TEST PLAN REPORT

Senior Design Project II
CMPE492

Berfin Dolunay Karaçay - 42700117978
Deren Ekin Özdoğan - 12070118202
Doğa Günbatan - 11947049588

Contents

Purpose

Purpose of this report: to provide an overview of the latest versions of applications and modules, to test the available features and verify that they meet requirements and needs as planned, to conduct testing, to identify potential risks in testing, and to evaluate after testing is complete.

Application Overview

The application was created with the help of TED University students, faculty, and staff. When the application is entered, the login screen will welcome the user. If the user has not previously entered the application and created an account, s/he will need to register in the system. Name, tedu.edu.tr e-mail address and a password will be requested from the user on the register page. An OTP code will be sent to the e-mail address with the tedu.edu.tr extension entered by the user who performs register operation. The user will be logged into the application by entering this code sent to the e-mail address. At the same time, the name and e-mail address information obtained from the user will be kept in the realtime database. Thus, registered users will be able to access their home screen with the sign in button from the first login screen that opens when they enter the application. Announcements entered from the admin panel on the main screen will appear in front of the user according to the last sending order. When the user clicks on these announcements, he or she will find the title of the announcement, the image related to the announcement, and if the announcement is an event, the date and description of the event.

In the footer of the main screen, there will be TEDUClass and Societies screens as well as the home screen. If the user is a student or teacher, it is aimed that s/he can easily view her/his weekly lessons by placing them on the timetable. When the user enters the TEDUClass screen, s/he will be able to add or remove courses from the edit courses section. If the add course button is selected, the timetable will be able to select the course s/he wants to enter from a list of all courses, select a color s/he wants and add it to the timetable. When the Remove course button is selected, it will provide an option for the user to delete the courses that the user has previously selected and processed on the timetable.

Users who choose the Societies screen will be able to see the logos and names of all the societies within the university. Users who want to see it in detail will be able to select the club they want and access the logo, society name, whatsapp and instagram links and information about the society.

In addition to the pages we plan to have in the application, if the user wants to change the password s/he created while registering, s/he can change it with the change password button. Likewise, if the user wants to delete her/his account, s/he will be able to delete it with the delete account button. With the Credits button, the users who developed the application and the contact information of TED university will be found. By pressing the sign out button, the user will be logged out of the application.

The admin panel will be a web format system where the society and notifications seen by the users in the application can be entered and the information of the users registered in the application can be viewed and managed.

After the admin logs into the system with the user name and password, the admin will be directed to a home page. From this page, s/he will be able to access the User Table, Society Page, Disable User, Add Society and Send Notification pages. If the admin presses the User Table button, s/he will be able to see the names and e-mail addresses of the users registered in the application and will be able to search within the information. If the admin clicks the Society Page button, s/he will be able to view the current information of the societies and edit them separately. If the admin clicks the Disable User button, s/he will have the authority to disable the user from the list of available users. On the Add Society button, they will be able to add their logos, names, e-mail addresses, descriptions, whatsapp and instagram links for newly established communities within the university. Finally, in the admin panel, if the admin clicks on the Send Notification button, s/he will be able to submit the content of the notifications that will fall on the homepage of the application as author, date, description, image, status and title.

Testing Scope

The test scope of the project is determined by a detailed evaluation process. This process considered which features and elements should be tested as part of the project and which parts of the applications should not be tested in the interest of the integrity of the project.

In Scope

API testing is used to test the functionality of key application features.

Out of Scope

Performance testing is not included in this testing process, as it was not performed because the integrity of the applications' implementation was compromised.

Types of Testing

API Testing

API testing is a form of software testing that involves scrutinizing an application program interface (API) to confirm its adherence to predetermined standards of functionality, security, performance, and reliability. The assessment may be conducted directly on the API or as a component of integration testing.

API Test advantages:

- **Faster testing:** Since API tests focus on the functionality of the API, they enable faster testing of software applications compared to other testing methods such as GUI tests.
- **Early defect detection:** API testing enables early detection of defects in software applications, which can help reduce overall development and maintenance costs.
- **Automation:** API testing can be easily automated, saving time and effort in manual testing. Automated API testing also ensures consistency in testing and reduces the risk of human error.
- **Integration testing:** API testing is an essential part of integration testing, ensuring that the different modules of the software application work together seamlessly.
- **Improved security:** API testing helps in identifying and fixing security vulnerabilities in the software application, which helps in improving the overall security posture.
- **Better performance:** API testing helps in identifying and fixing performance issues in the software application, resulting in better performance and user experience.

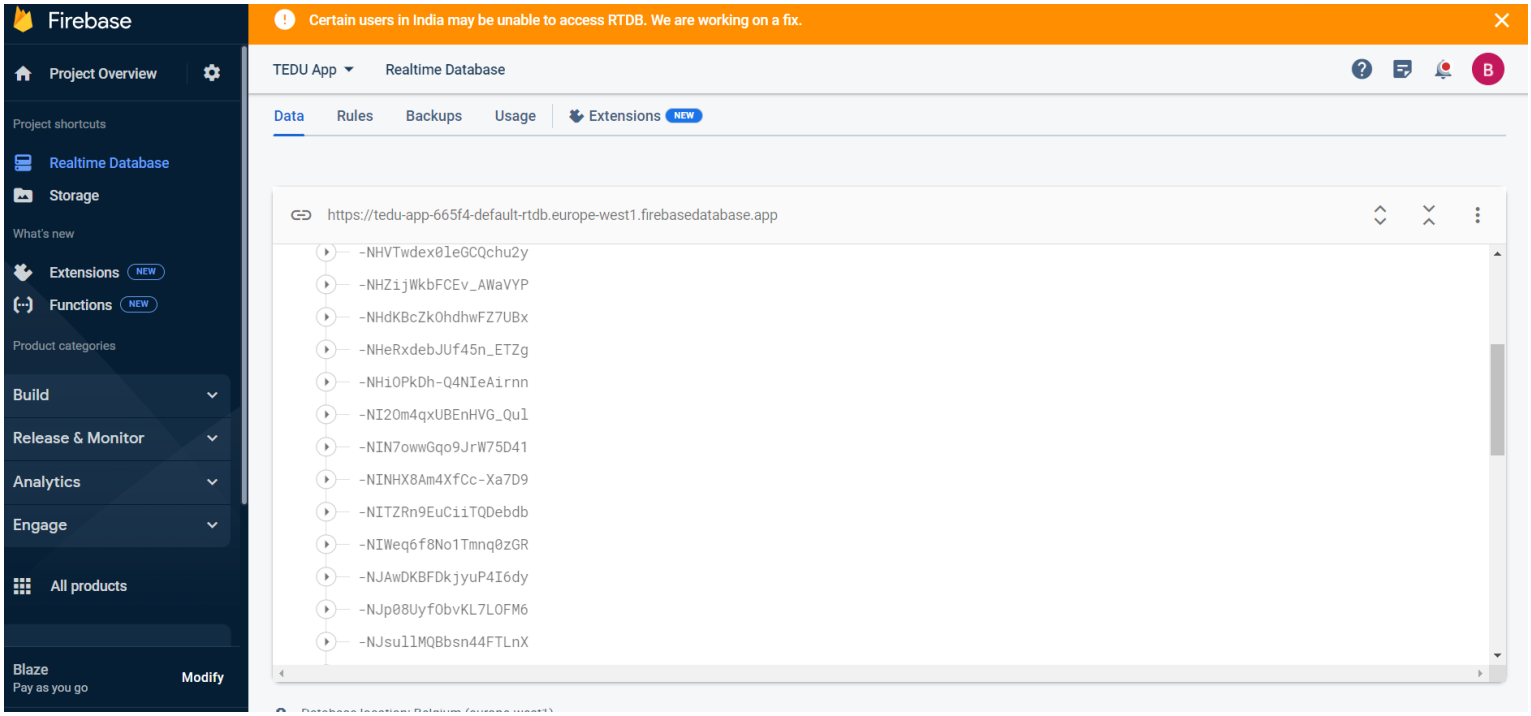
System integration testing

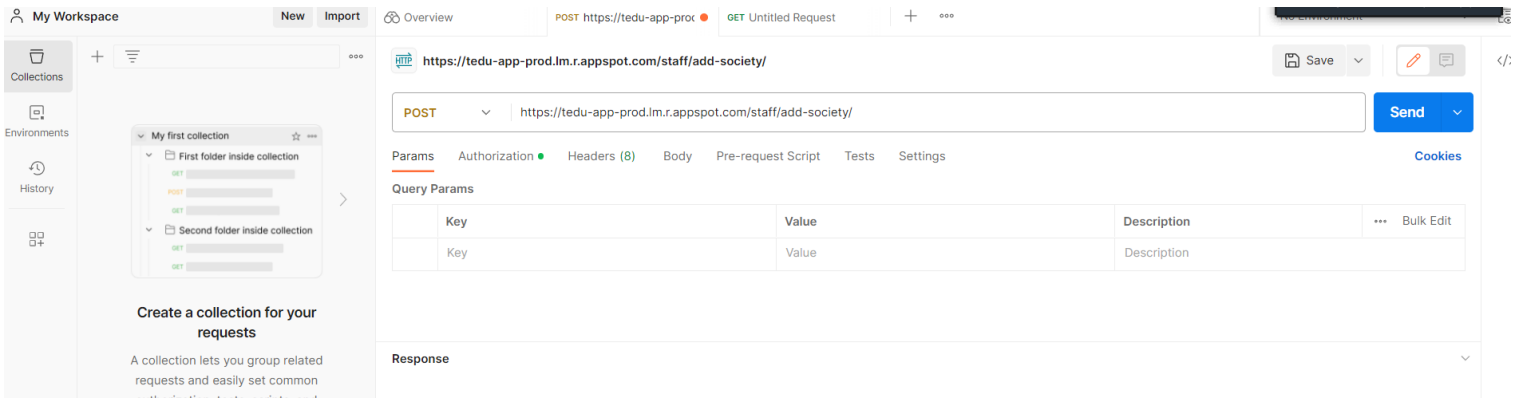
The objective of the system integration test is to verify the functioning of all individual submodules. This type of testing is aimed at ensuring that all software modules operate seamlessly and that data security is maintained between different modules of the complete system. Essentially, the system integration test serves to test the integration of various software components to ensure that they work together effectively and securely.

To form a complete system, modules like login, registration, administration area, and database need to be integrated. Testing is carried out to ensure the seamless integration of the app with the database, and certain issues are identified during this process. In essence, the integration of different modules of the system is critical, and the testing of this integration is crucial for identifying and resolving any issues that may arise.

Test Environment & Tools

We used Firebase, Postman and MySQL.





Evaluation

Test ID	Issues encountered	Solutions/Outcome
1	API testing is crucial to ensure the reliability, security, and performance of software applications. Problems that may be encountered; Incorrect data formatting, authorization issues, error handling, performance and scalability, versioning	Using Postman, we detected and fixed the errors in the application and the functionality in the admin panel.
2	There was a chance that some partitions' directory may undermine the integrity of their programs.	Manual integration testing is used to find problematic modules.

Risks

- Since paid applications and websites are used in application development, there may be difficulties in logging into the system and transferring data if the capacity is exceeded.

- A memory error may occur when the system is detecting meshes.

- In particular, in the TEDU Schedular section, the data is taken from the university. If the course information changes, it may not be reflected in the application at the same time.

- Correct operation with different screen orientations, interruptions of Internet connections, and memory and battery consumption.

Test Schedule

Task Name	Start Date	End Date
Determining the Tests	20.03.2023	25.03.2023
Add Society API Testing	30.03.2023	01.04.2023
Disable User API Testing	03.04.2023	04.04.2023
Edit Society API Testing	05.04.2023	08.04.2023
Login Page API Testing	09.04.2023	14.04.2023
Single Society API Testing	15.04.2023	16.04.2023
Society Page API Testing	17.04.2023	20.04.2023
User Table API Testing	21.04.2023	22.04.2023
Notification Page API Testing	25.04.2023	03.05.2023
System Integration Testing	05.05.2023	10.05.2023

Roles and Responsibilities

- **Berfin Dolunay Karaçay:** Berfin was performed in Notification Page and its test parts. She also took part in the system integration test.

- **Deren Ekin Özdoğan:** Ekin took part in the pages of the Societies and its tests. Additionally, she took part in the system integration test jointly with Berfin.

- **Doğa Günbatan:** Doğa took part in the login page and tests as well as the pages that interest the users. She helped Ekin in editing the Society's information.