

# **Software Development Practice SWE573**

## **Final Project Report**

### **EduMe (E-Learning Platform)**

Instructor

Asst. Prof. Suzan Uskudarli

Student

Bilal Berat Postalcioglu (2021719132)

Git Repository: <https://github.com/bpostalci/bounswe573-2022/tree/master/edume>

Git Tag: v0.9

Deployment URI: <http://ec2-54-161-242-148.compute-1.amazonaws.com/>

<b>Project Overview</b>	<b>3</b>
<b>Software Requirements Specification</b>	<b>3</b>
Functional Requirements	3
Use Case Diagram	6
Non-Functional Requirements	8
<b>Mockups</b>	<b>8</b>
<b>Software Design Artifacts</b>	<b>19</b>
<b>Status of the project</b>	<b>22</b>
Completed Functional Requirements:	22
<b>Status of Deployment</b>	<b>23</b>
<b>System Manual</b>	<b>24</b>
<b>User Manual</b>	<b>25</b>
<b>Test Results</b>	<b>28</b>
<b>Project Repository</b>	<b>28</b>

## **Project Overview**

# **Software Requirements Specification**

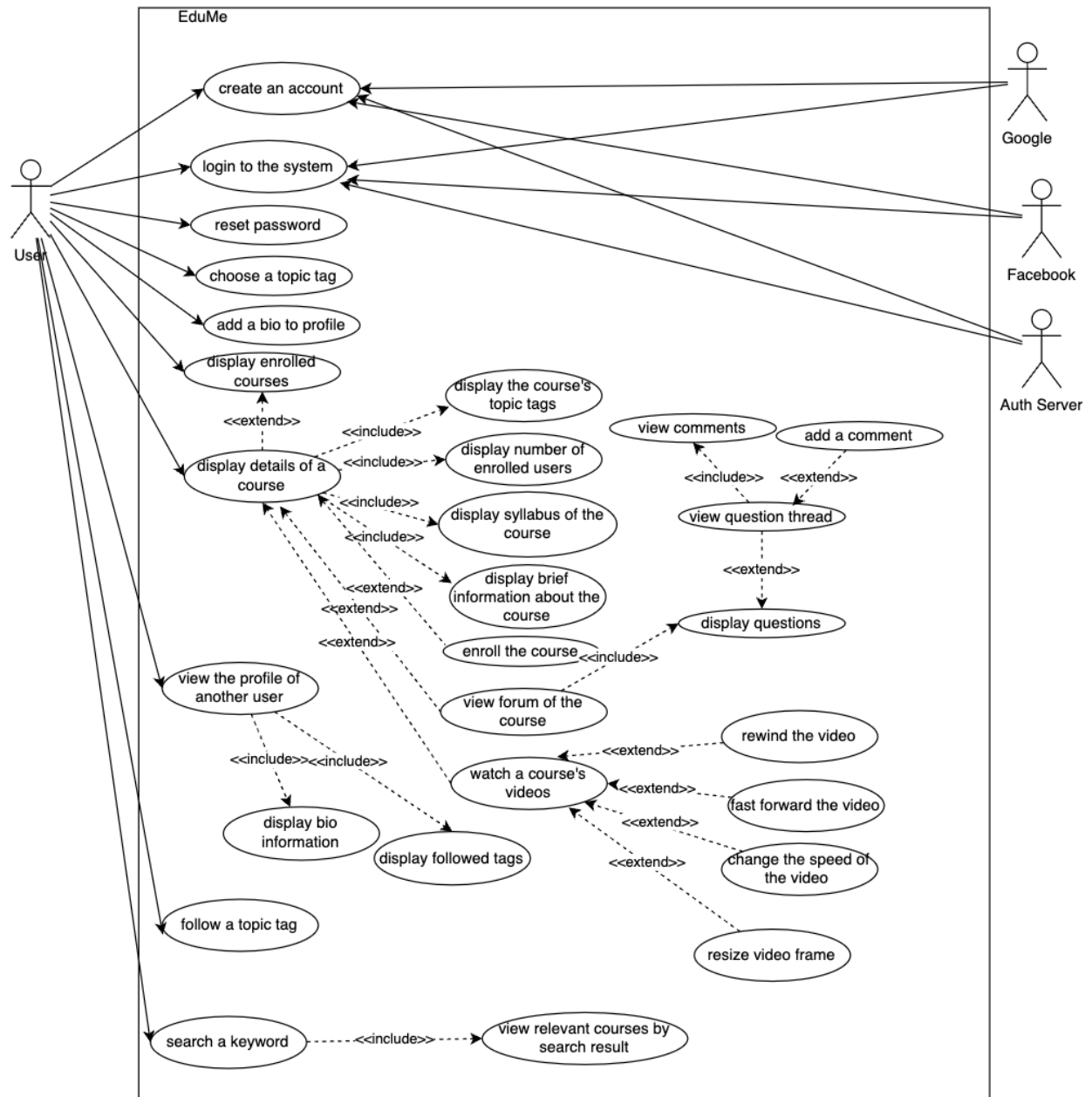
## **Functional Requirements**

1. The system shall use an authentication mechanism for all interactions with the system.
2. Users shall be able to create an account with their Google account.
3. Users shall be able to create an account with their Facebook account.
4. Users shall be able to create an account with their personal information (first name, last name, age) and username, email and password.
5. When a user creates an account, the system shall check whether a given username and email are in use.
6. When a user creates an account successfully, the system shall send a successful account creation email to the user.
7. If the given username is in use, then the system shall display an error message "The provided username is currently in use, please choose a different username" to the user.

8. If the given email is in use, then the system shall display an error message "The provided email is currently in use, please choose a different email" to the user.
9. Users shall be able to log in to the system by using their username and password.
10. Users shall be able to log in to the system with their Google account.
11. Users shall be able to log in to the system with their Facebook account.
12. Users shall be able to reset her/his password.
13. When a user request to reset her/his password, the system shall send "reset password" link to the user's email.
14. While a user is in the password reset page, the system shall enable the user to change her/his password.
15. Users shall be able to follow a topic tag that they are interested in after first login.
16. When a user followed a topic tag, the system shall recommend other topics that can be related to the chosen topic.
17. Users shall be able to add a bio to their profile.
18. Users shall be able to see a main page that is composed of courses that share the same topic tags with the user.
19. Users shall be able to display details of a course.
20. While a user is in the detail page of a course, the system shall enable the user to watch introduction video of the course.
21. While a user is in the detail page of a course, the system shall display the course's topic tags.
22. While a user is in the detail page of a course, the system shall display how many users have enrolled to the course.
23. While a user is in the detail page of a course, the system shall display syllabus of the course.
24. While a user is in the detail page of a course, the system shall display brief information about the course.
25. While a user is in the detail page of a course, the system shall display information about the instructors.
26. While a user is in the detail page of a course, the system shall enable the user to enroll the course.
27. Users shall be able to see their enrolled courses.
28. While a user is in the detail page of an enrolled course, the system shall display sections of the course.
29. While a user is in the detail page of an enrolled course, the system shall enable the user to choose a section.
30. While a user is in the section of an enrolled course, the system shall enable the user to watch the video of that section.

31. While a user is in the section of an enrolled course, the system shall display information about that section.
32. While a user is in the detail page of an enrolled course, the system shall enable a user to view forum of the course.
33. While a user is in the forum page of an enrolled course, the system shall display questions that are asked by other users of the course.
34. While a user is in the forum page of an enrolled course, the system shall enable the user to view thread of a chosen question.
35. While a user is in the forum page of an enrolled course, the system shall enable the user to ask a new question.
36. While a user is in the thread page of a question, the system shall display comments of other users to the question.
37. While a user is in the thread page of a question, the system shall enable the user to add a 250 characters length comment to that thread.
38. Users shall be able to view the profile of another user in the system.
39. While a user is in the profile page of another user, the system shall display bio information and followed tags of that user.
40. Users shall be able to rewind a video.
41. Users shall be able to fast forward a video.
42. Users shall be able to change the speed of a video to 0.5x, 0.75x, 1.0x, 1.25x, 1.50x, 2.0x.
43. Users shall be able to resize video frame.
44. Users shall be able to see length of the video in "hh:MM:ss" format.
45. When a user encounters a topic tag in the user interface, the system shall enable the user to follow that topic tag.
46. Users shall be able to search for a courses by entering a keyword.
47. When a user searches for a keyword, then the system shall list relevant courses by the given keyword.

## Use Case Diagram

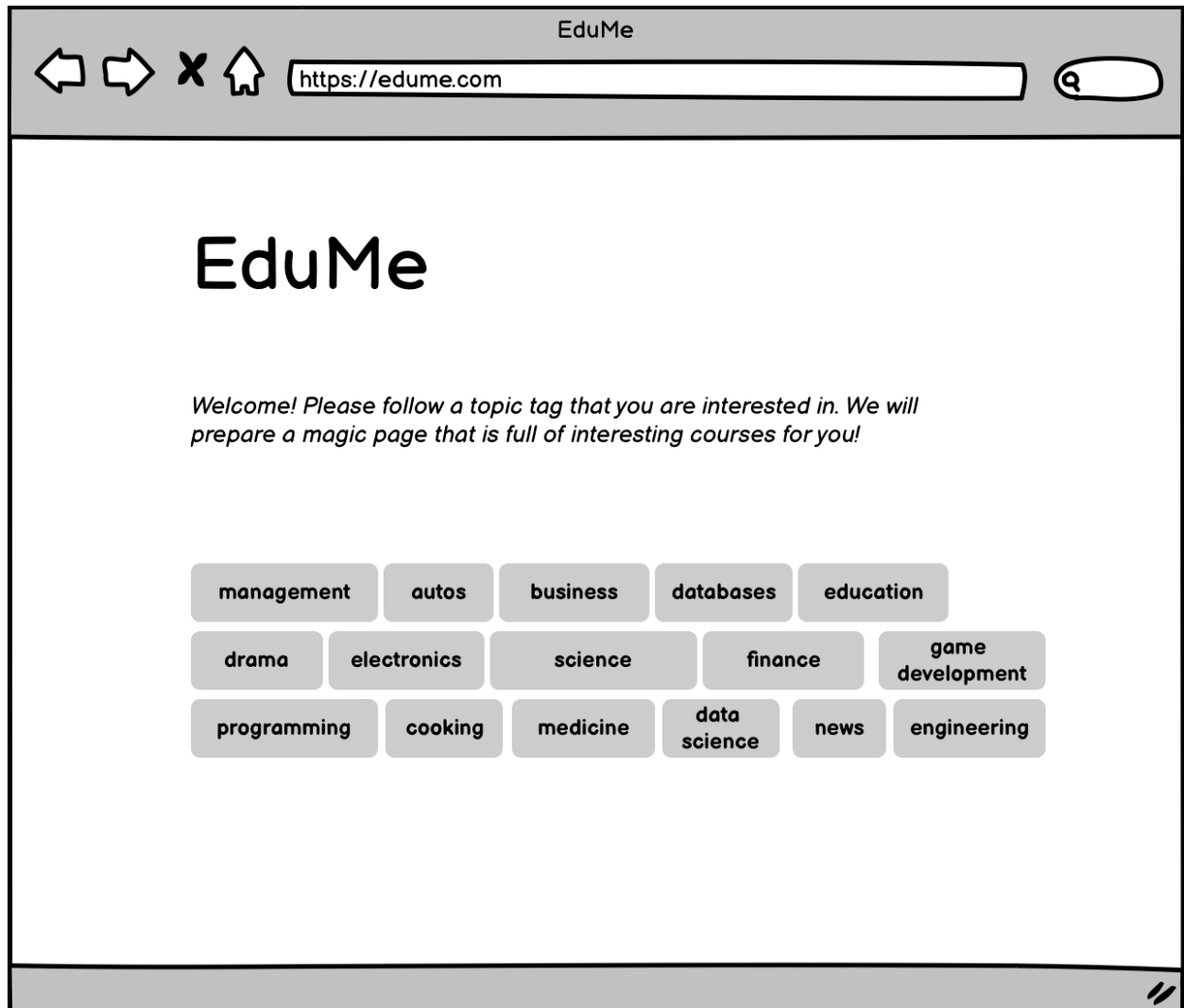


## **Non-Functional Requirements**

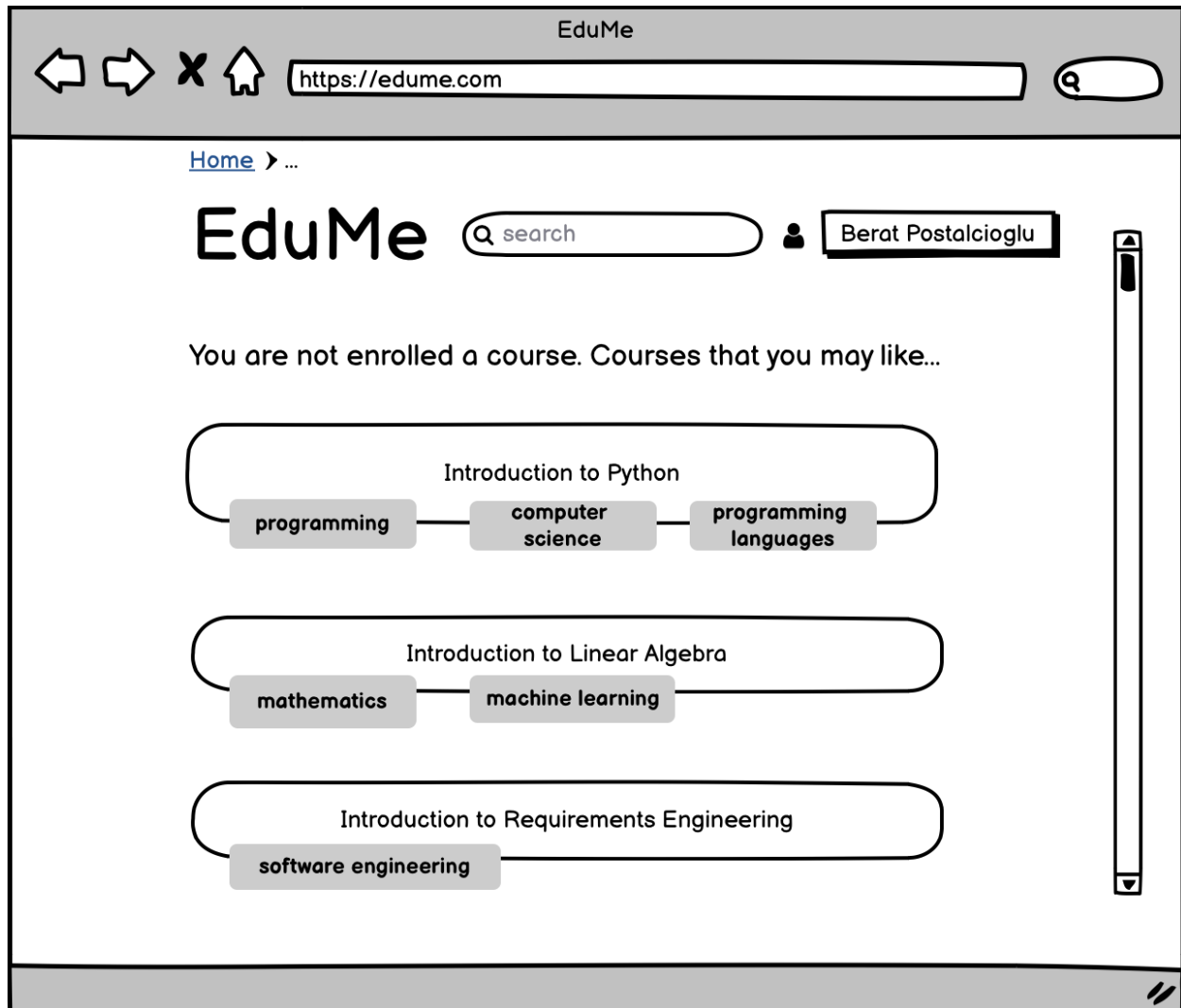
1. The system should be available 99% of the time.
2. The system operations shouldn't take longer than 5 seconds 99% of the time.
3. The system shall be developed with microservice architecture.
4. Frontend will be coded using ReactJS.
5. Backend will be coded using Python Django Framework.
6. The system will be running on AWS Cloud Services.
7. Github will be used as a version control system.
8. Trello and Github will be used as a task manager.
9. Slack will be used as a main communication tool.
10. MySQL will be used as a RDBMS.

## **Mockups**

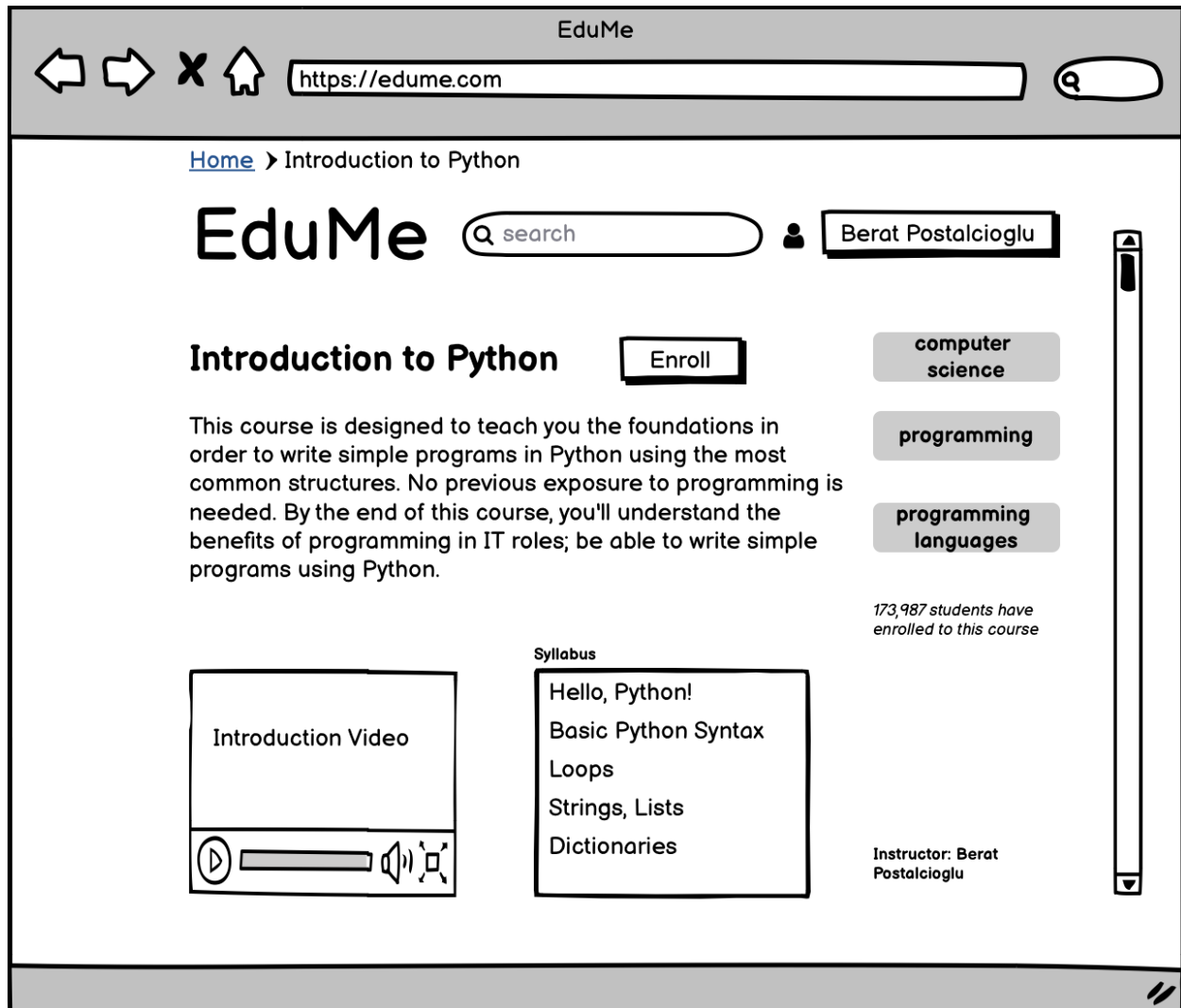




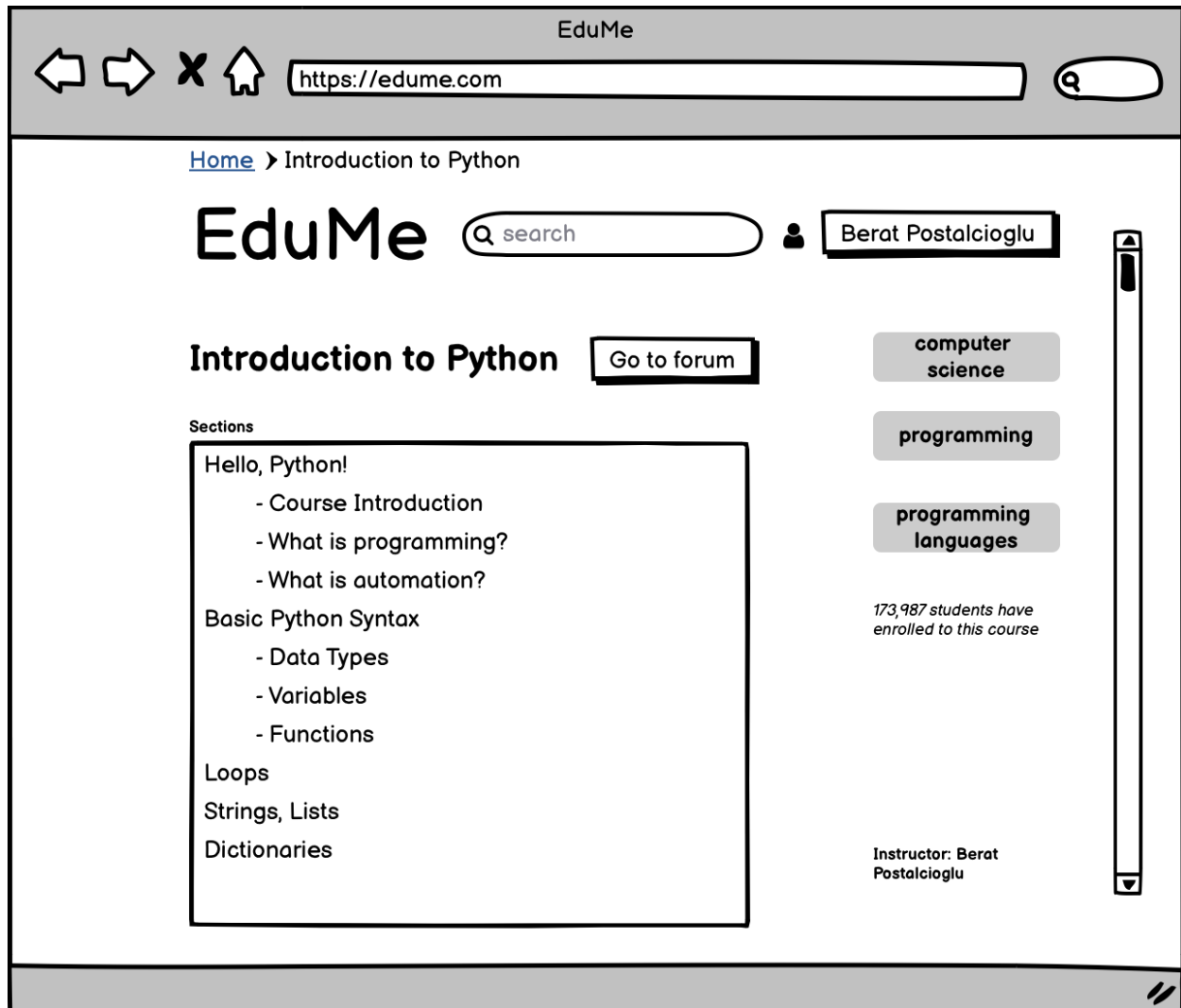
Requirements: 12



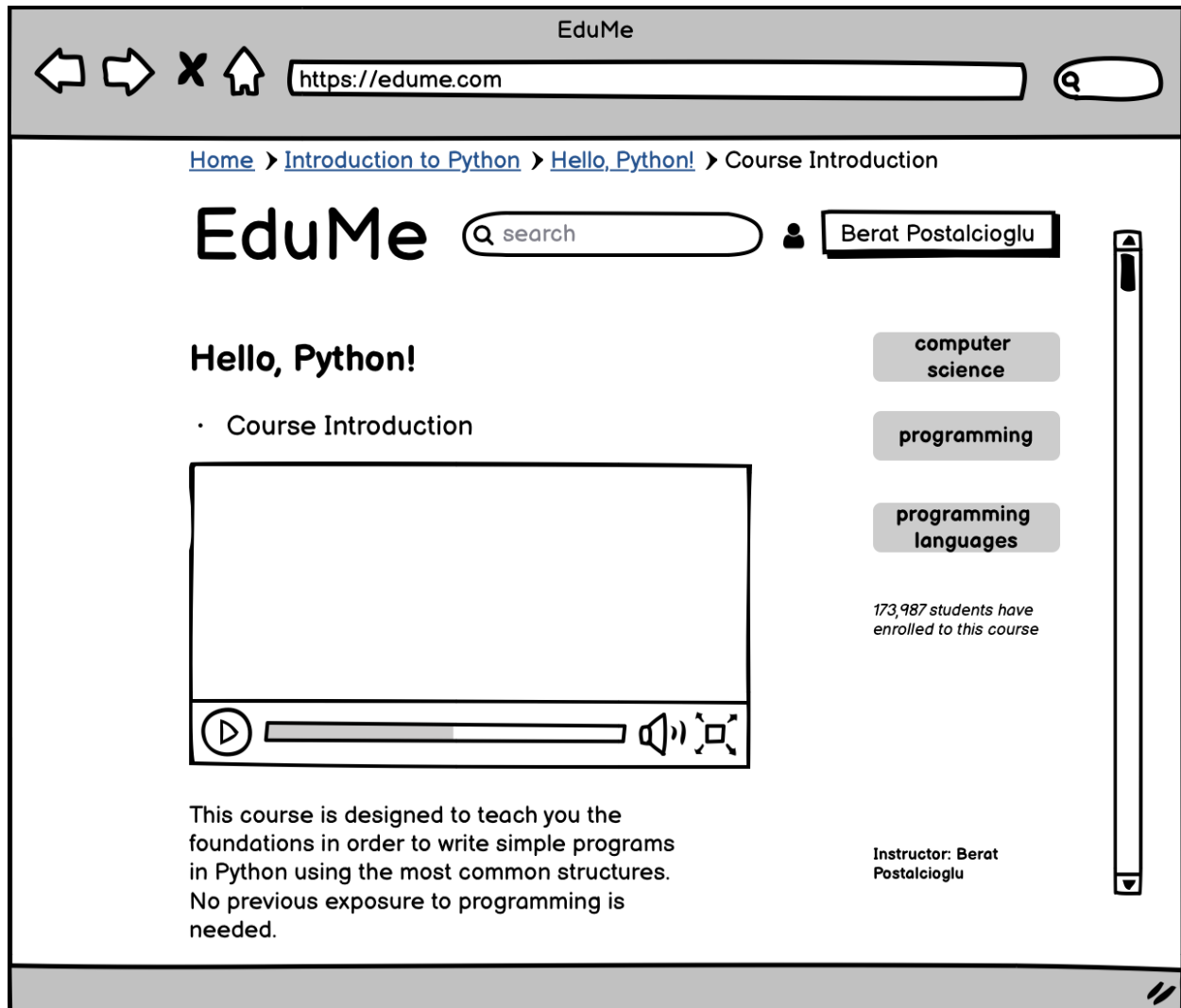
Requirements: 13, 15



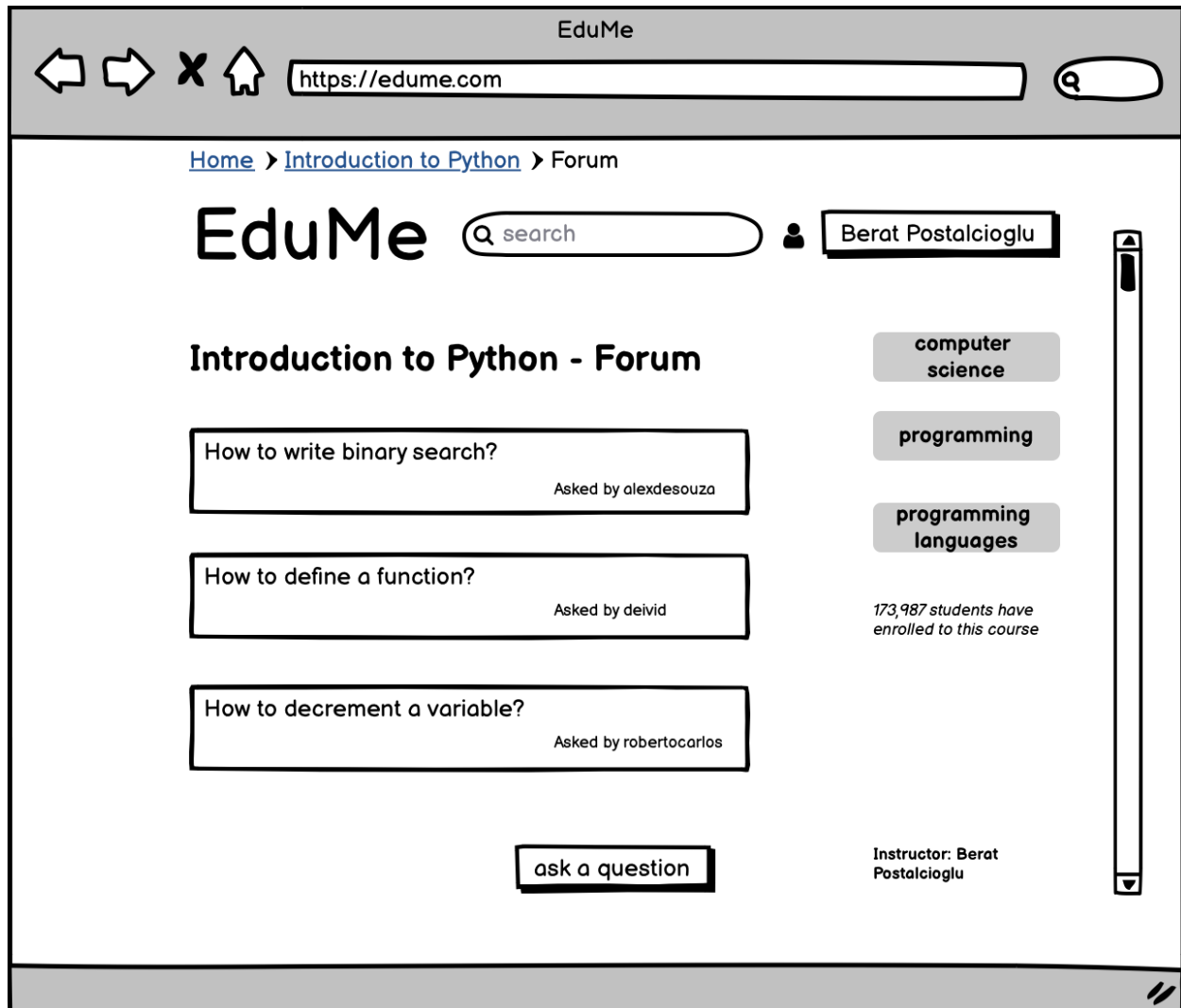
Requirements: 16, 17, 18, 19, 20, 21, 22, 23



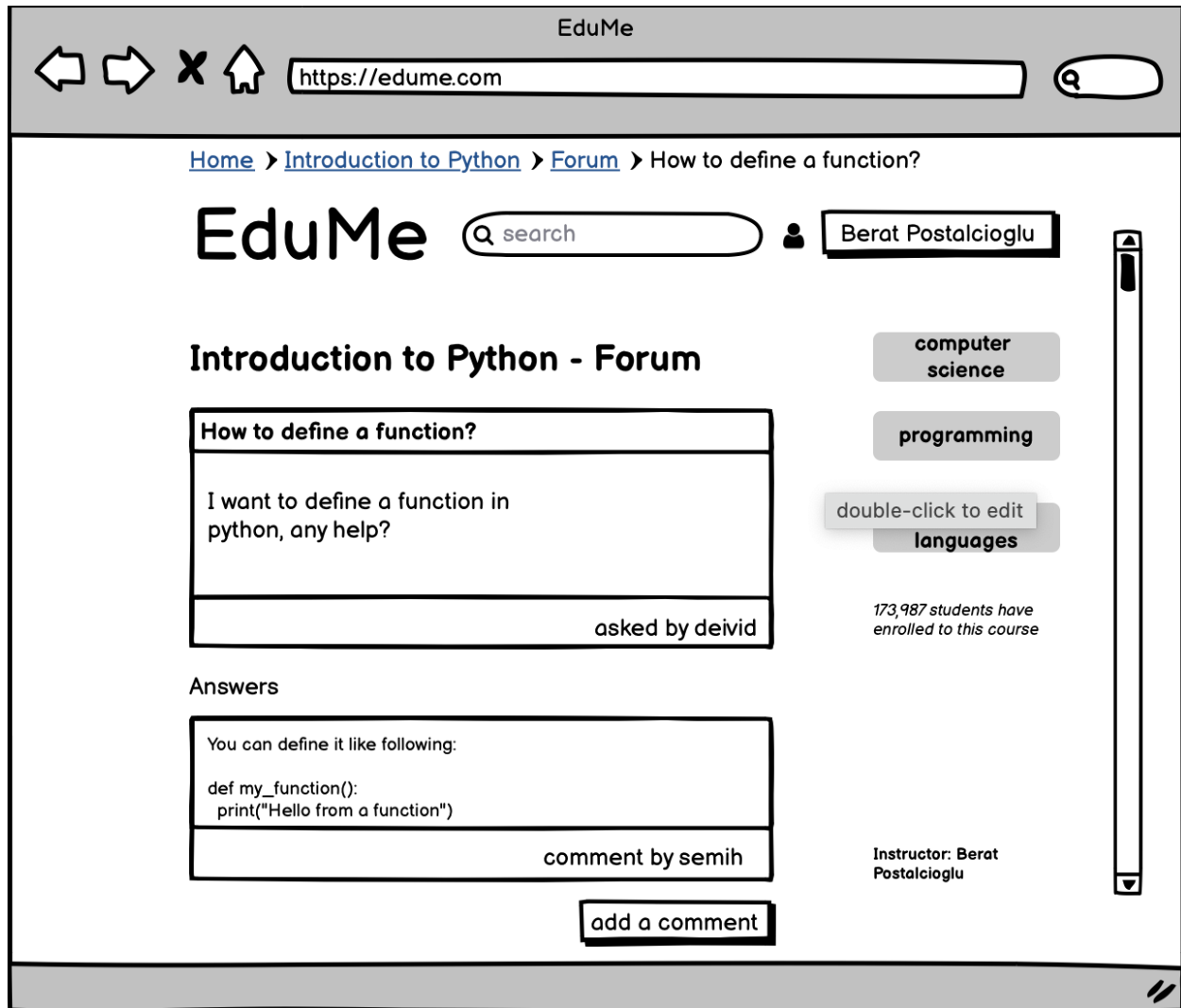
Requirements: 25, 26, 29



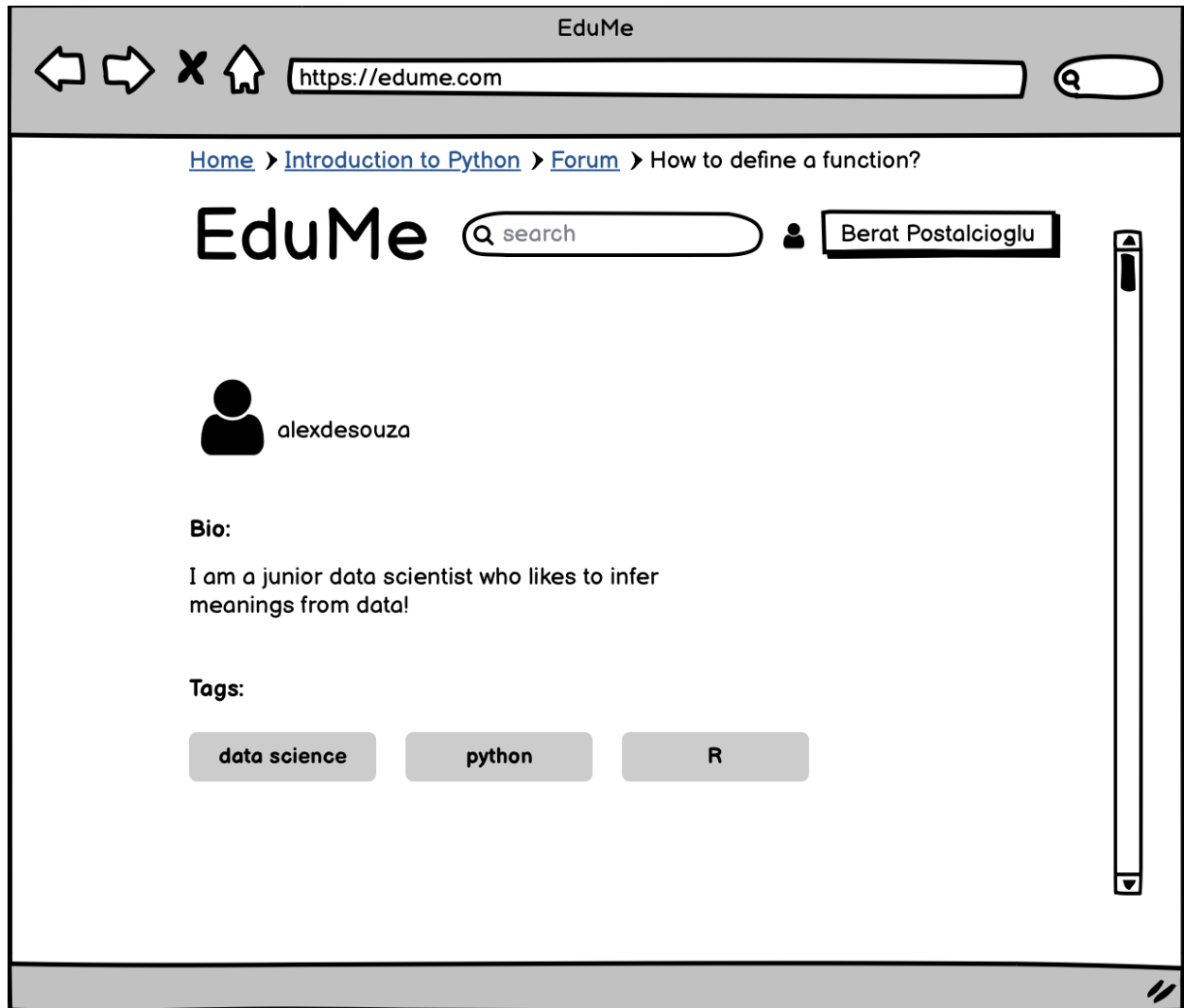
Requirements: 27, 28



Requirements: 29, 30, 32

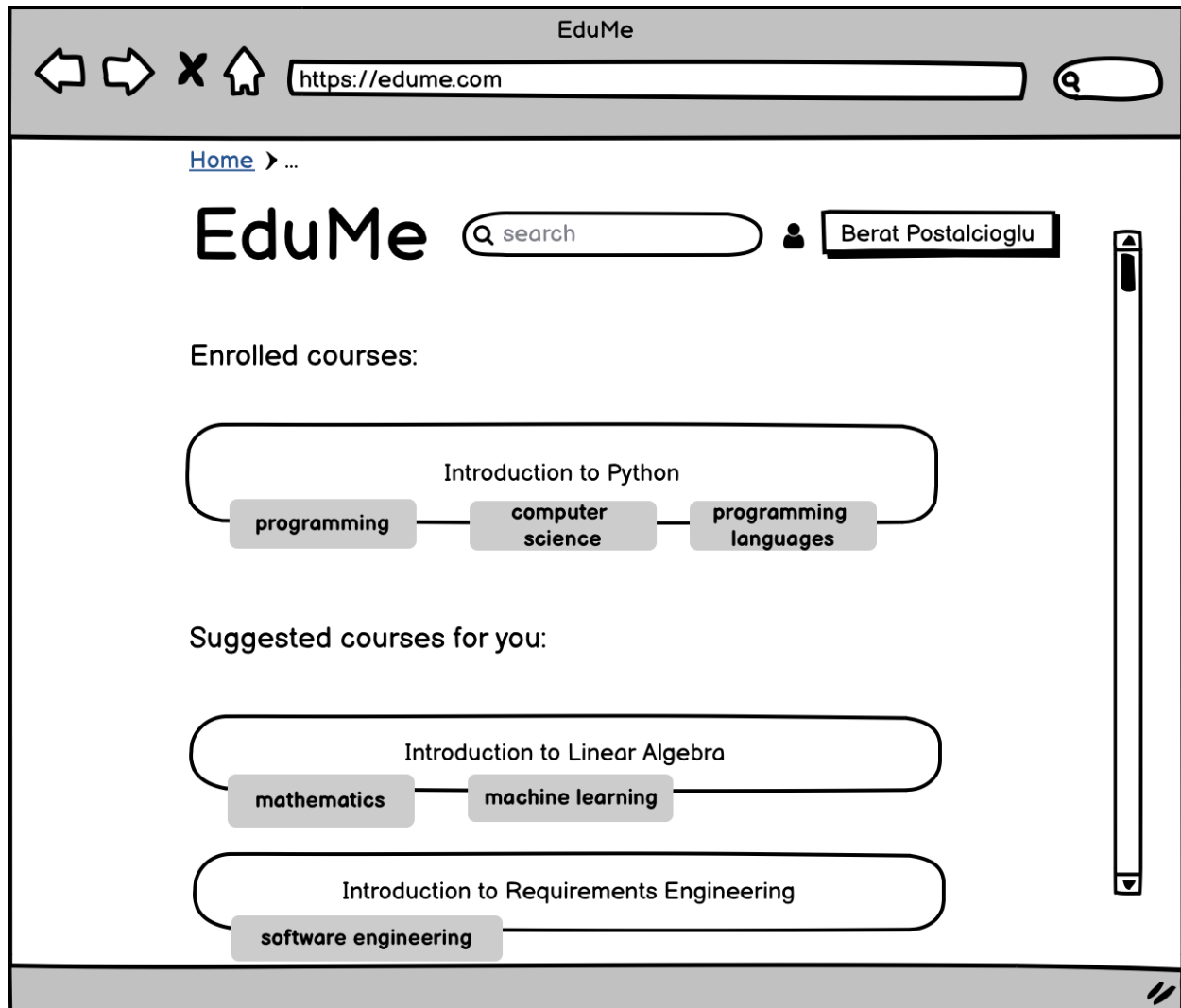


Requirements: 31, 33, 34

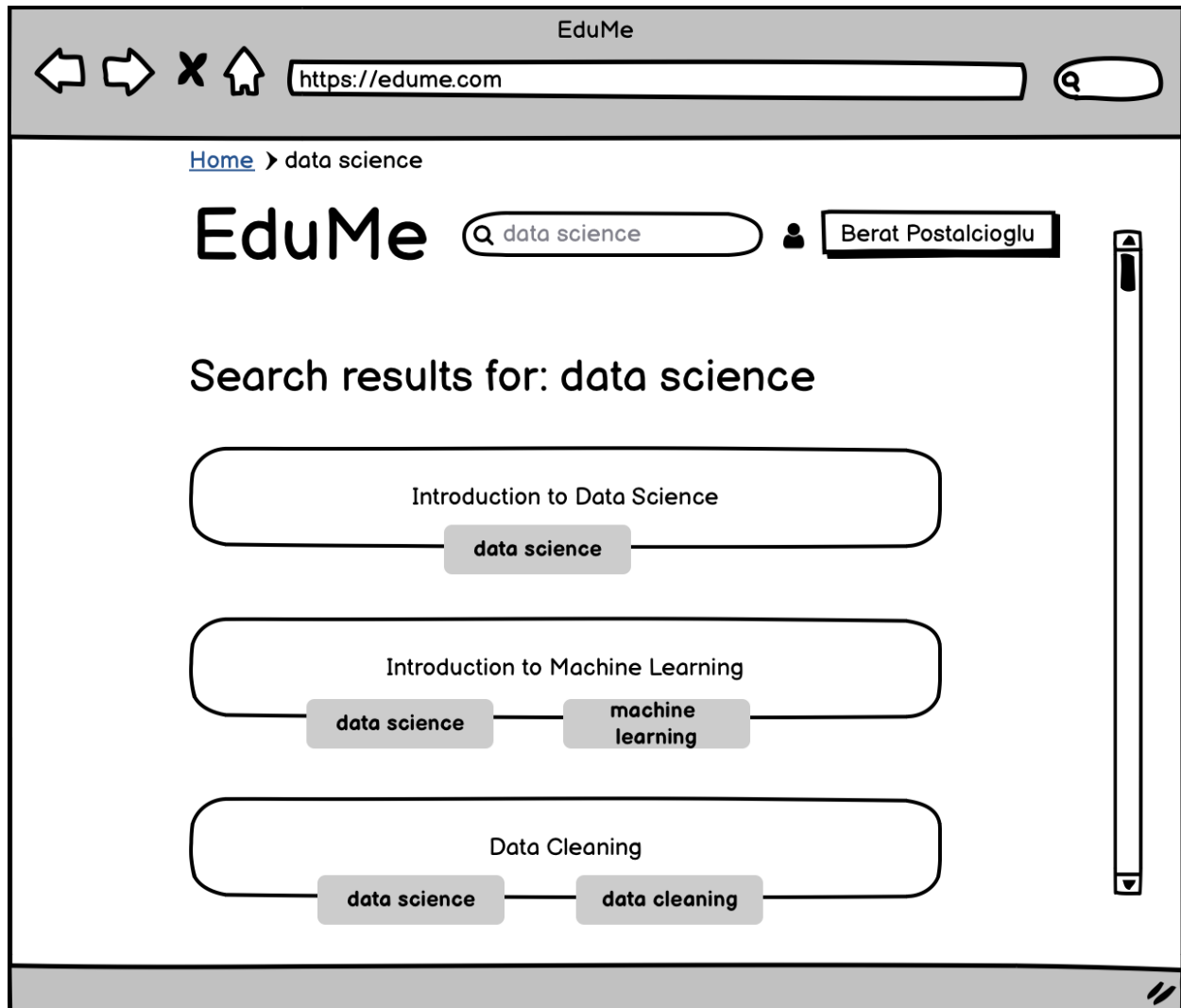


Requirements: 35, 36



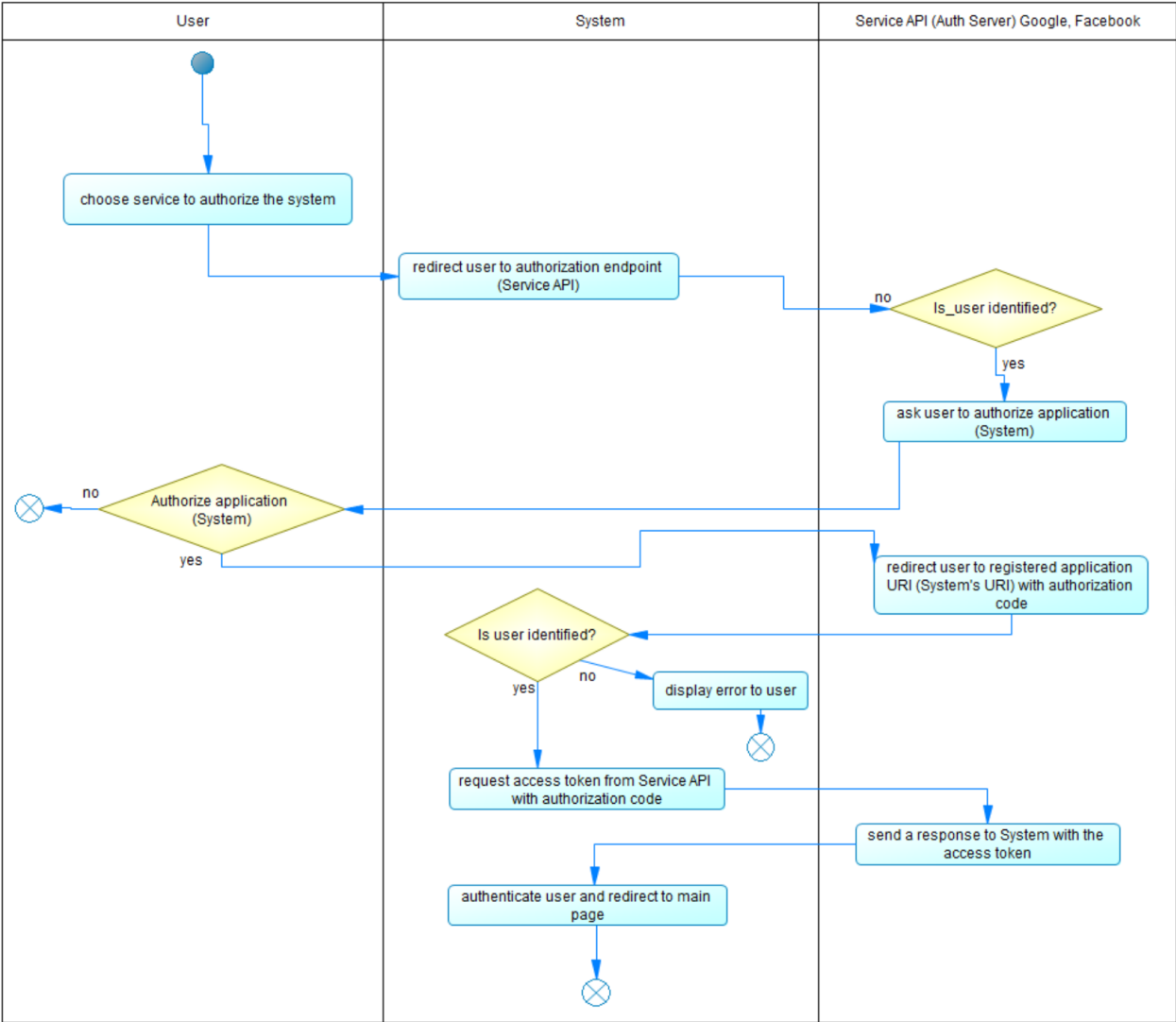


Requirements: 24



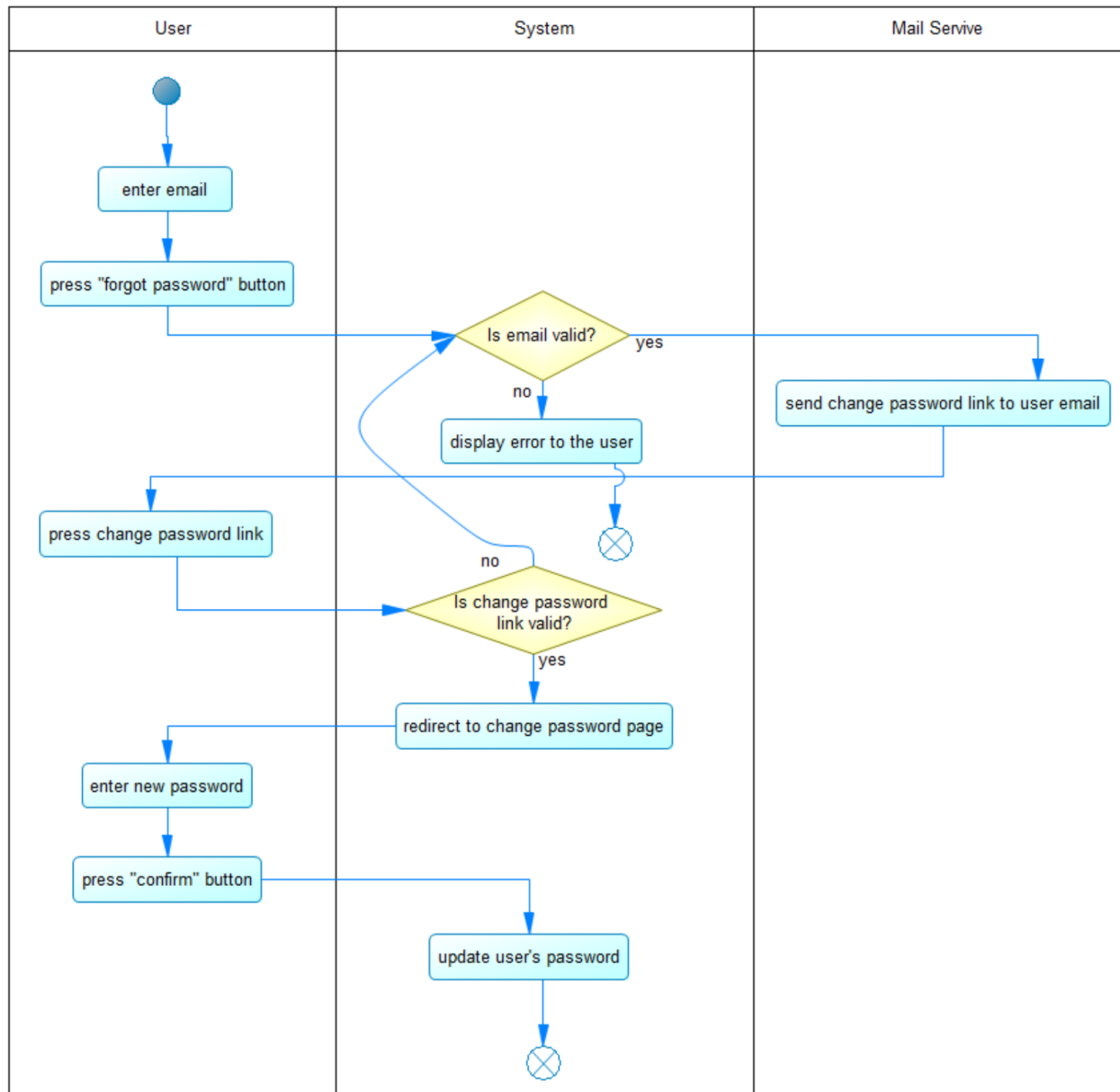
Requirements: 43, 44

# Software Design Artifacts



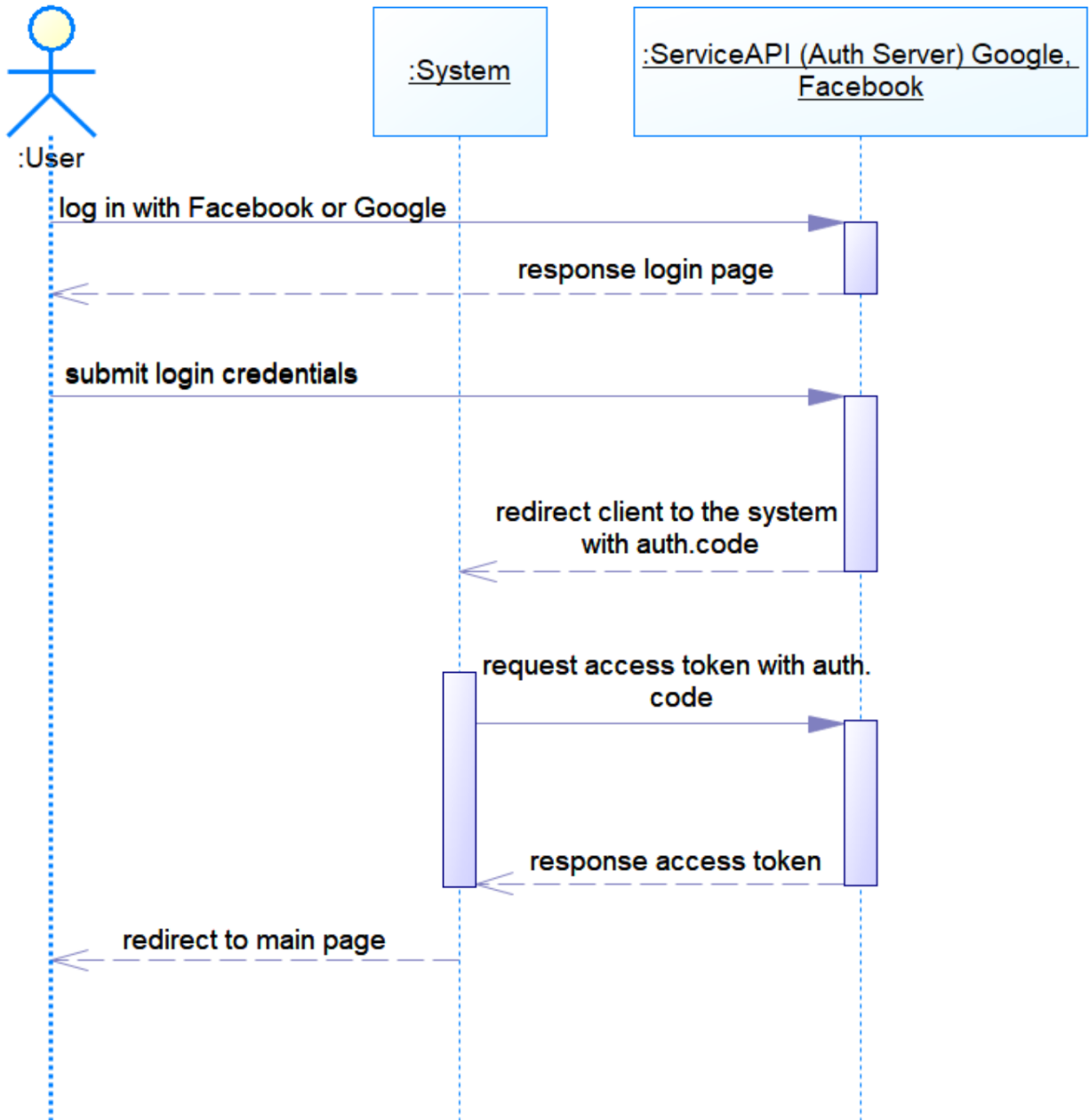
(Login to the system activity diagram)

Requirements: 9, 10



(Reset password activity diagram)

Requirements: 12, 13, 14



(Login to the system sequence diagram)

Requirements: 9, 10

# Status of the project

## Completed Functional Requirements:

4. Users shall be able to create an account with their personal information (first name, last name, age) and username, email and password.
15. Users shall be able to follow a topic tag that they are interested in after first login.
17. Users shall be able to add a bio to their profile.
18. Users shall be able to see a main page that is composed of courses that share the same topic tags with the user.
19. Users shall be able to display details of a course.
20. While a user is in the detail page of a course, the system shall enable the user to watch introduction video of the course.
21. While a user is in the detail page of a course, the system shall display the course's topic tags.
23. While a user is in the detail page of a course, the system shall display syllabus of the course.
24. While a user is in the detail page of a course, the system shall display brief information about the course.
28. While a user is in the detail page of an enrolled course, the system shall display sections of the course.
29. While a user is in the detail page of an enrolled course, the system shall enable the user to choose a section.
30. While a user is in the section of an enrolled course, the system shall enable the user to watch the video of that section.
31. While a user is in the section of an enrolled course, the system shall display information about that section.

- 32. While a user is in the detail page of an enrolled course, the system shall enable a user to view forum of the course.
- 33. While a user is in the forum page of an enrolled course, the system shall display questions that are asked by other users of the course.
- 34. While a user is in the forum page of an enrolled course, the system shall enable the user to view thread of a chosen question.
- 35. While a user is in the forum page of an enrolled course, the system shall enable the user to ask a new question.
- 36. While a user is in the thread page of a question, the system shall display comments of other users to the question.
- 37. While a user is in the thread page of a question, the system shall enable the user to add a 250 characters length comment to that thread.
- 40. Users shall be able to rewind a video.
- 41. Users shall be able to fast forward a video.
- 44. Users shall be able to see length of the video in "hh:MM:ss" format.

## **Status of Deployment**

The project is dockerized and deployed to  
<http://ec2-54-161-242-148.compute-1.amazonaws.com/> AWS Cloud.

# System Manual

Before running up the system you need to ensure your system has the Docker tool. You can install Docker to your system by following instructions on the link:

<https://docs.docker.com/engine/install/>

After installing Docker to your system you must add the “edume.env” file that contains environment variables to run the system to the “backend/edume/” path of the project. You need to define following environment variables:

```
DJANGO_SECRET
DB_NAME
DB_USER
DB_PASS
DB_HOST
DB_PORT
EDUME_DOMAIN
```

After creating the “edume.env” file. You need to run following commands in “backend” path of the project to run the system:

```
docker build -t edume .
```

```
docker run -p 8000:8000 -dit edume
```

Then you can reach the running system by visiting ‘localhost:8000’ in your preferred browser.



# User Manual

After reaching out the system with provided URI. You need to create a user profile by filling all the required fields, then press the “Create” button at the bottom right corner.

## Create A New User Profile

Name:

Surname:

Birth date:

Email:

Bio:

Password:

Re-type password:

[Create](#)

After you successfully created a user account you need to select topics that you are interested in, then press “Add Selected Topics” button.

# EduMe

Jorge

Hello Jorge to EduMe!

### Choose topics to follow:

programming  
python  
science  
education  
data science

Hold down the Ctrl (windows) or Command (Mac) button to select multiple options.

[Add Selected Topics](#)

After selecting topics, the courses page will meet you and you can see courses that are related to your chosen topics.



Courses for your selected topics:

Introduction to Data Science
Online Education
Introduction to Programming

You can click on a course to navigate to its detailed page



## Introduction to Data Science

### Course Description:

Python is the most important language in the field of data, and its libraries for analysis and modeling are the most relevant tools to use. In this course we will start building the basics of Python and then going to deepen the fundamental libraries like Numpy, Pandas, and Matplotlib.

**Topics:** python, data science,

### Chapters:

- [Introduction](#)

[Go to Forum](#)

In the course detail page you can see description of the course, topics that the course related to, chapters of the course and a button that navigates you to the forum of the course which allows users to experience collaborative learning.

Since there is only one section in this course click "Introduction" chapter.

### Chapter Description:

Introduction and overview of the course

### Chapter Video:

```
from django.utils import timezone

class Topic(models.Model):
    name = models.CharField(max_length=200)
    create_date = models.DateTimeField(default=timezone.now)

    def publish(self):
        self.save()

    def __str__(self):
        return self.name

class Course(models.Model):
    name = models.CharField(max_length=200)
```

This chapter contains only video content and you can watch this chapter's video.

In the forum section of the course. You can see the questions that are asked by other users in the course and answers to these questions. You can also ask a question about the course, and also answer an existing question of another user.

### Questions:

How to define a function in python?

Asked by Berat

def func():

Answered by Berat

Submit Answer

### Ask new question:

Submit Question

## Test Results

Manual tests performed several times and the system works as expected.

## Project Repository

<https://github.com/bpostalci/bounswe573-2022/tree/master/edume>