CmpE451 Fall 2017/2018

CUSTOMER MILESTONE#1 REPORT

InterestHub

Group 6

Alperen Yakut

Barış Uçaktürk

Burak Tomruk

Emre Eren

Gökçe Uludoğan

Hakan Kalaycı

Merve Cerit

Onur Orhan

Yusufcan Manav

Table of Contents

1 Executive Summary

For the first milestone, our objectives were implementing user authentication, group and content services as well as creating the nonfunctional pages for frontend and android applications, supported by a backend structure. The list of what we did so far is as follows:

- Creation and listing of groups in web
- Creation and listing of posts in web
- Nonfunctional pages of home, user profile, creation of template in web
- Deployment of both backend api and web application
- Listing Groups in Android
- Creating groups in Android
- Nonfunctional user profile page
- Nonfunctional user timeline page and view of posts in Android.
- Partially functional login and signup page in Android

2 List and status of deliverables

2.1 Front-end

Deliverables	**Status**
Group timeline page	Functional
Group creation page	Functional
Post creation in group-timeline page	Functional
User Timeline	Partially functional
Guest Timeline	Partially functional
Sign up and Login Page	Non-functional
Template Creation	Non-functional
Profile Page	Non-functional

Evaluation of status of the deliverables

Status of the deliverables were as planned mostly, so there is no major drawback in the project plan. The only thing we couldn’t achieve was implementing functionalities of the sign up and login but that will be sorted out until the second milestone. That can create a small extra workload for us but it is manageable to finish this for the second milestone and carry out the plan.

2.2 Android

Deliverables	**Status**
Synchronization of the app with the front-end when a group is added user should be able to see it from app too	Delivered, Functional
Extras	
Login-Signup Page	Functional except auth
Splash Page	Functional
User Page	Partly functional
User Timeline	Partly functional
Group Page	Partly functional
Group Creation	Functional

Evaluation of status of the deliverables

As Android team, we are ahead of the expectations of the customer. We are planning to finish and test our app earlier to satisfy our customer more than she wants.

3 Summary of coding work done by each team member

Alperen Yakut	Creating image and video components Creating corresponding end poinds serializers and views Writing view for user groups
Barış Uçaktürk	Implementation of Login Activity(Android) Implementation of Sign Up Activity(Android) Changing Button, EditText and TextView via creating .xml file Checking just user and e-mail data which come from backend when user tries to log in.
Burak Tomruk	Implementation of group timeline page Implementation of guest timeline page Adding calendars to user and group timeline pages for event scheduling Adding group image url in group creation and showing on group list and timeline with using service.
Emre Eren	Initial setup and creation of application structure (Android) Setting up Retrofit network package (Android) Implementation of User Activity Implementation of Group and Profile page fragments inside User activity.
Gökçe Uludoğan	Initial setup of AngularJS application Implementation of nonfunctional home and user profile page Implementation of server for frontend part Implementation of group, post and user services for getting data from backend and showing them in the frontend Deployment of front end application
Hakan Kalaycı	Initial setup of Django backend application. Implementation of models according to prepared class diagram.

	Implementation of endpoints that belongs to group, content, component of content and relation between group and content. Deployment of backend application to aws.
Merve Cerit	Android - Implementation of Splash Page, Design of the app, Logo creation and embedding it to app, Creation of group creation button, Changing the default layout design of the app
Onur Orhan	Created the ‘components’ app in the project directory. Added text fields to ‘components’, and a description field to ‘group’. Updated corresponding serializer and view files. Migration
Yusufcan Manav	Implementation of group creation page. Implementation of the template creation page (non-functional). Implementing post creation functionality from group timeline.

4 Requirements

One can find the requirements below. Completed requirements are highlighted in red.

Glossary

- System: The software that can be used via the web and mobile applications.
- Guest: A user of the system that has not authenticated themselves via logging in.
- User: A registered user of the system that has logged in (for short notation, user refers to registered users, and guest refers to unauthenticated users).
- Profile Page: A user page that contains general information about them, depending on how much they want to share. Has fields for name, occupation, company/school, date of birth, interested tags, and additional personal information.
- Personal Feed Page: A user page that the system provides to the user so that they can see recent activity of their groups and other users that they follow.
- Follow: To track other users' activities.
- Group: The interest group that users create and join to share content about a relevant interest.
- Post: Any content shared in the group.
- Poll: A special kind of post that contains a question and multiple choices. This can be interacted with by using one (or more, if allowed) of the choices.
- Event: A special kind of post, referring to a real-world event, that has a specified time and place:

- Comment: User responses to posts, or other previous comments.
- Upvote/Downvote: User feedback on posts that measure whether they like the content (upvote), or think it is irrelevant (downvote).
- Admin: The general moderator of a group. The first admin is the founder of the group, who can share this right with other users.
- Flag: The activity that a user undertakes when they feel a post or comment constitutes ill behaviour or is against the group rules.
- FAQ/Rules Section: A part of the group, which is always displayed, that contains general rules regarding posts, and/or frequently asked questions.
- Tag: A keyword that is assigned to content, so that content may be matched.
- Recommendation: The system's way of identifying relevant content that a certain user might find interesting.
- Video: A recording of moving pictures and sound, especially as a digital file, DVD, etc.
- Semantic: Semantics is the study of the meanings of words and phrases in a language.

1. Functional Requirements

1.1. User Requirements

- 1.1.1 General User Requirements
    - 1.1.1.1 Sign-up and Login
        - 1.1.1.1.1 People who want to be a member of application shall be able to create new account by providing their username, e-mail and password.
        - 1.1.1.1.2 People who want to be a member of application should be able to create new account by using their Google accounts or Facebook accounts.
        - 1.1.1.1.3 Users with an account shall be able to log in by entering their usernames/e-mails and their passwords.
        - 1.1.1.1.4 Users forgetting their passwords or usernames should be able to reset their passwords or to learn their usernames via their e-mail addresses.
    - 1.1.1.2 User Activity

- 1.1.1.2.1 Users shall be able to edit their personal pages.
- 1.1.1.2.2 Users shall be able to view other users' personal pages.
- 1.1.1.2.3 Users shall be able to create and join interest groups.
- 1.1.1.2.4 Users shall be able to search for information, members and groups.
- 1.1.1.2.5 Users shall be able to do advanced search.
  - 1.1.1.2.5.1 Users should be able to specify the group they are searching in, or choose to search the whole (public parts of) the system.
  - 1.1.1.2.5.2 Users should be able to filter results for a specific time interval (last hour, 8 hours, last day, last week, last month) in their search.
  - 1.1.1.2.5.3 Users should be able to filter results by whether the content they search for contained videos, photos, documents, or none.
  - 1.1.1.2.5.4 Users should be able to filter results by other users' names (whether that user shared or commented on the content).
  - 1.1.1.2.5.5 Users should be able to filter results by type (event or poll).
- 1.1.1.2.5 Users shall be able to follow other users.
  - 1.1.1.2.5.1 Users shall be able to see the public content shared by other users they follow, if the content is in a public group.
- 1.1.1.2.6 Users shall be able to search for a semantic tag. After searching, they shall be able to pick from the fetched list of the semantic tags.
- 1.1.1.2.7 Users should be able to specify what information is visible to others via privacy settings.
- 1.1.1.2.8 Users shall be able to create templates for their groups with options of text, multimedia(image,video,url), dropdown list, poll, radio button, check button and date.
- 1.1.1.2.9 Users shall be able to edit their own templates.

        - 1.1.1.3 Guest Activity
            - 1.1.1.3.1 Guests shall be able to use the search functionality for groups and people.
            - 1.1.1.3.2 Guests shall be able to view public groups.
            - 1.1.1.3.3 Guests shall be able to view public user profiles.
- 1.1.2 Group Requirements
    - 1.1.2.1 Group members shall have hierarchy levels.
        - 1.1.2.1.1 Groups shall have admins (see Admin Requirements below for details).
    - 1.1.2.2 Groups shall have privacy levels.
        - 1.1.2.2.1 Admins of a group shall be able to set privacy option.
        - 1.1.2.2.2 Admins shall be able to invite other users to private groups.
    - 1.1.2.3 Groups shall have specifications.
        - 1.1.2.3.1 Groups shall have a name.
        - 1.1.2.3.2 Groups should have a short description.
        - 1.1.2.3.3 Groups should have foundation date.
        - 1.1.2.3.4 Groups shall have an image, of the .jpg, .bmp or .png formats.
    - 1.1.2.4 Users shall be able to see the content of the group they are a part of.
    - 1.1.2.5 Groups shall have posts.
        - 1.1.2.5.1 Users shall be able to post messages in a group.
            - 1.1.2.5.1.1 Posts shall contain the post date and name of the owner of the post.
            - 1.1.2.5.1.2 Users shall be able to supplement their posts via videos linked from YouTube etc.
            - 1.1.2.5.1.3 Users shall be able to supplement their posts via images of .jpeg, .png, .gif, .bmp formats.
        - 1.1.2.5.2 Users shall be able to delete or edit the posts that they have shared.
        - 1.1.2.5.3 Users shall be able to create events in group.
            - 1.1.2.5.3.1 Users shall be able to specify the time of the event up to the dates, and hour(s) if necessary.

            - 1.1.2.5.3.2 Users shall be able to specify the place of the event.
            - 1.1.2.5.3.3 Users shall be able to create their own templates in order to specify the post they created.
        - 1.1.2.5.4 Users should be able to post polls.
            - 1.1.2.5.4.1 Users should be able to vote on polls.
            - 1.1.2.5.4.2 Users should be able to view the current results of the poll after they have voted.
        - 1.1.2.5.5 Users shall be able to interact with posts in groups.
            - 1.1.2.5.5.1 Users shall be able to upvote/downvote (or equivalent) posts.
            - 1.1.2.5.5.2 Users shall be able to comment on posts.
            - 1.1.2.5.5.3 Users shall be able to delete or edit their comments.
            - 1.1.2.5.5.4 Users shall be able to flag posts with ill content.
    - 1.1.2.7 Groups shall have a Rules/FAQ section.
- 1.1.3 Admin Requirements
    - 1.1.3.1 The founder of a group shall become the admin of the group.
    - 1.1.3.3 Admins shall be able to ban users permanently or temporarily from their respective groups.
    - 1.1.3.4 Admins shall be able to delete contents of other users.
    - 1.1.3.5 Admins shall be able to edit contents of other users.

1.2 System Requirements

- 1.2.1 Template Requirements
    - 1.2.1.1 The system shall be able to provide default templates for groups, events and posts.
    - 1.2.1.2 The system shall be able to show the templates based on popularity while creating posts, groups or events.
    - 1.2.1.3 The default template shall be able to include a text field for title of the post, a text field for the description of the post and an image attachment option.

    - 1.2.1.4 The system shall be able to give options to users for template creation. These options shall be text, multimedia(image,video,url), dropdown list, poll, radio button, check button and date.
- 1.2.2 Recommendation Requirements
    - 1.2.2.1 The recommendation system shall be able to relate similar tags and offer content based on tags.
    - 1.2.2.2 The recommendation system shall be able to analyze people based on the groups they are part of and suggest similar groups accordingly.
    - 1.2.2.3 The recommendation system shall be able to suggest similar people to follow each other based on their groups, the content they share and the events they attend.
    - 1.2.2.4 The recommendation system shall be able to offer events to users based on their past activities and profile.
    - 1.2.2.5 The recommendation system should be able to offer users other users, events and groups based on their location (e.g., most of the participants of a group might live in the same city).
- 1.2.3 Search Requirements
    - 1.2.3.1 The search system shall allow users to search for users, groups and tags.
    - 1.2.3.2 The search system shall work semantically over groups, posts and tags.
- 1.2.4 User Interface
    - 1.2.4.1 The system should provide a personal feed page for users, containing recent activity of their groups and followed users.
    - 1.2.4.2 The system shall list posts in groups in order of recent activity.
    - 1.2.4.3 The system shall provide admins with a special "Control Panel" page.
        - 1.2.4.3.1 The Control Panel page shall include flagged posts for ill behaviour.
        - 1.2.4.3.2 The Control Panel page shall allow admins to appoint other admins.
        - 1.2.4.3.3 The Control Panel page shall include a list of banned users.

        - 1.2.4.3.4 The Control Panel page shall allow admins to edit the Rules/FAQ section of the groups.

2. Non-functional Requirements

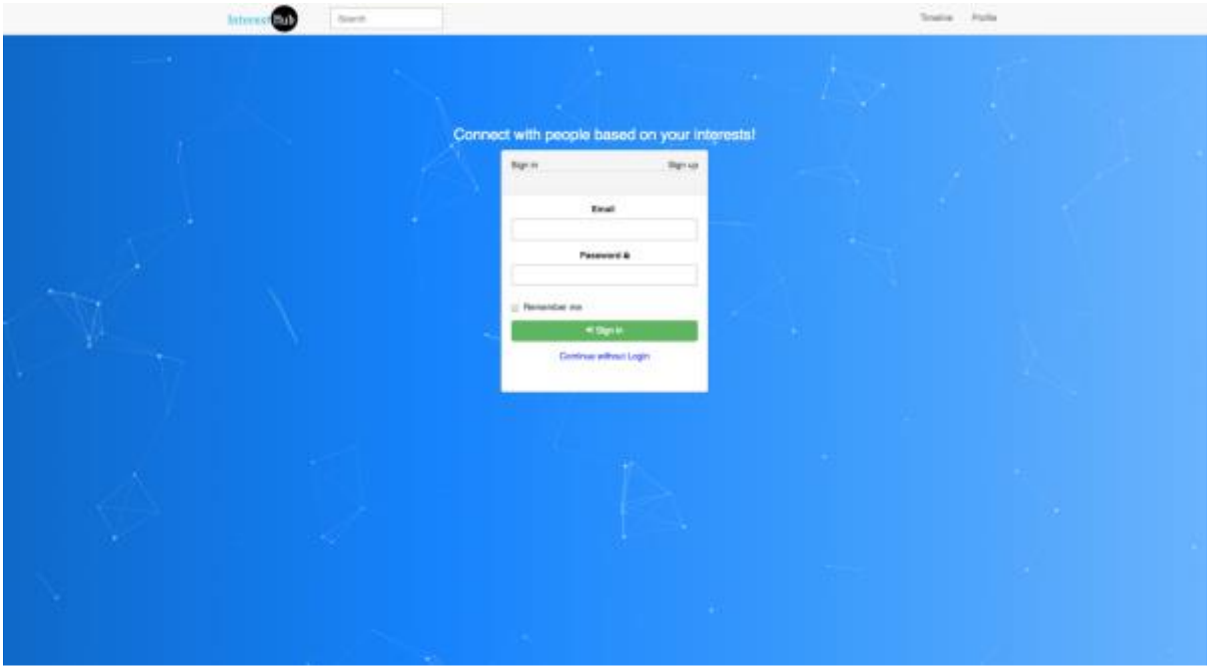
- 2.1 Availability
    - 2.1.1 The system should have a Web application which supports the latest version of Firefox (currently 53.0.0).
    - 2.1.2 The system should have an Android application which supports the latest version of Android operating system (currently Android 7 Nougat).
    - 2.1.3 A Firefox plugin should be created that enables users to create visual annotations.
        - 2.1.3.1 The annotations should follow the W3C Web Annotation Model standard.
- 2.2 Security
    - 2.2.1 The system should protect user information and passwords in the database via SHA-256 encryption.
    - 2.2.2 The system should use Google Captcha to verify users occasionally to prevent the website from spams and abuses.
    - 2.2.3 The system shall be able to support Hypertext Transfer Protocol.
- 2.3. Reliability and Back-ups
    - 2.3.1 The system shall be able to recover itself to its last known status after a crash within 5 minutes of coming online.
    - 2.3.2 The system shall do weekly full back-ups, and daily differential back-ups.
    - 2.3.3 The system shall support up to 1500 simultaneous users.
- 2.4 Performance
    - 2.4.1 The system shall respond to users in at most 5 seconds.
- 2.5 Maintainability
    - 2.5.1 The system shall support previous version for 6 months.
- 2.6 Database
    - 2.6.1 The database system shall have administrators.

- ○ 2.6.2 Database tables shall be able to change any time with error-checking to prevent error.

5 Design

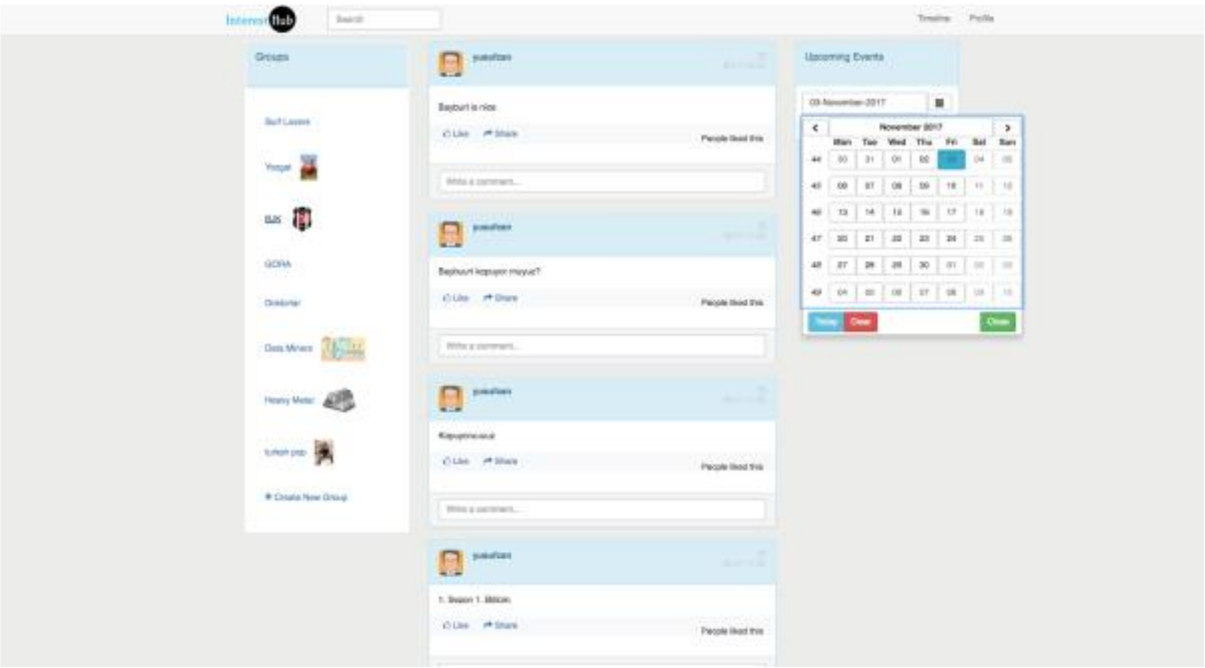
5.1 Web

5.1.1 Login Page



Connect with people based on your interests!

5.1.2 User Timeline



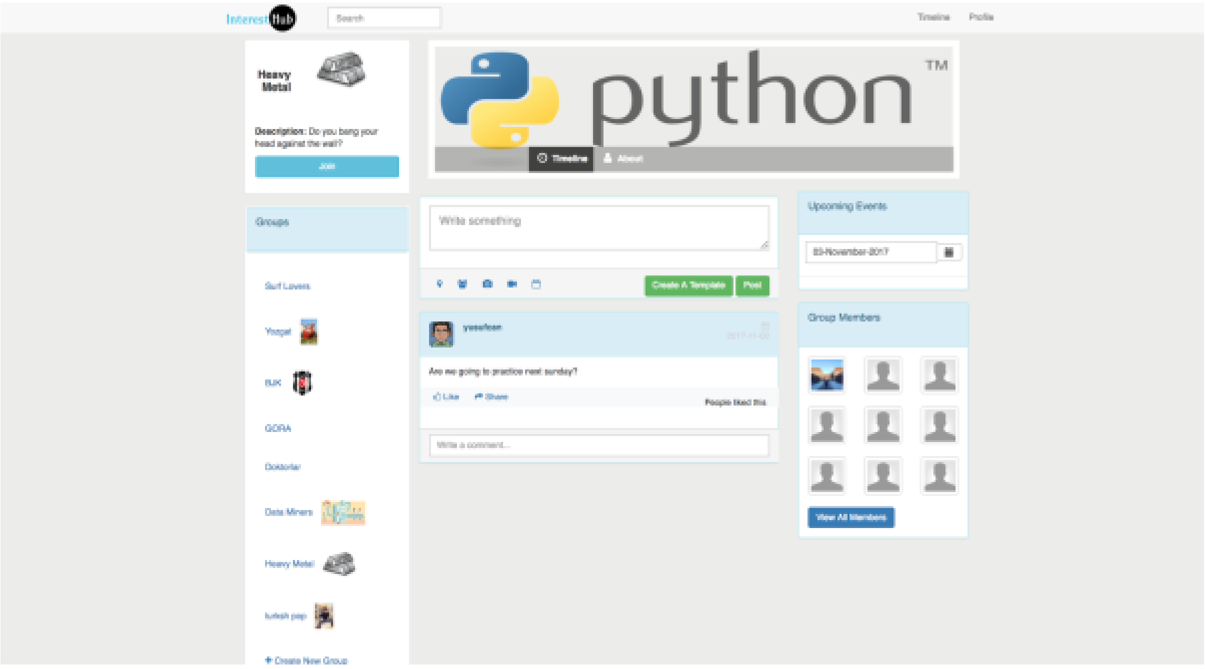
Bayburt is nice

Bayburt kopuyor muyuz?

Kopuyoruz

1. Sezon 1. Bölüm

5.1.3 Group Timeline



InterestHub Search Timeline Profile

Heavy Metal

Description: Do you bang your head against the wall?

Join

Timeline About

Write something

Create A Template Post

yusufcan 2017-11-02

Are we going to practice next sunday?

Like Share People liked this

Write a comment...

Upcoming Events

03-November-2017

Group Members

View All Members

Groups

Surf Lovers

Yozgat

BJK

GORA

Doktorlar

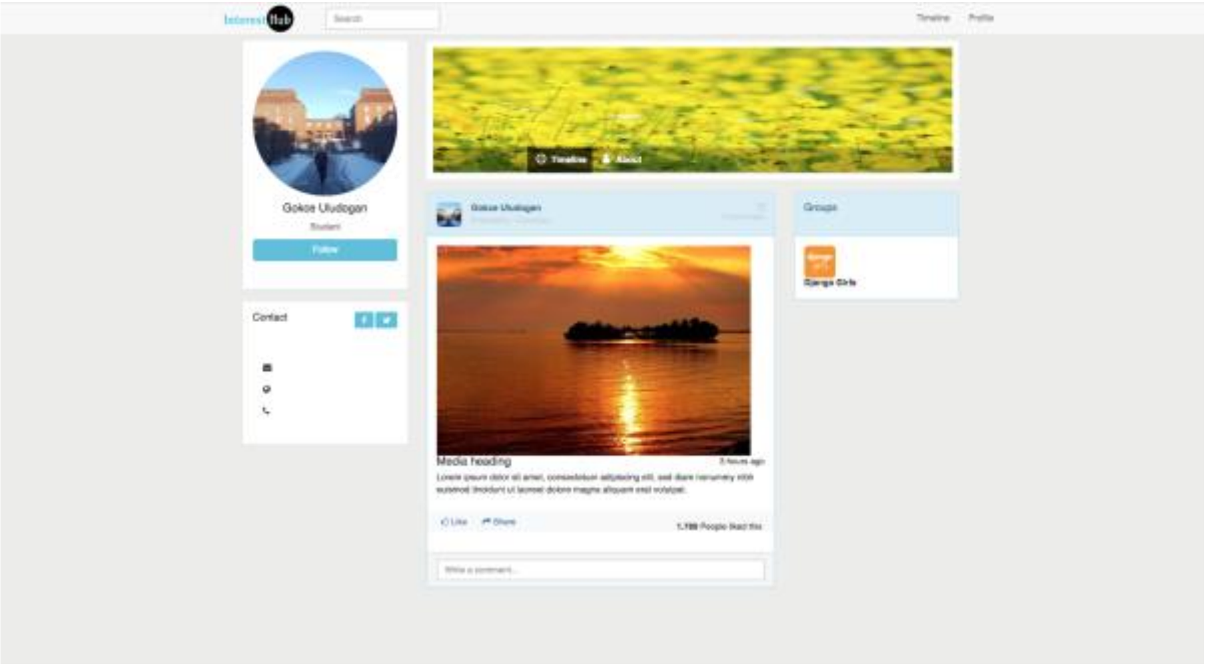
Data Miners

Heavy Metal

turkish pop

+ Create New Group

5.1.4 Profile Page



InterestHub Search Timeline Profile

Gokce Uludogan

Student

Follow

Contact

Timeline About

Gokce Uludogan

Media heading 3 hours ago

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

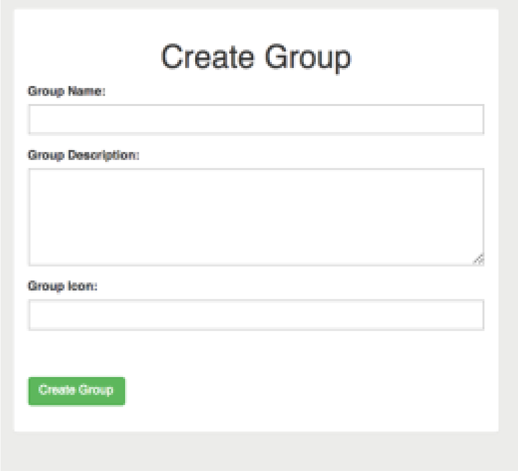
Like Share **1,788** People liked this

Write a comment...

Groups

Django Girls

5.1.5 Group Creation and Template Creation Page



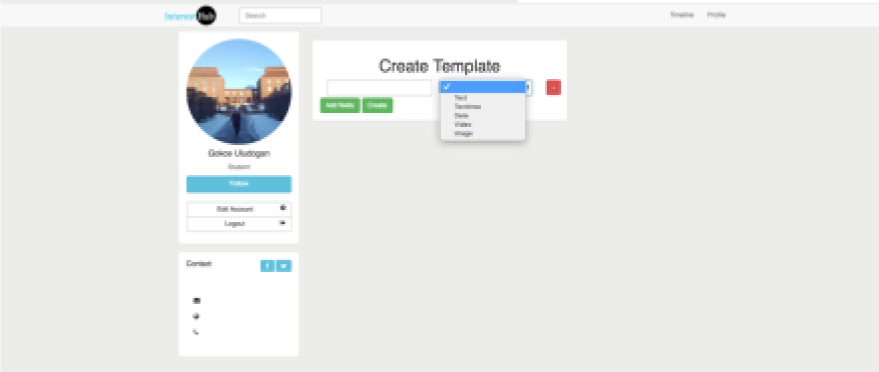
Create Group

Group Name:

Group Description:

Group Icon:

Create Group



Search

Timeline Profile

Create Template

Add fields Create

Text
Textarea
Date
Video
Image

Gokce Uludogan

Student

Follow

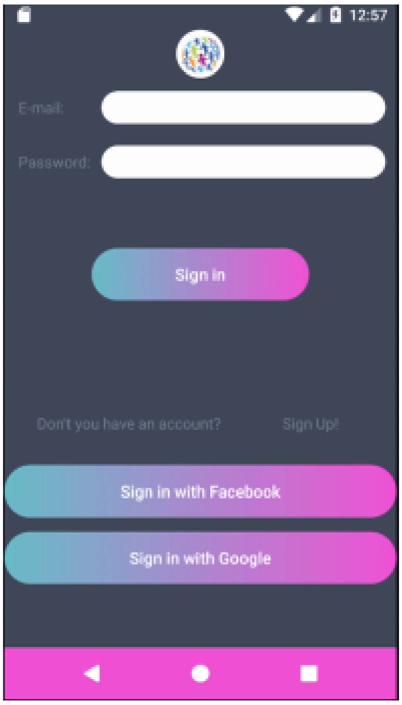
Edit Account

Logout

Contact

5.2 Android

5.2.1 Login page



12:57

E-mail:

Password:

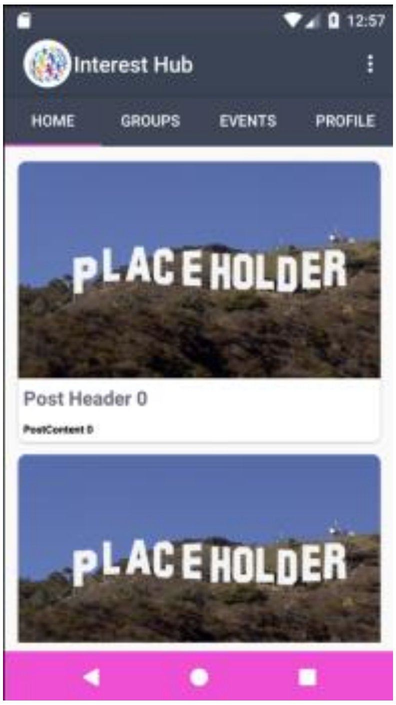
Sign in

Don't you have an account? Sign Up!

Sign in with Facebook

Sign in with Google

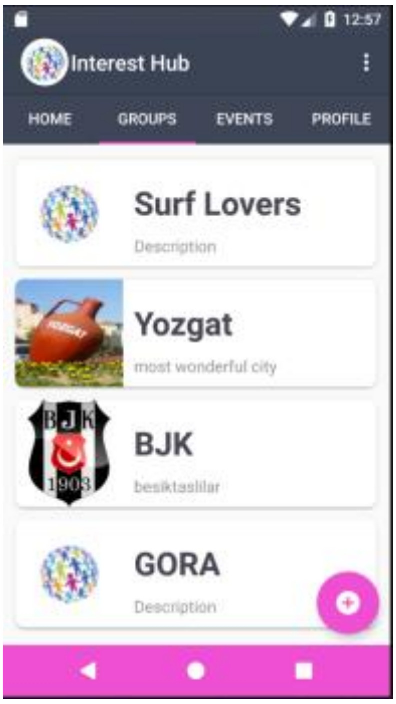
5.2.2 User Timeline



Post Header 0

PostContent 0

5.2.3 Group list



Surf Lovers

Description

Yozgat

most wonderful city

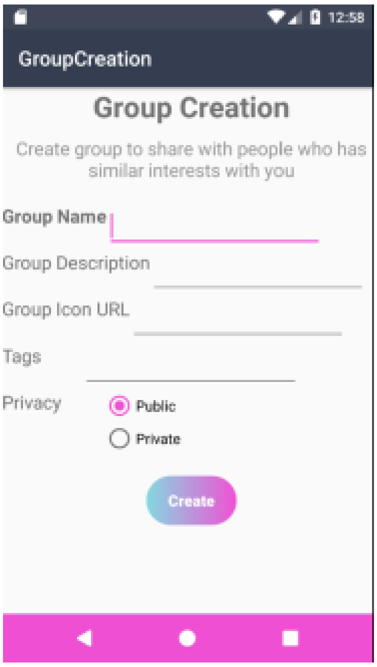
BJK

besiktaslilar

GORA

Description

5.2.4 Group creation page



12:58

GroupCreation

Group Creation

Create group to share with people who has similar interests with you

Group Name

Group Description

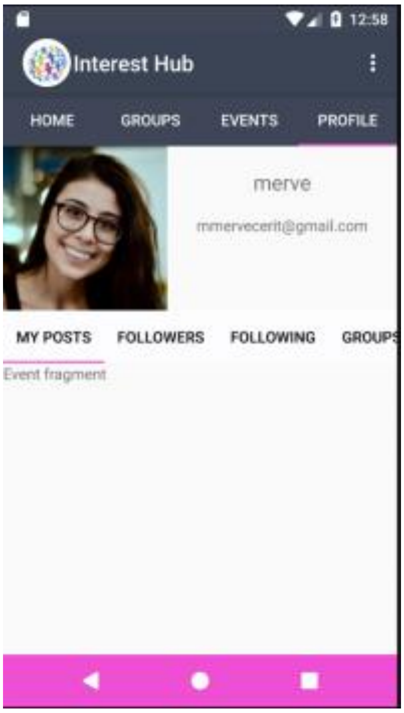
Group Icon URL

Tags

Privacy ● Public ○ Private

Create

5.2.5 User profile page



12:58

Interest Hub

HOME GROUPS EVENTS PROFILE

merve

mmervecerit@gmail.com

MY POSTS FOLLOWERS FOLLOWING GROUPS

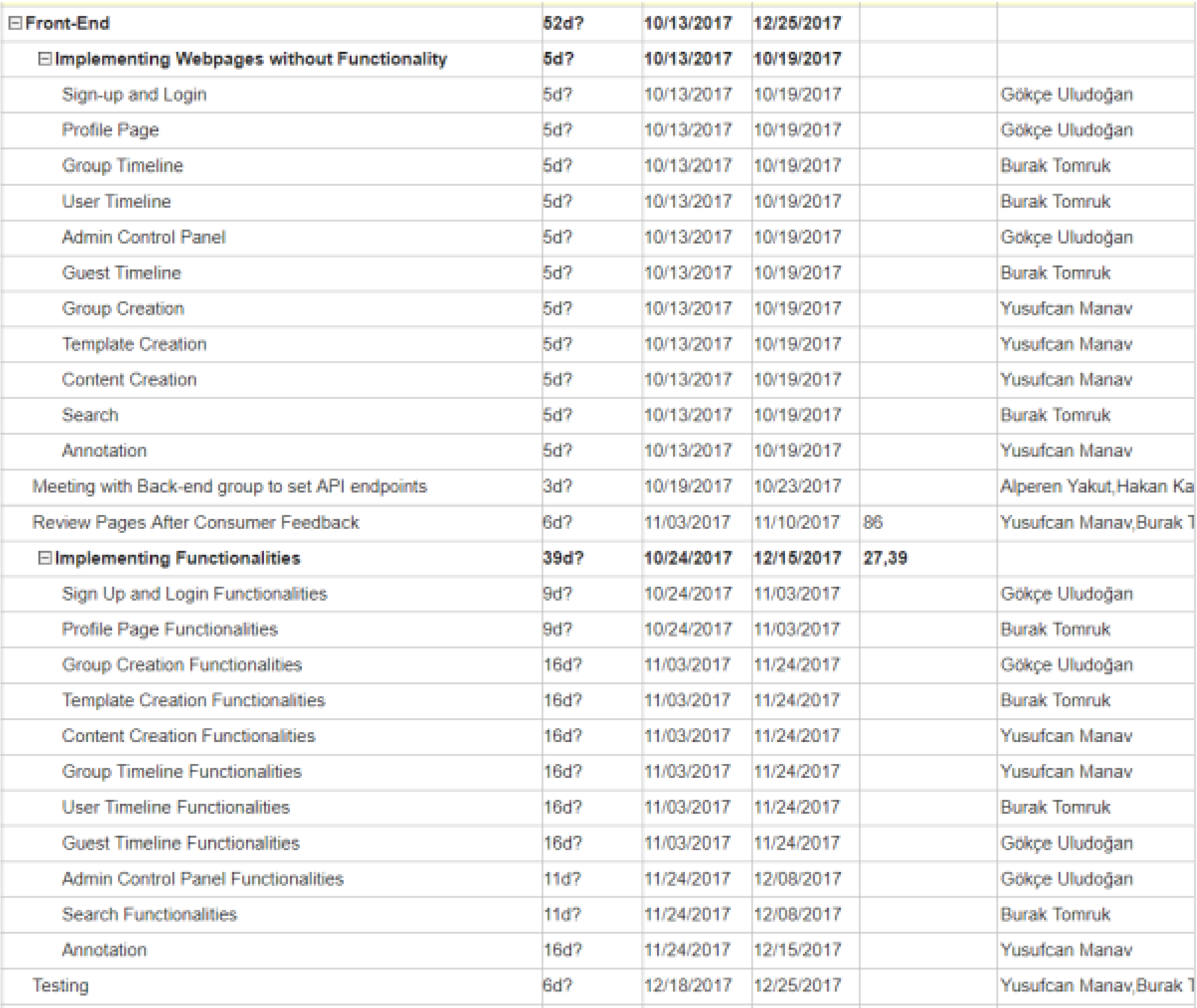
Event fragment

6 Project plan

Backend Plan

Task	Duration	Start	Finish
Whole Backend	52d	10/13/2017	12/25/2017
Class Diagram	7d	10/13/2017	10/23/2017
User API	20d	10/24/2017	11/20/2017
Group API	20d	10/24/2017	11/20/2017
Content API	20d	10/24/2017	11/20/2017
Custom Content API	20d	10/24/2017	11/20/2017
Search API	20d	11/21/2017	12/18/2017
Recommendations API	20d	11/21/2017	12/18/2017
Annotations API	20d	11/21/2017	12/18/2017

Frontend Plan



⊟ **Front-End**	**52d?**	**10/13/2017**	**12/25/2017**		
⊟ **Implementing Webpages without Functionality**	**5d?**	**10/13/2017**	**10/19/2017**		
Sign-up and Login	5d?	10/13/2017	10/19/2017		Gökçe Uludoğan
Profile Page	5d?	10/13/2017	10/19/2017		Gökçe Uludoğan
Group Timeline	5d?	10/13/2017	10/19/2017		Burak Tomruk
User Timeline	5d?	10/13/2017	10/19/2017		Burak Tomruk
Admin Control Panel	5d?	10/13/2017	10/19/2017		Gökçe Uludoğan
Guest Timeline	5d?	10/13/2017	10/19/2017		Burak Tomruk
Group Creation	5d?	10/13/2017	10/19/2017		Yusufcan Manav
Template Creation	5d?	10/13/2017	10/19/2017		Yusufcan Manav
Content Creation	5d?	10/13/2017	10/19/2017		Yusufcan Manav
Search	5d?	10/13/2017	10/19/2017		Burak Tomruk
Annotation	5d?	10/13/2017	10/19/2017		Yusufcan Manav
Meeting with Back-end group to set API endpoints	3d?	10/19/2017	10/23/2017		Alperen Yakut,Hakan Ka
Review Pages After Consumer Feedback	6d?	11/03/2017	11/10/2017	86	Yusufcan Manav,Burak T
⊟ **Implementing Functionalities**	**39d?**	**10/24/2017**	**12/15/2017**	**27,39**	
Sign Up and Login Functionalities	9d?	10/24/2017	11/03/2017		Gökçe Uludoğan
Profile Page Functionalities	9d?	10/24/2017	11/03/2017		Burak Tomruk
Group Creation Functionalities	16d?	11/03/2017	11/24/2017		Gökçe Uludoğan
Template Creation Functionalities	16d?	11/03/2017	11/24/2017		Burak Tomruk
Content Creation Functionalities	16d?	11/03/2017	11/24/2017		Yusufcan Manav
Group Timeline Functionalities	16d?	11/03/2017	11/24/2017		Yusufcan Manav
User Timeline Functionalities	16d?	11/03/2017	11/24/2017		Burak Tomruk
Guest Timeline Functionalities	16d?	11/03/2017	11/24/2017		Gökçe Uludoğan
Admin Control Panel Functionalities	11d?	11/24/2017	12/08/2017		Gökçe Uludoğan
Search Functionalities	11d?	11/24/2017	12/08/2017		Burak Tomruk
Annotation	16d?	11/24/2017	12/15/2017		Yusufcan Manav
Testing	6d?	12/18/2017	12/25/2017		Yusufcan Manav,Burak T

Android Plan

Task	Duration	Start	Finish
Android	49d	10/13/2017	12/2/2017
Research on Android	10d	10/13/2017	10/23/2017
Initial research projects	10d	10/13/2017	11/23/2017
User Page Implementation	2d	10/24/2017	10/26/2017
Login and Signup Page Implementation	1d	10/24/2017	10/25/2017
Splash Page Implementation	1d	10/24/2017	11/25/2017
Group Page Implementation	1d	10/25/2017	10/26/2017
Homepage Implementation	2d	10/26/2017	10/28/2017
Group creation page	1d	11/1/2017	11/2/2017
Adding functionalities	8d	10/24/2017	11/2/2017
First Customer Milestone	11/2/2017		
Developing fragments for User Page - Event, Follower, Following	7d	11/2/2017	11/11/2017
Developing fragments for Group Page - Event, Follower,Following	7d	11/2/2017	11/11/2017
Search functionality	5d	11/11/2017	11/16/2017
Semantic tags	7d	11/16/2017	11/23/2017
Testing	30d	11/2/2017	12/2/2017

7 Code Structure

7.1 Back-end

The first thing we did as the back-end team was to create a class diagram to visualize the project in our heads. We also chose Django as a developing framework. Then we could easily decide on what models to create, and how to relate them according to the class diagram.

In the back-end structure, we have ‘group’, ‘content’ and ‘components’ apps which provide necessary functionality. On the group page, group members share *content* which contains *components* such as text, images, and even videos. We only have these type of components for now, but our plan is to add interesting components like polls, locations etc. in the future. By the way, each shared content obeys a content-type standard. A content-type determines the structure and the appearance of a post. These content-types could anytime be created and modified by members of a group.

7.2 Front-end

As the front-end group first to talk about what are we needing in this project and how can we implement it. Since we are developing the pages as a group and the project was a big one our project structure should have least conflicts as possible in the development, because when we using branches in the github the more we branch and have conflicts the more we would have problems so we use modules to program our webpage. We implemented modules for each page and functionalities to decrease the conflict of changes in the web development to work more efficiently throughout the project.

We generally work on different branches for each functionality then we send pull requests and expected reviews from the team for merging the branches to the master branch after solving merge conflicts. Since we worked module by module there was mostly no conflicts when merging.

7.3 Android

On the android, Firstly we discussed how the project structure should be in order to ease implementation and fasten coding. Then we started to implement basic project with this structure. We thought about what will be our needs, what we might be use and we created some classes according to that. *SessionController* class is an example of that. It provides us necessary information about current session of user. It can be reached from anywhere in the application.

We are using less activities but more fragments. Fragments are better if we have common thing according to Android Documentation. For example, for user page, we don’t need to separate activities for both profile page and user timeline. Instead we created one User activity and connected two fragments to that activity.

Since it is team project, we tried to handle partially or totally different activities in the different classes so that we did not encounter with difficulties about conflict problems in git system.

8 Evaluation of tools and managing the project

8.1 Github

Since it is a software project, github and git system is vital. We tried to use all functionality of github, which make easier to develop as team, as much as possible. With issue management system we not only tracked our jobs but also requested some feature or improvements from other people. We also labeled our issues and pull request as much as possible for clarity. We designed our application according to the requirements, mock-ups, use diagrams in the GitHub wiki page.

As Android team when we wanted to push an improvement or new feature to the project. We firstly pushed to our local forked repository. Then we created new pull request from original repository. At this point everyone can see differences and can read report of pull request. After some people reviewed, generally somebody else merged with master branch in original repository.

8.2 Slack

Slack is our second most used communication tool. We have a channel on slack and we created different rooms with different subjects. In this platform we discussed about general project with whole team or we discussed design of Android project with people who works with Android only. We also shared useful informations, links with each other .

8.3 Django, django-rest

In order to create backend application, we used django with django-rest module. Django framework is easy to use to develop backend application. It provides many ease to store, get and show data with many desired relations. With the help of django-rest module, one can easily create a serializer and viewset to provide get/post/put/delete functionality for any model.

Although django has many convenience to develop backend application, its solid structure may prevent users to get used to. Due to the barrier of learning, developers can not start to develop in an easy way.

8.4 AngularJS

We used AngularJS in order to create frontend application. Angular is a structural web application framework that provides building dynamic web apps. We decided to use first version of Angular because of its stability and reliability. It also provides services that allows us to abstract API calls into reusable services which is very useful in our project for backend connection. We tested those API calls with Postman software tool. Bootstrap was also used in order to style the main structure of our application. In addition, we used a package manager called npm to install packages easily.