Bilkent University

Department of Computer Engineering

CS 319 Term Project

ReviewTool

Final Report

Mustafa Anıl TAŞDAN, Göktuğ ÇAĞIRAN, Cihan Can KILIÇ

Instructor: Eray TÜZÜN
Teaching Assistant(s): Erdem TUNA, Elgun JABRAYILZADE

Table of Contents

1. Introduction

We made a change to DAO object to follow DAO design pattern. At first, we used DAO class to access all tables in the database. This resulted in repetitive codes and poor performance. Also, this kind of approach was against best practices and DAO design pattern. So, we decided to make a different DAO class for each table in the database. In terms of implementation our program is 85% done. In terms of subsystems missing implementations are discussed below.

Interface:

- Artifact Review Screen: Screen to view and submit artifact reviews is missing.

Business Logic:

- Some methods of the BusinessLogicTopController
- Some methods of the AssigmentController

DAO:

- DAOArtifactReview: Database Access Object to communicate with ArtifactReview table.
- DAOArtifact: Database Access Object to communicate with Artifact table.

2. Lessons Learned

As we are kind of inexperienced in projects like this there was a lot to learn throughout the way. First of we would like to begin with technical issues. When we were thinking about how to implement the project at the very early times we thought of using Firebase services. However we learned that we couldn’t use it like a built-in tool like we can in Android Studio. Then after realizing this we decided on using a Microsoft SQL Database which we had no technical knowledge of so we learned building and implementing the SQL services. Other than this we learned design patterns in CS319 so we didn’t just start writing and making up bad designs while writing the code as we did in past years. First we designed the project very detailed. Then designed implementation process with class diagrams and started writing the code after the whole code was designed to be written. We saw that is a lot easier than diving heads first to writing the code. In past years when we did poor designs or no designs at all we faced lots of problems through the process but this time we had just couple of problems and they were solved easily thanks to the design patterns, as the sub-systems are working almost completely remotely we only had to change the code where the problems happened and some small changes was needed in other parts in order to solve it. We also learned a lot about the communications between sub-systems and classes that form them. As the project was designed beforehand we knew which classes should pass which information to which class so writing code with this information was easier than any project we did. Also we knew basic functions of GitHub but with this project we became very comfortable with using it. In conclusion of the technical part we can sum up the things we learned as database systems, design patterns and importance of designing the whole project beforehand and last but not least we became comfortable with GitHub.

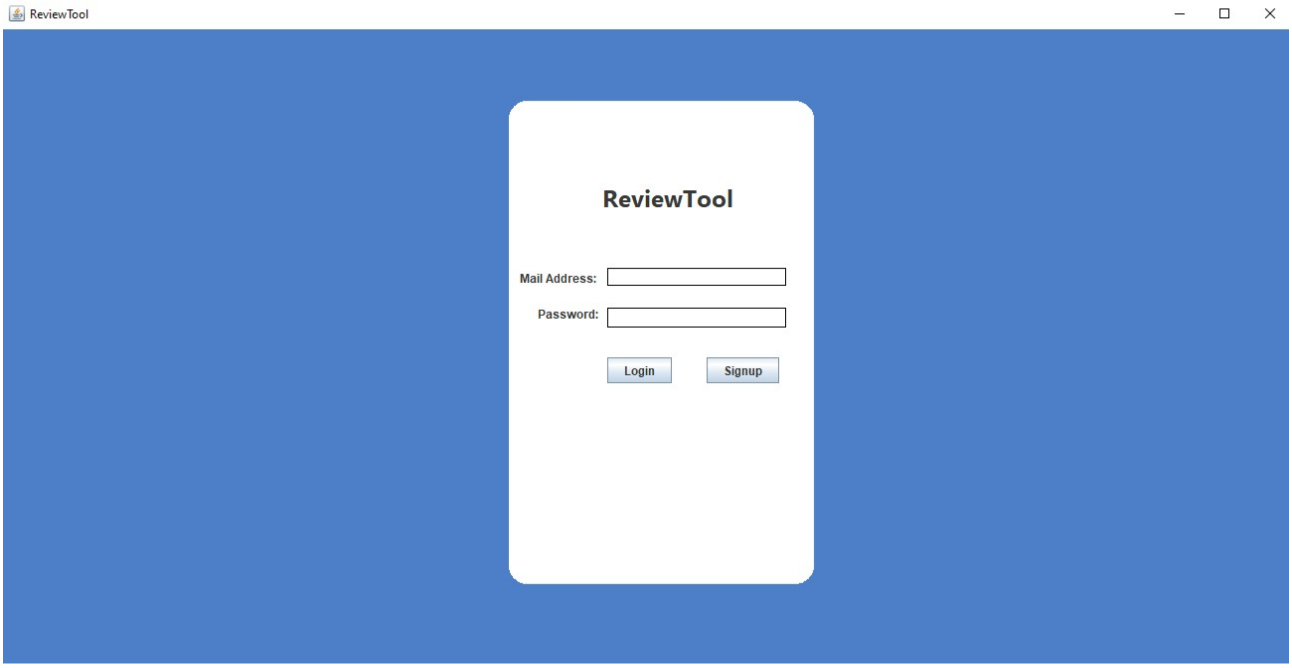
Other than technical stuff, our experiences throughout the project thought us many things about group work and how to work on a project as a team as well. At the beginning our group was of five members but we couldn't reach one of them and another one dropped the class while we were just beginning the designs. At first we were actually scared of the workload it brings and we had doubts about if we could finish the project. Also as the remaining three members we had some communication problems between us and we couldn't manage the time so we actually started off pretty bad. After the first iteration we understood that we have to be more punctual and we have to work a lot tidier in order to put on a good project at the end since the workload is more when compared to a group of five. After realizing that there is a real problem that is preventing us from catching up with deadlines we had a meeting talked about this and after that meeting the problem was like never occurred, we started working a lot better and faster. So we learned the importance of communication between members and teamwork in the hard way but the important thing is we learned it.

3. User Guide

The application is run from a jar file, either from the one provided by us in GitHub or one built from the source files.

Running the jar file takes the user to the login screen.

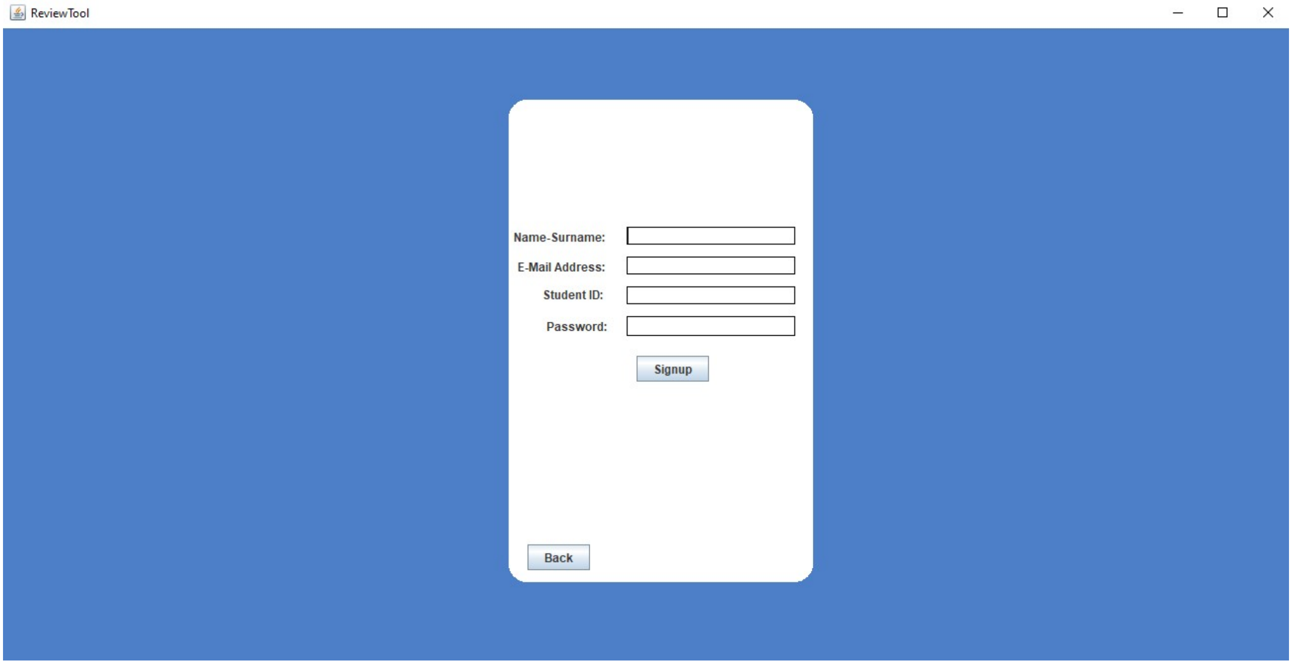
3.1) Logging In



All users, whether an instructor, a TA or a student has to go through the login screen for authentication. A user logs in to the system using their e-mail address and password stored in the system. The user interface will display an error if the user enters invalid credentials.

A new user will have to go through the sign up screen to gain access to the application.

3.2) Signing up

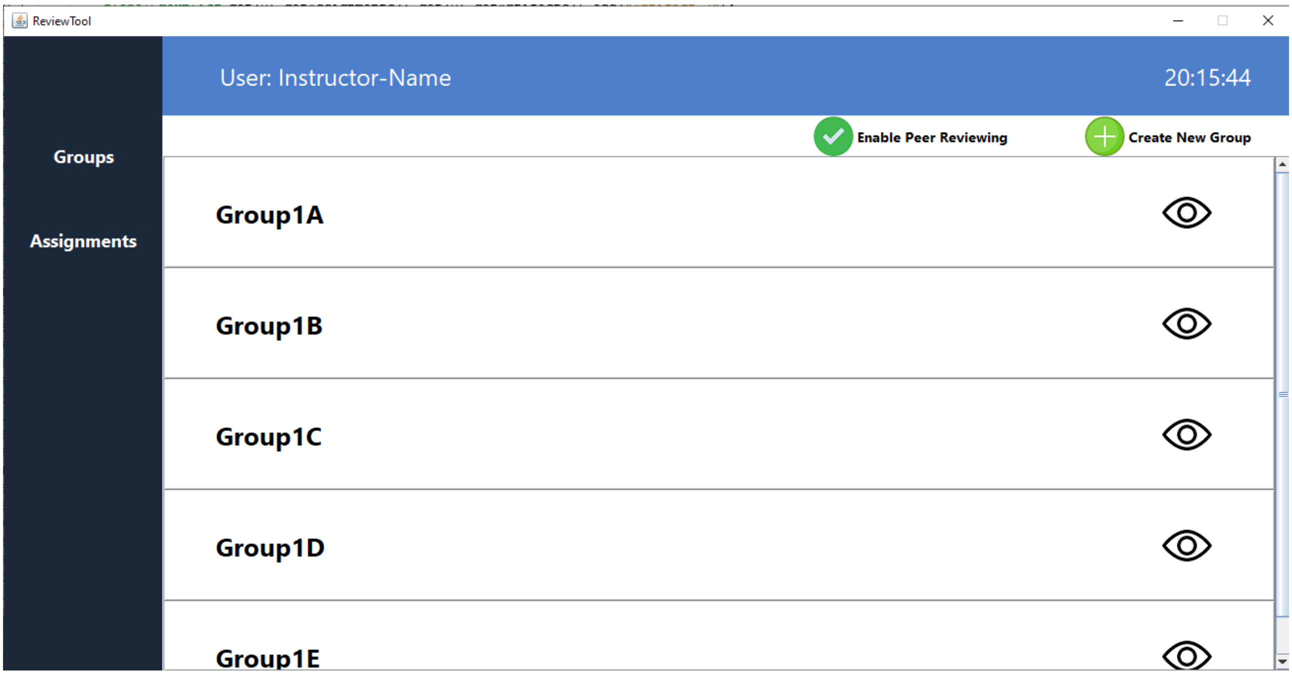


The sign up procedure changes for user types:

- An instructor has to be added manually to the database, an instructor account type cannot be created through this screen.
- A student has to enter his Name-Surname, E-Mail Address, Student ID and Password to send a sign up request.
- A TA also enters the same information, but leaves the Student ID field empty, which prompts the system to create a TA account type in the database.

The system will show an error if the user already exists in the database.

3.3) Instructor Dashboard



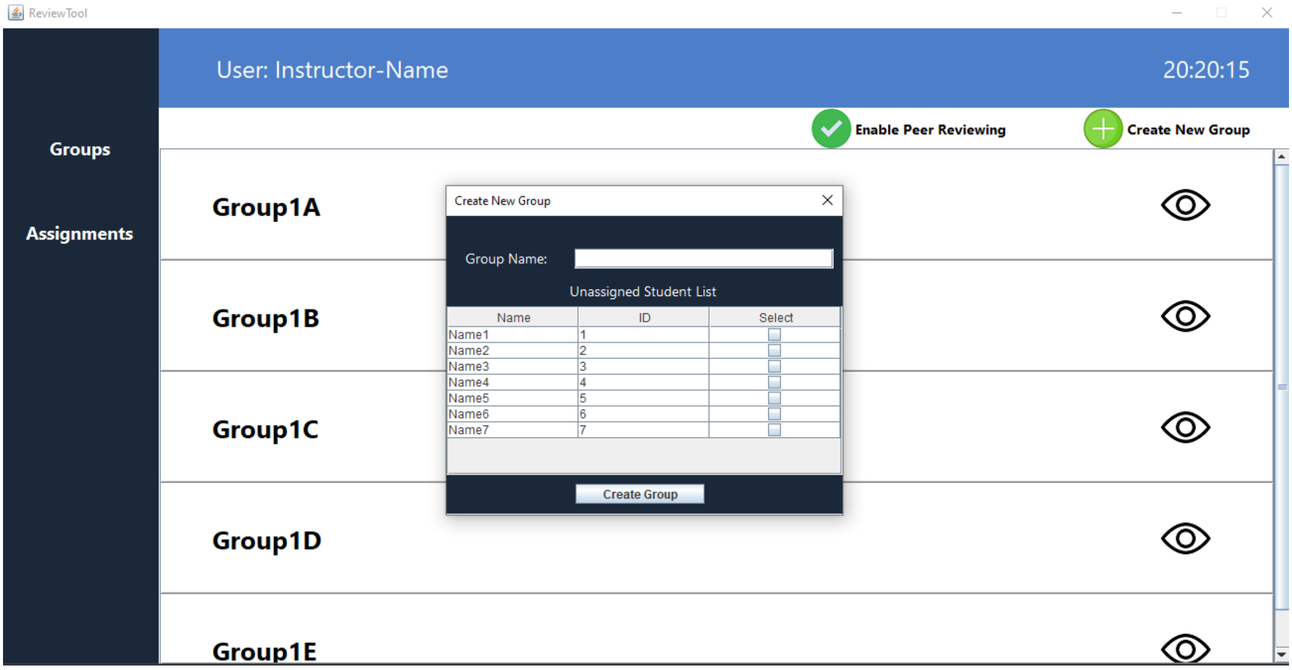
After login, an instructor lands at the instructor dashboard. The dashboard is divided into three panels:

1) Left panel allows navigation between group and assignment lists.
2) Top panel displays logged in user’s name and the current time.
3) Middle panel is shows entity lists and buttons, fields to control those entities.

The first two panels are constant, while the middle panel changes depending on the taken actions.

3.3.1) Groups Screen

3.3.1.1) Creating a New Group



Name	ID	Select
Name1	1	☐
Name2	2	☐
Name3	3	☐
Name4	4	☐
Name5	5	☐
Name6	6	☐
Name7	7	☐

Clicking the “Create New Group” button on top right corner displays a pop-up window to the instructor. The instructor has to enter a Group Name and select students without a group. The instructor can select as many students as he wants, but both fields must be filled/selected, and the system will present the user with an error pop-up in failure to do so.

3.3.1.2) Enabling/Disabling Peer Reviewing

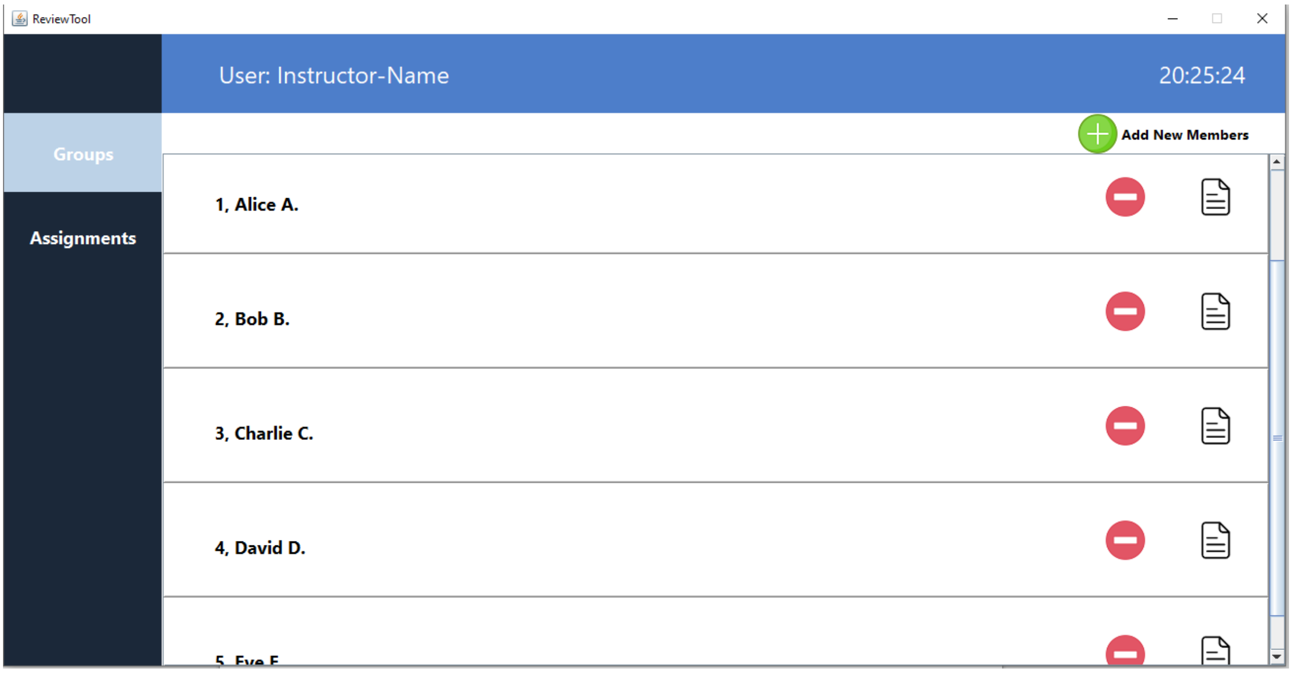
The “Enable Peer Reviewing” button is a button that switches states when clicked, it changes into “Disable Peer Reviewing” button. When peer reviewing is disabled, students cannot update their peer reviews from their screens.

3.3.1.3) Viewing Group Details

Clicking the “eye” icon in a group’s panel takes the instructor to the Selected Group screen.

3.3.2) Selected Group Screen

3.3.2.1) Adding/Removing Members



1, Alice A.

2, Bob B.

3, Charlie C.

4, David D.

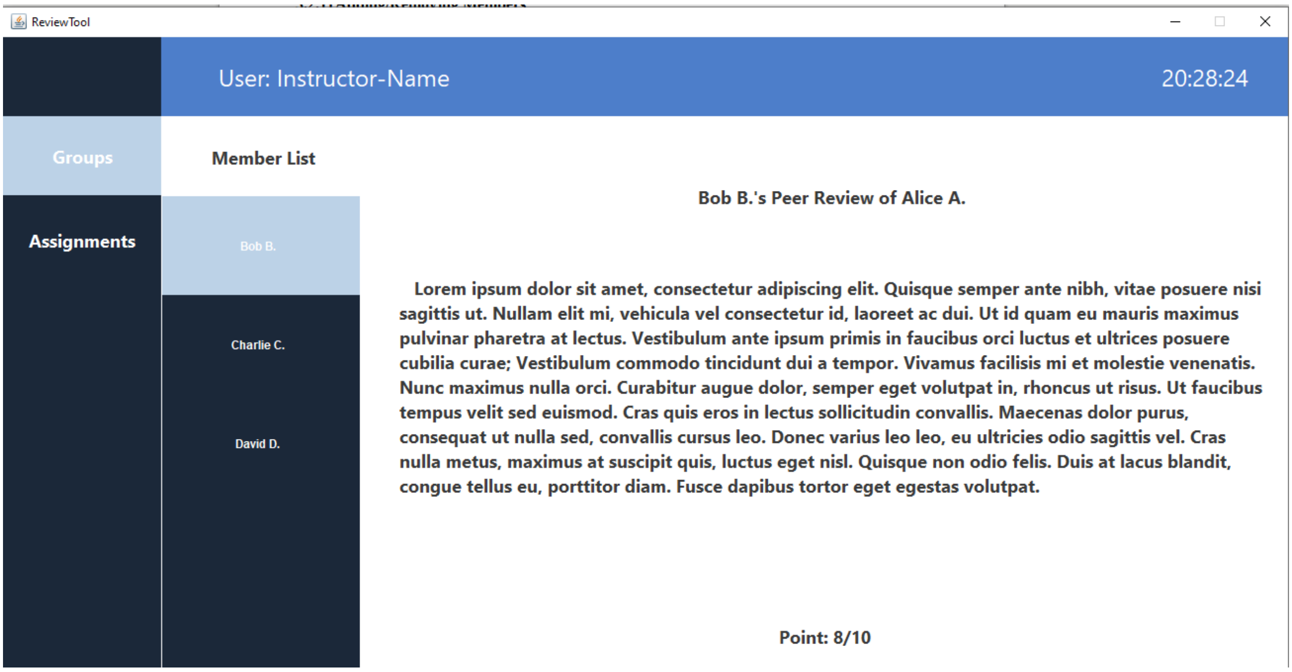
5, Eve E.

The Selected Group screen displays the member list of the selected group. The “Add New Members” screen shows the same pop-up used in “Create New Group” screen, where the instructor can add members.

The red minus icon in a member’s panel removes them from the group.

The page icon takes the instructor to the user reviews made on that member.

3.3.2.2) Viewing Peer Reviews



Bob B.'s Peer Review of Alice A.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque semper ante nibh, vitae posuere nisi sagittis ut. Nullam elit mi, vehicula vel consectetur id, laoreet ac dui. Ut id quam eu mauris maximus pulvinar pharetra at lectus. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia curae; Vestibulum commodo tincidunt dui a tempor. Vivamus facilisis mi et molestie venenatis. Nunc maximus nulla orci. Curabitur augue dolor, semper eget volutpat in, rhoncus ut risus. Ut faucibus tempus velit sed euismod. Cras quis eros in lectus sollicitudin convallis. Maecenas dolor purus, consequat ut nulla sed, convallis cursus leo. Donec varius leo leo, eu ultricies odio sagittis vel. Cras nulla metus, maximus at suscipit quis, luctus eget nisl. Quisque non odio felis. Duis at lacus blandit, congue tellus eu, porttitor diam. Fusce dapibus tortor eget egestas volutpat.

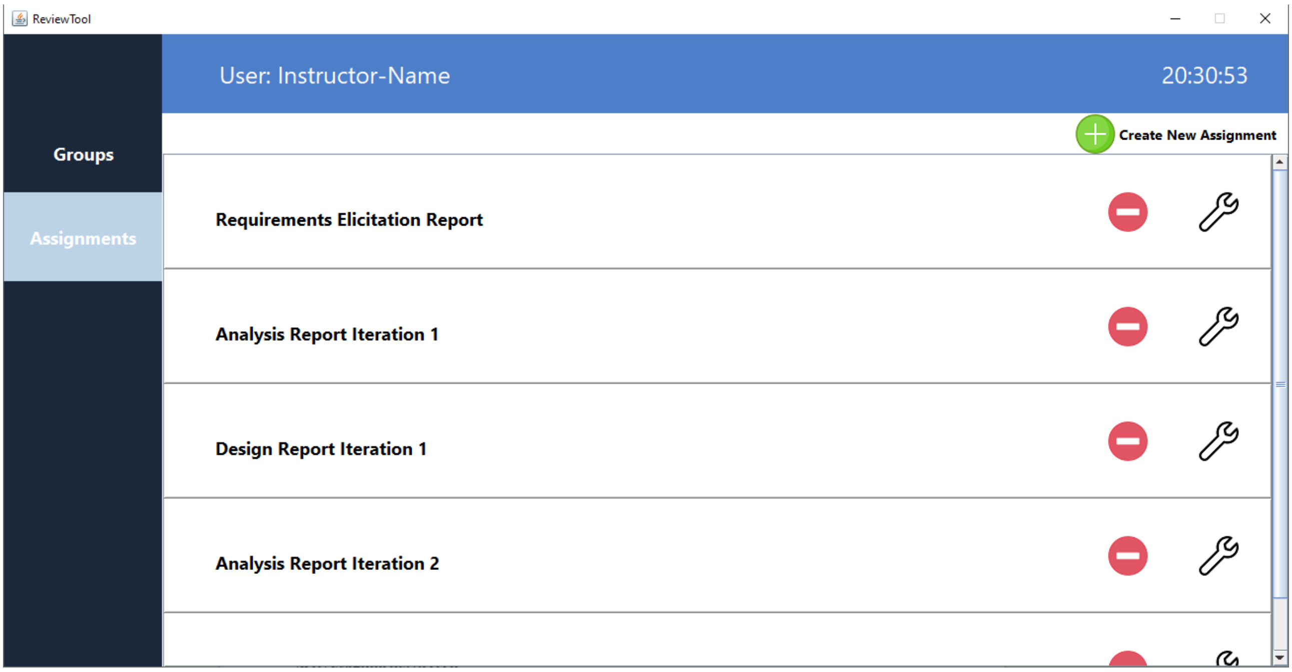
Point: 8/10

The instructor can select a “reviewer” from the list on the left and see the comment/grade given to the selected user.

3.3.2.3) Viewing / Adding Artifact Reviews

This screen could not be implemented yet as of this report’s writing, but it will be accessed from the “Selected Group” screen.

3.3.4) Assignments Screen



Requirements Elicitation Report

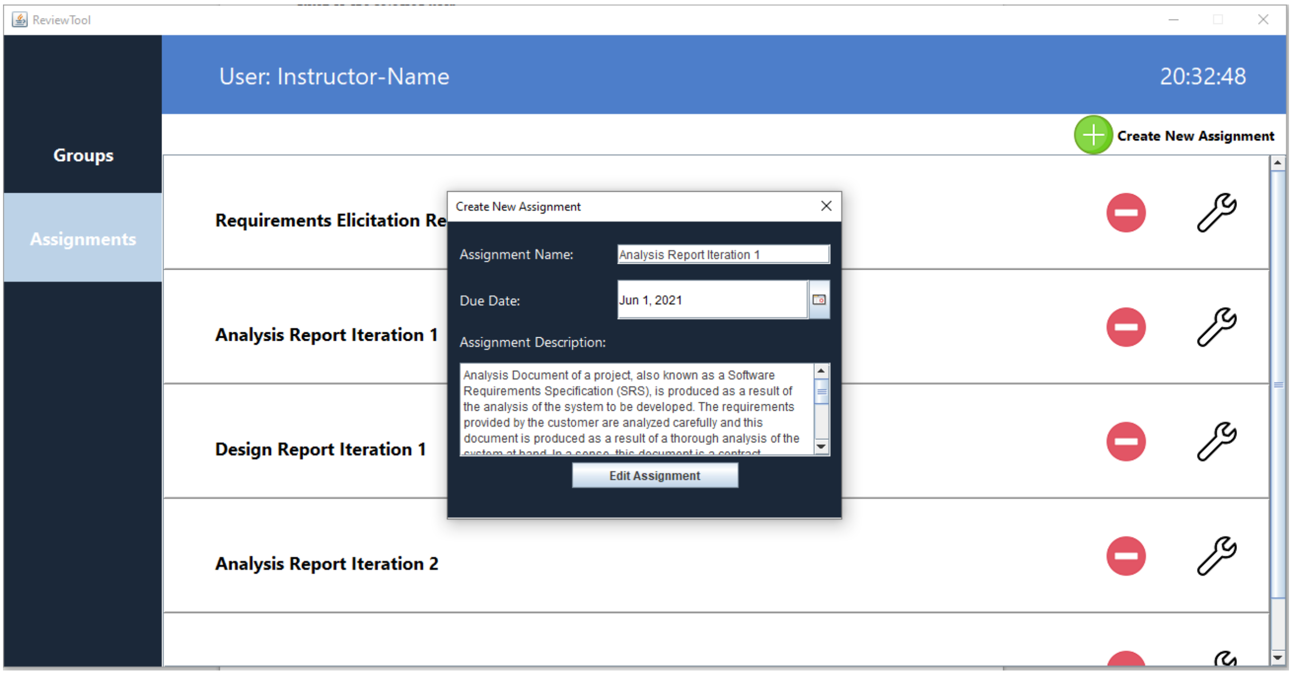
Analysis Report Iteration 1

Design Report Iteration 1

Analysis Report Iteration 2

The “Assignments” screen can be accessed from the left panel. It shows the assignments created by an Instructor that was stored in the database.

3.3.4.1) Adding/Removing/Editing Assignments



Requirements Elicitation Re

Analysis Report Iteration 1

Design Report Iteration 1

Analysis Report Iteration 2

Assignment Name: Analysis Report Iteration 1

Due Date: Jun 1, 2021

Assignment Description:

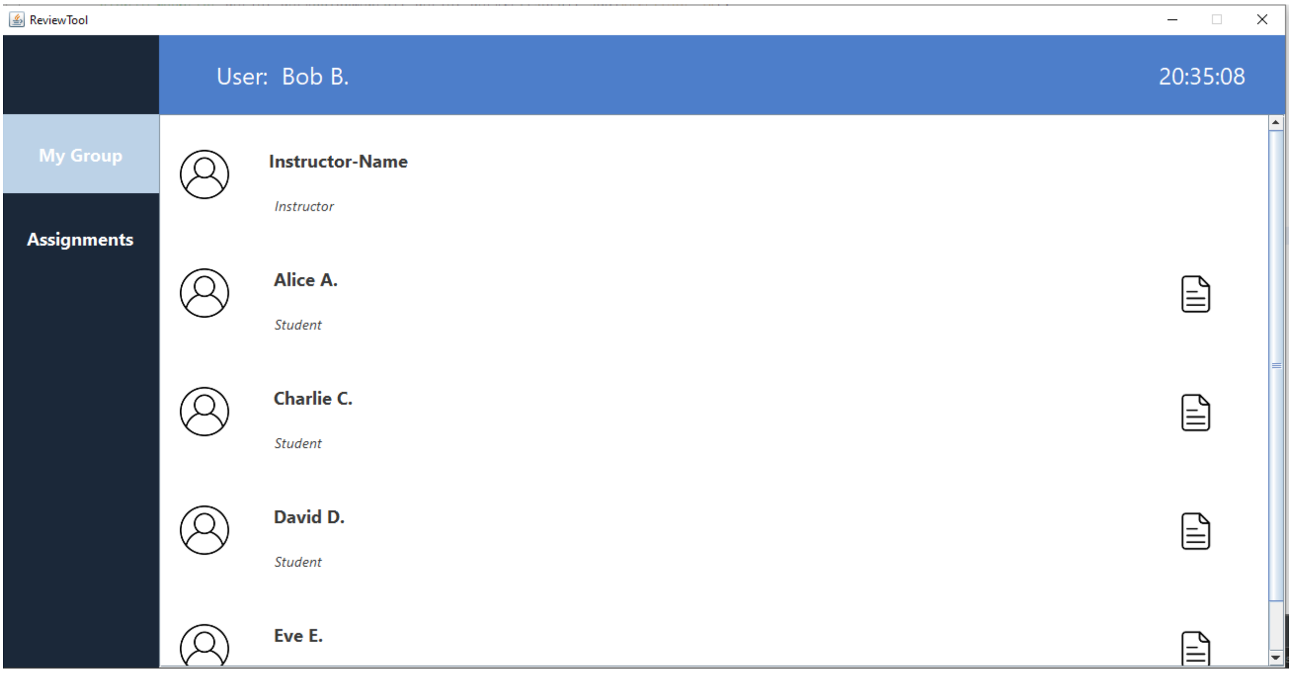
Analysis Document of a project, also known as a Software Requirements Specification (SRS), is produced as a result of the analysis of the system to be developed. The requirements provided by the customer are analyzed carefully and this document is produced as a result of a thorough analysis of the system at hand. In a sense, this document is a contract

An assignment can be removed with the red minus icon.

The “Create New Assignment” and “Edit Assignment” (wrench symbol) buttons prompt the Instructor with a pop-up, where he has to enter an assignment name, an assignment description and a due date.

3.4) Student Dashboard

3.4.1) My Group Screen



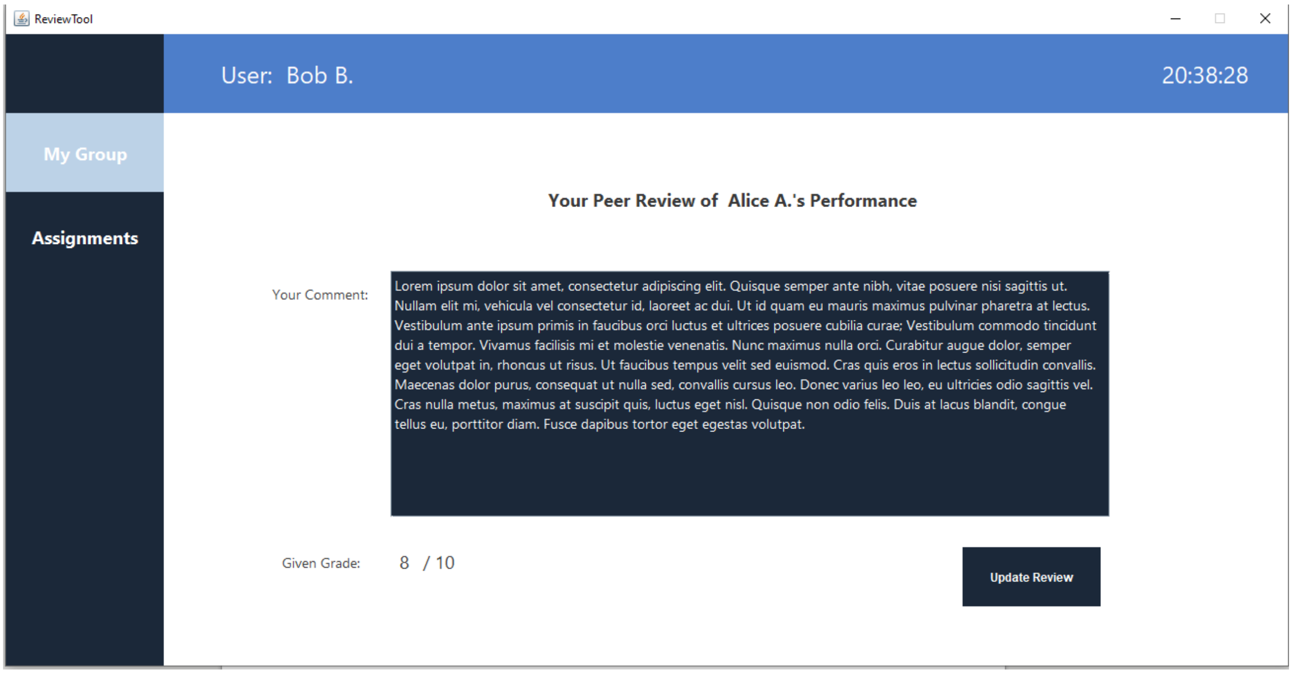
A student will land on the “My Group” screen after login. If the student is not assigned to a group yet, they will be prompted with an error.

The layout of the dashboard is identical to an Instructor’s, with the left panel providing navigation and middle panel providing interaction with entities.

The member list is displayed to the student, along with any Instructor or TAs that has access to the group.

Clicking the page icon near a Student takes the user to the Peer Reviewing screen.

3.4.2) Peer Reviewing Screen



User: Bob B.

20:38:28

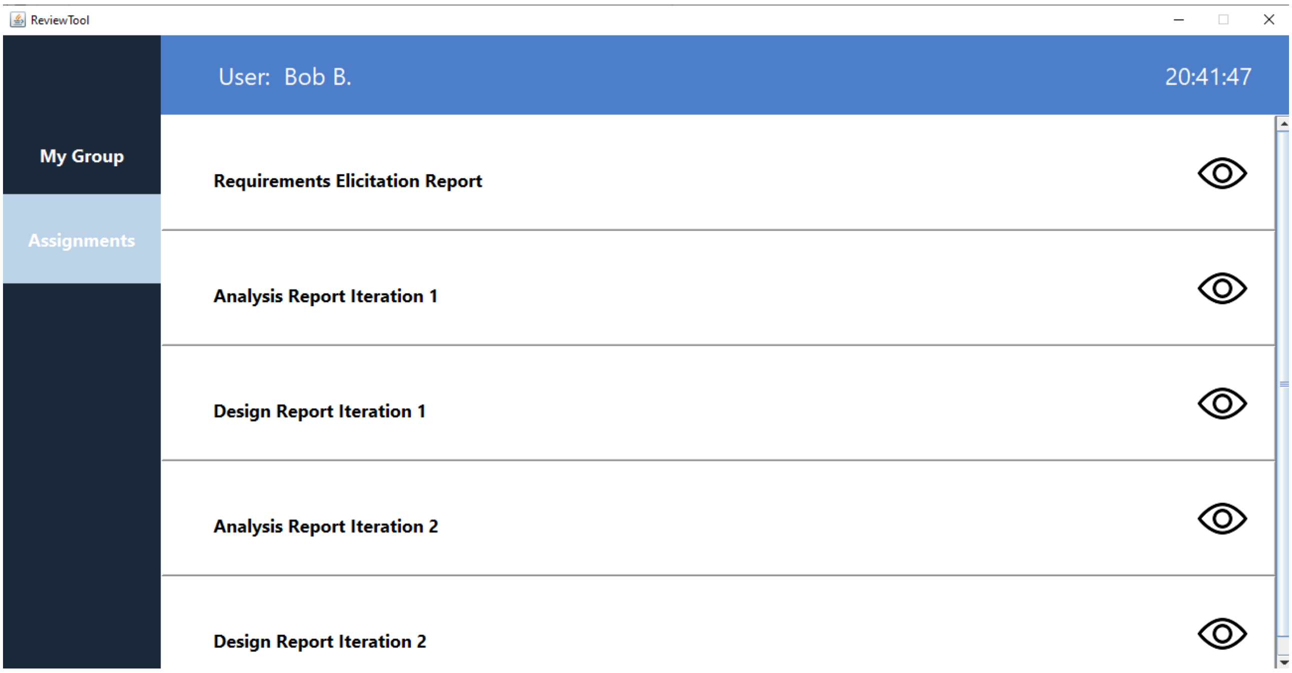
Your Peer Review of Alice A.'s Performance

Your Comment: Lorem ipsum dolor sit amet, consectetur adipiscing elit. Quisque semper ante nibh, vitae posuere nisi sagittis ut. Nullam elit mi, vehicula vel consectetur id, laoreet ac dui. Ut id quam eu mauris maximus pulvinar pharetra at lectus. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia curae; Vestibulum commodo tincidunt dui a tempor. Vivamus facilisis mi et molestie venenatis. Nunc maximus nulla orci. Curabitur augue dolor, semper eget volutpat in, rhoncus ut risus. Ut faucibus tempus velit sed euismod. Cras quis eros in lectus sollicitudin convallis. Maecenas dolor purus, consequat ut nulla sed, convallis cursus leo. Donec varius leo leo, eu ultricies odio sagittis vel. Cras nulla metus, maximus at suscipit quis, luctus eget nisl. Quisque non odio felis. Duis at lacus blandit, congue tellus eu, porttitor diam. Fusce dapibus tortor eget egestas volutpat.

Given Grade: 8 / 10

Here, a student can enter a comment and grade for their fellow group member. Clicking the “Update Review” button will change an existing review, or create a new one if there isn’t a review made yet.

3.4.3) Assignments Screen



Requirements Elicitation Report

Analysis Report Iteration 1

Design Report Iteration 1

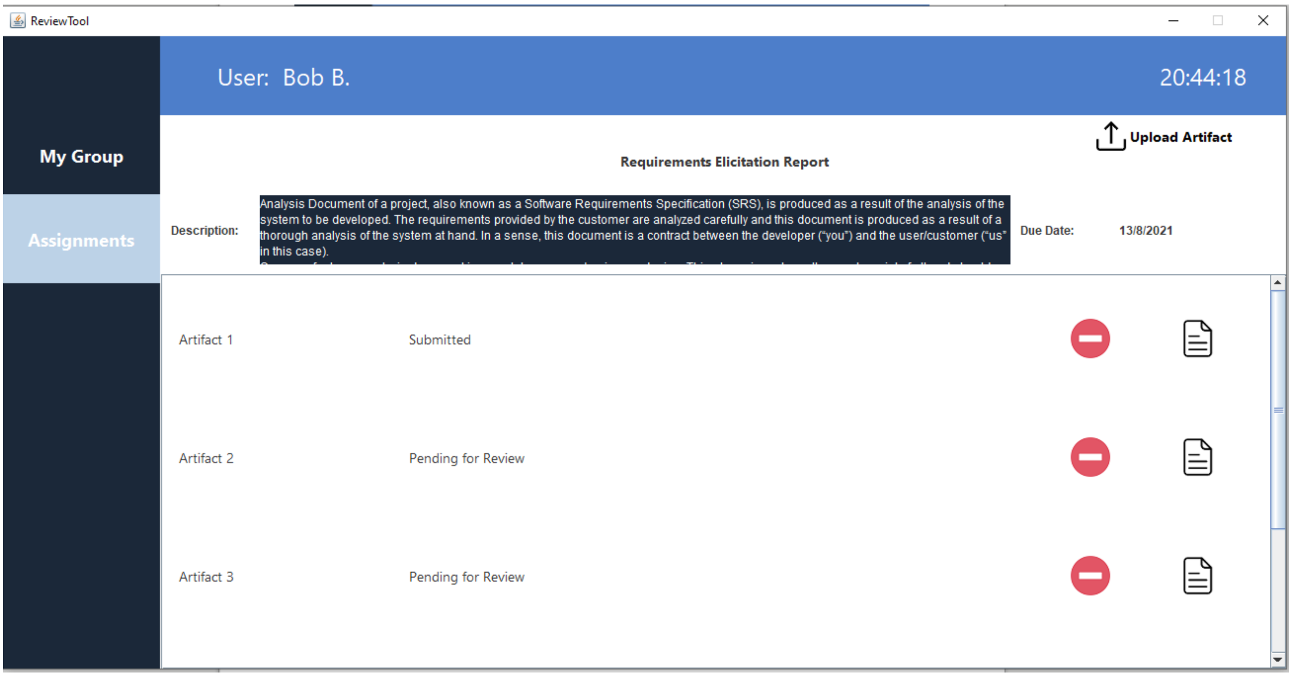
Analysis Report Iteration 2

Design Report Iteration 2

This screen is identical to the one in the Instructor screen, with the exception of managing the assignment of course.

The selected assignment screen can be reached using the eye icon.

3.4.4) Selected Assignment Screen



Requirements Elicitation Report

Description: Analysis Document of a project, also known as a Software Requirements Specification (SRS), is produced as a result of the analysis of the system to be developed. The requirements provided by the customer are analyzed carefully and this document is produced as a result of a thorough analysis of the system at hand. In a sense, this document is a contract between the developer ("you") and the user/customer ("us" in this case).

Due Date: 13/8/2021

Artifact 1	Submitted
Artifact 2	Pending for Review
Artifact 3	Pending for Review

The "Selected Assignment Screen" provides the user with assignment description, due date and artifact information/status.

The Student can upload an artifact (a PDF file) using the "Upload Artifact" file on the top right.

The Artifact list can be seen below. An artifact will be "Pending for Review" if no fellow student reviewed the artifact. An Instructor or a TA cannot see "Pending for Review" artifacts. When reviewed, the artifact will be "Submitted" and visible to everyone.

The "Artifact Reviews" screen can be reached using the page icon, but it is not implemented yet as discussed in the Instructor's User Guide.

3.5) TA Dashboard

The TA dashboard is a cut-down version of the Instructor dashboard, with certain features missing such as “Managing Assignments” or “Enabling/Disabling Peer Reviews”. We did not see the point in repeating the same instructions provided in the Instructor’s User Guide.

4. Build Instructions

1 Install Java JDK:
  - 1.i Go to: “ https://www.oracle.com/java/technologies/javase-downloads.html” which is the oracle website for downloading java JDK. Currently (3/05/2021) latest version is Java SE 16.
  - 1.ii Click on JDK Download link below Oracle JDK header.
  - 1.iii Download the correct product based on your operating system.
  - 1.iv Run the downloaded executable file.
  - 1.v Add the Java installation folder to your computers PATH variable.

2 Install Microsoft SQL Server on Windows:
  - 2.i Go to: “ https://www.microsoft.com/en-us/sql-server/sql-server-downloads” which is the MS SQL Server download page.
  - 2.ii Pick Developer or Express version and download.
  - 2.iii Run the executable file and follow the instructions.

3 Install Microsoft SQL Server Management Studio (SSMS) on Windows:
  - 3.i Go to: “ https://docs.microsoft.com/tr-tr/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver15” which is the SSMS download page.
  - 3.ii Click on the download link under Download SSMS header.
  - 3.iii Run the executable file and follow the instructions.

4 Run SSMS and Connect To MS SQL Server:
  - 4.i Run SSMS.
  - 4.ii Enter “localhost” as the “Server Name” and click Connect.
  - 4.iii Expand the server seen on the left panel and right click Databases to select “New Database…”

5 Create Database on MS SQL Server
  - 5.i Enter “CS319Project” as “Database Name” and click “OK”.
  - 5.ii Click the refresh icon under “Object Explorer”.

6 Open initialData.sql:
  - 6.i Click “File” on the top left of the program.
  - 6.ii Select “Open” → “File”.
  - 6.iii Find the project folder and select initialData.sql

7 Run initialData.sql:
  - 7.i Select “CS319Project” from the dropdown menu right of “Execute” button on the menu.
  - 7.ii Click “Execute” on the menu.
  - 7.iii After execution you should be able to see all tables with their initial data.

8 Compile source code and create a new executable jar file
  - 8.i Open Eclipse IDE for Java Development.
  - 8.ii Select “File” → “Open projects from File System…”.
  - 8.iii Select the Review Tool folder from the downloaded file.

8.iv Click Finish.
8.v Right click “Review Tool” in the “Project Explorer” screen and select “Export”.
8.vi Expand “Java” and select “Runnable Jar File”. Click “Next”.
8.vii Name the jar file as “CS319Project_Group1J”.
8.viii Choose “Export Destination”;
8.ix Select “Package required libraries into generated JAR”.
8.x Click “Finish”.
8.xi Find the CS319Project_Group1J in the selected export destination and run it.

5. Work Allocation

5.1 Analysis Report

For the first iteration of Analysis Report our group consisted of four people. We did the necessary parts for the report together using Discord.

For the second iteration of Analysis Report our group consisted of three people. We did the scenarios and the use case model together. For the second iteration work allocations are described below:

- Anıl: Introduction, Proposed Systems, Sequence Diagrams
- Göktuğ: Object Model, Activity Diagram, State Diagrams
- Cihan: State Diagrams, UI Mockup, Object Model

5.2 Design Report

For the first iteration we had some communication problems. The first report was written by Anıl.

For the second iteration work allocations are described below:

- Anıl: Introduction, System Architecture, Hardware-software mapping, Persistent Data Management, Access Control And Security, Global Software Controls, Boundary Conditions, UserInterface Subsystem, Class Diagram of Interface Package
- Göktuğ: Subsystem Decomposition, Business Logic Subsystem, Database Subsystem, Database Diagrams, Class Diagrams for Business Logic, Class Diagrams for Entities
- Cihan: Design Goals, Criteria For System Design, Business Logic Text, Database Text, Trade-offs, Business Logic Class Diagram Text, Entities Class Diagram Text

5.3 Final Report

- Anıl: User Guide, Demo Video
- Göktuğ: Step by Step Instructions, Work Allocation, Introduction, Demo Video
- Cihan: Lessons Learnt, Introduction, Demo Video

5.4 Implementation

- Anıl: User Interface and Implementations
- Göktuğ: MS SQL Server, DAO Objects and Implementations
- Cihan: Controller Objects and Implementations