CS 319 - Object-Oriented Software Engineering

Final Report

v.map

Group 3-C

Yağız Gani

Kanan Asadov

Ömer Faruk Karakaya

Mert Osman Dönmezyürek

1. Introduction

This final report includes the user guide on how to install the program, also the changes, additions, incomplete parts, and difficulties during the implementation process.

2. User Guide

Since v.map is a web application, it does not require any extra installation but a javascript supported web browser on any operating system on the user's side, which are the followings:

Operating Systems:

Windows, Mac, Linux / Ubuntu, Google Chrome OS (Chromebook)

Web Browsers:

Google Chrome v34 or later, Mozilla Firefox v34 or later, Internet Explorer v10 or later,all versions of Microsot Edge or Apple Safari v6 or later

For the server side, system requires following applications to be installed on the server:

JDK 8+ and JRE 8+

Java Application Server (Apache TomCat 8)

PostgreSQL

All of them are compatible with Linux, Windows or MacOS ditributions. To run the web application, the source files sould be uploaded on the server.

3. Changes and Additions

3.1. Session Management

Session Manager Class

private boolean timerCheck attribute, indicated in the **SessionManager** class is deleted. There is no need for it, since we will not use the 2 timer methods in the **SessionManager** class. The reason is explained below.

void timerForLogin() & **void timerForVCode()** functions created in SessionManager class, will not be implemented, because we will deal with the timing while implementing the view classes which are indicated in the “design report”.

Additionally to the design stage, an **EmailServer** object is indicated in the SessionManager class, in order to be used in the **sendVmail()** method.

A constructor of SessionManager class is created. **User & DatabaseManager** objects are dynamically implemented inside the constructor.

Also, in the **getUserDetails(int userID)** method, **userID** was String type, it is converted into int type.

UserManager Class:

Default constructor is created in UserManager class.

The parameter of **createAnnouncement()** method is changed from **String userID** to **int userID.**

The parameter of **deleteAnnouncement()** method is changed from **String announcementID** to **int announcementID**.

3.2. E-Mail Service

EmailService Class:

private string prepareHTMLEmail(String text), this method was planned to create an email template in HTML by inserting the given text into the HTML codes and return string of the codes. However, it doesn't create a user-friendly email template. Therefore, it is decided to change it as **private String**

prepareVerificationEmailInHTML(String userFullName, String verificationCode) method. Instead of generating a general template for all emails, this method creates a specific well-designed verification e-mail template in HTML by taking the full name of the user and the verification code, then returns HTML code in string.

private void sendEmail (String recieverEmail, String mailSubject, String contentOfEmail), this method was meant to be a void method but other classes that use EmailService class need a feedback on whether the e-mail is sent successfully or not. Hence, the type of the class has been changed as boolean type to notify its user by returning true if the process is successful or false if it is failed.

3.3. View Classes

Recently Added Classes:

CreateAnnouncementView, **ManageAnnouncementsView**, **ManageUsersView** classes are recently added. Functionality of these classes were provided by MapView class previously but in order to increase modularity and simplicity of design they are seperated.

3.4. Database Connection

DatabaseManager:

getAnnouncementsConcise() was deleted and replaced with **getAnnouncements()** that returns all anouncements completely instead of only their coordinates and titles. This was made because Hibernate methods for getting specific columns from the database tables were deprecated and there was lack of sources explaining how to solve the problem. Moreover, this change did not make any difference in performance. Thus, there was no point on having the first method.

ID’s:

ID’s of users and announcements in parameters of methods were changed from String to int.

Database:

Our database was downgraded from PostgreSQL 10 to PostgreSQL 9.5 since it is very new and Hibernate does not a specific dialect for the newer version.

3.5. Model Classes

Model Classes were altered to support Hibernate persistance.

Position:

Position is now not an inner class of Announcement and it is Embeddable.

announcementIdList now consists of integers and not strings.

4. Incompleted parts

4.1. Session Management

- The Session Management part is totally finished as planned. However, additional changes, modifications or updates will be done according to the feedback from the Instructor or according to the possible errors based on other sections of the project.

4.2. E-mail Service

- All parts are done.

4.3. View Classes

- Filters are created as selection boxes and radio buttons however their functionality is not defined yet because of the incomplete parts at Database Connection part. These filters functionalities are dependent to announcement data which will be fetched from Database Connection subsystem.
- Style and Theme for User Interface is not completed yet because achieving providing expected functionality is prioritized.
- Log-in, sign-in, manage users, manage announcement functionalities are not completed because of incompleted parts form Database Connection and Session Management subsytems.

4.4. Database Connection

- When a user adds an announcement, its id is not assigned to the user's *announcementIdList.*

5. Difficulties

5.1. Session Management

- Writing SessionManager's class code part was difficult because the User class and the DatabaseManager classes were not finished at the time. So it was a bit difficult to predict whether the code will work or not.

5.2. E-mail Service

- Since Gmail does not support CSS, <html> and <head> tags in HTML while sending an e-mail, adjustment of the user-friendly e-mail template in HTML (the method **prepareVerificationEmailInHTML**) was significantly time consuming.[1]

5.3. View Classes

- There were no major diffictulties during the implementation of this part.

5.4. Database Connection

- A lot of methods that were being used for years were deprecated. Thus, it was very hard to figure out how to provide the connection correctly in some parts. This forced us to change our structure. Since we were not able to get specific columns from the database table we had to change getAnnouncementsConcise() to getAnnouncements()
- We wanted to use PostgreSQL 10, however, since it has just got out Hibernate had some problems supporting it. Thus, we had to change our database and downgrade it.

6. References

[1] https://www.campaignmonitor.com/css/