their help we had been able to functionalize this undertaking in some way with this documentation, we are utilizing this chance to offer our thanks to everybody who bolstered us through all the inconvenience of this venture, we are appreciative for their rousing direction and benevolent advices amid the procedure of this undertaking. We want to thank our Head of Department **Mr. Rashid Ali** and our project supervisor **Mr. Mamoon Rasheed** for always helping us out in solving our project-related problems.

ABSTRACT

Imagine a software that makes it unique and simple to split some particular manual problem known as “The Automatic Time Table Management System” (ATTMS) is a desktop application developed to automate the generation of timetables for educational institutions, addressing the challenges associated with manual scheduling. Traditional manual methods are time-consuming, error-prone, and inflexible, often leading to class conflicts, resource misallocation, and administrative inefficiencies. ATTMS aims to streamline this process by generating conflict-free schedules using advanced algorithms, ensuring optimal utilization of classrooms, instructors, and other resources. The system's objectives include automating timetable creation, ensuring conflict-free scheduling, enhancing flexibility and adaptability, centralizing information, improving accessibility, and optimizing resource utilization. By achieving these objectives, ATTMS significantly enhances the efficiency and effectiveness of academic scheduling, creating a more organized and productive educational environment. Finally, this project (ATTMS) provides security, implementing measures such as authentication systems and auto-logout features to protect sensitive data. Overall, ATTMS represents a comprehensive solution to the complexities of academic scheduling, benefiting students, faculty, and administrators by reducing manual effort, minimizing errors, and providing real-time access to updated timetables.

Table of Contents

TIME TABLE MANAGEMENT SYSTEM.....	1
DEDICATION	3
ACKNOWLEDGEMENT	4
ABSTRACT.....	5
Chapter 1.....	9
Introduction.....	Error! Bookmark not defined.
1.1. Introduction.....	Error! Bookmark not defined.
1.2. Background	9
Manual Process Challenges:	9
Impact on Stakeholders:	10
1.3. Objective	10
1.4. Problem Statement.....	11
1.4.1. Problem:.....	11
1.5. Proposed Solution.....	11
Solution:	11
1.6. Significance of Project.....	11
1.7. Functionalities.....	12
1.8. TTMS Users	12
1.9. User Roles and Responsibilities	12
1.10. Security	12
1.11. Technologies and Hardware Devices	13
LITERATURE REVIEW	14
2.2. Theoretical Review.....	14
2.2.2 Scheduling Theory.....	14
2.2.3 Decision Support Systems (DSS)	15
2.2.4 Human-Computer Interaction (HCI)	15
2.2.5 Educational Theory	15
2.2.6 Operations Research.....	16
2.2.7 Data Analytics and Machine Learning.....	16
2.3 Conclusion.....	16
Project 1 review	17
2.3.1 Introduction	17

2.3.2 Objectives.....	17
2.3.3 Methodology.....	18
2.3.4 Features	18
2.3.5 Future Scope	18
2.3.6 Reference	19
2.3.7 Lessons Learned	19
Project 2 review	20
2.3.1 Introduction	20
2.3.2 Objectives.....	20
2.3.3 Methodology.....	20
2.3.4 Features	21
2.3.5 Outcomes	22
2.3.6 Challenges	22
2.3.7 Lessons Learned	23
2.4 Conclusion	23
Chapter 3.....	24
System Analysis.....	24
3.1 Introduction:.....	24
3.2. System Requirements	24
3.2.1. Functional Requirements.....	24
3.2.2 Non-Functional Requirements	24
3.2.3 Logical Database	25
3.3 Data Collection	25
3.3.1 Primary Source:.....	25
3.3.2 Secondary Source.....	25
3.4 Data presentation	25
3.4.1 System Model	25
3.4.2 Architecture Model.....	26
3.4.3 UML.....	26
Chapter 4.....	27
System Design	27
4.1. Introduction	27
4.2. Process model	27

4.2.1. Incremental model.....	27
4.3. System Architecture.....	28
4.4 UML Diagram	28
4.4.1. Use case diagram	28
4.5 Use cases.....	31
4.5.1 Use case No# 1 Login	31
4.5.2. Use case No#2 create Users.....	32
4.5.3. Use case No#3 view Users	32
4.5.4 Use case No#4 Delete Users	33
4.5.5. Use case No#5 edit Users.....	33
4.5.6. Use case No#6 change password.....	34
4.5.7. Use case No#7 view active Timetables	34
4.5.8. Use case No#8 Logout.....	35
4.5.9. Use case No#9 Users Login	36
4.5.10. Use case No#9 Timetable creation	36
4.5.11. use case No#11 Users to View Timetable.....	37
4.5.12. Use case No#12 Delete Timetable	37
4.5.13 Use case No#8 Logout.....	38
Data Flow Diagram.....	39
4.6. User Interface	40
4.6.1. Introduction:	40

Chapter 1

Introduction

1.1. Introduction

Automatic Time Table Management System (ATTMS) is a desktop application that helps us generate timetables automatically and reduces conflict of classes and duplication of the teacher classes at the same time slot, leading to mismanagement. Manual timetable management is time-consuming and conflicting since teachers face multiple problems when they are taking classes in other departments too, like the same time slot with different department classes at the same time. Every department has its timetable when Department 1 needs a teacher from Department 2 the department needs to gather information about the free slot of that teacher which is time-consuming. The automatic timetable management system is brought to reduce this time-consuming.

1.2. Background

Before the implementation of an automated timetable management system, educational institutions primarily relied on manual methods to create and manage timetables. This process involved a significant amount of paperwork, manual scheduling, and frequent revisions to accommodate changes such as faculty availability, classroom allocation, and course adjustments.

Manual Process Challenges:

Time-Consuming:

A University timetable is a temporal arrangement of a set of lectures and classrooms in which all given constraints are satisfied. Creating such timetables manually is a complex and time-consuming process.

Error-Prone:

According to the manual scheduling process is susceptible to human errors, leading to issues such as overlapping classes, double-booked rooms, and misallocation of resources, accessing extra classrooms with unavailability of teachers.

Inflexibility:

Adapting to sudden changes, such as unexpected faculty absences or changes in course offerings, was cumbersome and often led to disruptions in the academic schedule.

Lack of Centralization:

Information was scattered across various documents and departments, making it difficult to access and update timetables efficiently.

Limited Accessibility:

Students and faculty had limited access to updated timetables, often relying on printed schedules that were posted on notice boards or distributed as hard copies.

Impact on Stakeholders:

Students:

Experienced confusion and inconvenience due to frequent changes in the timetable and lack of real-time updates.

Faculty:

Faced difficulties in managing their schedules and ensuring that they were available for their assigned classes and slots.

Administrators:

Spent a significant amount of time and resources on creating and maintaining timetables, diverting attention from other critical administrative tasks.

1.3. Objective

1. Automate Timetable Creation:

Develop a system that automatically generates timetables for different departments and classes, minimizing manual intervention and reducing the likelihood of errors.

2. Ensure Conflict-Free Scheduling:

Implement algorithms to ensure that there are no overlapping classes, double-booked rooms, or scheduling conflicts with faculty availability.

3. Enhance Flexibility and Adaptability:

Enable the system to quickly adapt to changes such as faculty absences, room reassignments, and modifications in course offerings, ensuring minimal disruption to the academic schedule.

4. Centralize Information:

Create a centralized database that stores all timetable-related information, making it easily accessible to administrators, faculty, and students.

5. Improve Accessibility:

Provide students and faculty with real-time access to updated timetables through a user-friendly web or mobile interface.

6. Optimize Resource Utilization:

Efficiently allocate classrooms, laboratories, and other resources to ensure optimal use and avoid underutilization or overbooking.

7. Support Multiple Departments and Courses:

Ensure the system can handle the complexity of managing timetables for multiple departments, courses, and academic programs simultaneously.

By achieving these objectives, the timetable management system will significantly improve the efficiency and effectiveness of academic scheduling, leading to a more organized and productive educational environment.

1.4. Problem Statement

1.4.1. Problem:

Class Conflicts and Class Mismanagement:

The current manual timetable system causes conflicts such as class mismanagement, teacher mismanagement, inconvenience, and disruptions to the teaching and learning process. Timetables may not be optimized for efficient class allocation, leading to underutilization of resources and inefficiencies in scheduling.

1.5. Proposed Solution

Solution:

Reducing Class, Conflicts:

Utilizing advanced algorithms to generate conflict-free timetables automatically. Employing constraint-based scheduling techniques to ensure classes are scheduled without overlapping time slots. Implementing conflict detection mechanisms to identify and resolve potential conflicts during the timetable generation.

Improving Class Management:

Prioritizing efficient allocation of classes, teachers, and resources based on predefined criteria. Incorporating features for administrators to easily manage and optimize class schedules, such as adjusting class durations and reallocating resources. Providing insights and analytics tools to help administrators analyze and optimize class distribution, teacher workload, and resource utilization.

1.6. Significance of Project

The significance of this project lies in its ability to streamline and optimize the timetable management process within educational institutions. By addressing common issues such as class conflicts and mismanagement, accessing the other department timetable, and unavailability of teachers, the proposed automatic timetable management system(ATTMS) will enhance efficiency, improve resource utilization, and create a more conducive learning environment with user-friendly interface for students and educators alike.

Proposed system	Existing Manual System
Improving class management	Class conflict
Less time to be consumed	Time-consuming

Data from SIS	Manual data entry
Flexibility	Lack of flexibility

Table 1: proposed system and existing system

1.7. Functionalities

2. By automating the process, the system minimizes the risk of errors and conflicts, resulting in a more streamlined and organized timetable. Such that the Automatic Timetable Management System offers a user-friendly interface that is accessible to both students and faculty members. It includes conflict detection and resolution features to identify scheduling conflicts, such as overlapping classes or double-booked rooms, accessing other department classes with other department teachers and offers suggestions for resolving them. User authentication and role management ensure secure access, with varying levels of permissions for different roles like administrators and faculty. A centralized database will store all relevant information, including course schedules, faculty availability, room assignments, and student enrollments. The system will efficiently allocate resources such as classrooms and laboratories based on availability and specific requirements. Additionally, it will offer options for exporting and printing timetables in various formats like PDF for offline access. The system will also support multiple academic programs, departments, and course structures, handling their complexity seamlessly. Furthermore, integration with existing systems such as Student Information Systems (SIS) or Learning Management Systems (LMS) will improve data flow and overall efficiency. This comprehensive solution will enhance academic scheduling, improving efficiency, accuracy, and user satisfaction.

2.1. ATTMS Users

Coordinators are the users of this system they are building timetables manually and this system will help them to automatically generate timetables based on their department's requirements while the users of ATTMS are the members of specific origination they are the only type of users who are experts and have the right access of ATTMS.

2.2. User Roles and Responsibilities

The roles and responsibilities are to select each relevant data of class, teacher, timing slots of classes, and semester to generate a Timetable to be used for the department classes.

2.3. Security

The difficult task is to secure data which are fetched from the University of Turbat SIS for generating a timetable we can use the following steps to secure these data and other relevant things:

a) Authentication System:

Only authorized persons who is working at the University of Turbat must be allowed to access the system or Application.

b) Auto Logout:

The data should be protected by the logged-in person who is auto-logged out of the system.

2.4. Technologies and Hardware Devices

Printer:

A printer of any kind is used to print the timetable to be fixed on the notice board for the students to see the class's timings and teachers.



Chapter 2

LITERATURE REVIEW

2.1. Introduction

In the quickly changing field of educational administration, efficient scheduling of classes is essential to maintaining efficiency and making the most use of available resources. This chapter examines the research on Automatic Timetable Management Systems (ATMS) and how they might help with class mismanagement and conflict reduction. By examining existing research, methodologies, and case studies, this chapter aims to provide a comprehensive understanding of how ATTMS can enhance the efficiency and effectiveness of educational institutions. The purpose of this literature review is to understand the current state of TMS, identify common problems, explore various solutions implemented in different contexts, and analyze the benefits and limitations of these systems. This review will also highlight gaps in the existing research and propose potential areas for further investigation.

2.2. Theoretical Review

This section explains the theories and models behind Automatic Timetable Management Systems (ATMS). By looking at these theories, we can better understand how ATTMS works and why it is effective.

2.2.1 Systems Theory

Systems theory, which views an organization as a complex set of interrelated and interdependent parts, provides a foundational framework for understanding AATMS. In the context of educational institutions, systems theory suggests that a timetable is a subsystem that interacts with other subsystems, such as curriculum, faculty, and facilities management. A well-designed ATMS ensures that these interactions are efficient and harmonious, reducing conflicts and optimizing resource use.

- **Application to ATTMS:** Systems theory emphasizes the importance of considering the entire system when designing an ATMS. This means ensuring that the ATMS integrates seamlessly with other institutional systems and processes, thereby facilitating smooth operations and decision-making.

2.2.2 Scheduling Theory

Scheduling theory focuses on the allocation of resources over time to perform a collection of tasks. This theory is crucial for understanding the algorithms and methods used in ATTMS to generate optimal timetables.

- **Job Shop Scheduling:** In educational institutions, scheduling can be likened to job shop scheduling where different classes (jobs) require specific resources (classrooms, instructors) at different times. ATMS uses algorithms from scheduling theory to allocate these resources efficiently.

- **Constraint Satisfaction Problems (CSP):** Many scheduling problems can be modeled as CSPs, where the goal is to find a solution that satisfies a set of constraints (e.g., no two classes can be in the same room at the same time). ATTMS employs various CSP-solving techniques to ensure that generated timetables meet all necessary constraints.

2.2.3 Decision Support Systems (DSS)

Decision Support Systems are interactive software-based systems intended to help decision-makers use data and models to solve unstructured problems. ATTMS can be viewed as a specialized form of DSS that assists administrators in making informed scheduling decisions.

- **Components of DSS in ATTMS:**
 - **Database Management:** Storing and managing data related to courses, instructors, rooms, and schedules.
 - **Model Management:** Algorithms and models used to generate and optimize timetables.
 - **User Interface:** Tools and interfaces that allow users to interact with the system, input data, and retrieve schedules.

2.2.4 Human-Computer Interaction (HCI)

Human-Computer Interaction (HCI) theory studies how people interact with computers and to design technologies that let humans interact with computers in novel ways. Effective TMS must consider HCI principles to ensure that the system is user-friendly and meets the needs of its users.

- **Usability:** A key aspect of HCI in TMS is ensuring that the system is easy to use, with intuitive navigation and clear instructions.
- **Accessibility:** TMS should be accessible to all users, including those with disabilities, ensuring that the interface is designed according to accessibility standards.
- **Feedback:** Providing immediate and clear feedback to users regarding their actions within the system is essential for maintaining usability and user satisfaction.

2.2.5 Educational Theory

Educational theories provide insights into the pedagogical requirements that a TTMS must support. These theories ensure that the ATTMS aligns with the educational goals and teaching methodologies of the institution.

- **Constructivist Theory:** Emphasizes the importance of active, student-centered learning environments. ATTMS should facilitate the creation of timetables that support such

environments, ensuring appropriate allocation of time for interactive and collaborative learning activities.

- **Behaviorist Theory:** Focuses on observable behaviors and how they can be shaped by the environment. TMS should enable schedules that reinforce positive learning behaviors, such as regular study periods and balanced workloads.

2.2.6 Operations Research

Operations research (OR) involves the application of analytical methods to help make better decisions. ATTMS heavily relies on OR techniques for optimizing schedules, managing resources, and solving complex scheduling problems.

- **Linear Programming:** Used in TMS for optimizing resource allocation while meeting a set of linear constraints.
- **Integer Programming:** Often applied in TMS where decisions are discrete (e.g., assigning a specific class to a specific time slot).
- **Simulation:** Used to model and analyze the performance of different scheduling scenarios, helping to identify the most effective timetable configurations.

2.2.7 Data Analytics and Machine Learning

The integration of data analytics and machine learning in ATTMS represents a growing area of research and application. These technologies can enhance the functionality and effectiveness of TMS by providing predictive insights and adaptive learning capabilities.

- **Predictive Analytics:** Analyzing historical data to predict future scheduling needs and trends, helping institutions to proactively manage resources.
- **Machine Learning Algorithms:** These can be used to improve the accuracy and efficiency of scheduling algorithms by learning from past scheduling outcomes and continuously optimizing the process.

2.3 Conclusion

The theoretical review highlights the multifaceted nature of Timetable Management Systems, drawing from diverse fields such as systems theory, scheduling theory, decision support systems, human-computer interaction, change management, educational theory, operations research, and data analytics. By understanding these theoretical frameworks, we can better appreciate the complexities involved in designing and implementing effective TMS solutions. This knowledge forms a foundation for the practical aspects of TMS development and implementation, which will be explored in subsequent chapters.

2.3. Review of Previous Projects

This section reviews previous projects related to Timetable Management Systems (TMS). By examining these projects, we can identify successful strategies, common challenges, and innovative solutions that have been implemented. This review will cover two significant projects that have contributed to the field of TMS.

Project 1 review

Project 1 Review: Ijrasnet Journal for Research in Applied Science and Engineering Technology

Automatic Timetable Generator

Authors: Prof. Jyothi Patil, Shambhavi V, Sneha N T, Sweta Jadhav, Tahura Sadaf

DOI Link: <https://doi.org/10.22214/ijrasnet.2023.53821>

Certificate: [View Certificate](#)

2.3.1 Introduction

Every semester, colleges are required to create timetables, which used to be an extremely time-consuming task. The "Time Table Generator" project is created with HTML and CSS for the front end and Python and MySQL 3 for the back end. This method for creating timetables connects with numerous modules and processing. The most crucial area for college efficiency is frequently automation and control. The use of information technology for quicker and simpler forms of communication is widespread. The following information was added by the administrator to Add the student, the staff, and the subject, enter the timetable, and update the timetable. Staff and students can see the details of the timetable. Python is being used to implement this system for the timetable generator. Since HTML & CSS make up our front end, the

online application has a far more effective and secure appearance. A project for a very helpful for Students to read the timetable details on this website is the timetable generator system.

2.3.2 Objectives

The major objective of a college timetable generator is to create an efficient and effective schedule for classes and other activities in a college or university. This objective is achieved by taking into account various constraints and criteria, such as

1. Resource availability,
2. Student enrolment,
3. Faculty preferences and availability,
4. Optimization criteria.

2.3.3 Methodology

The software can simply be compared to a schedule computing tool that accepts simple data sets as input and produces organized results. Using the straightforward input-process model We fill out the form with all the information, and a genetic algorithm applies it to the data on the backend to create class schedules.

2.3.4 Features

1. Unlike the manual timetabling system, the system offers flexibility.
2. It utilizes minimal processing/computing power.
3. It greatly reduces the time needed to generate maximum error-free timetables.
4. It provides an easy means for data entry and revision through an intuitive interface.
5. It increases productivity.
6. Timetables generated are between 60% – 80% best solution.
7. It almost eliminates paperwork.
8. It simplifies the timetabling process.
9. Gives accurate information.
10. Simplifies the manual work.

2.3.5 Future Scope

This project will be very beneficial to the university because managing numerous faculties and assigning courses to them simultaneously by hand is a very challenging task that this project will assist in managing effectively. This faculty timetable can be readily controlled while taking into account the maximum and lowest workload. The faculty data in the database can also be used to keep track of the faculty's expertise in specific fields. Attribute The accuracy of the project will allow for a more corrective approach to the creation of this schedule. This project will produce output that is mostly corrective and error-free. The project's potential future improvement is the creation of a master schedule for the departments and the entire college. Further adjustments can be made while maintaining the project's approach and methods to accomplish this improvement. Additionally, it can be utilised to assign a certain time slot that the instructor prefers. The university website may incorporate this timetable maker, making it more useful. The implementation of a time table management system can make it simpler for the schools to assign a teacher to a class in the event of an absent teacher.

2.3.6 Reference

[1] K. Nguyen, D. Nguyen, K. Trieu, and N. Tran, "Automating a real-world university timetabling problem with Tabu search algorithm," in 2010 IEEE RIVF International Conference on Computing & Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), 2010 [2] A. Azlan and N. M. Hussin, "Implementing graph coloring heuristic in construction phase of curriculum-based course timetabling problem," in 2013 IEEE Symposium on Computers & Informatics (ISCI), 2013. [3] R. E. Febrita and W. F. Mahmudy, "Modified genetic algorithm for high school time-table scheduling with fuzzy time window," in 2017 International Conference on Sustainable Information Engineering and Technology (SIET), 2017. [4] T. Elsaka, "Autonomous generation of conflict-free examination timetable using constraint satisfaction modelling," in 2017 International Artificial Intelligence and Data Processing Symposium (IDAP), 2017. [5] F. D. Wihartiko, H. Wijayanti, and F. Virgantari, "Performance comparison of genetic algorithms and particle swarm optimization for model integer programming bus timetabling problem," IOP Conf. Ser. Mater. Sci. Eng., vol. 332, p. 012020, 2018.

Paper Id : IJRASET53821

Publish Date : 2023-06-07

ISSN : 2321-9653

Publisher Name : IJRASET

Project 2 Review

Automated Time Table Generator A Project

Presented to the faculty of the Department of Computer Science

Umma University, Kajiado

Submitted in partial satisfaction of the requirements for the degree of Bachelor of Science In
Computer Science By

Abdulhamid Abdirahim Mohamed November 2022 BSCS/2020/45478

2.3.1 Introduction

The lecturing staff usually spends a lot of time in examination timetable generation and timetable management. The software program to be design for this project work will captures all parameters used in creating a university examinations timetable and automatically creates one with its timetable generation tool. This university examination timetable generation software also considers the availability of courses and other resources while creating levels in the department. Moreover, changes can be easily made in the timetable as and when necessary depending on the availability of courses. Since the inception of the computer science department, the examination time table schedule has been done manually. Relying on manual schedule processes a lot of disadvantages. In order to overcome some setbacks in the method used by the department this project design is brought up, titled Automated time table generator. This project is to develop a program that will allow the department to provide a timely and accurate schedule timetable in a form of program. Automated timetable generator is to assist the department on deciding the examination room for each course without conflict at a specific time in a day. Moreover, the program will also assign a unique examination to each course but it will ensure that the examination is not in another class at that specific period.

2.3.2 Objectives

- (i)To automate examination timetable generation for Umma university.
- (ii)To reduce the time interval used in setting examination time table.
- (iii)To enable computer science department to dynamically generate timetable for student to access directly from the schedule table data.
- (iv)To enables the department to present changes to their schedule immediately to save on time.
- (v)To enables the computer science department to plan well in advance and prepare exam schedules for students

2.3.3 Methodology

The method of sources of data collection, the evaluation of current system and the organization structure of the current system were presented. It includes specific methods that I used to in order to achieve specific objectives, particular requirements of the automated examination timetable generator system implementation and a brief explanation of why specific methods was used to implement the system.

Prototyping

A prototyping methodology is a software development process which allows developers to create portions of the solution to demonstrate functionality and make needed refinements before developing the final solution. The advantages of prototyping include: Reduced time and costs Prototyping can improve the quality of requirements and specifications provided to developers. Because changes cost exponentially more to implement as they are detected later in development, the early determination of what the user really wants can result in faster and less expensive software (Adiono et. al, 2016).

Requirements gathering and analysis

A prototyping model starts with requirement analysis. In this phase, the requirements of the system are defined in detail. During the process, the users of the system are interviewed to know what is their expectation from the system.

Quick design

The second phase is a preliminary design or a quick design. In this stage, a simple design of the system is created. However, it is not a complete design. It gives a brief idea of the system to the user. The quick design helps in developing the prototype.

Build a Prototype

In this phase, an actual prototype is designed based on the information gathered from quick design. It is a small working model of the required system.

Rapid application development

It's a software development methodology that uses minimal planning in favor of rapid prototyping. A prototype is a working model that is functionally equivalent to a component of the product. In RAD model the functional modules are developed in parallel as prototypes and are integrated to make the complete product for faster product delivery. The advantages of using RAD is that the time required to develop the software is drastically reduced due to a reduced requirement analysis business requirements documentation and software requirement specification) and planning stage (Sasmito et. al, 2020). All the software prototypes produced can be kept in a repository for future use. The reusability of the components also enhances the speediness of the process of software development. It was much easier for a project manager to be accurate in estimating project costs which of course means that project cost controls are easier to implement and manage as well.

2.3.4 Features

Completely automated.

This system is able to generate time tables in completely automated way which will save a lot of time and effort of an institute administration.

Interface capabilities

The system will be having an easy to use and interactive interface to enter all the inputs.

Processing Capabilities

The system will have algorithms to process all the data present in the database and keeping in view the various constraints like that a teacher should not have two consecutive lectures/labs, students have minimum one-hour gaps, proper rooms are allocated for the lectures and tutorials, labs are used optimally so that they are used for the maximum possible time, it will generate the time table.

This project work is greatly hoped to eliminate the manual way of scheduling examination time table in the university. It will more also eliminate stress in planning of the time table. It will also avail the students to go for their extra nonacademic activities without to worry date of exams. Students will now write exams comfortably with good venue and much time and more also eliminate examination clash.

2.3.5 Outcomes

The implementation of the ESS led to several positive outcomes:

- **Efficiency Gains:** Significant reduction in time and effort required for manual scheduling.
- **Compliance:** Improved compliance with labor laws and company policies.
- **Employee Satisfaction:** Higher satisfaction levels among employees due to fair and transparent scheduling practices.
- **Operational Efficiency:** Enhanced operational efficiency and productivity.

2.3.6 Challenges

The project faced several challenges, including:

- It is large and highly constrained, but above all the problem differs greatly for diverse colleges and learning institutions. It is hard to write a universal agenda, fitting for all possible timetable problems. Even though manual creation of timetable is sustained, it is still universal, because of the lack of suitable computer programs (Gore & Poonam Sonawane, March 2017).
- There exist a lot of diverse timetable problems such as: - University Timetable, Exam Timetable, School Timetable, Sports Timetable, Worker Timetable Moreover, there exist a lot of problem- solving methods, which typically use the concept of customary optimization algorithms such as genetic algorithms, Backtracking, Constraint Logic Programming.

- Ensuring all employees were trained to use the new system effectively.

2.3.7 Lessons Learned

The key lessons learned from the SanFrancisco Corporation project include:

- **Thorough Planning:** Detailed planning and clear objectives are essential for the success of large-scale projects.
- **Stakeholder Involvement:** Active involvement of all stakeholders throughout the project lifecycle enhances acceptance and success.
- **Continuous Improvement:** Regular feedback and continuous improvement are vital for refining the system and addressing any issues promptly.

2.4 Conclusion

Automatic Timetable manger is a Java based software used to generate timetable automatically. Will help you to manage all the periods automatically. Proposed system will help to generate it automatically also helps to save time. There is no need for Faculty to worry about their period details and maximum workload. It is a comprehensive timetable management solution for Colleges which helps to overcome the challenges in current system. The final system should able to generate time tables in completely automated way which will save a lot of time and effort of an institute administration. To make a timetable system generic so that I can work equally well for different School, Colleges and Universities. User defined constraints handling. Ease of use for user of system so that he/she can make automatic time table.

Chapter 3

System Analysis

3.1 Introduction:

Our target users are limited in number and there is a lot of data available regarding timetable management system already. Basic requirements can be obtained by analyzing the manual timetable management system as the proposed automatic timetable management system is intended to be a replica of the timetable management system. However, a survey has been launched to collect new requirements and to verify the basic requirements obtained from the manual timetable management system as below.

Officials were asked to give importance scores ranging from 1 to 5 (5 the most important). Percent scores were obtained for all requirements stated above. The percent score was calculated, and the most important requirements were highlighted. After analyzing the requirements functional requirements are specified and a few use cases and activities are presented below:

3.2. System Requirements

3.2.1. Functional Requirements

The functional requirements are the needs of the user that a system should provide. Those needs can be generating a timetable and viewing those generated timetables.

3.2.2 Non-Functional Requirements

Non-functional requirements are system requirements that what a system shall do. The system should be able to provide the facilities to justify the needs of the system. The non-functional attributes of the system are described below:

1. **Availability:** The system will be available to all users
2. **Security:** security is very important in the Timetable management system because the system is accessing every piece of information from the SIS of the university and these data need to be secure.
3. **Maintainability:** The timetable management system has been designed in such a way that it can be maintained easily in the future.
4. **Usability:** The user interface has been designed simple enough that the organization members can use it without having any difficulty
5. **Scalability:** the system has been designed in a way that it can be extended in the future.
6. **Cost Effective:** as the system is client-independent and is installed at server only therefore it saves enhance cost of installation on the client's machine

3.2.3 Logical Database

As our Data are already stored in the SIS we just need to retrieve them. Although we need to keep storing our classes' data locally they should be used.

3.3 Data collection

Data collection is the process of gathering and measuring information on targeted variables in an established systematic fashion, which then enables one to answer relevant questions and evaluate outcomes.

3.3.1 Primary Source:

Primary data means original data that has been collected specially for the purpose of building the project "Automatic timetable management system". That is the data collected from the source firsthand. Since the data has not been published yet the data is more reliable authentic and objective. And the data included is valid

➤ Questioner

I observed that the people who need to use the automatic timetable management system to overcome the conflict issue are not aware and thus use their manual process to work. if it is built then the people who use it will definitely come with positive feedback

➤ Interview

People in the organization have been asked how they work on timetable managing and from the interview my team found how difficult to arrange the timetable of each semester plus the teachers from another department when teaching other departments.

3.3.2 Secondary Source

We collected other secondary information like the analysis of the system, and system design from the following sources:

➤ Software Engineering by Lan Somerville

3.4 Data presentation

Data presentation is the method by which people summarize, organize, and communicate information using a variety of tools, such as diagrams, distribution charts, histograms, and graphs

3.4.1 System model

A system model is a conceptual model because of system modeling that describes and represents a system. A system comprises multiple views such as planning, requirement (analysis), design, implementation, deployment, structure, behavior, input data, and output data views. A system model is required to describe and represent all these multiple views.

3.4.2 Architecture model

An architectural model is a type of scale model –a physical representation of a structure – build to study aspects of an architectural design or to communicate design ideas.

3.4.3 UML

Unified Modeling Language is a standardized modeling language enabling developers to specify, visualize, construct and document artifacts of a software system. Thus, UML makes these artifacts scalable, secure and robust in execution. UML is an important aspect involved in object-oriented software development, it uses graphic notation to create visual models of software systems

Chapter 4

System Design

4.1. Introduction

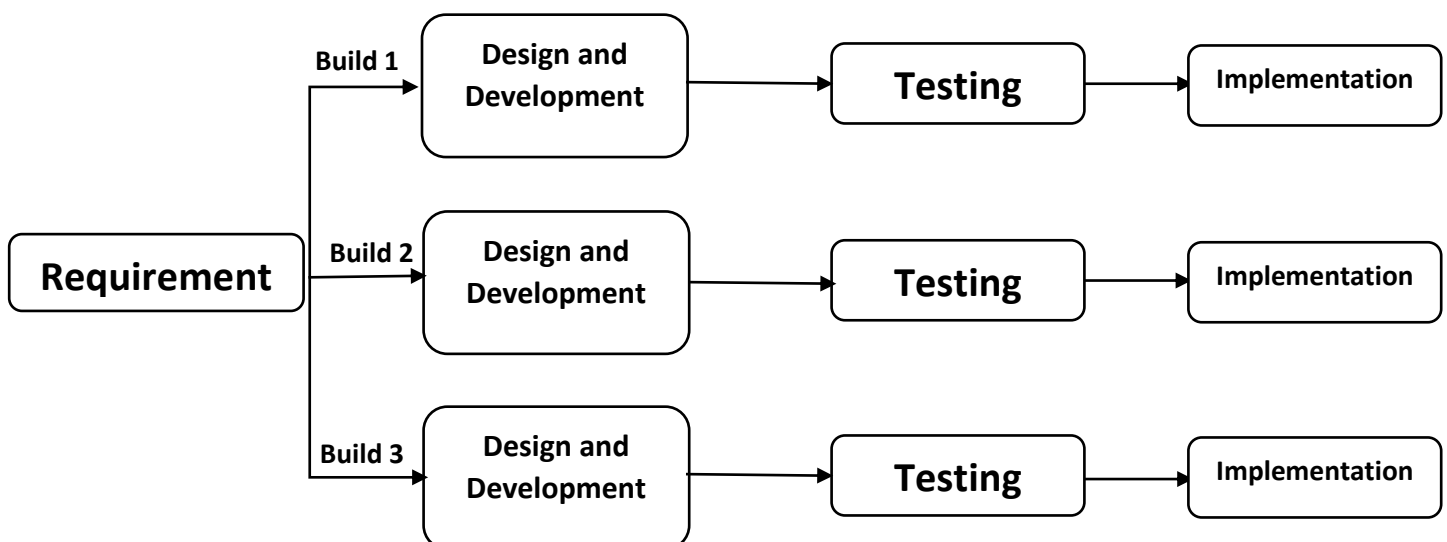
As we have included modeling the proposed Automatic timetable management system using the Unified Modelling Language (UML). Short for Unified Modeling Language is a standardized modeling language consisting of an integrated set of diagrams, developed to help system and software developers for specifying, visualizing, constructing, and documenting the artifacts of software systems, The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software. The requirement specifications from the system design are studied in this design phase and system design is prepared. System Design helps in specifying hardware and system requirements and also helps in defining overall system architecture. The software code to be written in the next stage is created now.

4.2. Process model

4.2.1. Incremental model

For project life cycle we used Incremental Model. The main essence of incremental model is known as builds. The incremental build model is a method of software development where the product is designed, implemented, and tested incrementally (a little more is added each time) until the product is finished. One or more builds may overlap. Incremental model has following characteristics which are the foremost strength of this model:

- a) Portion of product becomes operational within weeks
- b) Change and adaptation are natural to incremental model.
- c) Phased delivery requires no large capital outlay; rapid return on investment.



4.3. System Architecture

The Automatic Timetable management system relies on a three-tiered architecture. In a three-tier architecture, the user interface, functional process logic, data storage, and data access are developed and maintained independently on a separate platform. The user interface runs on a workstation, and functional process logic runs on a workstation or an application server. The DBMS contains data storage logic. The 3 tier architecture consists of the following three tiers.

- a) Presentation Tier
- b) Application Tier/Logic Tier/Business Logic Tier
- c) Data Tier

The top-most layer is the presentation tier and it consists of user interfaces. The purpose of the second tier i.e. Logic Tier is to handle the movement and processing of data between the two surrounding tiers. The last tier i.e. Data tier consists of a database and deals with storage and retrieval of data, in our case we are just going to be retrieving the data only. The advantage of using a three-tiered architecture is that if system requirements or technology changes, any tier can be upgraded or replaced independently.

In the table management system, the presentation tier includes a workstation, which will be used for user interaction with the system, the application tier consists of a TMS server

The server will provide communication information and services. The data tier comprises of data management system.

The workstations will be connected to the TMS server via LAN (local area network).

Diagram needed

4.4 UML Diagram

4.4.1. Use case diagram

Use case diagrams to show a system from an outsider (e.g. User) perspective. use-case diagrams model the behavior of a system and help to capture the requirements of the system. Use-case diagrams describe the high-level functions and scope of a system. These diagrams also identify the interactions between the system and its actors. The system is treated as a black box and one solely identifies what the system is used for. The use case diagram for the ATTMS is depicted in Figure 2 figure 3

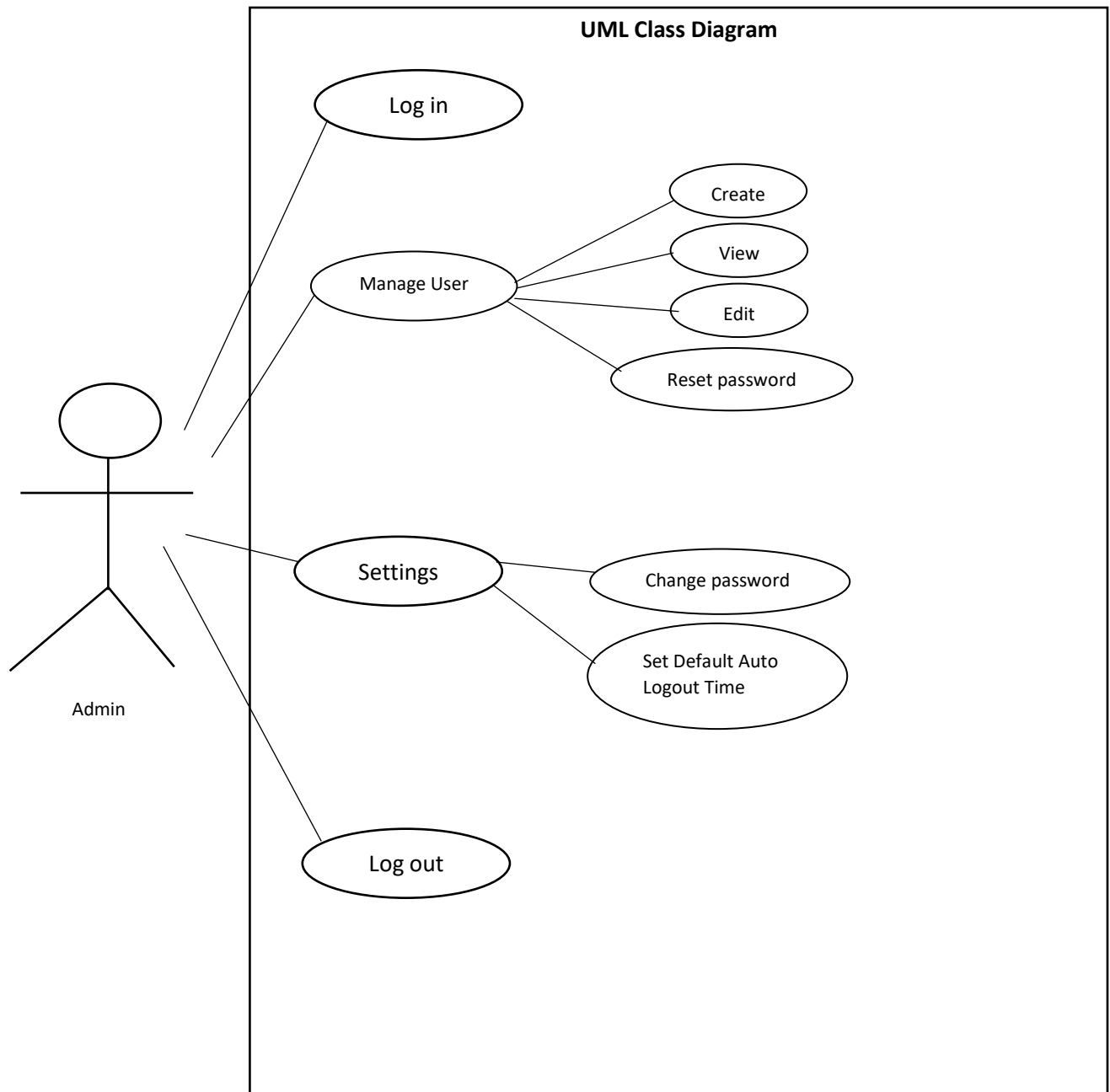


Figure 6 – Admin Use cases

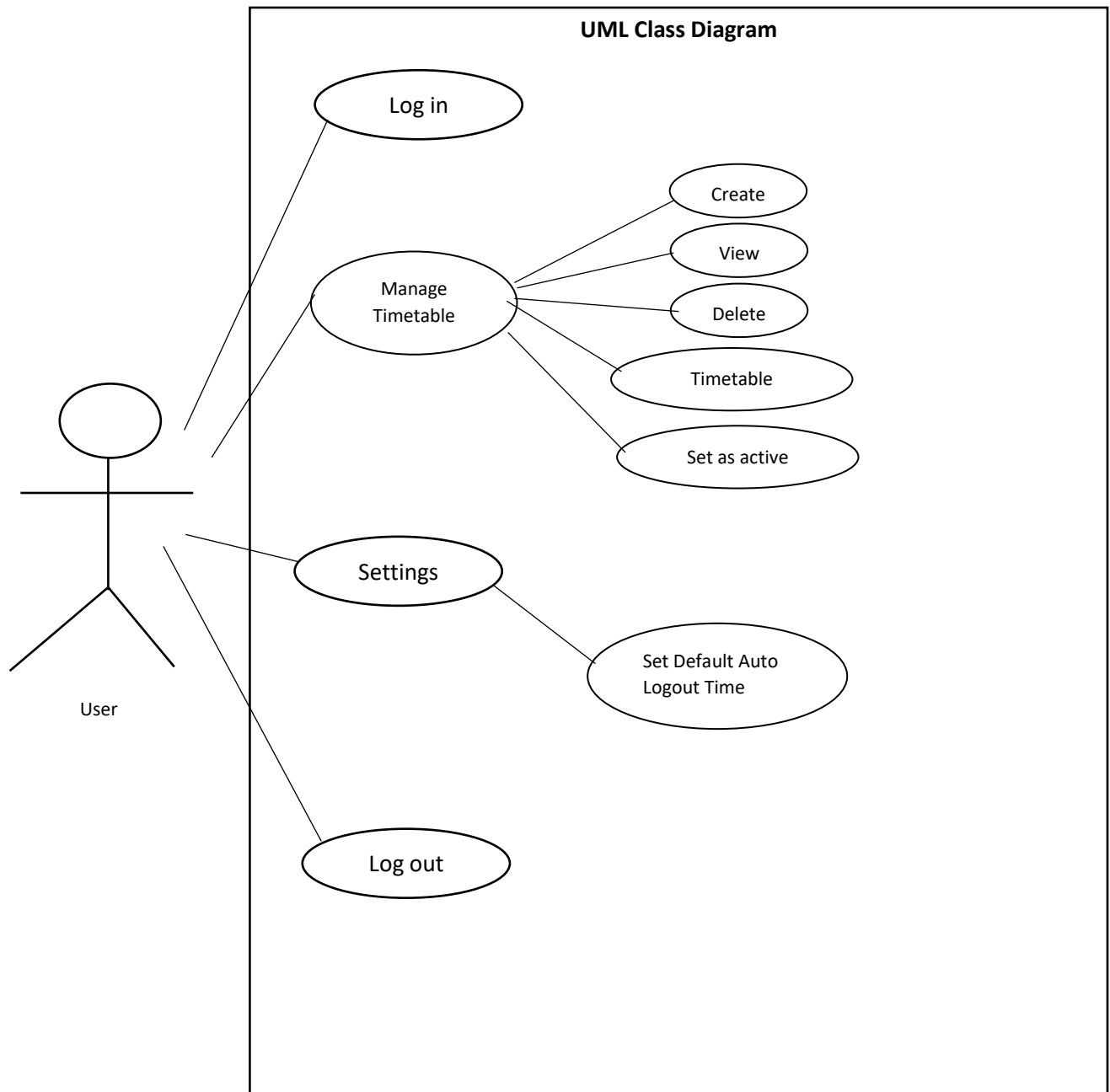


Figure 7 – User use cases

4.5 Use cases

4.5.1 Use case No# 1 Login

Use Case ID	01
Use Case Name	Login
Actors	Admin
Description	Admin wants to login into system
Pre-Condition	Admin should have the login credentials
Post-Condition	Admin is successfully logged in into the system
Normal course of events	<ol style="list-style-type: none">1. Admin enters username and password into fields2. Admin clicks "Login" button3. Admin is successfully logged in to the system
Alternative flow	None
Exceptions	Admin enters invalid username and password system displays an error message

4.5.2. Use case No#2 create Users

Use Case ID	02
Use Case Name	Create users
Actors	Admin
Description	Admin wants to create a user into a system
Pre-Condition	Admin should click on the create a user button
Post-Condition	Admin is successfully created the user into system
Normal course of events	<ol style="list-style-type: none">1. Admin should have click on create a user button2. Admin should fill the fields related to the user3. Admin click submit button4. User is successfully registered into the system and notified via Gmail
Alternative flow	
Exceptions	

4.5.3. Use case No#3 view Users

Use Case ID	03
Use Case Name	View users
Actors	Admin
Description	Admin wants to view a user from a system
Pre-Condition	Admin should click on view user button
Post-Condition	User should be shown in the view user filed
Normal course of events	<ol style="list-style-type: none">1. Admin should have click on view user button2. Users are successfully shown
Alternative flow	None
Exceptions	None

4.5.4 Use case No#4 Delete Users

Use Case ID	04
Use Case Name	Delete Users
Actors	Admin
Description	Admin wants to Delete a user into a system
Pre-Condition	Admin should click on the view user and delete button
Post-Condition	Admin has successfully deleted a user
Normal course of events	<ol style="list-style-type: none">1. Admin should have click on View a user button and delete button2. Admin should be clicking yes button into the system dialog box3. Admin has successfully deleted a user from the system
Alternative flow	None
Exceptions	If admin clicks No button then the user will not be deleted

4.5.5. Use case No#5 edit Users

Use Case ID	05
Use Case Name	Edit user
Actors	Admin
Description	Admin wants to edit a user from a system
Pre-Condition	Admin should click on view and edit user button and update button
Post-Condition	Users data should be edited from the system
Normal course of events	<ol style="list-style-type: none">1. Admin should have click on view and edit user button2. Admin should have edited the data of user and clicked update button3. Users data are successfully edited
Alternative flow	None
Exceptions	None

4.5.6. Use case No#6 change password

Use Case ID	06
Use Case Name	Changed password
Actors	Admin
Description	Admin wants to change the password
Pre-Condition	<ol style="list-style-type: none">1. Admin should click on setting button2. Then click changed password
Post-Condition	Password has changed
Normal course of events	<ol style="list-style-type: none">1. Admin should have click on setting2. Admin should have click on changed password button3. Admin should enter the old password4. Admin should enter the new password5. Admin should click on Submit Button6. Password is successfully changed
Alternative flow	None
Exceptions	If old password is wrong then the error message is shown

4.5.7. Use case No#7 view active Timetables

Use Case ID	07
Use Case Name	View active Timetables
Actors	Admin
Description	Admin should view active Timetable in the system
Pre-Condition	Admin should open the system into pc
Post-Condition	All active timetable should be listed on the first screen
Normal course of events	<ol style="list-style-type: none">1. Admin should have opened the system2. Active timetable are listed on the screen3. Admin has successfully viewed all departments timetable
Alternative flow	None
Exceptions	None

4.5.8. Use case No#8 Logout

Use Case ID	08
Use Case Name	Log out
Actors	Admin
Description	Admin wants to logout into system
Pre-Condition	Admin should be logged into the system
Post-Condition	Admin is successfully logged out to the system
Normal course of events	<ol style="list-style-type: none">1. Admin wants to log out to the system2. Admin clicks on log out button3. On success Admin is successfully logged out to the system and redirects to login page
Alternative flow	Admin is automatically logged out from the system if automatic logging out functionality is active
Exceptions	None

4.5.9. Use case No#9 Users Login

Use Case ID	09
Use Case Name	Login
Actors	User
Description	User wants to login into system
Pre-Condition	User should have the login credentials given by the admin
Post-Condition	User is successfully logged in to the system
Normal course of events	<ol style="list-style-type: none">1. User enters username and password2. User clicks on log in button3. On success user is successfully logged in to the system
Alternative flow	None
Exceptions	User enters invalid username and password system shows error message

4.5.10. Use case No#9 Timetable creation

Use Case ID	10
Use Case Name	creating a timetable
Actors	User
Description	User wants to make a new timetable into system
Pre-Condition	User is successfully created a timetable into the system
Post-Condition	User is successfully logged in to the system
Normal course of events	<ol style="list-style-type: none">1. User clicks on the add new timetable button on the screen2. User is redirected to multiple pages for entering data into fields for creating a new timetable3. User clicks on generate timetable button4. And system automatically generate timetable related to provided details5. User successfully created a new timetable
Alternative flow	None
Exceptions	None

4.5.11. use case No#11 Users to View Timetable

Use Case ID	011
Use Case Name	Viewing timetable
Actors	User
Description	User wants to view a timetable into system
Pre-Condition	User should click on the view icon button
Post-Condition	User is successfully viewed the timetable
Normal course of events	<ol style="list-style-type: none">1. Users enter the view icon button by the side of the listed timetables2. User is successfully viewed the clicked timetable
Alternative flow	None
Exceptions	None

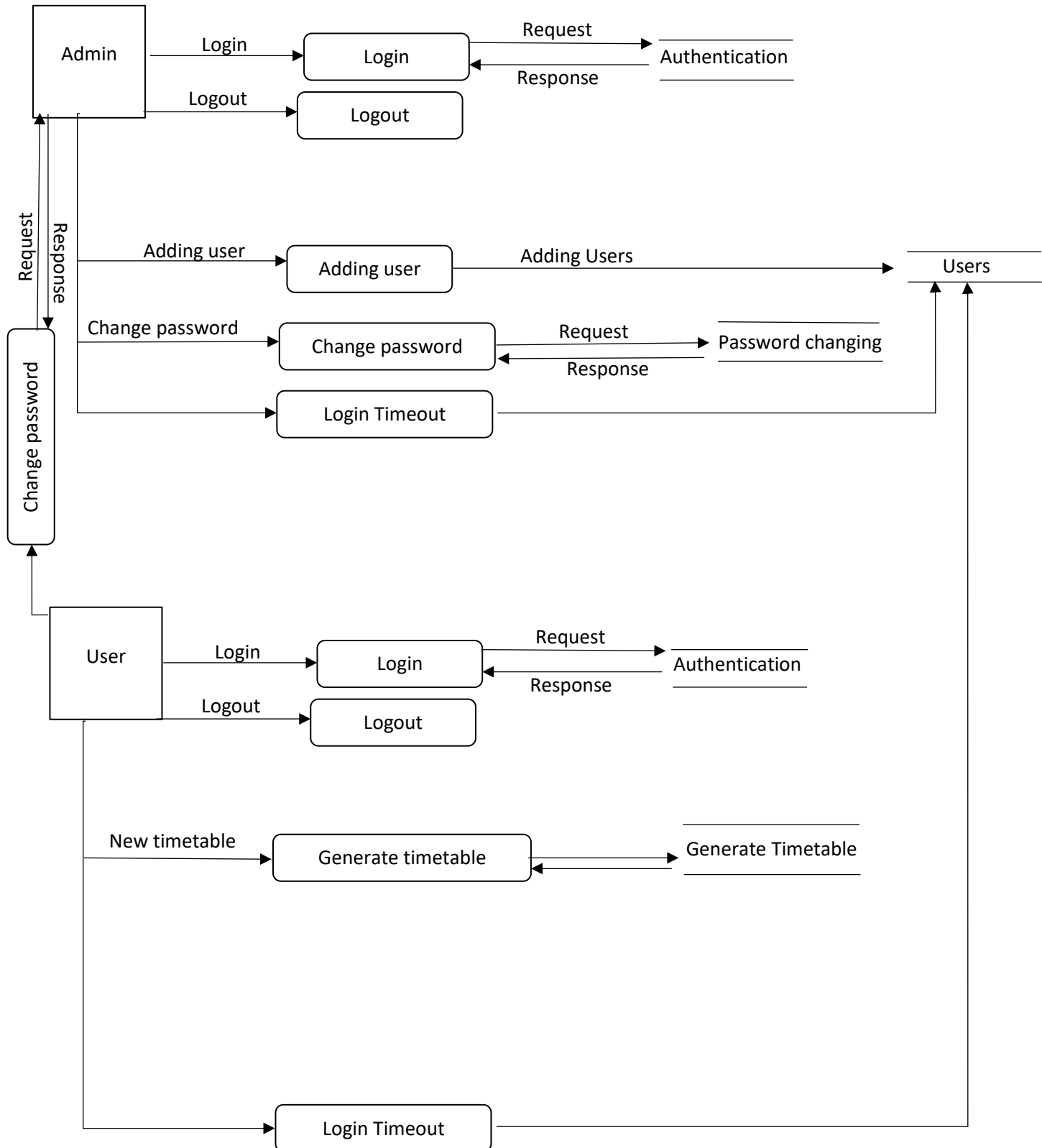
4.5.12. Use case No#12 Delete Timetable

Use Case ID	04
Use Case Name	Delete a timetable
Actors	User
Description	User wants to Delete a Timetable from a system
Pre-Condition	User should click on delete icon button
Post-Condition	User has successfully deleted a timetable
Normal course of events	<ol style="list-style-type: none">1. User clicks on the delete icon button2. User then clicks on the yes button on the dialoge box3. Timetable is successfully deleted from the sysem
Alternative flow	None
Exceptions	If user clicks on no button on dialoge box, the timetable is not deleted from the system.

4.5.13 Use case No#8 Logout

Use Case ID	08
Use Case Name	Log out
Actors	User
Description	User wants to logout into system
Pre-Condition	User should be logged into the system
Post-Condition	User is successfully logged out to the system
Normal course of events	<ol style="list-style-type: none">1. User wants to log out to the system2. User clicks on log out button3. On success user is successfully logged out to the system and redirects to login page
Alternative flow	User is automatically logged out from the system if automatic logging out functionality is active
Exceptions	None

Data Flow Diagram



4.6. User Interface

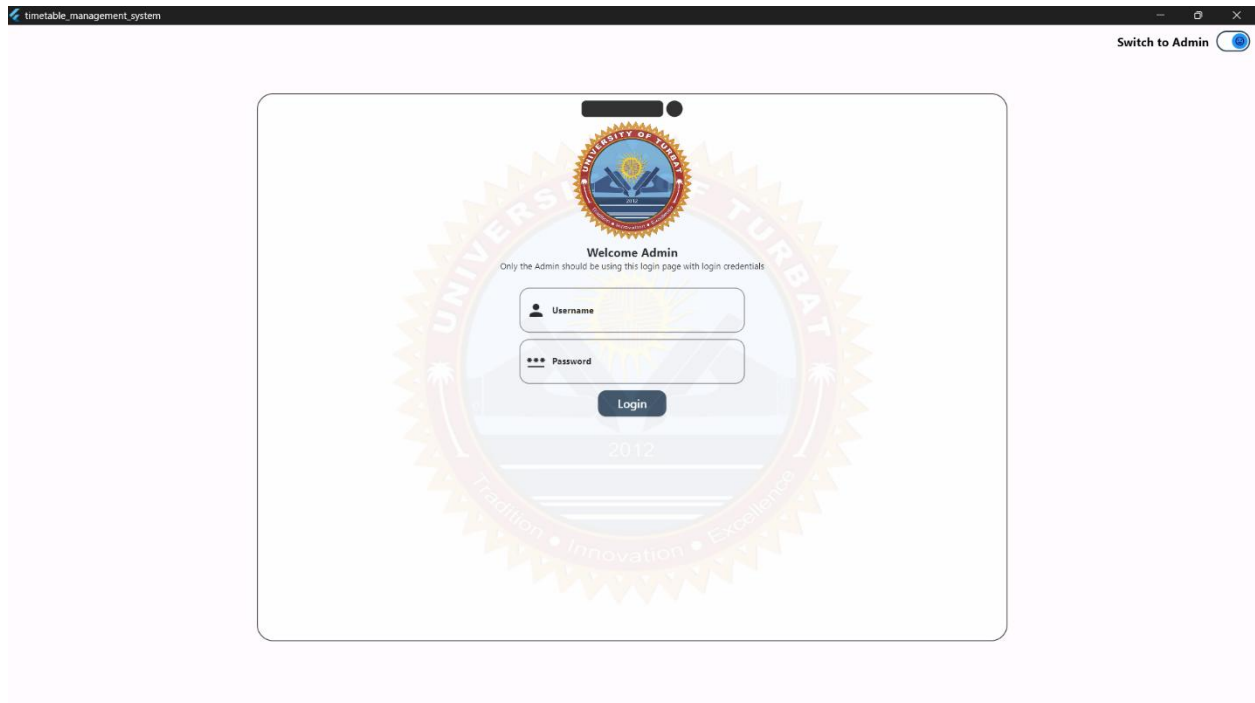
4.6.1. Introduction:

To design user-friendly interfaces, UI designers need to understand how people work and how each visual, interactive element shapes their experience, the user interface is the physical view of the system through which a user can interact with the system. Our focus to make the interfaced view friendlier and easier to understand. It includes the display pf the screen, the usage of input device the language and symbol usage etc. And how the system responds to the requests.

There are some screen shots of the user interface.

4.6.2. User Interface for Admin Login Page

The system can be accessed by double clicking on the executable file of ATTMS on windows Computer. The system requires admin credentials for access. Once logged in with these credentials, the admin is granted full control of the system .Which includes the ability to control coordinators, departments, classrooms, and subjects, and other administrative tasks.

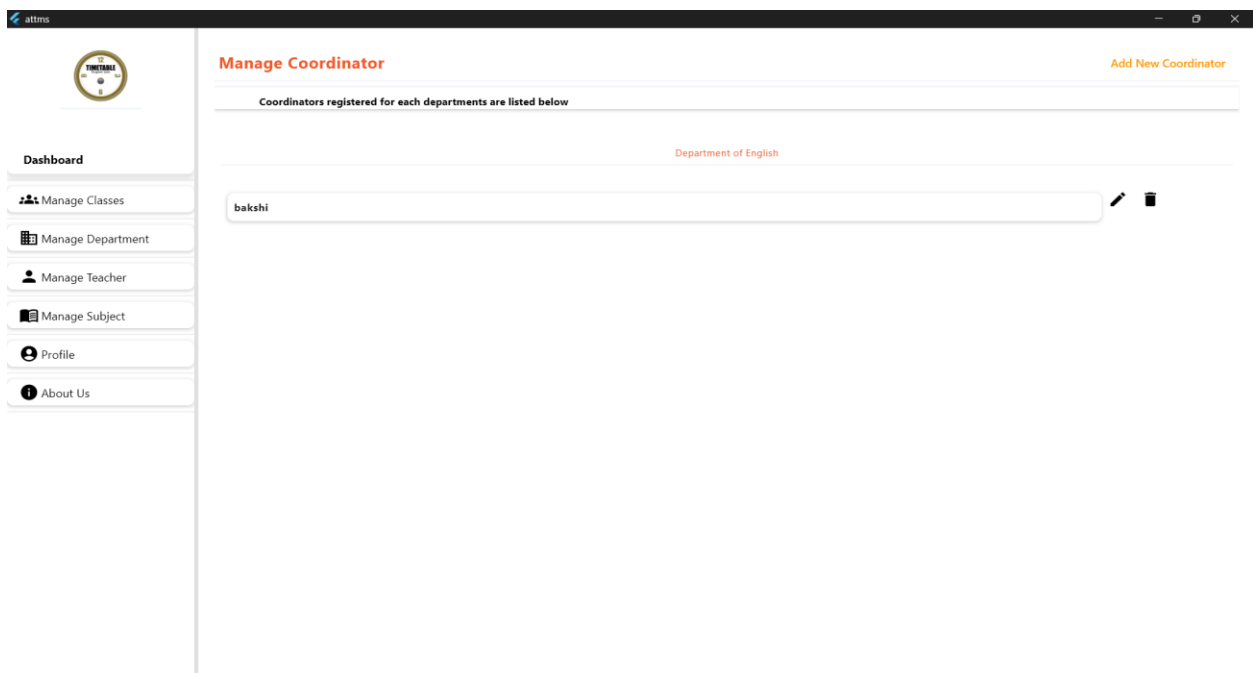


Dashboard

Admin credentials, the user can successfully access the Admin Dashboard, which serves as the The hub for managing all functionalities. And here all the dashboard system

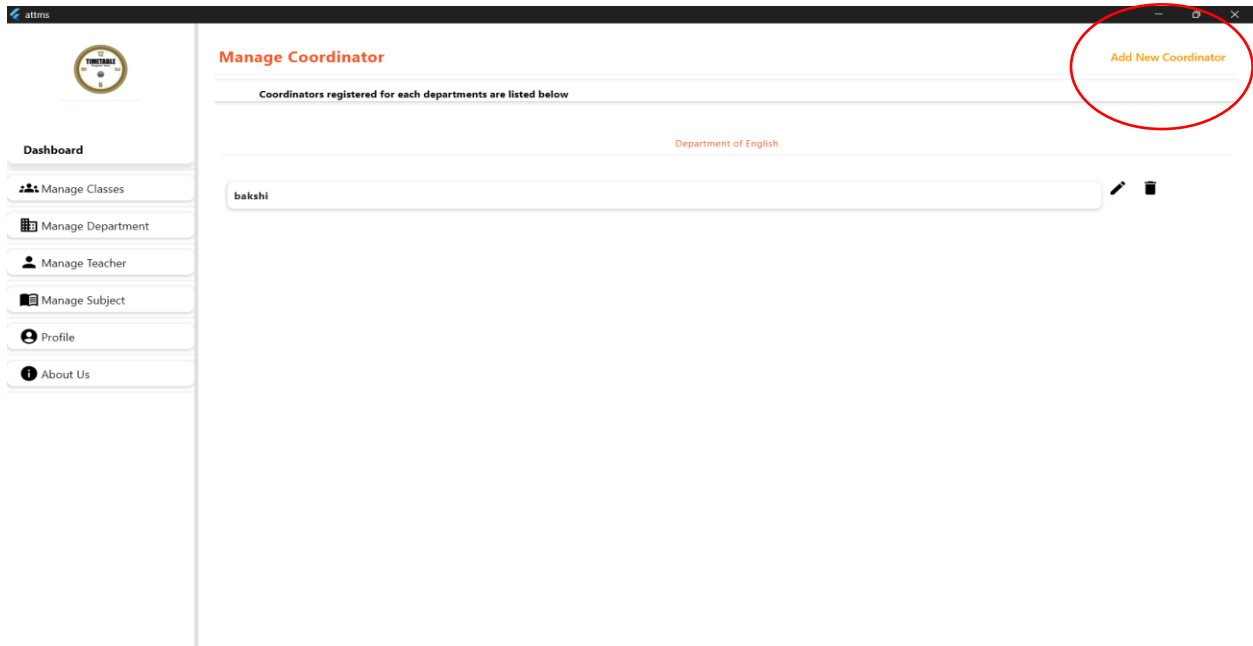
Managing Coordinators

The admin has control for managing coordinators. And the admin can update information for existing coordinators. If needed, coordinators can also be removed from the system. All coordinators are an organized within the system, making it easy for the admin to view and manage their profiles efficiently.



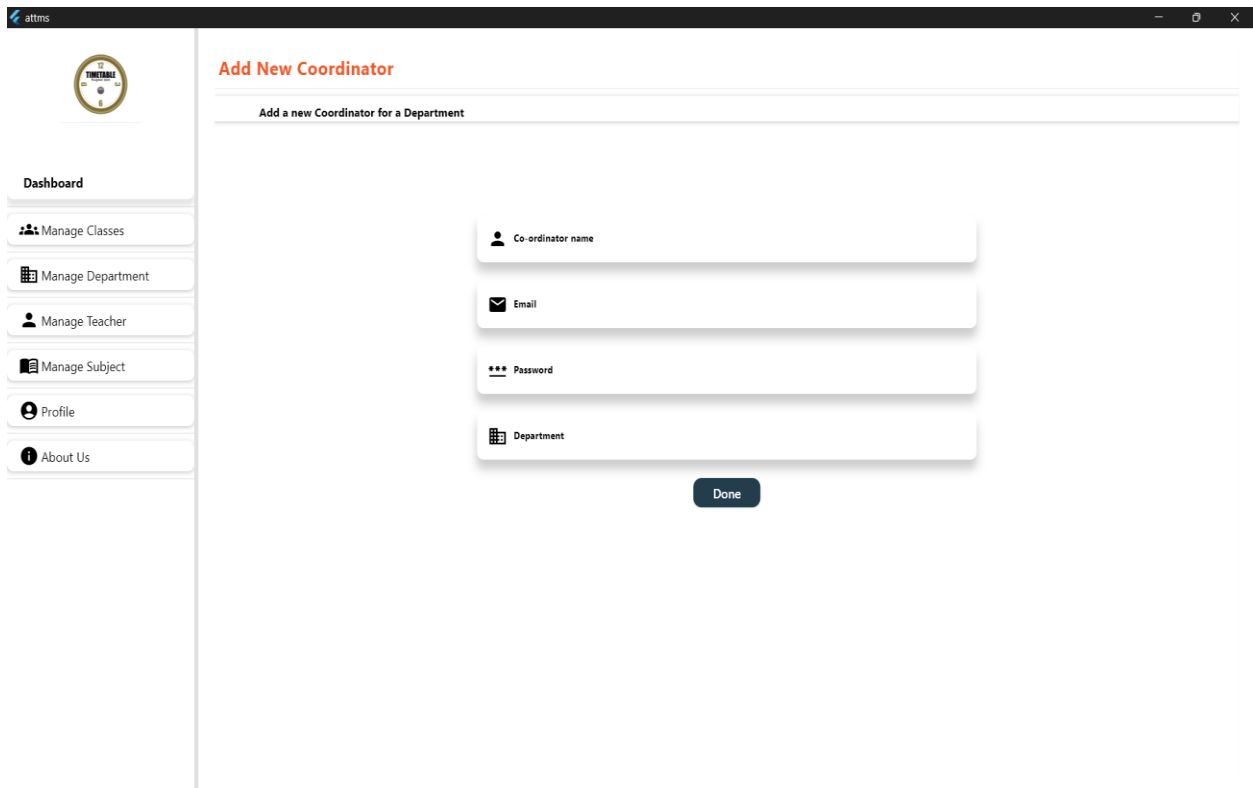
Adding a New Coordinator

The admin can easily add a new coordinator through Manage Coordinator.



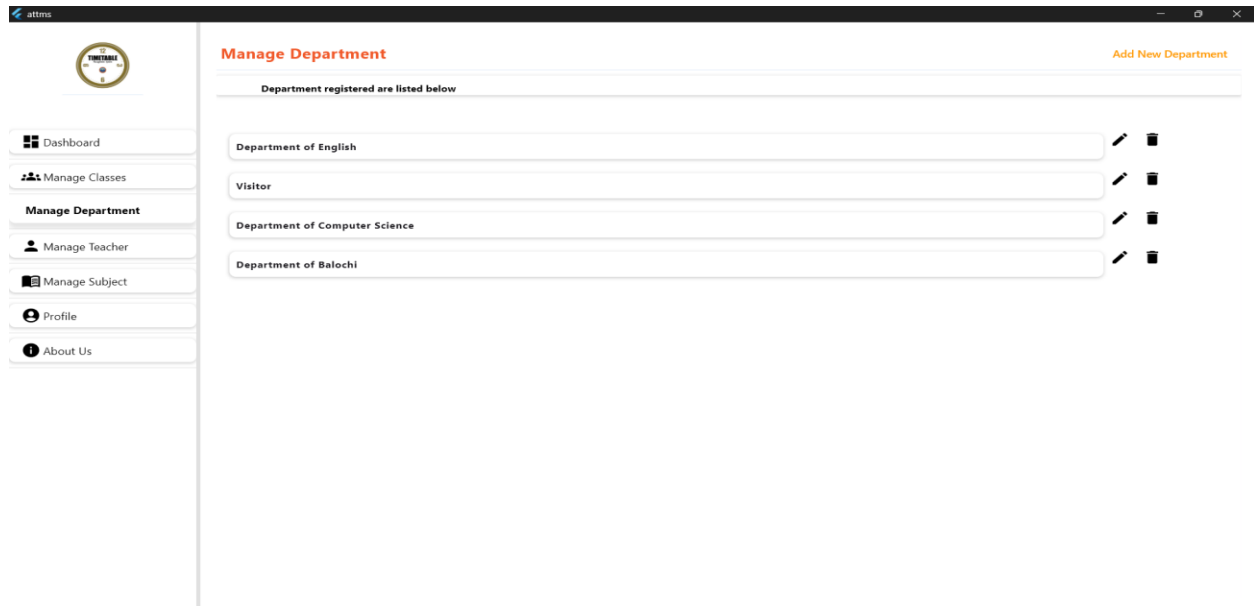
Adding a New Coordinator

After clicking Button of add new coordinator the admin will be able to add a coordinator and This process needs to enter coordinator details, like the coordinator's name, Email, department, Password. Once the information is submitted, the new coordinator is successfully added to the system.



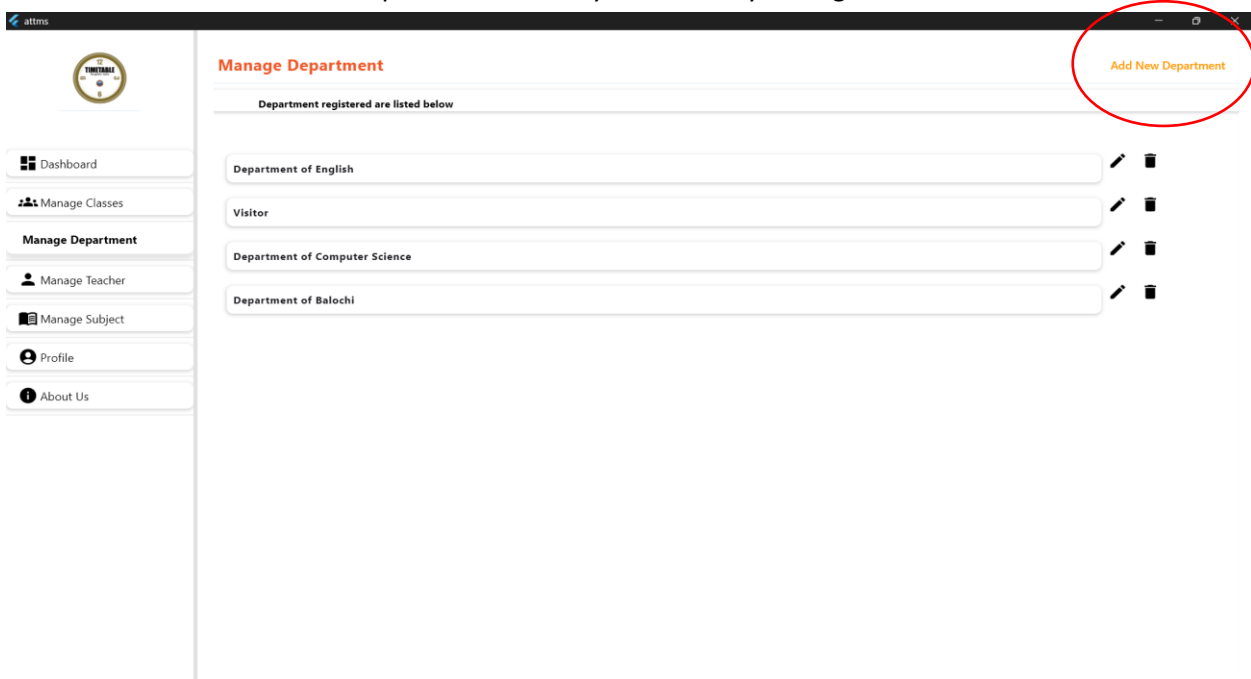
Managing Departments

The admin can manage all departments within the system. The list of existing departments and The admin can also make modifications to department information. And if any departments that are no longer required can be removed from the system.



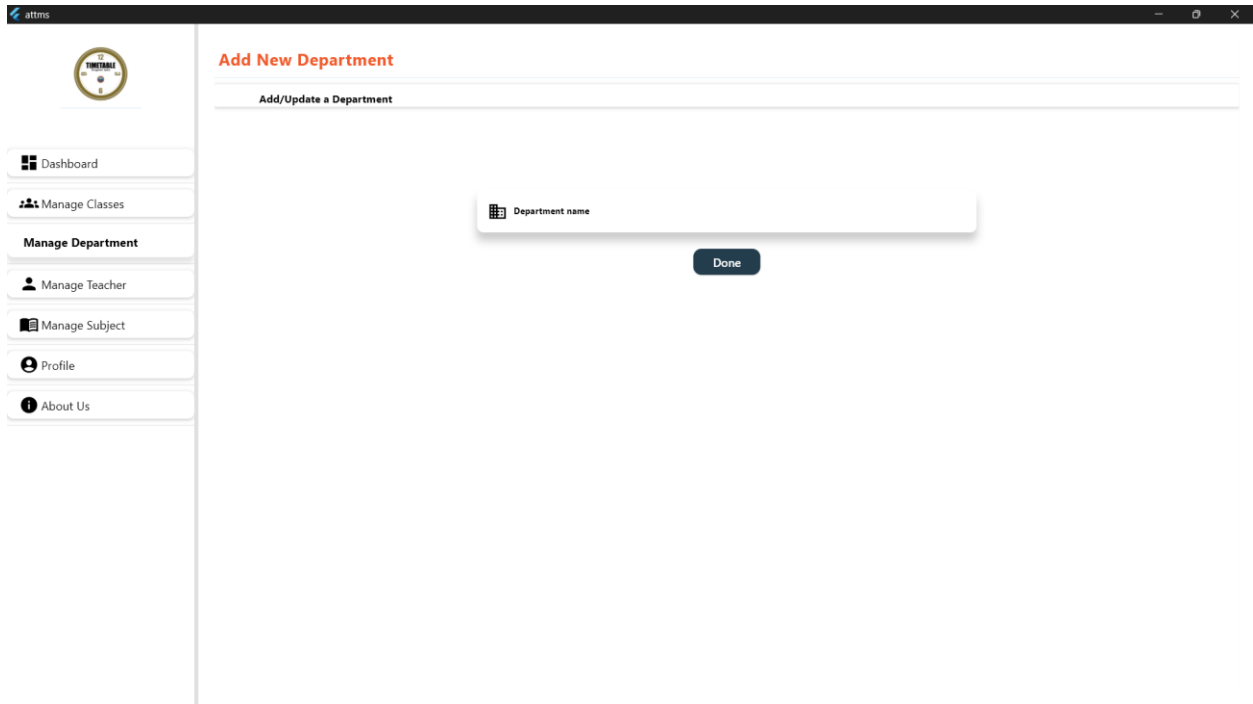
Adding a New Department

The admin can add a new department to the system directly through the Admin Dashboard If need it.





Adding a New Department

The admin can add new departments to the system through the Admin Dashboard. This needs to enter information, such as the department name. Once added, the department becomes part of the system and it will be listed, and view.



The screenshot shows a web application interface for adding a new department. On the left is a sidebar with a logo and a list of navigation items: Dashboard, Manage Classes, Manage Department (highlighted), Manage Teacher, Manage Subject, Profile, and About Us. The main content area is titled 'Add New Department' in red. Below the title is a sub-header 'Add/Update a Department'. The form contains a single input field labeled 'Department name' with a calendar icon on the left. A 'Done' button is positioned below the input field.

 atms



Dashboard

Manage Classes

Manage Department

Manage Teacher


Manage Subject

Profile

About Us

Add New Department

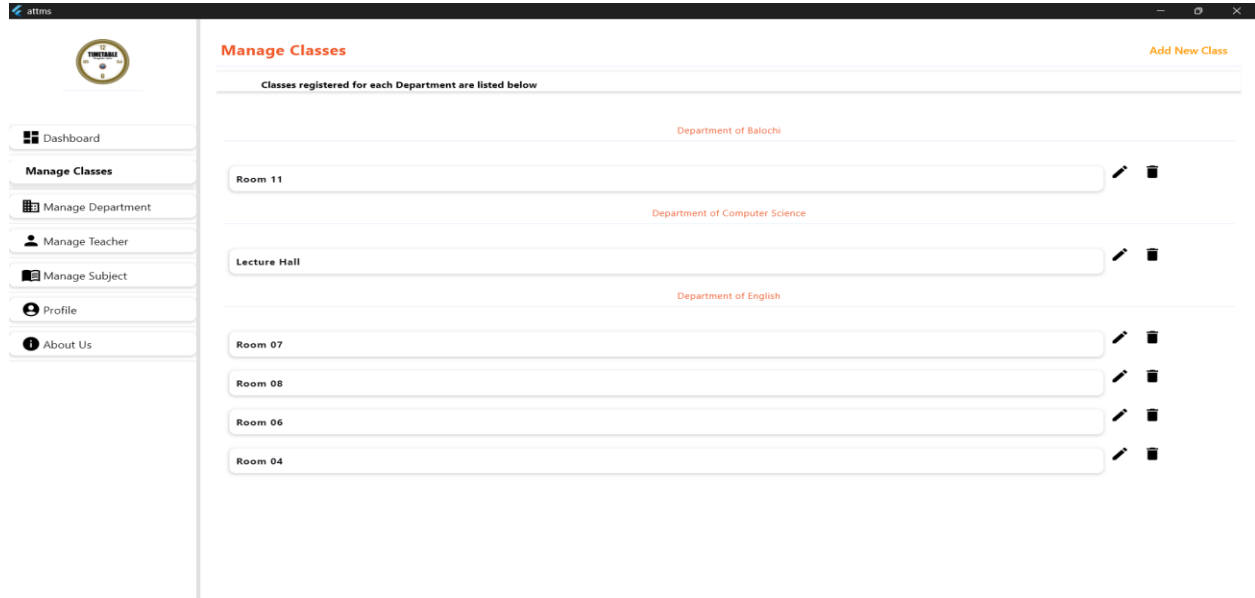
Add/Update a Department

 Department name

Done

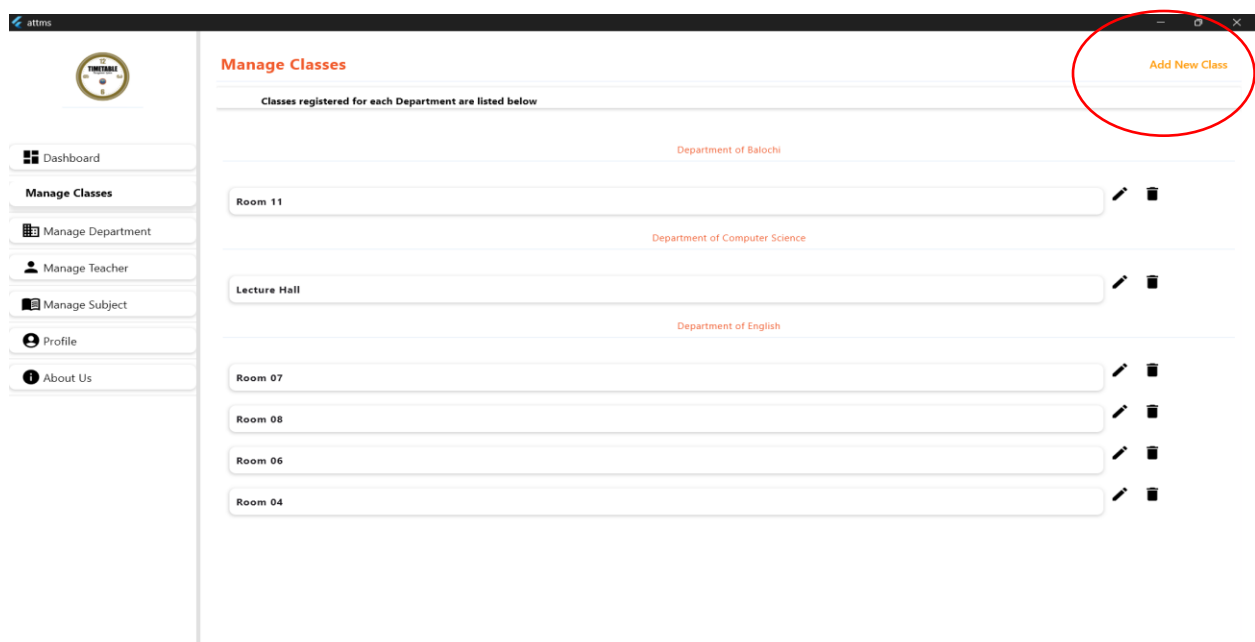
Managing Classrooms

The admin can manage all classrooms within the system. And list of classrooms and organizing them as required. The admin has the authority to update classroom details, and also can remove the classroom if not available at the time.



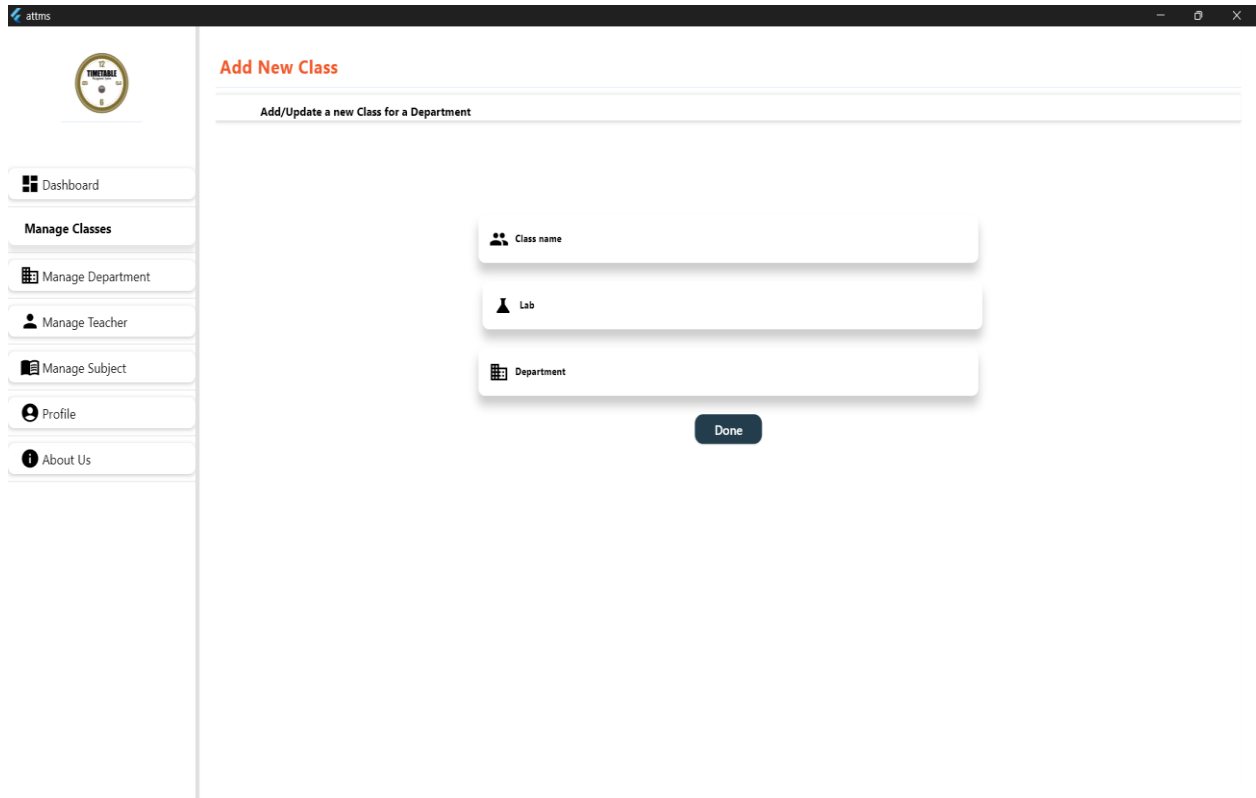
Adding a New Classroom

The admin can add new classrooms to the system through the manage Classes Dashboard.





Adding a New Classroom

Admin also can add a new classrooms in the system , through the manage classes Dashboard. To enter Details, such as the classroom's name, Department, and classroom nature lab/simple classroom. After submission, the classroom is added and becomes accessible to assign.



The screenshot shows a web application window titled 'attms'. On the left is a sidebar with a circular logo at the top and a list of menu items: Dashboard, Manage Classes (highlighted), Manage Department, Manage Teacher, Manage Subject, Profile, and About Us. The main content area is titled 'Add New Class' in orange. Below the title is a subtitle 'Add/Update a new Class for a Department'. The form contains three input fields: 'Class name' with a group of people icon, 'Lab' with a person icon, and 'Department' with a grid icon. A dark blue 'Done' button is positioned below the 'Department' field.

 attms



Dashboard

Manage Classes

Manage Department

Manage Teacher

Manage Subject

Profile

About Us

Add New Class

Add/Update a new Class for a Department

Class name

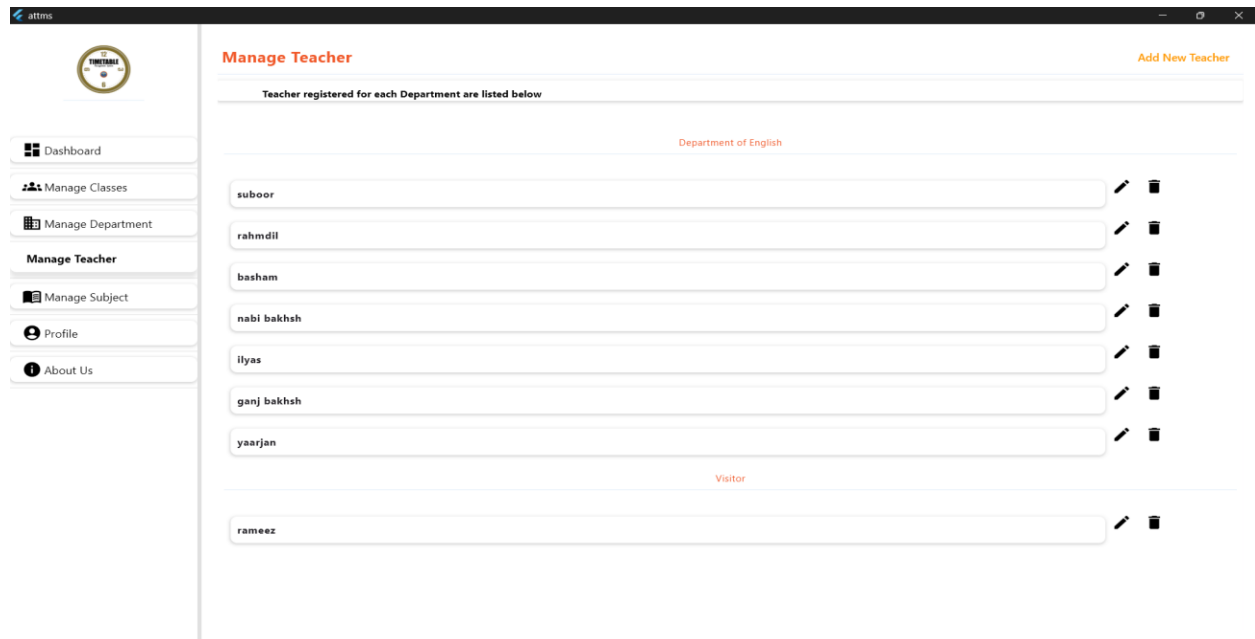
Lab

Department

Done

Managing Teachers

The admin can manage all Teachers within the system. And add a new teacher if. The admin has the authority to update teachers details, and also can remove teachers if not available at the time.



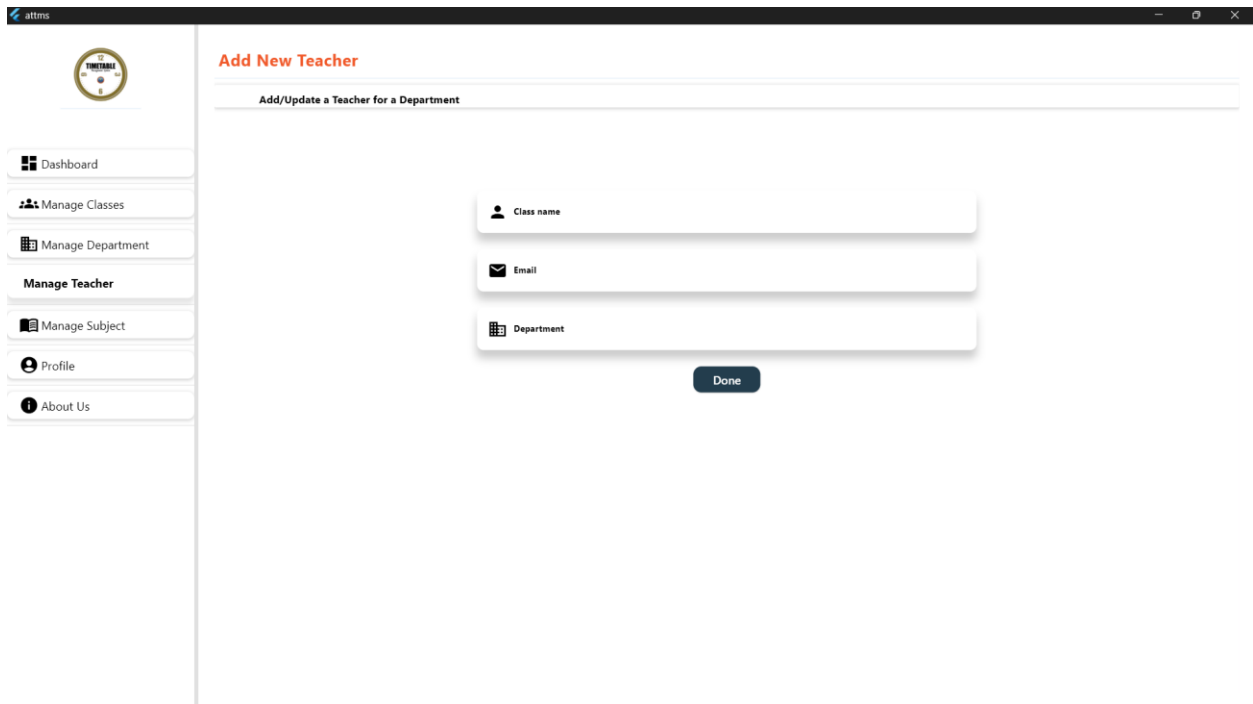
Adding New Teachers

The admin can add teachers to the system through the manage teachers dashboard if a teaches available.





Adding a New Classroom

Admin also can add a new Teacher in the system , through the manage Teachers Dashboard. To enter The Teachers Details, Teacher's name, Department and Mail. After submission, the Teacher is added and becomes accessible to assign.



The screenshot shows a web application interface for adding a new teacher. On the left is a sidebar with a logo and a menu containing: Dashboard, Manage Classes, Manage Department, **Manage Teacher** (highlighted), Manage Subject, Profile, and About Us. The main content area is titled 'Add New Teacher' in red. Below the title is a subtitle 'Add/Update a Teacher for a Department'. The form consists of three input fields: 'Class name' (with a person icon), 'Email' (with an envelope icon), and 'Department' (with a grid icon). A 'Done' button is located at the bottom right of the form.

atoms



Dashboard

Manage Classes

Manage Department

Manage Teacher

Manage Subject

Profile

About Us

Add New Teacher

Add/Update a Teacher for a Department

Class name

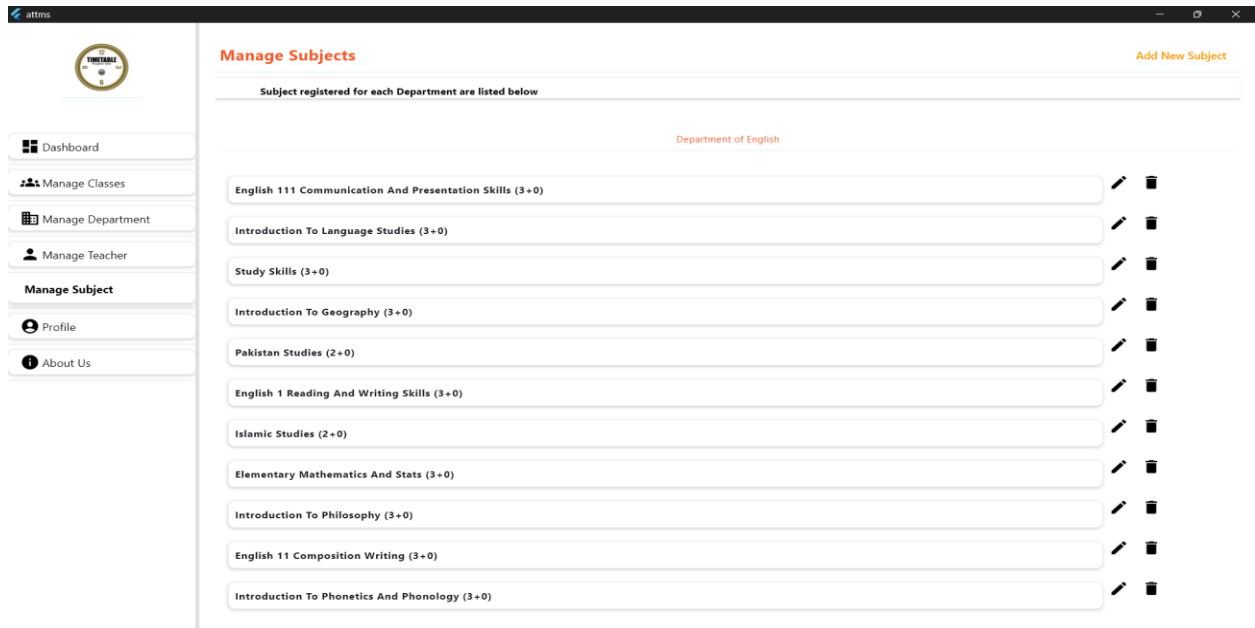
Email

Department

Done

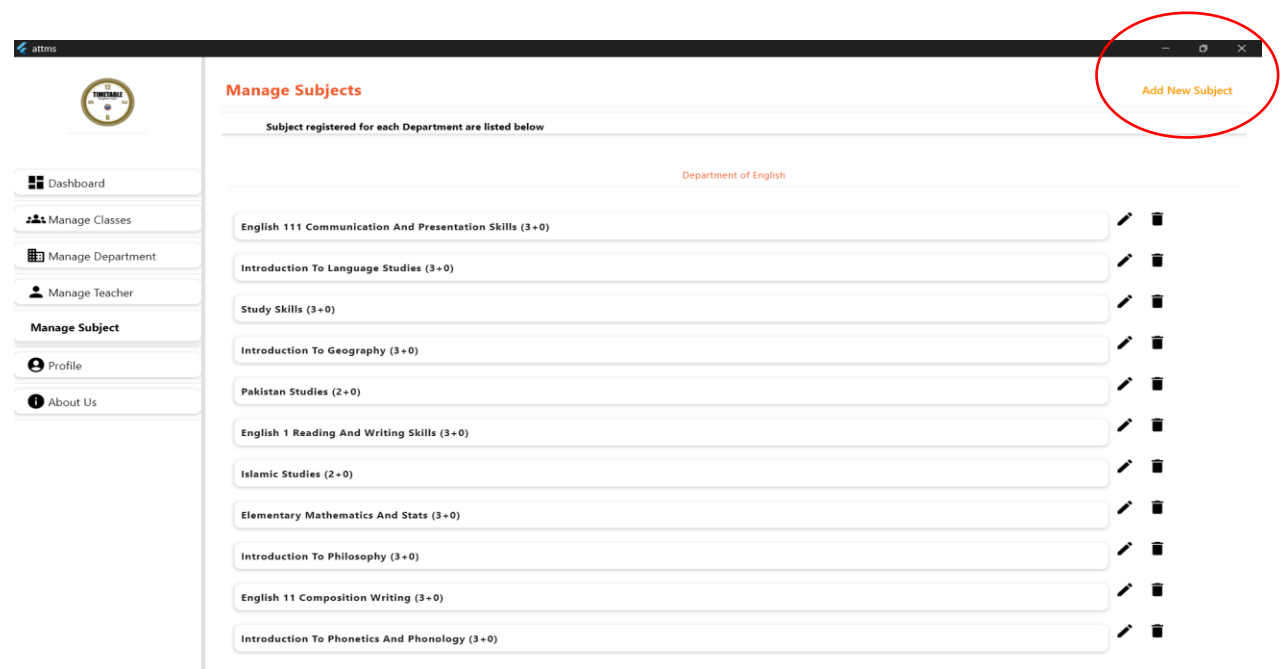
Manage Subject

The admin can manage all the subjects within the system. And also can add new subjects. If any new subject is being offered. The admin has the authority to update and remove the subjects if not being offer further more.



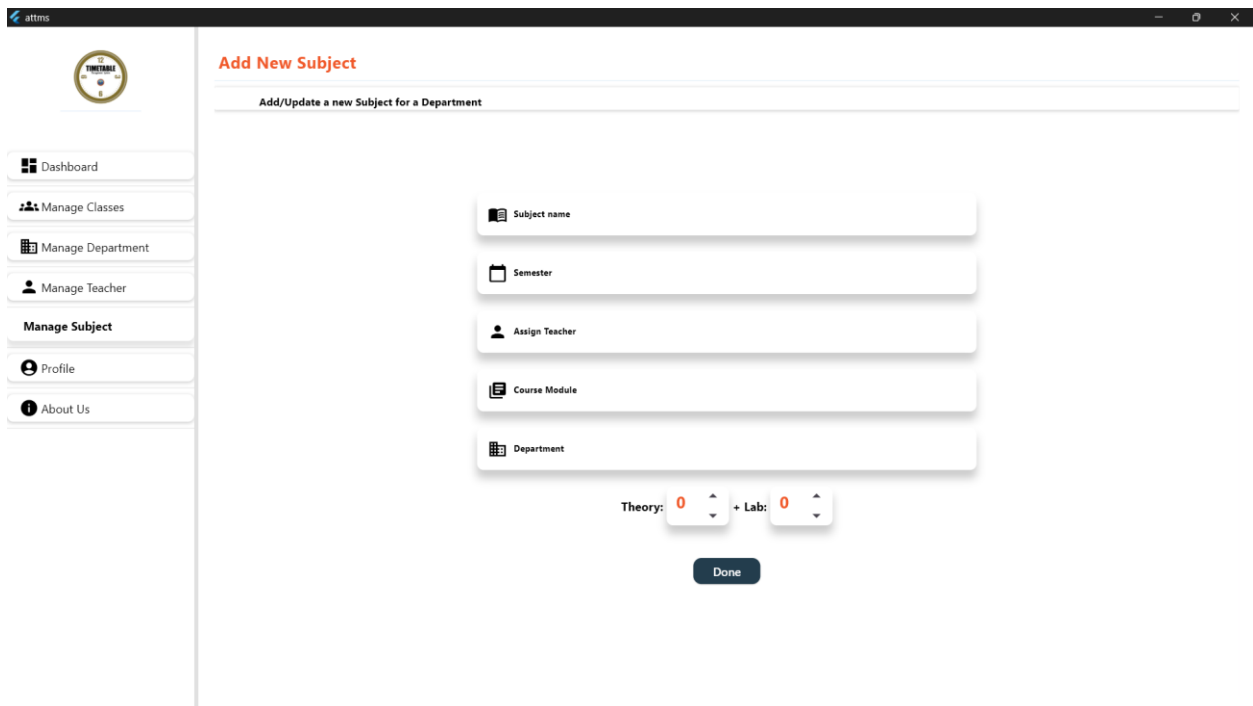
Adding a new subjects

Admin can add new subjects if upcoming or newly being offered so that subjects also Admin can add it to the system through the managing subjects through clicking the button of add a new subject then it will be available to offer.



Adding a New subject

Admin also can add a new subjects in the system. through the manage Techers Dashboard. To enter The subjects Details, Subject's name, semester name, assign to teachers, course module and Department. After submission the new subject is added and becomes accessible to assign.



The screenshot shows a web application window titled 'atims'. On the left is a sidebar with a logo and a list of menu items: Dashboard, Manage Classes, Manage Department, Manage Teacher, **Manage Subject**, Profile, and About Us. The main content area is titled 'Add New Subject' and contains a subtitle 'Add/Update a new Subject for a Department'. Below this are five input fields: 'Subject name', 'Semester', 'Assign Teacher', 'Course Module', and 'Department'. At the bottom of the form are two spinners for 'Theory' and 'Lab' counts, both currently set to 0, and a 'Done' button.

atims

Add New Subject

Add/Update a new Subject for a Department

Subject name

Semester

Assign Teacher

Course Module

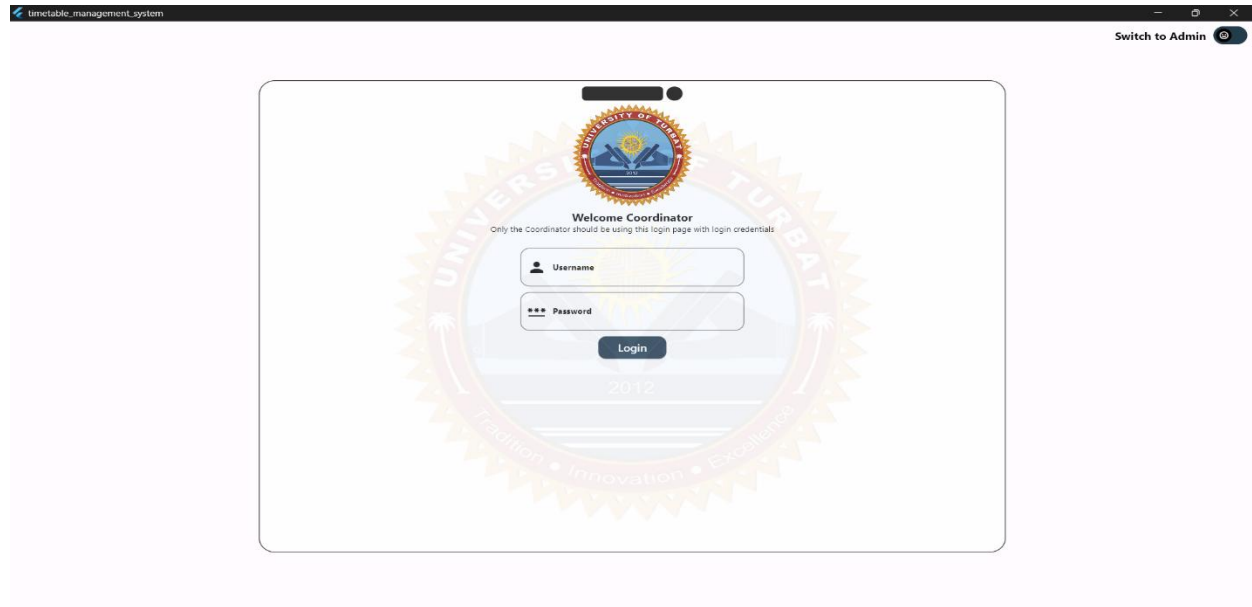
Department

Theory: 0 + Lab: 0

Done

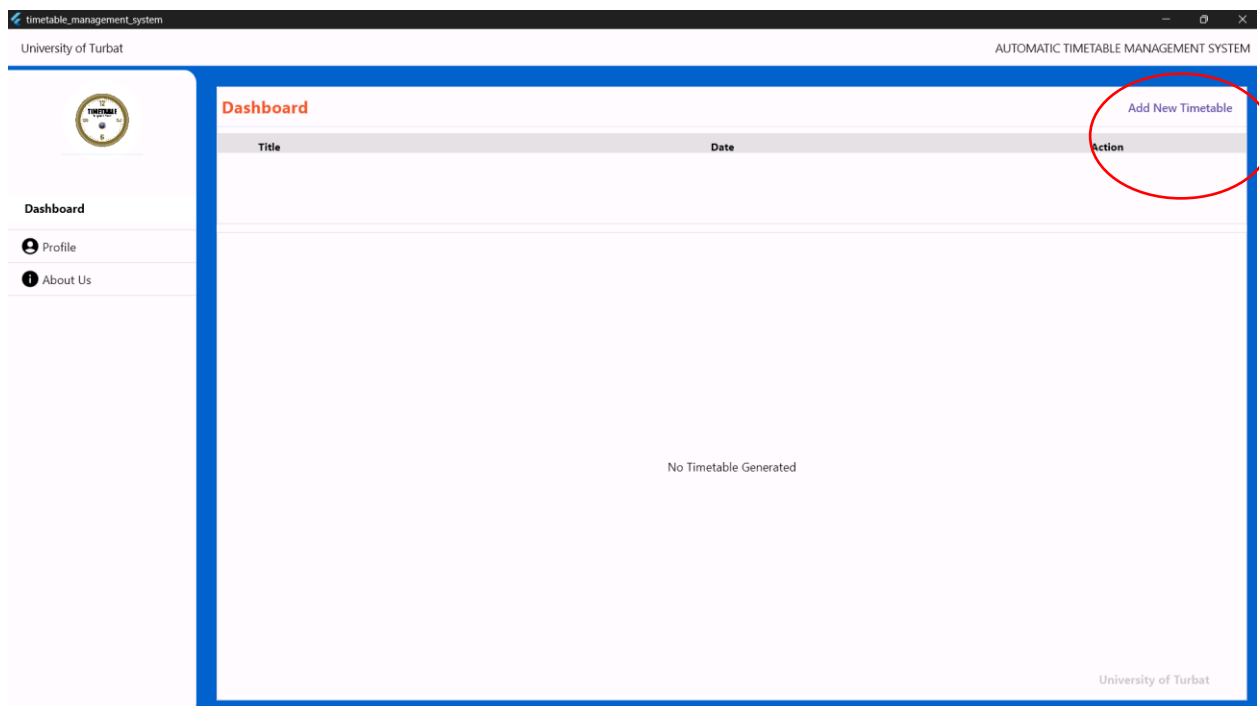
User Interface for Coordinator Login Page

The coordinator login into the system by entering their email and password on the login screen. After successfully logging in, they gain access to the timetable management feature.



User Interface for Coordinator Dashboard Page

After logging in, the coordinator accesses the Timetable Dashboard, where all timetables are listed. And if any other new timetable needed then by clicking the Add New Timetable button, coordinator can create a new timetable as needed.



Coordinator creating A new Timetable

By clicking the Add new Timetable on the top right side of the Dashboard the coordinator will be able to create new timetable by entering the details select Semester, to enter time, ending time the panel with the select semester page is redirected to the coordinator

timetable_management_system

AUTOMATIC TIMETABLE MANAGEMENT SYSTEM

Select Semester And Time Subjects Select Class Select Teacher Generate Timetable

New Timetable

Go Back

Choose Semester, By choosing semester 1 will enable automatic selecting other semester with subjects respectively

Select the Semester(by selecting semester 01, every odd semesters are selected automatically by Timetable management system)

Select Semester

Select start and End time for 1st slot classes(Time from 9:00 to 3:30 are accepted and Friday is a special case for the Time Selection)

From Enter Time To Enter Time

Select start and End time for 2nd slot classes(Time from 9:00 to 3:30 are accepted and Friday is a special case for the Time Selection)

From Enter Time To Enter Time

Select start and End time for 3rd slot classes(Time from 9:00 to 3:30 are accepted and Friday is a special case for the Time Selection)

From Enter Time To Enter Time

2012

Tradition • Innovation • Excellence

University of Turbat

Manage timetable

After adding a new timetable it would be listed here to the current timetable can be like update and delete

intems

Dashboard

Add New Timetable

Title	Date	Action
No Timetable Generated		

University of Turbat

Dashboard

- Manage Sub Teacher
- Request Free Class
- Profile
- About Us

Assigning subject teacher

The coordinator can assign subject to the teachers by selecting a course from the list and select the teacher for it. Then the teacher will be selecting for the course which is assigned.

The screenshot shows the 'Manage subjects' page in the altms system. The left sidebar contains a navigation menu with options: Dashboard, Manage Sub Teacher (selected), Request Free Class, Profile, and About Us. The main content area is titled 'Manage subjects' and includes a sub-header 'Subject registered for each Department are listed below'. The page is divided into two sections: 'semester 01' and 'semester 02'. Each section lists subjects with their credit values and a button to assign a teacher.

Semester	Subject	Credits	Teacher
semester 01	Introduction To Literary Studies	3+0	suboor
	Introduction To Language Studies	3+0	suboor
	Study Skills	3+0	rahmdil
	Introduction To Geography	3+0	basham
	Pakistan Studies	2+0	nabi bakhsh
	English 1 Reading And Writing Skills	3+0	ilyas
semester 02	Islamic Studies	2+0	yaarjan
	Elementary Mathematics And Stats	3+0	yaarjan

University of Turbat

Request for classes

If classroom/slot from another department is needed, a request can be sent to that department's coordinator. Once the request is approved, then classroom/slot becomes available to be assigned

The screenshot shows the 'Request For Classes' page in the altms system. The left sidebar is identical to the previous screenshot, with 'Request Free Class' selected. The main content area is titled 'Request For Classes' and includes a sub-header 'Request other Free Classes'. There are two tabs: 'Request a Class' (selected) and 'Manage Requests'. The page is divided into two sections: 'Department of Balochi' and 'Department of Computer Science'. Each section has a text input field for requesting a class.

Department	Request
Department of Balochi	Room 11
Department of Computer Science	Lecture Hall

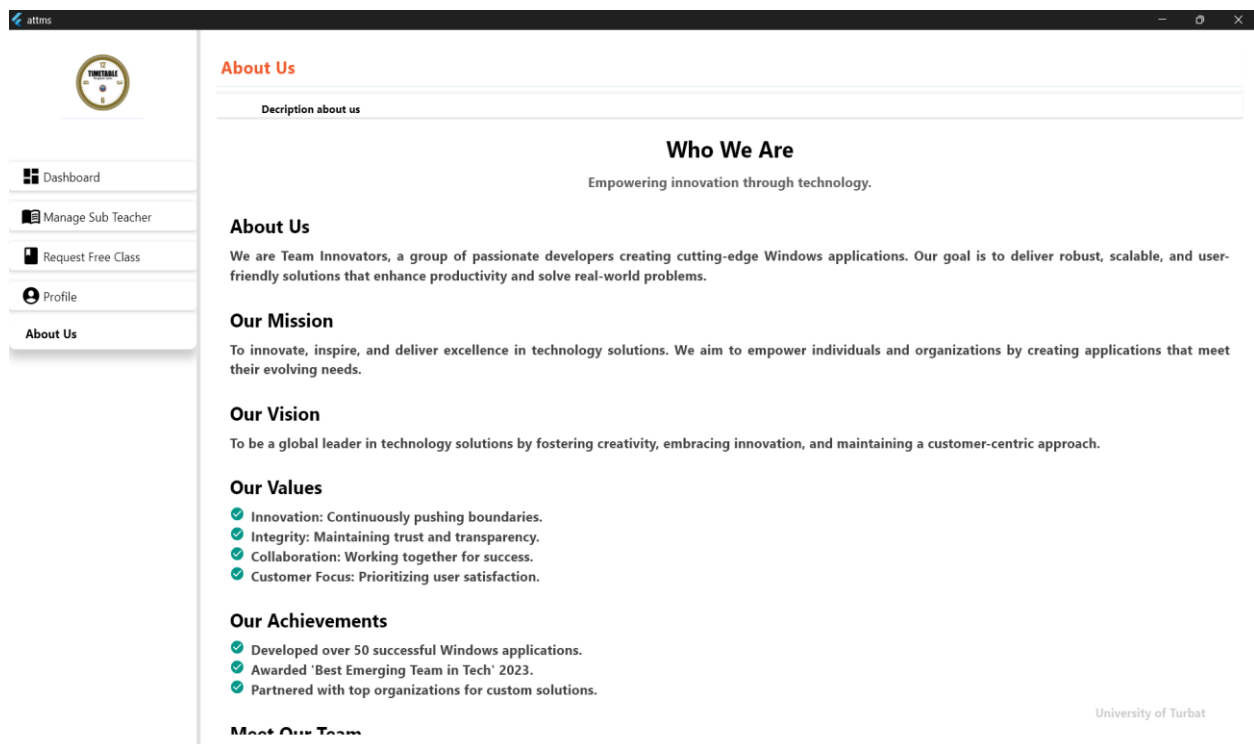
University of Turbat

About us

We are Team Innovators, dedicated to developing cutting-edge Windows applications that enhance productivity and solve real-world challenges. Our mission is to empower organizations with user-friendly, scalable, and innovative technology solutions.

Our Team

- Group Leader: Bakhshullah Wahid
- Team Leaders: Abdul Wahab, Rahat Manzoor, Robina Muhammad Karim



The screenshot shows a web application interface for 'attms'. On the left is a sidebar with a circular logo at the top and a list of menu items: Dashboard, Manage Sub Teacher, Request Free Class, Profile, and About Us (which is highlighted). The main content area is titled 'About Us' in orange. Below this is a section 'Who We Are' with the tagline 'Empowering innovation through technology.' followed by a paragraph about Team Innovators. Below that are sections for 'About Us' (repeating the team description), 'Our Mission' (to innovate, inspire, and deliver excellence), 'Our Vision' (to be a global leader), 'Our Values' (listing four values with checkmarks), and 'Our Achievements' (listing three achievements with checkmarks). At the bottom right, it says 'University of Turbat'.

About Us

Description about us

Who We Are

Empowering innovation through technology.

About Us

We are Team Innovators, a group of passionate developers creating cutting-edge Windows applications. Our goal is to deliver robust, scalable, and user-friendly solutions that enhance productivity and solve real-world problems.

Our Mission

To innovate, inspire, and deliver excellence in technology solutions. We aim to empower individuals and organizations by creating applications that meet their evolving needs.

Our Vision

To be a global leader in technology solutions by fostering creativity, embracing innovation, and maintaining a customer-centric approach.

Our Values

- ✔ Innovation: Continuously pushing boundaries.
- ✔ Integrity: Maintaining trust and transparency.
- ✔ Collaboration: Working together for success.
- ✔ Customer Focus: Prioritizing user satisfaction.

Our Achievements

- ✔ Developed over 50 successful Windows applications.
- ✔ Awarded 'Best Emerging Team in Tech' 2023.
- ✔ Partnered with top organizations for custom solutions.

University of Turbat

Meet Our Team

Select Classes

The coordinator select the classes by double clicking the classes name and if any other department classes needed then also can select. By selecting the classes will be listed right side. And then the selected classes will be assign further.

atms

Back

Select Semester And Time

Select Class

Select Teacher

Subjects

Generate Timetable

Class Selection

Next

Your department_name classes are already selected if you want another department class for your batches. You can skip the class selection Process if you Dont want other class of Other department_name

Department of Balochi

Room 11

Department of Computer Science

Lecture Hall

Department of English

Room 07

Room 08

Room 06

Room 04

→

←

Double click on a class to add

University of Turbat

Selecting Semester and Time

By selecting the semester and time a will be adding a new semester and select he semester name and select the timing of slots/classes timing. And the given time is universal time for all, time can be set according to the needs of slots.

atms

Back

Select Semester And Time

Select Class

Select Teacher

Subjects

Generate Timetable

New Timetable

Next

Choose Semester, By choosing semester 1 will enable automatic selecting other semester with subjects respectively

⌚ Timetable Name


Select the Semester(by selecting semester 01, every odd semesters are selected automatically by Timetable management system)

Select Semester

Selected start and End time for 1st slot classes
From 9:30 AM To 11:00 AM

Selected start and End time for 2nd slot classes
From 11:30 PM To 1:00 PM

Selected start and End time for 3rd slot classes
From 1:30 PM To 3:00 PM



University of Turbat

Teacher Selection

Department Teachers are listed .The coordinator can select teachers by double-clicking on the Teacher name. If Teacher from another department is needed, the coordinator can click on Others Teacher and all teachers are listed from all departments and select the required teacher. Selected Teacher is added to the list for assign.

atims

Back

Select Semester And Time

Select Class

Select Teacher

Subjects

Generate Timetable

Teacher Selection

Next

Select Teacher(visitor/Other department_name Teacher/Your Own department_name for each subject listed below)

Department of English

suboor

rahmdil

basham

nabi bakhsh

ilyas

ganj bakhsh

yaarjan

Visitor

rameez

→
←

Department of English

suboor (selected)

University of Turbat

Subject Selection

All semester subjects are listed and next to them list of available teachers. By selecting a subject, the coordinator can assign it to the best domain teacher from their department. If a teacher from another department is needed, the coordinator can select other Teachers and assign the course to the Teacher. Once assigned, the subject is updated with the teacher's name.

atms

BackSelect Semester And TimeSelect ClassSelect TeacherSubjectsGenerate Timetable

Subject SelectedNext

Select Teacher(visitor/Other department_name Teacher/Your Own department_name for each subject listed below)

semester 01

Introduction To Literary Studiessuboor

Introduction To Language Studiessuboor

Study Skillssahmdil

Introduction To Geographybasham

Pakistan Studiesnabi bakhsh

English 1 Reading And Writing Skillsslyas

semester 03

Introductin To Information And Computer Technology (ICT) Skillsganj bakhsh

University of Turbat

TimeTable Genetated

After providing the requirements the at the end when the coordinator clicks the "Generate Timetable" button, the system automatically compiles a complete timetable. Once the timetable, the coordinator can print or download the timetable in a format of pdf.

altms

Back

Select Semester And Time

Select Class

Select Teacher

Subjects

Generate Timetable

Generate Timetable

When informations are selected completely by clicking the generate button you will be generating the timetable

Generate Timetable

BSCS 1st					
	Monday	Tuesday	Wednesday	Thursday	Friday
9:30 AM	English 1 Reading And Writing Skills Room 08 ilyas	Break	English 1 Reading And Writing Skills Room 07 ilyas	Study Skills Room 08 rahmdil	Break
11:30 AM	Introduction To Literary Studies Room 06 suboor	Study Skills Room 08 rahmdil	Pakistan Studies Room 06 nabi bakhsh	Introduction To Geography Room 06 basham	Introduction To Literary Studies Room 06 suboor
1:30 PM	Introduction To Language Studies Room 07 suboor	Pakistan Studies Room 07 nabi bakhsh	Introduction To Geography Room 06 basham	Introduction To Language Studies Room 08 suboor	Break
BSCS 3rd					
	Monday	Tuesday	Wednesday	Thursday	Friday
9:30 AM	Introductin To Information And Computer Technology (ICT) Skills Room 06 ganj bakhsh	Global Poetry Room 06 nabi bakhsh	Break	Introductin To Information And Computer Technology (ICT) Skills Room 07 ganj bakhsh	Introduction To Morphology Room 08 rameez University of Turbat

Chapter 5

System Implementation

5.1. Introduction:

The stage to bring together the modules of the framework which were outlined before into one design. The approach of this stage is technical particulars or algorithms as programming, and programming software components development. The design phase just proclaims the static interface of the ATTMS while this face centers on segment gathering, making the system dynamic. This stage is not just bound with coding; it likewise talks about the platforms and the picked languages for the development of the system. The following are the points of interest for the implementation of this ATTMS system.

5.2. Architecture:

Architecture is the center association of a framework typified in its part, their connections to each other, and to other, and the environment managing its plan and development we chose MVC(Model-View-controller) as architecture for implementing our ATTMS application.

5.3. MVC (Model-View-Controller):

Model-View-Controller is a structural example that isolates an application into three fundamental sensible parts: the model, the view, and the controller. Every one of these parts are worked to deal with particular improvement parts of an application. MVC is a standout amongst the most much of the time utilized industry-standard ATTMS application improvement structure to make adaptable and extensible projects.

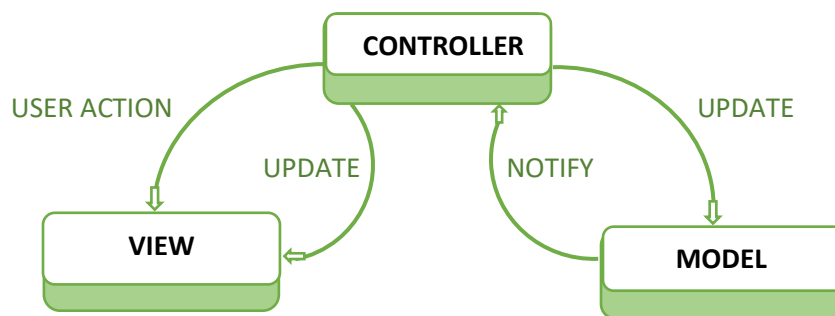


Figure 8 MVC Diagram

5.3.1. View:

View is a collection of classes representing the elements in the user interface (all of the things the user can see and respond to in the screen, such as buttons, display boxes, and so forth).

5.3.2. Model:

Model represents the underlying, logical structure of data in a software application and the high-level class associated with it. This object model does not contain any information about the user interface.

5.3.3 Controller:

The controller represents the classes connecting the model and view where it is used to communicate between classes in the model view.

5.4 Selection of Language:

The choice of a programming language is an imperative perspective because a language resembles choosing materials for building a pinnacle. The language must satisfy the reason for the framework. It ought to be perfect for the prerequisites.

5.4.1 Flutter and Dart:

For the development of ATTMS, we selected Flutter and Dart as our core tools. Flutter, a versatile open-source UI software development kit by Google, provides a seamless way to create natively compiled applications for mobile, web, and desktop from a single codebase. Dart, its underlying programming language, offers robust support for reactive programming, ensuring dynamic, efficient, and scalable development. These technologies allowed us to design a highly responsive and visually appealing application with efficient cross-platform capabilities.

5.4.2 Django and Firebase:

For backend development, we adopted Django and Firebase due to their efficiency and scalability. Django, a high-level Python web framework, excels in implementing the Model-View-Template (MVT) architecture, ensuring robust and secure server-side functionality. Firebase complements this setup by providing a real-time database solution, enabling seamless synchronization of data across devices. This combination allowed us to handle dynamic data flows and maintain real-time interactions with ease.

5.4.3 Adobe XD:

For system design and wireframing, we utilized Adobe XD, a powerful tool for creating interactive prototypes and wireframes. Adobe XD allowed us to visualize the user interface and application flow before development, ensuring an intuitive and user-friendly design. By iterating on these wireframes, we streamlined the development process and effectively aligned with user needs.

5.4.4 Proposed Gantt Chart:

	Task							
		March	April	May	August	September	October	November

5.4.5 Present Gantt Chart:

	Task							
		March	April	May	August	September	October	November

Chapter 6

System testing

6.1 Introduction:

The stability of the system is one of the main concern; the hardware can have some software and many physical hardware related concern while the mobile app, website and backend has multiple programming and scalability concerns. The purpose of testing is to give assurance whether a system fulfill the requirement of a system, which was mentioned already. the test can be either to check the performance, reliability, speed or security. Therefore, there are certain types of strategies to do testing.

6.2 Testing Strategies:

Each component was tested; the unit test performed for functionality of the components security test is conducted to check the vulnerabilities while a beta test was performed to examine the bugs of the system. Following are the testing strategies applied for this system.

6.2.1 Unit testing:

Unit testing focuses on the functionality of the system. In addition, makes insure whether the logic is complete and correct. Test all the states that brings changes in state. Like login to log out activities, adding, deleting and updating an Employees and Files in the system.

- The components fulfill the expectations.
- If there is any defect in components, these should reveal by the test cases.

6.2.2 Integration testing:

The whole system components tested separately at first. Then evaluates the interaction between them and all components are tested once they ware integrated.

6.2.3 Validation testing:

The validation testing is done to assure if the system is working according to the requirements. The validation is highly concerned with the product rights; it is necessary to be done to avail a good product.

6.2.4Security testing:

There is no system, which does not demand for security. Therefore, the security purpose is checked like to verify user identities, and to encrypt confidential information is paramount importance. Who should be authorized and authenticate the system.

6.2.5Performance testing:

The performance testing was done by having multiple users using the system concurrently and having multiple files were sent to multiple department and being inserted into and accessed from the system.

6.2.6 Regression testing:

Regression testing is a regular based test, after bringing each change it is a checked whether any new bug has occurred or not.

6.2.7 Security testing:

Any computer-based system has high concern with its security. In security testing the system security level is tested that how much it is capable of running the website or any other application. During this testing tester try to acquire passwords from external clerical means, attack the system with custom and software designed to break down any defenses, cause system error.

6.3 Testing technique:

Black box testing

The black box testing is called behavioral testing. It focuses on the functional requirement of the software. Black box is not an alternate of the white box testing, rather it is a complementary approach that is likely to uncover a different class of errors than white method. This type of testing tells about the presence of the errors. The tester is only aware of what the software is supposed to do, but not know how. The internal structures which are used for this testing technique are: software specifications, requirements, and the design to drive the test cases. Possibly, the test cases are functional or non-functional, through usually they are functional. In the black box testing, selection based on the valid and invalid inputs and determines the correct output.

6.4 Test cases:

A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirement or works correctly. The mechanism for determining whether a software program or system has passed or failed such as test is known as an oracle. A series of test performed repeatedly in the order to find that system is working.

6.4.1 Test Case 1 for User Login

Test Case ID	FTS-01
Test Case Name	Admin Login
Precondition	The user must be registered in the system.
Purpose	The purpose of this case is that, except the actual administration no other user has the access this account.
Input	Username : admin@gmail.com Password: abc123

Expected Output	Admin should be logged in the web portal
Actual Output	User is successfully logged in.
Result of test	The expected and actual result is matched and no error were found.

6.4.2 Test Case 2 for User Login

Test Case ID	FTS-02
Test Case Name	User Login
Precondition	The user must be registered in the system.
Purpose	The purpose of this case is that, except the actual administration no other user has the access this account.
Input	Username : user@gmail.com Password: user123
Expected Output	Admin should be logged in the web portal
Actual Output	User is successfully logged in.
Result of test	The expected and actual result is matched and no error were found.

6.4.3 Test Case 3 for User Invalid Login

Test Case ID	FTS-03
Test Case Name	User Invalid Login
Precondition	The user should not be registered.
Purpose	The purpose of this case is to confirm if any random user can access the user form
Input	Username : dilmurad@gmail.com Password: 1234
Expected Output	User would not be logged in and error message should be displayed.
Actual Output	User would not be logged in and error message "Login Failed: Invalid username or password" is displayed.
Result of test	The expected and actual result is matched and no error were found.

6.4.4 Test Case 4 for Add a new User Account

Test Case ID	FTS-04
Test Case Name	Add a new user account
Precondition	Admin user must be logged in to admin account.

Purpose	The purpose of doing this test case is to check successful creation of a new user account.
Input	Username : dilmurad Email: dilmurad@gmail.com Password: 1234 Phone number:098765432
Expected Output	New user should be added to the database.
Actual Output	New user is added to the database and user is redirected to “Users” page.
Result of test	No error

Table 43 Test case for create a new user account

6.4.5 Test Case 1 for Logoff User

Test Case ID	FTS-06
Test Case Name	Logoff User
Precondition	User must be logged in the system.
Purpose	The purpose of this case is to check if user is able to successfully logoff the user from the system
Input	Click the “Logout” button at the navigate menu.
Expected Output	User should be logged from the system.

Actual Output	User is successfully logout from the system.
Result of test	There is no error between Expected and Actual output.

6.4.6 Test Case 1 for add a new Department

Test Case ID	FTS-09
Test Case Name	Add a new Department
Precondition	Admin user must be logged in to admin account.
Purpose	The purpose of doing this test case is to check successful creation of a new department.
Input	Click on "Manage Block" in the navigation Click on "Department" Click on "Add New" Enter the Name and Phone Number for the Department. Click on "Submit"
Expected Output	New Department should be added to the database.
Actual Output	New Department is added to the database and user is redirected to "Department" page.
Result of test	No error

