منصة حافظ

حافظ - منصة إدارة حلقات تحفيظ القرآن

Hafez - A platform for managing Quran memorization circles

منصّة رقمية تمثل جسراً مبتكراً، لتطوير تجربة تحفيظ القرآن الكريم.

Jubail Industrial College

Department of Business Administration

Semester 462

Final Report

Course Name:		**Capstone Project**
Course Code:		**MIS 451**
Section:		**101**
No.	**Student ID**	**Student Name**
1	**392902969**	**Mohammed Alhammad**
2	**441101144**	**Salem Almahri**
3	**401902187**	**Aseeel Alghamdi**
4	**431100087**	**Sulaiman Alfuhaid**
5	**421900615**	**Ibraheem Alnahdi**
6	**401900570**	**Hatem alqahtani**
Group:		**3**
Supervised by:		**ABDULRAHMAN BANDAR ALHARBI**

Acknowledgement

Firstly, we thank God for His guidance and blessings, which enabled us to achieve this success and win first place in the MIS exhibition.

We also extend our sincere gratitude to our esteemed professor, **Mr**. Abdulrahman Al-Harbi, for his invaluable efforts, insightful recommendations, and unwavering support throughout our graduation project, which significantly contributed to our victory

Executive Summary

The Hafez System is a comprehensive web-based platform designed to transform the Quran memorization experience for students, teachers, and parents. It addresses critical challenges in tracking student memorization progress, attendance monitoring, and creating positive competition environments in Quran study circles. The platform provides real-time progress tracking, personalized reports, and a gamified learning experience through leaderboards and point systems.

Core functionalities of the system include:

- Student enrollment and attendance tracking in study circles
- Comprehensive evaluation and progress monitoring
- Automated reporting system for students and parents
- Competitive leaderboards to motivate student achievement
- Direct communication channels between teachers and parents

The system was successfully implemented using modern web technologies (HTML/CSS/JavaScript frontend, PHP backend, MySQL database) following a structured software development lifecycle approach. The project has demonstrated significant potential to enhance the Quran memorization experience by improving tracking accuracy, increasing student motivation, and strengthening communication between all stakeholders.

Introduction

Background

Traditional Quran memorization centers face significant challenges in accurately tracking student progress, maintaining consistent attendance records, and keeping parents informed about their children's development. Additionally, these centers often lack structured means to foster positive competition among students, missing a valuable motivational tool for learning.

Problem Statement

Students, Quran teachers, and parents struggle with:

Limited visibility into memorization progress and comparison metrics
Inconsistent attendance tracking mechanisms
Lack of structured communication between teachers and parents
Absence of motivational tools to encourage consistent memorization efforts

Project Objectives

The Hafez System aims to:

Create a centralized platform for tracking student memorization progress
Improve communication between teachers, students, and parents
Implement a points-based competitive system to motivate students
Provide comprehensive reporting tools for all stakeholders
Enhance the overall Quran memorization experience through technology

Scope

The system encompasses:

User management for teachers, students, and parents
Study circle administration
Attendance recording and reporting
Evaluation tracking and feedback
Point-based leaderboards
Notification systems
Comprehensive reporting tools

SDLC Phases

1. Initiation Phase

During the initiation phase, the project team was formed, and the project idea was conceptualized based on identified needs in Quran memorization centers. Key activities included:

Team formation with six members having complementary skills
Initial project scope definition
Preliminary feasibility assessment
Stakeholder identification and engagement planning

2. Planning Phase

The planning phase established a solid foundation for the project through detailed documentation and analysis:
System Vision Document Development:

Problem description was articulated, highlighting challenges faced by students, teachers, and parents.
System capabilities were defined, focusing on attendance tracking, progress monitoring, and competitive features.
Business benefits were identified, emphasizing improved communication, tracking accuracy, and motivation.

Project Management:

Detailed Work Breakdown Structure (WBS) was created
Cost-benefit analysis was conducted, identifying development costs (12,000 SAR), annual hosting costs (970 SAR), marketing costs (4,500 SAR), and maintenance costs (6,000-18,000 SAR annually)
Project timeline and milestones were established

3. Analysis Phase

The analysis phase focused on understanding system requirements and developing conceptual models:

Business Model Development:

Stakeholder analysis and needs assessment
User stories and requirements gathering
Business rules identification

Data Modeling:

Entity-Relationship Diagram (ERD) development defining key entities: Users, Family Members, Teachers, Students, Study Circles, Attendance, Evaluations, Reports, and Leaderboards
Relationship mapping between entities
Business rules documentation

Process Modeling:

Use case diagram creation showing system interactions
Detailed case descriptions for all major system functions
Process workflow definition.

4. Design Phase

The design phase translated requirements into technical specifications:

Application Architecture:

Three-tier client-server architecture (view layer, business layer, data layer)
Front-end development specifications using HTML, JavaScript, and CSS
Back-end development specifications using PHP and SQL
Cloud-based infrastructure design

Network Architecture:

Network diagram development showing system communication flows
Cloud firewall implementation planning
Web, application, and database server configurations.

System Controls:

Integrity controls implementation (input controls, transaction logging, complex update control, backup and recovery, output control)
Security controls design (authentication, authorization, secure transactions, data encryption).

User Interface Design:

Wireframe development for all system screens
Navigation flow design
Responsive design principles implementation

5. Implementation & Testing Phase

The implementation phase brought the design to life through coding and testing:

System Development:

Front-end development using HTML, CSS3, and JavaScript
Back-end development using PHP
Database implementation using MySQL
Integration of system components

Testing:

Unit testing of individual functions
Integration testing of connected components
System testing of end-to-end workflows
User acceptance testing with stakeholders

System Reports Generation:

Development of memorization reports
Implementation of attendance tracking reports
Creation of points and ranking reports
Teacher evaluation report generation

6. Project Evaluation

The final phases included:

Team Evaluation:

Peer evaluation process
Performance assessment

Final Documentation:

Comprehensive project report compilation
Technical documentation finalization

System Demonstration:

Presentation slide preparation
Final presentation
Live system demonstration planning

System Vision Document

Problem Description:

Students, Quran teachers, and parents face challenges in tracking a student's memorization progress, understanding their level compared to peers, and monitoring attendance records. Additionally, the concept of positive competition within Quran memorization circles is either absent or implemented in a complex manner.
An advanced platform which resolves these problems will greatly improve the learning process for both educators and students as well as their parents.

System Capabilities:

- Allowing teachers to register their students, record attendance, and track memorization progress.
- Enables both students and parents to access statistics and reports, monitor memorization progress, and identify areas that need improvement.
- Introduces monthly competitions within each study circle based on attendance and recitation scores, recognizing top-performing students.
- Through its teacher portal, students' memorization progress is monitored in real-time at an efficient pace. The system allows teachers to assign new learning objectives and revision activities to students who must adhere to a systematic educational sequence.

Business Benefits:

- ❖ The system enhances the communication triangle between teachers, students, and parents by providing an accessible web-based platform.
- ❖ Improves tracking accuracy and ensures documented and measurable progress.
- ❖ The system fosters a motivational competitive environment by displaying students' progress and points regularly.
- ❖ This web-based approach ensures broader accessibility and ease of use for all stakeholders.
- ❖ The platform implements rewards, badges, and leaderboards to stimulate student learning activities.

Cost-Benefit Analysis

1. Development Costs:

Design Cost (One-Time): Design for the website and branding (identity).
Estimated cost: 600 - 1200 SAR.
Programming Cost (One-Time): Development of the website and necessary functionalities.
Estimated cost: 10,000 - 12000SAR.

2. Hosting & Domain Costs (Annual):

Domain Name (Annual Fee): Website address.
Estimated cost: 70 SAR per year.
Web Hosting (Monthly Fee): Cloud VPS with medium performance.
Estimated cost: 75 SAR per month.
Estimated annual cost: 900 SAR.

3. Marketing Costs (Monthly):

Social Media Marketing (Monthly Fee): Advertising via social media channels.
Estimated cost: 375 SAR per month.
Estimated annual cost: 4,500 SAR.

4. Maintenance & Operations Costs (Monthly):

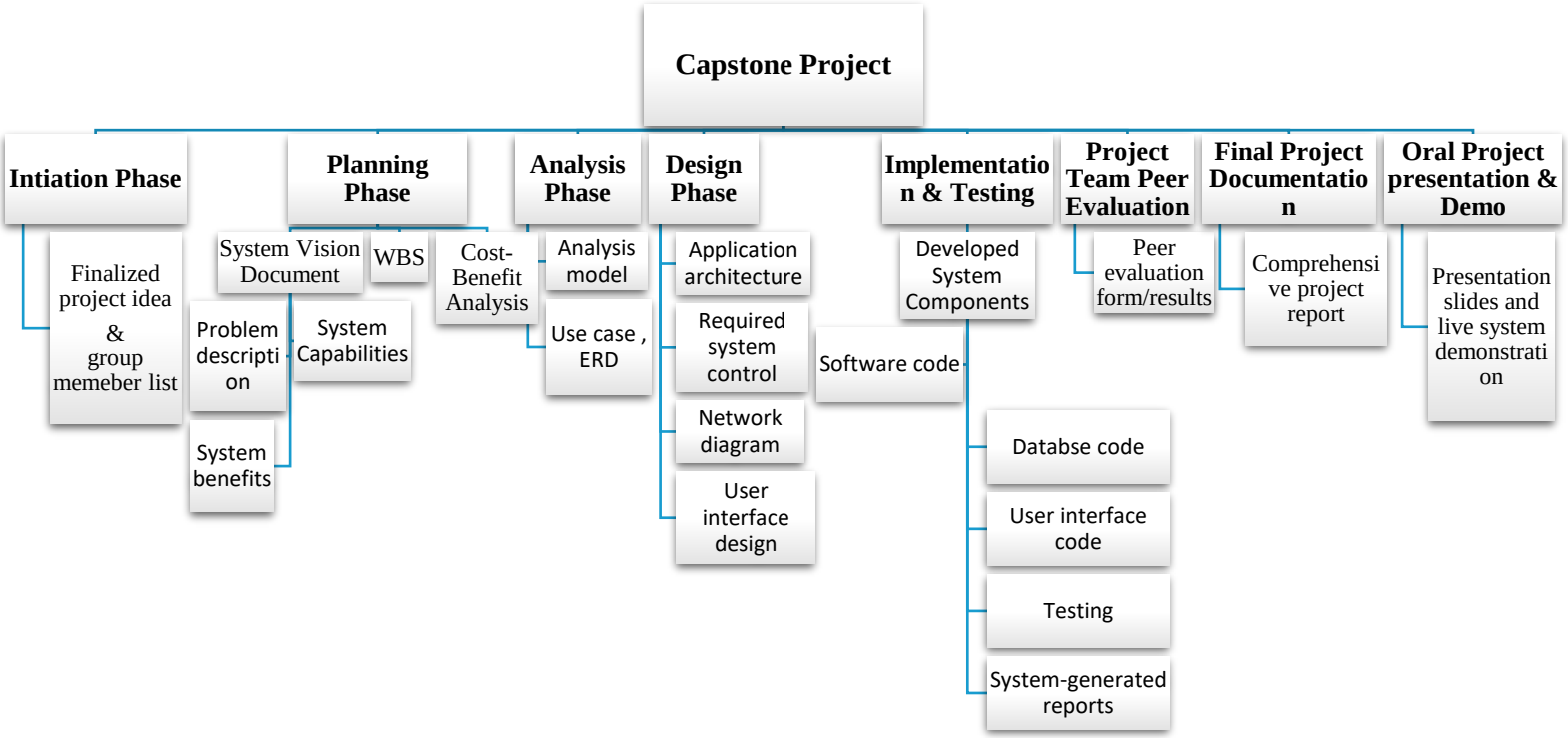
Maintenance & Operations (Monthly Fee): Ongoing maintenance and operation of the website (bug fixes, system updates, general technical support).
Estimated cost: 500 - 1,500 SAR per month.
Estimated annual cost: 6,000 - 18,000 SAR.

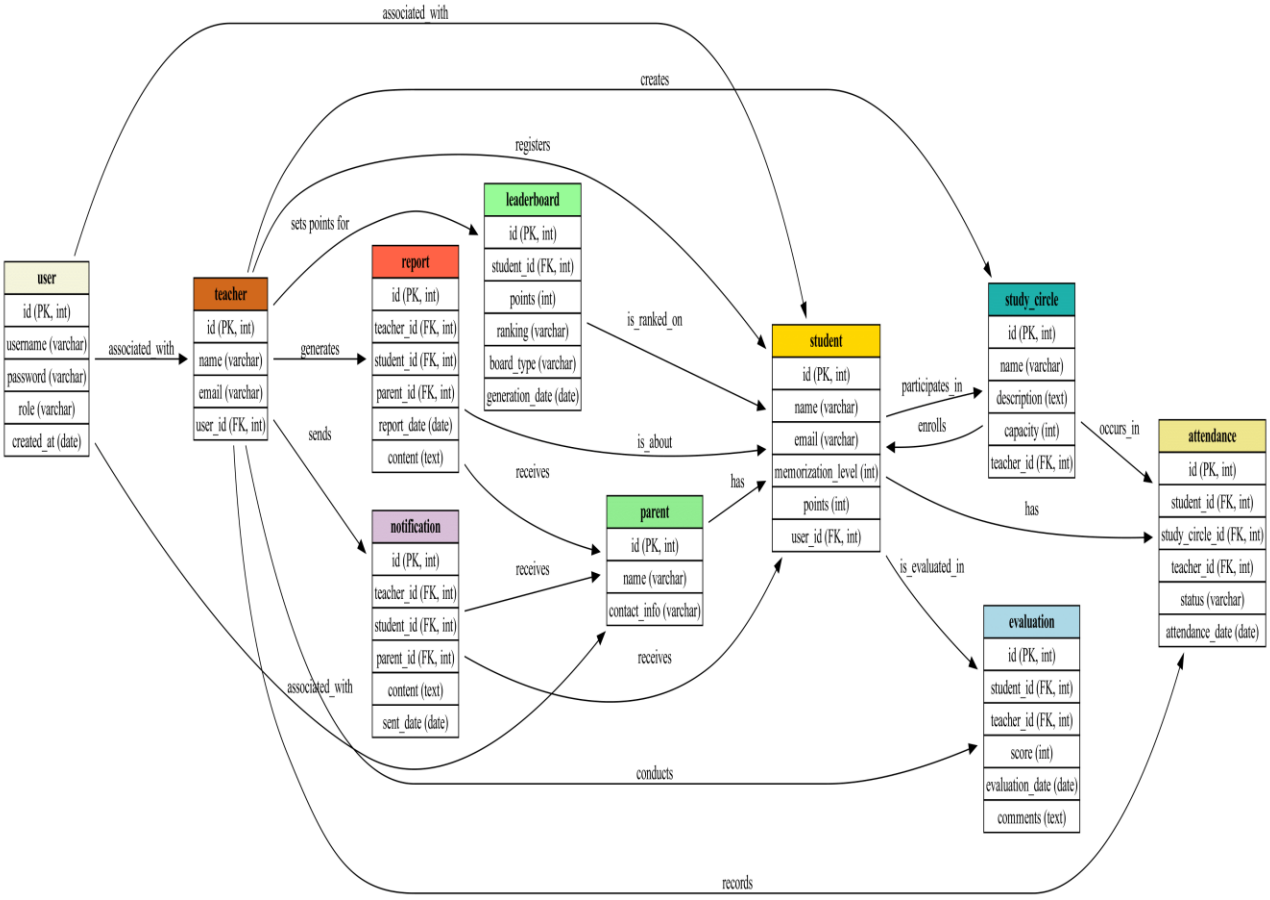
Non-Profit Approach:

The project is intended to be non-profit, and low-cost subscription plans will be adopted for students, teachers, and parents to make the platform accessible to as many users as possible.

WBS	Task	Submission Week	Start Date
1	Initiation Phase	Week 1	2025-01-15
1.1	Team formation list		
1.2	Project idea		
2	Planning Phase	Week 2	2025-01-23
2.1	System vision document		
2.1.1	Problem description		
2.1.2	System capabilities		
2.1.3	System benefits		
2.2	Work breakdown structure WBS		
2.3	Cost-benefits analysis		
3	Analysis Phase	Week 4	2025-01-30
3.1	Business model		
3.2	ERD, Use case diagram		
4	Design Phase	Week 6	2025-02-12
4.1	Application architecture		
4.2	Required system control		
4.3	Network diagram		
4.4	User interface diagram		
5	Implementation & Testing Phase	Week 14	2025-04-09
5.1	Develop system component		
5.2	Software code		
5.3	Database code		
5.4	User interface code		

5.5	Testing		
5.6	System-generated report		
6	Project Team Peer Evaluation	Week 14	2025-04-09
6.1	Peer evaluation form / result		
7	Final Project Documentation	Week 15	2025-04-16
7.1	Comprehensive a project report		
8	Oral Presentation & Demo	Week 15	2025-04-16
8.1	Presentation slides & live system demonstration		




user
id (PK, int)
username (varchar)
password (varchar)
role (varchar)
created_at (date)
teacher
id (PK, int)
name (varchar)
email (varchar)
user_id (FK, int)
leaderboard
id (PK, int)
student_id (FK, int)
points (int)
ranking (varchar)
board_type (varchar)
generation_date (date)
report
id (PK, int)
teacher_id (FK, int)
student_id (FK, int)
parent_id (FK, int)
report_date (date)
content (text)
notification
id (PK, int)
teacher_id (FK, int)
student_id (FK, int)
parent_id (FK, int)
content (text)
sent_date (date)
parent
id (PK, int)
name (varchar)
contact_info (varchar)
student
id (PK, int)
name (varchar)
email (varchar)
memorization_level (int)
points (int)
user_id (FK, int)
study_circle
id (PK, int)
name (varchar)
description (text)
capacity (int)
teacher_id (FK, int)
attendance
id (PK, int)
student_id (FK, int)
study_circle_id (FK, int)
teacher_id (FK, int)
status (varchar)
attendance_date (date)
evaluation
id (PK, int)
student_id (FK, int)
teacher_id (FK, int)
score (int)
evaluation_date (date)
comments (text)
associated_with
associated_with
creates
registers
sets points for
generates
sends
is_ranked_on
is_about
receives
receives
receives
has
has
participates_in
enrolls
occurs_in
is_evaluated_in
associated_with
conducts
records

Business Rules

1. A parent can add many children, but each child belongs to only one parent.
2. A student can view only their own profile, attendance, progress, and evaluations.
3. A parent can view the profiles, attendance, progress, and evaluations of all their children.
4. A student must be assigned to at least one study circle.
5. A student cannot enroll in more than one study circle at the same time unless allowed by the admin.

Teacher Rules

1. A teacher can manage only the study circles assigned to them.
2. A teacher can conduct evaluations only for the students in their assigned study circles.
3. A teacher can track attendance only for students in their assigned study circles.
4. A teacher can generate reports only for students they teach.
5. A teacher can send notifications only to students and parents of their assigned study circles.

Admin Rules

1. An admin can manage all users, including students, parents, teachers, and other admins.
2. An admin can manage all reports, including viewing, editing, and deleting them.
3. An admin can monitor the full system including activity logs and system health.
4. An admin can override any restriction in the system when necessary.

Evaluation & Progress

1. A student's progress is calculated automatically based on their evaluation scores and attendance records.
2. Evaluations must be submitted by the teacher within 48 hours of the session.
3. Progress reports are generated monthly and sent to the parent automatically.

Study Circle Rules

1. A study circle must have at least 3 students and no more than 30 students.
2. A teacher can be assigned to multiple study circles.
3. A student can request to enroll in a study circle, but final approval is given by the teacher or admin.

Attendance Rules

1. Attendance must be marked by the teacher on the same day of the session.
2. Attendance records can be modified only within 24 hours by the teacher, and after that only by the admin.
3. Students and parents can view attendance records but cannot modify them.

Leaderboard & Points

1. The leaderboard is updated automatically once a week based on memorization progress and attendance.
2. Points are awarded to students automatically based on attendance, progress, and evaluation scores.
3. Any attempt to manipulate points or leaderboard rankings results in an automatic system flag and admin review.

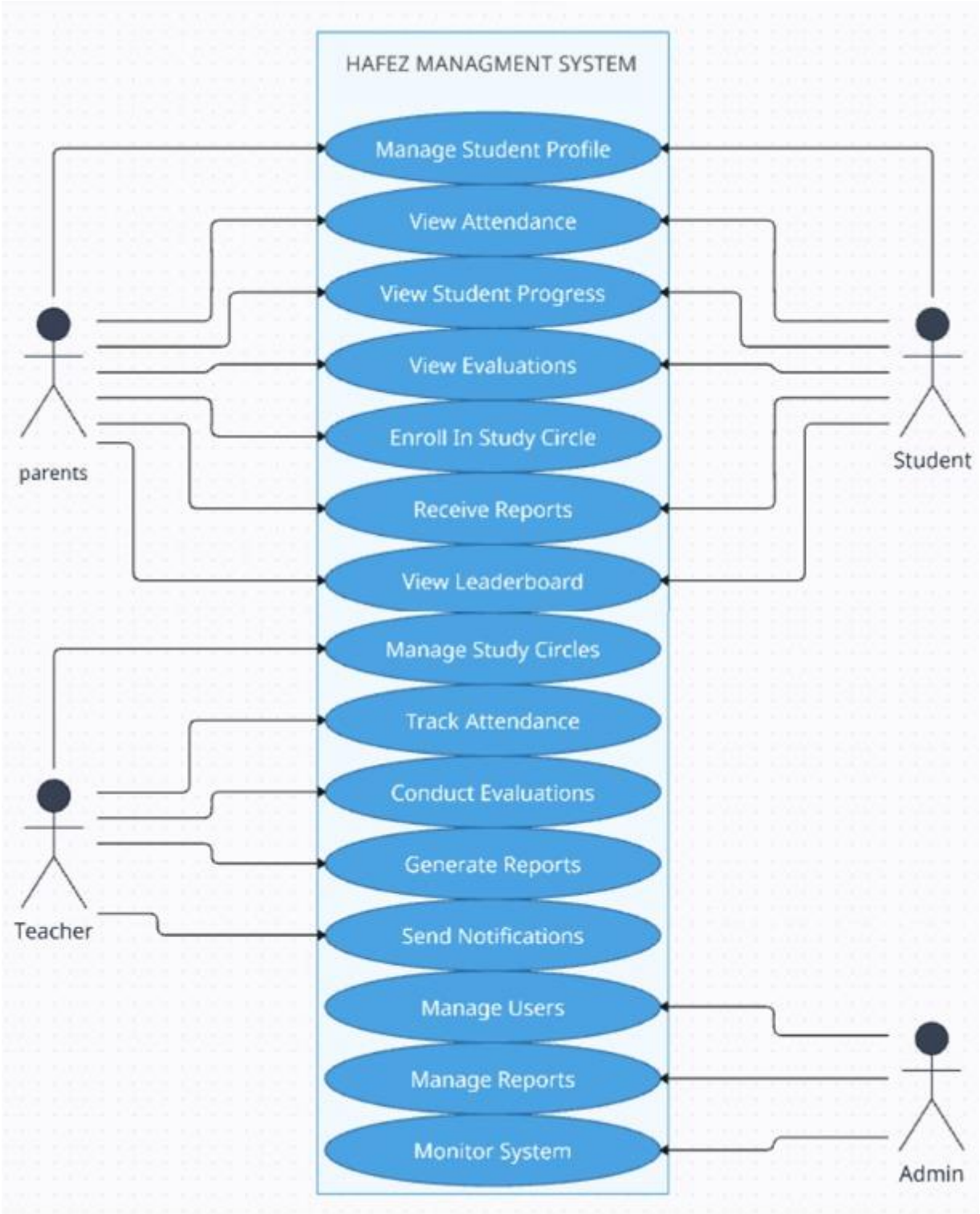
Notification Rules

1. Notifications can be sent by teachers and admins only.
2. Students and parents can receive notifications but cannot send them.

Security & Access

1. Each user must log in with a valid username and password to access the system.
2. Users can only access the features and data allowed by their role.

Use case diagram



Use case descriptions

Use Case: Manage Student Profile

- **Scenario:** Family member or teacher updates the student's profile
- **Triggering Event:** Family member or teacher creates or updates the student's profile
- **Actors:** Family Member, Teacher
- **Preconditions:** Student account is created by a family member or teacher
- **Postconditions:** Profile is updated successfully
- **Flow of Events:**
    1. Family member or teacher selects "Manage Student Profile"
    2. System displays profile details
    3. Family member or teacher updates information
    4. System validates and saves changes

Use Case: View Attendance

- **Scenario:** Student checks attendance records
- **Triggering Event:** Student wants to review attendance
- **Actors:** Student
- **Preconditions:** Student must be enrolled in study circle
- **Postconditions:** Attendance records are displayed
- **Flow of Events:**
    1. Student selects "View Attendance"
    2. System retrieves and displays records

Use Case: View Student Progress

- **Scenario:** Student reviews academic progress
- **Triggering Event:** Student wants to track progress
- **Actors:** Student
- **Preconditions:** Student must be enrolled
- **Postconditions:** Progress details are displayed
- **Flow of Events:**
    1. Student selects "View Progress"
    2. System retrieves and displays progress data

Use Case: View Evaluations

- **Scenario:** Student checks evaluation results
- **Triggering Event:** Student wants to see feedback and scores
- **Actors:** Student
- **Preconditions:** Evaluations must be completed
- **Postconditions:** Evaluation results are displayed
- **Flow of Events:**
    1. Student selects "View Evaluations"
    2. System retrieves and displays evaluation data

Use Case: Enroll in Study Circle

- **Scenario:** Teacher or family member enrolls the student in a study circle
- **Triggering Event:** Teacher or family member wants to enroll the student in a study group
- **Actors:** Teacher, Family Member
- **Preconditions:** Study circle must be available
- **Postconditions:** Student is enrolled in the study circle
- **Flow of Events:**
    1. Teacher or family member selects "Enroll in Study Circle"
    2. System verifies availability
    3. Teacher or family member confirms enrollment
    4. System updates enrollment status

Use Case: Receive Report

- **Scenario:** Student receives academic reports
- **Triggering Event:** A report is generated for the student
- **Actors:** Student
- **Preconditions:** Reports must be available
- **Postconditions:** Report is received and accessible
- **Flow of Events:**
    1. System generates the report
    2. Student receives notification
    3. Student accesses and views the report

Use Case: View Leaderboard

- **Scenario:** Student views ranking and achievements
- **Triggering Event:** Student wants to check leaderboard
- **Actors:** Student
- **Preconditions:** Leaderboard data must be available
- **Postconditions:** Leaderboard is displayed
- **Flow of Events:**
    1. Student selects "View Leaderboard"
    2. System retrieves and displays rankings

Use Case: Track Attendance

- **Scenario:** Teacher tracks student attendance
- **Triggering Event:** Teacher marks attendance
- **Actors:** Teacher
- **Preconditions:** Students must be enrolled in study circle
- **Postconditions:** Attendance is recorded
- **Flow of Events:**
    1. Teacher selects "Track Attendance"
    2. System displays student list
    3. Teacher marks attendance
    4. System updates attendance records

Use Case: Conduct Evaluations

- **Scenario:** Teacher evaluates students
- **Triggering Event:** Teacher wants to assess students
- **Actors:** Teacher
- **Preconditions:** Evaluation criteria must be defined
- **Postconditions:** Evaluations are recorded
- **Flow of Events:**
    1. Teacher selects "Conduct Evaluations"
    2. System displays students and criteria
    3. Teacher submits evaluations
    4. System records results

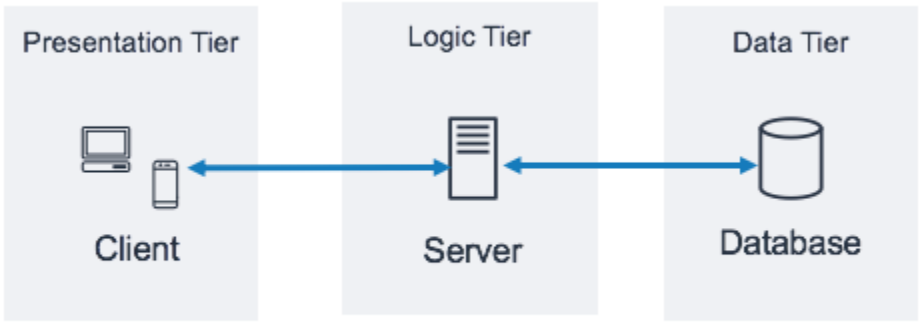
Use Case: Generate Reports

- **Scenario:** Teacher or family member generates performance reports
- **Triggering Event:** Teacher or family member wants to create reports
- **Actors:** Teacher, Family Member
- **Preconditions:** Data must be available for reporting
- **Postconditions:** Report is generated
- **Flow of Events:**
    1. Teacher or family member selects "Generate Reports"
    2. System retrieves data
    3. System generates and displays report

Use Case: Send Notifications

- **Scenario:** System sends notifications to users
- **Triggering Event:** An event triggers a notification
- **Actors:** System, Users
- **Preconditions:** Users must have notifications enabled
- **Postconditions:** Notification is sent and received
- **Flow of Events:**
    1. System detects an event
    2. System generates notification
    3. User receives notification

The Application Architecture

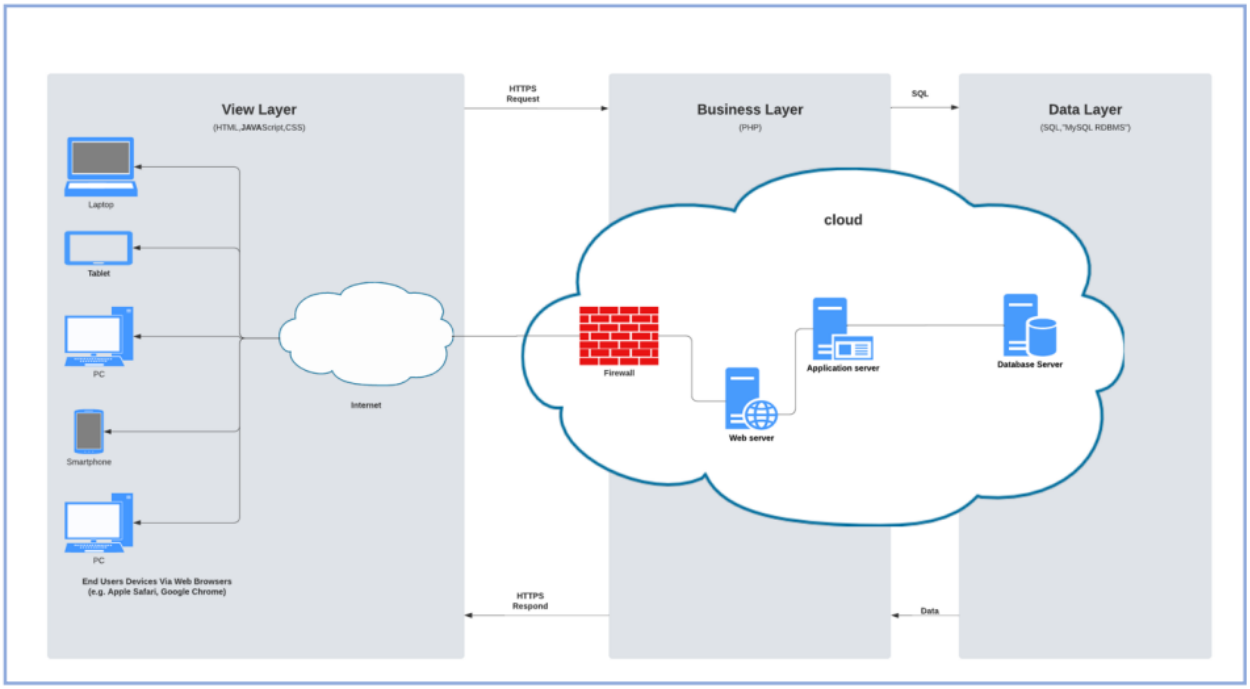


The system **Hafez** is a web-based system that is available via the internet, and it can be accessed by any browser and any device.

A three-tier client-server architecture will be more appropriate to offer a higher level of reliability, flexibility, and availability for the client since each tier is deployed separately and running on its own infrastructure. The three-tier architecture will be the view layer, the business layer, and the data layer. The view layer will include the user interface where the user can interact with the system and access it using any browser and any device. The view layer front- 26 end development is made with (HTML, JavaScript, CSS). The business layer will include a cloud firewall as well as the application server and the web server that receives the request and information from the view layer and processes it is using (HTTPS and HTTP). The data layer will include the database server where data is stored and processed. The programming languages for back-end development are SQL and PHP (Hypertext Preprocessor).

For external usage, cloud computing will be a suitable option and has many advantages. Unlike traditional computing, cloud computing can help the users to access the system from any location at any time and at any scale, as well as reducing the cost and it is more secure and reliable. Applications and websites are accessible via cloud resources thanks to cloud hosting. The solution is not set up on a single server, unlike conventional hosting. The program or website is instead hosted by a network of linked physical and virtual cloud servers, offering more flexibility and scalability.

Network Diagram



The Network Diagram shows by virtue of what **Hafez** system will be used across networks and designs and represents the interactions that occur between each of the three layers, the view layer, the application layer, and the database layer. The system will support various types of devices accompanying various screen sizes and accompanying various operating.

systems, containing phones, desktops, laptops, and tablets. Also, all consumers will approach the system by way of various network browsers (such as Apple Safari, Google Chrome) utilizing the Internet. There's a cloud Firewall provided by the GoDaddy package to cover the system introduction points and avert some pirated approaches like hackers and dismissal of valuable attacks. In addition, it will stop the bug attacks and control the solitude by obstructing some hateful beginnings.

Since the system contains the web structure, we used the web server, the application server, and the database server. Web Server that will take and process the HTTP/HTTPS requests from customers (network browsers) constantly and returns HTTP/HTTPS responses in conditions of HTML pages, figures, etc., as a consumer interface.

And the server that is located between the web server and the database server is an application server. Application Server. Its main responsibility is to centrally accomplish the database server and the web server and control communication and interplays between customers and content. It is software that performs all the coding and security tasks necessary for the operation of the system as a whole.

Database Server will monitor and store the database on the server and form the **Hafez** system more securely by providing an approach only to approved consumers. **Hafez** system will consume a cloud database as noticed before that will supply an approach to the database by way of the Internet. It will allow the **Hafez** group to host the database outside purchasing fittings and to scale up or below IT resources as wanted. It will go on a database management system that is MySQL. Finally, all the servers (web server, application server, database server) will be hosted utilizing Cloud Computing.

Required System Controls

Controls that protect data and the system from any attacks. Our system has two types of controls **integrity** and **security** control.

Integrity Controls

Integrity controls are produced to protect the system and information overall to ensure correct system functions by rejecting erroneous data inputs, preventing illegal data outputs, and protecting data and systems from malicious or unintentional attacks. The system will use several integrity control types, including the ones listed below.

1. **Input controls**

   - Value limit control

   Value limit control is used to determine whether the value entered in the field is reasonable or not. For example, when the user enters a password above 10 values, the system will reject it.

   - Completeness control

   These controls are used to determine whether all the required field values have been entered or not. For example: if the user didn't complete all the required fields (full name, email, phone number, password) and goes to the next page the website will show the user they didn't complete the required fields.

   - Data validation control

   These ensure that numeric fields containing codes or identifiers are correct. For example, the vendor will need to choose their country.

2. **Transaction Logging**

   Transaction logging is a record and store of any actions updates and changes by the database. Also, its aim is to help deter fraudulent transactions or malicious database changes and to provide a recovery mechanism for wrong transactions.

3. **Complex Update Control**

   Complex Update Control is a control that prevents any error that may happen when multiple programs attempt to update the same data concurrently or when recording a single transaction requires multiple related database modifications.

4. **Backup and Recovery**

   Backup and recovery describe the process of creating and storing copies to ensure the users' data availability if any attack or failure happens to the system.

5. Output Control

Output control ensures output is correct, up-to-date, complete, and reaches the right place. For example: when you open **Hafez** the website will display the trip on the home page. In another specific example When we generate a report will contain the date info at top of the page and the format of the page numbers with the total number of pages will be at the bottom of the page. Our logo **Hafez** will be included in all the generated reports.

Security Controls

Security controls will be used to protect the system from any potential external threats, the users must be able to trust the system to work in a secure environment.

1. Access Controls

- Authentication

The system will check if the email, password length 6 and user type match what we had in the database.

- Authorization

There are three types of users (Admin, Vendor, Customer) each have authority in our system.

2. Secure Transaction

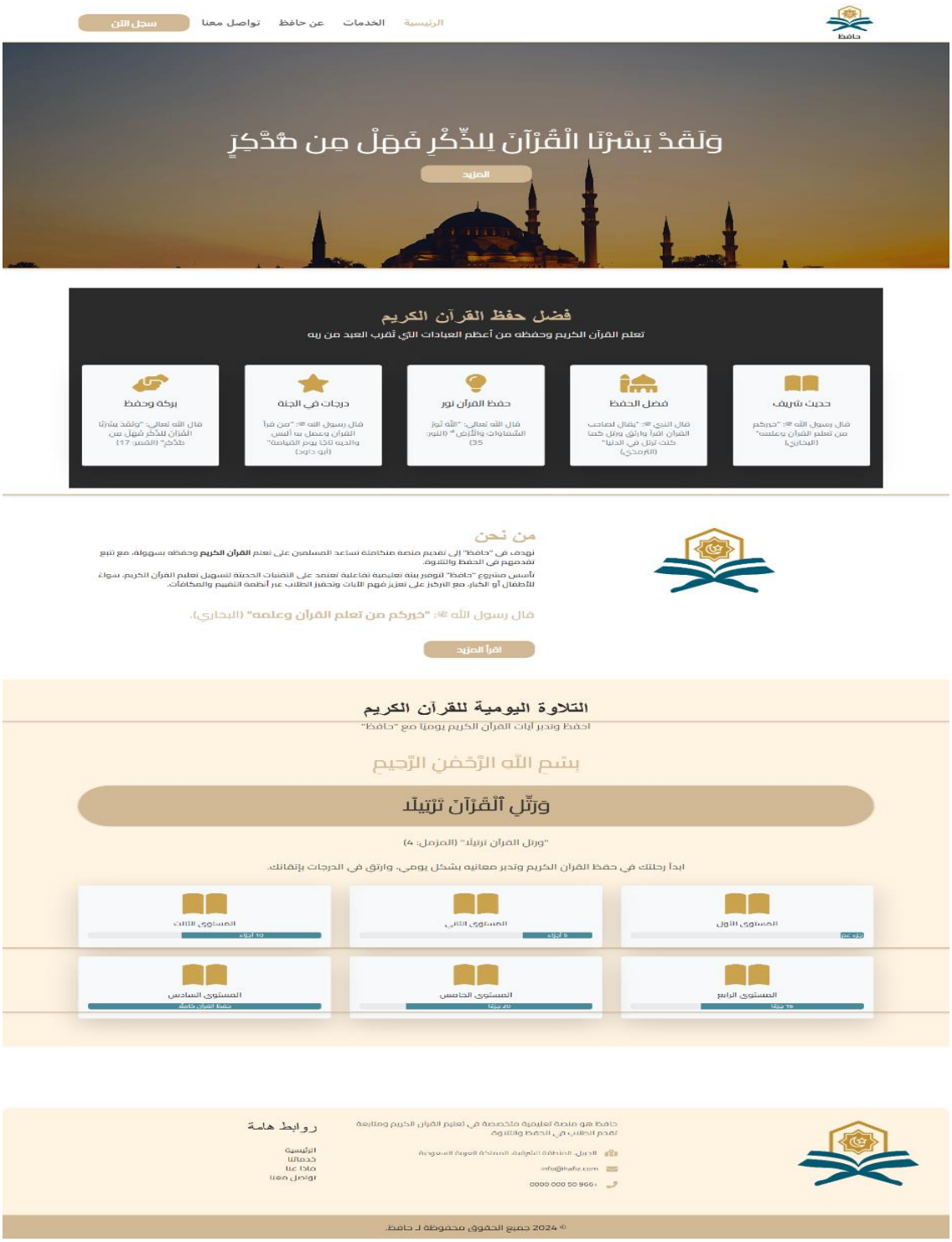
A Secure Sockets Layer (SSL) certificate will be gained by the GoDaddy package which has a free SSL. It is used with HTTP. To convert it into HTTPS, the data transmitted over the internet is encrypted and ensures the transactions between the server and browser are secure. This limits the ability of hackers to read sensitive data over the internet. For more security, the system will have a session timeout if the user doesn't take an action on the site in a while the system will show up that the session is ended.

3. Data Encryption

By using encryption technique “Hash”, we make users’ sensitive data more secure and protect it from unauthorized access. They can't see sensitive data such as passwords as readable text. All this data will show as encrypted text.

Simplified User Interface Design

Homepage



تواصل معنا

نحن هنا لمساعدتك في أي استفسارات أو أسئلة تتعلق بحفظ القرآن الكريم.

العنوان

الجبيل، المنطقة الشرقية، المملكة العربية السعودية

البريد الإلكتروني

info@hafiz.com

الهاتف

0000 000 50 966+

هل لديك استفسار؟

يمكنك التواصل معنا عبر البريد الإلكتروني أو الهاتف للحصول على الدعم والمساعدة.

حافظ هو منصة تعليمية متخصصة في تعليم القرآن الكريم ومتابعة تقدم الطلاب في الحفظ والتلاوة.

الجبيل، المنطقة الشرقية، المملكة العربية السعودية

info@hafiz.com

0000 000 50 966+

روابط هامة

الرئيسية
خدماتنا
ماذا عنا
تواصل معنا

من نحن

تعرف على رسالتنا، أهدافنا، وكيف نساعدك في حفظ القرآن الكريم بكل سهولة ويسر.

رسالتنا

نهدف إلى توفير منصة تعليمية متطورة تساعد المسلمين من جميع الأعمار على حفظ وتلاوة القرآن الكريم بسهولة، مع متابعة تقدمهم وتحفيزهم بشكل مستمر.

رؤيتنا

أن نكون الوجهة الأولى عالميًا في تعليم وتحفيظ القرآن الكريم عبر الوسائل التقنية الحديثة، مع تقديم تجربة تعليمية تفاعلية ومتميزة.

انضم إلى "حافظ" اليوم!

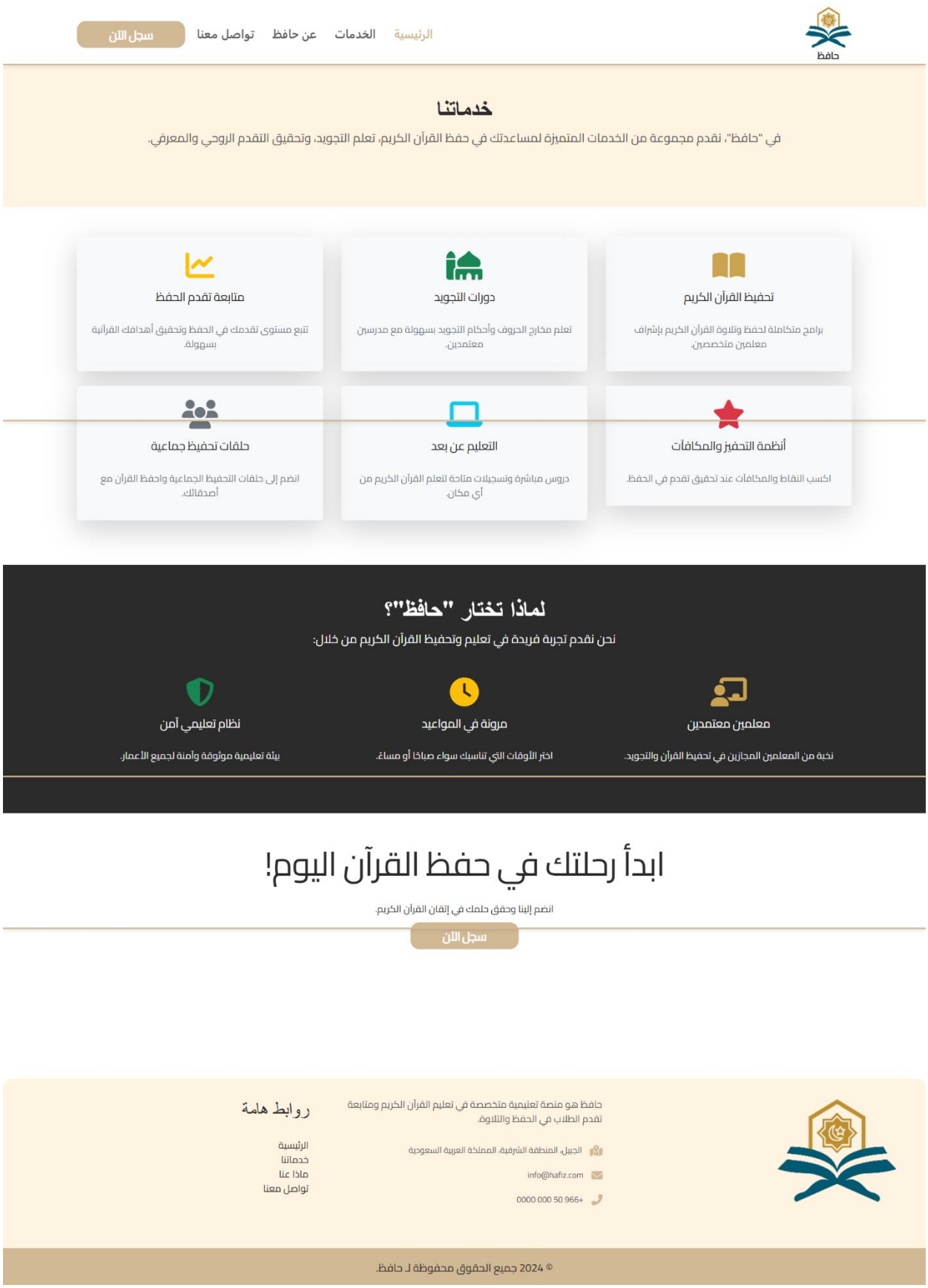
ابدأ رحلتك في حفظ وتعلم القرآن الكريم معنا.

سجل الآن

حافظ هو منصة تعليمية متخصصة في تعليم القرآن الكريم ومتابعة تقدم الطلاب في الحفظ والتلاوة.

الجبيل، المنطقة الشرقية، المملكة العربية السعودية
info@hafiz.com
+966 50 000 0000

روابط هامة


حافظ
الرئيسية
الخدمات
عن حافظ
تواصل معنا
سجل الآن
خدماتنا
في "حافظ"، نقدم مجموعة من الخدمات المتميزة لمساعدتك في حفظ القرآن الكريم، تعلم التجويد، وتحقيق التقدم الروحي والمعرفي.
تحفيظ القرآن الكريم
برامج متكاملة لحفظ وتلاوة القرآن الكريم بإشراف معلمين متخصصين.
دورات التجويد
تعلم مخارج الحروف وأحكام التجويد بسهولة مع مدرسين معتمدين.
متابعة تقدم الحفظ
تتبع مستوى تقدمك في الحفظ وتحقيق أهدافك القرآنية بسهولة.
أنظمة التحفيز والمكافآت
اكسب النقاط والمكافآت عند تحقيق تقدم في الحفظ.
التعليم عن بعد
دروس مباشرة وتسجيلات متاحة لتعلم القرآن الكريم من أي مكان.
حلقات تحفيظ جماعية
انضم إلى حلقات التحفيظ الجماعية واحفظ القرآن مع أصدقائك.
لماذا تختار "حافظ"؟
نحن نقدم تجربة فريدة في تعليم وتحفيظ القرآن الكريم من خلال:
معلمين معتمدين
نخبة من المعلمين المجازين في تحفيظ القرآن والتجويد.
مرونة في المواعيد
اختر الأوقات التي تناسبك سواء صباحًا أو مساءً.
نظام تعليمي آمن
بيئة تعليمية موثوقة وآمنة لجميع الأعمار.
ابدأ رحلتك في حفظ القرآن اليوم!
انضم إلينا وحقق حلمك في إتقان القرآن الكريم.
سجل الآن
حافظ هو منصة تعليمية متخصصة في تعليم القرآن الكريم ومتابعة تقدم الطلاب في الحفظ والتلاوة.
الجبيل، المنطقة الشرقية، المملكة العربية السعودية
info@hafiz.com
+966 50 000 0000
روابط هامة
الرئيسية
خدماتنا
ماذا عنا
تواصل معنا
© 2024 جميع الحقوق محفوظة لـ حافظ.

Registration / Log in

Introduction

The next document outlines the technical implementation and testing process for Hafez. The system was built using modern web technologies and rigorous testing methodologies to ensure reliability, usability, and performance. Below, we detail the programming languages, tools, testing approaches, and key deliverables, including user interface and database structure.

Implementation & Testing

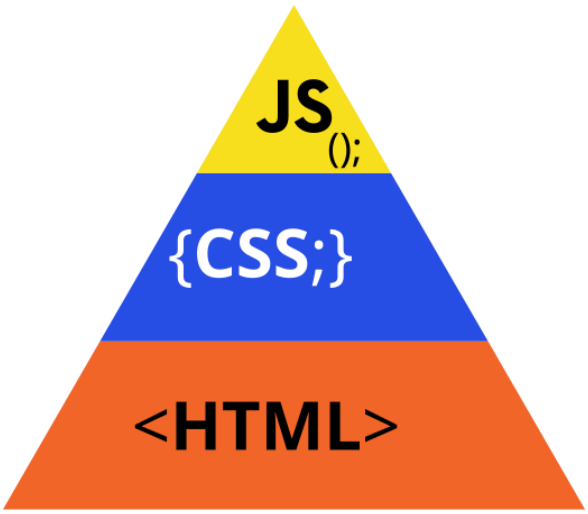
1. Programming Languages

Frontend:

HTML: Page structure and content.

CSS3: Styling and responsive design.

JavaScript: Interactive features (e.g., dynamic reports, form validation).



Backend:

PHP: Server-side logic (e.g., data processing, authentication).

2. Tools Used

Development: VS Code (with extensions: PHP Intelephense, ESLint).

Version Control: GitHub for code collaboration and history tracking.

Database Management: phpMyAdmin for SQL operations.

3. Types of Testing

Unit Testing: Verified individual functions (e.g., login authentication).

Integration Testing: Tested connections (e.g., frontend ↔ backend API).

System Testing: Validated end-to-end workflows (e.g., report generation).

User Acceptance Testing (UAT): Confirmed usability with stakeholders.

4. Types of Reports

Memorization Report: Tracks progress in Quranic/prescribed material.

Attendance Report: Records student presence/absence.

Points & Ranking Report: Displays earned points and leaderboards.

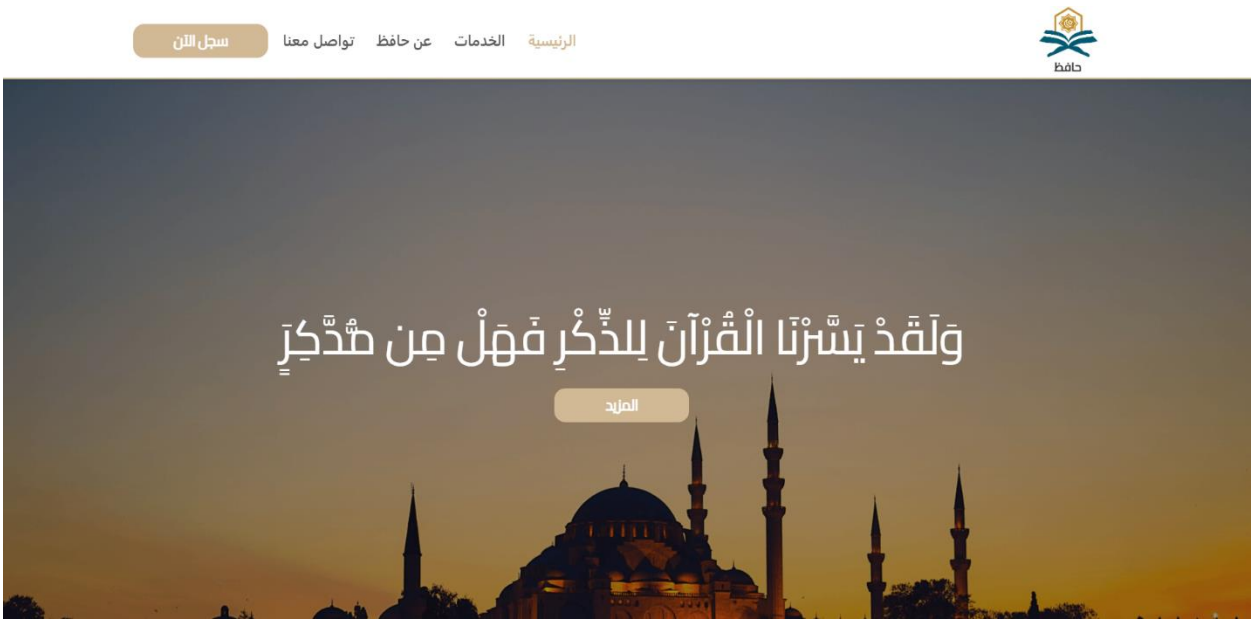
Evaluations Report: Summarizes teacher assessments.

Teachers' Reports: Instructional feedback and class summaries.

5. Detailed User Interface

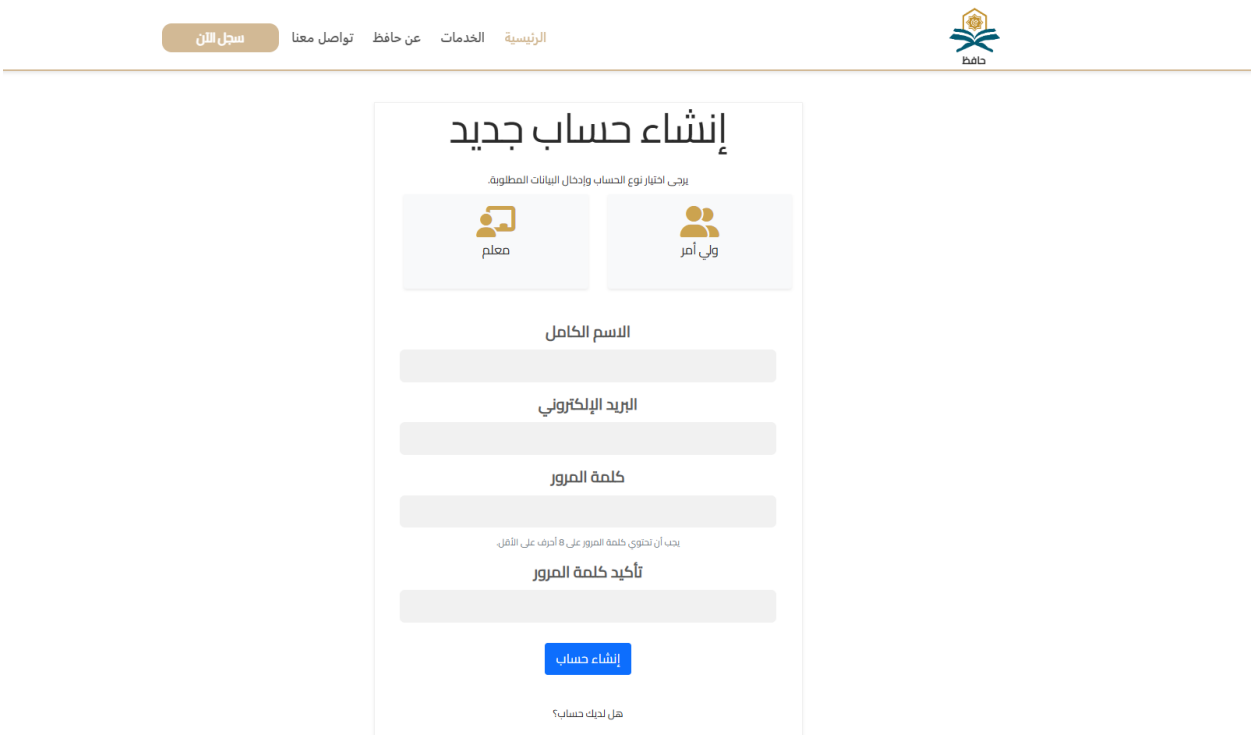
Homepage

The homepage is the first page you see when accessing the website. It includes several options such as a page to browse services, an about page explaining the website and its concept, a contact page, and buttons for registration or login.



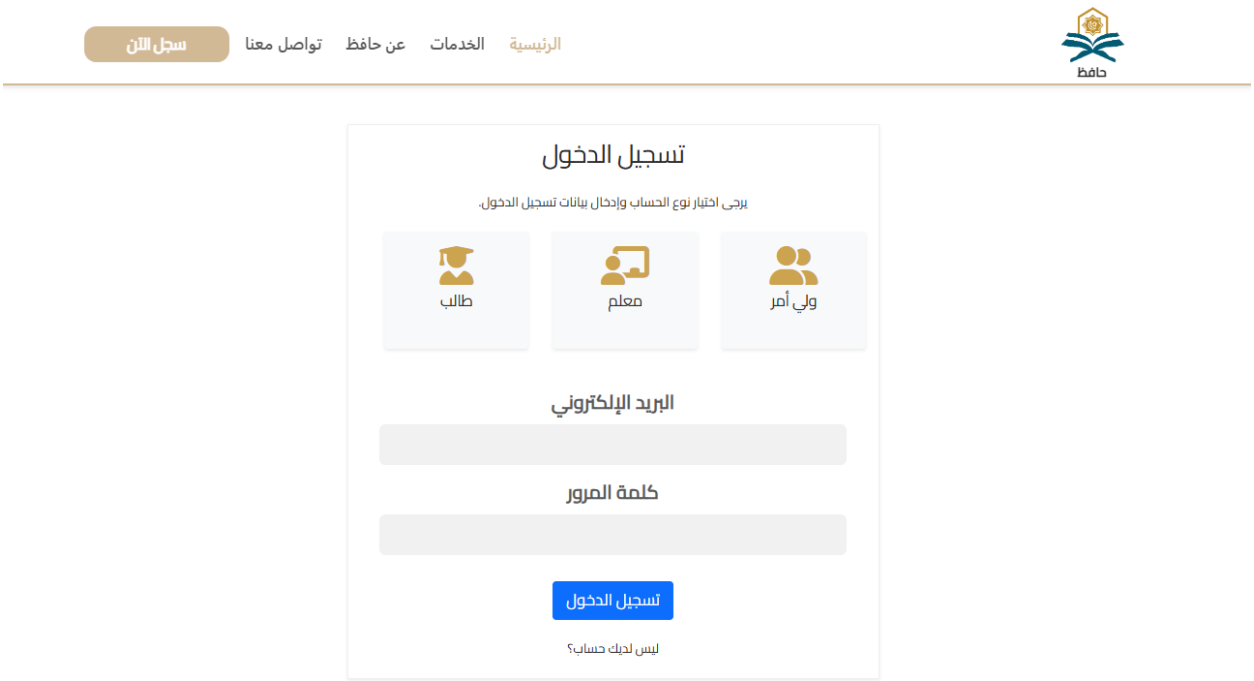
Registration Page

Through this page, you can create a new account and choose your membership type (Family Supervisor or Teacher).
During registration, you are required to enter your full name, email address, and password.



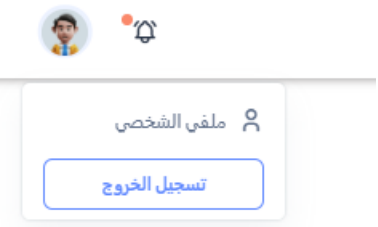
Login Page

If you already have an account, you can log in by selecting your account type (Family Supervisor – Teacher – Student) and entering your email and password.



Profile Access and Logout

Through this small dropdown menu, you can access your profile or log out of your account.

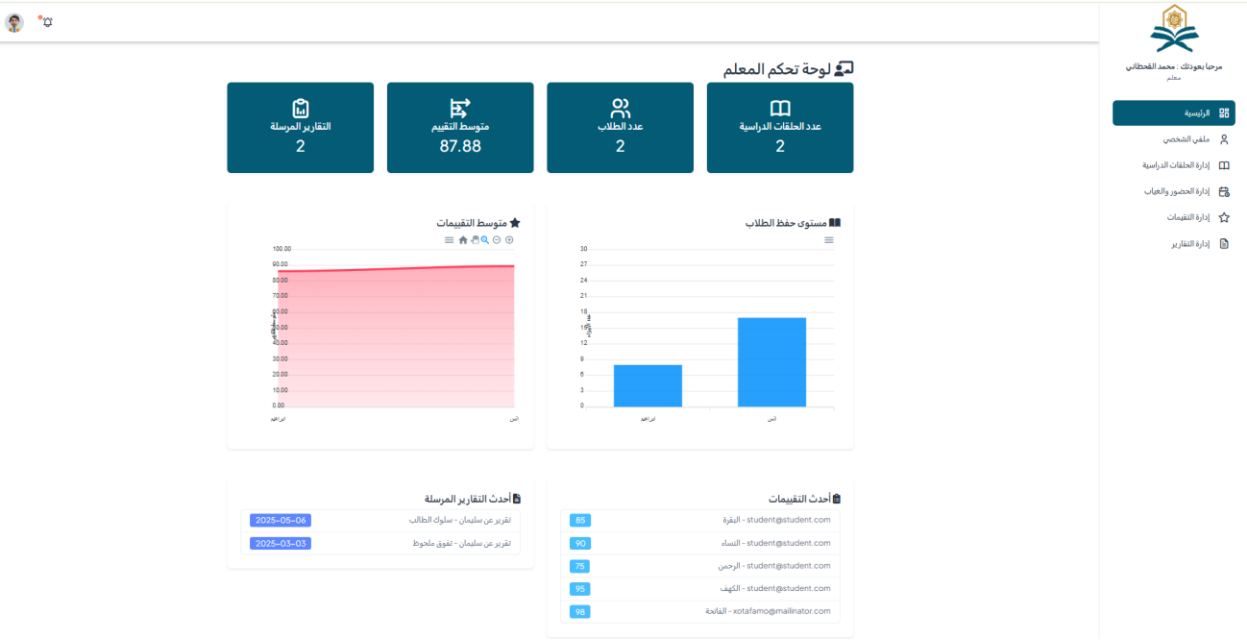


Account Information Update Page

From here, you can easily update your account information such as name, email, phone number, and more.

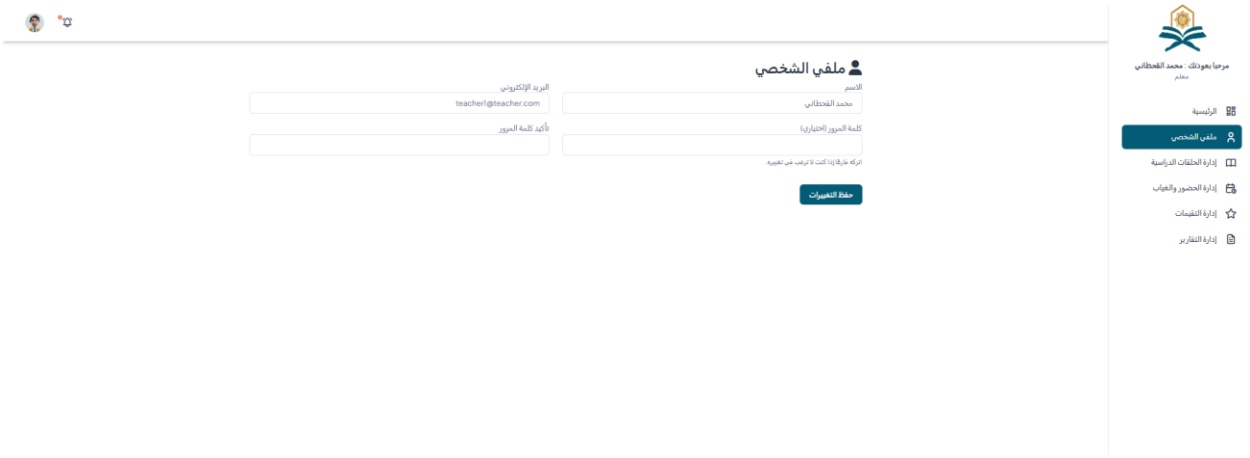
Teacher – Main Page

Through the teacher's main page, you can view your different options, general information and reports about study groups, students, average evaluations, and an overview of the reports.



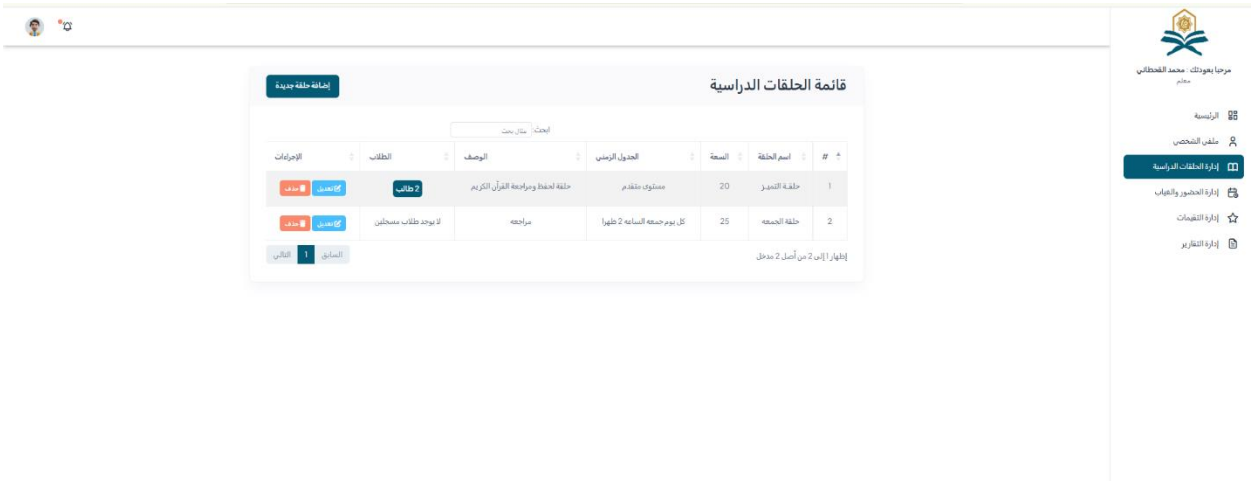
Teacher – Profile Page

Here you can manage and edit your personal profile information as a teacher.



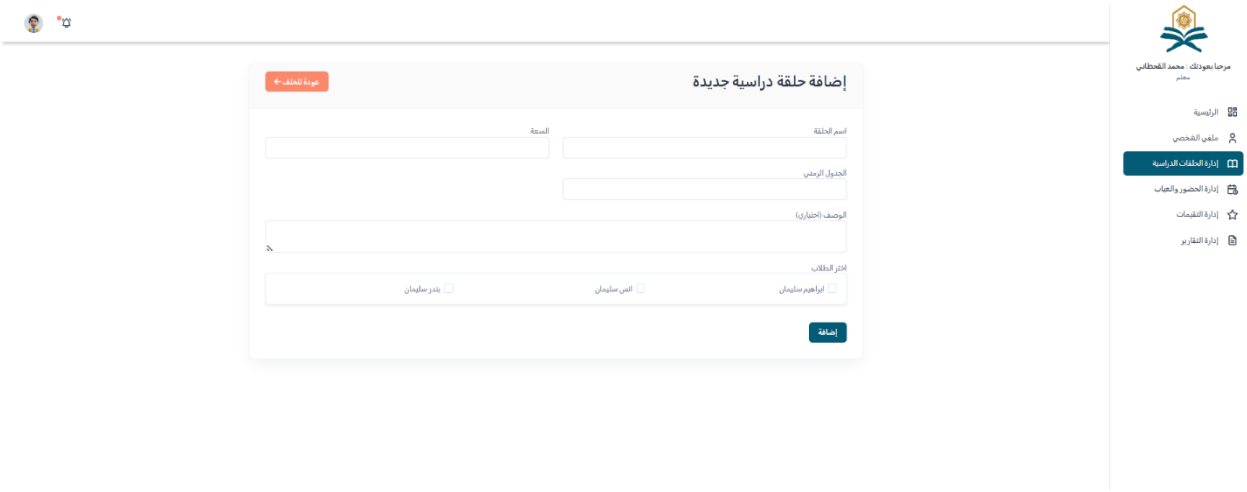
Teacher – Manage Study Groups

This page allows you to browse your list of study groups with an overview of each.



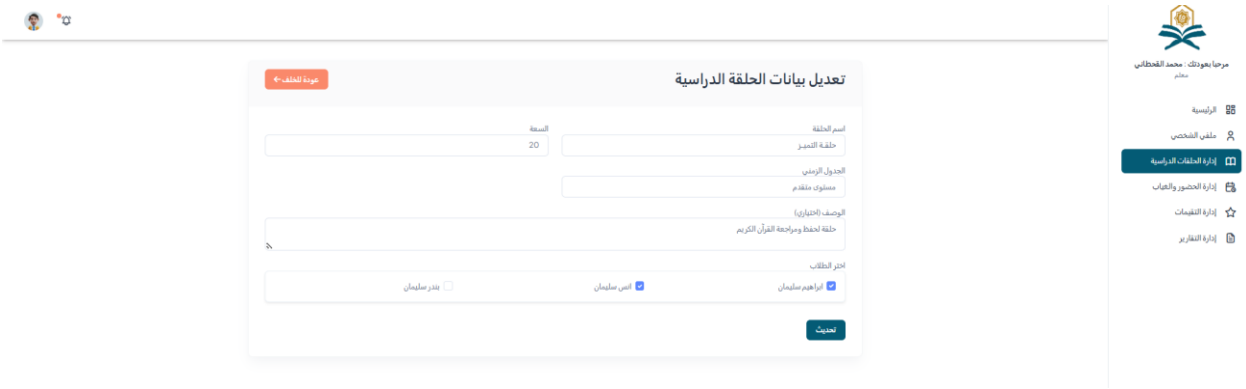
Teacher – Manage Study Group – Add

Here you can add a new study group by filling in its relevant information.



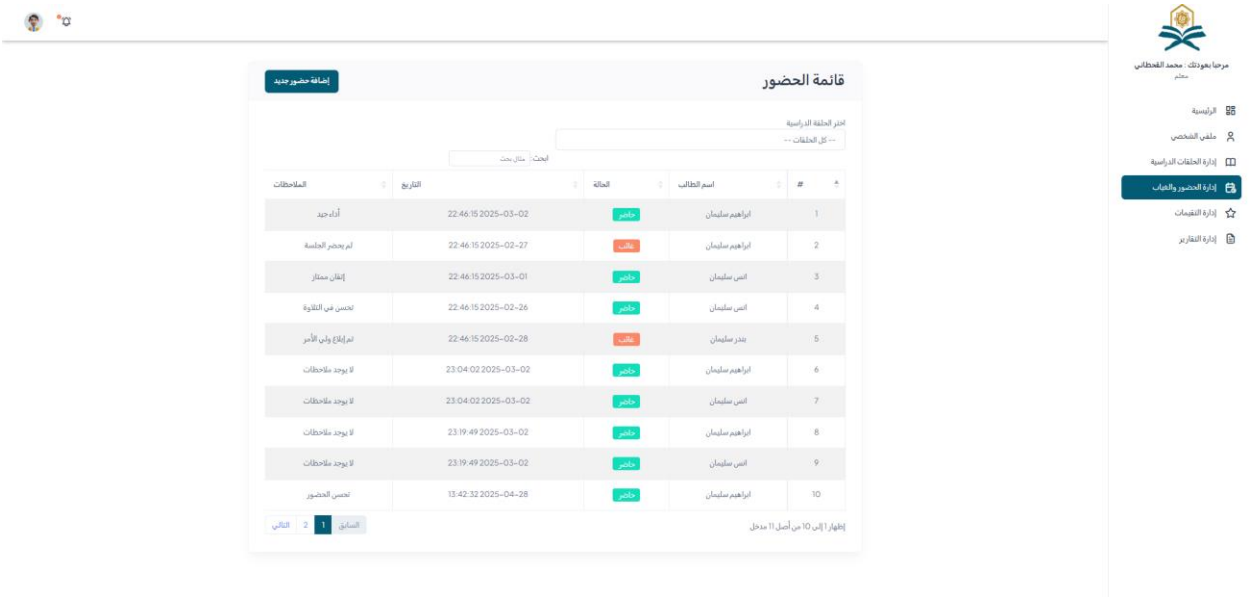
Teacher – Manage Study Groups – Edit

From here, you can easily edit the data of your study groups without needing to delete and re-create them.



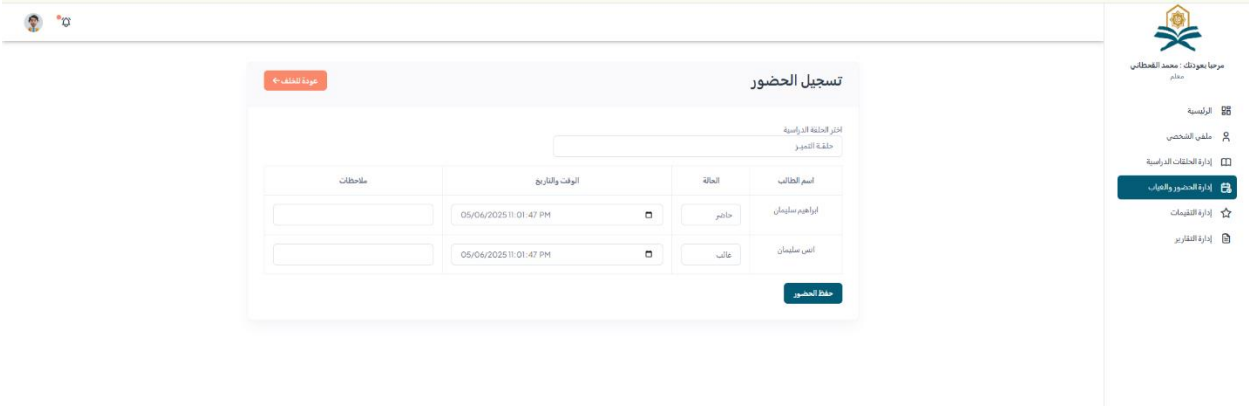
Teacher – Attendance Management

From this page, the teacher can get an overview of the attendance records.



Teacher – Attendance Management – Mark Attendance

This page allows the teacher to easily mark student attendance and absences.

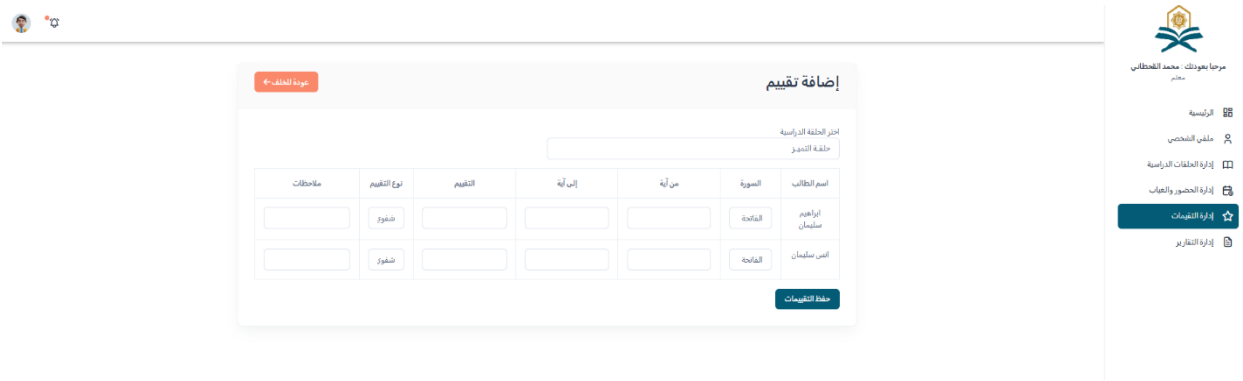


Teacher – Evaluation List

Through this page, you can evaluate students by selecting the group and the student.

Teacher – Evaluation List – Record Evaluation

Here you can easily and accurately assess the student.



Teacher – Reports to Parents or Student Supervisors

A special feature we offer is the ability to send direct written reports from the teacher to the student's guardian (parent or otherwise).

Teacher – Send Report to Parent or Student Supervisor

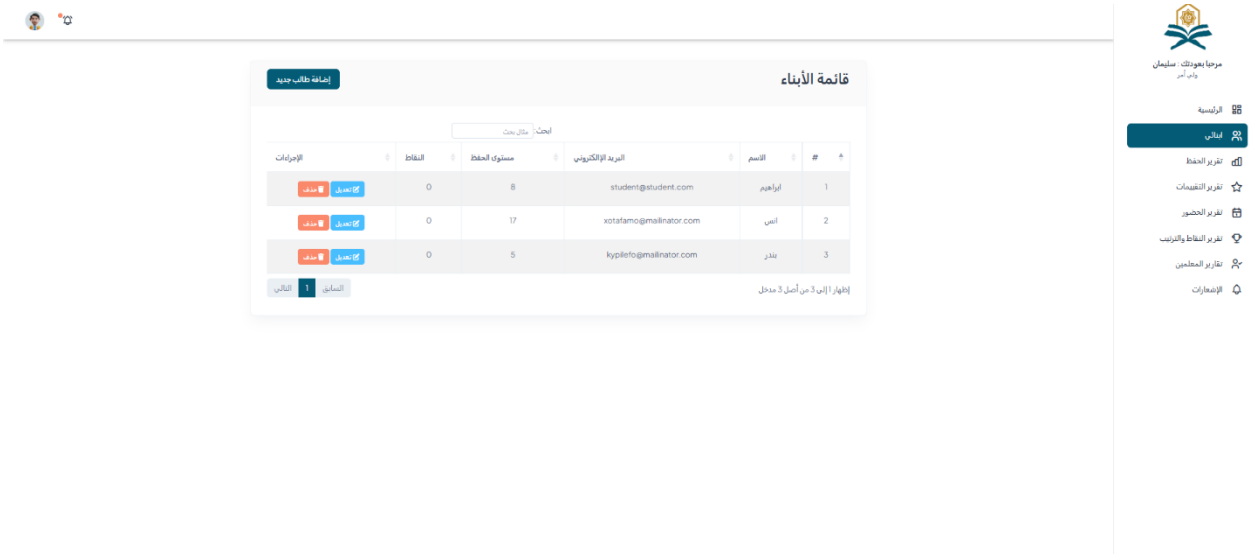
From here, you can select the recipient of the report, choose the report type, and add a message.

Parent – Main Page

This page gives an overview of your children's statistics, reports, and evaluations.

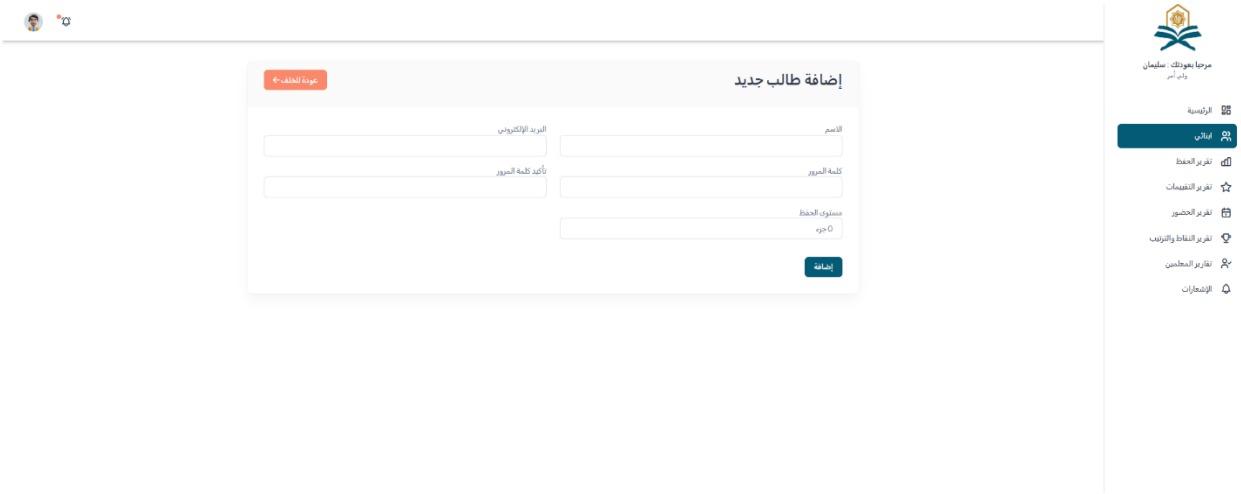
Parent – My Children

This page displays all your children or those you supervise, with options to edit, add, or delete.



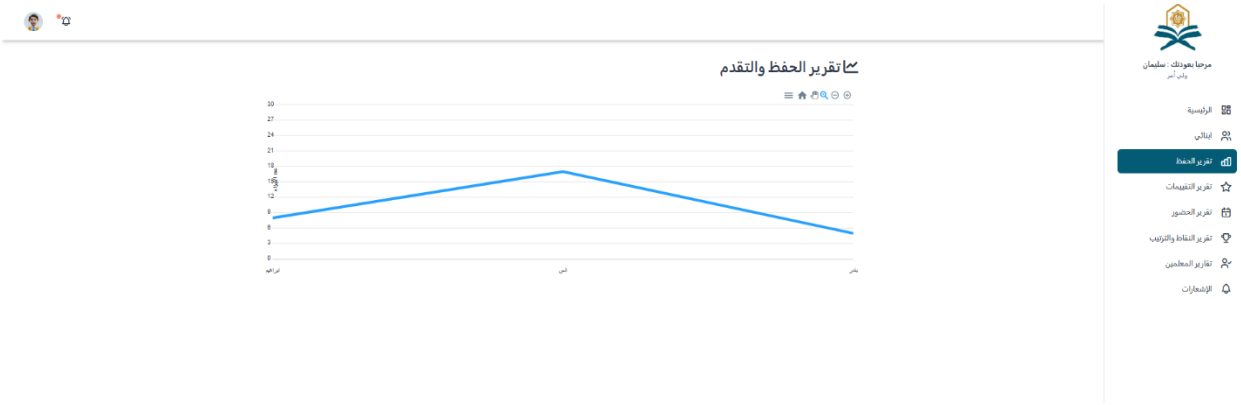
Parent – Add Student

Through this page, you can add a new student (child or otherwise).



Parent – Memorization Reports

From this page, you can monitor your child's memorization reports and compare them with others.



Parent – Evaluation Report

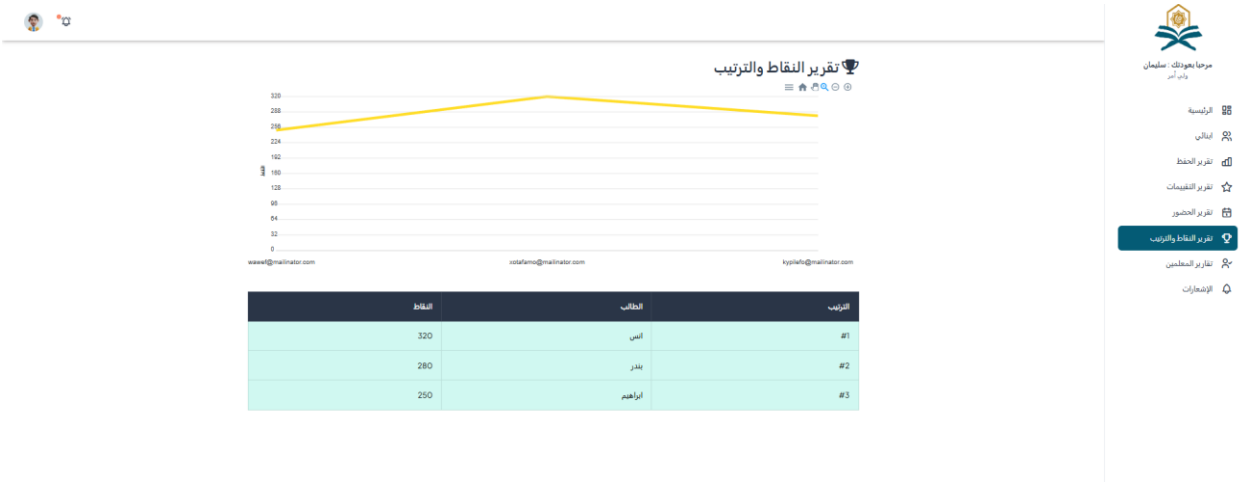
This page provides personalized and detailed evaluations for each student individually.

Parent – Attendance Report

This page offers a view of each student's attendance records along with dates and a dashboard for comparing attendance and absences.

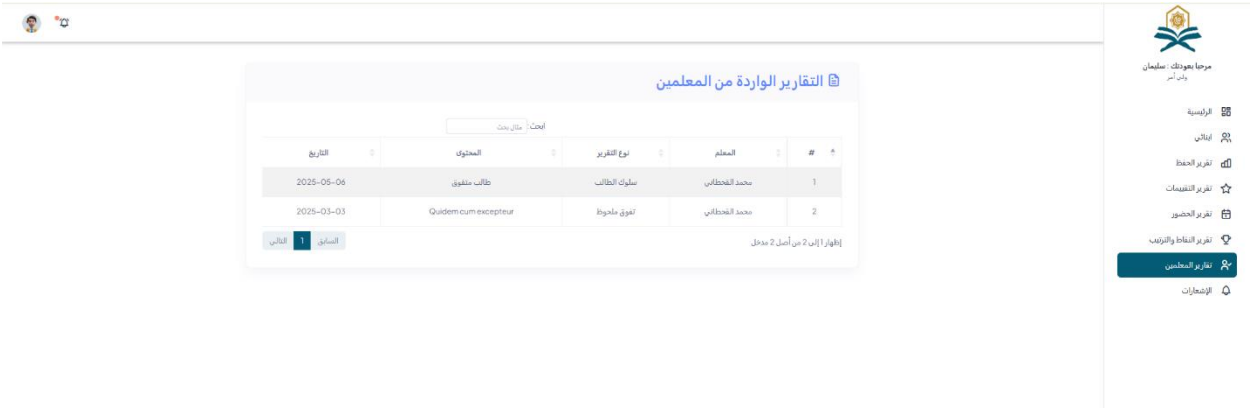
Parent – Points and Ranking Report

This page shows student rankings and their points, along with a comparison graph.



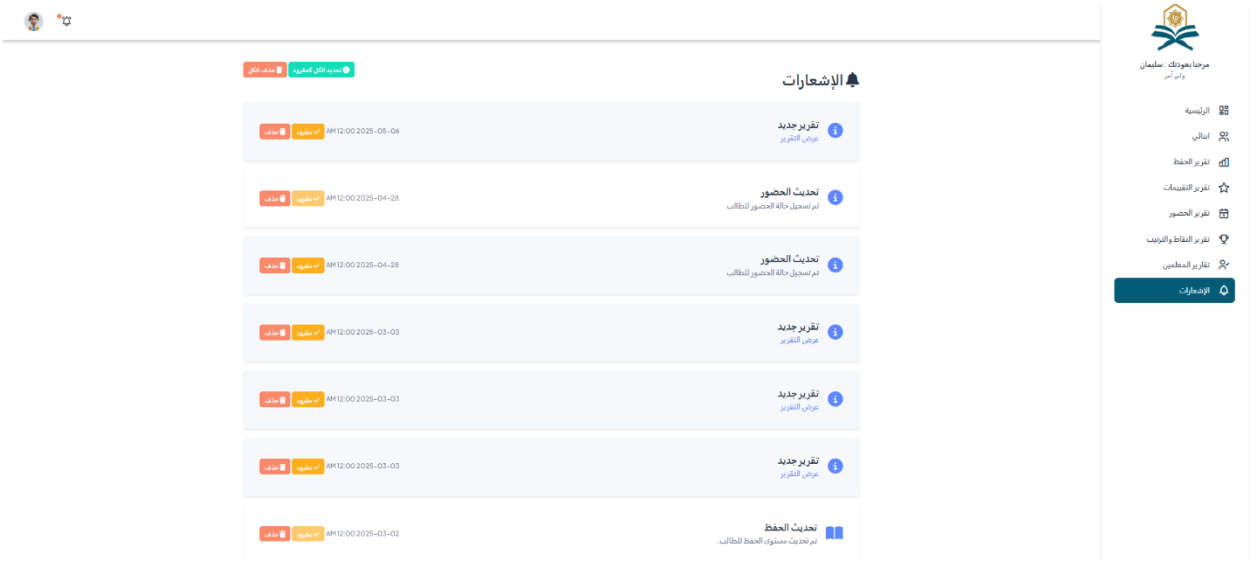
Parent – Teacher Reports

You will receive direct reports from teachers via this page.



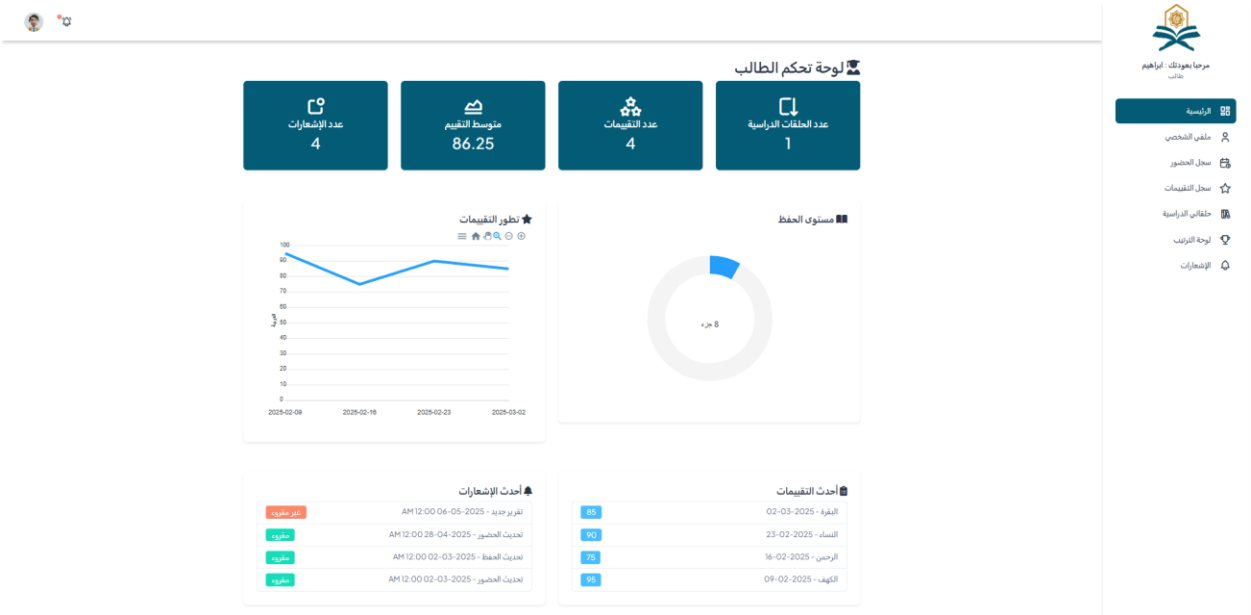
Parent – Notifications

This page provides an overview of all types of notifications and their dates.



Student – Main Page

The student homepage offers an overview of statistics and reports.

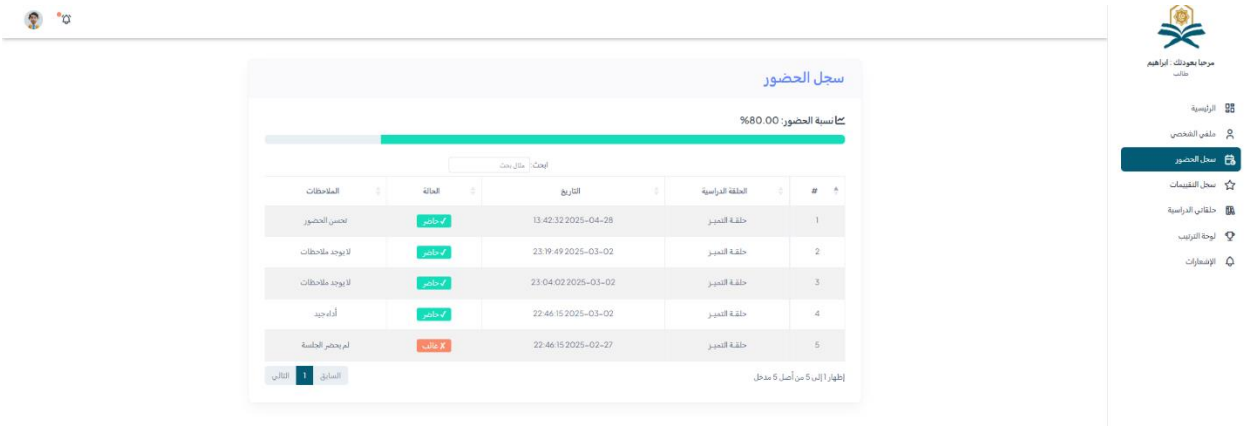


Student – Profile Page

From here, students can manage their profile, including editing their name, email, and password.

Student – Attendance Record

Here, the student can track their attendance and absence history.

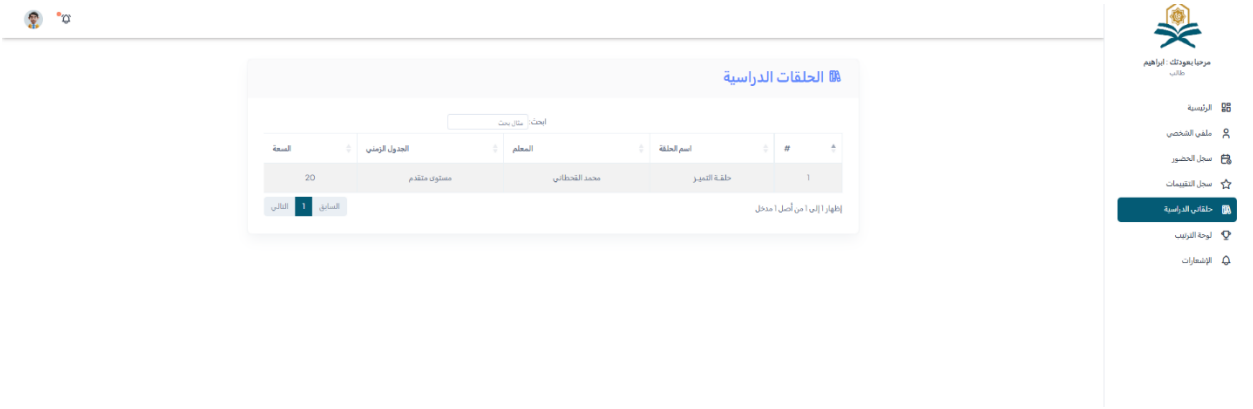


Student – Evaluation Record

From this page, the student can follow up on their evaluation history with dates and details.

Student – Study Groups

Students can view the study groups they are enrolled in along with details.



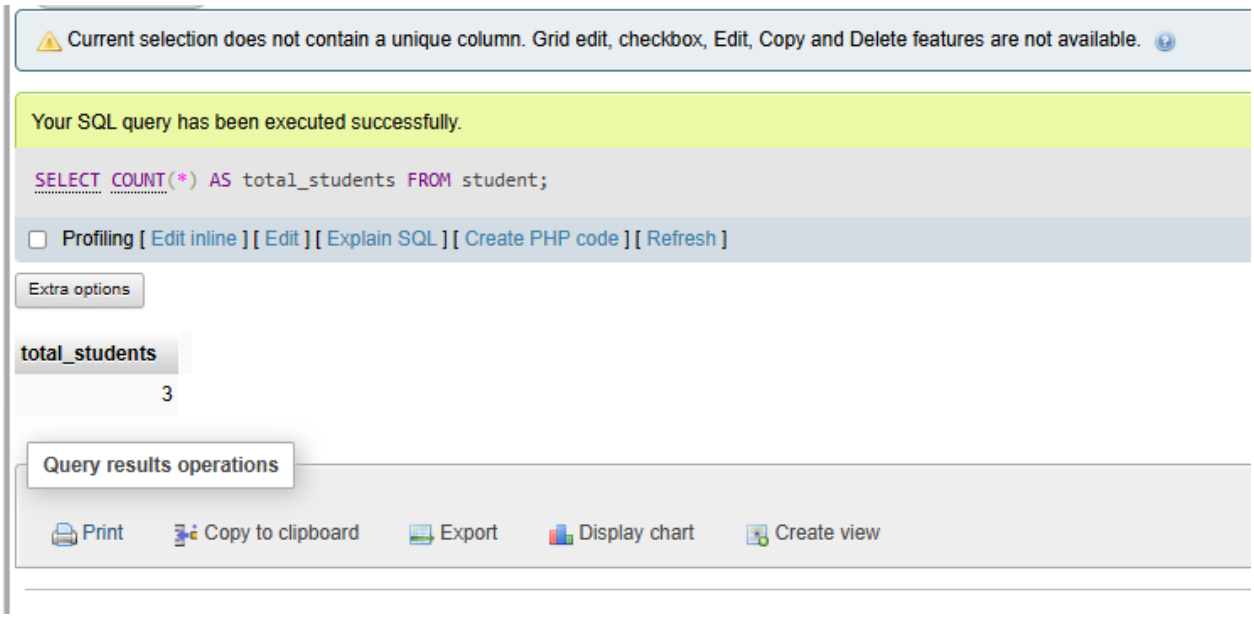
Student – Leaderboard

Students can view the group leaderboards and their total points.

1. Report Name: Total Number of Students

Description: Displays the total number of students in the system.
SQL CODE:

```
SELECT COUNT(*) AS total_students FROM student;
```



---

2. Report Name: Average Evaluation Score per Student

Description: Calculates the average evaluation score for each student.
SQL CODE:

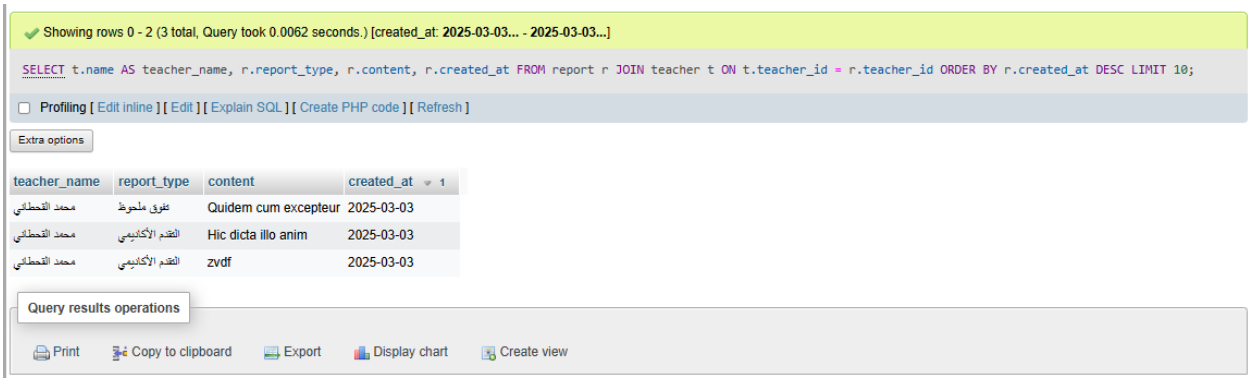
```
SELECT s.name, AVG(e.score) AS average_score

FROM student s

JOIN evaluation e ON s.id = e.student_id

GROUP BY s.id;
```

3. Report Name: Latest Teacher Reports

Description: Shows the latest 10 reports submitted by teachers.

SQL CODE:

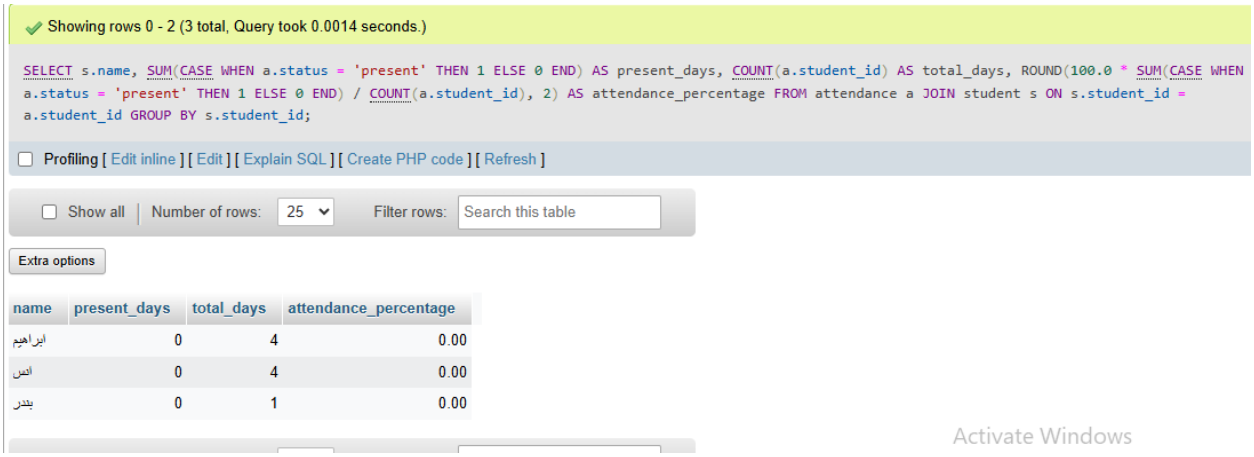
```
SELECT t.name AS teacher_name, r.report_type, r.content, r.created_at

FROM report r

JOIN teacher t ON t. teacher_id= r.teacher_id

ORDER BY r.created_at DESC

LIMIT 10;
```



4. Report Name: Student Attendance Percentage

Description: Calculates the attendance percentage for each student.
SQL CODE:

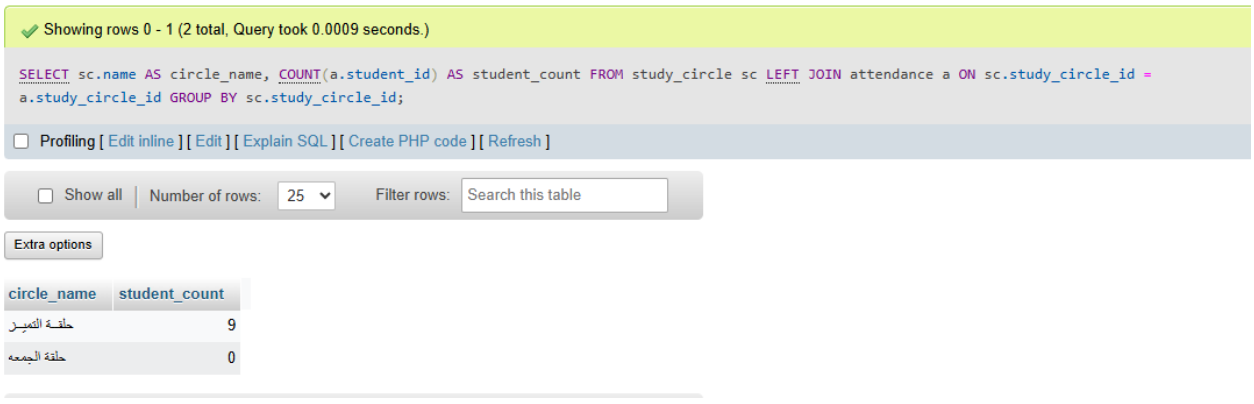
```
SELECT s.name, SUM(CASE WHEN a.status = 'present' THEN 1 ELSE 0 END) AS present_days, COUNT(a.student_id) AS total_days, ROUND(100.0 * SUM(CASE WHEN a.status = 'present' THEN 1 ELSE 0 END) / COUNT(a.student_id), 2) AS attendance_percentage FROM attendance a JOIN student s ON s.student_id = a.student_id GROUP BY s.student_id;
```



5. Report Name: Study Circles and Student Count

Description: Displays each study circle and the number of students attending.
SQL CODE:

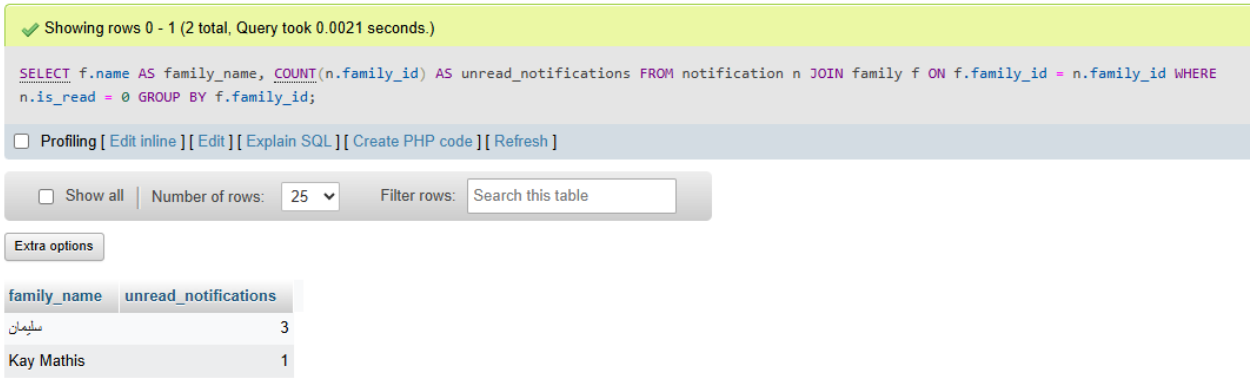
```
SELECT sc.name AS circle_name, COUNT(a.student_id) AS student_count FROM study_circle sc LEFT JOIN attendance a ON sc.study_circle_id = a.study_circle_id GROUP BY sc.study_circle_id;
```



6. Report Name: Unread Notifications per Family

Description: Shows the number of unread notifications for each family.
SQL CODE:

```
SELECT f.name AS family_name, COUNT(n.family_id) AS unread_notifications FROM notification n JOIN family f ON f.family_id = n.family_id WHERE n.is_read = 0 GROUP BY f.family_id;
```



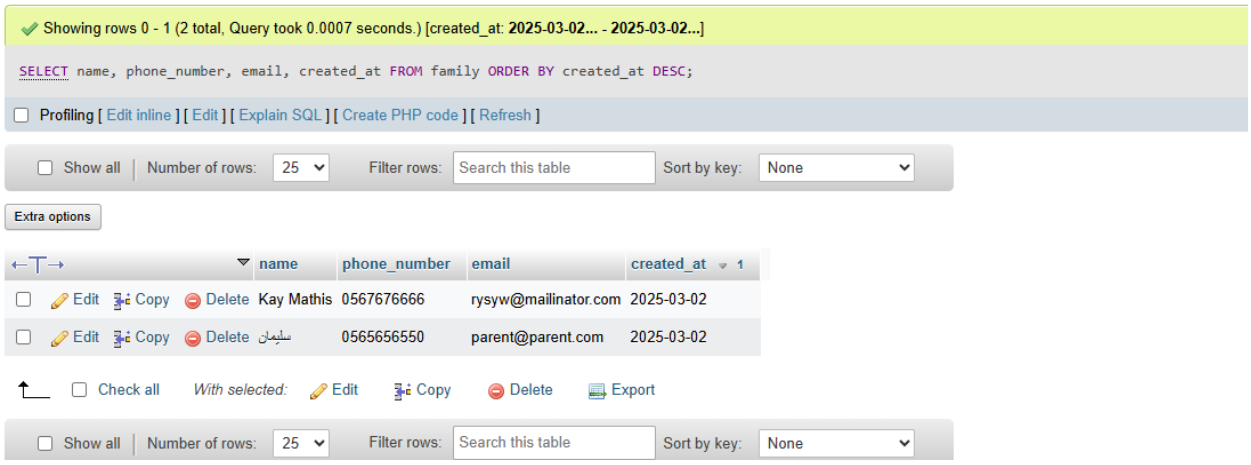
7. Report Name: Family Registration History

Description: Lists families and their registration date.
SQL CODE:

SELECT name, phone_number, email, created_at

FROM family

ORDER BY created_at DESC;



8. Report Name: Unread Reports by Family

Description: Displays all reports that have not been read by the family.
SQL CODE:

```
SELECT r.content, r.created_at, f.name AS family_name

FROM report r

JOIN family f ON f. family_id= r.family_id

WHERE r.is_read = 0;
```

Conclusion and Recommendations

Project Achievements

The Hafez System has successfully delivered a comprehensive platform that addresses the core challenges in Quran memorization tracking. The system provides a user-friendly interface for teachers to manage study circles, track attendance, and evaluate student progress, while giving students and parents visibility into performance metrics and comparative standing. Key achievements include:

1. Centralized student management and progress tracking system
2. Enhanced communication mechanisms between teachers and parents
3. Motivational tools through leaderboards and point systems
4. Comprehensive reporting capabilities for all stakeholders
5. Secure, web-based architecture accessible from any device

Recommendations for Future Development

1. Mobile Application Development

- Develop dedicated mobile applications for iOS and Android platforms
- Implement push notifications for real-time updates
- Add offline capability for areas with limited internet connectivity

2. Advanced Analytics

- Implement predictive analytics to identify students at risk of falling behind
- Develop personalized learning path recommendations based on performance data
- Create advanced visualization tools for progress tracking

3. Expanded Features

- Integrate audio recording and evaluation tools for recitation assessment
- Add direct messaging capabilities between teachers and students
- Implement automated scheduling for study circles and evaluations

4. System Integration

- Develop APIs for integration with other educational management systems
- Create export capabilities to standard educational reporting formats
- Enable integration with popular calendar applications

5. Scalability Enhancements

- Optimize database structure for handling larger user volumes
- Implement caching mechanisms for improved performance
- Develop a microservices architecture for better scalability

The Hafez System represents a significant advancement in technology-enabled Quran memorization tracking. By implementing these recommendations, the system can further enhance its value proposition and address additional needs within the educational community while maintaining its core focus on improving the memorization experience for students, teachers, and parents.

Reference List

- Mozilla Developer Network. (n.d.). *HTML, CSS, and JavaScript documentation*. Retrieved from https://developer.mozilla.org
- W3Schools. (2023). *JavaScript Objects*. Retrieved from https://www.w3schools.com/js/js_objects.asp
- PHP.net. (n.d.). *PHP Manual*. Retrieved from https://www.php.net/docs.php
- MySQL. (n.d.). *MySQL 8.0 Reference Manual*. Retrieved from https://dev.mysql.com/doc/
- OWASP Foundation. (n.d.). *OWASP Top Ten Web Application Security Risks*. Retrieved from https://owasp.org
- UNESCO. (n.d.). *Education Sector*. Retrieved from https://www.unesco.org/en/education
- Welling, L., & Thomson, L. (2016). PHP and MySQL web development (5th ed.). Addison-Wesley.
- Manning, M. (2015). Modern PHP: New features and good practices. O'Reilly Media.
- laboratory and in life. Journal of Personality and Social Psychology, 78(4), 772–790. https://doi.org/10.1037/0022-3514.78.4.772
- Pressman, R. S. (2014). Software engineering: A practitioner's approach (8th ed.). McGraw-Hill Education
- **GitHub.** (n.d.). *The world's leading software development platform*. Retrieved from https://github.com
- Let's Encrypt**. (n.d.). *Free SSL Certificates for HTTPS*. Retrieved from https://letsencrypt.org**
- GoDaddy SSL. (n.d.). Secure website certificates. Retrieved from https://www.godaddy.com/web-security/ssl-certificate
- **Visual Studio Code**. (n.d.). *Code editor used for web development*. Retrieved from https://code.visualstudio.com
- Saudi Vision 2030 – Education Objectives**. (n.d.). *Digitizing education and supporting e-learning*. Retrieved from https://www.vision2030.gov.sa**
- **Islamic Online University**. (n.d.). *Islamic learning through online platforms*. Retrieved from https://iou.edu.gm

Appendix

Demo of the System:

https://drive.google.com/drive/folders/1keFHvoYNWGbcbdLkN5xf_6LSpFCPQ7l7?usp=drive_link

Code and Database :

https://drive.google.com/drive/folders/1voRHCz3UrlrUWD7tFarPK6zySDFv3GOe?usp=drive_link

https://github.com/SulAlfuhaid/Hafed-Capstone-Project/tree/main