CmpE 352 - Milestone 2 Report

GROUP 1

May 27th, 2020

CONTENTS

1. Executive Summary

1.1. Introduction

Tursu is a e-commerce platform. It provides some functionalities to users such as buying or selling products, making comments about a product, reviewing the comments written by previous customers to ease the one's decision process and etc. The platform also provides a filtering and searching engine so that the customer can easily find a product which satisfies his or her desires. According to the search history and viewed products on the platform, the system recommends new products to the users. Therefore the experience of each user becomes individual.

In this part of the project, we are assigned to develop an application for practise. The APIs that we made use of were chosen according to the needs for our main project. The functionality of our practise app is as follows:
A user can add a product or list all the products already exists. He or she can also view the product details and comments about a product in a separate page. The price of the products can be viewed in different currencies in the product details page.

1.2. Work Done So Far

After the first milestone, we searched useful APIs for our project and then, we created three subgroups in our team. First group dealt with creating the base code of our practise application. Second group's mission was creating and designing front-end parts. The job of the last group was writing codes for API requests. After completing these works, pull requests were created, and all the branches have been merged.

1.3. Road Ahead

We have completed Milestone 2 until this point. From now on, we will start to implement our main project. In the main project, we will make use of more APIs and we will have more front-end and back-end functionalities than we used in our practise app. However, we may utilise the practise app since it has similar features with our project. Since the API implementation using Flask has lots of benefits, it is likely possible that we also use Python in the main project. In addition to what we developed so far, Android application for the project also will be developed. Since the main project will be a lot more broad and complex, we will need more comprehensive tests and controls for the project.

1.4. Challenges you met as a group

Most of us did not have enough experience in Web development before this class. Because of that we faced some challenges. For instance, we had to learn how to write API requests, how to deal with front-end issues, or how to use branches and pull requests. Since we did not know much about them, we had some difficulties while we were planning about how to make progress.

2. List and Status of Deliverables

Deliverable	Status	Accessible at
Practice App	Completed	http://52.20.110.175:5000/

3. Evaluation of the Status of Deliverables

3.1. Practice App

Everyone wrote her code and unittest on her own branch and requested to merge it with master branch. Then, each of the pull-requests are reviewed by at least two of the group members. If there is not any problem, the code was merged.

4. Summary of Work Done by Each Member

TEAM MEMBER	CONTRIBUTION
Ali Batır	I attended every group meeting so far.I searched APIs that we can use for our application.We divided into three subgroups as API, backend and frontend. I was responsible from the implementation of API requests.I implemented code and tests for spam detection api and bad words filter api. https://github.com/bounswe/bounswe2020group1/blob/master/practice_app/backend/flaskr/spam_detection_api.py https://github.com/bounswe/bounswe2020group1/blob/master/practice_app/backend/flaskr/bad_word_filter_api.py We did some improvements to frontend pages with Ufuk. We did the design for our application by using bootstrap framework. I wrote code for the part of navbar.html, layout.html and create_product.html.I also created a pull request for revision of the APIs and they were reviewed by other API group members. Then we merged my branch.I reviewed other APIs and the design part of frontend pages.
Asena Karolin Özdemir	I attended every meeting since the last milestone. I also took the meeting notes of the Meeting #12. I did research on APIs and found an API that could be useful for our project (mailbox layer) which validates and verifies emails, however we decided to implement different functionalities of our project as a group, so that API was not used during our implementation. I created the html file called 'home.html' which is the frontend part of the home page of our API. During this process we had a meeting with Onur Kılıçoğlu to ensure backend and frontend compatibility for the home page. I also created a pull request to merge the branch in which I had created 'home.html' to the master branch. I also reviewed the pull requests of my teammates who

	have implemented other parts of the frontend. Furthermore, I created a set of products to be displayed on our application's homepage and to form the database in which the search operation will be performed. I also wrote test cases for the search functionality (with assistance of Onur Kılıçoğlu who corrected syntactical errors) and tested the search functionality.
Barış Alhan	I have attended every meeting since the last milestone.We discussed which API to choose, and I took active role while deciding it. I suggested one API to use. I was in backend side of the practice-app. We had two other meetings for the backend side. I was responsible for creating the base code and database schemas. I implemented them by using flask. Then, I tested the base application by implementing simple prototype frontend and backend. Afterwards, I created pull request and adjusted my code according to the feedbacks that I received. Besides that, I reviewed pull requests of Onur and Murat and gave feedback about the general code style and consistency.
Barış Mutlu	I attended every meeting since the last milestone.We arranged meetings every week regularly.Also arranged for backend side for two times.In API research I suggested the Snipcart to automate the cart and shipping process.It was required key so it is not being used in this step. I worked on backend side. I implemented createproduct.py to add products to database and let frontend reach them by get and post methods.I tested my code by some inputs.We checked they are compatible with frontend side. I made pull requests for my implementations to my friends(can be seen in Git). I reviewed some codes of friends in Github(can be seen in Git).I gave them some feedback. I took the Meeting Notes 13.
Buse Kabakoğlu	I have attended all the meetings since the Milestone 1 is completed. My first task before starting to develop the practise app, was making some research about the sample APIs that we can use in the practise app. We decided that the sample APIs should be useful also for the main project. I came up with a location API which returns the information for the user's location by checking the device's IP address.After every team member offered an API, we chose four of them to continue. My part of task for implementing the practise API was writing functions that will be used in backend to retrieve information from the location API. I used Flask and implemented two function for the two ways that the URL can be used. It can retrieve location information with or without the IP address. In the second case, It uses the request itself to acquire the IP. https://github.com/bounswe/bounswe2020group1/blob/master/practice_app/backend/flaskr/location_api.py Afterwards, by using UnitTest class, I implemented two test functions. When the implementation is done, I made a pull request and assigned Ali, Ömer and Onur to review my code. I also made reviews about their APIs. After

	the reviews returned positive, my branch is merged to the master. In addition to the code, I wrote Introduction and Road Ahead part in the Milestone 2 report and also a individual report to explain what I have completed.
Mehmet Çelimli	I have contributed to the API research and found two API that could fit in well with the project. We have divided into three subgroups as API, backend and frontend. I was responsible for the implementation of the frontend page of creating new products. I have implemented html and css for the page and check the compatibility of the page with the backend. Frontend code can be viewed here: https://github.com/bounswe/bounswe2020group1/blob/master/API/FrontEnd/createP.html Afterwards, I made a pull request and added my frontend teammates as reviewers. I reviewed the codes of Asena, Yağız and Ufuk. I gave each of them feedback. After the project was merged I had tested the page that I had implemented. I had implemented unit test for creating new product functionality. Test cases can be viewed here: https://github.com/bounswe/bounswe2020group1/blob/master/Test%20Cases/test_createP.py I missed no meeting since the last milestone and I took notes of the 14th meeting.
Murat Ekici	I suggested what to build as our API and we decided on my suggestion. I did some research, then suggested to use Flask for API implementation. Finally we decided on Flask. I implemented the product route for the API. Link for the implementation: https://github.com/bounswe/bounswe2020group1/blob/master/practice_app/backend/flaskr/routes_product.py I used the `exchange_rate_api` to get the exchange rates for different currencies for the product page. I used `bad_word_filter_api` to check whether a comment contains bad words or not. In the product page, one can add a comment (POST request) to the database or can view a product's information and comments. I helped Baris Mutlu on create product page. I implemented the route for the home page (just one function). Link to that part: https://github.com/bounswe/bounswe2020group1/blob/master/practice_app/backend/flaskr/__init__.py After I finished my parts, i opened the pull request. Then we merged my branch. I reviewed some pull requests.

Onur Kılıçoğlu	For the research part I researched for different APIs and I found 3 different APIs that we can use in our implementation. For the implementation part I implemented the code and tests for communicating with the exchange rate API at "https://github.com/bounswe/bounswe2020group1/blob/master/practice_app/backend/flaskr/exchange_rate_api.py" which is later on used in the product endpoint to display prices in different currencies. Then I implemented the code for the search endpoint of the API at "https://github.com/bounswe/bounswe2020group1/blob/master/practice_app/backend/flaskr/routes_search.py". For the code review part I reviewed the code written by the team members Buse Kabakoğlu and Ali Batır who implemented the code for communicating with other external APIs that are used in the implementation. I also reviewed the base code for the backend implemented by Barış Alhan and the code for product endpoint implemented by Murat Ekici. To find out more you can see the pull requests 97, 105, 115 and 133. For the planning of milestone I attended 5 group meetings to discuss on the action items to be done for different parts of the project. I did 2 meetings with Barış Alhan, Barış Mutlu and Murat Ekici to discuss on the backend structure of the project. I also did a meeting with Asena Özdemir to ensure the compatibility between the frontend and the backend of the search endpoint. For the deployment part I created an Ubuntu 18.04 EC2 instance from AWS. I setup the app environment, downloaded the necessary tools and libraries and deployed the app onto server. You can find the deployed app at "http://52.20.110.175:5000/". For the testing part I wrote unit tests for the code for communicating with the exchange rate API "https://github.com/bounswe/bounswe2020group1/blob/master/practice_app/backend/flaskr/exchange_rate_api.py". I did user testing on the deployed application and fixed small bugs that are encountered during the integration of the code. I also helped Asena Özdemir to write unit tests for testing search endpoint of our API.

Ömer Ak	I have attended all meetings since Milestone 1. Firstly, we made research about APIs that we can use in practice app. I found 2 APIs that are used for sign-up (Telesign) and for shopping cart (Shopify). Then we chose 4 APIs to use our practice app, and my job was creating requests for the API called Datamuse which is used to find similar words to a given word. I wrote 5 functions for getting words with similar meaning, getting words with similar sound, getting words with similar spelling, getting words that strongly associated with the given word and getting suggestions to a given incomplete word. https://github.com/bounswe/bounswe2020group1/blob/master/API/find_similar_words_API.py After coding all these, I made a pull request assigning Ali, Buse and Onur to review. I reviewed their code, also. Then my branch is merged to master branch. I also wrote “Work done so far”, “Challenges you met as a group”, “Evaluation of the status of deliverables”, and some parts of “Evaluation of tools and processes” parts of the Milestone 2 report.
Ufuk Karagöz	I have attended all the meetings except for one meeting in total. Our first assignment for the practice app is researching and selecting APIs. I choose an API, but my API is a little bit offtopic so we decided to use some other APIs. Then we split three main groups; the frontend, the backend and the APIs. Asena, Mehmet, Yağız and me were assigned as the frontend group. They create three main HTML files and I manage them by using Flask. After that Ali and I did some improvements to our frontend. Such as add a navigation bar to the top of the page, specify background and text colours, add a footer which is a good sentence about our app, etc. When we are done on the frontend part as a group, I do some reviews and checks which are added location info, solve bugs about forms, editing URL names, change add comment design and add different currencies to our products. Afterall I created a pull request for my codes to the master branch and it is reviewed by other frontend group members. Also, I reviewed other frontend members' pull requests to merge their code to the master branch.
Yağız Can Çolak	I have attended every meeting since the last milestone. I have contributed to the API research by finding an API that could be useful for our project. But we decided that my API would be more useful a little ways down the road, so it's not used. I was in the frontend subgroup. I was responsible for creating the html code of product page. Later, I made a pull request and requested my frontend teammates as reviewers. I also reviewed the codes of Asena, Mehmet and Ufuk, and gave them feedback. We merged the frontend part of the project to the master branch. Furthermore, I wrote test cases for checking the comments of products, and tested if that comment exists or not.

5. Evaluation of tools and processes

5.1. GitHub

We used GitHub as our project version management system. We had a chance to see how useful it is. We have used issues for the works we needed to do, assignments section for managing the project, and wiki for documentation.

5.2. Slack

We used slack for communication.

5.3. Zoom

During the distance-education, we have used Zoom for our weekly meetings.

5.4. Python

We used Python to implement our backend and API functions since python provides many functionalities.

5.5. Flask

Flask is a framework that provides you tools and libraries for web application. We decided to use Flask in our practice app since it makes the web development faster.

5.6. HTML

We used HTML to design our front-end pages.

6. API documentation and URL

/product/<product-id>

The URL is used to list the informations of the product with the given product id

METHOD: GET
Sample query:
http://52.20.110.175:5000/product/3

/product/<product-id>

The URL is also used to make comments about the corresponding product using the POST method (comment form in the page).

METHOD: POST

Parameter	Description
author	string: Author of the comment
commentText	string: Comment text

Sample query:
http://52.20.110.175:5000/product/2

*Send the request with a form that contains corresponding parameters

/createproduct

Create product command redirects you to the product adding page which consists of a form to be filled with product info.

METHOD: POST

Parameter	Description
name	string: Name of the product
seller	string: Seller of the product
price	integer: Price of the product
description	string: Description of the product
url	url address: Link of an image of the product

Sample query:
http://52.20.110.175:5000/createproduct
*Send the request with a form that contains corresponding parameters

/search

Search endpoint is used to list the products whose names contains the provided keyword or a semantically close word to the keyword . The searched products are listed in the main page.

Method: GET

Parameter	Description
keyword	string: The keyword to be used for searching the products.
json (Optional)	boolean: Optional parameter to return result in JSON format. Set to “True” to get the result in JSON format.

http://52.20.110.175:5000/search?keyword=book&json=True
http://52.20.110.175:5000/search?keyword=shoe

API URL:

http://52.20.110.175:5000/