TESTING REPORT

Süleyman Furkan Öztürk 17070006009

Enes Ayberk Balcı 17070006007

Eren Uçkun 18070006023

Muhammed Berkay Yetiş 18070006059

1-

At the end of the software development, next is the testing phase. Our goal at this stage is to perform certain tests (unit, integration, usability, etc.) and see if there are any errors in the application. The purpose of these functional tests is to test each function of the software application, by providing appropriate input, and verifying the output against the functional requirements.

2-

We have implemented different test types for this application. These are unit, integration, acceptance, usability, performance tests. One of the main purposes here is to test the system from the smallest unit to the largest unit by dividing it into parts. In the result of these test stages. The result after these test stages was successful.

3-

*Upload Method--Uploading content to related areas—Usability testing--success—Manual

*Make comment--Adding comment—Acceptance testing—success--Manual

*Getdatafromfirerestorecomments—Unit testing—success--Automated

*Searchfilter--Categorizing specific search options—Usability testing/Performance testing—success--Automated

*Makerate--Preparing and do rate points—Unit testing—success--Automated

*Createuser--Creating user profiles—Integration testing/Security testing—success—Manual—Security testing not prioritized

4-

We have simplified the designs

in the application to give the user

more visual experience. The effects of

this on the application was to make the

application more understandable.

5-

• Total number of bugs found = 10

• Total number of fixed bugs = 10

• Total number of unfixed bugs (breakdown by priority and severity) = 0

6-

We have faced we could not user informations of comments from database as critical issue. We solved by using collections in documents.

7-

When there is data and load accumulation in the system,

 the application puts itself into maintenance mode so

that users are least affected.

Response time occurs approximately 1 hour after the

accumulation starts in online systems.

Users are never victimized in in-app delays, either the problem

is resolved as soon as possible or the application is taken into

maintenance until the problem is resolved.

Putting captcha during user logins in order to avoid accumulation

prevents sudden heavy logins or accumulations to the application.

Working on adding captcha.

Comments should keep in collections in firestore.