

# Learning Lantern

Web Based

Learning Management System Project

*Under the guidance of*  
**Dr. Mustafa Salem Abu-Bakr**

Faculty of Computer and Information  
Assiut University  
**22/6/2022**

## PROJECT TEAM MEMBERS

No.	Name	Dep.	Role
1	Ahmed Ezzat Gaber أحمد عزت جابر	CS	Backend Developer
2	Omnia Ahmed Bakr أمنية أحمد بكر	IS	Backend Developer
3	Ahmed Mohamed Fathy أحمد محمد فتحي	CS	Backend Developer
4	Ahmed Samir Meky أحمد سمير مكي	CS	Backend Developer
5	Omnia Mohamed أمنية محمد عبد العال	IS	Frontend Developer
6	Aya Abo El-Ela Khaled آيه أبو العلا خالد	IS	Frontend Developer
7	Ibrahim Mohamed Ali إبراهيم محمد علي	IS	Frontend Developer
8	Khlood Elhuisseney خلود الحسيني	MM	Backend Developer

## LIST OF FIGURES

Figure	Page No.
Figure 1: Database Design	10
Figure 2: Design Architecture	14
Figure 3: User API Architecture	15
Figure 4: User Service	15
Figure 5: User Controller	16
Figure 6: Authorization service	16
Figure 7: Sign up	17
Figure 8: Mail Service	17
Figure 9: Validate Email	18
Figure 10: Sign In	18

# TABLE OF CONTENTS

Content	Page No.
Title Page	1
List of Team Members	2
List of Figures	2
Chapter-1	5
Introduction of the project	5
1.1. What we will Create.....	6
1.2. Project Objective.....	6
1.3. Project Overview.....	6
1.4. Services of Site.....	7
1.5. Technology used.....	8
1.6. Advantages of Site.....	8
1.6.1. Just for Learning (not general purpose).	8
1.6.2. On demand Services.....	8
1.6.3. Lectures.....	8
Chapter-2	9
Database Design & Implementation	9
2.1. Database Design.....	10
2.2. Database Implementation.....	11
Chapter-3	12
Construction & Implementation	12
3.1. Design Architecture.....	14
3.2. Authentication and Authorization.....	15
3.2.1. User API Architecture.....	15
3.2.2. User Service .....	16
3.2.3. User Controller.....	16
3.2.4. Authorization service.....	17
3.2.5. Sign Up.....	17
3.2.6. Validate Email.....	18
3.2.7. Sign In.....	18
3.2.8. Roles .....	19
3.3. Classroom.....	19
3.3.1. Instant Chat.....	20
3.3.2. Video Chat.....	20
3.3.3. Calendar.....	20

3.3.4. Text Lesson.....	20
3.3.4.1. Interactive Video.....	21
3.4. To-do.....	21
	22
<b>Chapter-4</b>	23
<b>System Testing</b>	23
4.1. Framework used.....	24
4.2. Tests.....	24
<b>Chapter-5</b>	24
<b>Snapshots</b>	

# Chapter 1

## Introduction

- 1.1. What we will create.....
- 1.2. Project Objective.....
- 1.3. Project Overview.....
- 1.4. Services of Site.....
- 1.5. Technology used.....
- 1.6. Advantages of Site.....
  - 1.6.1. Just for Learning (not general purpose).
  - 1.6.2. On demand Services.....
  - 1.6.3. Lectures.....

### 1.1. What we will Create:

We will create a group of integrated services that enhance the E-learning experience and offers simplicity of use and convenience of Usage

### 1.2. Project Objective:

Building a set of services to help educational institutions moving to online learning

### 1.3. Project Overview:

1. This Project is about helping universities & educational institutions moving to Online Learning easily by providing a set of useful services that are essential in the online learning process.
2. The universities & educational institutions can subscribe to one of more of the provided services according to their needs.

## 1.4. Services:

### **1. virtual classroom:**

This is the core of the project and the most important service, this service enables instructors and students to:

- Interact through live lectures
- Assign Projects
- Take in Lecture Quizzes
- Take Attendance
- Calculate Participation of Each student
- instant chat (audio , video , text , share screen)

### **2. online text editor:**

This service allows Instructors to:

- Create online courses (ex. W3schools)
- The ability to use text, images, tables, charts, inline Quizzes
- The course pages should track if the student has visited and completed each lesson. (ex. Udemy courses)

### **3. video hosting**

This service allows instructors to:

- Host recorded lessons.
- Create in video quizzes at a given timestamp.

### **4. Calendar & To-do List**

These are two services that are highly related that they are considered as ,

1) This service allows instructors to:

- Schedule lectures, quizzes & exams, these schedule . should be added automatically to students' calendar.
- Automatically search for common available times for all students before scheduling any activity.

2) This service allows students to:

- Create personal to-do Lists to organize their works.
- Create collaborative to do Lists to organize teams works.
- Create pre-defined to-do lists

## 1.5. Technology:

### **Frontend:**

- 1) Html5, css3, JavaScript.
- 2) Angular.
- 3) VedioJS.
- 4) TinyMCE.

### **Backend:**

- 1) ASP .Net core
- 2) Entity framework
- 3) SQL Server
- 4) Microsoft Azure

## 1.6. Advantages of Site:

### **1.6.1. Just for Learning (not general purpose):**

Our System focuses only on the e-learning process more than the communications.

### **1.6.2. On demand Services:**

Each university can only pay for the services it needs.

### **1.6.3. Lectures:**

- We record all live lectures so they will be accessible all time.
- The lectures can have quizzes at a given timestamp.



# Chapter 2

## Database Design & Implementation

<u>2.1.</u> Database Design.....	
<u>2.2.</u> Database Implementation.....	

## 2.1. Database Design:

- We follow a standard step to design the database:

- 1) Build a conceptual model.
- 2) Build a logical model.
- 3) Build a physical model.
- 4) Identify the primary keys.
- 5) Identify the Foreign keys.
- 6) Normalize the database:

Our database is normalized in the 3rd normal form.

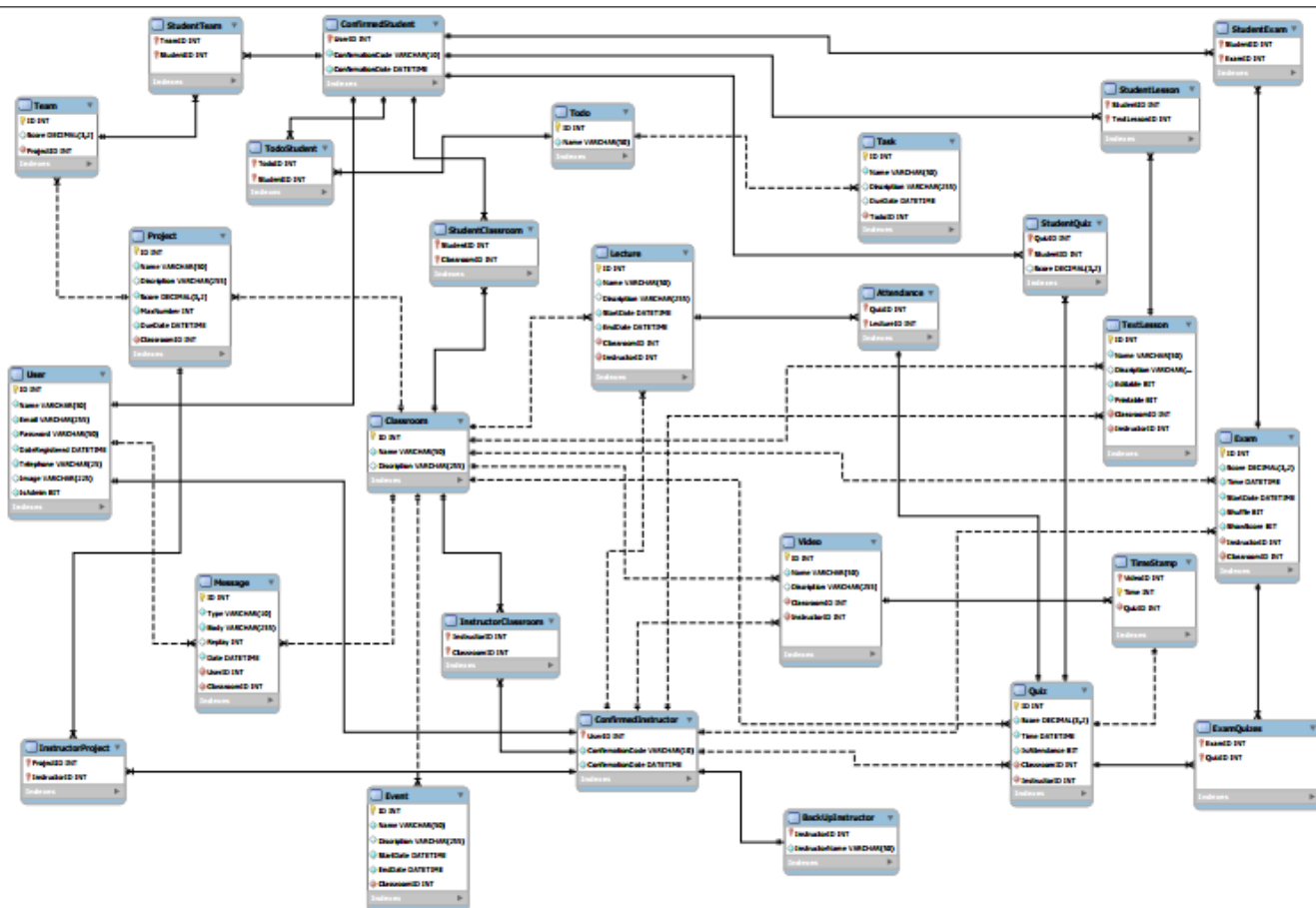


Figure 1: Database Design

### 2.3. Database Implementation:

1. We used SQL language to implement our database.
2. We used SQL Server Management Studio (SSMS).
3. Our database model consists of 28 tables including bridge tables.

# Chapter 3

## Construction & Implementation

- 3.1. Design Architecture.....
  - 3.2. Authentication and Authorization.....
    - 3.2.1. User API Architecture.....
    - 3.2.2. User Service .....
    - 3.2.3. User Controller.....
    - 3.2.4. Authorization service.....
    - 3.2.5. Sign Up.....
    - 3.2.6. Validate Email.....
    - 3.2.7. Sign In.....
    - 3.2.8. Roles.....
  - 3.3. Classroom.....
    - 3.3.1. Chat.....
    - 3.3.2. Video Chat.....
    - 3.3.3. Calendar.....
    - 3.3.4. Text Lesson.....
      - 3.3.4.1. Interactive Video.....
  - 3.4. To-do.....
- Frontend Implementation:

Use *HTML, CSS, Java script, Angular framework, Video JS, and TinyMCE* to design and implement website pages like:

- Landing page (Home)
- Log in page
- Sign up page
- Classroom page
- Chat
- Text Lesson
- Etc.....

➤ Backend Implementation:

Use *Microservice* Architecture and *Web API*.

Schema used to implement each service's API is:

Each service is a project itself that contain the following:

- **Controller:**

It handles incoming HTTP requests and send response back to the caller.

- **Data Transfer Objects (DTOs):**

Is a type that has no behavior, only state. DTOs are frequently used to transfer data over some medium, being serialized and then deserialized in the process. Behavior (methods, logic) doesn't transfer.

- **Models:**

A *model* is an object that represents the data in your application. ASP.NET Web API can automatically serialize your model to JSON, XML, or some other format, and then write the serialized data into the body of the HTTP response message.

- **Repositories:**

With the Repository, we create an abstraction layer between the data access and the business logic layer of an application. By using it, we are promoting a more loosely coupled approach to access our data from the database. Also, the code is cleaner and easier to maintain and reuse. Data access logic is in a separate class or sets of classes called a repository.

### 3.1. Design Architecture:

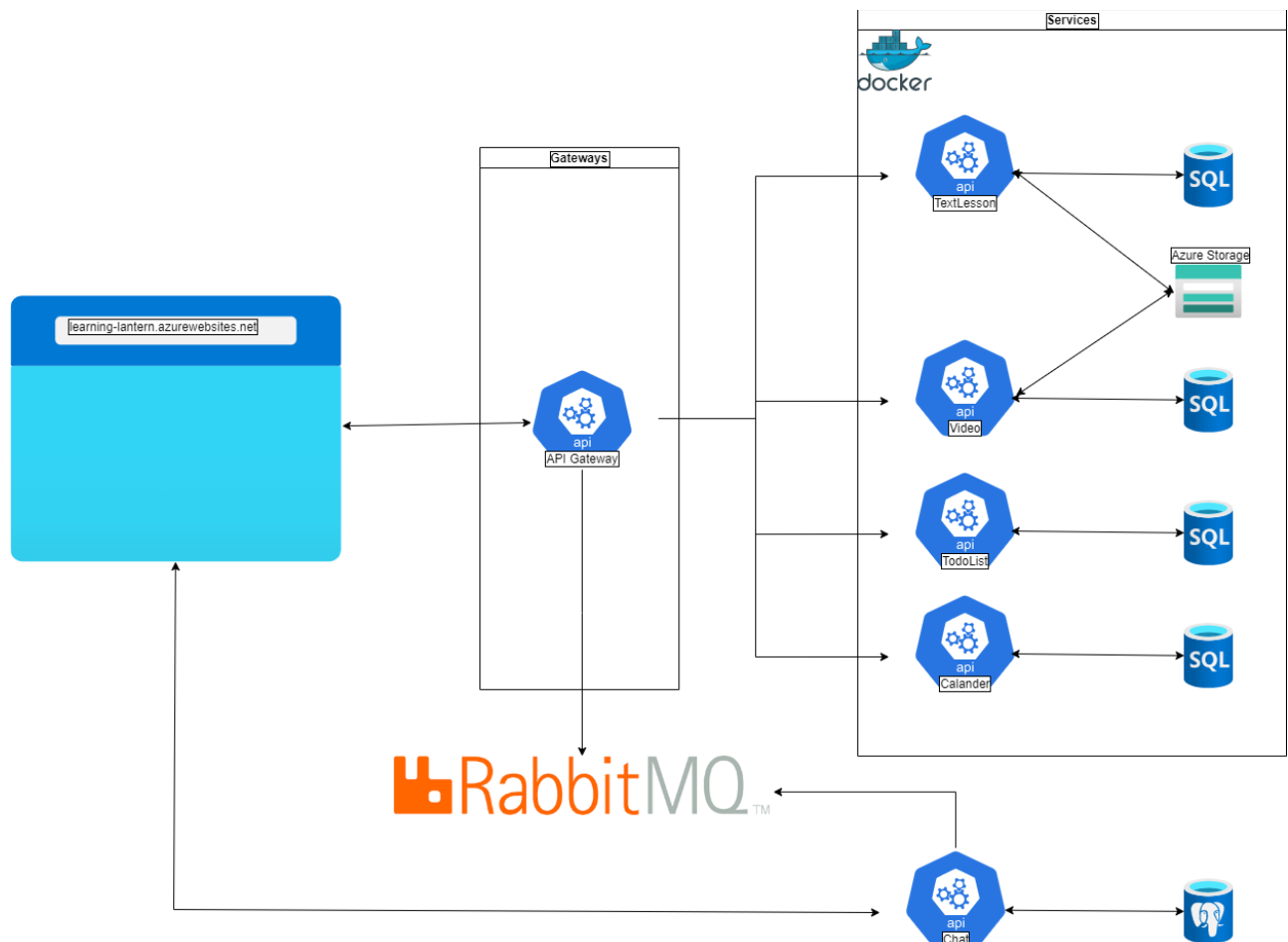


Figure 2: Design Architecture

## 3.2. Authentication and Authorization:

### 3.1.1. User API Architecture:

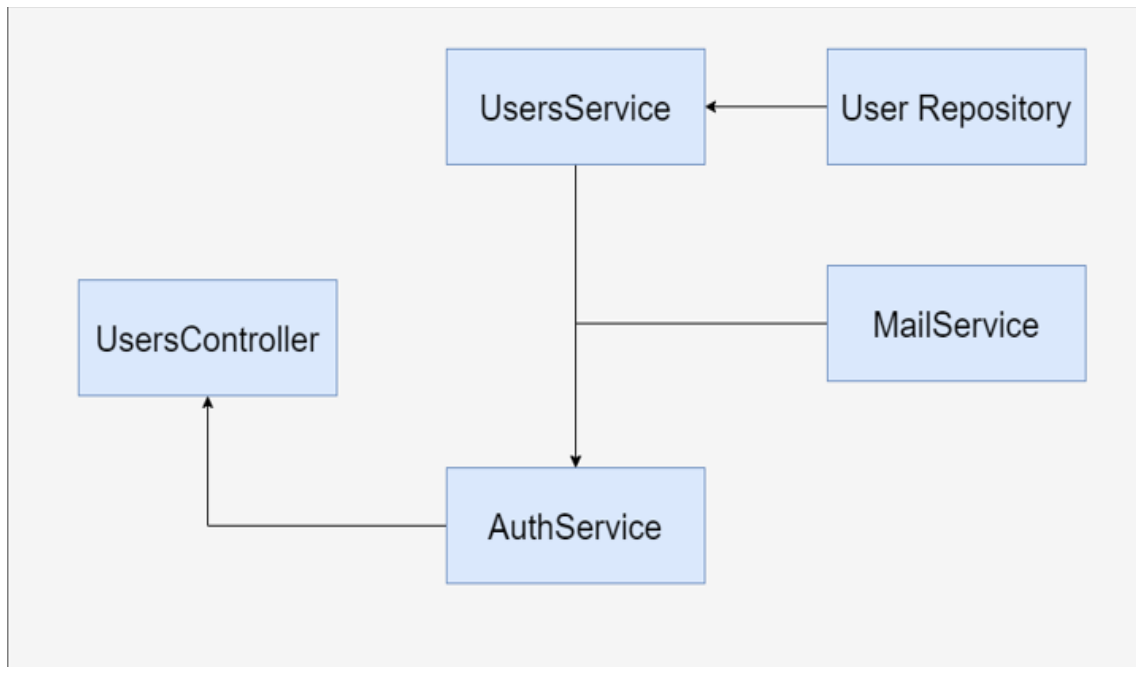


Figure 3: User API Architecture

### 3.1.2. User Service:

Method	Description
CreateUser(Email:string, Password:string, FirstName:string, LastName:string, Image?:[]byte)	Create new user entity and saves it to the database #return userId
FindById(Id:string)	Find the user with the given id
FindByEmail(Email:string)	Find the user with the given email, this function is used to check if the email already registered or not
UpdateUser(Id:string, user: Partial<UserEntity>)	update the user with the given id

Figure 4: User Service

### 3.1.3. User Controller:

Method & Route	Body	Description
POST auth/signup	{userUniversity,userFName, userMName, userLName, userEmail, userPassword}	Create new user and login
POST auth/login	{userEmail, userPassword}	Login an existing user
PATCH auth/validate-email	{userId, validationCode}	validate the email of the user with the given id
PATCH User/{userId}	{id:string, user: Partial<UserEntity>}	update the user with the given id

Figure 5: User Controller

### 3.1.4. Authorization service:

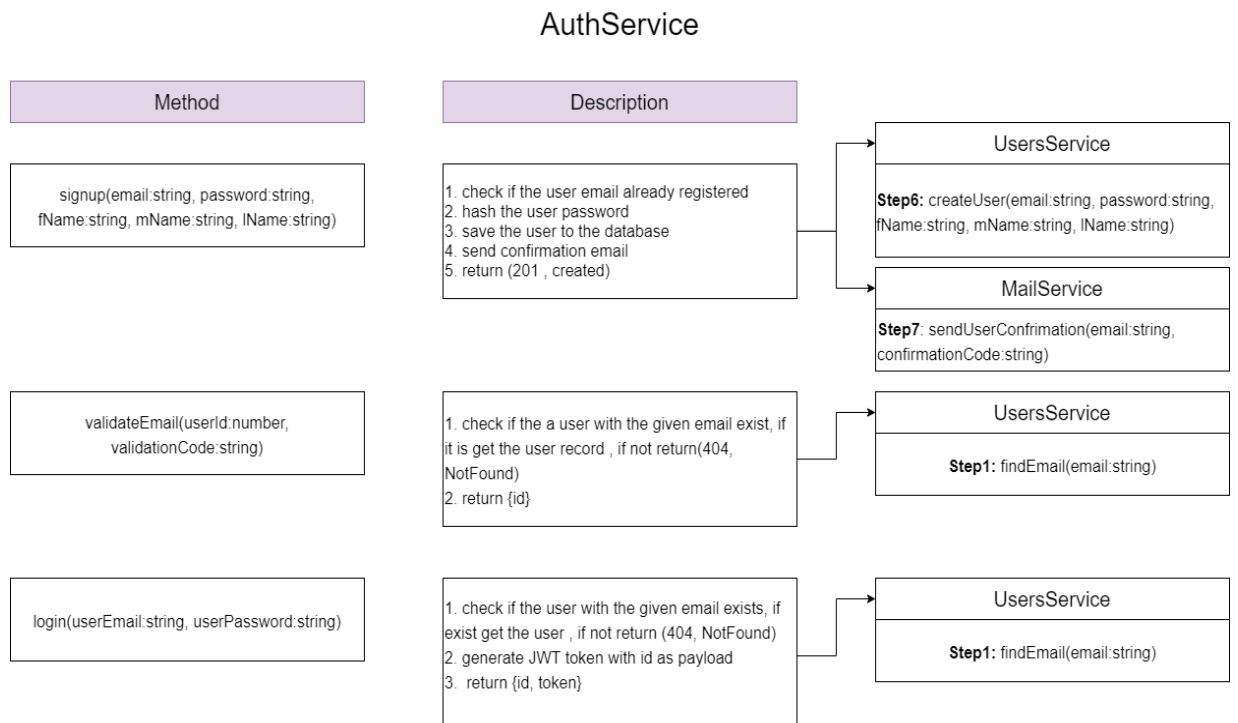


Figure 6: Authorization service



### 3.1.5. Sign Up or Registration:

In this, the interested users get registered by selecting their desired first name, last name, password and by providing other necessary details.

Each user can register only one time. Details of each person along with their username and password are saved permanently in the database.

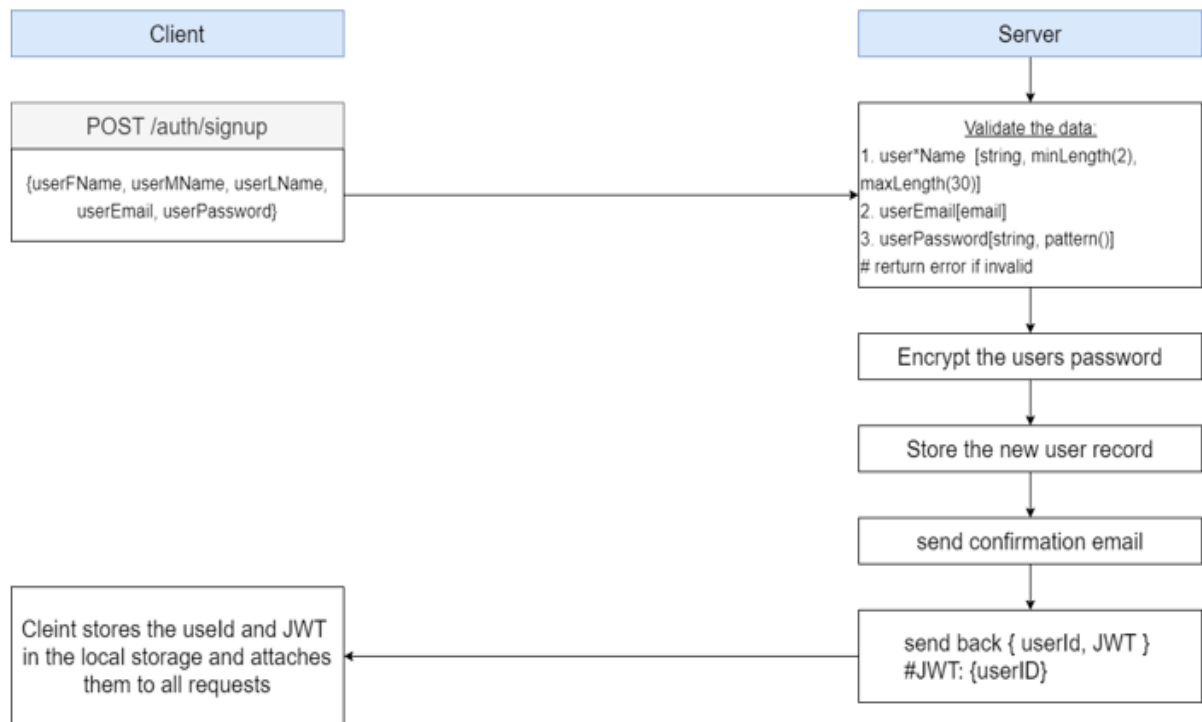


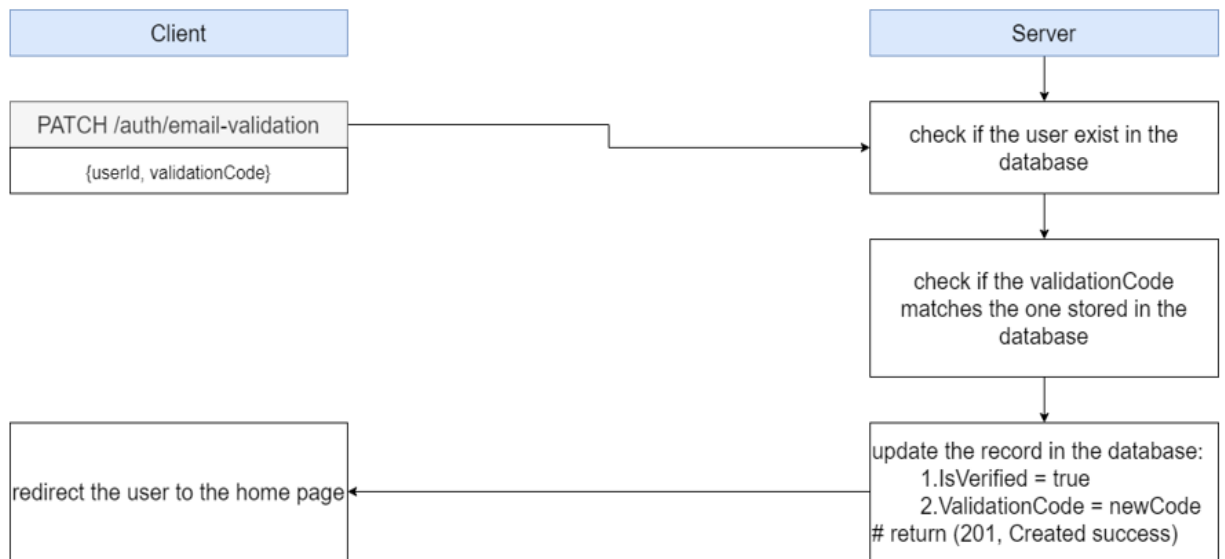
Figure 7: Sign up

### 3.1.6. Validate Email:

After the registration process, the user receives a confirmation message to the email he registered with.

Method	Description	Template
SendUserConfirmation(Email string, Id string, ConfirmationCode string)	Sends validation link to the given email #ValidationLink is on the following format: https://learning-lantern.web.app/en/auth/confirmation/{userId}/{confirmationCode}	<pre> &lt;h1&gt;Welcome To Learning Lantern&lt;/h1&gt; &lt;br&gt; &lt;p&gt;Thanks for registering at learning lantern please click &lt;strong&gt;&lt;a href=" ValidationLink"&gt; target="_blank"&gt;here&lt;/a&gt;&lt;/strong&gt; to activate your account&lt;/p&gt; </pre>

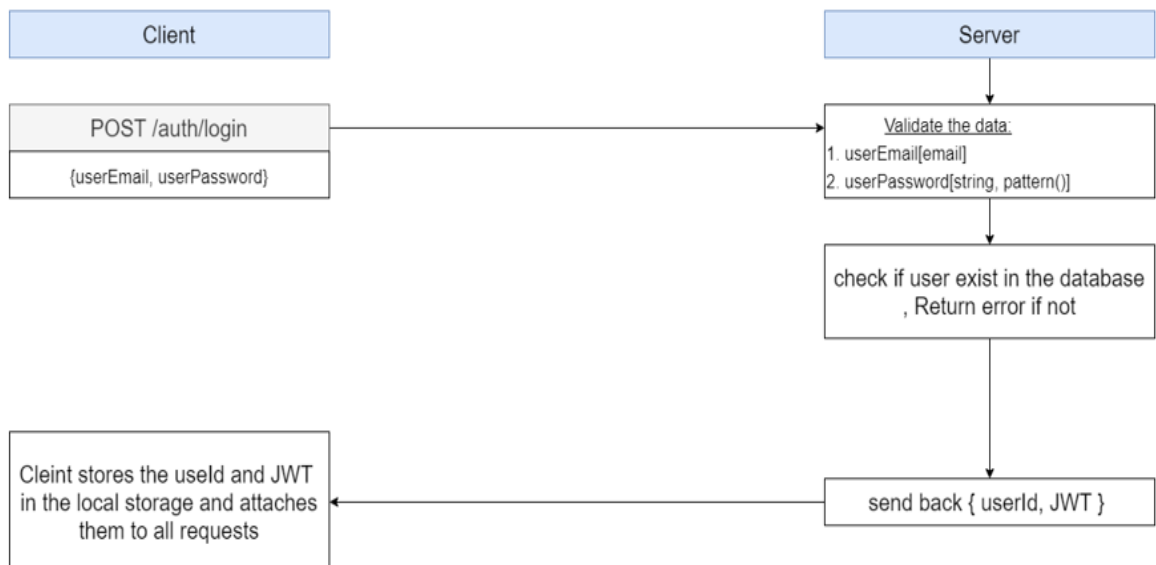
Figure 8: Mail Service



**Figure 9: Validate Email**

### 3.1.7. Sign In:

After confirmation process, the user can try to sign in by entering the university name, email, and password.



**Figure 10: Sign In**

### 3.1.8. Roles:

- **There are 4 types of roles on the website:**

1. Admin
2. University admin
3. Instructor
4. Student

- **Admin:**

Developers of the website

- **University admin:**

University managers and administrators

- **Instructor:**

Doctors and assistants

- **Student:**

University students

## 3.2. Classroom:

This is the core of the project and the most important service, it contains a major set of services provided by website.

### 3.2.1. Instant Chat:

Instructors and students stay connected via real time Instant Messages.

- **Permissions:**

1. Admin:

Has a permission to chat but doesn't need it.

2. University Admin:

Has a permission to chat but doesn't need it.

3. Instructor:  
Has a permission to chat.
4. Student:  
Has a permission to chat.

### **3.2.2. Video Chat:**

Connect students and instructors together via Interactive Online Meetings which emulate the real learning Environment.

- Permissions:
  1. Admin:  
Has a permission to schedule meeting but doesn't need it.
  2. University Admin:  
Has a permission to schedule meeting but doesn't need it.
  3. Instructor:  
Has a permission to schedule meeting.
  4. Student:  
Has a permission to schedule meeting.

### **3.2.3. Calendar:**

Students can keep organized with our integrated Calendar.

- Permissions:
  1. Admin:  
Has a permission to use a calendar but doesn't need it.
  2. University Admin:  
Has a permission to use a calendar but doesn't need it.
  3. Instructor:  
Has a permission to use a calendar.
  4. Student:  
Has a permission to use a calendar.

### 3.2.4. Text Lesson:

Use our Online Text Editor to create text lessons for the student, including inline quizzes and progress monitor.

- Permissions:

1. Admin:

Has a permission to use a Text Editor but doesn't need it.

2. University Admin:

Has a permission to use a Text Editor but doesn't need it.

3. Instructor:

Has a permission to use a Text Editor.

4. Student:

Can access it but has not a permission to use a Text Editor.

#### 3.2.4.1. Interactive Video:

Learning is available anywhere any time using our Video hosting service with interactivity features like in-video quizzes, caption...and more.

- Permissions:

1. Admin:

Has a permission to make interactive video but doesn't need it.

2. University Admin:

Has a permission to make interactive video but doesn't need it.

3. Instructor:

Has a permission to make interactive video.

---

4. Student:

Can access it but has not a permission to make interactive video.

### 3.3. To-do:

Students can keep organized with our integrated To-do list service.

- Permissions:

1. Admin:

Has a permission to use To-Do list but doesn't need it.

2. University Admin:

Has a permission to use To-Do list but doesn't need it.

3. Instructor:

Has a permission to use To-Do list but doesn't need it.

4. Student:

Has a permission to use To-Do list.

# Chapter-4

# System Testing

4.1. Framework used.....	
4.2. Tests.....	

We use a test scenario strategy, for each endpoint, we test every possible scenario that can happen by the user.

First, we Mock the database by using a new In-Memory database.

Then for each repository we write a unit testing for each function.

#### **4.1. Framework used:**

1. xUnit.net: is a free, open source, community-focused unit testing tool for the .NET Framework.
2. Moq: is a free mocking library, we use it to mock the dependency need for each repository.
3. Faker and AutoFixture: to generate random objects.

#### **4.2. Tests:**

##### **1. To-do List**

- a. Add Task
  - i. Add A new Task
- b. Update Task
  - i. Update existing Task
  - ii. Update Task request with Id not exists in database
- c. Remove Task
  - i. Remove existing Task
  - ii. Remove Task request with Id not exists in database

##### **2. Calendar**

- a. Add Event
  - i. Add A new Events
- b. Update Event
  - i. Update existing Event



- ii. Update Event request with Id not exists in database

c. Remove Event

- i. Remove existing Event
- ii. Remove Event request with Id not exists in database

# Chapter-5

## Snapshots

# Desktop View

[HOME](#)[SERVICES](#)[PRICES](#)[CONTACT US](#)[PROFILE](#) ▼

## Welcome to Learning Lantern

The best [Online Educatiaonal Environment](#), with an integrated and easy-to-use set of [cloud services](#) and [content creating tools](#) for leveraging the online eductaion processes.

[Get Started](#)[HOME](#)[SERVICES](#)[PRICES](#)[CONTACT US](#)[PROFILE](#) ▼



## Our Services




### Live Lectures

Connect students and instructors together via [Interactive Online Meetings](#) which emulates the real learning environment

[Demo](#)

[HOME](#)[SERVICES](#)[PRICES](#)[CONTACT US](#)[PROFILE](#)



## Our Services




### Interactive Video

Learning is available any where any time using our [Vidoe hosting service](#) with interactivity features like in-video quizzes, caption ...and more.

[Demo](#)

[HOME](#)[SERVICES](#)[PRICES](#)[CONTACT US](#)[PROFILE](#)


## Our Services



### Instant Chat

Instructors and students stay connected via realtime [Instant Messages](#).


[Demo](#)



HOME SERVICES PRICES **CONTACT US**

PROFILE ▾ ⚙


## Our Services



### Integrated Calendar & Todo-list

Students can keep organized with our [integrated Calender & ToDo list](#) service.


[Demo](#)



HOME SERVICES PRICES **CONTACT US**

PROFILE ▾ ⚙



## Our Services




### Online Exams & Quizzes

Test your student online using our [Online Exam System](#), with a handful of features like automatic grading, anti-cheat strategies, ...and more.

[Demo](#)


[HOME](#)
[SERVICES](#)
[PRICES](#)
[CONTACT US](#)
[PROFILE](#)




## Our Services



### Live Lectures

Connect students and instructors together via [Interactive Online Meetings](#) which emulates the real learning environment

[Demo](#)


[HOME](#)
[SERVICES](#)
[PRICES](#)
[CONTACT US](#)
[PROFILE](#)


## Pick Your Plane

### Free Plan

All prices are for 1 year subscription.

Calender & Todo List	Unlimited
Instant Chat	NA
Interactive Video	Up to 1 GB
Live Lectures	NA
Online Exam & Quizes	5 Exams
Text Lessons	250 MB


[Get Your Portal](#)

### Premium Plan

All prices are for 1 year subscription.

Calender & Todo List	Unlimited
Instant Chat	Unlimited
Interactive Video	1 USD/1GB
Live Lectures	5 USD/student
Online Exam & Quizes	Unlimited
Text Lessons	1 USD/1GB

[Get Your Portal](#)



HOME SERVICES PRICES CONTACT US PROFILE

### Contact Us


Your Email


Your Name

Your Email

Your Message


Send





Learning Lantern

Learning Lantern helps Universities, Institutes, or Companies to start their own Online Learning Portal. With easy to use on demand services.



HOME SERVICES PRICES CONTACT US LOGIN

### SIGNUP

University:

Your FullName:

Your Email:

Your Password:

Confirm Password:

☐ I Agree on [Terms & Conditions](#)

[Already have account? login here](#)

CREATE ACCOUNT



## LOGIN

University:

Your Email:

Your Password:

[Don't have account? Signup here](#)

LOGIN



Learning Lantern

Learning Lantern helps Universities, Institutes, or Companies to start their own Online Learning Portal. With easy to use on demand services.

Copyright © 2022 Learning-Lantern

All Rights Reserved





August 2022


today



Sun	Mon	Tue	Wed	Thu	Fri	Sat
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3





PROFILE ▾ 

## Todo List

Create Task



My Day



All Tasks


Important

Completed

★ task2





PROFILE ▾ 

## Todo List

Create Task



My Day

All Tasks



Important

Completed



★ task1

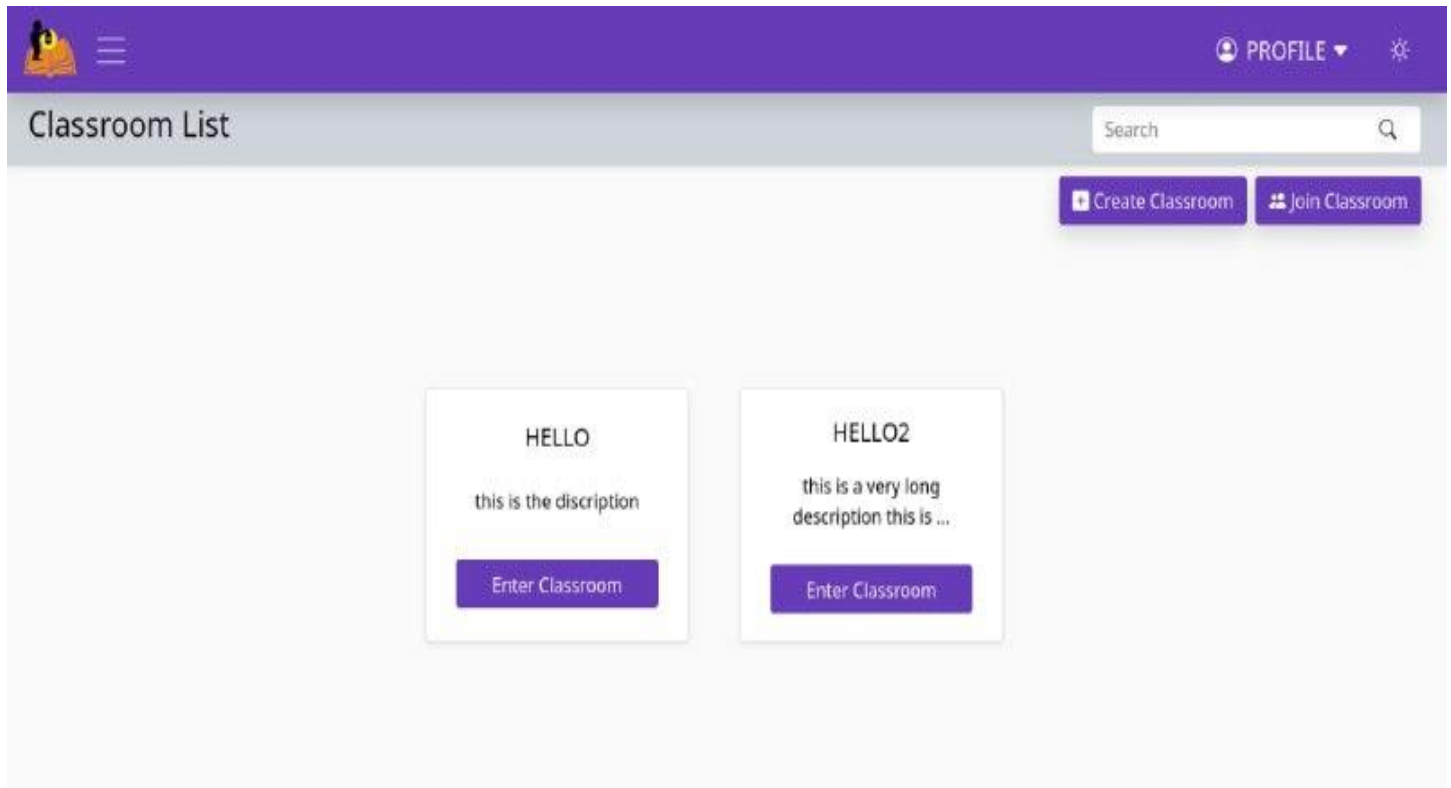


★ task1

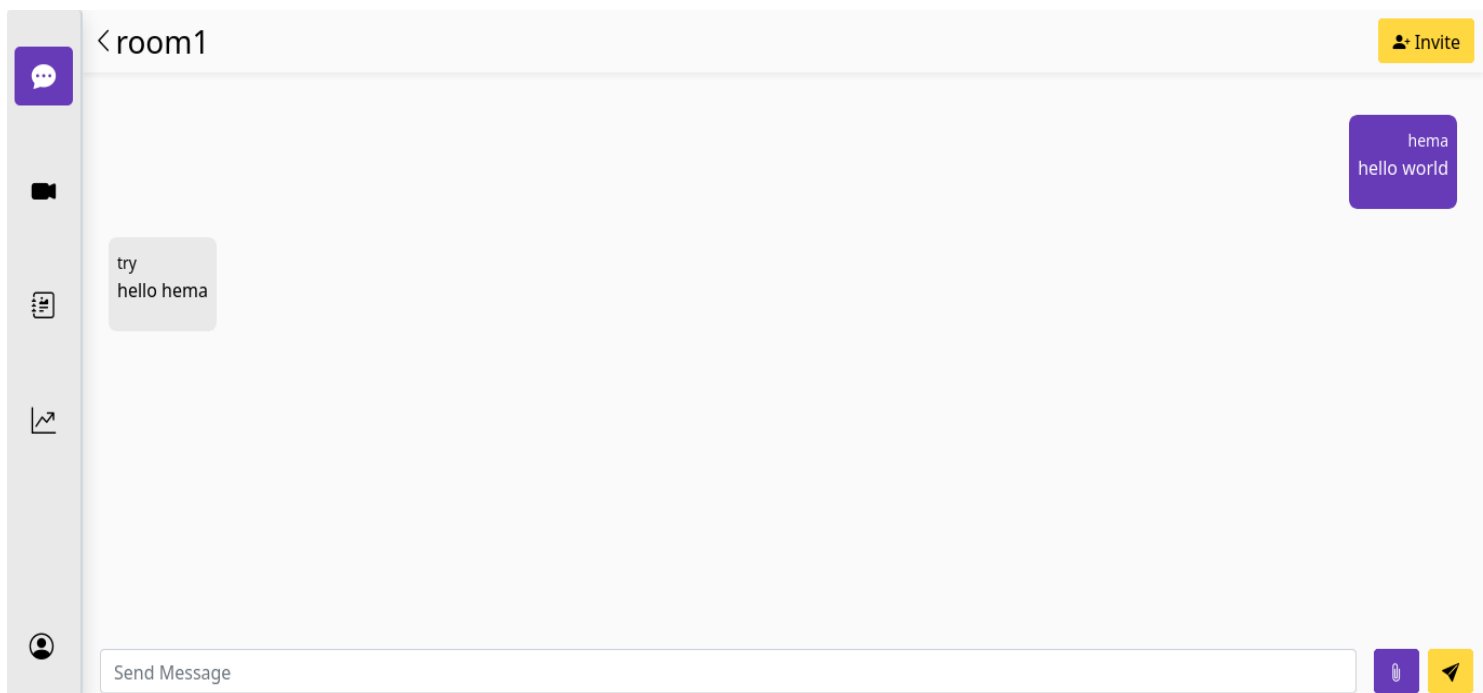


★ task2





## Chat



# Text Lesson View

< Class Title

Lesson Title

EditVideo

## What Is a Database?

A database is an organized collection of structured information, or data, typically stored electronically in a computer system. A database is usually controlled by a [database management system \(DBMS\)](#). Together, the data and the DBMS, along with the applications that are associated with them, are referred to as a database system, often shortened to just database.

Data within the most common types of databases in operation today is typically modeled in rows and columns in a series of tables to make processing and data querying efficient. The data can then be easily accessed, managed, modified, updated, controlled, and organized. Most databases use structured query language (SQL) for writing and querying data.

## What is Structured Query Language (SQL)?

SQL is a programming language used by nearly all [relational databases](#) to query, manipulate, and define data, and to provide access control. SQL was first developed at IBM in the 1970s with Oracle as a major contributor, which led to implementation of the SQL ANSI standard. SQL has spurred many extensions from companies such as IBM, Oracle, and Microsoft. Although SQL is still widely used today, new programming languages are beginning to appear.

# Text Lesson Editor

< Class Title

Lesson Title

Save

Video

Edit View Format Tools Table



↶ ↷


Heading 1



**B** *I*


≡ ≡ ≡ ≡

≡ ≡ ≡ ≡



A  



## What Is a Database?

A database is an organized collection of structured information, or data, typically stored electronically in a computer system. A database is usually controlled by a [database management system \(DBMS\)](#). Together, the data and the DBMS, along with the applications that are associated with them, are referred to as a database system, often shortened to just database.

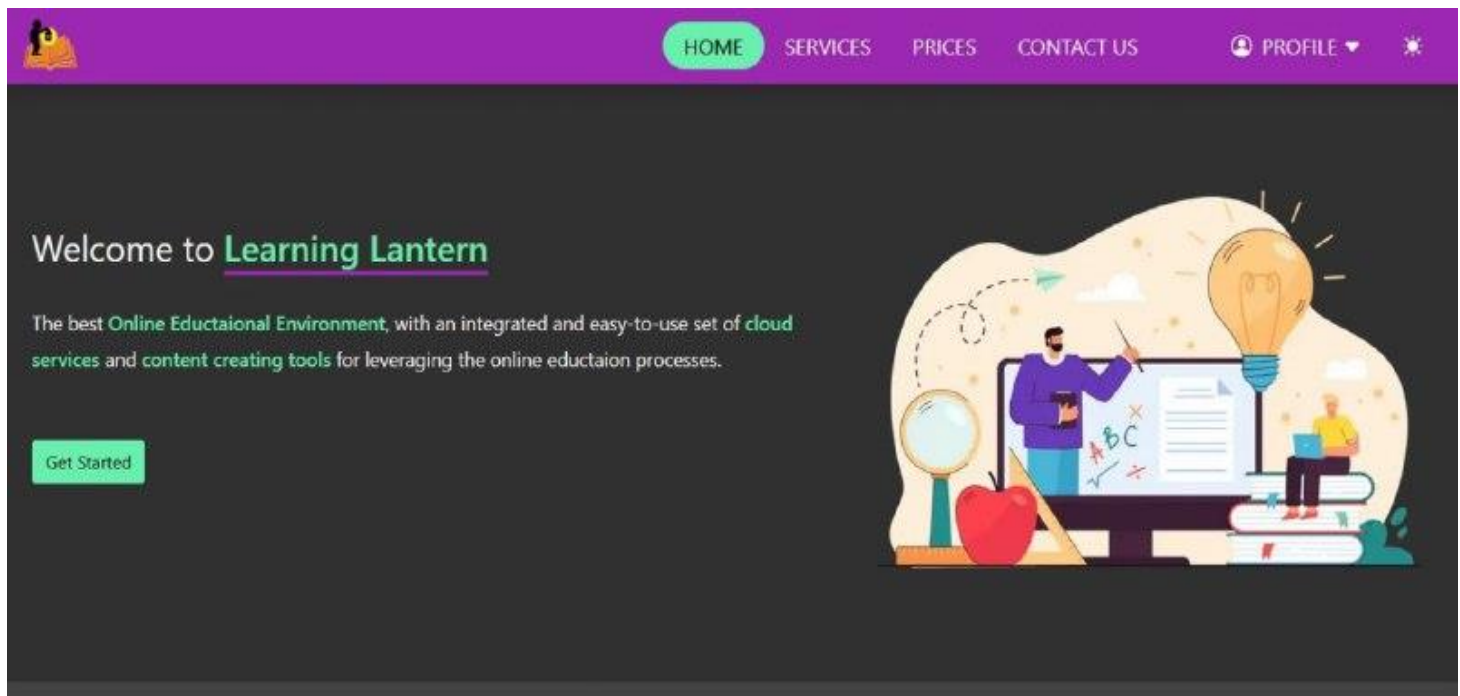
Data within the most common types of databases in operation today is typically modeled in rows and columns in a series of tables to make processing and data querying efficient. The data can then be easily accessed, managed, modified, updated, controlled, and organized. Most databases use structured query language (SQL) for writing and querying data.

## What is Structured Query Language (SQL)?

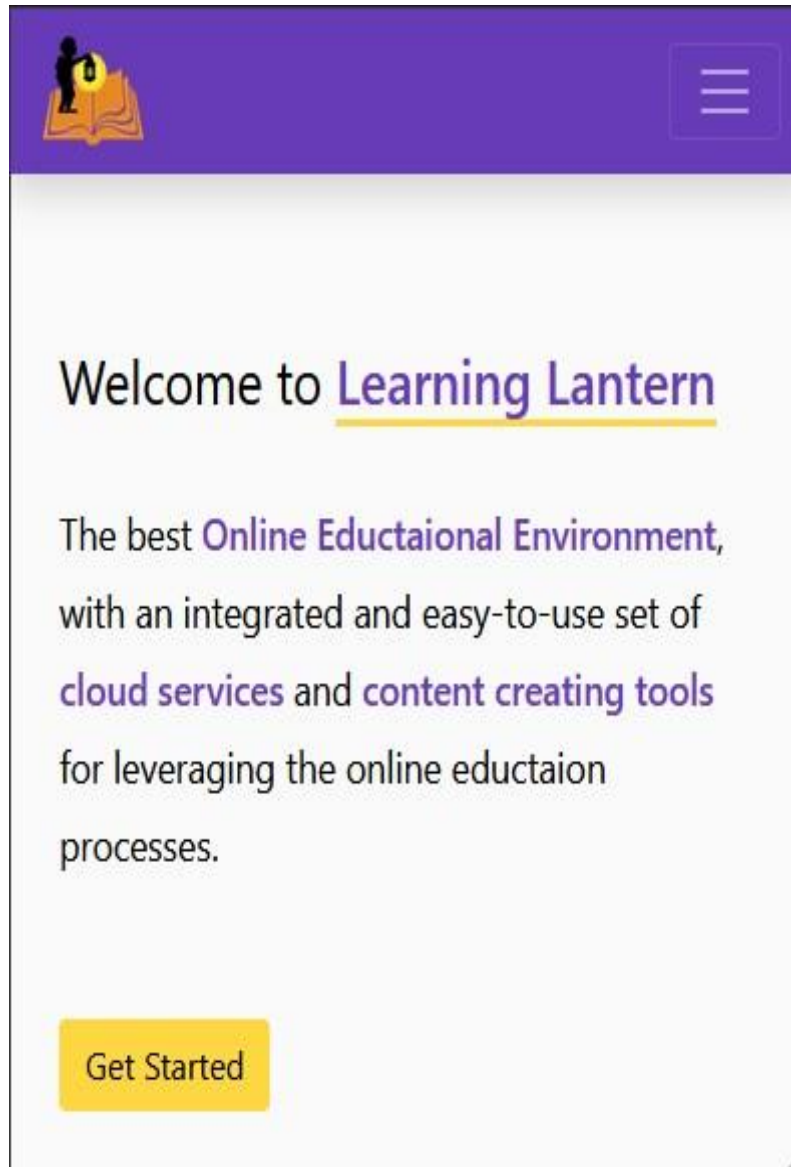
SQL is a programming language used by nearly all [relational databases](#) to query, manipulate, and define data, and to provide access control. SQL was first developed at IBM in the 1970s with Oracle as a major contributor, which led to implementation of the SQL ANSI standard, SQL has spurred many extensions from companies such as IBM, Oracle, and Microsoft. Although SQL is still widely used today, new programming languages are beginning to appear.

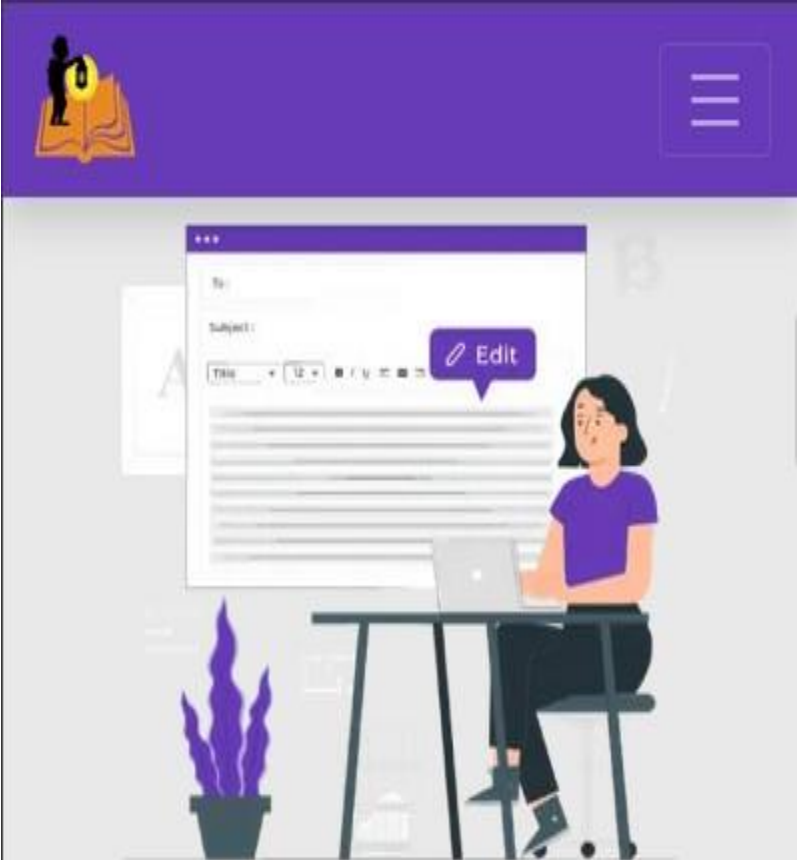
h1 > span194 words

# Dark Mode



# Mobile View





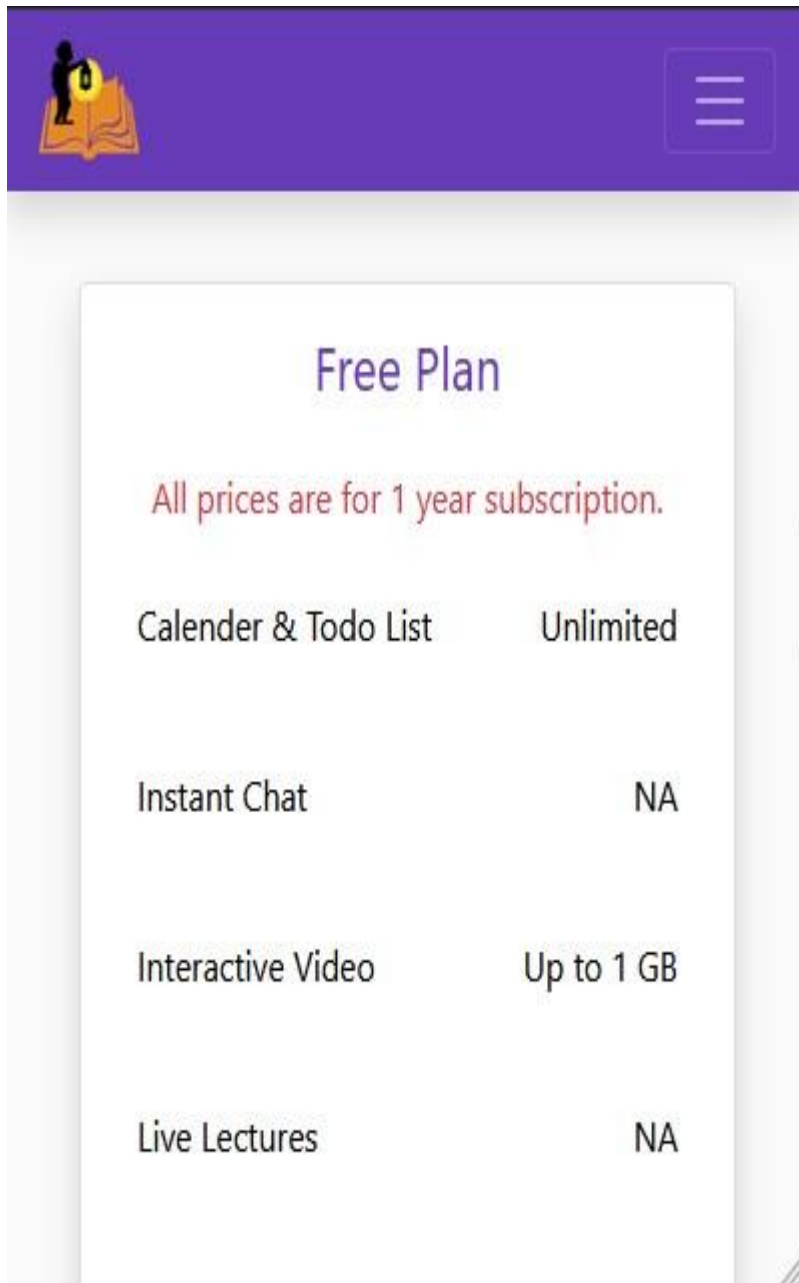
The illustration shows a person with dark hair, wearing a purple shirt and black pants, sitting at a black desk and working on a laptop. A large, semi-transparent window of a text editor is overlaid on the scene. The text editor has a purple header bar with three dots on the left. Below the header, there are fields for 'To:' and 'Subject:'. The main body of the text editor contains several lines of placeholder text. A purple 'Edit' button with a pencil icon is visible on the right side of the text editor. To the left of the person, there is a potted plant with purple leaves. The background is a light gray wall with some faint, stylized letters like 'A' and 'B'.

### Online Text Lesson

Use our [Online Text Editor](#) to create text lessons for the student, including inline quizez and progress monitor.



# Price

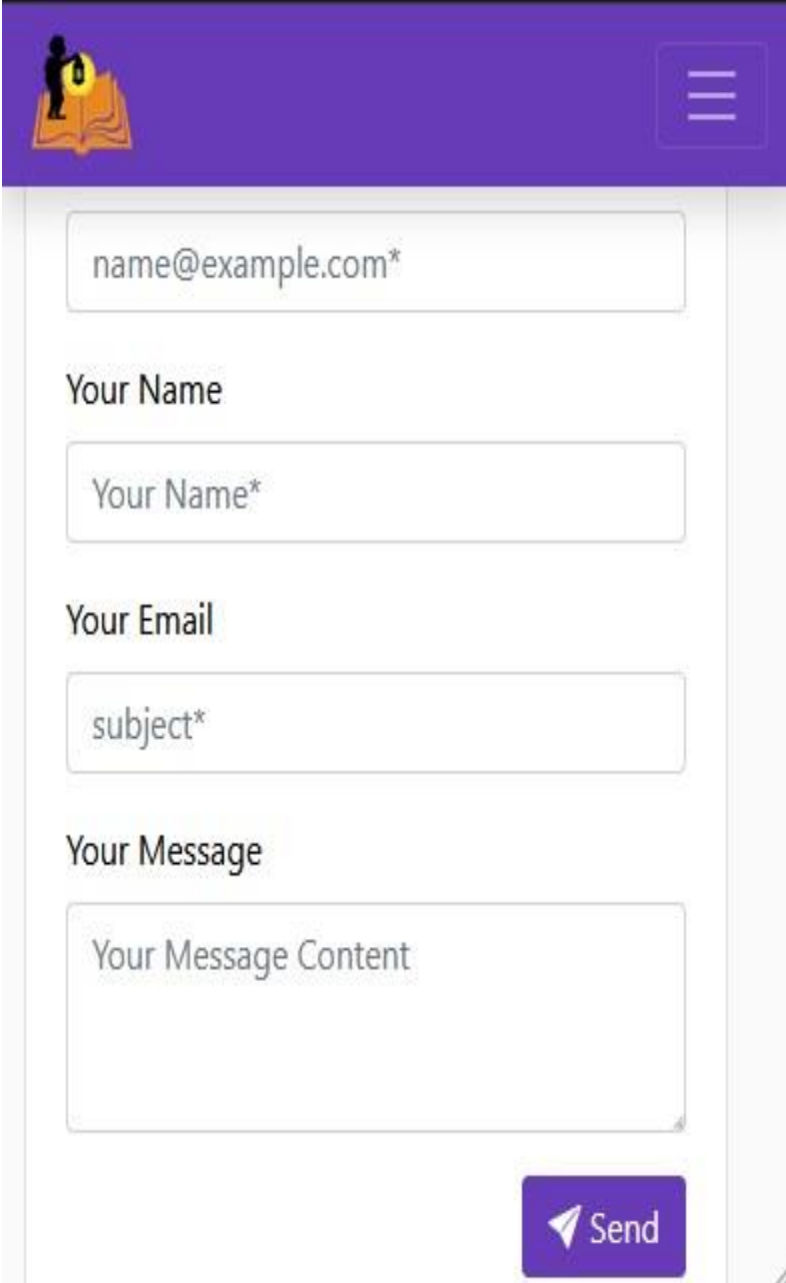


The image shows a mobile application interface for a 'Free Plan'. At the top is a purple header bar. On the left of the header is an icon of a person standing on an open book with a yellow circle above it. On the right is a hamburger menu icon. Below the header, the text 'Free Plan' is centered in a purple font. A red note states 'All prices are for 1 year subscription.' Below this, a table lists features and their availability. The features are: Calender & Todo List (Unlimited), Instant Chat (NA), Interactive Video (Up to 1 GB), and Live Lectures (NA).

Free Plan	
All prices are for 1 year subscription.	
Calender & Todo List	Unlimited
Instant Chat	NA
Interactive Video	Up to 1 GB
Live Lectures	NA



# Contact Us



The image shows a mobile application interface for a 'Contact Us' form. The header is purple with a logo on the left (a person holding a lantern over an open book) and a hamburger menu icon on the right. The form fields are white with rounded corners and are arranged vertically. The first field is for an email address, containing 'name@example.com\*'. The second field is for 'Your Name', containing 'Your Name\*'. The third field is for 'Your Email', containing 'subject\*'. The fourth field is for 'Your Message', containing 'Your Message Content'. At the bottom right of the form is a purple 'Send' button with a white paper plane icon.

name@example.com\*

Your Name

Your Name\*

Your Email

subject\*

Your Message

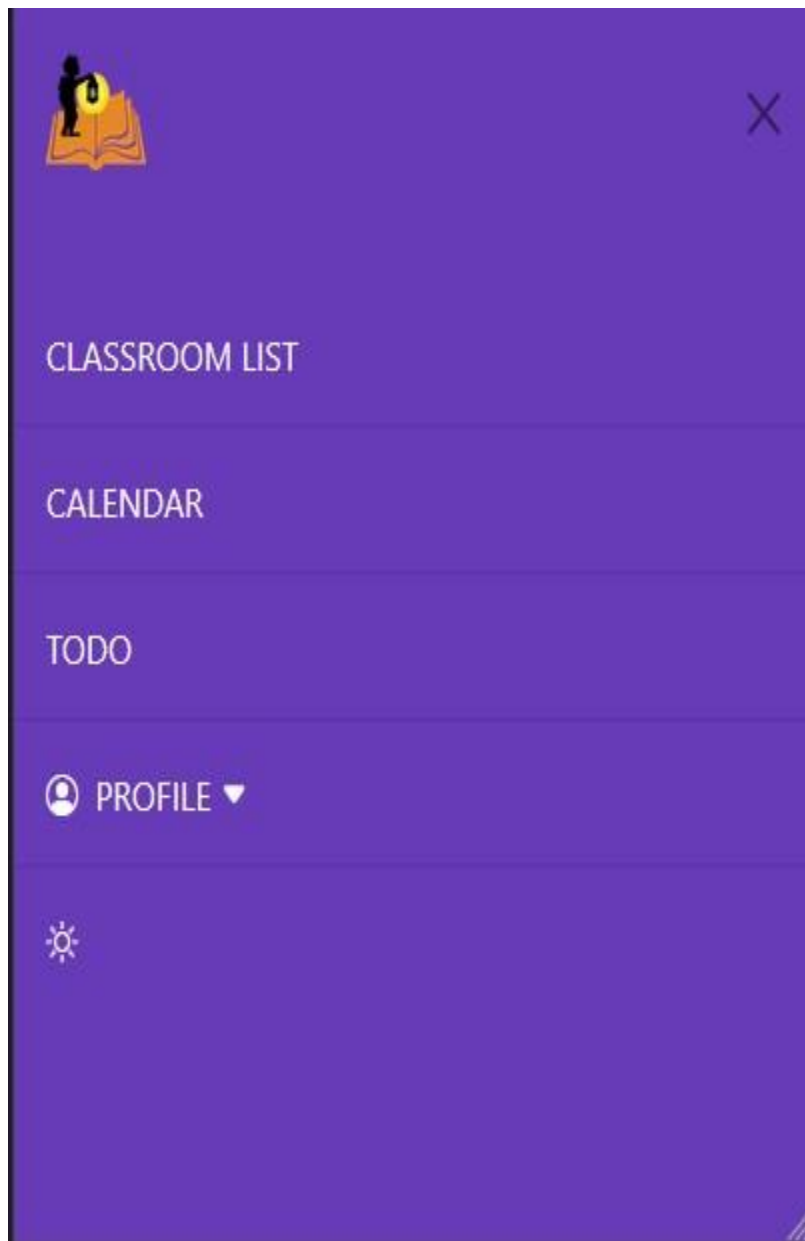
Your Message Content

Send

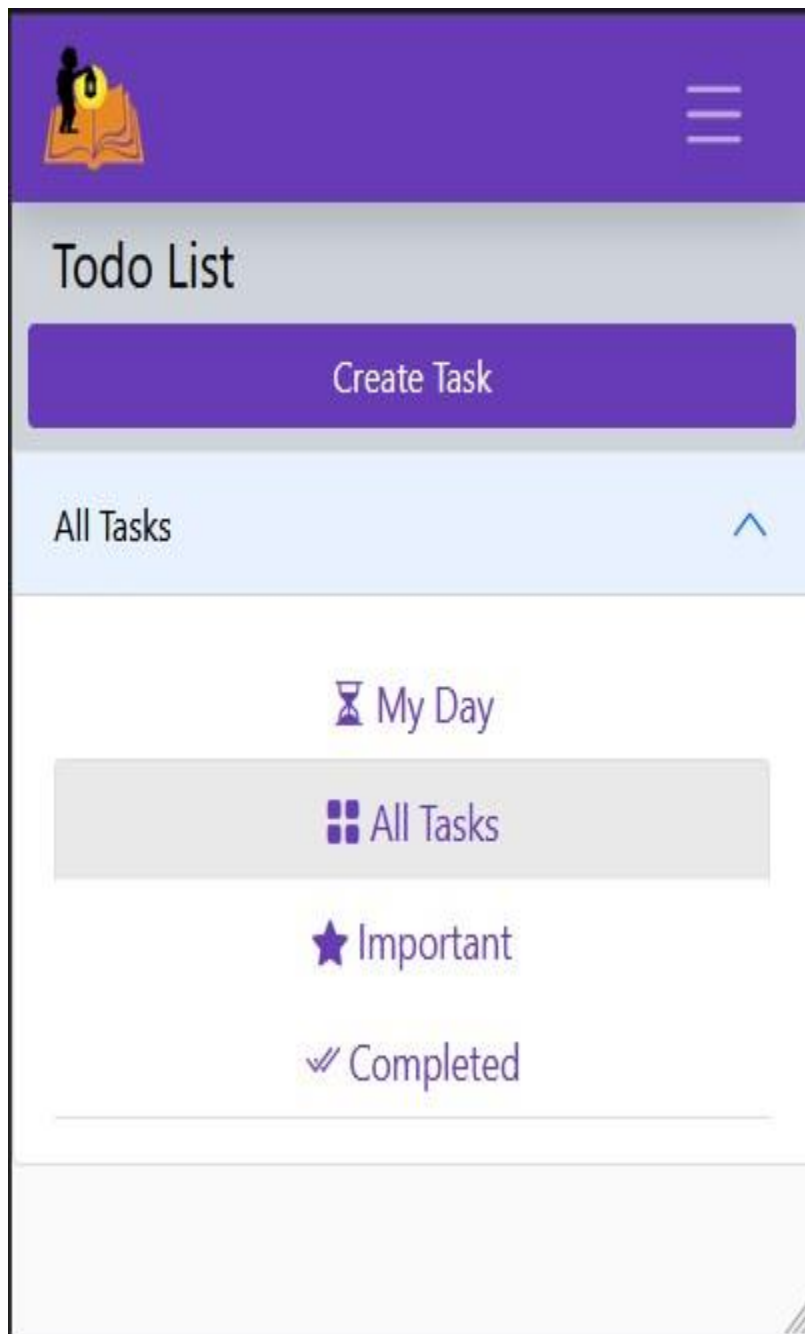
# Footer

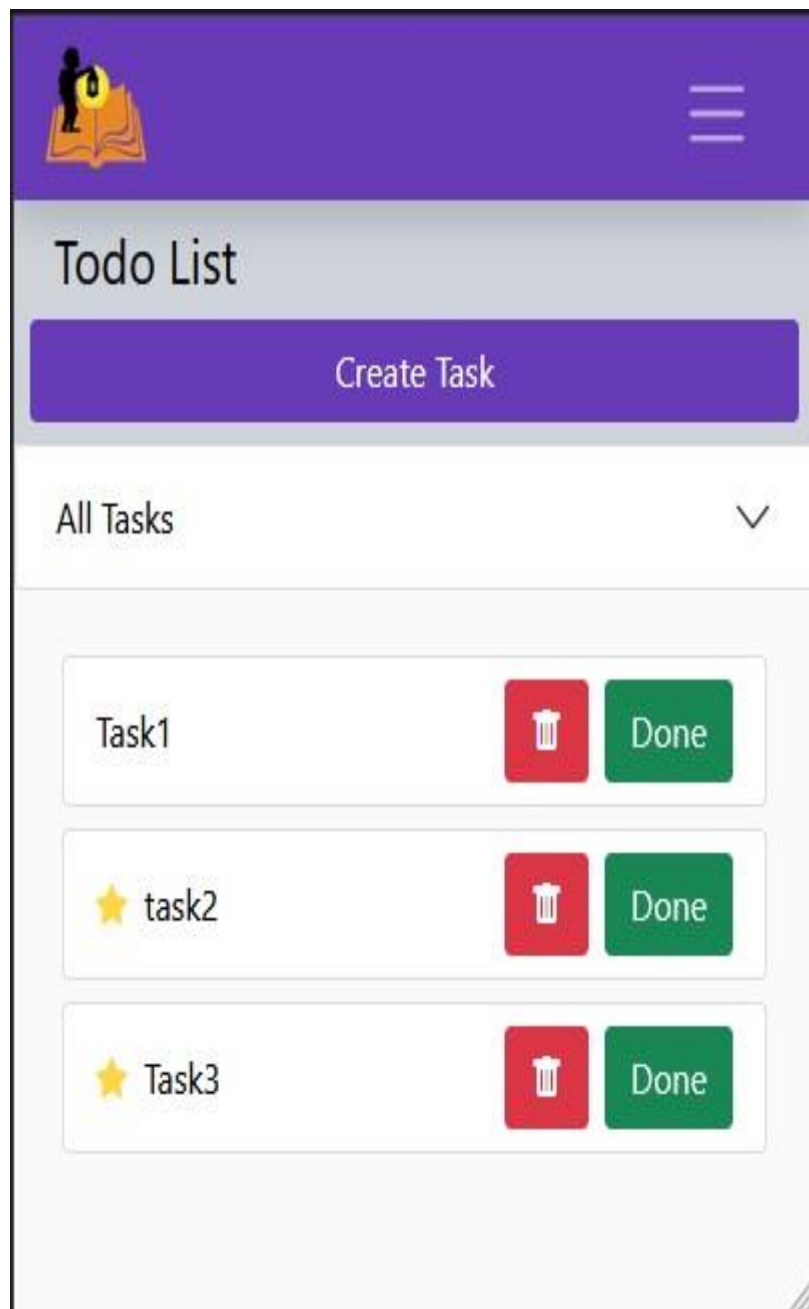


# Slide Bar



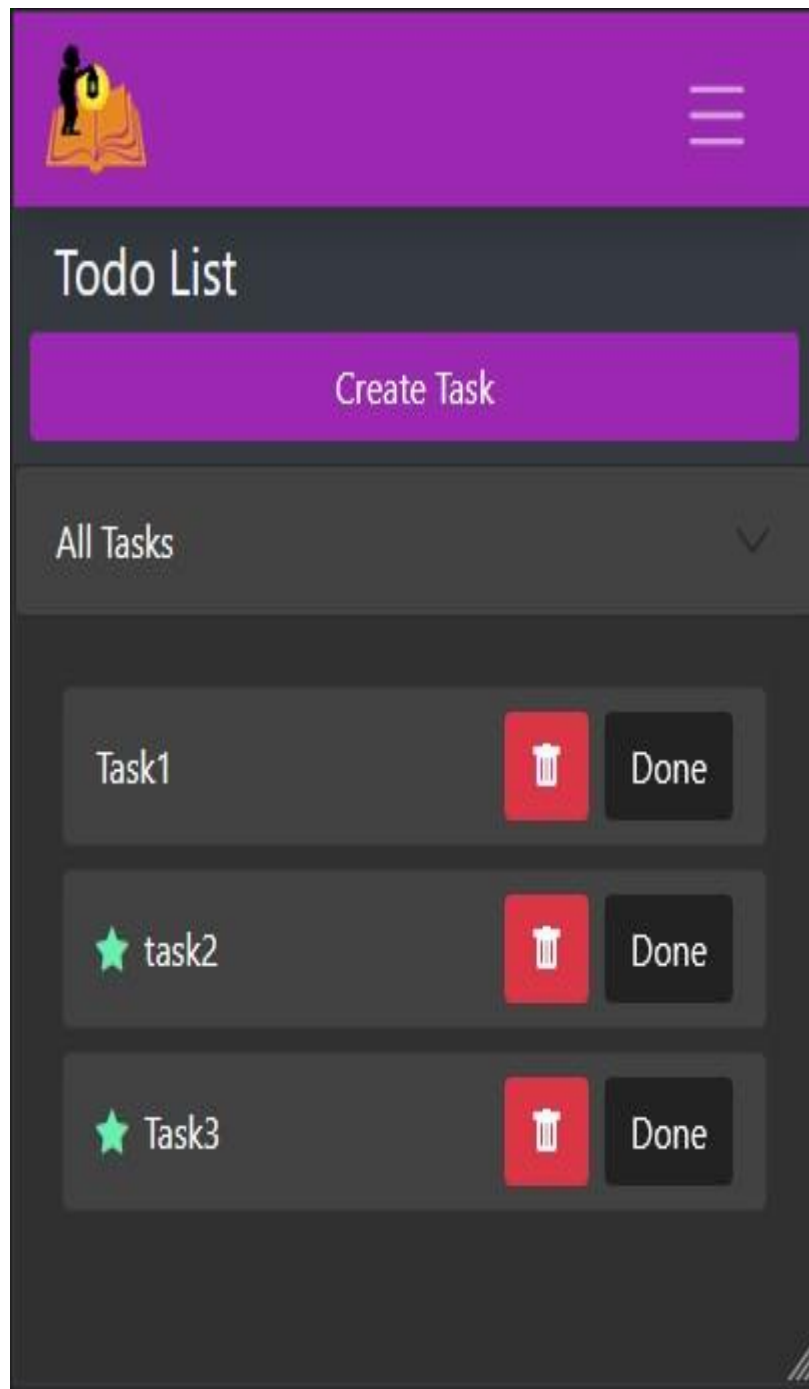
# To-Do








# Dark Mode





## Text Lesson View

 Class Title 

 Lesson Title  

### What Is a Database?

A database is an organized collection of structured information, or data, typically stored electronically in a computer system. A database is usually controlled by a [database management system \(DBMS\)](#). Together, the data and the DBMS, along with the applications that are associated with them, are referred to as a database system, often shortened to just database.

Data within the most common types of databases



## Text Lesson Editor

