SWE 573

2019 Spring

Project Report

by

Kasım Bozdağ

SWE 573 Project Report

1. Project description

In this project, we will try to create a platform for learning where users will be able to find the desirable contents by searching. The goal is to create a learning space that is engaging and to allow users to find more related data on the subject

There will be two roles in this project: teachers and students. The reason we are calling them teachers and students is for a lack of better word. The teacher here doesn't have to be a real teacher. It could be anyone who wishes to share their knowledge or research on this platform in order to reach an audience that is interested which we will call as students.

Teachers can create any content that is engaging and motivates critical thinking. The content can be anything that teachers and students are interested in. It could be sports, hobbies, professional interests, politics, social and economic issues, governments, health care etc. After the teacher create the topic, he/she will provide more material in order to allow students to understand the concepts and further exploration.

Teachers will be able publish material that contains basic knowledge and questions about the knowledge. They will be able to tag the material with specific keywords which will allow the students to search and find related data. Same words can have different meaning for different field so the teachers must be able to specify which meaning they referring to. There should be a glossary and reference relating topic to be published. The same concepts can be published by many different teachers based on their style and research.

The purpose of this project is to create a platform which share materials on basic concepts that can arouse the interest of the students and make them more engaged in learning. So, the questions should be able further this interest and motivate the students to look for more.

The students will be able to search and explore the contents based on basic concepts they contain, answer the question relating to the topic and follow their progress on said topics.

Since the progress of the student is followed it could provide the teacher a feedback on the quality of the published topic. The teacher then can edit the content and change the questions to make the topic more engaging.

Users should be able follow other users and recommend user and topic.
In order to differentiate keywords semantic tags should be used. Wikidata and DBpedia can be used for semantics.

2. Requirements

Glossary

System: The platform software about the learning space we are developing

Users: The people who will use the web applications.

Unregistered User: A person who enters the system with limited access.

Registered User: A person who has created his/her account and use the application with partial access based on user type.

Teacher: A user that adds content to the application

Learner: A user that views the contents in the application and take quizzes.

Sub-content: A sub-content consists of a sub-title and a picture and a short text. A sub-content must have a short text, picture and sub-title are optional.

Lesson: A lesson consists of a title picture and a description, and a series of arranged sub-content.

Course: A series of lessons arranged in a sequence by a teacher. A course will have a title, a short text description and a picture all mandatory.

Password: A secret series of characters that enables a user to access his/her own account. The password helps ensure that unauthorized users do not access that account. In addition, data files and programs may require a password.

Quiz: A series of question design to assess learners understanding of the lesson.

Question: A multi-choice question about the content of the lesson. A question can have many choices and a single answer.

Pre-requisite: A condition for being able to take a lesson or a course or a quiz. It could be a course, a lesson or a quiz which was answered correctly %100. Not every lesson has a pre-requisite.

Tag: A label attached to a topic for the purpose of identification that has been retrieved from wikidata. A tag consists of a label(keyword), a source (wikidata reference (URI)) and a description. There will be a relation between a tag and a target (course, lesson, question ...) with an issuer.

Functional Requirements

1. *User Requirements*
   - 1.1. Unregistered User Requirements
     - 1.1.1. Users shall be able to view the list of course available.
     - 1.1.2. Users shall be able to sign up to the system by using an email and a 'password'.
   - 1.2. Registered User Requirements
     - 1.2.1. Users shall be login to the system using email and 'password'.
       - 1.2.1.1. Users shall be able to reset their password with the 'forgot password' option.
     - 1.2.2. Users shall be able to view the list of course available.
     - 1.2.3. Teachers shall be able to manage their courses.
       - 1.2.3.1. Teachers shall be able to add a course to the system.

1.2.3.2. Teachers shall be able to add tags to a course that they created.
1.2.3.3. Teachers shall be able to edit a course that they created.
1.2.3.4. Teachers shall be able to make the course invisible by making it inactive.
1.2.3.5. Teachers shall be able to add pre-requisite to a course that they created.
1.2.3.6. Teachers shall be able to arrange the lesson list (change their places)
1.2.4. Teachers shall be able to manage their course lessons.
1.2.4.1. Teachers shall be able to add a lesson to a course.
1.2.4.2. Teachers shall be able to add tags to a lesson that they created.
1.2.4.3. Teachers shall be able to add pre-requisite to a lesson that they created.
1.2.4.4. Teachers shall be able to edit a lesson that they created.
1.2.4.5. Teachers shall be able to make the lesson invisible by making it inactive.
1.2.4.6. Teachers shall be able to arrange sub-content list
1.2.5. Teacher shall be able to search for content
1.2.6. Teacher shall be able to bookmark the content they find
1.2.7. Teacher shall be able to add reference to the lessons
1.2.8. Teachers shall be able to manage their lesson sub-contents.
1.2.8.1. Teachers shall be able to add a sub-content to a lesson
1.2.8.2. Teachers shall be able to edit a sub-content
1.2.8.3. Teachers shall be able to make content invisible by making it inactive
1.2.8.4. Teachers shall be able to manage their lesson quizzes.
1.2.9. Teachers shall be able to add a quiz to a lesson.
1.2.9.1. Teachers shall be able to make the quiz invisible by making it inactive.
1.2.10. Teachers shall be able to manage their quiz questions.
1.2.10.1. Teachers shall be able to add a question to a quiz.
1.2.10.2. Teachers shall be able to add tags to a question that they created.
1.2.10.3. Teachers shall be able to edit a question that they created.
1.2.10.4. Teachers shall be able to make the question invisible by making it inactive.
1.2.11. Teachers shall be able to add choices to the questions
1.2.11.1. Teachers shall be able to choose a correct choice
1.2.11.2. Teachers shall be able to edit choices
1.2.11.3. Teachers shall be able to deactivate choices
1.2.12. Learners shall be able to enroll to a course
1.2.13. Learners shall be able to view a lesson if they fulfilled the pre-requisite.
1.2.14. Learners shall be able to take the quiz of a lesson that they have taken.
1.2.14.1. Learners shall be able to submit the quiz at any stage of the quiz.
1.2.14.2. Learners shall be able to resume a quiz that they have taken and didn't finish.

1.2.14.3. Learners shall be able to see the answer to the question that they have answered.
1.2.14.4. Learners shall be able to view the result of a quiz that they have taken.
1.2.14.5. Learners shall be able to answer the question in a quiz.
1.2.15. Users shall be able to do a search.
1.2.15.1. Users shall be able to search topics by a teacher.
1.2.15.2. Users shall be able to search topics by tags.
1.2.15.3. Users shall be able to search topics by keyword in the topic.
1.2.15.4. Users shall be able to search topics by keyword in the content.
1.2.15.5. Users shall be able to search topics by a question.
1.2.15.6. Users shall be able to search users by name.
1.2.16. Users shall be able to follow and recommend
1.2.16.1. Users shall be able to follow other users.
1.2.16.2. Users shall be able to follow a Course.
1.2.16.3. Users shall be able to recommend users to other users
1.2.16.4. Users shall be able to recommend courses to other users

2. System Requirements

2.1. System shall assess the quizzes.
2.1.1. System shall be asses the question
2.2. System shall be able get statistics
2.2.1. System shall be able keep statics of quizzes
2.2.1.1. Number of attempts, number of completed
2.2.2. System shall be able to keep statics of question
2.2.2.1. Correct answer ratio
2.2.3. System shall be able get statistic of courses
2.2.3.1. Number of visits, number of enrolled, number of completed
2.3. System shall be able to check pre-requisites
2.4. System shall be able to get tags from wikidata base on a keyword entered.

Non-functional requirements

1. Portability
1.1. The system should be able to run on any platform
2. Ease of use
3. Availability

The requirements change a lot during design and implementation stage the original requirements are "https://github.com/kasimbozdag/SWE_573/wiki/Requirements"

3. *Project Plan*

After the requirement stage a project plan was drawn for design, implementation, test and deployment stages

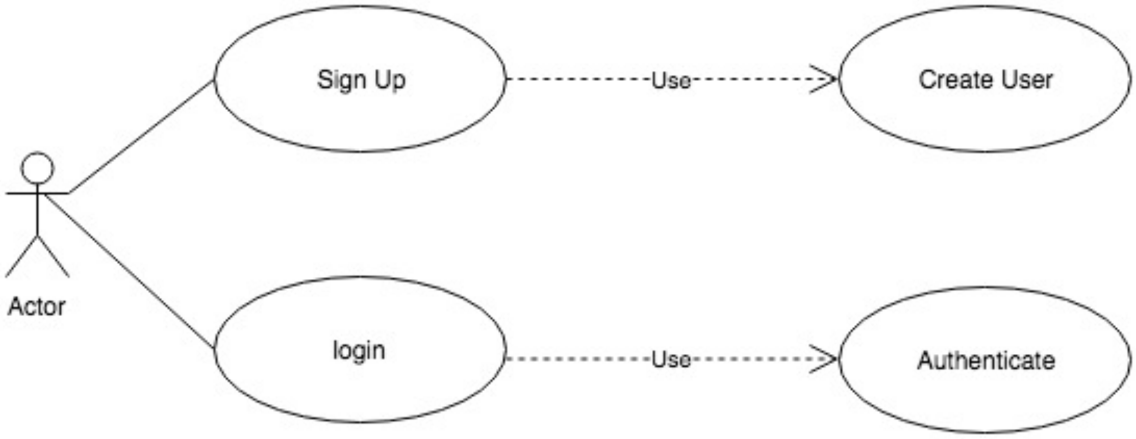
Task	**Start Date**	**End Date**
Requirements	***25/02/2019***	***11/03/2019***
Design	***11/03/2019***	***25/03/2019***
Use Case Diagram	11/03/2019	14/03/2019
Class Diagram	14/03/2019	17/03/2019
Sequence Diagram	17/03/2019	25/03/2019
Implementation	***25/03/2019***	***30/04/2019***
Project Setup	25/03/2019	28/03/2019
Model Implementation	28/03/2019	2/04/2019
Login - Signup	2/04/2019	4/04/2019
Wikidata operation (Tag creation)	4/04/2019	7/04/2019
Course Operations (Create, Update, Deactivate, View and List for student and teacher)	7/04/2019	12/04/2019
Lesson Operations (Create, Update, Deactivate, View and List for student and teacher)	12/04/2019	16/04/2019
Content Operations (Create, Update, Deactivate, View and List for student and teacher)	16/04/2019	20/04/2019
Quiz Operations (Create, Update, Deactivate, View and List for student and teacher, Evaluate)	20/04/2019	22/04/2019
Question Operations (Create, Update, Deactivate, View and List for student and teacher, Evaluate)	22/04/2019	25/04/2019
Wikidata operation (Information search, bookmarking)	25/04/2019	30/04/2019
Test	***25/03/2019***	***07/05/2019***
Deployment	***15/04/2019***	***07/05/2019***

4. Software Design

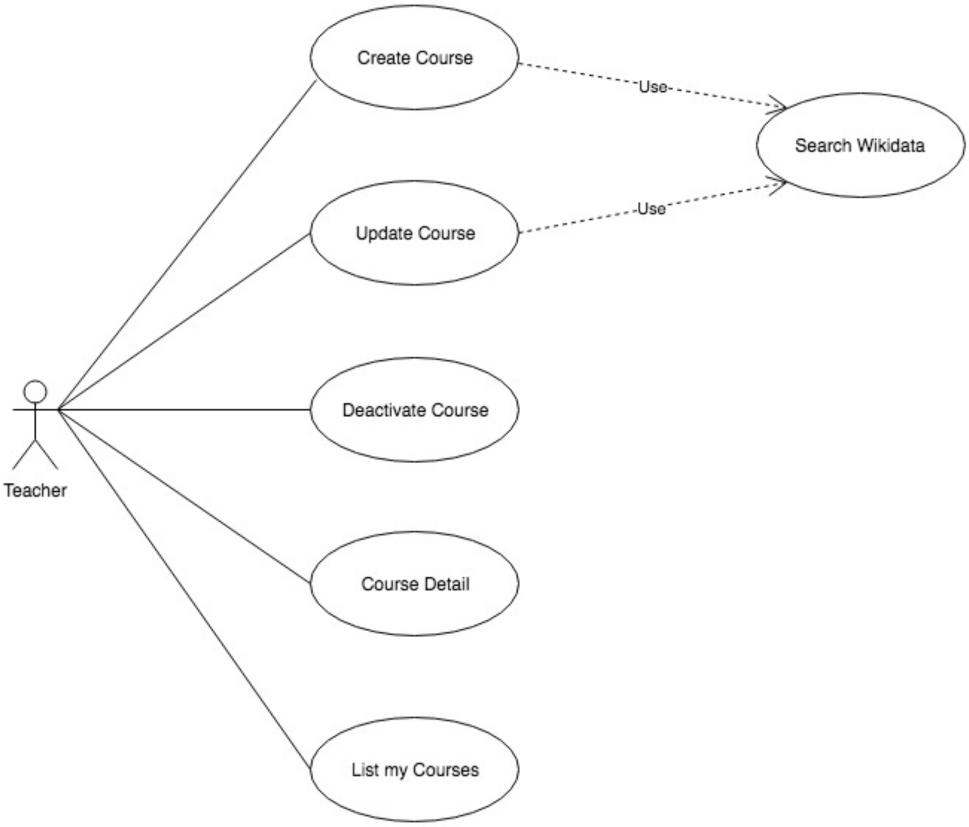
The software design was done after the requirements stage and it does not reflect the changes done in implementation stage

Use cases

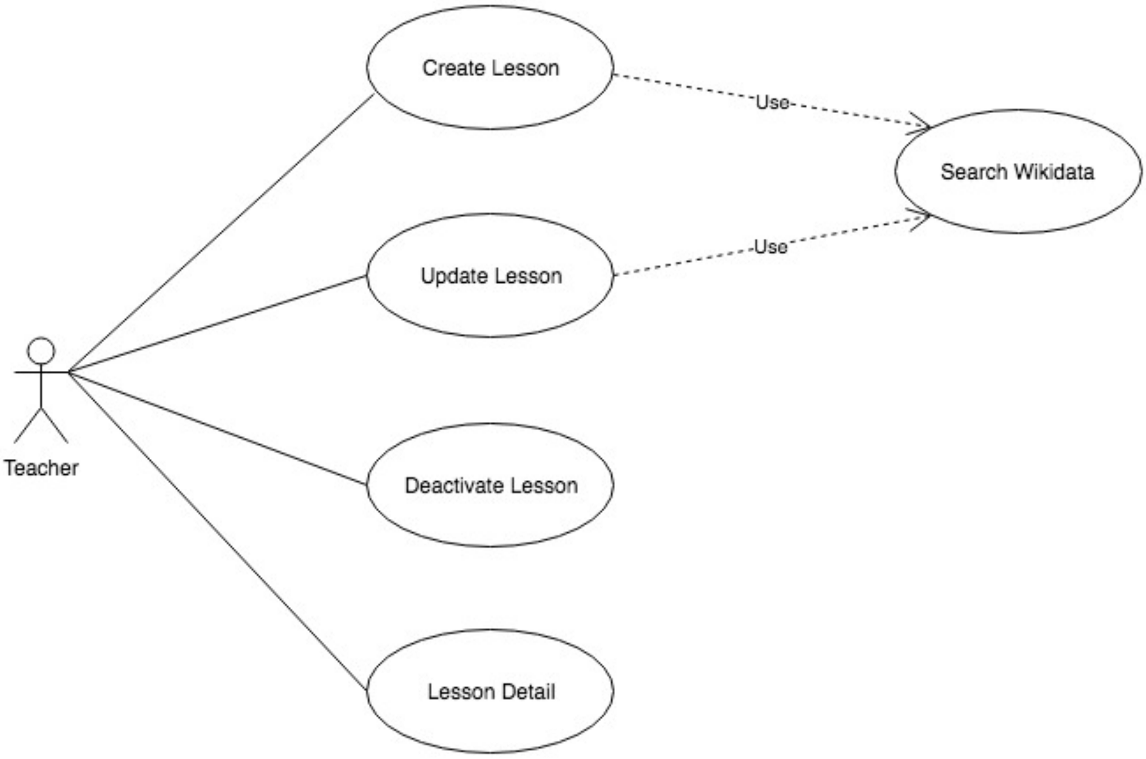
1. User operations



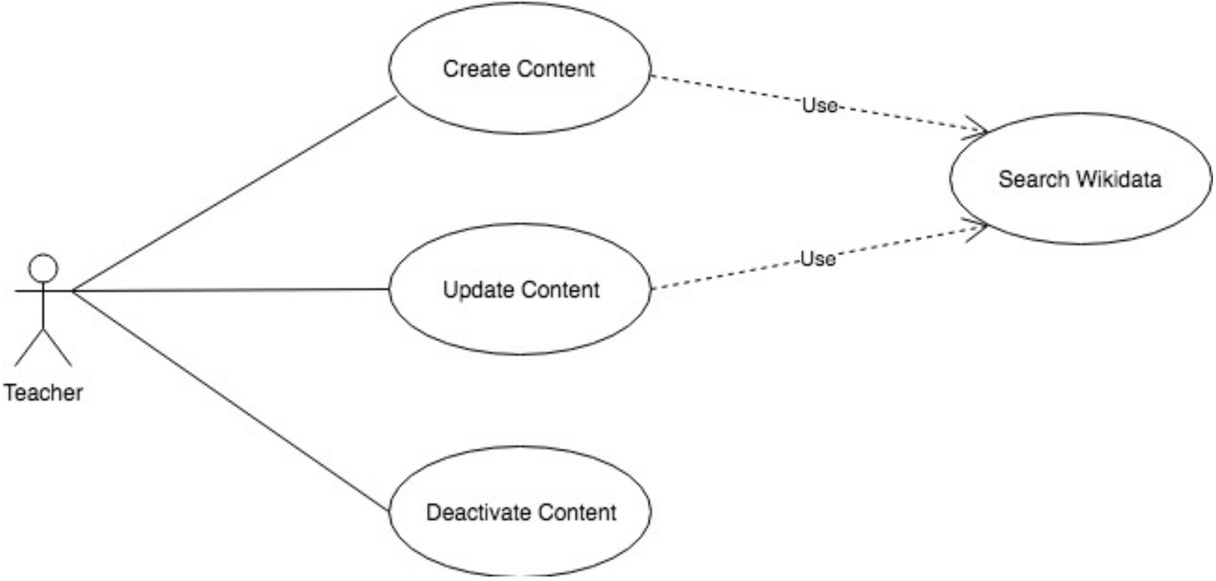
2. Course operations for teacher



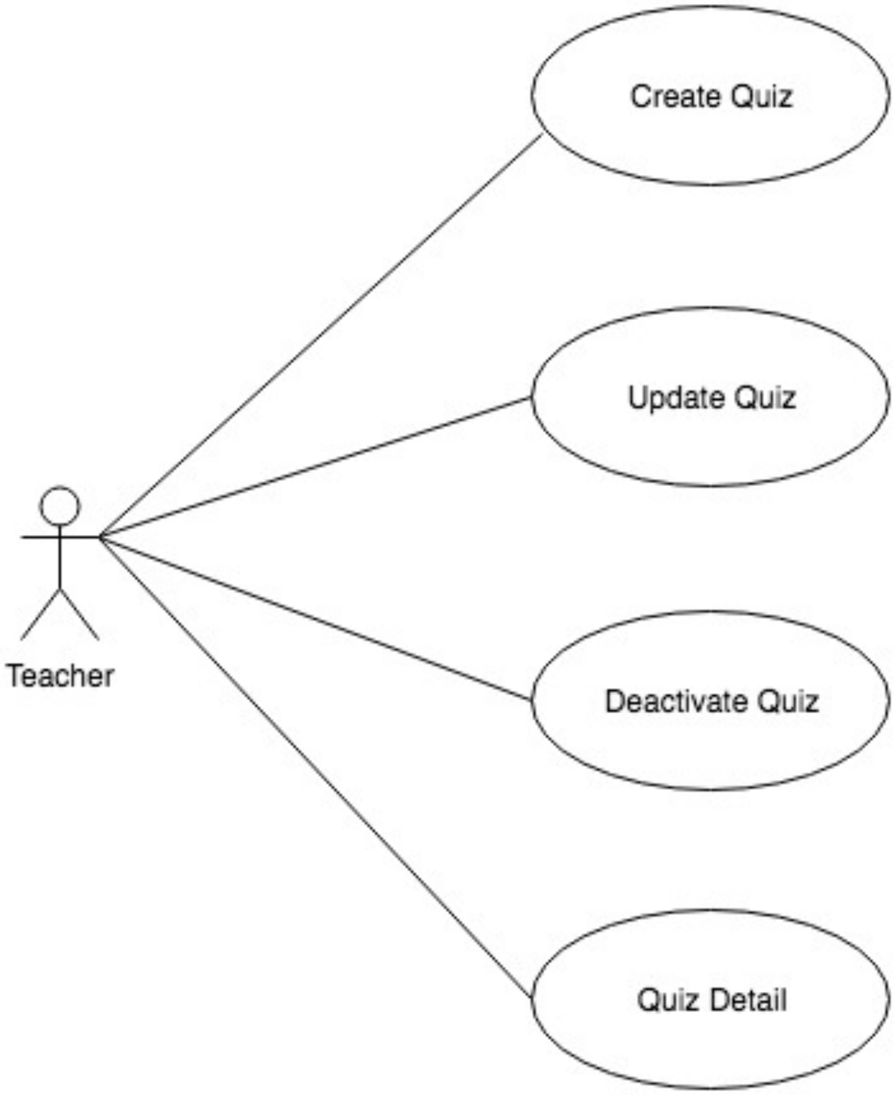
3. Lesson operations for teacher



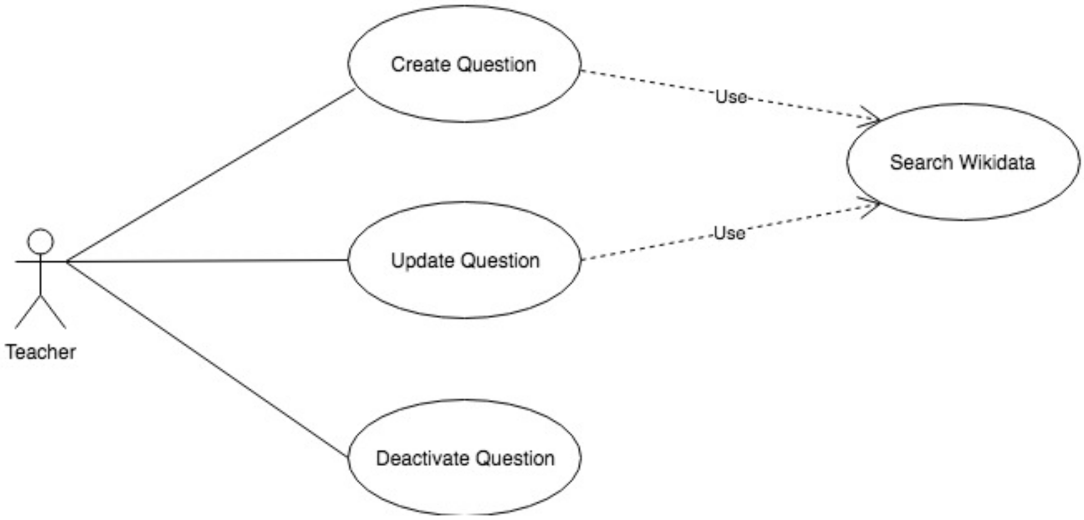
4. Content operation for teacher



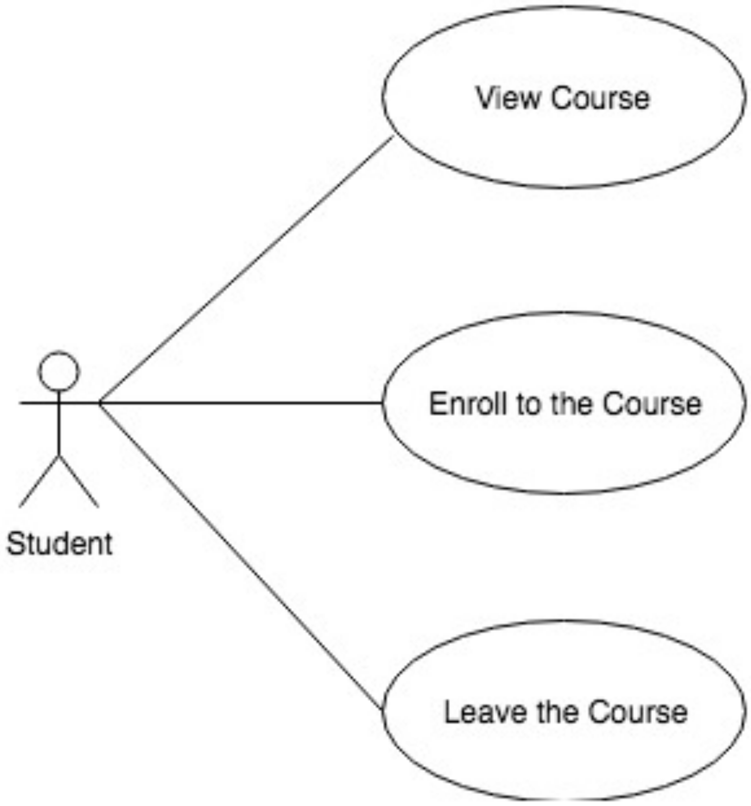
5. Quiz operation for teacher

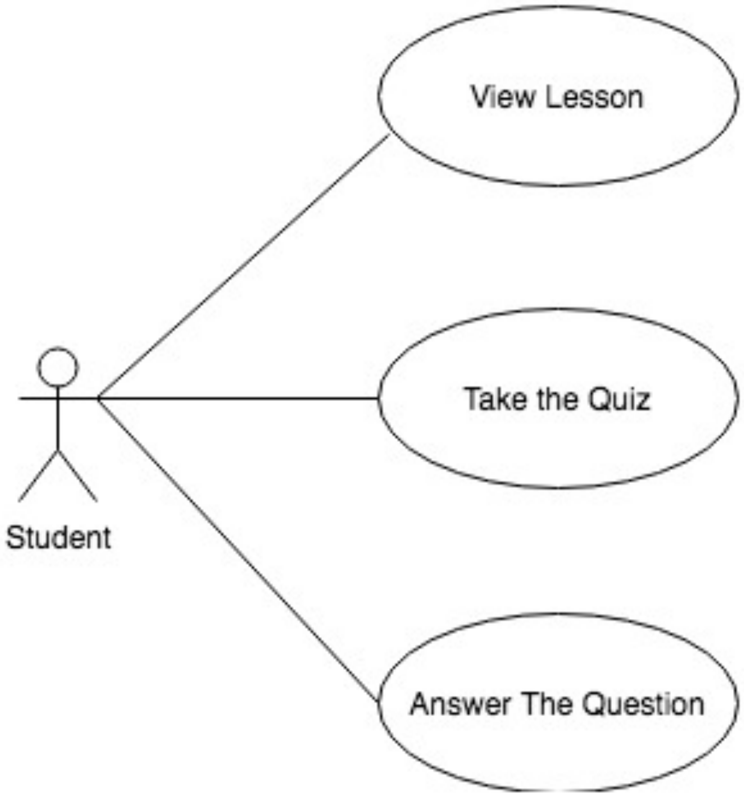


6. Question operations for teacher

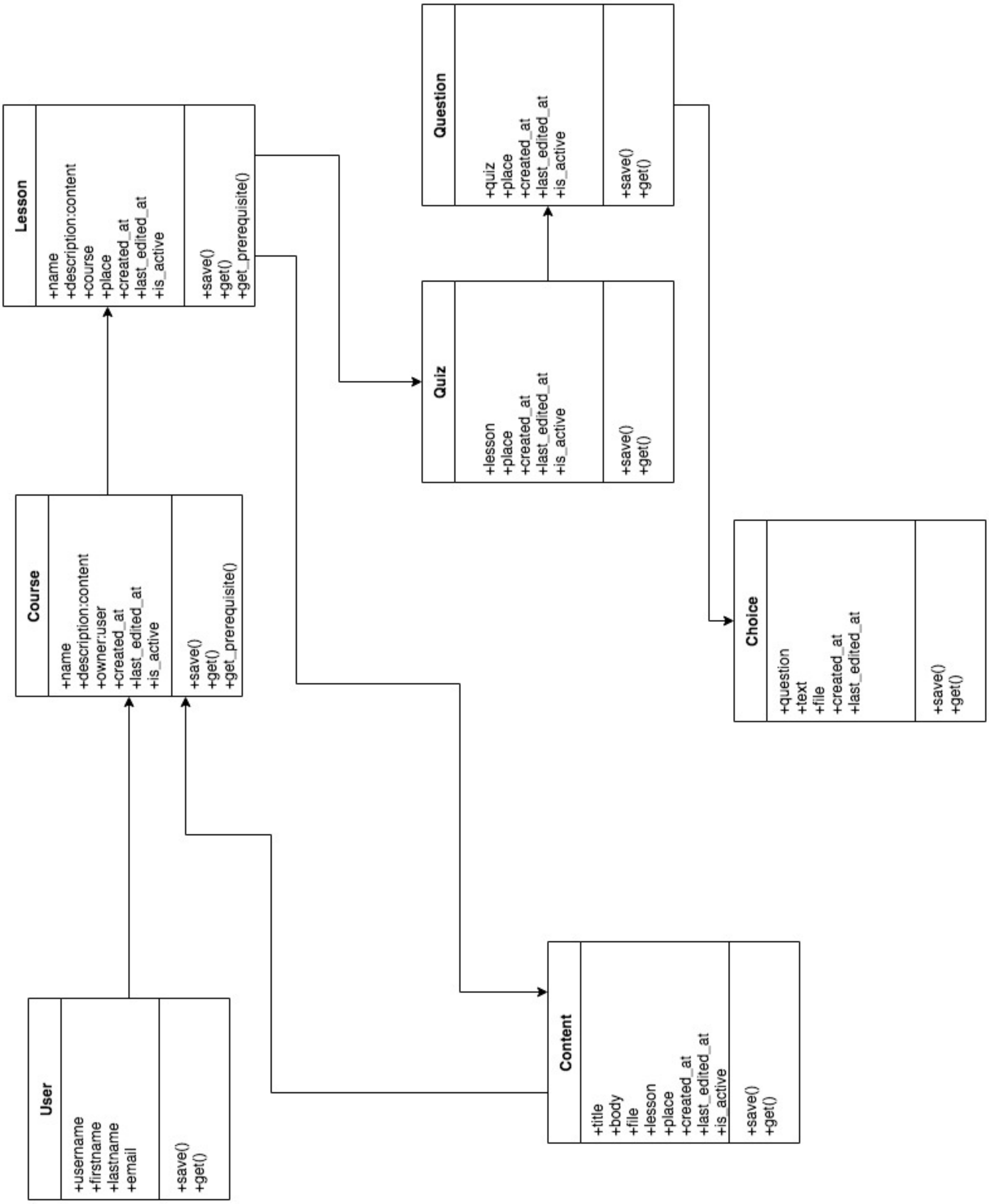


7. Student operations



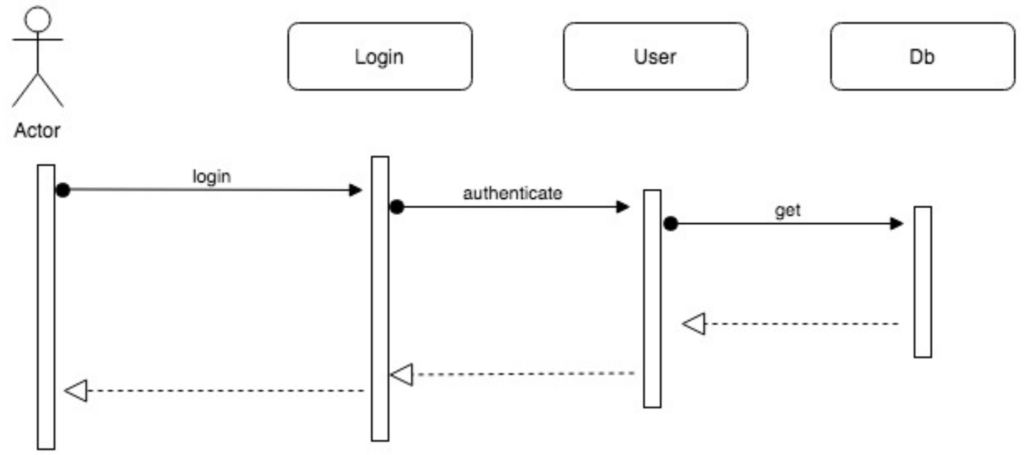


Class diagram

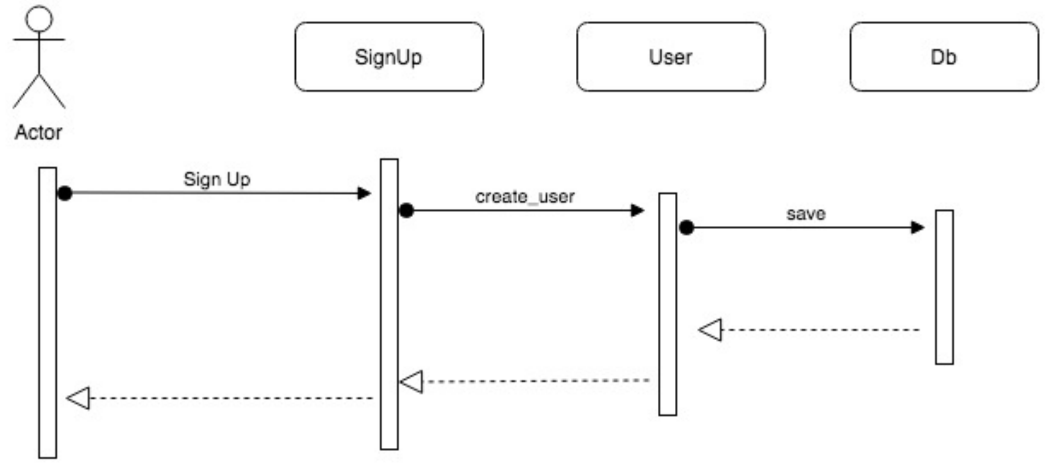


Sequence Diagrams

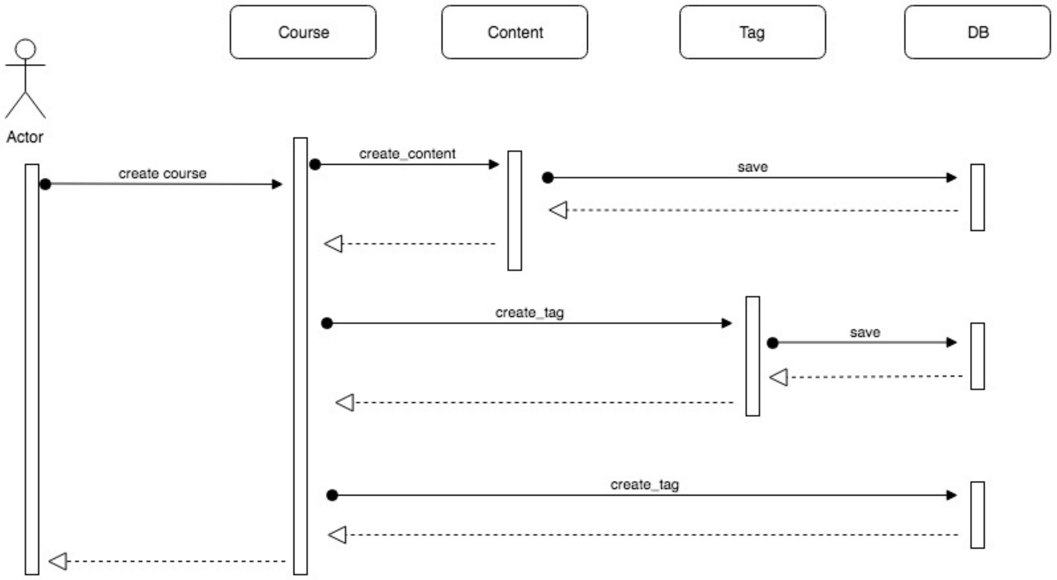
Login



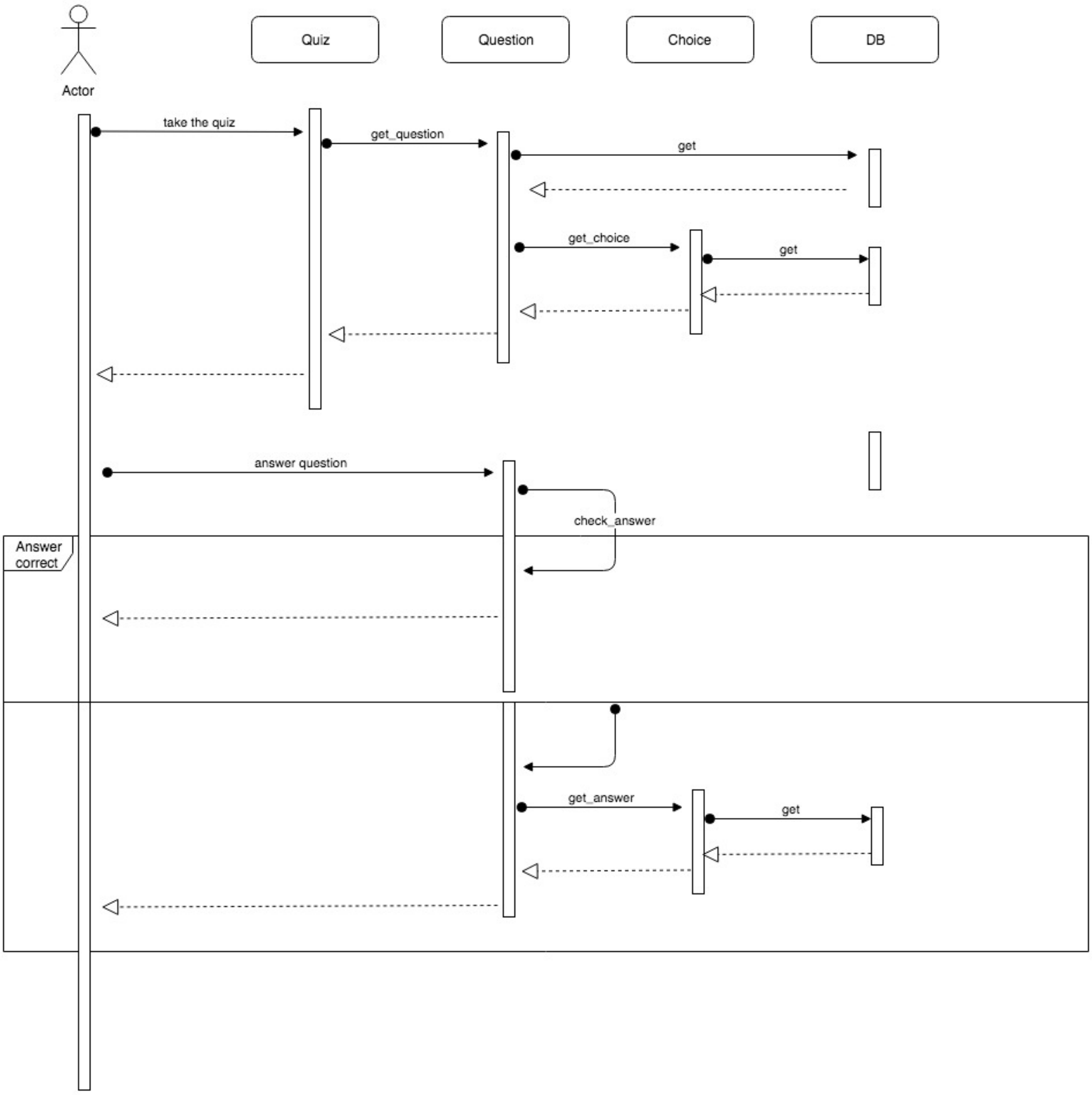
Sign Up



Create course



Answer a Question



5. Implementation

Our implementation process was divided into two parts: frontend and backend. For frontend, we used ReactJs and for backend we used Python's Django framework.

ReactJs is a javascript library for building user interfaces. With ReactJs it becomes easier develop interactive user interfaces. You can design simple view that will display a state of your app. The view is rendered automatically every time there is a change in state. You can design components that have their own state. Once you design them you can use them anywhere you want. ReactJs is an open source managed by Facebook. It is a very popular UI development tool and the community that uses it keeps growing.

Although we decided to use ReactJs for our project we didn't have any previous knowledge or experience with it. Because of that there was a training period where we learned ReactJs and how to use efficiently. The training took longer than expected which resulted in our development plan to be delayed for more than two weeks.

Django is python framework that is used for high-level web development. Django uses MTV design pattern. Normally in Django you can create a full-fledged web site without needing any other web development libraries but in this project, we used Django for developing our RESTful API. We used Django REST Framework to accomplish this task. Django REST framework is a toolkit for developing APIs.

For database, we decided to use PostreSql since it is open source and free. We didn't do any configuration on the PostgreSql. Django enables migration in database as long as database configurations are done correctly.

In our project, we take database information as environment variables. After you set up your PostgreSql database, all you have to is set the necessary environment variables and you are good to go. Further information will be provided in deployment section.

The original plan was to create APIs for a function then design UI with ReactJs and connect them to complete task but since ReactJs training took longer than expected we developed APIs first then start to the UI which caused some of the APIs to be designed wrongly and we were only able to realize them when we started UI implementation which caused the project to delay further.

6. Deployment

The original plan was to deploy the project as Docker containers. But since we were unfamiliar with Docker and there was not enough time for another training session we decided to go with a normal deployment where we deploy our code to a server and do configuration ourselves.

We used Google cloud for our deployment site. And we developed the project on a OSX operating system. Although we didn't test our system on a Windows OS, it should work just fine on windows to.

Now we going to start to explain steps necessary for deploying our system.

1. Secure a server

We used a Google cloud compute engine to create a virtual machine. The machine's OS is Debian GNU/Linux 9 (stretch), it has core vCPU, 3.75 GB memory and 10 GB storage capacity. The system should work on any server with any operating system that supports python.

2. Clone git repo

Install git if it is not installed already. Run the fallowing codes:

```
git clone https://github.com/kasimbozdag/SWE_573.git
cd SWE_573/
```

3. Setting database environment variables.

Assuming you have a postgresql database setup, you have set the database configuration to the environment as variable so that API can reach the database. Follow these steps to set environment variables.

```
$ touch ~/.bash_proflle
$ vi ~/.bash_proflle
```

add following content to the document change the necessary arguments

```
export POSTGRES_HOST="localhost"
export POSTGRES_PORT="5432"
export POSTGRES_DB_NAME='SWE'
export POSTGRES_USERNAME="kasim"
export POSTGRES_PASSWORD="*******"
```

save the file and exit then run

```
$ source ~/.bash_profile
```

now your environment variables are set.

4. Deploy the API

The API was develop using *python 3.7* and was deploy with *python 3.5.3* it should work with 3.5.*, 3.6.* and 3.7. when you're in project main folder(SWE_573) run the following command codes to deploy the API. We will assume you want to run the API as a background service so it does not stop when you close the terminal.

```
$ cd API
$ python manage.py migrate
$ python -m venv Swe_venv
$ source Swe_venv/bin/activate
$ pip install -r requirement.txt
$ mkdir /var/log/apis/swe.live.log
$ python manage.py 0.0.0.0:8000 --settings=API.settings.dev &>> /var/log/apis/swe.live.log &
```

now you're API is running in background and it can be accessed from 8000 port of the server from outside. You have to open the port from firewall settings to allow access.

5. Deploy the UI

Before starting to deploy the UI, you need change API configuration in the ReactJs. Go to project folder (SWE_573)

```
$ cd frontend
$ vi src/redux/services.js
```

change line 3 API_URL should be equal to API uri there is a "/" at end of the uri. Save and exit

```
$ vi src/redux/configureStore.js
```

change line 19 API_URL should be equal to API Uri there is not a "/" at end of the Uri. Save and exit
Now you can build the ReactJs project.

```
$ npm run build
```

this will create a build folder now you can move the contents of this folder to anywhere you can serve a static html file.

I serve it on the same server so I did the following

```
$ cp -r build/* /var/www/html
```

This allowed my apache server to serve the project.

7. Known Issues

In this section, we will talk about shortcomings of our project and what more we could have done.

One of the biggest shortcomings of this project was we could not keep to the project plan. There were difficulties with learning new technologies, time was restrained with other projects. Because we could not keep with the time line there were a lot of features there were originally in the project requirements that we could not do.

Requirements 1.2.15 and 1.2.16 were left out of the project because we could not complete. The requirements about statistic (2.2), only some of them were done. The search for content, bookmark and reference modal was never started (1.2.5,1.2.6,1.2.7). The pre-requisite for courses, lessons and quizzes were design in the backend and APIs were written for it but there was not enough time to implement on the UI. While we created the learning path(Course) and there was a system in place in the backend to track if it was completed we could not implement on the UI.

The UI needed a lot more work, we done it in a way that will show the functionality. There was not a lot of thought put in to design. This was because we were inexperienced with both UI and ReactJs.

There were planned Unit tests since we could not finish the project the tests were all together left out.

The original deployment plan was to make the project into Docker images so the deployment will be easier but since it required us to learn Docker from scratch and time was an issue, it was abandoned.

What we learn from this project were:

Requirements should be made clearer at the beginning of project.

The research for the tool to be used should be done more thoroughly.

When determining training time, a longer period should be used.

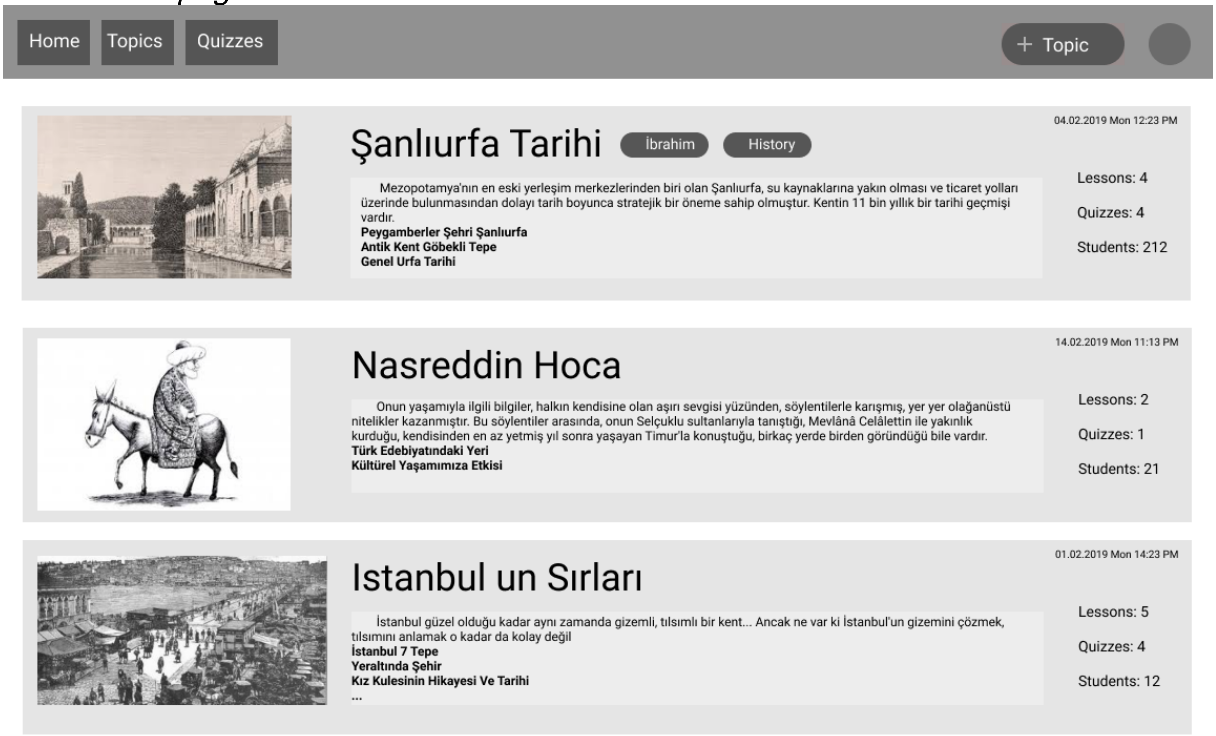
The time table should be followed if not the stakeholder should be made aware and steps to remedy it should be taken.

Test should develop as the implementation progresses.

Deployment environment should be prepared from the beginning and the code should be deployed as developed.

8. Mock-ups

Main page



Şanlıurfa Tarihi İbrahim History

04.02.2019 Mon 12:23 PM

Mezopotamya'nın en eski yerleşim merkezlerinden biri olan Şanlıurfa, su kaynaklarına yakın olması ve ticaret yolları üzerinde bulunmasından dolayı tarih boyunca stratejik bir öneme sahip olmuştur. Kentin 11 bin yıllık bir tarihi geçmişi vardır.
Peygamberler Şehri Şanlıurfa
Antik Kent Göbekli Tepe
Genel Urfa Tarihi

Lessons: 4
Quizzes: 4
Students: 212

Nasreddin Hoca

14.02.2019 Mon 11:13 PM

Onun yaşamıyla ilgili bilgiler, halkın kendisine olan aşırı sevgisi yüzünden, söylentilerle karışmış, yer yer olağanüstü nitelikler kazanmıştır. Bu söylentiler arasında, onun Selçuklu sultanlarıyla tanıştığı, Mevlânâ Celâlettin ile yakınlık kurduğu, kendisinden en az yetmiş yıl sonra yaşayan Timur'la konuştuğu, birkaç yerde birden göründüğü bile vardır.
Türk Edebiyatındaki Yeri
Kültürel Yaşamımıza Etkisi

Lessons: 2
Quizzes: 1
Students: 21

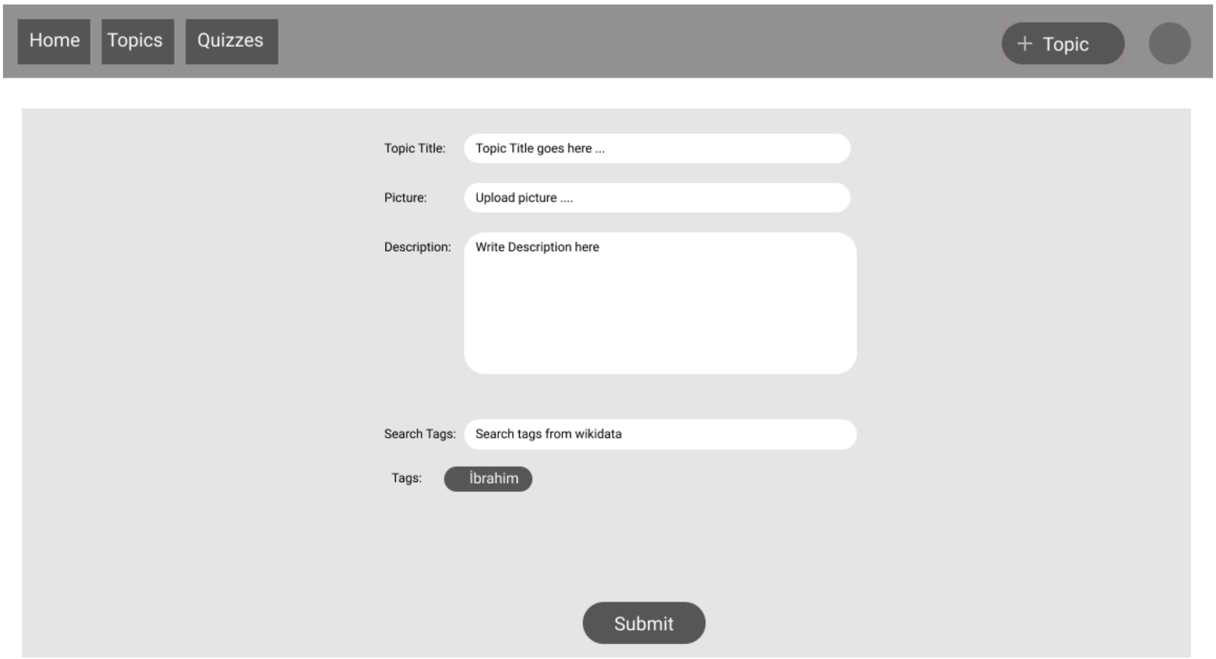
Istanbul un Sırları

01.02.2019 Mon 14:23 PM

İstanbul güzel olduğu kadar aynı zamanda gizemli, tılsımlı bir kent... Ancak ne var ki İstanbul'un gizemini çözmek, tılsımını anlamak o kadar da kolay değil
İstanbul 7 Tepe
Yeraltında Şehir
Kız Kulesinin Hikayesi Ve Tarihi
...

Lessons: 5
Quizzes: 4
Students: 12

Create course



Topic Title: Topic Title goes here ...

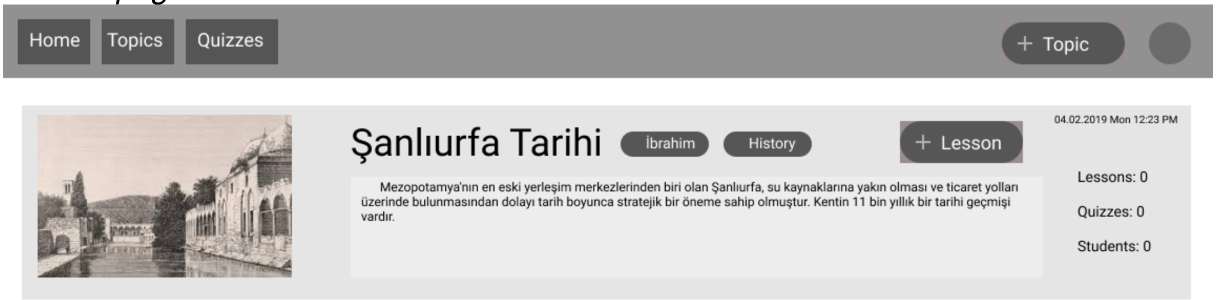
Picture: Upload picture ....

Description: Write Description here

Search Tags: Search tags from wikidata

Tags: İbrahim

Course page after creation



Şanlıurfa Tarihi

04.02.2019 Mon 12:23 PM

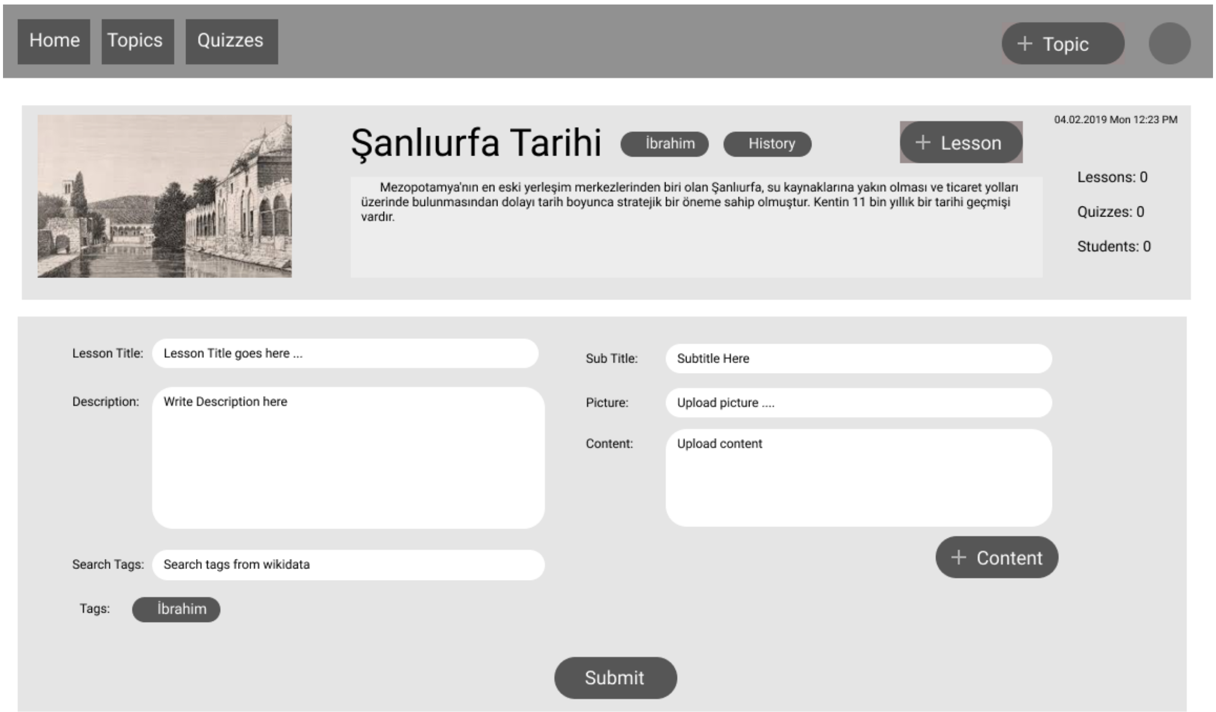
Mezopotamya'nın en eski yerleşim merkezlerinden biri olan Şanlıurfa, su kaynaklarına yakın olması ve ticaret yolları üzerinde bulunmasından dolayı tarih boyunca stratejik bir öneme sahip olmuştur. Kentin 11 bin yıllık bir tarihi geçmişi vardır.

Lessons: 0

Quizzes: 0

Students: 0

Add lesson



Şanlıurfa Tarihi

04.02.2019 Mon 12:23 PM

Mezopotamya'nın en eski yerleşim merkezlerinden biri olan Şanlıurfa, su kaynaklarına yakın olması ve ticaret yolları üzerinde bulunmasından dolayı tarih boyunca stratejik bir öneme sahip olmuştur. Kentin 11 bin yıllık bir tarihi geçmişi vardır.

Lessons: 0

Quizzes: 0

Students: 0

Lesson Title: Lesson Title goes here ...

Description: Write Description here

Search Tags: Search tags from wikidata

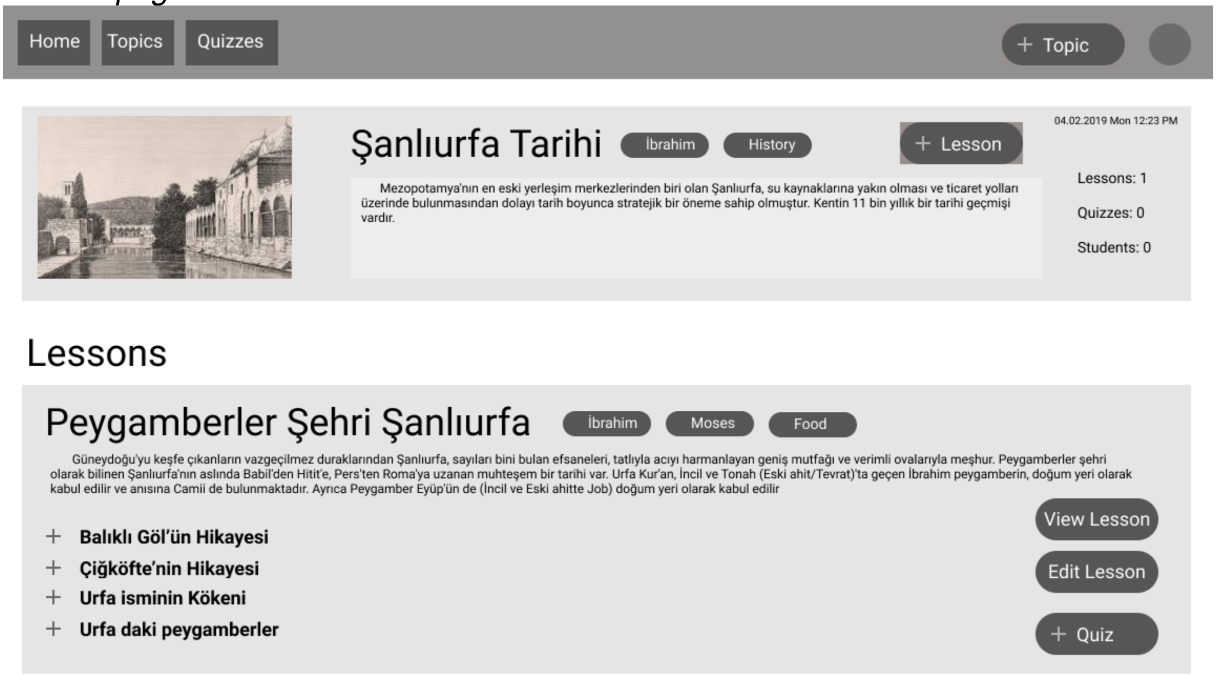
Tags: İbrahim

Sub Title: Subtitle Here

Picture: Upload picture ....

Content: Upload content

Course page after lessons created



Şanlıurfa Tarihi

04.02.2019 Mon 12:23 PM

Mezopotamya'nın en eski yerleşim merkezlerinden biri olan Şanlıurfa, su kaynaklarına yakın olması ve ticaret yolları üzerinde bulunmasından dolayı tarih boyunca stratejik bir öneme sahip olmuştur. Kentin 11 bin yıllık bir tarihi geçmişi vardır.

Lessons: 1

Quizzes: 0

Students: 0

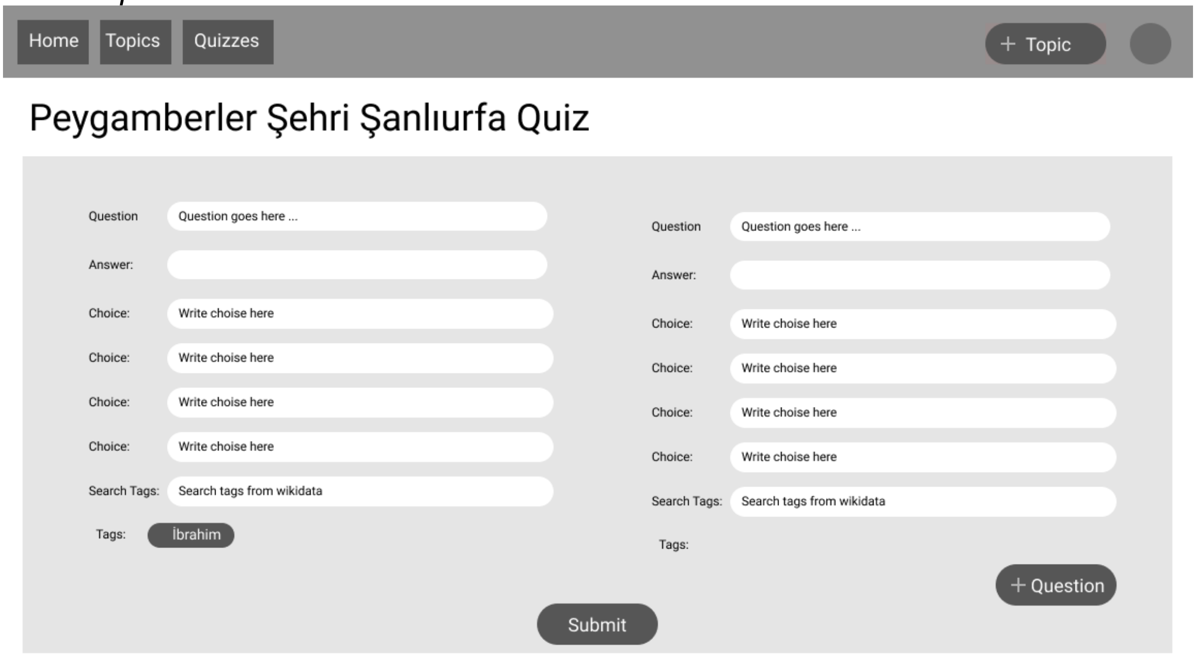
Lessons

Peygamberler Şehri Şanlıurfa

Güneydoğu'yu keşfe çıkanların vazgeçilmez duraklarından Şanlıurfa, sayıları bini bulan efsaneleri, tatlıyla acıyı harmanlayan geniş mutfağı ve verimli ovalarıyla meşhur. Peygamberler şehri olarak bilinen Şanlıurfa'nın aslında Babil'den Hitit'e, Pers'ten Roma'ya uzanan muhteşem bir tarihi var. Urfa Kur'an, İncil ve Tonah (Eski ahit/Tevrat)'ta geçen İbrahim peygamberin, doğum yeri olarak kabul edilir ve anısına Camii de bulunmaktadır. Ayrıca Peygamber Eyüp'ün de (İncil ve Eski ahitte Job) doğum yeri olarak kabul edilir

+ Balıklı Göl'ün Hikayesi
+ Çiğköfte'nin Hikayesi
+ Urfa isminin Kökeni
+ Urfa daki peygamberler

Create quiz

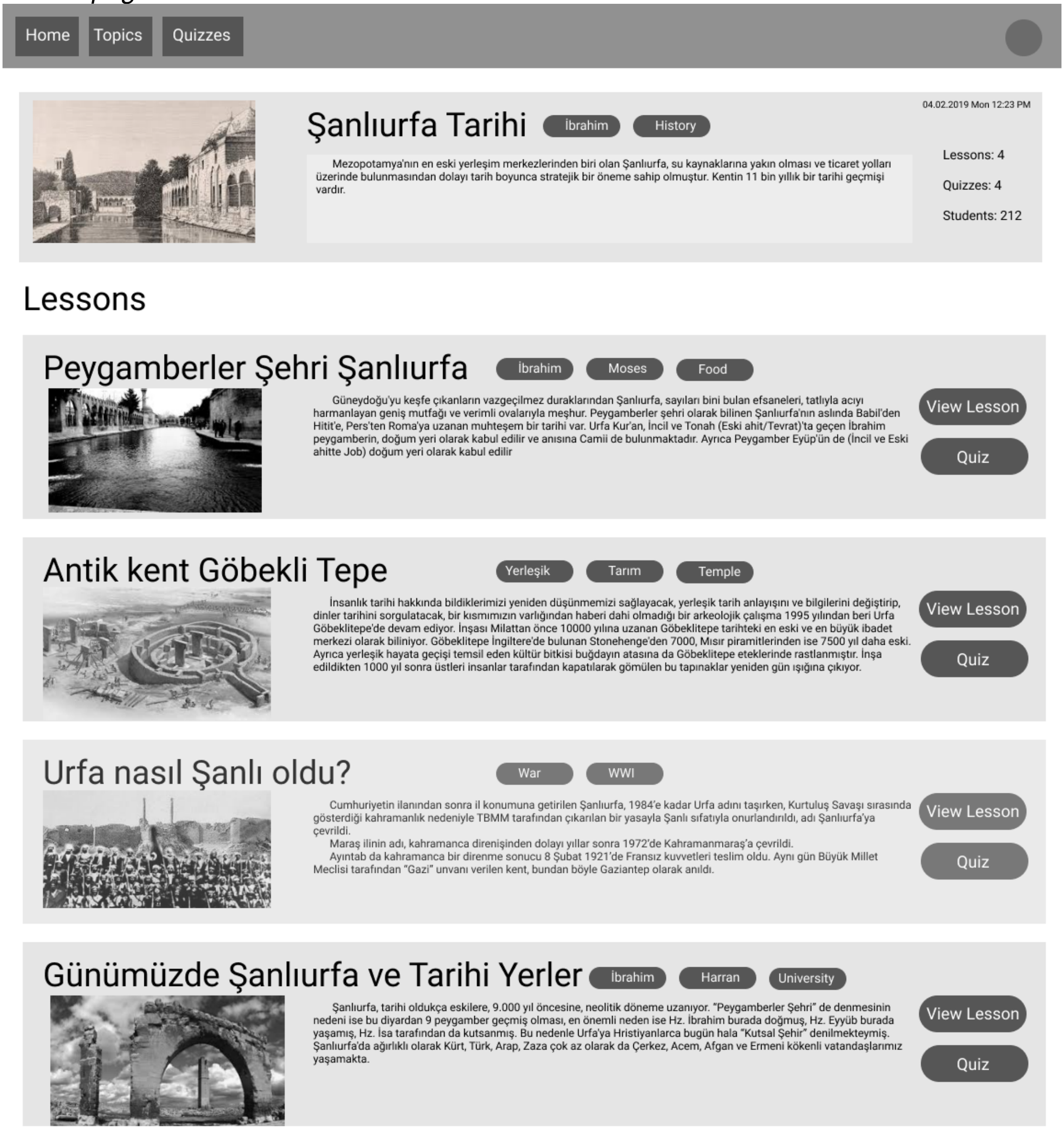


Peygamberler Şehri Şanlıurfa Quiz

Question: Question goes here ...
Answer:
Choice: Write choise here
Choice: Write choise here
Choice: Write choise here
Choice: Write choise here
Search Tags: Search tags from wikidata
Tags: İbrahim

Question: Question goes here ...
Answer:
Choice: Write choise here
Choice: Write choise here
Choice: Write choise here
Choice: Write choise here
Search Tags: Search tags from wikidata
Tags:

course page for students



Şanlıurfa Tarihi

04.02.2019 Mon 12:23 PM

Mezopotamya'nın en eski yerleşim merkezlerinden biri olan Şanlıurfa, su kaynaklarına yakın olması ve ticaret yolları üzerinde bulunmasından dolayı tarih boyunca stratejik bir öneme sahip olmuştur. Kentin 11 bin yıllık bir tarihi geçmişi vardır.

Lessons: 4

Quizzes: 4

Students: 212

Lessons

Peygamberler Şehri Şanlıurfa

Güneydoğu'yu keşfe çıkanların vazgeçilmez duraklarından Şanlıurfa, sayıları bini bulan efsaneleri, tatlıyla acıyı harmanlayan geniş mutfağı ve verimli ovalarıyla meşhur. Peygamberler şehri olarak bilinen Şanlıurfa'nın aslında Babil'den Hitit'e, Pers'ten Roma'ya uzanan muhteşem bir tarihi var. Urfa Kur'an, İncil ve Tonah (Eski ahit/Tevrat)'ta geçen İbrahim peygamberin, doğum yeri olarak kabul edilir ve anısına Camii de bulunmaktadır. Ayrıca Peygamber Eyüp'ün de (İncil ve Eski ahitte Job) doğum yeri olarak kabul edilir

Antik kent Göbekli Tepe

İnsanlık tarihi hakkında bildiklerimizi yeniden düşünmemizi sağlayacak, yerleşik tarih anlayışını ve bilgilerini değiştirip, dinler tarihini sorgulatacak, bir kısmımızın varlığından haberi dahi olmadığı bir arkeolojik çalışma 1995 yılından beri Urfa Göbeklitepe'de devam ediyor. İnşası Milattan önce 10000 yılına uzanan Göbeklitepe tarihteki en eski ve en büyük ibadet merkezi olarak biliniyor. Göbeklitepe İngiltere'de bulunan Stonehenge'den 7000, Mısır piramitlerinden ise 7500 yıl daha eski. Ayrıca yerleşik hayata geçişi temsil eden kültür bitkisi buğdayın atasına da Göbeklitepe eteklerinde rastlanmıştır. İnşa edildikten 1000 yıl sonra üstleri insanlar tarafından kapatılarak gömülen bu tapınaklar yeniden gün ışığına çıkıyor.

Urfa nasıl Şanlı oldu?

Cumhuriyetin ilanından sonra il konumuna getirilen Şanlıurfa, 1984'e kadar Urfa adını taşırken, Kurtuluş Savaşı sırasında gösterdiği kahramanlık nedeniyle TBMM tarafından çıkarılan bir yasayla Şanlı sıfatıyla onurlandırıldı, adı Şanlıurfa'ya çevrildi.

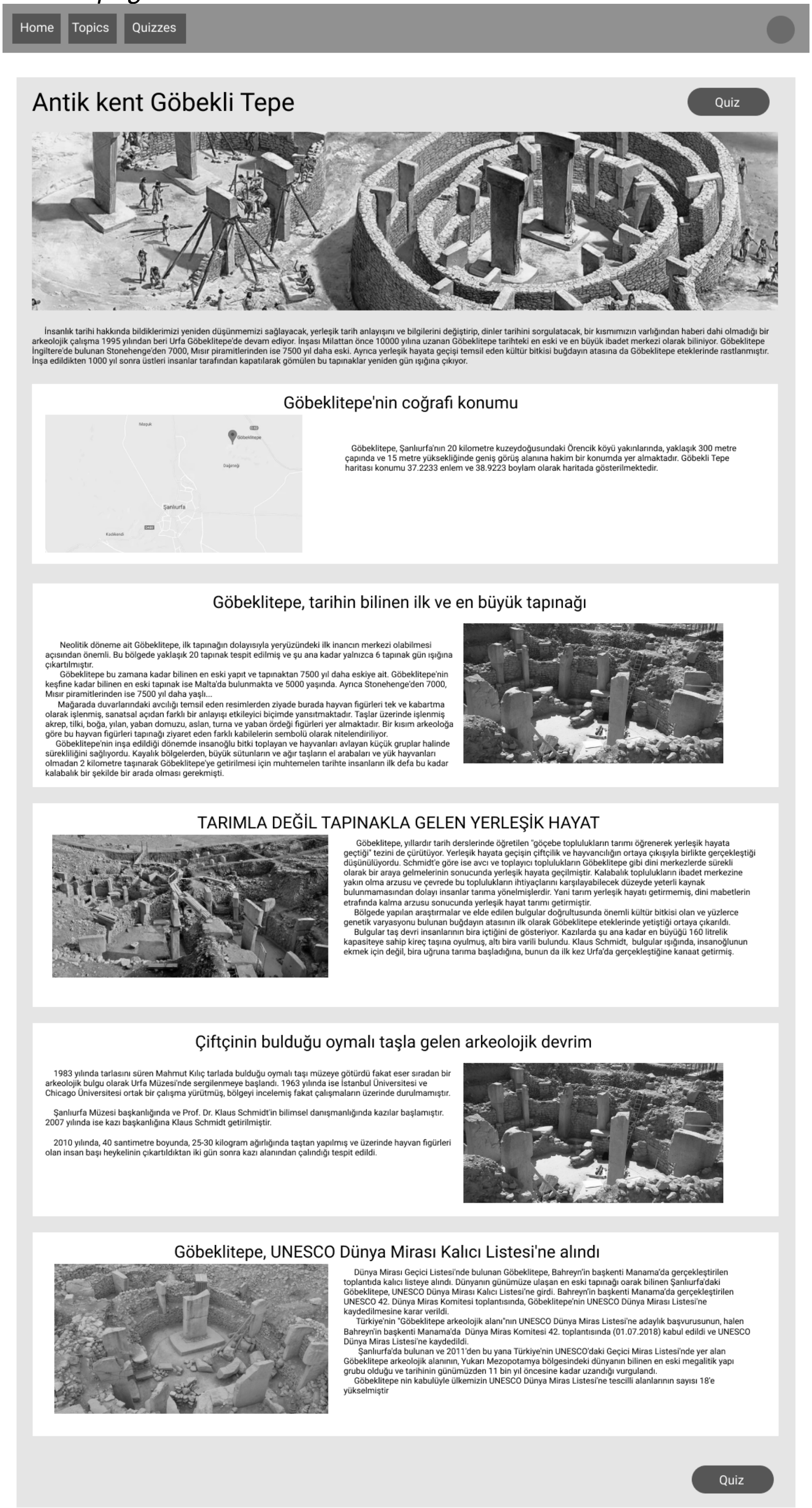
Maraş ilinin adı, kahramanca direnişinden dolayı yıllar sonra 1972'de Kahramanmaraş'a çevrildi.

Ayıntab da kahramanca bir direnme sonucu 8 Şubat 1921'de Fransız kuvvetleri teslim oldu. Aynı gün Büyük Millet Meclisi tarafından "Gazi" unvanı verilen kent, bundan böyle Gaziantep olarak anıldı.

Günümüzde Şanlıurfa ve Tarihi Yerler

Şanlıurfa, tarihi oldukça eskilere, 9.000 yıl öncesine, neolitik döneme uzanıyor. "Peygamberler Şehri" de denmesinin nedeni ise bu diyardan 9 peygamber geçmiş olması, en önemli neden ise Hz. İbrahim burada doğmuş, Hz. Eyyüb burada yaşamış, Hz. İsa tarafından da kutsanmış. Bu nedenle Urfa'ya Hristiyanlarca bugün hala "Kutsal Şehir" denilmekteymiş. Şanlıurfa'da ağırlıklı olarak Kürt, Türk, Arap, Zaza çok az olarak da Çerkez, Acem, Afgan ve Ermeni kökenli vatandaşlarımız yaşamakta.

Lesson page



Antik kent Göbekli Tepe

İnsanlık tarihi hakkında bildiklerimizi yeniden düşünmemizi sağlayacak, yerleşik tarih anlayışını ve bilgilerini değiştirip, dinler tarihini sorgulatacak, bir kısmımızın varlığından haberi dahi olmadığı bir arkeolojik çalışma 1995 yılından beri Urfa Göbeklitepe'de devam ediyor. İnşası Milattan önce 10000 yılına uzanan Göbeklitepe tarihteki en eski ve en büyük ibadet merkezi olarak biliniyor. Göbeklitepe İngiltere'de bulunan Stonehenge'den 7000, Mısır piramitlerinden ise 7500 yıl daha eski. Ayrıca yerleşik hayata geçişi temsil eden kültür bitkisi buğdayın atasına da Göbeklitepe eteklerinde rastlanmıştır. İnşa edildikten 1000 yıl sonra üstleri insanlar tarafından kapatılarak gömülen bu tapınaklar yeniden gün ışığına çıkıyor.

Göbeklitepe'nin coğrafi konumu

Göbeklitepe, Şanlıurfa'nın 20 kilometre kuzeydoğusundaki Örencik köyü yakınlarında, yaklaşık 300 metre çapında ve 15 metre yüksekliğinde geniş görüş alanına hakim bir konumda yer almaktadır. Göbekli Tepe haritası konumu 37.2233 enlem ve 38.9223 boylam olarak haritada gösterilmektedir.

Göbeklitepe, tarihin bilinen ilk ve en büyük tapınağı

Neolitik döneme ait Göbeklitepe, ilk tapınağın dolayısıyla yeryüzündeki ilk inancın merkezi olabilmesi açısından önemli. Bu bölgede yaklaşık 20 tapınak tespit edilmiş ve şu ana kadar yalnızca 6 tapınak gün ışığına çıkartılmıştır.

Göbeklitepe bu zamana kadar bilinen en eski yapıt ve tapınaktan 7500 yıl daha eskiye ait. Göbeklitepe'nin keşfine kadar bilinen en eski tapınak ise Malta'da bulunmakta ve 5000 yaşında. Ayrıca Stonehenge'den 7000, Mısır piramitlerinden ise 7500 yıl daha yaşlı...

Mağarada duvarlarındaki avcılığı temsil eden resimlerden ziyade burada hayvan figürleri tek ve kabartma olarak işlenmiş, sanatsal açıdan farklı bir anlayışı etkileyici biçimde yansıtmaktadır. Taşlar üzerinde işlenmiş akrep, tilki, boğa, yılan, yaban domuzu, aslan, turna ve yaban ördeği figürleri yer almaktadır. Bir kısım arkeoloğa göre bu hayvan figürleri tapınağı ziyaret eden farklı kabilelerin sembolü olarak nitelendiriliyor.

Göbeklitepe'nin inşa edildiği dönemde insanoğlu bitki toplayan ve hayvanları avlayan küçük gruplar halinde sürekliliğini sağlıyordu. Kayalık bölgelerden, büyük sütunların ve ağır taşların el arabaları ve yük hayvanları olmadan 2 kilometre taşınarak Göbeklitepe'ye getirilmesi için muhtemelen tarihte insanların ilk defa bu kadar kalabalık bir şekilde bir arada olması gerekmişti.

TARIMLA DEĞİL TAPINAKLA GELEN YERLEŞİK HAYAT

Göbeklitepe, yıllardır tarih derslerinde öğretilen "göçebe toplulukların tarımı öğrenerek yerleşik hayata geçtiği" tezini de çürütüyor. Yerleşik hayata geçişin çiftçilik ve hayvancılığın ortaya çıkışıyla birlikte gerçekleştiği düşünülüyordu. Schmidt'e göre ise avcı ve toplayıcı toplulukların Göbeklitepe gibi dini merkezlerde sürekli olarak bir araya gelmelerinin sonucunda yerleşik hayata geçilmiştir. Kalabalık toplulukların ibadet merkezine yakın olma arzusu ve çevrede bu toplulukların ihtiyaçlarını karşılayabilecek düzeyde yeterli kaynak bulunmamasından dolayı insanlar tarıma yönelmişlerdir. Yani tarım yerleşik hayatı getirmemiş, dini mabetlerin etrafında kalma arzusu sonucunda yerleşik hayat tarımı getirmiştir.

Bölgede yapılan araştırmalar ve elde edilen bulgular doğrultusunda önemli kültür bitkisi olan ve yüzlerce genetik varyasyonu bulunan buğdayın atasının ilk olarak Göbeklitepe eteklerinde yetiştiği ortaya çıkarıldı.

Bulgular taş devri insanlarının bira içtiğini de gösteriyor. Kazılarda şu ana kadar en büyüğü 160 litrelik kapasiteye sahip kireç taşına oyulmuş, altı bira varili bulundu. Klaus Schmidt, bulgular ışığında, insanoğlunun ekmek için değil, bira uğruna tarıma başladığına, bunun da ilk kez Urfa'da gerçekleştiğine kanaat getirmiş.

Çiftçinin bulduğu oymalı taşla gelen arkeolojik devrim

1983 yılında tarlasını süren Mahmut Kılıç tarlada bulduğu oymalı taşı müzeye götürdü fakat eser sıradan bir arkeolojik bulgu olarak Urfa Müzesi'nde sergilenmeye başlandı. 1963 yılında ise İstanbul Üniversitesi ve Chicago Üniversitesi ortak bir çalışma yürütmüş, bölgeyi incelemiş fakat çalışmaların üzerinde durulmamıştır.

Şanlıurfa Müzesi başkanlığında ve Prof. Dr. Klaus Schmidt'in bilimsel danışmanlığında kazılar başlamıştır. 2007 yılında ise kazı başkanlığına Klaus Schmidt getirilmiştir.

2010 yılında, 40 santimetre boyunda, 25-30 kilogram ağırlığında taştan yapılmış ve üzerinde hayvan figürleri olan insan başı heykelinin çıkartıldıktan iki gün sonra kazı alanından çalındığı tespit edildi.

Göbeklitepe, UNESCO Dünya Mirası Kalıcı Listesi'ne alındı

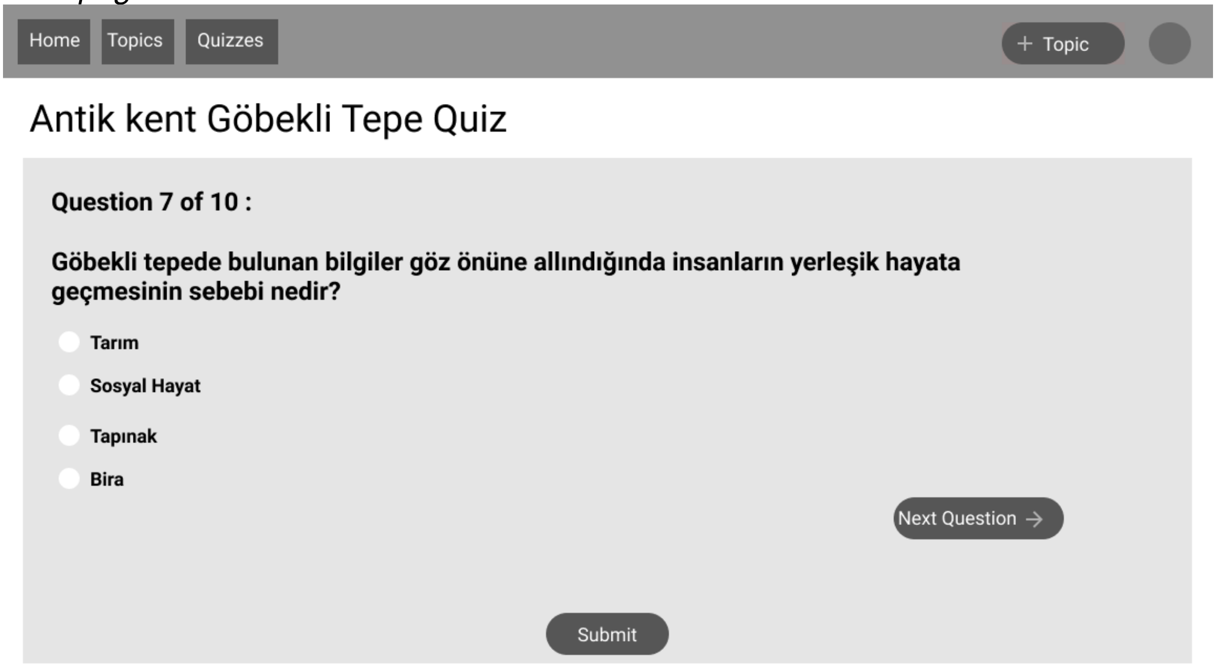
Dünya Mirası Geçici Listesi'nde bulunan Göbeklitepe, Bahreyn'in başkenti Manama'da gerçekleştirilen toplantıda kalıcı listeye alındı. Dünyanın günümüze ulaşan en eski tapınağı oarak bilinen Şanlıurfa'daki Göbeklitepe, UNESCO Dünya Mirası Kalıcı Listesi'ne girdi. Bahreyn'in başkenti Manama'da gerçekleştirilen UNESCO 42. Dünya Miras Komitesi toplantısında, Göbeklitepe'nin UNESCO Dünya Mirası Listesi'ne kaydedilmesine karar verildi.

Türkiye'nin "Göbeklitepe arkeolojik alanı"nın UNESCO Dünya Miras Listesi'ne adaylık başvurusunun, halen Bahreyn'in başkenti Manama'da Dünya Miras Komitesi 42. toplantısında (01.07.2018) kabul edildi ve UNESCO Dünya Miras Listesi'ne kaydedildi.

Şanlıurfa'da bulunan ve 2011'den bu yana Türkiye'nin UNESCO'daki Geçici Miras Listesi'nde yer alan Göbeklitepe arkeolojik alanının, Yukarı Mezopotamya bölgesindeki dünyanın bilinen en eski megalitik yapı grubu olduğu ve tarihinin günümüzden 11 bin yıl öncesine kadar uzandığı vurgulandı.

Göbeklitepe nin kabulüyle ülkemizin UNESCO Dünya Miras Listesi'ne tescilli alanlarının sayısı 18'e yükselmiştir

Quiz page for student



Antik kent Göbekli Tepe Quiz

Question 7 of 10 :

Göbekli tepede bulunan bilgiler göz önüne allındığında insanların yerleşik hayata geçmesinin sebebi nedir?

- ○ **Tarım**
- ○ **Sosyal Hayat**
- ○ **Tapınak**
- ○ **Bira**