1- Executive Summary

We have reached the milestone objectives which were:

- Running the backend api in amazon cloud server.
- Sign in, sign up, sign out, authorization, and item creation in web.
- Sign in, sign up, sign out in android.

We plan to add our other models to the backend api in the following two weeks and then we will focus on recommendations and search system.
Web-frontend and android will work in parallel with backend.

2- List and status of deliverables

- Milestone report
- Requirements
- Project Plan
- DB Diagram in https://github.com/bounswe/bounswe2017group1/wiki/Database-Model
- Web Frontend Screen Shots in https://github.com/bounswe/bounswe2017group1/wiki/Milestone-1-Frontend-ScreenShots
- Android Screen Shots in https://github.com/bounswe/bounswe2017group1/wiki/Milestone-1-Android-ScreenShots

- Api documentation
    - https://github.com/bounswe/bounswe2017group1/wiki/API-Documentation
    - https://github.com/bounswe/bounswe2017group1/blob/web/web/api/README.md

3- Summary of coding work done by each team member

Mehmet Sefa Balık Team: Android	• Creation of android project base code, implementation of sign-up Activity and UI.	• Implemented Retrofit API Interface, POJO classes for Retrofit.
Taha Metin Bayi	• Android Environment Setup	

Team: Android		
Yiğit Alperen Bildik Team: Backend	● First creation of Get-Post functions of Heritage Item at the backend.	
Mustafa Feyzioğlu Team: Android	● Implementation of login activity and UI	
Mehmet Gülşen Team: Backend	● Creation of the database model.	● Worked on user serializer and signup function.
Abdullah Furkan Ilısu Team: Backend	● Established the project design	● Worked on serializers and API functions
Ali Kireçligöl Team: Frontend	● Implementation of sign up and sign in web pages.	● Implematation of create and display heritage item web pages and home page
Hakan Şirin (Communicator) Team: Backend	● Implemented backend for heritage item controllers from where my fellow group mate Ceyhun left.	● Created profile model controller. ● Some customization in url redirection to prevent possible errors.
Ceyhun Uzunoğlu Team: Backend	● First creation of Get-Post functions of Heritage Item at the backend.	● Code review and test for Heritage Item's further Get-Post function developments.
Basri Yılmaztürk Team: Frontend	● Implementation of sign up and sign in web pages.	● Implematation of create and display heritage item web pages and home page

4- Requirements Achieved In Milestone 1

Following requirements includes only a subset of full requirements which is on https://github.com/bounswe/bounswe2017group1/wiki/Requirements

Glossary

* 1 Cultural Heritage Item: An item with description that somewhat connected to culture and heritage.

* 2 Annotations: Any further info regarding an any cultural heritage item as elaboration, translation, meta information, question, and answer.

* 3 Tag: predefined words that helps categorize and search heritage items and searches.

* 4 Meta Information: Informations that defines the cultural heritage item as title, geo-tag tag, creator and so forth.

* 5 User:

* 5.1 Guest user: Any user that visits the website without any registration who can view but not edit any heritage items ,annotations and comments.

* 5.2 Registered User: Users that passed the login process and can edit or create annotation and heritage items and commends in addition to guest user privileges.

* 6 Score: Difference between the number of upvotes and downvotes of a heritage item.

1.Functional

1.1 Users

1.1.1 Guest Users

1.1.1.2 Guest users shall be able to register and become a registered user.

1.1.1.2.1 Guest users shall be able to register via their email address and a username.

1.1.1.3 Guest users shall be able to view the content of a cultural heritage item.

1.1.2 Registered Users

1.1.2.1 Registered users shall be able to create cultural heritage items.

1.1.2.3 Registered users shall be able to edit and delete the cultural heritage items and annotations that they have created.

1.1.2.5 Registered users shall be able to update their profile.

1.1.2.5.1 Registered users shall be able to change their password.

1.1.2.5.2 Registered users shall be able to change their email address.

1.1.2.6 Registered users shall be log in with their username and password.

1.1.2.10 Registered users shall be able to view the content of a cultural heritage item.

1.2.Heritage Items

1.2.1 Heritage items shall have a description.

1.2.2.1 Heritage items shall have creation date of the heritage item object.

1.2.2.2 Heritage items shall support a date representation of the time that heritage item actually came into existence.

1.2.2.2.1 It can be an exact date.

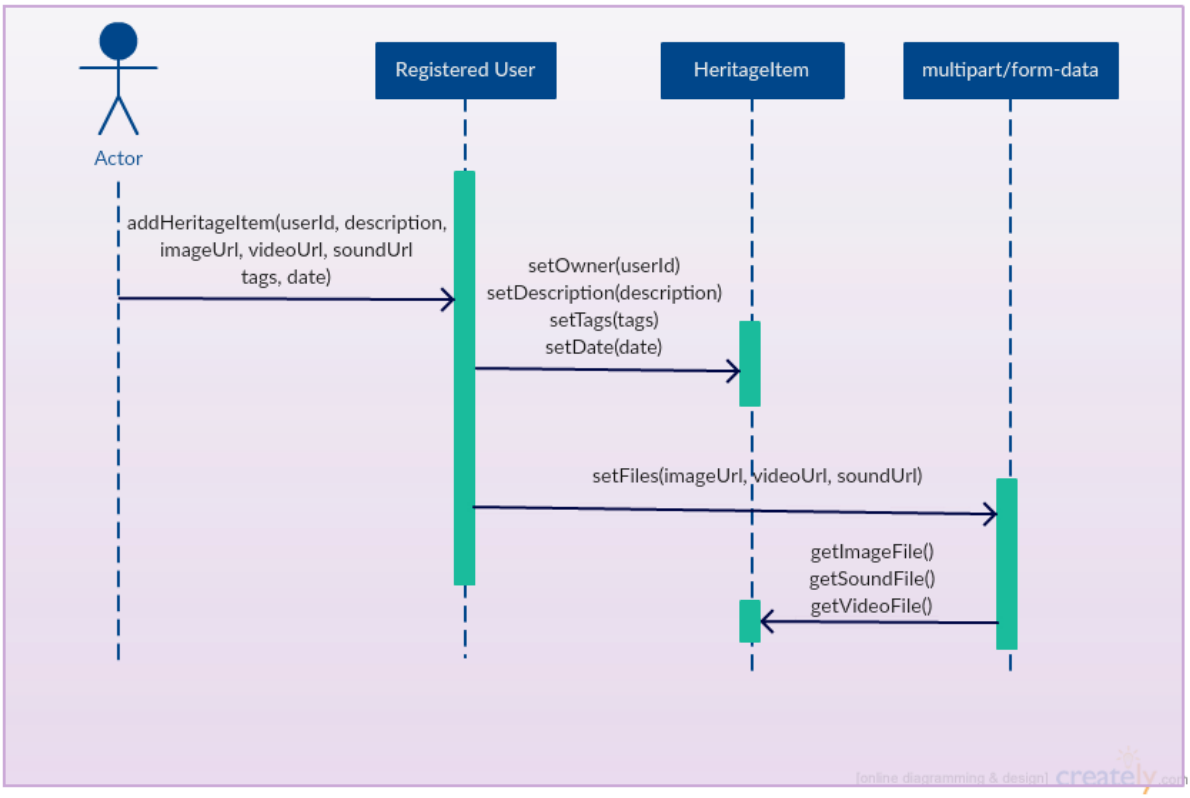
1.2.2.2.2 It can be duration between two dates.

1.2.2.3 Heritage items shall have a title.

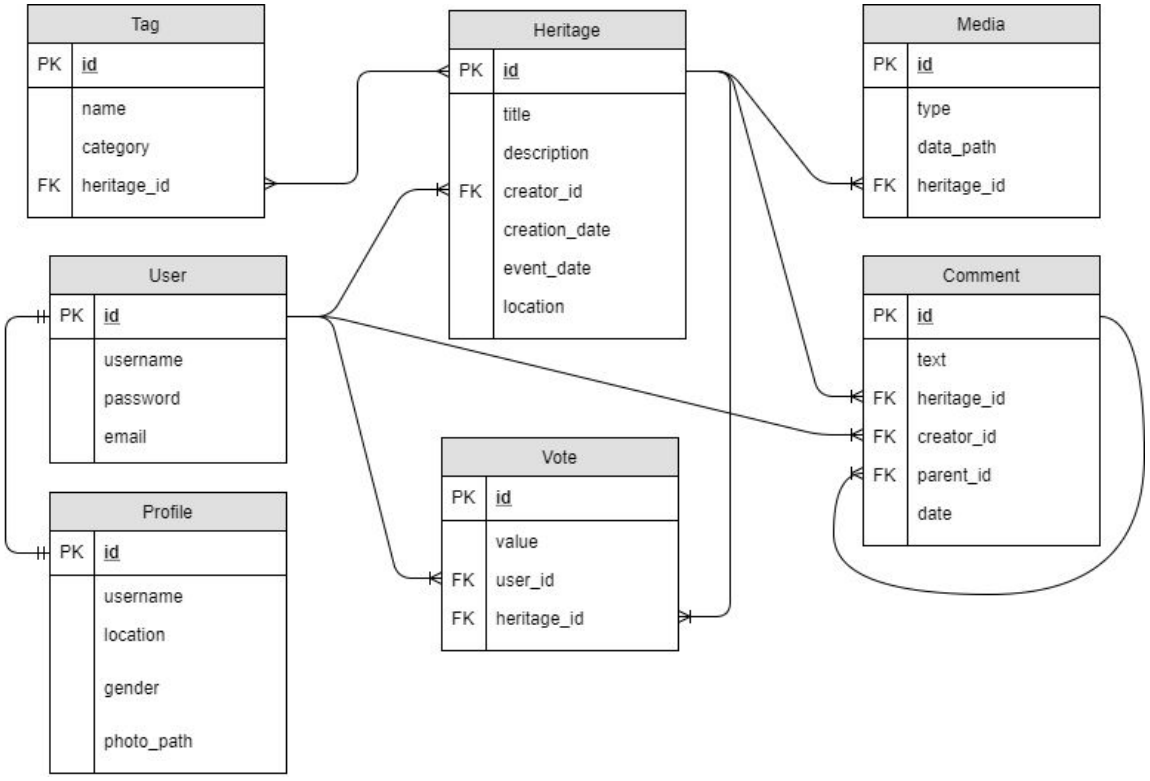
5- Design

Sequence Diagram

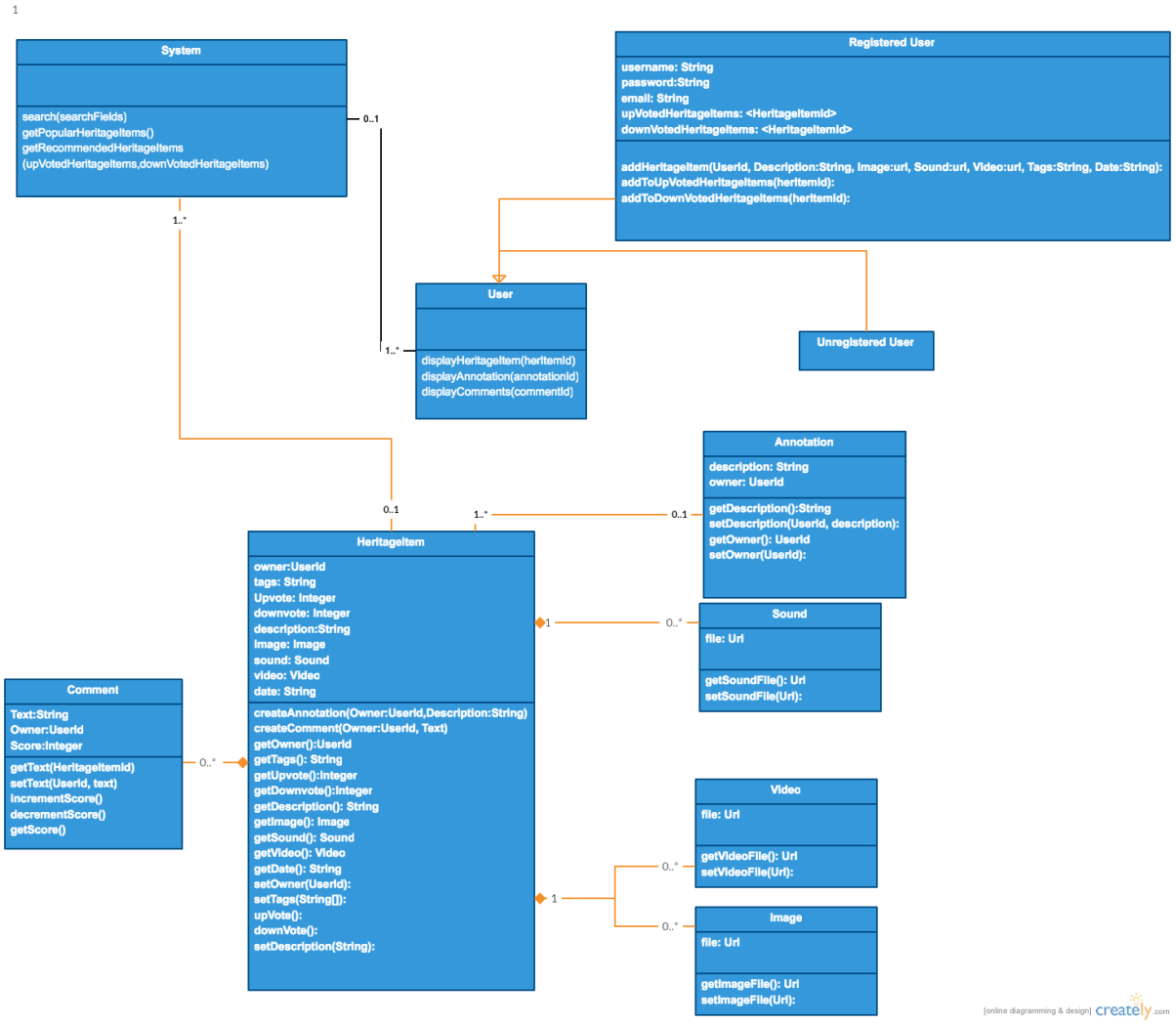
Sequence Diagram for adding a heritage item is as shown.



DataBase Diagram



Class Diagram:



Django Backend Design

We create db objects in models folder. Implement their serializers in serializer folder and controllers that handles and return the requests in controller folder for each model. If we need some extra functions in the controllers we have service folder and related model`s services in it.

6- Project plan

Time Period	Task Name	Assignee
24.09.2017 - 01.10.2017	Mockup Review	Mehmet Gülşen, Mustafa Feyzioğlu, Mehmet Sefa Balık
24.09.2017 - 1.10.2017	Project Plan Review	Hakan Şirin
24.09.2017 - 1.10.2017	Scenario Review, Requirement Review	Everyone
1.10.2017 - 8.10.2017	Web Environment setup	Frontend and Backend teams
1.10.2017 - 8.10.2017	Android Environment setup	Mehmet Sefa Balık, Mustafa Feyzioğlu, Taha Bayi
8.10.2017 - 22.10.2017	Sign up/Sing in Backend Web	Hakan Şirin, Mehmet Gülşen, Furkan İlısu
8.10.2017 - 22.10.2017	Sign up/Sing in Frontend Web	Basri Yılmaztürk, Ali Kireçligöl
22.10.2017 - 26.10.2017	Sign up/Sing in Android	Mehmet Sefa Balık, Mustafa Feyzioğlu
22.10.2017 - 26.10.2017	Heritage ItemBackend	Yiğit Alp Bildik, Ceyhun Uzunoğlu
22.10.2017 - 26.10.2017	Heritage Item Frontend	Basri Yılmaztürk, Ali Kireçligöl

7- Evaluation of tools and managing the project

At the project development process we used hangouts for team meetings. For code reviews and code related communications we used github, and also whatsapp for general communication. Hangout meetings were productive and very useful for solving problems that relates all team members. Because of the fact that the problem of being in school and having no course at that hour at the same is so difficult for all of us, hangout was a useful solution for us. We use Github as much as possible. This project is first team project of most of us. For that reason, we were struggled with the tools of Github and Git (pull requests, issues, comments, merges, branches, lots of branches) at the beginning, but we eventually overcome this hard period with little flesh wounds. And also, although it is an unprofessional way of software development communication, we used Whatsapp a lot. We will work on that in the following weeks.

In terms of branch usage in git, what we did was creating branches based on features from the web branch and then creating pull request returning to once again web branch when the feature is completed. Adding db to git caused great trouble since it is impossible to merge binary SQLite objects but lesson learned there. We are surely not gonna add db files to git in the future.

Another lesson learned is never skip weekly meetings. It was a really challenging Milestone week because we skipped the meeting in the previous week.