A

PROJECT REPORT ON

REAL LEARNING APPLICATION

By

ASHISH KOTECHA (CE-064) (17CEUOS066)
HARSH MORADIYA (CE-072) (17CEUOG056)

B.Tech CE Semester-VI
Subject: System Design Practice

Guided by:
Prof. Pandav K. Patel
Assistant Professor
Dept. of Comp. Engg.

Faculty of Technology
Department of Computer Engineering
Dharmsinh Desai University

Faculty of Technology
Department of Computer Engineering
Dharmsinh Desai University

CERTIFICATE

This is to certify that the practical / term work carried out in the subject

of

System Design Practice and recorded in this journal is the

bonafide work of

Ashish Kotecha (CE-064) (17CEUOS066)
Harsh Moradiya (CE-072) (17CEUOG056)

of B.Tech semester **VI** in the branch of **Computer Engineering**

during the academic year **2019-2020**.

Prof. Pandav K. Patel
Assistant Professor,
Dept. of Computer Engg.,
Faculty of Technology
Dharmsinh Desai University, Nadiad

Dr. C. K. Bhensdadia,
Head,
Dept. of Computer Engg.,
Faculty of Technology
Dharmsinh Desai University, Nadiad

Table of Contents

1. Abstract

E-learning is a platform which provides a learning material of various domain Like Competitive Programming, HTML, CSS, JavaScript and many more. So we try to develop a Real Learning app in Flutter. Purpose of our application is to provide you a learning material at free of cost so that everyone can learn something new. After some analysis we found some of interesting and most popular course and we try to add most of them in our app. So as a result we develop Real Learning which can be used by any normal user if he/she have this app in his device. User can select any course from list and can learn it and check the knowledge for any topic. For admin we developed an admin-dashboard which will handle all the operation related to course, course subject and quiz questions. To avoid more version up gradation we try to create this app live on firebase so that any small modification in course will be reflect and use don't need to upgrade app itself. And we are trying to extend this app by adding more visual feature like augmented reality.

2. Introduction

2.1 Brief Introduction

Real-Learning is a mobile application which helpful to learn things online, means it's an e-learning system. Using this app you can learn given courses based on your choice and also most important is this app provide self-paced learning so you can learn as you want, whenever you want. Real-Learning is also provide quiz feature so after learning any topic thoroughly you can test your knowledge through test and if you want then you can start any particular topic again . As a backend support we have implemented a web app through which admin can add new courses and can manage courses. Admin can also manage its content and quiz question. And it's a live app so two way binding is there in our app. Means as soon as admin add or update any course, content or question it will directly reflect to user . We developed a front-end using flutter and dart language, because flutter supports a both iOS and android so we need to write for one app for both. And for a back-end we used django-python framework . We have try to develop a real life application.

2.2 Tools/Technologies used

Technologies:

1) Flutter (Dart)

2) Django Framework (Python)

3) Firebase Database

Tools:

Android Studio IDE

Sublime Text

3. Software Requirement Specifications

3.1 Types of User

1. Admin
2. Users (Learner)

3.2 System Function Requirement

R.1 User functionalities

R.1.1 View Course

Description: Users can view all course.

Input: User Command.

Output: All Course List**.**

R.1.2 View Course Content

Description: User can view Course Content to learn new couse

Input: Course Name

Output: Display All Tutorials.

R.1.3 Take Quiz

Description: User can take part in quiz topic wise.

Input: Select Topic

Output: All Questions related to topic.

R.2 Admin functionalities:

R.2.1 View Dashboard

Description: Admin can view dashboard for manage course, manage course-content and manage quiz.

Input: Admin login

Output: Display Dashboard.

R.2.2 Add Course

Description: Admin can add new course.

Input: Course details

Output: Added Successful/ Unsuccessful message.

R.2.3 Update Course

Description: Admin can update course details.

Input: Course Details

Output: Update status.

R.2.4 Delete Course

Description: Admin can delete any Course.

Input: Course name

Output: Deleted message.

R.2.5 Add Course Content

Description: Admin can add topic wise content for course.

Input: Content Details

Output: Added Successful/Unsuccessful message.

R.2.6 Update Course Content

Description: Admin can update topic wise content for course.

Input: Content Details

Output: Update status.

R.2.7 Delete Course Content

Description: Admin can delete topic wise content for course.

Input: Content Details

Output: Deleted message.

R.2.8 Add Quiz

Description: Admin can add topic wise quiz for course.

Input: Quiz Details

Output: Added Successful/Unsuccessful message.

Common Functionalities (User, Admin):

R.3 User Authentication-Sign Up

Input: User Details

Output: Data Stored Successfully

Description: User Enters Details Like User Id, Name, Password, Phone Number, E-Mail As Per Type Of User.

R.4 User Login

Input: User Credentials

Output: User Logged In Account/Error Message

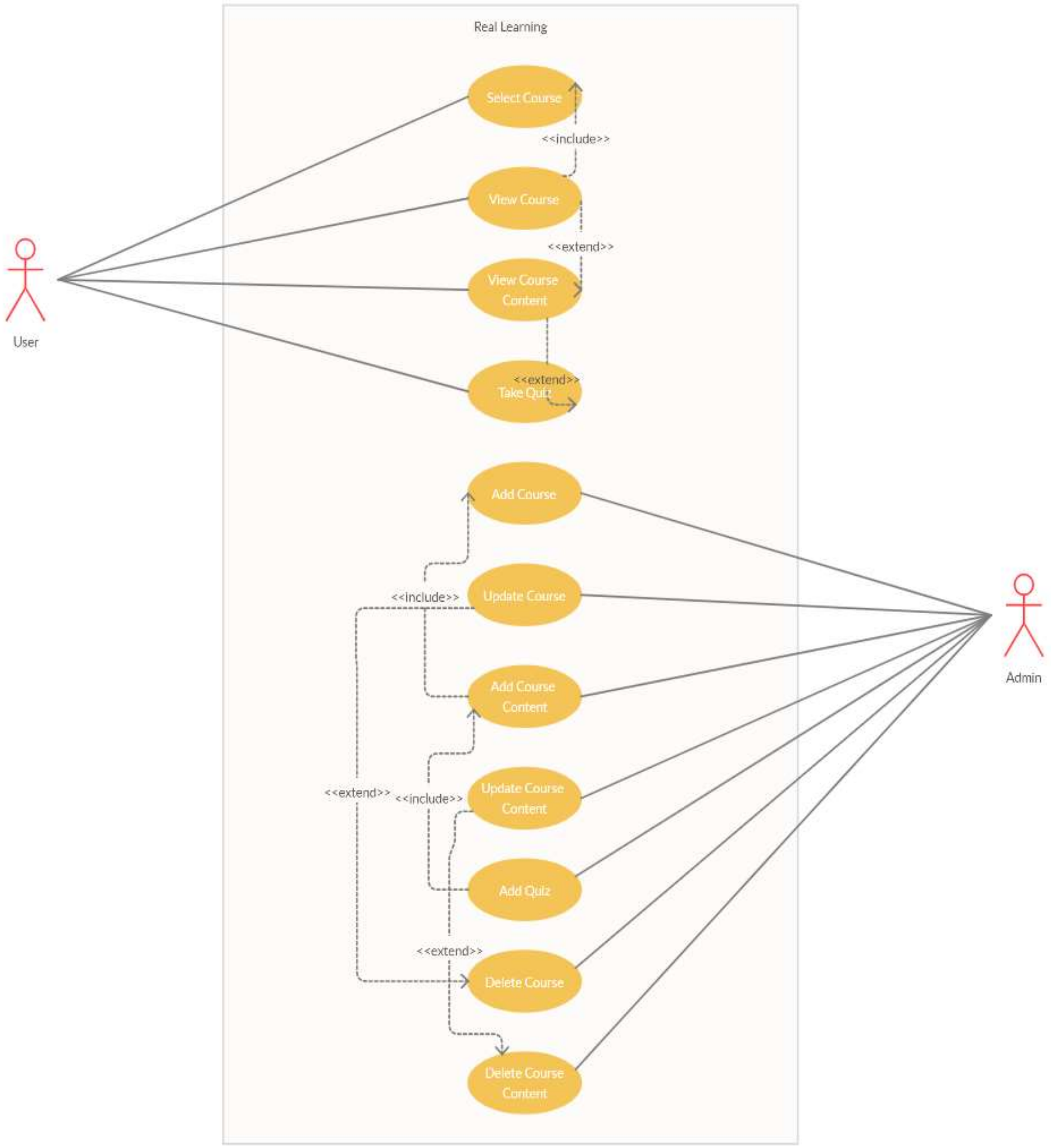
Description: User Enters The Username/Email And Password And Checks Into The Web Application By Validating In Database.

R.5 Log Out

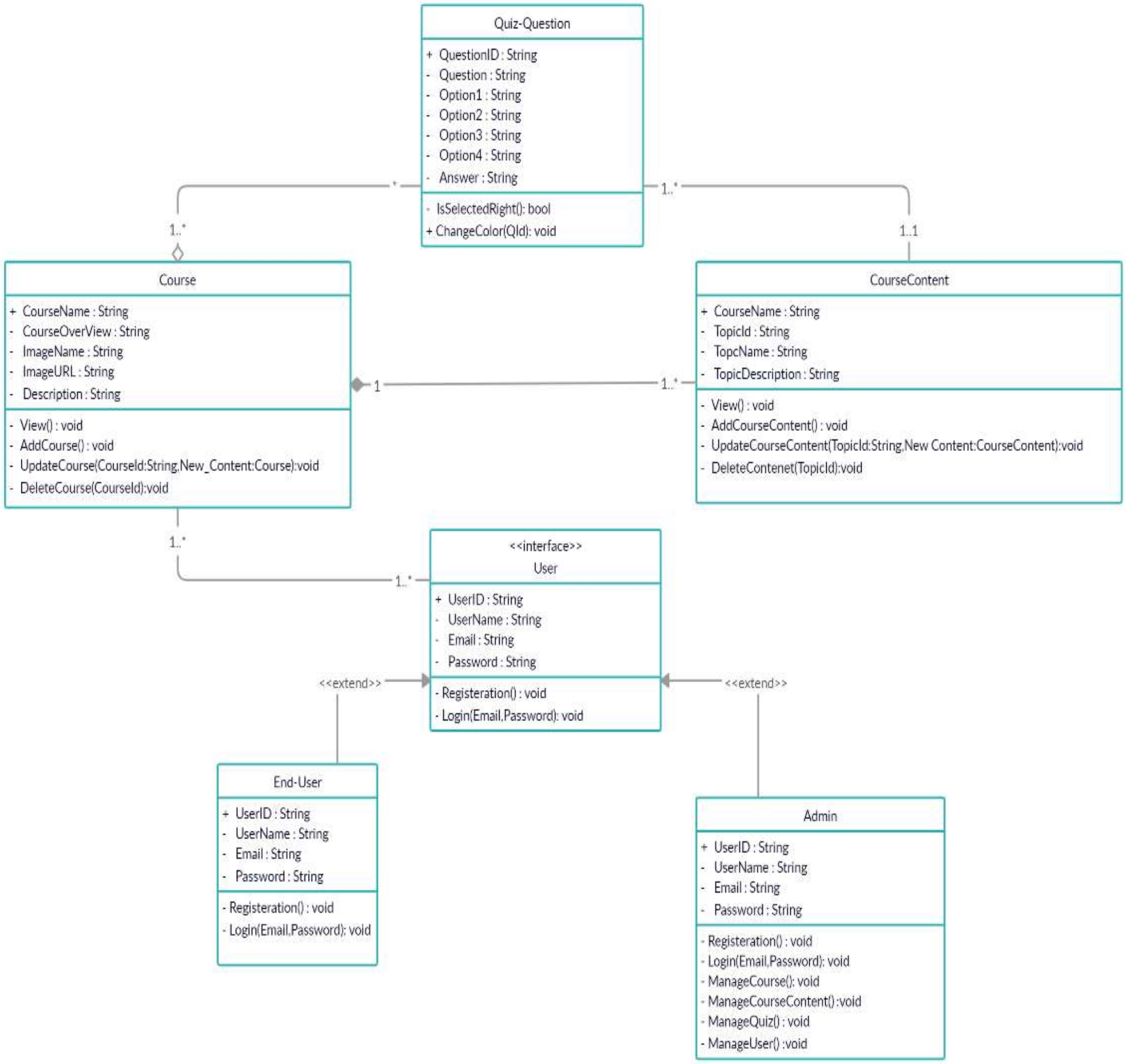
Description: User Logs Out Of The Website.

4. Designs

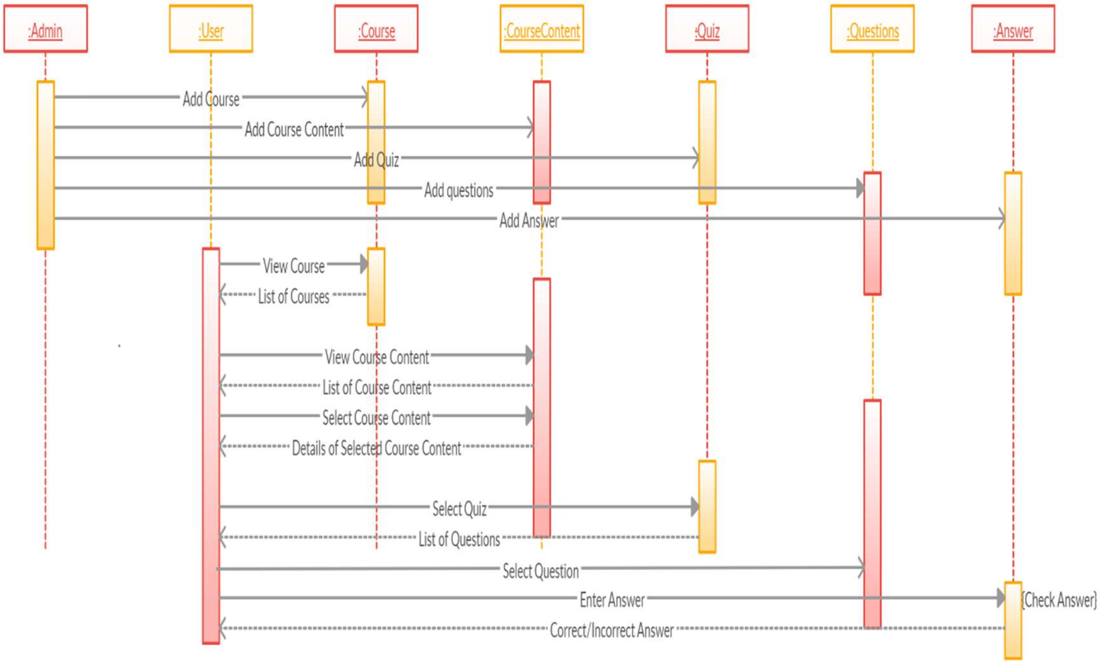
Use case Diagram:



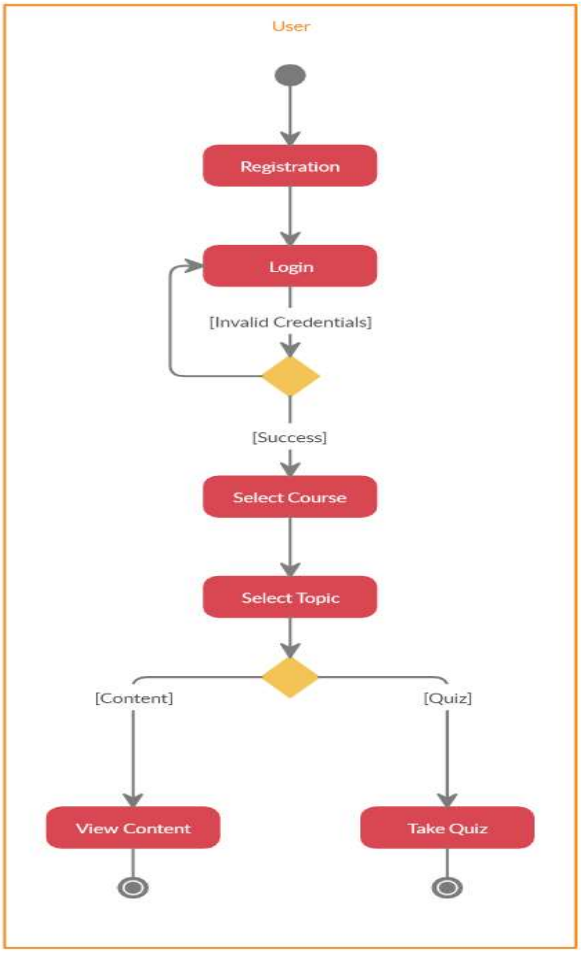
Class diagrams

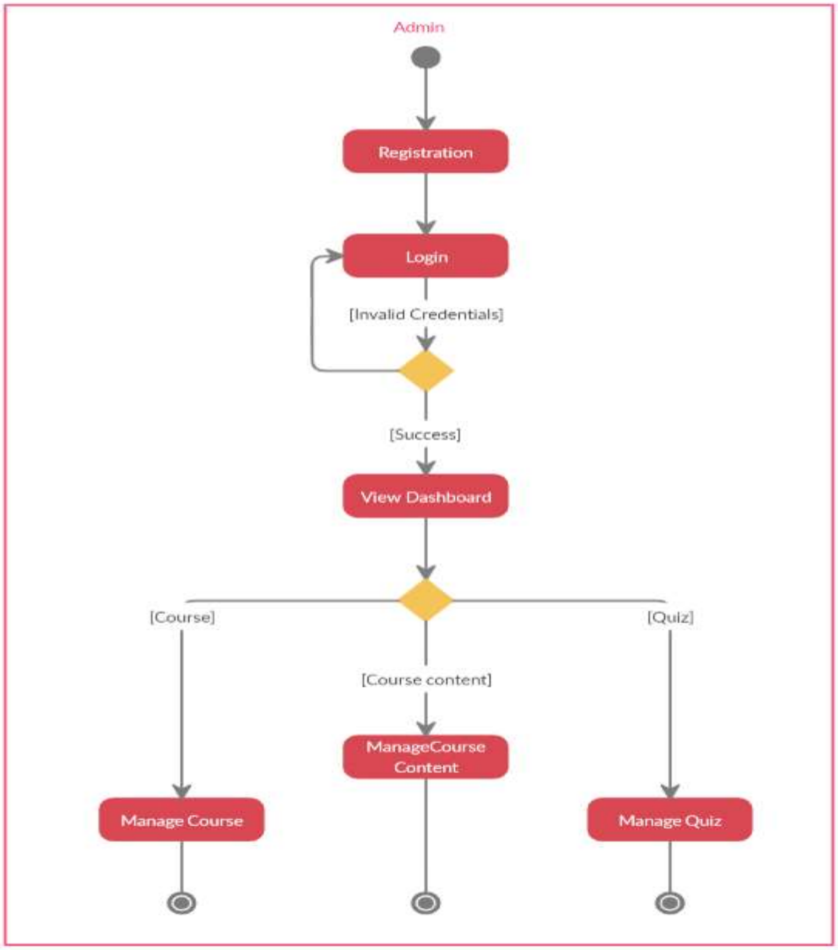


Sequence diagrams

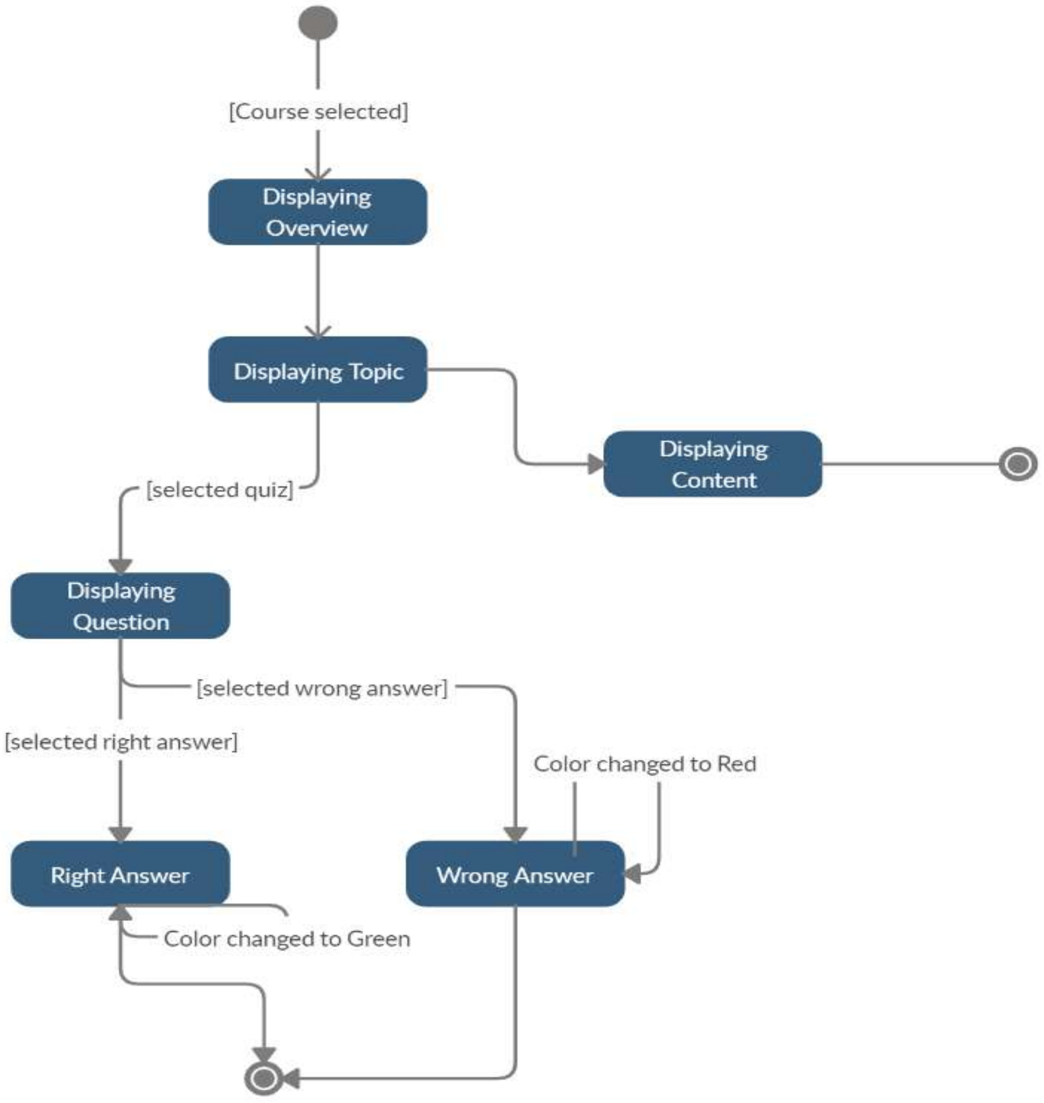


Activity diagrams:

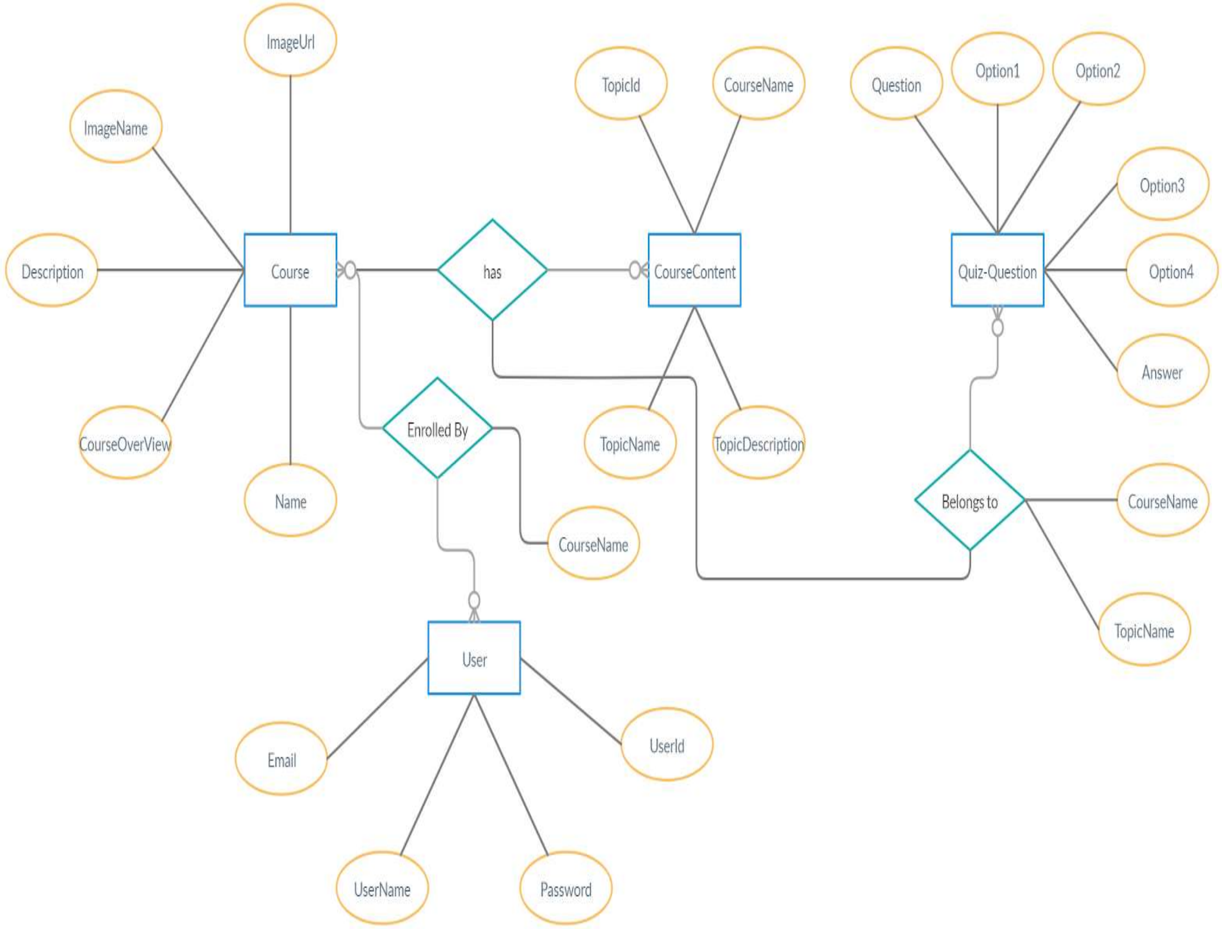



Admin
Registration
Login
[Invalid Credentials]
[Success]
View Dashboard
[Course]
[Quiz]
[Course content]
ManageCourse Content
Manage Course
Manage Quiz

State diagrams:



E-R Diagram:


ImageUrl
ImageName
TopicId
CourseName
Question
Option1
Option2
Option3
Description
Course
has
CourseContent
Quiz-Question
Option4
Answer
Enrolled By
CourseOverView
TopicName
TopicDescription
Name
Belongs to
CourseName
CourseName
TopicName
User
Email
UserId
UserName
Password

Data Dictionary:

USER							
SR.NO	NAME	DATA TYPE	WIDTH	REQUIRED	UNIQUE	PK/FK	REFERENCE TABLE DESCRIPTION
1	USERID	NUMERIC	20	YES	YES	PK	
2	PASSWORD	VARCHAR	20	YES	NO		
3	USERNAME	VARCHAR	20	YES	NO		
4	EMAIL	VARCHAR	20	YES	YES		

COURSE							
SR.NO	NAME	DATA TYPE	WIDTH	REQUIRED	UNIQUE	PK/FK	REFERENCE TABLE DESCRIPTION
1	ID	NUMERIC	20	YES	YES	PK	
2	NAME	VARCHAR	20	YES	NO		
3	DESCRIPTION	VARCHAR	20	YES	NO		
4	IMAGENAME	VARCHAR	20	YES	NO		
5	IMAGEURL	VARCHAR	20	YES	NO		
6	COURSE OVERVIEW	VARCHAR	20	YES	NO		

COURSE_CONTENT							
SR.NO	NAME	DATA TYPE	WIDTH	REQUIRED	UNIQUE	PK/ FK	REFERENCE TABLE DESCRIPTION
1	ID	NUMERIC	20	YES	YES	PK	
2	COURSENAME	VARCHAR	20	YES	NO	FK	COURSE
3	TOPICNAME	VARCHAR	20	YES	NO		
4	TOPIC DESCRIPTION	VARCHAR	20	YES	NO		

QUIZ							
SR.NO	NAME	DATA TYPE	WIDTH	REQUIRED	UNIQUE	PK/ FK	REFERENCE TABLE DESCRIPTION
1	ID	NUMERIC	20	YES	YES	PK	
2	TOPICNAME	NUMERIC	20	YES	YES	FK	COURSE_ CONTENT
2	QUESTION	VARCHAR	20	YES	NO		
3	OPTION1	VARCHAR	20	YES	NO		
4	OPTION2	VARCHAR	20	YES	NO		
5	OPTION3	VARCHAR	20	YES	NO		
6	OPTION4	VARCHAR	20	YES	NO		
7	ANSWER	VARCHAR	20	YES	NO		

5. Implementation Details

1. Modules

- **User Module:**

User module is basically manage all the activity of users. In this module user can register and login by giving some basic details.

User can view all course. User can see all the topics of any course and go through any particular topic of any course. Also user can take part in quiz for any particular topic of quiz.

- **Admin Module:**

This module handles all the activity of the Admin. Admin can add course, course content and quiz for all topics. Admin can also update the details of any course content or course. Also admin can delete the course content and whole course.

2. Function Prototype

Users:

- To get all course

  Future GetAllCourse()

- TO get specific course

  Stateless widget CourseDetail()

- TO get all sub topic

  Future getallsubTopic()

- To get quiz

  Stateful widget Quiz()

Admin:

- Login:

```
def login(request)
```

- Add Course:

```
def courseupload(request)
```

- Add Course-Content:

```
def uploadcoursecontent(request)
```

- View Course:

```
def viewselectedcourse(request)
```

- View Course-Content:

```
def viewselectedtopic(request):
```

- Update Course:

```
def courseupdate(request)
```

- Update course content:

```
def coursecontentupdate(request)
```

- Add Question:

```
def addQuestion(request)
```

- Delete Course:

```
def deletecourse(request)
```

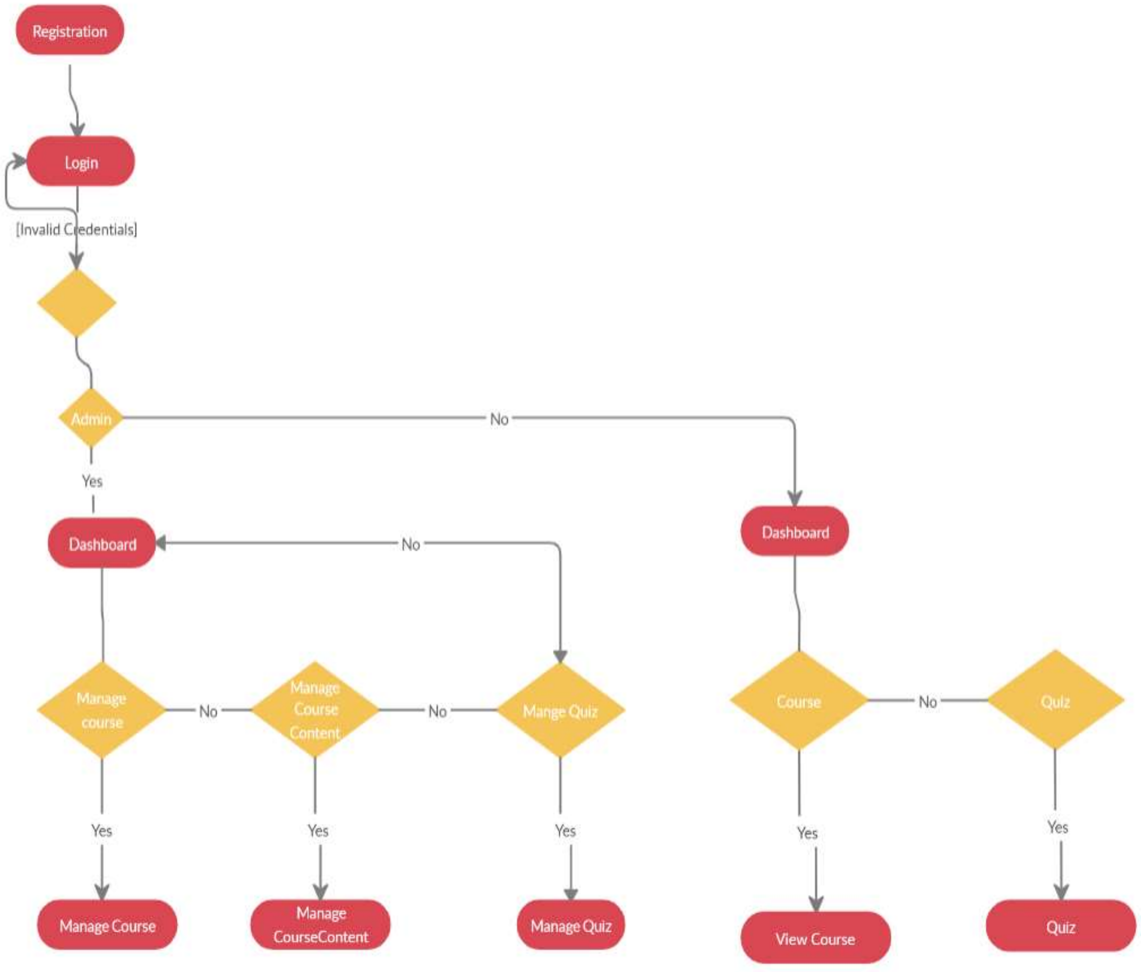
- Delete Course Content:

```
def addQuestion(request)
```

- Logout:

```
def logout(request)
```

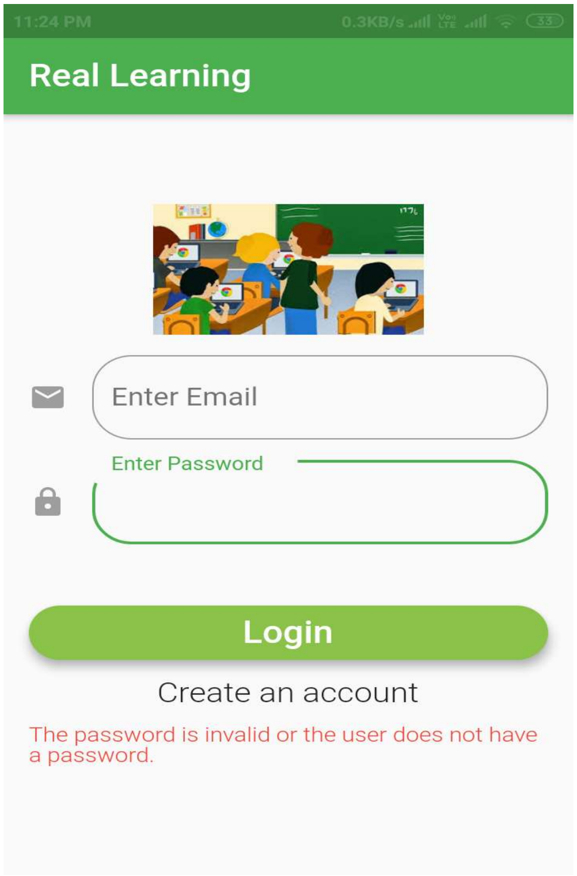
3. Flow Chart


Registration
Login
[Invalid Credentials]
Admin
No
Yes
Dashboard
No
Dashboard
Manage course
No
Manage Course Content
No
Mange Quiz
Course
No
Quiz
Yes
Yes
Yes
Yes
Yes
Manage Course
Manage CourseContent
Manage Quiz
View Course
Quiz

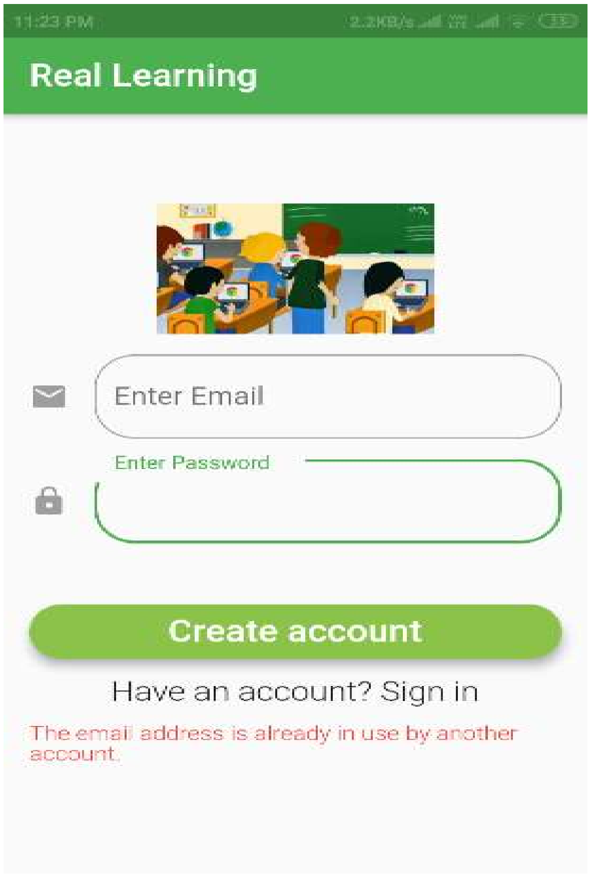
6. Testing

We performed black box testing.

If user does not provide password or wrong details system will throw error.

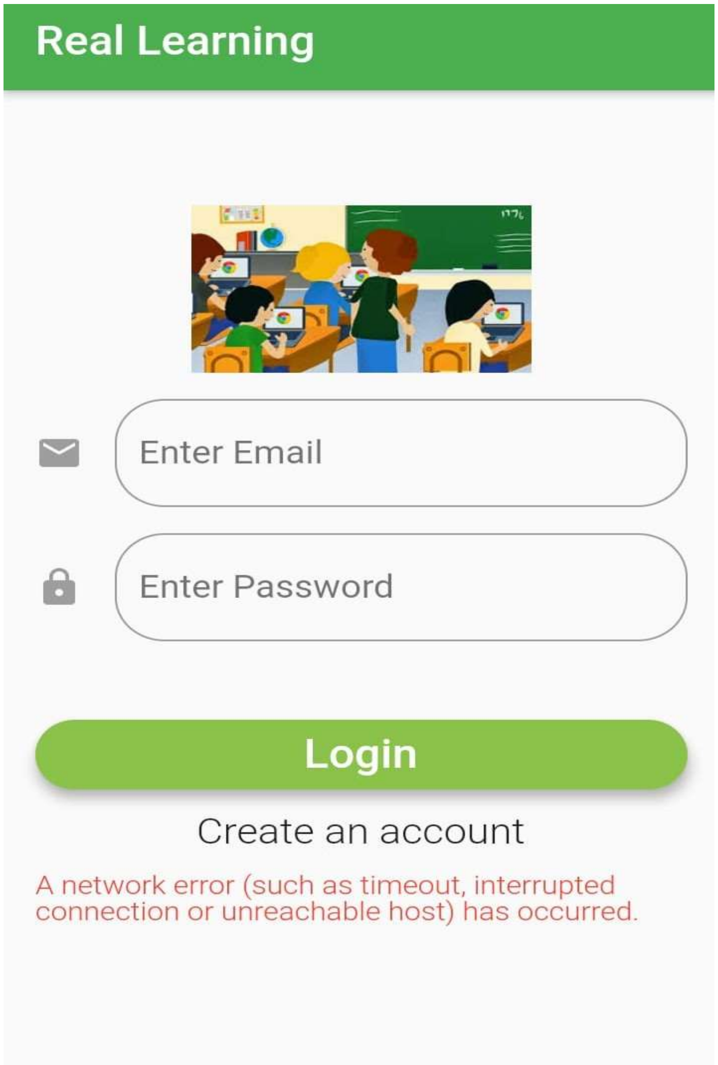


The password is invalid or the user does not have a password.



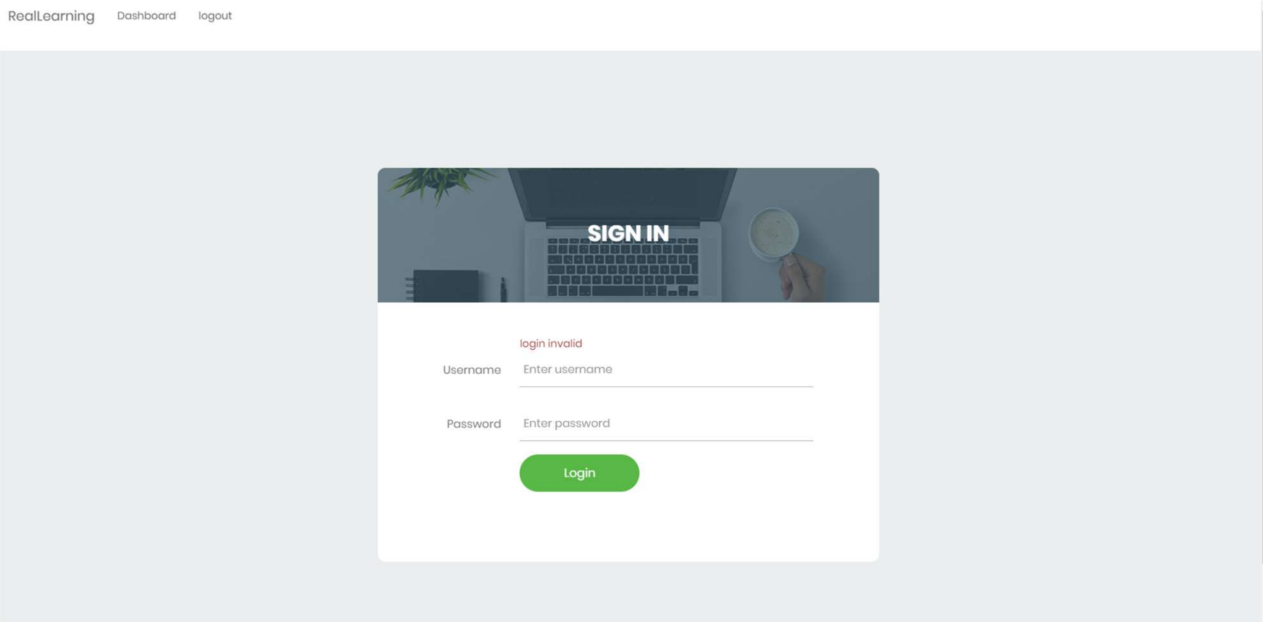
The email address is already in use by another account.

If users tries to access application without data connection it will show “Network Error”.



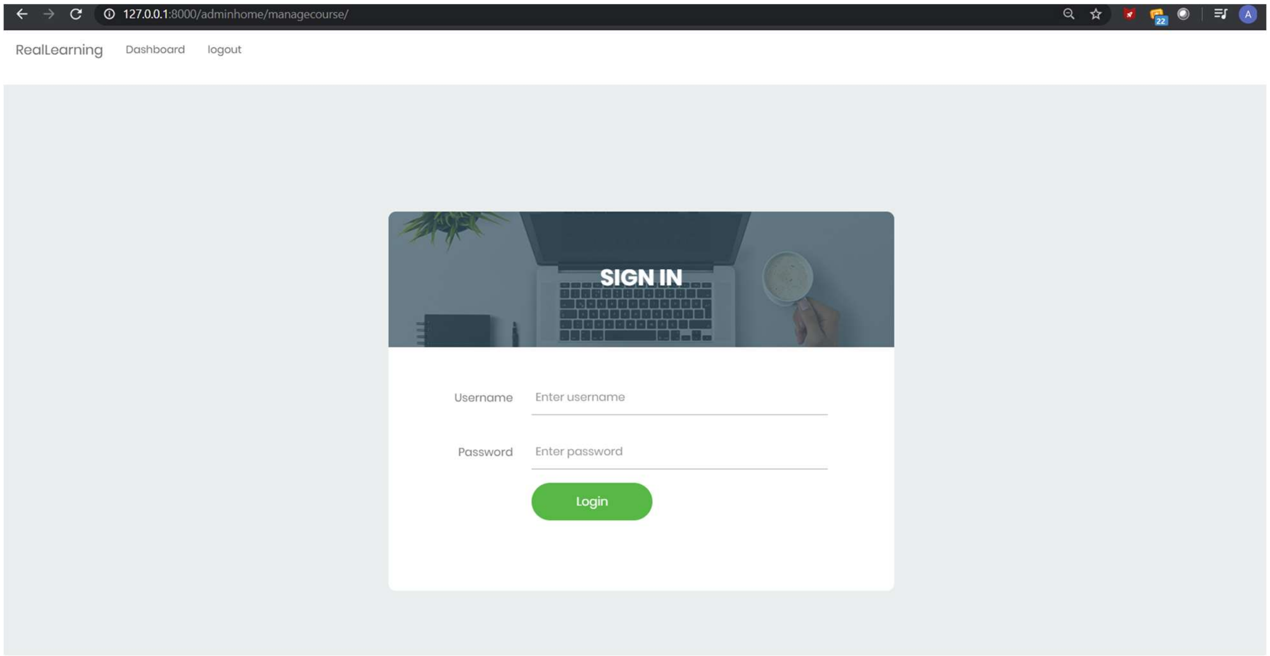
A network error (such as timeout, interrupted connection or unreachable host) has occurred.

If admin enters wrong credentials then it show error message “login invalid”.



login invalid

If admin attempt to access any page without login then it will redirect to login page.


127.0.0.1:8000/adminhome/managecourse/

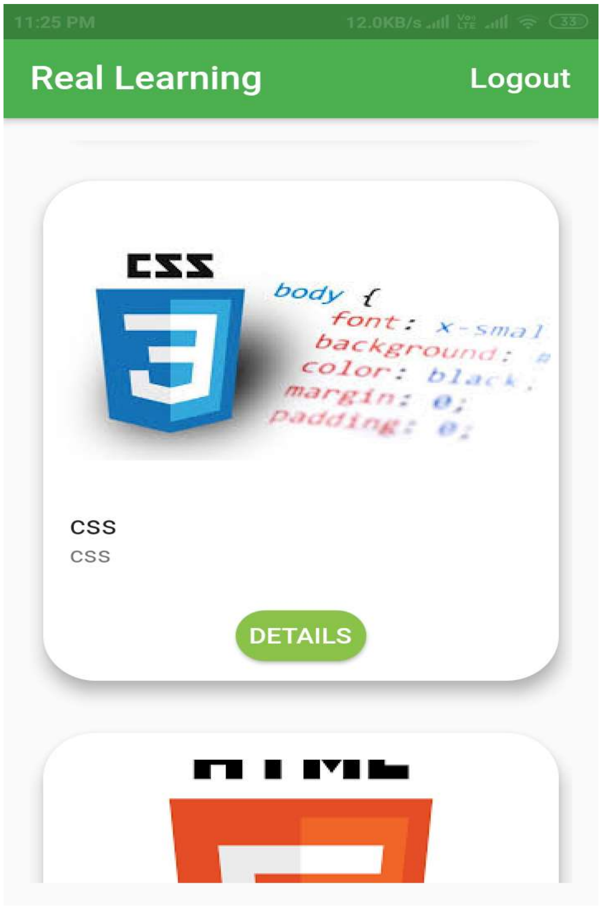
TestCases:

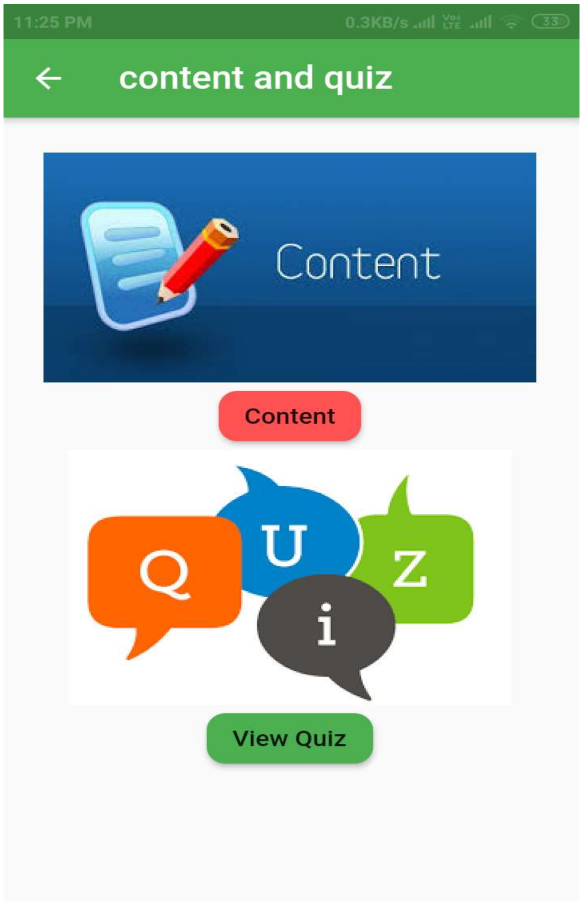
No	Test Scenarios	Expected Result	Actual Result	Pass/Fail
1.	User enters wrong credentilas	EmailId/ Password invalid	EmailId/ Password invalid	Pass
2.	User enters wrong format of an emailid	EmailId is invalid	EmailId is invalid	Pass
3.	User enters emailid that already taken by another user	EmailId is already in use by another user	EmailId is already in use by another user	Pass
4.	Admin will try to access page without login	Redirect to Login page	Redirect to login page.	Pass
5.	Admin add formatted data while adding course data	It should be reflect in users application	It should be reflect in users application	Pass
6.	User tries to access application without data connection	Network error	Network error	Pass

7. Screenshot

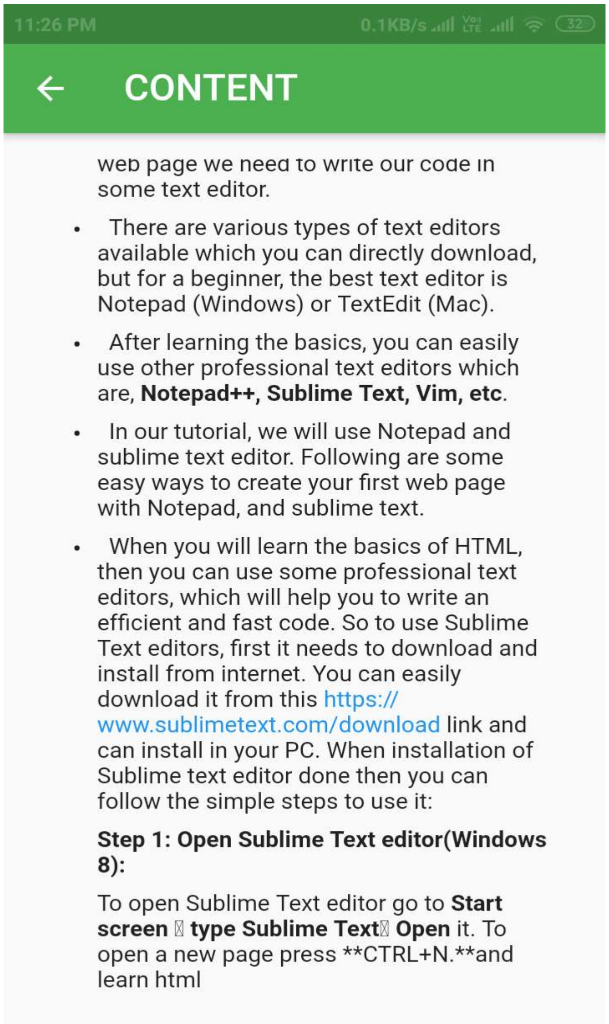
Users:

After Login user can see all courses:





After Clicking on content or quiz user can see below screen. If answer of question is wrong then red color is displayed.



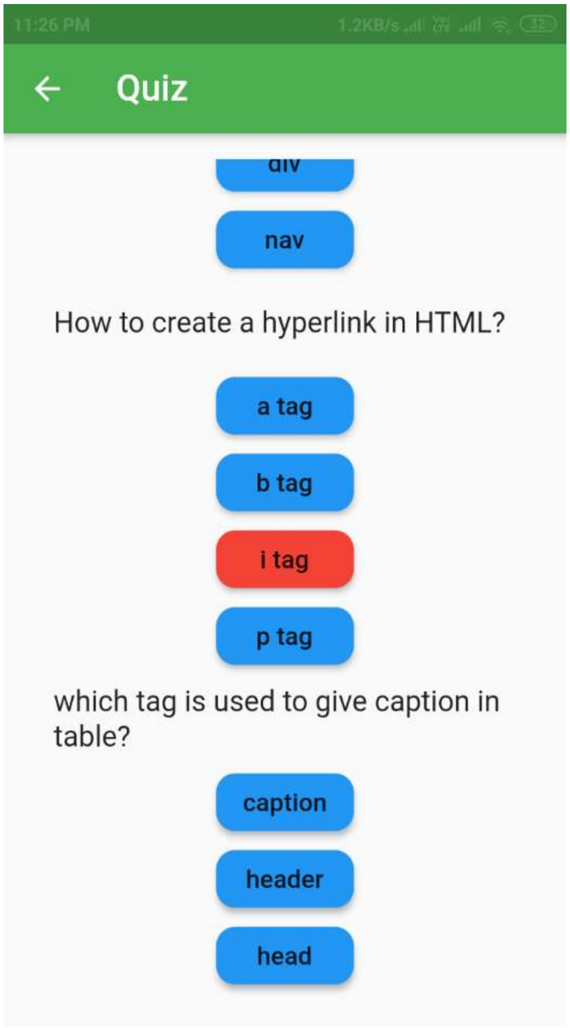
CONTENT

web page we need to write our code in some text editor.

- There are various types of text editors available which you can directly download, but for a beginner, the best text editor is Notepad (Windows) or TextEdit (Mac).
- After learning the basics, you can easily use other professional text editors which are, **Notepad++, Sublime Text, Vim, etc**.
- In our tutorial, we will use Notepad and sublime text editor. Following are some easy ways to create your first web page with Notepad, and sublime text.
- When you will learn the basics of HTML, then you can use some professional text editors, which will help you to write an efficient and fast code. So to use Sublime Text editors, first it needs to download and install from internet. You can easily download it from this https://www.sublimetext.com/download link and can install in your PC. When installation of Sublime text editor done then you can follow the simple steps to use it:

Step 1: Open Sublime Text editor(Windows 8):

To open Sublime Text editor go to **Start screen type Sublime Text Open** it. To open a new page press **CTRL+N.**and learn html

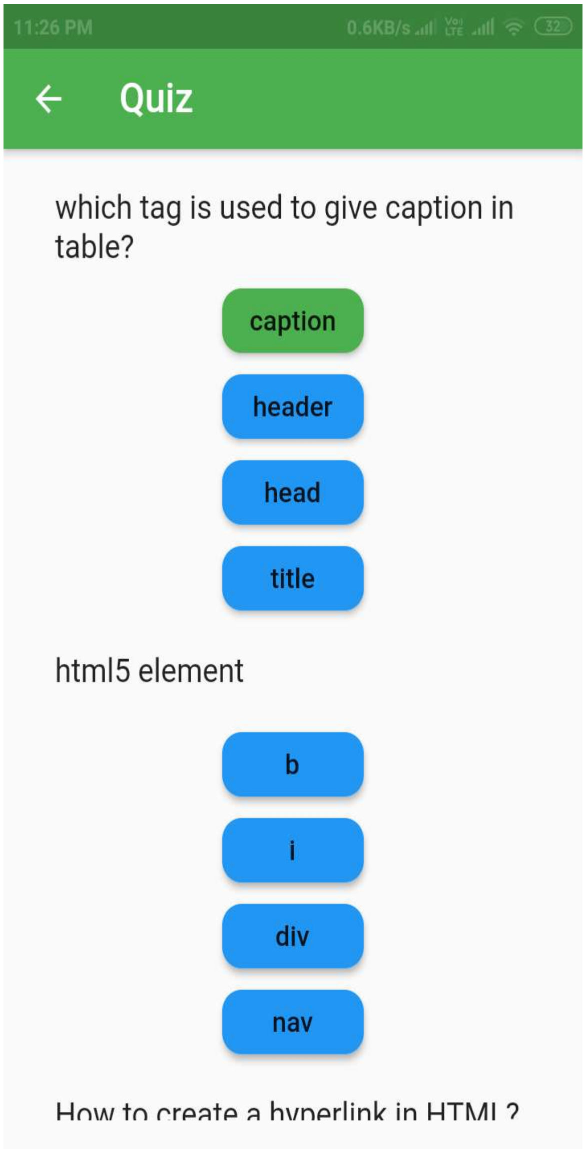


Quiz

How to create a hyperlink in HTML?

which tag is used to give caption in table?

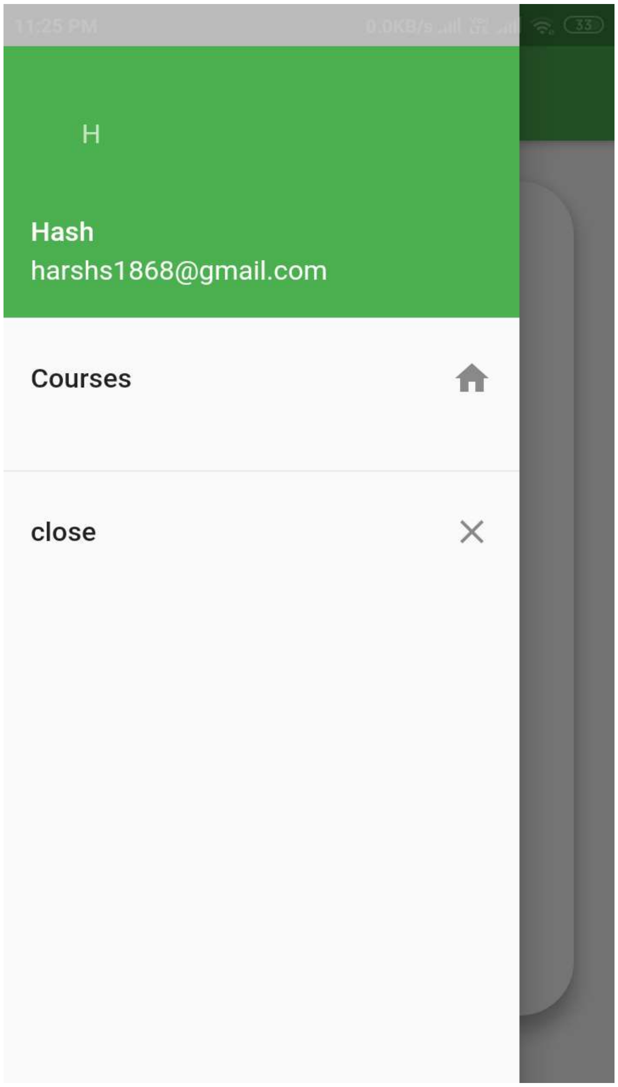
If answer of question is correct then green color is displayed.



which tag is used to give caption in table?

html5 element

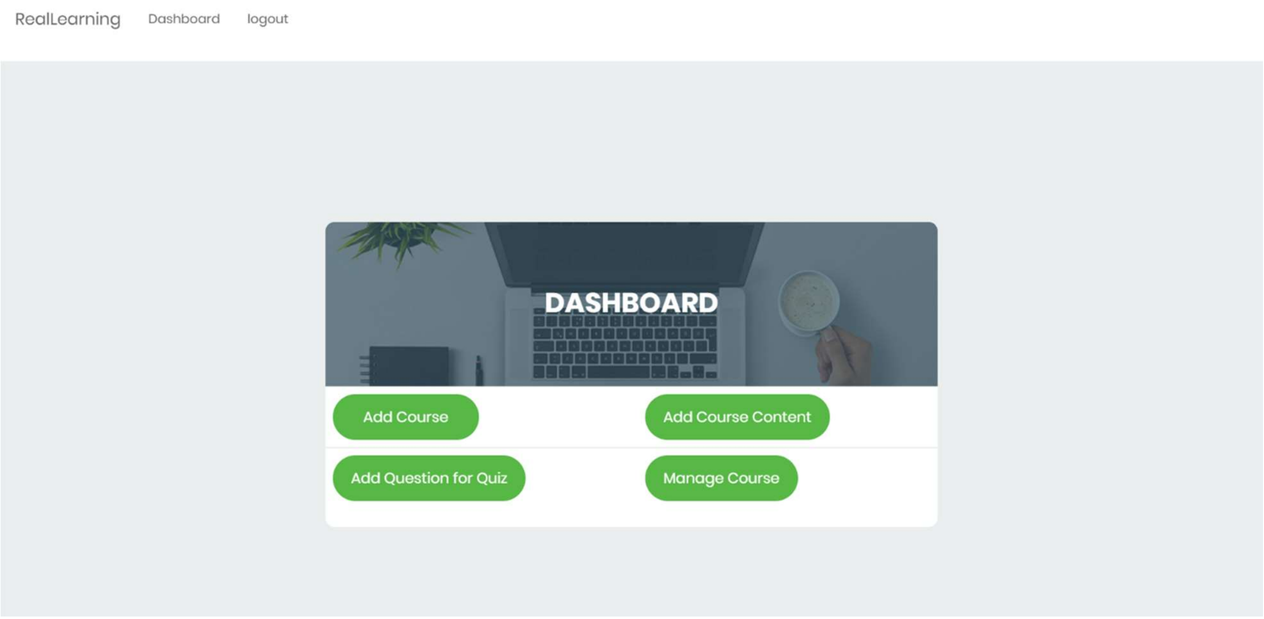
How to create a hyperlink in HTML ?



Hash
harshs1868@gmail.com

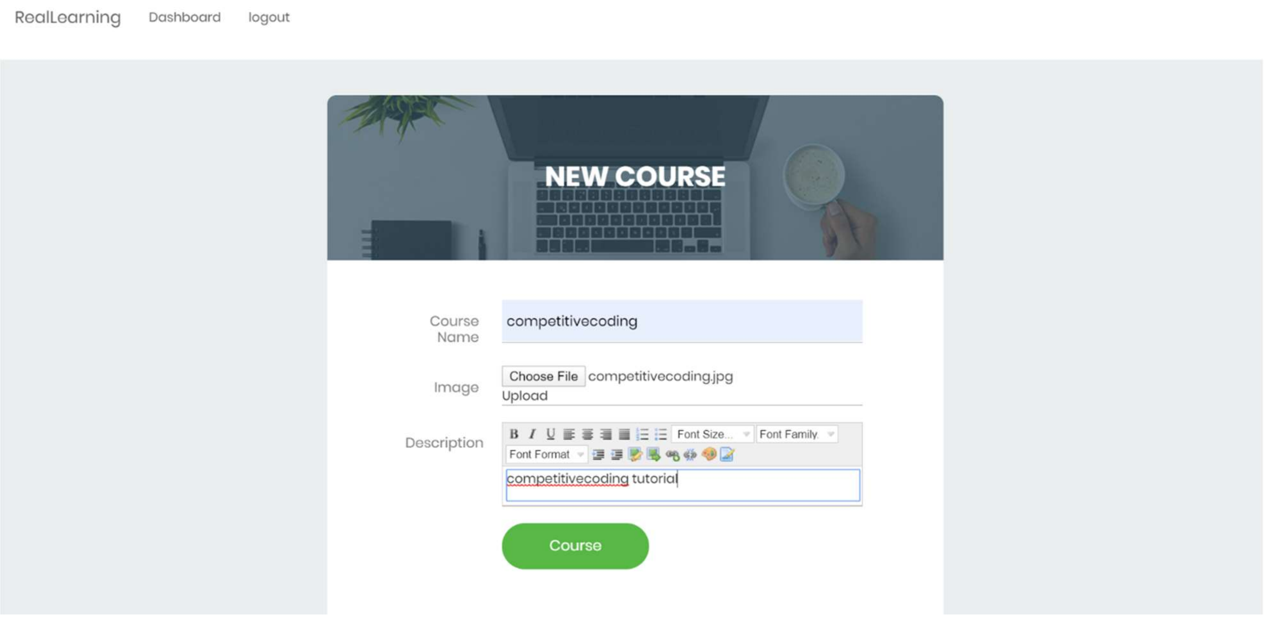
Admin:

1. Admin Dashboard:



After login as Admin can see dashboard.

2. Add New Course:

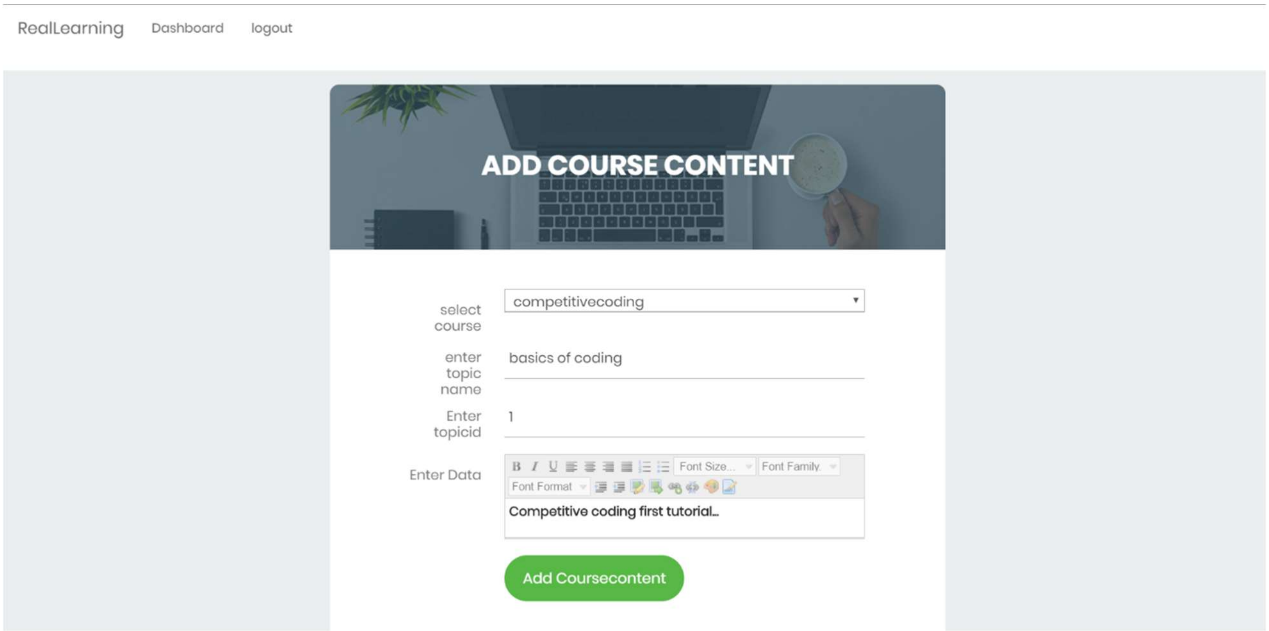


Course Name: competitivecoding

Image: competitivecoding.jpg

Description: competitivecoding tutorial

3. Add New Course-Content:



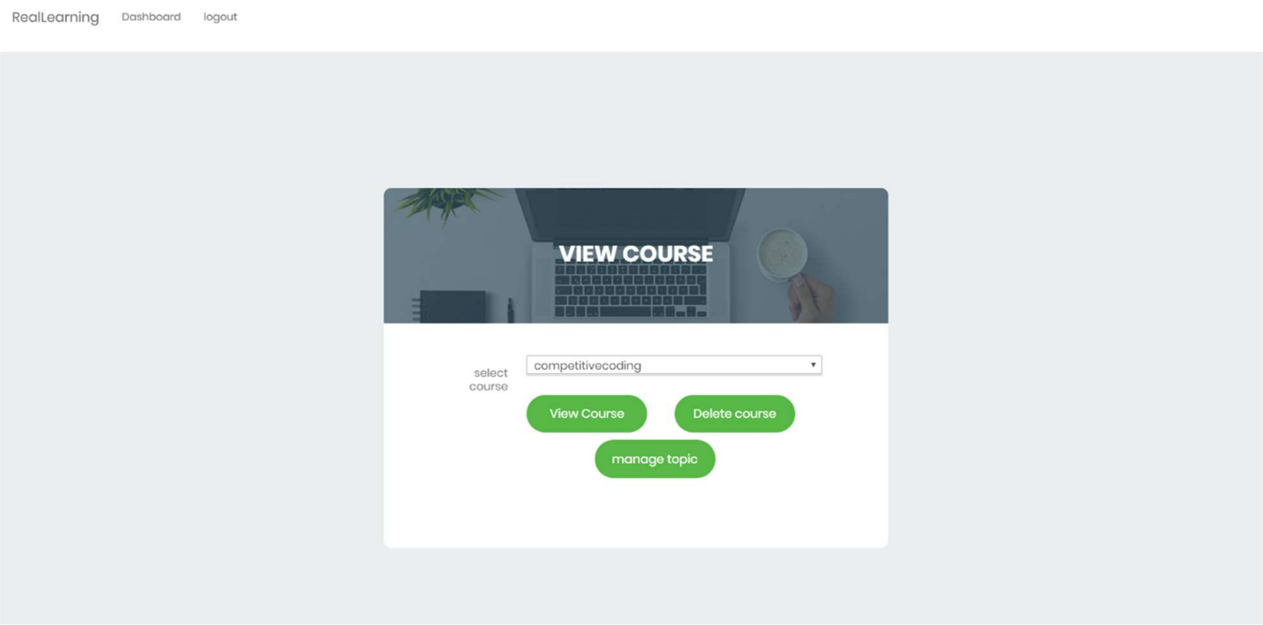
select course: competitivecoding

enter topic name: basics of coding

Enter topicid: 1

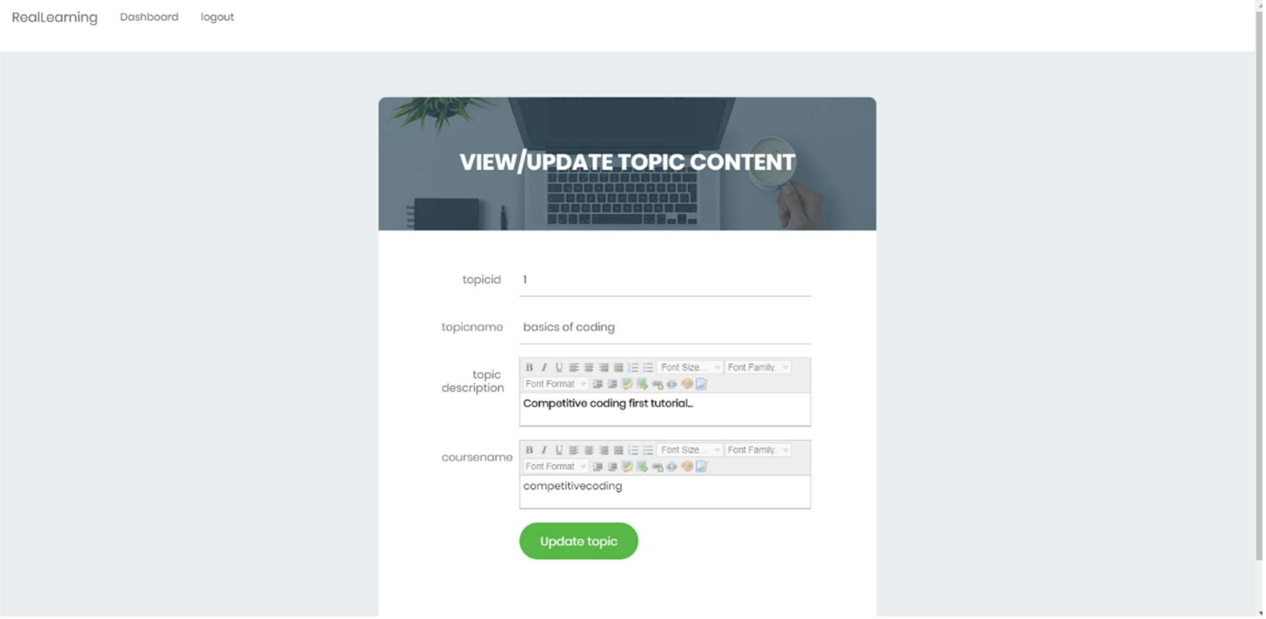
Enter Data: Competitive coding first tutorial..

4. Manage Course:



select course: competitivecoding

5. Update Course-Content:



VIEW/UPDATE TOPIC CONTENT

topicid: 1

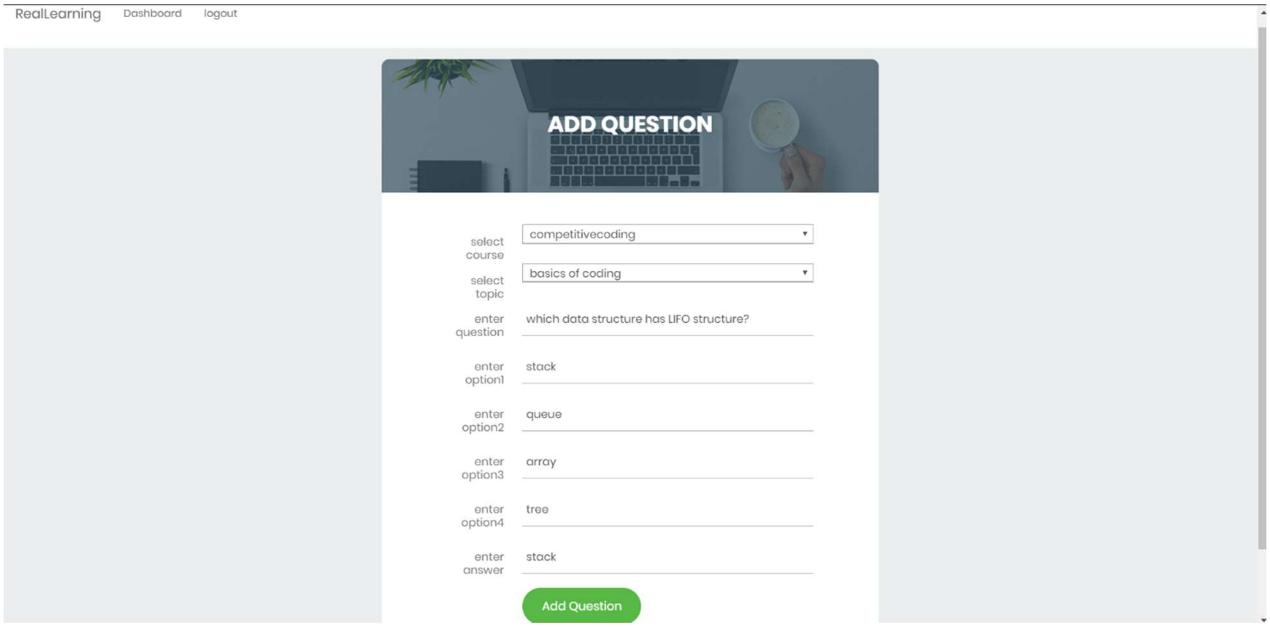
topicname: basics of coding

topic description: Competitive coding first tutorial..

coursename: competitivecoding

Update topic

6. Add Question.



ADD QUESTION

select course: competitivecoding

select topic: basics of coding

enter question: which data structure has LIFO structure?

enter option1: stack

enter option2: queue

enter option3: array

enter option4: tree

enter answer: stack

Add Question

8. Limitation and Future Extension

Limitations

1. There is no enrollment feature for any course.
2. Currently system can not recommend any course based on learner's past course.
3. Our system is not tracking user so that user can know how much course is completed and how many module he has finished.
4. Our system not providing video content ,currently it supports only image and text format. You can add your rich text.

Future Extension

1. We will try to add Augmented Reality for better user experience.
2. We will provide subscription feature for course.
3. We will provide video content for any particular topic.
4. We will also track the record of learner to provide suggestion for best course.
5. We will add recommendation system for user.

9. Conclusion

The functionality of this system is developed after understanding whole system flow and all module and it is as per requirement.

The functionality which were successfully implemented :

1. Registration/Login

2. View Course

3. View Course Content

4. View Quiz

5. Manage Course

6. Manage Course Content

7. Manage Quiz

10. Bibliography

Reference Links:

https://flutter.dev/docs

https://docs.djangoproject.com/en/3.0/

https://stackoverflow.com/