

# KIDS AREA APPLICATION

## Software project management

The kids area application offers engaging, educational, and safe content for children.

2024



# Kids area Application

**Project Manger**

Mohamed Agami Fikry Qub

**The Supervision of**

Innova Egypt

# Contents

<b>1. Project overview .....</b>	<b>4</b>
1.1. Project Description.....	4
1.2. Project Scope.....	5
1.3. Assumptions .....	6
1.4. Constraints .....	6
<b>2. Project Start-Up .....</b>	<b>7</b>
2.1. Project Life Cycle .....	7
2.2. Methods, Tools, and Techniques.....	8
2.3. Estimation Methods and Estimates .....	9
2.4. Schedule Allocation .....	9
2.5. Resource Allocation .....	10
2.6. Budget Allocation .....	10
<b>3 Risk Management .....</b>	<b>11</b>
<b>4. Project time management .....</b>	<b>12</b>
4.1. Name task. ....	12
4.2. PERT approach.....	13
4.3. Network diagram.....	14
4.4. Calculator early start and early finish and late start and late finish and free float and total float.....	15
4.5. Gantt chart.....	16
<b>5. Economic Feasibility Study.....</b>	<b>18</b>
<b>6. Data flow diagram .....</b>	<b>20</b>
6.1. Context diagram (Level 0) .....	20
6.2. data flow diagram (Level 1) .....	21
<b>7. Use Case .....</b>	<b>24</b>
<b>8. Activity diagram.....</b>	<b>25</b>
<b>9. Design and implementation.....</b>	<b>26</b>
<b>10. Testing .....</b>	<b>39</b>

# 1. Project overview

## 1.1. Project Description

The Kids Area Software Application is a dedicated platform designed to provide children with a safe, engaging, and educational digital environment. The application aims to balance entertainment and learning by offering interactive games, educational content, and creative activities tailored to different age groups. Additionally, it will feature robust parental controls to ensure a secure experience for children while providing parents with the tools to monitor and manage their child's digital interactions.

## Objectives

1. **Engagement and Learning:** Develop an application that captivates children's interest while fostering their cognitive, social, and emotional development through educational content.
2. **Safety and Security:** Ensure a secure digital environment with advanced parental controls and data protection measures.
3. **User-Friendly Experience:** Create an intuitive and visually appealing interface that is easy for children to navigate and enjoyable to use.
4. **Performance and Reliability:** Deliver a responsive and reliable application that performs consistently across various devices and platforms.

## Key Features

1. **Interactive Games:** A collection of fun and educational games that promote learning in areas such as math, language, science, and problem-solving.
2. **Educational Content:** Videos, stories, and quizzes designed to enhance knowledge and comprehension skills.
3. **Creative Activities:** Tools for drawing, storytelling, and music creation to stimulate creativity and self-expression.
4. **Parental Controls:** Features allowing parents to set usage limits, monitor activity, and control accessible content.
5. **User Profiles:** Customizable profiles for each child, enabling personalized experiences and tracking progress.

# 1.2. Project Scope

The Kids Area Application is designed to provide a secure, engaging, and educational digital environment for children. The application will feature interactive games, educational content, creative activities, and robust parental controls. It will be developed to support multiple platforms including iOS, Android, and web browsers, desktop applications ensuring accessibility and convenience for users. This project aims to deliver a high-quality, reliable, and user-friendly application that meets the needs of children and their parents.

Project Includes
<b>User Interface and Experience:</b> Development of an intuitive and visually appealing interface tailored for children.
<b>Educational Content:</b> A library of videos, stories, quizzes, and other educational materials.
<b>Creative Activities:</b> Tools for drawing, storytelling, and music creation.
<b>User Profiles:</b> Customizable profiles for each child to provide personalized experiences.
<b>Interactive Games:</b> A variety of educational and entertaining games focusing on subjects like math, language, science, and problem-solving.
<b>Security:</b> Implementation of stringent security measures to protect user data.

Project Excludes
<b>Custom Hardware:</b> No development or support for custom hardware devices or accessories.
<b>In-Person Support:</b> No provision for on-site customer support or physical training sessions.
<b>Advanced AI Features:</b> No implementation of advanced artificial intelligence or machine learning features beyond basic recommendations and progress tracking.

## 1.3. Assumptions

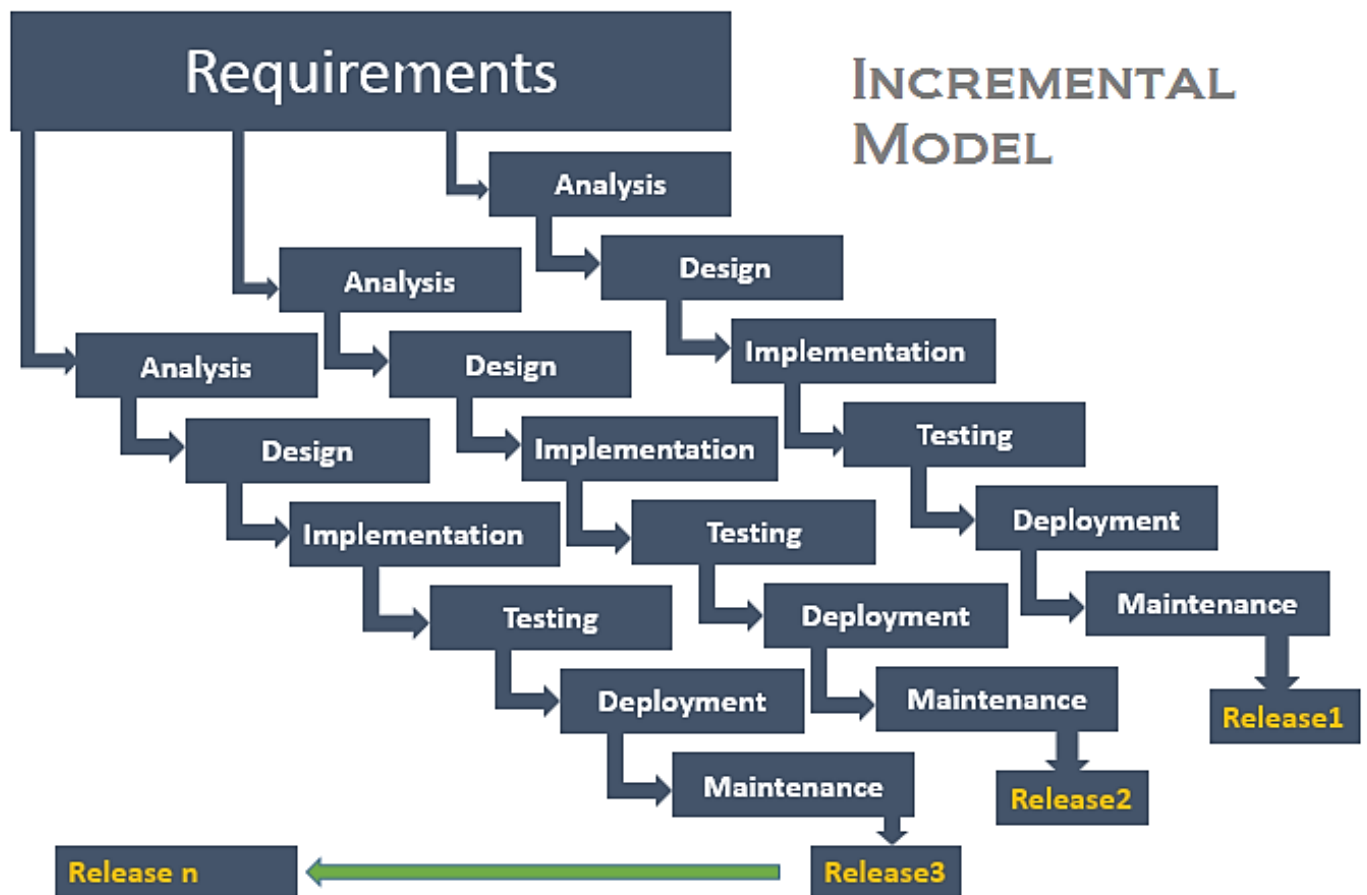
Assumptions
<b>Platform Compatibility:</b> The application will be compatible with major operating systems, including iOS, Android, and web browsers.
<b>Internet Connectivity:</b> Users will have a stable internet connection to access most features of the application.
<b>Device Specifications:</b> The target user devices (tablets, smartphones, and PCs) will meet the minimum hardware and software requirements to run the application efficiently.
<b>Security and Privacy:</b> Users will provide consent for data collection in compliance with relevant data protection regulations such as GDPR and COPPA.

## 1.4. Constraints

Constraints
<b>Project Deadlines:</b> The project has a fixed deadline for initial launch, which must be met to align with market opportunities and stakeholder expectations.
<b>Development Schedule:</b> Limited time for development sprints due to the fixed project timeline.
<b>Funding Limitations:</b> The project has a defined budget that cannot be exceeded.
<b>Resource Allocation:</b> Limited financial resources for hiring additional personnel or acquiring expensive tools and technologies.
<b>Human Resources:</b> Availability of skilled developers, designers, testers, and other essential team members may be limited.

## 2. Project Start-Up

### 2.1. Project Life Cycle



1. **Requirement analysis:** The product analysis expertise identifies the requirements in the first phase of the incremental model. The requirement analysis team also understands the system's functional needs. This phase is critical for developing software using the incremental methodology.
2. **Design & Development:** The design of the system functionality and the development technique is completed successfully in this phase of the Incremental model of SDLC. The incremental model employs the style and development phase when software creates new functionality.
3. **Implementation:** The coding phase of the development system is enabled by the implementation phase. It entails the final code created throughout the design, development, and testing process. Following the conclusion of this phase, the number of working products has increased and updated to the final system product.
4. **Testing:** The testing step of the incremental approach examines the performance of each current function and new capabilities. Various techniques are utilized to test the behavior of each task throughout the testing phase.

## 2.2. Methods, Tools, and Techniques

- **Methods**

- a. **Incremental development:** is an approach to software development where a project is broken down into smaller, manageable segments called increments. Each increment represents a portion of the overall project functionality, which is developed, tested, and delivered independently.
- b. **User-Centered Design (UCD):** Design the application with a focus on the needs and preferences of the target users, namely children, parents, and educators. Conduct user research, usability testing, and persona development to ensure the application's usability and appeal.

- **Tools**

- c. **NetBeans IDE** is a free and open-source integrated development environment for application development on Windows, Mac, Linux, and Solaris operating systems. The IDE simplifies the development of web, enterprise, desktop, and mobile applications that use the Java and HTML5 platforms.
- d. **Microsoft Access** is a popular information management tool that helps you store all kinds of information for reporting, analysis, and reference. With Microsoft Access, you can manage data more efficiently and analyze large amounts of information. As a business, using Microsoft Excel can't prove to be sufficient.
- e. **Microsoft Word** is a word processing program that allows for the creation of both simple and complex documents. With Office 365, you can download the application to your hard drive and will also have access to the online version.

- **Techniques**

- a. **Java** is a multi-platform, object-oriented, and network-centric language that can be used as a platform. It is a fast, secure, reliable programming language for coding everything from mobile apps and enterprise software to big data applications and server-side technologies.
- b. **SQL** is used to communicate with a database. According to ANSI (American National Standards Institute), it is the standard language for relational database management systems. SQL statements are used to perform tasks such as updating data on a database or retrieving data from a database.



## 2.3. Estimation Methods and Estimates

Description	[ Best / Most Likely / Worst]
Effort in person-months or person-hours	280 / 310 / 330 person-months or person-hours
Schedule in calendar months	5 / 6 / 7 calendar months
Budget in dollars	100,000 LE / 150,000 LE / 200,000 LE
Level of Uncertainty	Moderate

## 2.4. Schedule Allocation

Major Milestone/Deliverable	Planned Completion Date
complete Project Plan	June 1, 2025
Finalize UI/UX Design	January 15, 2025
Complete Frontend Development	February 30, 2025
Complete Backend Development	May 30, 2025
Desktop application	April 15, 2025
Conduct QA Testing	July 15, 2025
Create Educational Content	July 15, 2025
Complete Animations	July 15, 2025
Launch Marketing Campaign	July 1, 2025
Release Beta Version	July 15, 2025
Collect Feedback and Implement Improvements	July 30, 2025
Official Launch	July 15, 2025
Provide Customer Support	Ongoing

## 2.5. Resource Allocation

Resource	Total	Skill Set Requirements	Timeframe
Project Manager	1	Project management, communication, planning	6 months
UI/UX Designer	2	Graphic design, user interface design, user experience	4 months
Frontend Developer	3	HTML, CSS, JavaScript, React	5 months
Backend Developer	2	Node.js, Express, MongoDB	5 months
Desktop Application	3	Java, Access data base	7 months
QA Tester	2	Manual testing, automated testing, bug tracking	2 months
Content Creator	1	Writing, graphic design, education knowledge	3 months
Animator	1	Animation, graphic design	3 months
Marketing Specialist	1	Marketing strategy, digital marketing, social media	2 months
Support Specialist	1	Customer support, communication	2 months

## 2.6. Budget Allocation

Key Budget Category	Budget Amount (L.E)	Time Period
Project Manager	10000	6 months
UI/UX Designer	5000	4 months
Frontend Developer	2000	5 months
Backend Developer	3000	5 months
Desktop Application	6000	7 months
QA Tester	8000	2 months
Content Creator	9000	3 months
Animator	1000	3 months
Marketing Specialist	2000	2 months
Support Specialist	3000	2 months

### 3. Risk Management

Risk Description	Probability	Impact	Strategy
<b>Cost Estimates Unrealistic</b>	Medium	High	<ul style="list-style-type: none"> <li>• Conduct thorough market research.</li> <li>• Include contingency budget.</li> <li>• Regular budget reviews</li> </ul>
<b>Time Estimates Unrealistic</b>	Medium	High	<ul style="list-style-type: none"> <li>• Use historical data for estimates.</li> <li>• Implement buffer times.</li> <li>• Regular schedule reviews</li> </ul>
<b>Team Size</b>	Low	Medium	<ul style="list-style-type: none"> <li>• Hire additional resources needed.</li> <li>• Cross-train team members</li> <li>• Regular team evaluations</li> </ul>
<b>Project Scope Creep</b>	High	High	<ul style="list-style-type: none"> <li>• Define clear project scope.</li> <li>• Implement change control process.</li> <li>• Regular stakeholder meetings</li> </ul>
<b>Team Members Unknowledgeable of Business</b>	Medium	Medium	<ul style="list-style-type: none"> <li>• Provide business domain training.</li> <li>• Include business analysts in team.</li> <li>• Regular knowledge sharing sessions</li> </ul>
<b>Available Documentation</b>	Low	Medium	<ul style="list-style-type: none"> <li>• Ensure thorough documentation from the start.</li> <li>• Regularly update documentation.</li> <li>• Use documentation tools</li> </ul>
<b>Narrow Knowledge Level of Users</b>	Medium	Medium	<ul style="list-style-type: none"> <li>• Conduct user research and testing.</li> <li>• Develop user-friendly design.</li> <li>• Provide user training and support</li> </ul>

## 4. Project time management

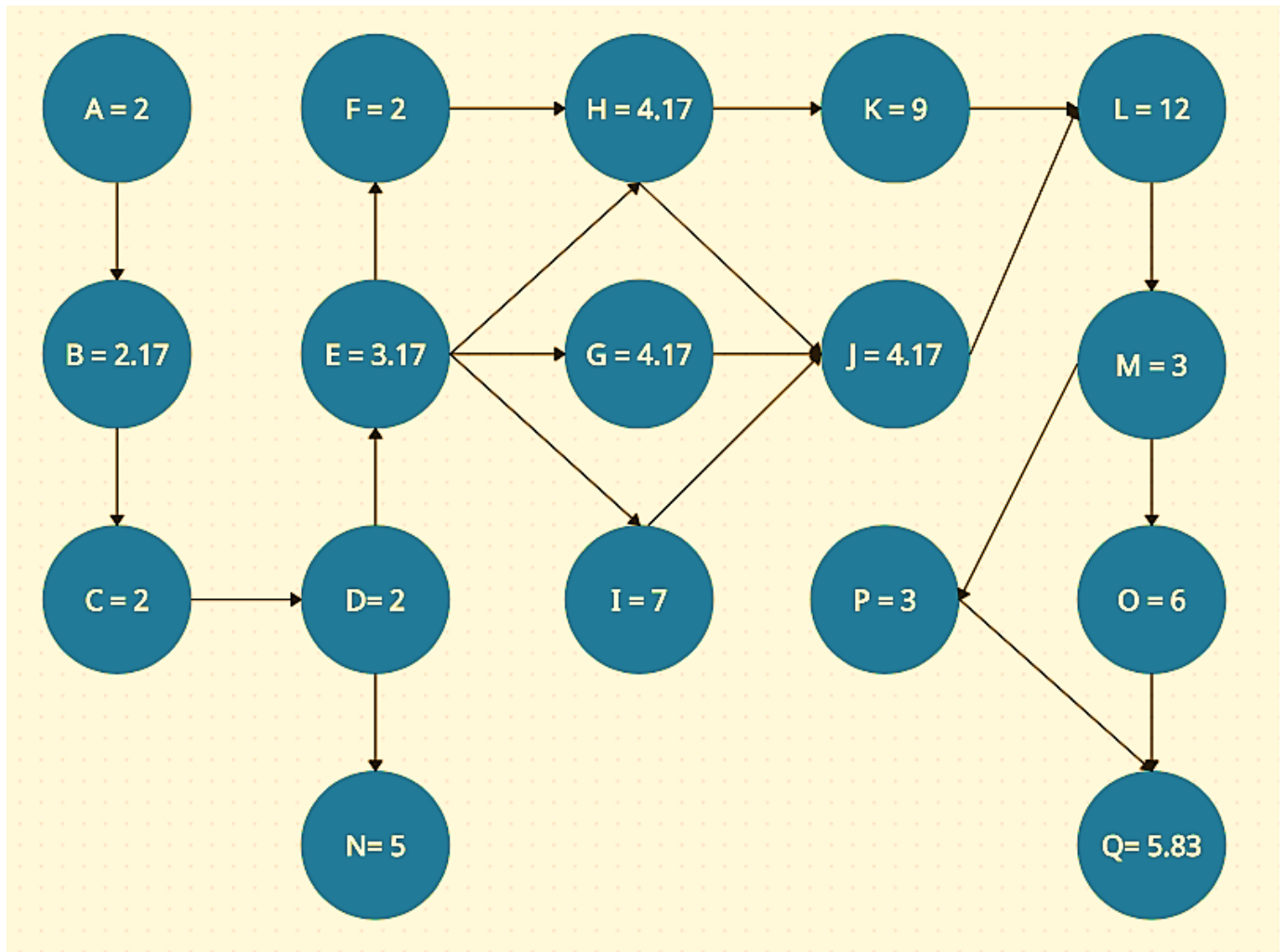
### 4.1. Name task.

Task	Activity	Predecessor Activity
A	Requirement Gathering	-
B	Concept Design	Requirement Gathering
C	Prototype Development	Concept Design
D	User Feedback	Prototype Development
E	Final Design	User Feedback
F	Database Design	Final Design
G	Frontend Development	Final Design
H	Backend Development (desktop application)	Final Design, Database Design
I	Content Creation	Final Design
J	Integration Testing	Frontend Development, Backend Development, Content Creation
K	Security Implementation	Backend Development
L	Performance Testing	Integration Testing, Security Implementation
M	Beta Release	Performance Testing
N	Marketing Plan Development	User Feedback
O	Bug Fixing	Beta Release
P	User Training and Documentation	Beta Release
Q	Final Release	Bug Fixing, User Training and Documentation

## 4.2. PERT approach

Activity	Predecessor Activity	Optimistic (week)	Most Likely (week)	Pessimistic (week)	Time estimate (week)
A	-	1	2	3	2
B	A	1	2	4	2.17
C	B	1	2	3	2
D	C	1	2	3	2
E	D	2	3	5	3.17
F	E	1	2	3	2
G	E	3	4	6	4.17
H	E, F	3	4	6	4.17
I	E	6	7	8	7
J	G, H, I	2	4	7	4.17
K	H	8	9	10	9
L	J, K	9	12	15	12
M	L	1	3	5	3
N	D	2	5	8	5
O	M	3	6	9	6
P	M	1	3	5	3
Q	O, P	2	6	9	5.83

### 4.3. Network diagram.



- The critical activity =  $A + B + C + D + E + F + H + K + L + M + O + Q$   
 $= 2 + 2.17 + 2 + 2 + 3.17 + 2 + 4.17 + 9 + 12 + 3 + 6 + 5.83 =$   
**53.34 weeks**

## 4.4. Calculator early start and early finish and late start and late finish and free float and total float.

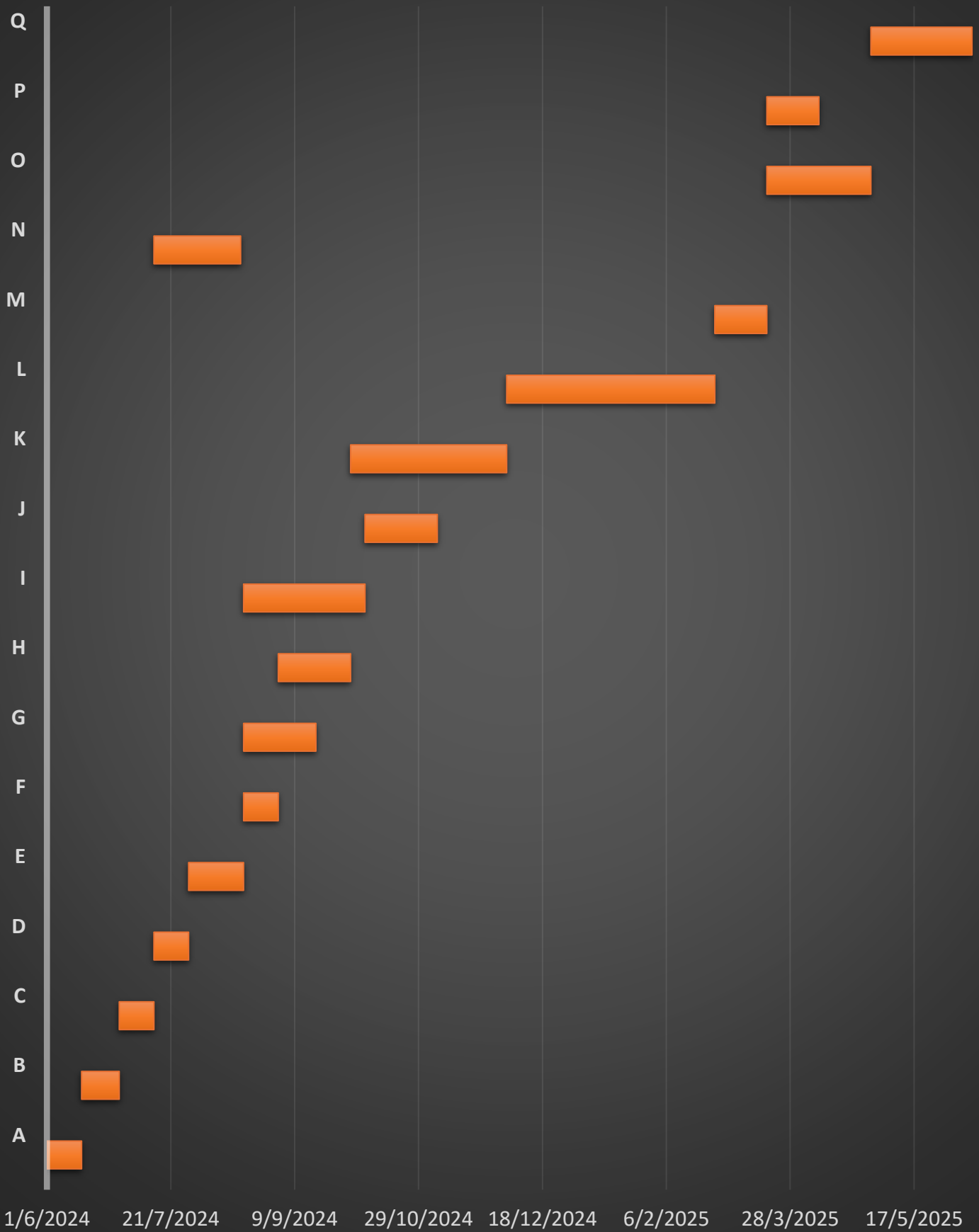
Task	Duration	ES	EF	LS	LF	Free float	Total float	Critical path
A	2	0	2	0	2	0	0	✓
B	2.17	2	4.17	2	4.17	0	0	✓
C	2	4.17	6.17	4.17	6.17	0	0	✓
D	2	6.17	8.17	6.17	8.17	0	0	✓
E	3.17	8.17	11.34	8.17	11.34	0	0	✓
F	2	11.34	13.34	11.34	13.34	0	0	✓
G	4.17	11.34	15.51	18.17	22.34	6.83	6.83	✗
H	4.17	13.34	17.51	13.34	17.51	0	0	✓
I	7	11.34	18.34	15.34	22.34	4	4	✗
J	4.17	17.51	21.68	22.34	26.51	3.38	4.38	✗
K	9	17.51	26.51	17.51	26.51	0	0	✓
L	12	26.51	38.51	26.51	38.51	0	0	✓
M	3	38.51	41.51	38.51	41.51	0	0	✓
N	5	8.17	13.17	1.17	6.17	0	0	✓
O	6	41.51	47.51	41.51	47.51	0	0	✓
P	3	41.51	44.51	44.51	47.51	3	3	✗
Q	5.83	47.51	53.34	47.51	53.34	0	0	✓

## 4.5. Gantt chart

Task	task	Duration	Start data	End data
Requirement Gathering	A	2	1/6/2024	15/6/2024
Concept Design	B	2.17	15/6/2024	30/6/2024
Prototype Development	C	2	30/6/2024	14/7/2024
User Feedback	D	2	14/7/2024	28/7/2024
Final Design	E	3.17	28/7/2024	19/8/2024
Database Design	F	2	19/8/2024	2/9/2024
Frontend Development	G	4.17	19/8/2024	17/9/2024
Backend Development	H	4.17	2/9/2024	1/10/2024
Content Creation	I	7	19/8/2024	7/10/2024
Integration Testing	J	4.17	7/10/2024	5/11/2024
Security Implementation	K	9	1/10/2024	3/12/2024
Performance Testing	L	12	3/12/2024	25/2/2025
Beta Release	M	3	25/2/2025	18/3/2025
Marketing Plan Development	N	5	14/7/2024	18/8/2024
Bug Fixing	O	6	18/3/2025	29/4/2025
User Training and Documentation	P	3	18/3/2025	8/4/2025
Final Release	Q	5.83	29/4/2025	9/6/2025



## Cantt Chart for kids Area Application



## 5. Economic Feasibility Study

Purpose for assessing economic feasibility is to identify the financial benefits and costs associated with the development project.

Economic feasibility is often referred to as cost benefit analysis.

### Financial benefits:

In our management system we are assuming that monetary benefits of our management system at (100,000LE) per year.

And the one-time costs of (50,000LE) and the recurring costs of (40,000LE) a discount rate of 12 percent we are going to calculate the financial benefits over 5 years.

- **The net present value of these cost and benefits of an information system?**
- **The overall return on investment of the project?**
  - **Benefits per year 100,000 LE**
  - **Recurring cost per year = 40,000 LE**
  - **One time cost 50,000 LE**
  - **Discount rate = 12%**

$$PV_n = Y * \frac{1}{(i + 1)^n}$$

**Overall NPV = NPV of all benefits – NPV of all costs**

**Overall, ROL = Overall NPV / NPV of all costs**

	Year of project						
	Year0	Year1	Year2	Year3	Year4	Year5	Total
Net benefits	0	100,000	100,000	100,000	100,000	100,000	_____
Discount rate	1	0.8928	0.7972	0.7118	0.6355	0.5674	_____
PV benefits	0	89,285.71	79,719.38	71,178.02	63,551.80	56,742.68	_____
NPV of all benefits	0	89,285.71	169,005.09	240,183.11	303,734.91	360,477.59	360,477.59
One time cost	50,000	_____	_____	_____	_____	_____	_____
Net economic cost	0	40,000	40,000	40,000	40,000	40,000	_____
Discount rate	1	0.8928	0.7972	0.7118	0.6355	0.5674	_____
PV of costs	0	35,714.2	31,887.7	28,471.2	25,420.7	22,697.1	_____
NPV of all costs	50,000	85,714.2	117,602	146,073.2	171,494	194,191	194,191
Overall NPV	_____	_____	_____	_____	_____	_____	166,286.59
Overall ROI	_____	_____	_____	_____	_____	_____	0.85

- **Present a break-even analysis of, at what point does break-even analysis occur?**

**Yearly PV cash flow= Yearly PV (benefit) – Yearly PV (cost)**

**Yearly overall NPV cash flow = Yearly NPV (benefit) – Yearly NPV (cost)**

Year	Year 0	Year 1	Year 2	Year 3	Year 4	Year5
Yearly PV cash flow	50,000	53,571.51	47,831.68	42,706.82	38,131.10	34,045.58
Yearly overall NPV cash flow	50,000	3,571.51	51,403.09	94,109.91	132,240.91	166,286.59

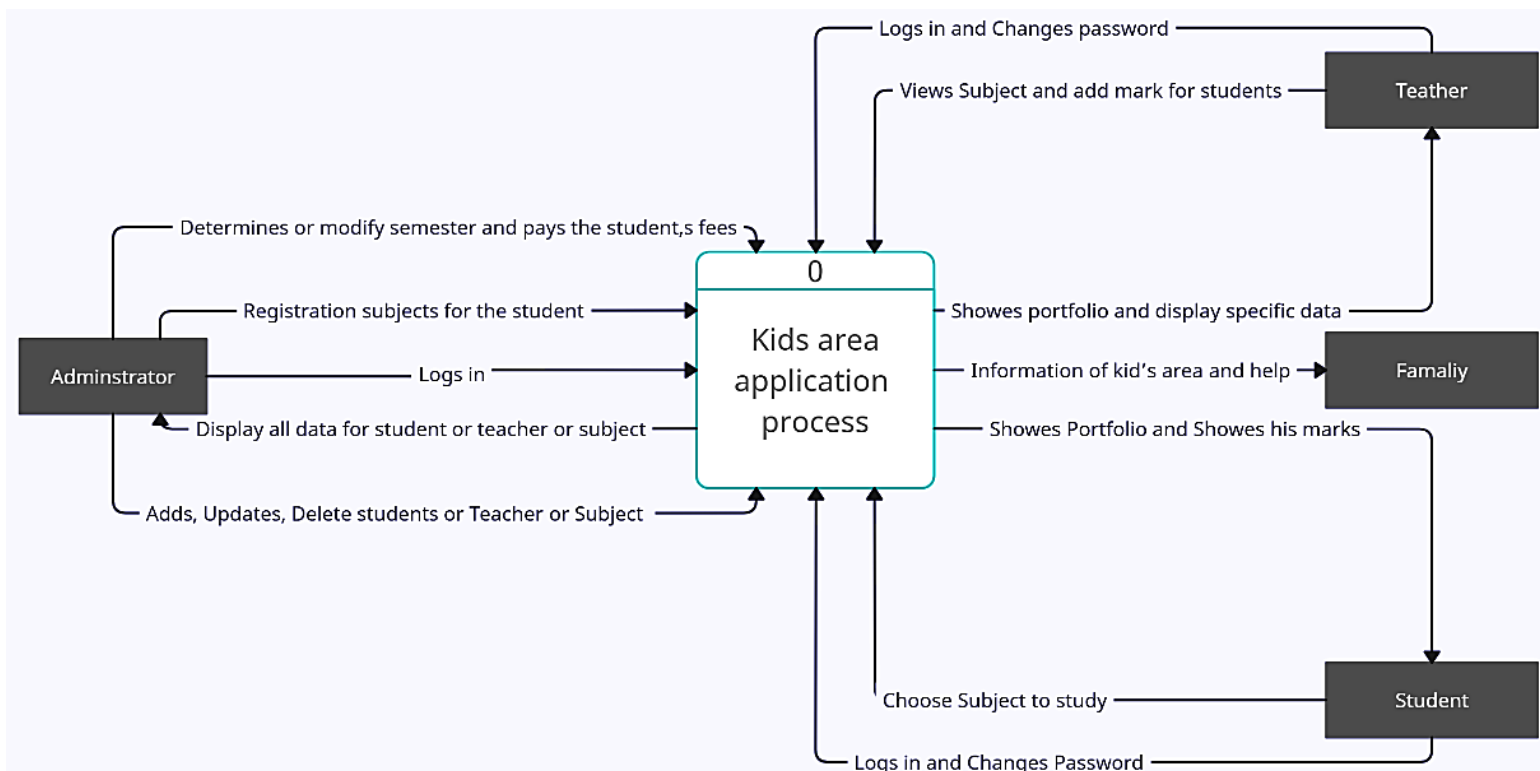
$$\text{Break point} = \frac{(\text{yearly PV cash flow} - \text{yearly overall NPV cash flow})}{\text{yearly PV cash flow}}$$

$$\text{Break point} = \frac{53,571.51 - 3,571.51}{53,571.51} = 0.93$$

Break point occurs at 1.39.

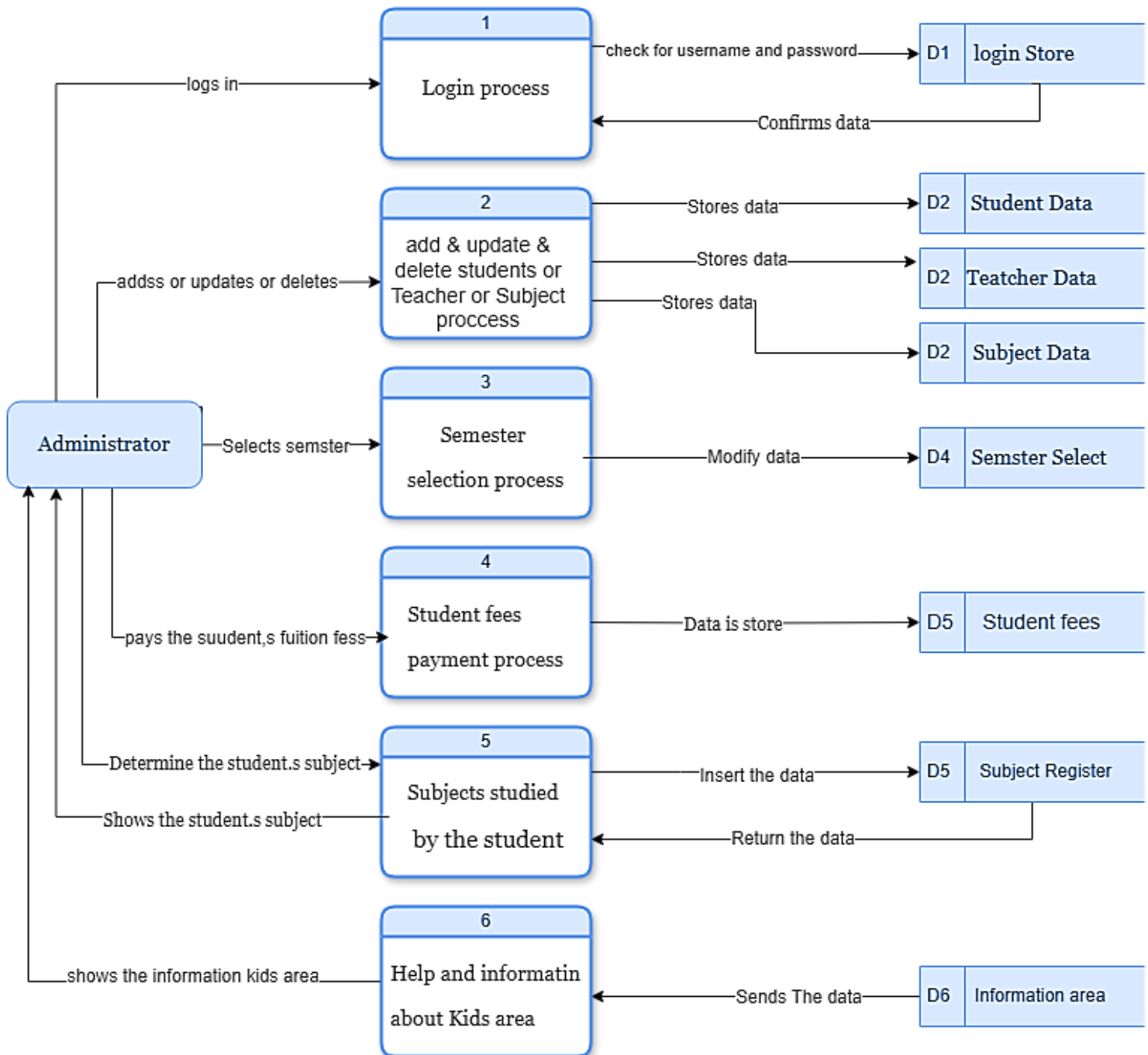
## 6. Data flow diagram

### 6.1. Context diagram (Level 0)

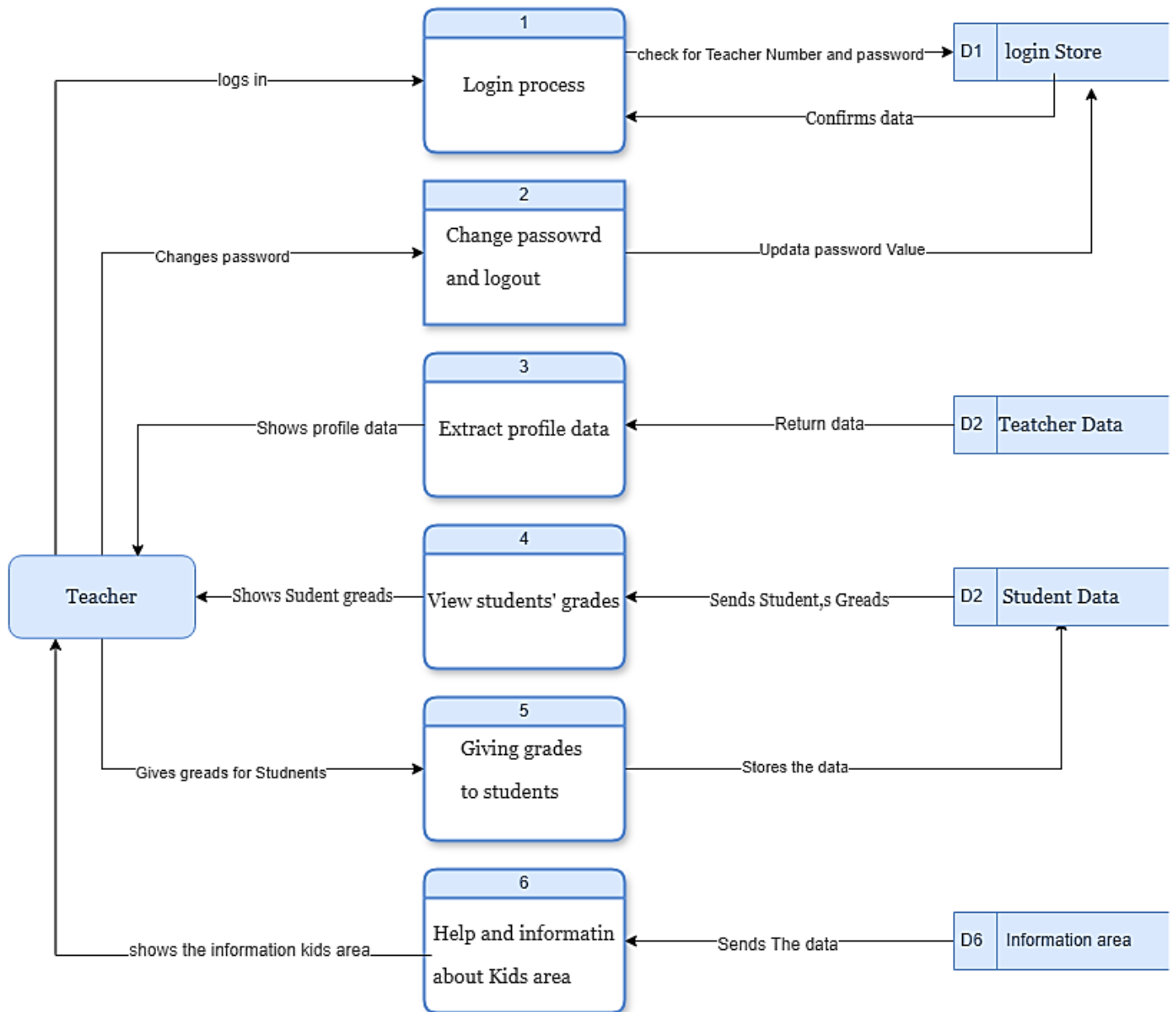


## 6.2. data flow diagram (Level 1)

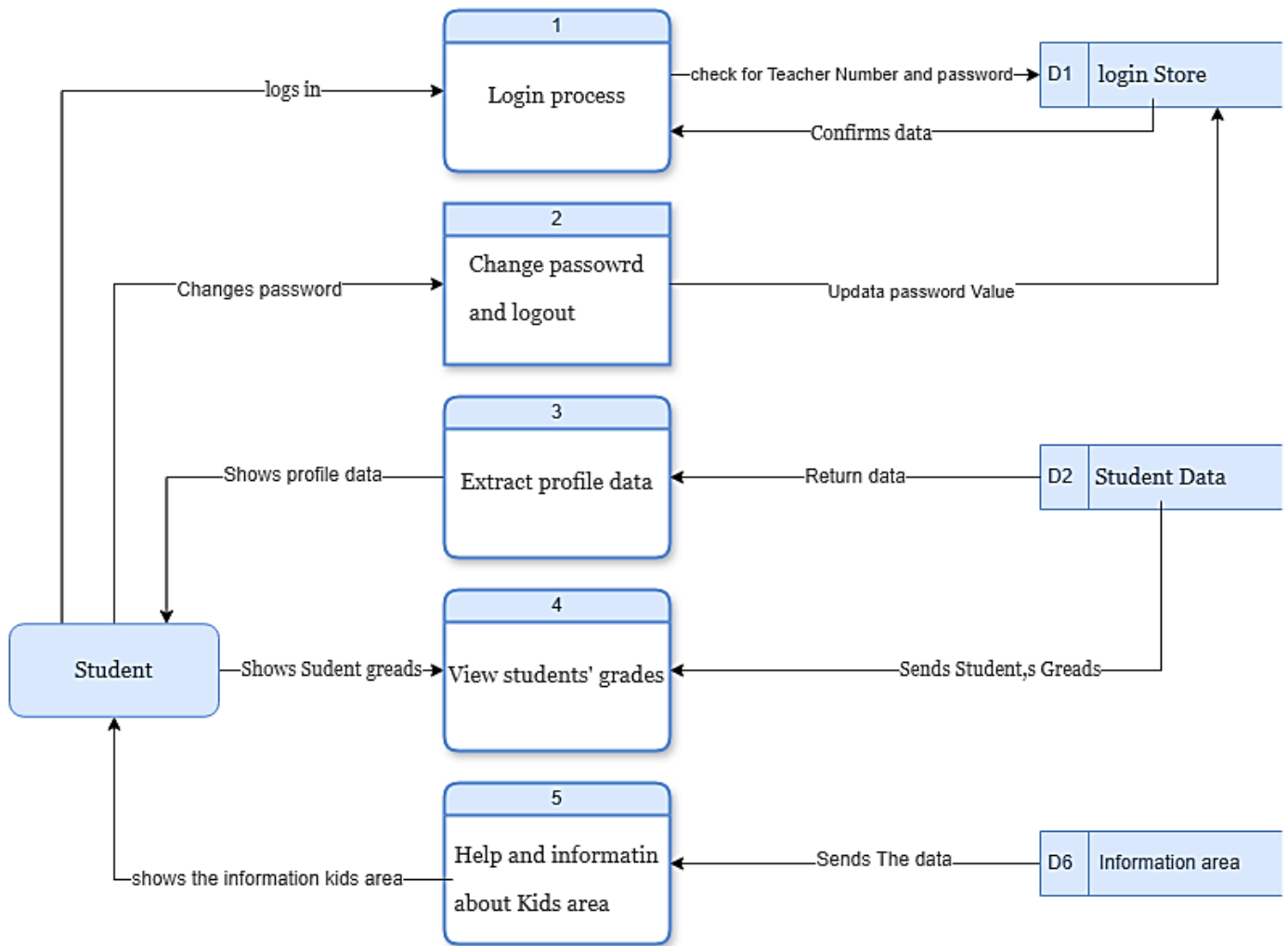
### a. Administrator



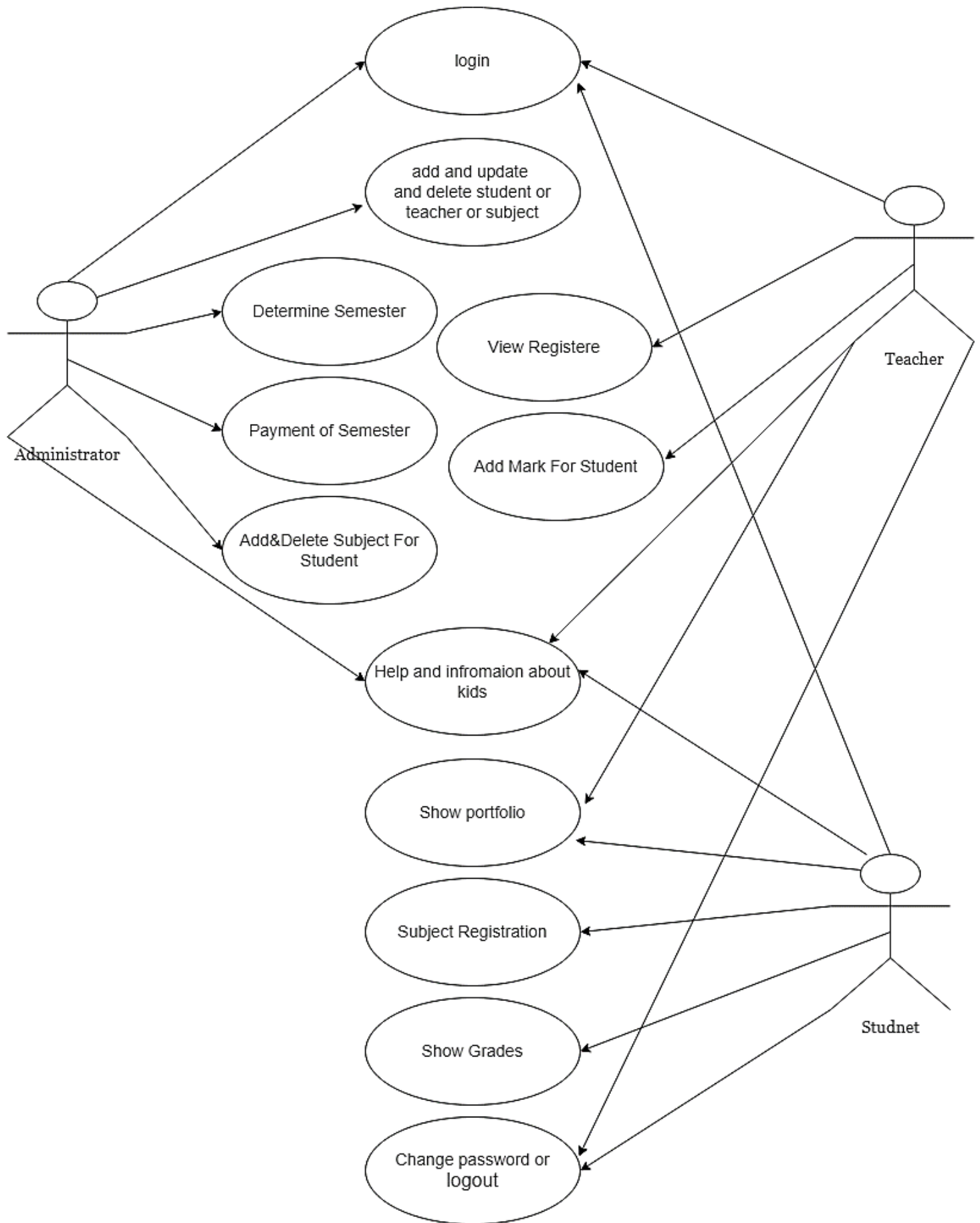
## b. Teacher



## c. Student



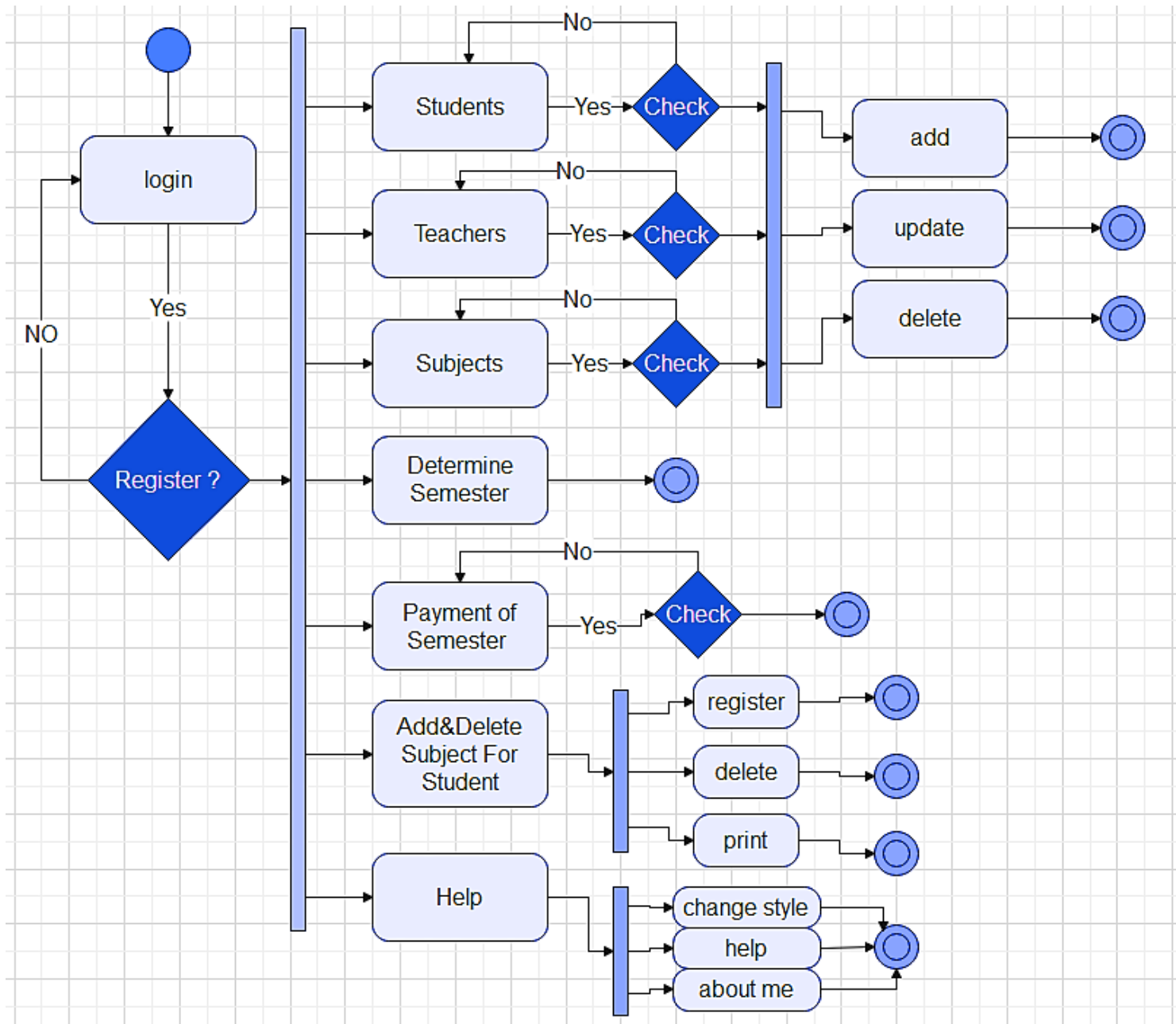
## 7. Use Case





## 8. Activity diagram

### administrator



## 9. Design and implementation.

### Admonitor

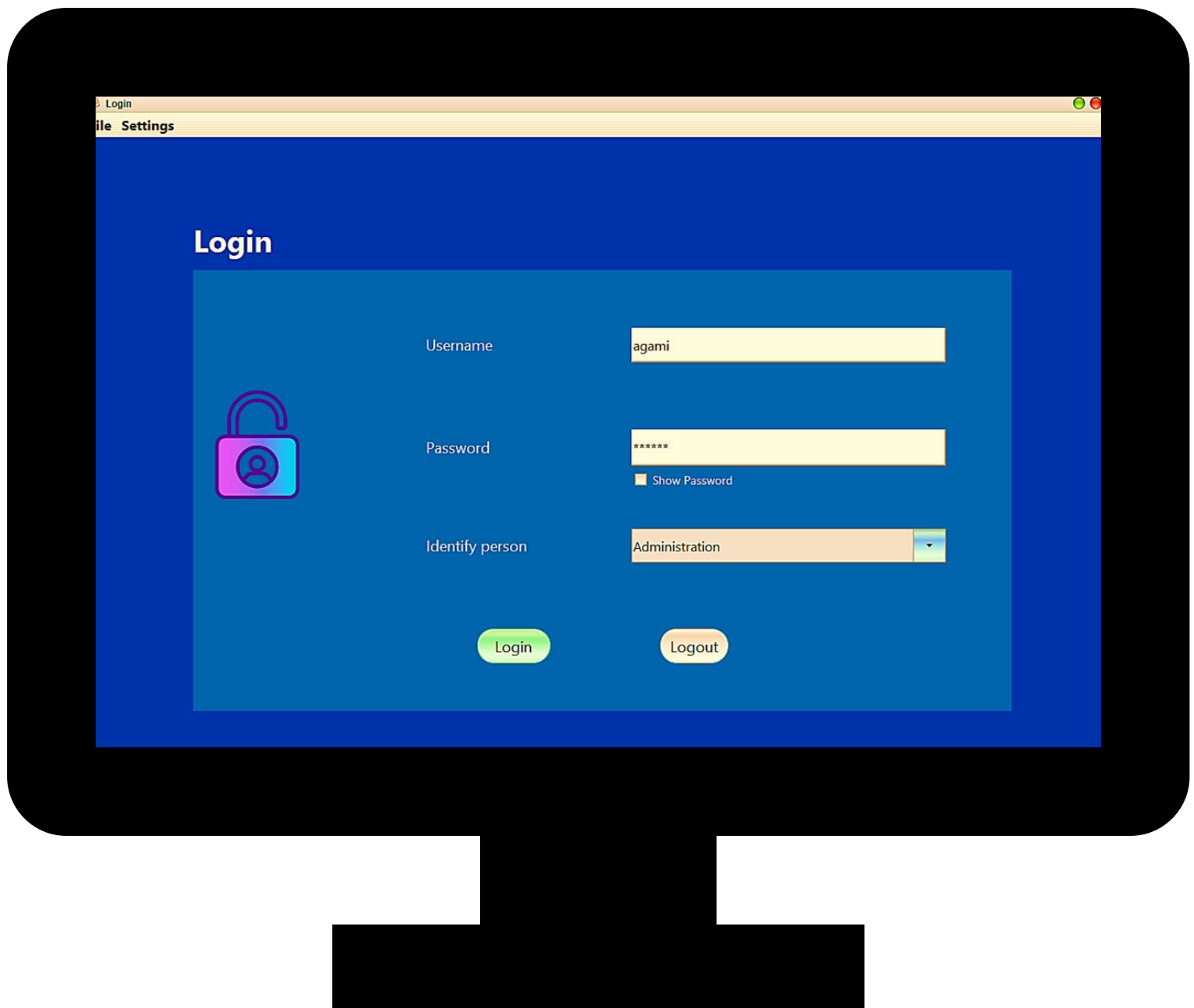
The Admonitor logs in through this screen.

If the Username and password are correct, you can log in.

If it is wrong, he cannot enter.

Username: agami

password: 123456

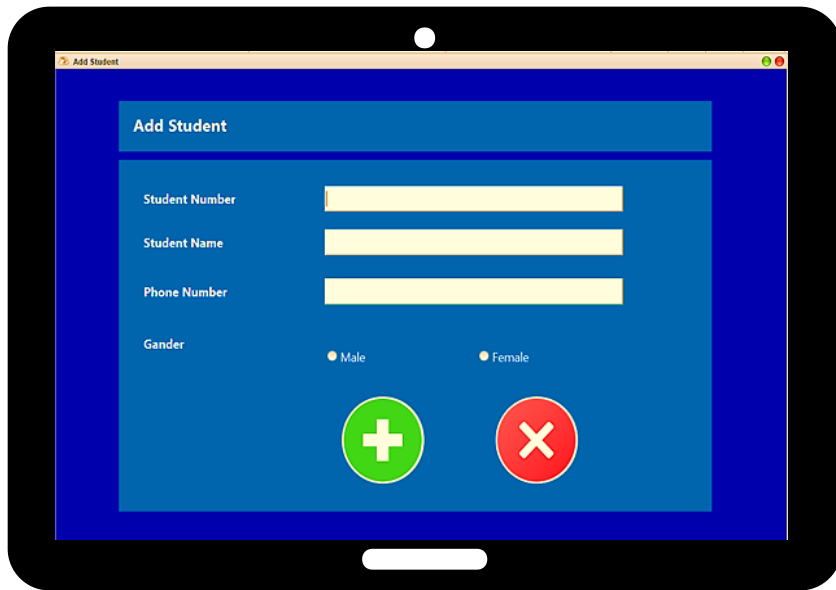


The Administrator's home page will be available to him:

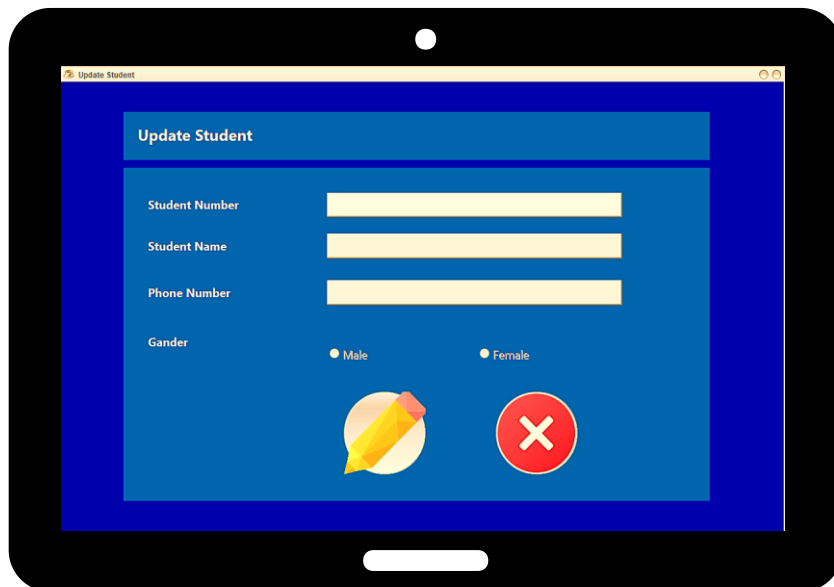
- Add or update or delete Students.
- Add or update or delete Teacher.
- Add or update or delete Subject.
- Determine Semester
- Payment of Semester
- Add or Delete subject for Student.



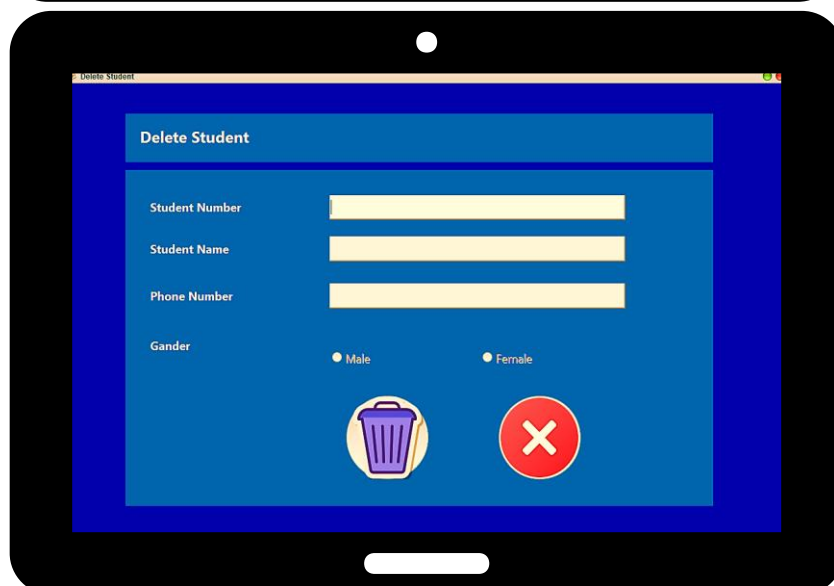
The administrator adds, updates, or deletes the data of a Students, Teachers, or Subject.



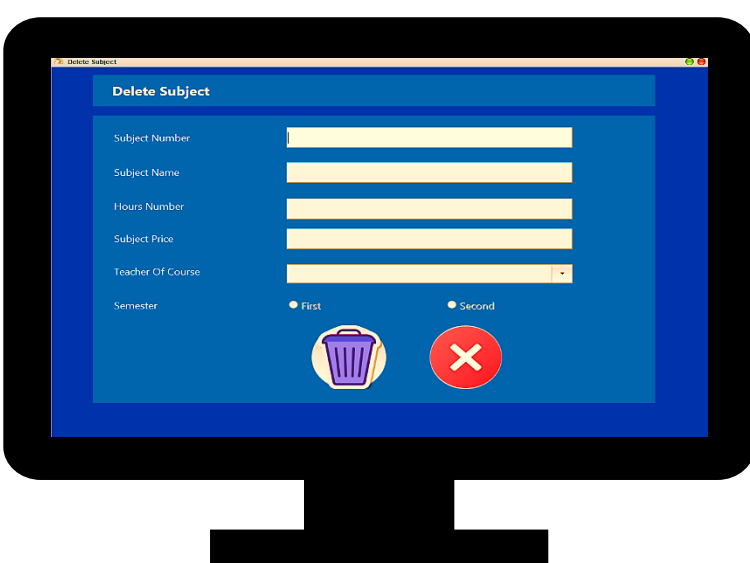
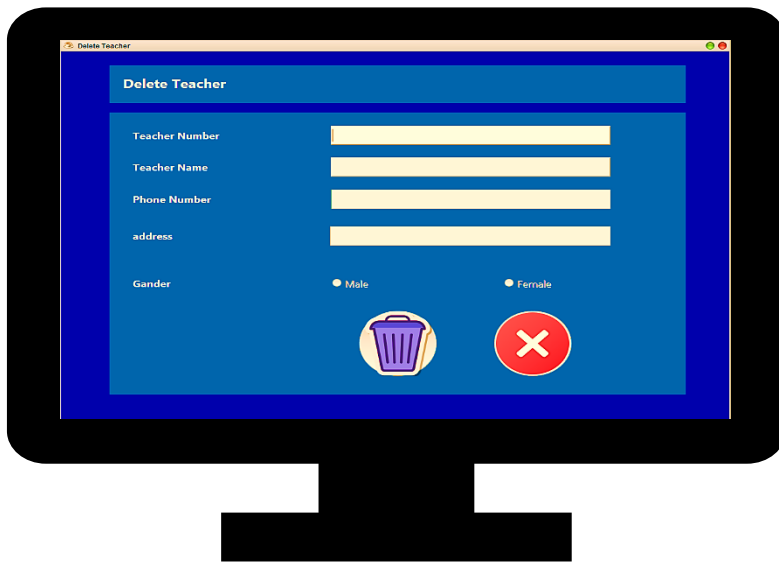
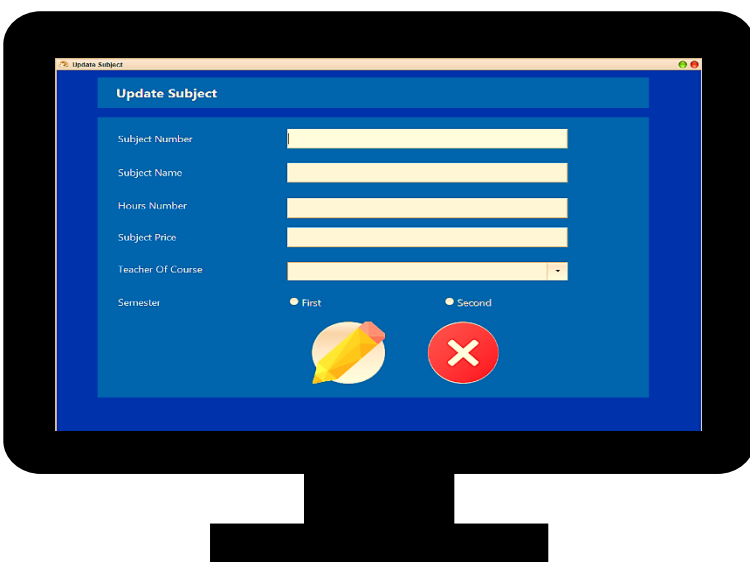
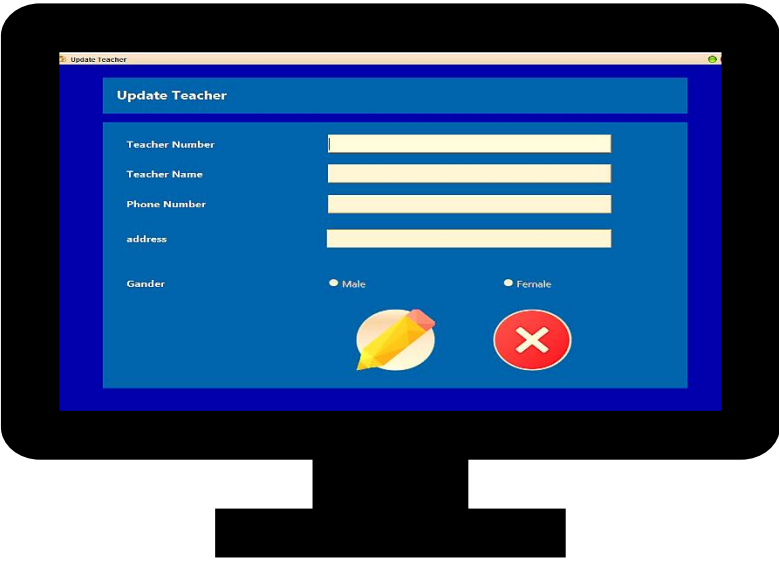
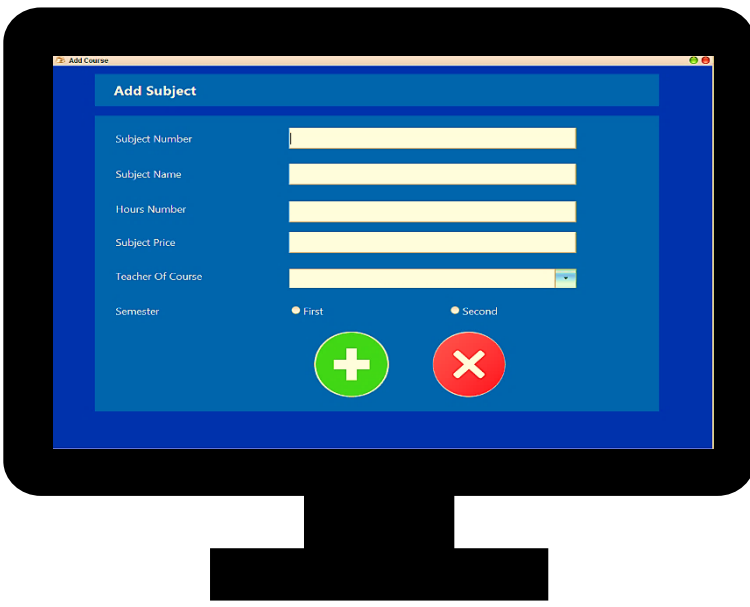
The 'Add Student' form is displayed on a tablet screen. It features a blue header bar with the title 'Add Student'. Below the header, there are four input fields: 'Student Number', 'Student Name', 'Phone Number', and 'Gender'. The 'Gender' field has two radio buttons labeled 'Male' and 'Female'. At the bottom of the form, there are two large circular buttons: a green one with a white plus sign and a red one with a white minus sign.



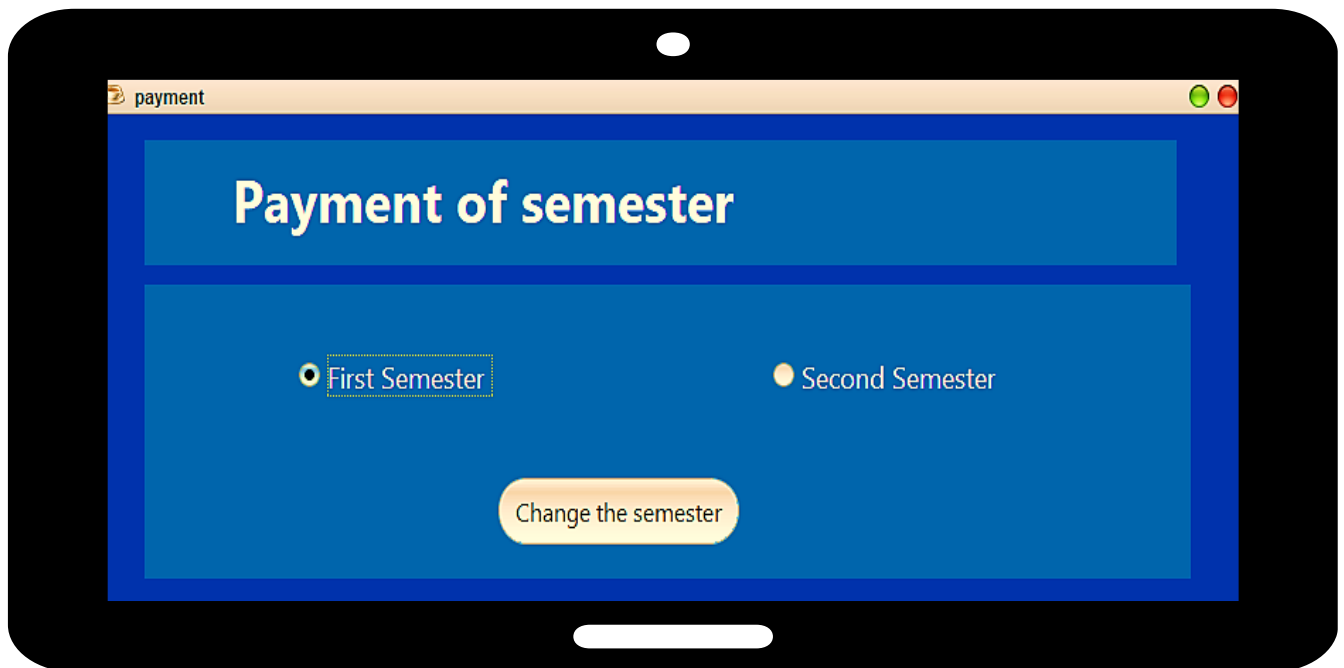
The 'Update Student' form is displayed on a tablet screen. It features a blue header bar with the title 'Update Student'. Below the header, there are four input fields: 'Student Number', 'Student Name', 'Phone Number', and 'Gender'. The 'Gender' field has two radio buttons labeled 'Male' and 'Female'. At the bottom of the form, there are two large circular buttons: a yellow one with a white pencil icon and a red one with a white minus sign.



The 'Delete Student' form is displayed on a tablet screen. It features a blue header bar with the title 'Delete Student'. Below the header, there are four input fields: 'Student Number', 'Student Name', 'Phone Number', and 'Gender'. The 'Gender' field has two radio buttons labeled 'Male' and 'Female'. At the bottom of the form, there are two large circular buttons: a purple one with a white trash can icon and a red one with a white minus sign.



The administrator sets the semester and pays the students' tuition fees.



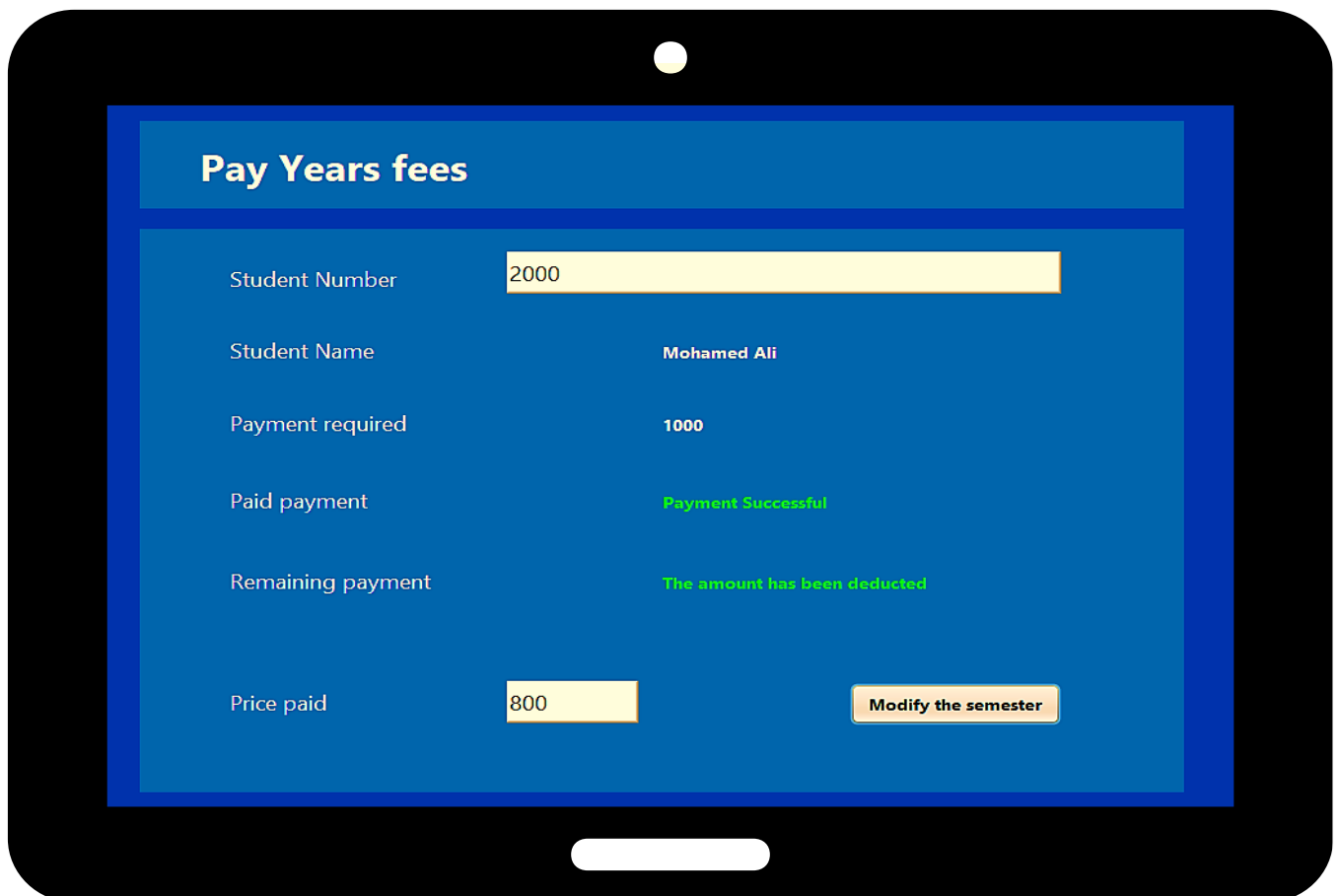
The screenshot shows a tablet with a web browser window titled "payment". The page has a blue background and a white header area with the title "Payment of semester". Below the header, there are two radio buttons: "First Semester" (selected) and "Second Semester". A yellow button labeled "Change the semester" is positioned below the radio buttons.

payment

## Payment of semester

☒ First Semester ☐ Second Semester

Change the semester



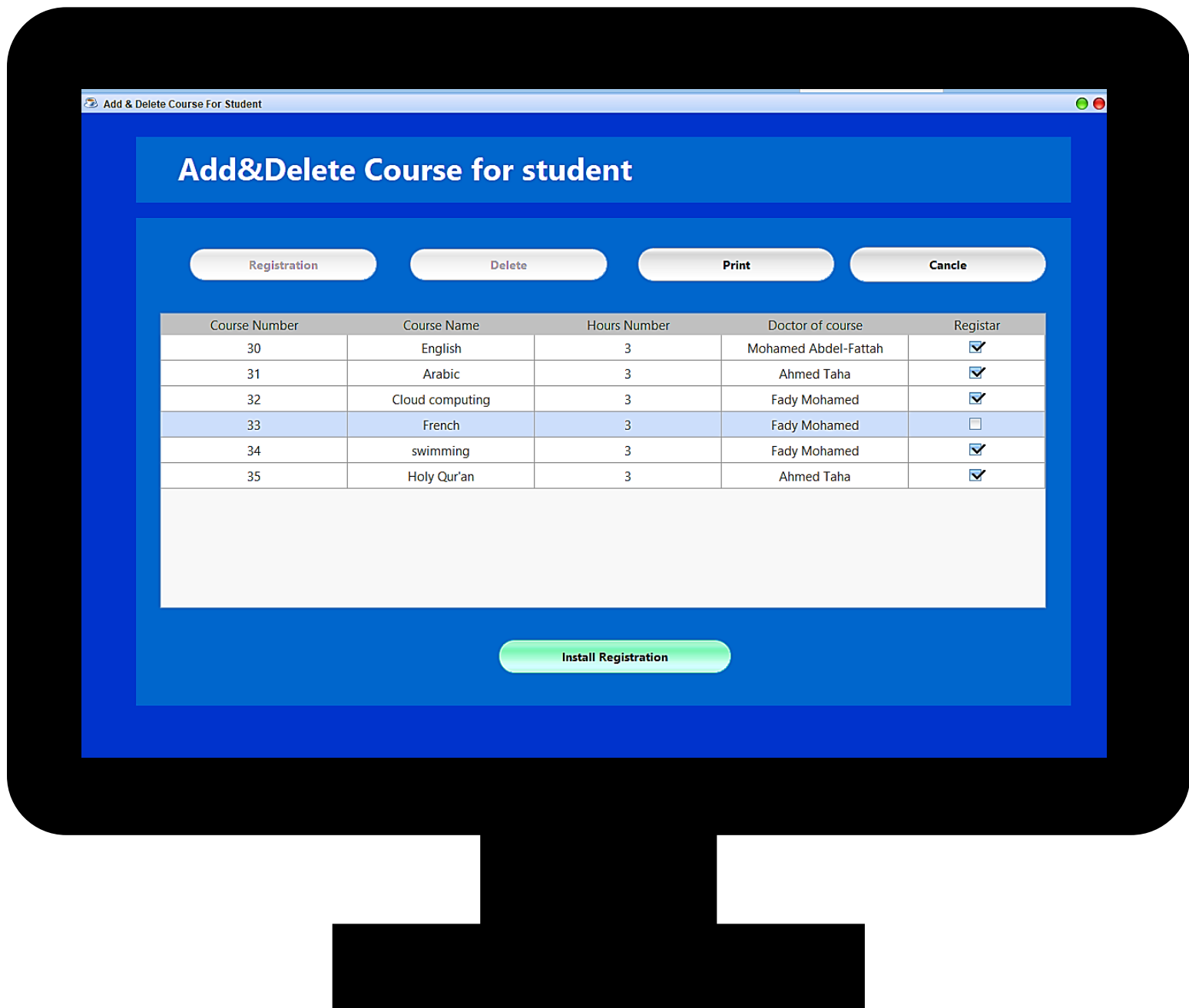
The screenshot shows a tablet with a web browser window titled "Pay Years fees". The page has a blue background and a white header area with the title "Pay Years fees". Below the header, there is a form with several fields and a button. The fields are: "Student Number" (value: 2000), "Student Name" (value: Mohamed Ali), "Payment required" (value: 1000), "Paid payment" (value: Payment Successful), "Remaining payment" (value: The amount has been deducted), and "Price paid" (value: 800). A yellow button labeled "Modify the semester" is positioned to the right of the "Price paid" field.

## Pay Years fees

Student Number	2000
Student Name	Mohamed Ali
Payment required	1000
Paid payment	Payment Successful
Remaining payment	The amount has been deducted
Price paid	800

Modify the semester

The director presents the subjects the student is required to study.



# Teachers

The Teacher logs in through this screen.

If the Teacher Number and password are correct, you can log in.

If it is wrong, he cannot enter.

**Teacher Number: 111**

**password: 000**



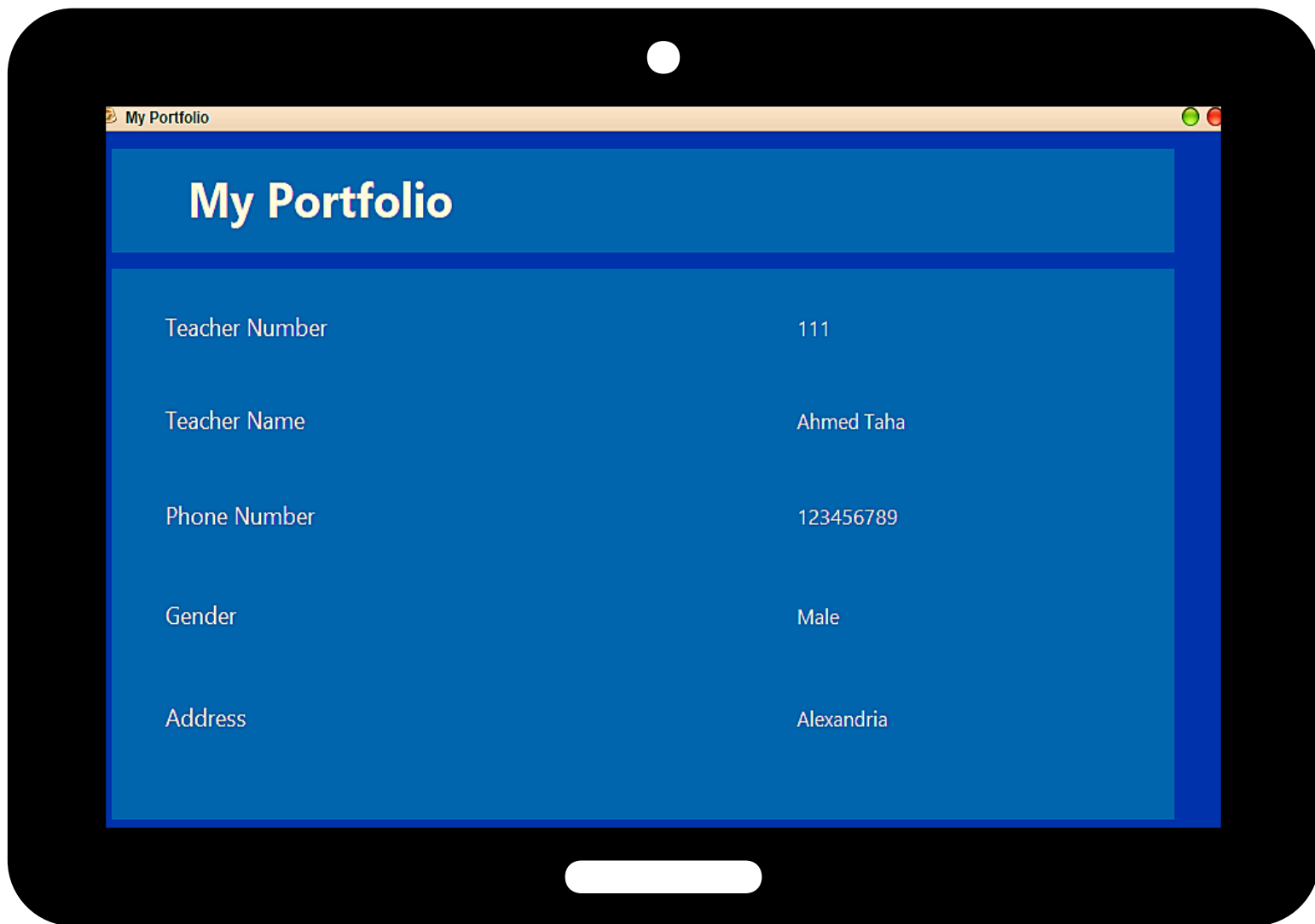


The Teacher's home page will be available to him:

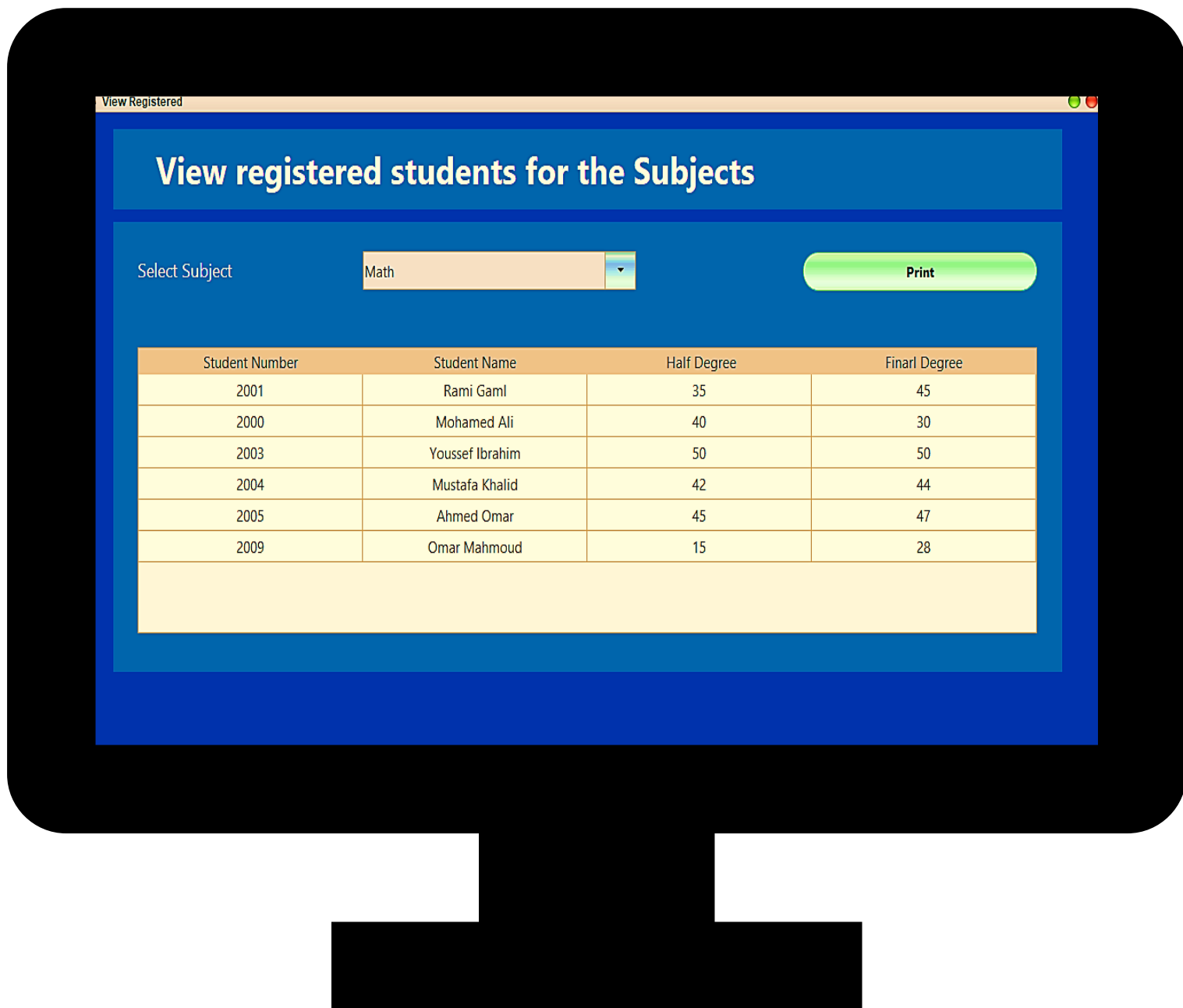
- Show your portfolio.
- View registered students for the Subjects.
- Adds Student Greats for Students
- Change Password
- Help and logout



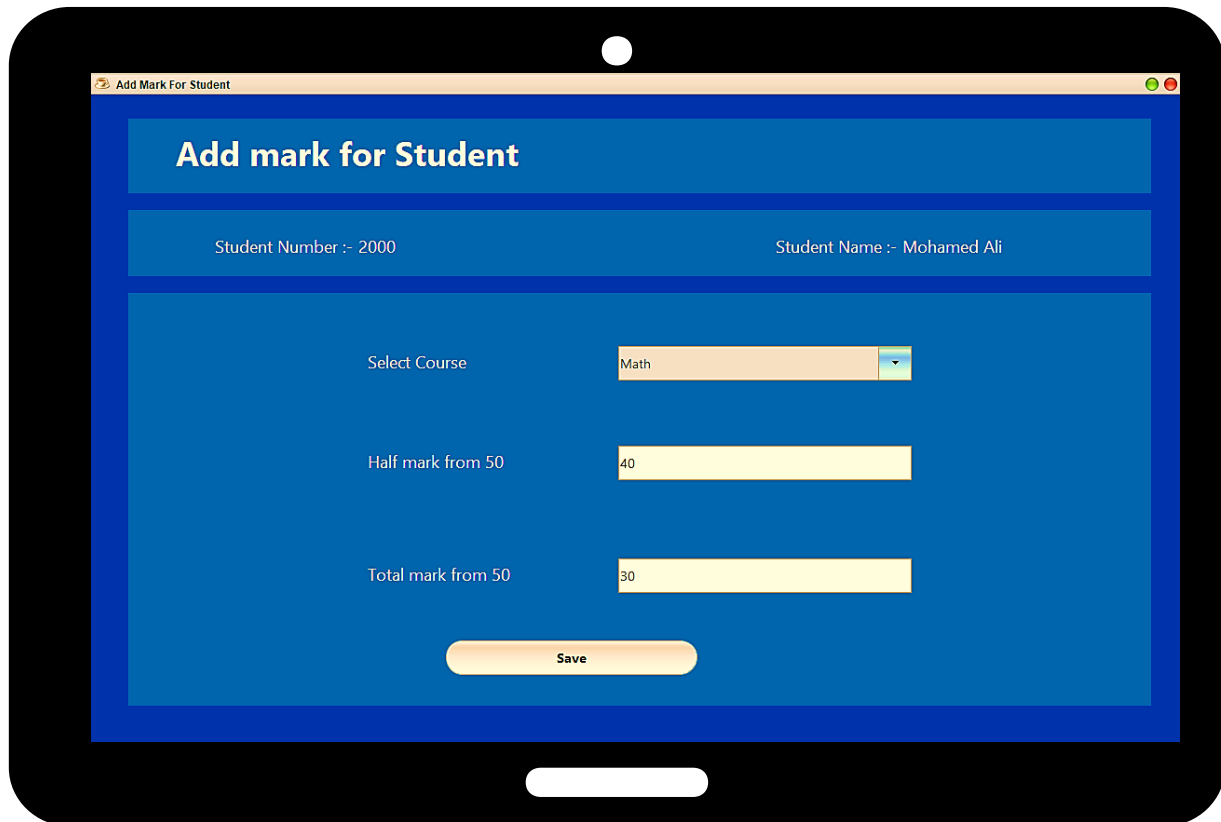
Through this page, the teacher can view his own page and the data it contains.



Through this screen, the teacher tallies the grades of all students in all subjects taught by this teacher.



Through this page, the teacher can add student grades and change the password.



The screenshot shows a web application titled "Add Mark For Student" running on a tablet. The interface has a blue background. At the top, there's a header bar with the title. Below it, two fields display "Student Number :- 2000" and "Student Name :- Mohamed Ali". The main area contains three input fields: "Select Course" with a dropdown menu showing "Math", "Half mark from 50" with a text input containing "40", and "Total mark from 50" with a text input containing "30". A yellow "Save" button is at the bottom.

Field	Value
Student Number	2000
Student Name	Mohamed Ali
Select Course	Math
Half mark from 50	40
Total mark from 50	30



The screenshot shows a web application titled "Change Password for Doctor" running on a tablet. The interface has a blue background. It contains four input fields: "Identification Number" with "111", "Old Password" with "...", "New Password" with "...", and "Confirm Password" with "...". A green message "The passwords are identical" is displayed below the "Confirm Password" field. At the bottom, there are two buttons: a yellow "Change Password" button and a white "Exit" button.

Field	Value
Identification Number	111
Old Password	...
New Password	...
Confirm Password	...

# Student

The Teacher logs in through this screen.

If the Student Number and password are correct, you can log in.

If it is wrong, he cannot enter.

**Teacher Number: 200**

**password: 123**



The student's home page will be available to him:

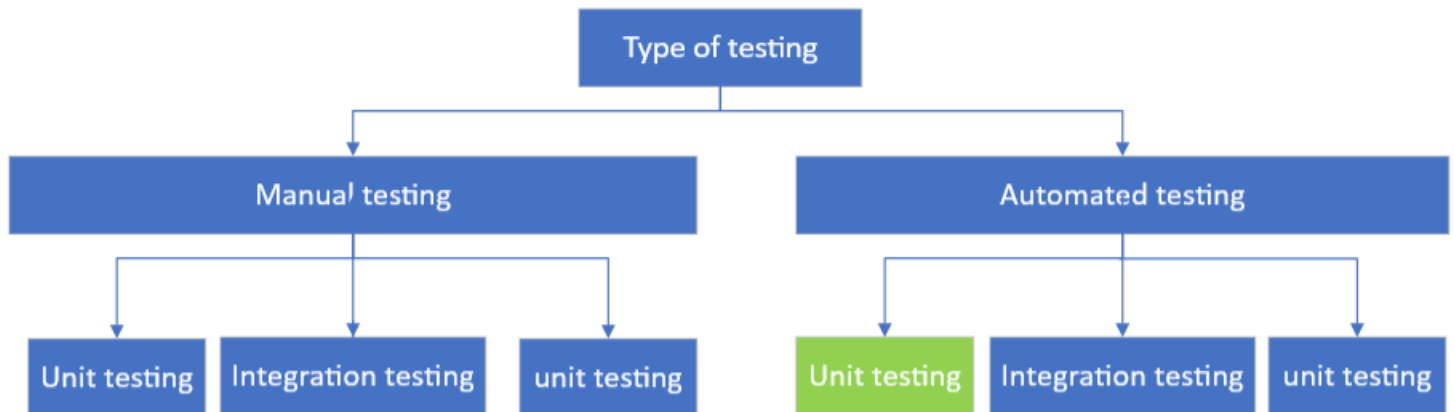
- Show your portfolio.
- View subject Registration.
- View his grades.
- Change Password
- Help and logout.





# 10. Testing

## White box testing



## function administration login

```
import org.junit.jupiter.api.BeforeEach;
import org.junit.jupiter.api.Test;
import org.mockito.InjectMocks;
import org.mockito.Mock;
import org.mockito.MockitoAnnotations;
import javax.swing.*;
import static org.mockito.Mockito.*;

class AdminLoginTest {
    @InjectMocks
    private AdminLoginClass adminLogin; // Replace AdminLoginClass with the actual class name containing the AdminLogin method
    @Mock
    private JTextField id;
    @Mock
    private JPasswordField password;
    @Mock
    private MainMenu mainMenu;
    @BeforeEach
    void setUp() {
        MockitoAnnotations.openMocks(this);
    }
    @Test
    void testAdminLoginSuccess() {
        when(id.getText()).thenReturn("agami");
        when(password.getText()).thenReturn("123456");

        adminLogin.AdminLogin();

        verify(mainMenu.Doctor).setEnabled(false);
        verify(mainMenu.Student).setEnabled(false);
        verify(mainMenu).setVisible(true);
        verify(adminLogin).setVisible(false);
        verifyStatic(JOptionPane.class);
        JOptionPane.showMessageDialog(null, "Welcome professor AGAMI");
    }
    @Test
    void testAdminLoginFailure() {
        when(id.getText()).thenReturn("wrongId");
        when(password.getText()).thenReturn("wrongPassword");

        adminLogin.AdminLogin();

        verifyStatic(JOptionPane.class);
        JOptionPane.showMessageDialog(null, "The username or password is incorrect", "Data validation", JOptionPane.ERROR_MESSAGE);
    }
}
```

## Project files

