# CS400 Project Two Proposal

**GROUP: KD FLIPGRID: https://flipgrid.com/spr21kd**

**TA: Keren TA EMAIL: kchen359@wisc.edu**

**RED TEAM (Role: Name - Email):**

**Data Wrangler:**

**Backend Developer:**

**Frontend Developer:**

**Integration Manager:**

**BLUE TEAM (Role: Name - Email):**

**Data Wrangler：**

**Backend Developer:**

**Frontend Developer:** Zhi Zheng - 4. zzheng94@wisc.edu

**Integration Manager:**

**Project Title:** Trivia game

**Brief Project Description:**

<brief project description here: 1) what will the app that you are developing do, and 2) who would use an application like this and why>

The trivia game app will have multiple choice trivia questions for a user to answer. The app computes a score based on the number they got correct and then stores the user’s name and score into a red black tree sorted by the scores. It will return the name, score, and rank received as well as the top three scores stored in the red black tree when prompted. This application would be used by anyone who loves answering trivia questions and is a little competitive because of the ability to see your ranking as well as if you placed in the top three.

**Data Wranglers:** Minyi Li Jack Gundrum

<brief description of the Data Wrangler’s role and responsibilities in this project>

Collecting questions and answers to organize a database that can be used for the back/front end. Writing Question class to create objects that contain questions, correct answers, and incorrect answers.

**Application Data:**

Trivia Questions will be searched for online, and then added to a database of questions and answers

**Data Format:**

Questions Interface- Has methods to return actual questions, getCorrectAnswer, getIncorrectAnswer

Trivia Interface- ArrayLists of questions, Types of questions, Score of user, username

**Data Wrangler Deliverables and Deadline:**

<describe at least five tests that your working implementation will pass>

* testGetScore
  + This will test the current user's total score based on how many questions they got wrong vs how many they got right.
* testFileReader
  + Check the correctness of data passed into the file reader
* testGetIncorrectAnswer
  + This will test my getter method to see if the incorrect answer stored is what I expect it to be
* testGetCorrectAnswer
  + This will test my getter method to see if the correct answer stored is what I expect it to be
* testListOfQuestions
  + This first test will check the implementation of obtaining questions of trivia from a database I create

<describe how and when your code will be shared with other team roles, note this should be before the group project deadline>

Since that data will be used by the backend, and then consequently the frontend, it should be completed at least a week before the project is due. This gives us time to fix any bugs that might show up, and it will allow time for other team members to complete their project.

**Back End Developer:** Muhammad Ismail Ryan Stevenson

The backend developer’s task for this project is to develop methods that help the user guess answers and store their points inside a Red Black Tree.

**Data Processing:**

The program gets user score points from the trivia questions and stores those points in the structure of a RedBlackTree while maintaining its integrity with rotations and recoloring, making sure there are no violations in the RedBlackTree.

**Front End Interface:**

<describe the Java Interfaces that expose this functionality to the front end of your application>

Interface includes functions that allow the user to access data about their score and the correctness of their answer. Some methods include getScore(), and correctness(question, answered).

**Back End Developer Deliverables and Deadline:**

<describe at least five tests that your working implementation will pass>

* Test on whether the root value is correct
* Test on insert method of the Backend class
* Test on search method of the backend class
* Test the contains method and check if the correct score relates to the right person
* Test the integrity of the RedBlackTree whether it has any violations or not.

**Front End Developer:** Almann Brague Zhi Zheng

<brief description of the Front End Developer’s role and responsibilities in this project>

The role of the Front End Developer is to integrate the Game class which will be responsible for taking the inputs of the user and being the main interface. This will be what asks questions to the user and it will take the input of the answer from the user. As well as it will be able to keep track of the users score which will come from the backend. Another thing that will happen is all questions and answers will be stored into a list by the Front end developer.

**Integration with Back End:**

<brief description of how front end and back end will work together and how the front end will instantiate the back end>

Use score.class (Backend) to store and get the scores and ranking for the user. This is the main interaction between the Back End. Also there will be an integration with the data wrangler because they will be developing the Question class which the Front end will also be responsible for integrating.

**User Commands:**

<brief description of the different modes or screens of the front end and the primary commands that users of your application will invoke>

After beginning the app, the user would receive each question and answer (correct or incorrect). The user will use the command line to type the answer they would love to choose.

After finishing all the questions, the users would be asked whether they would like to get the rank for their scores. The user can type y or n to receive/not receive their rankings.

**Front End Developer Deliverables and Deadline:**

<describe at least five tests that your working implementation will pass>

* Test on the Questions output for the user

-This tests that the frontend will print the welcome message and the number of questions.

* Test on the score ranking after the user wants to see his/her ranking

-check whether the system prints out the ranking of the user after receiving input y/Y

* Test on the get scores for user

-check whether the system shows the score of the user after the user finishes all the questions

* Test on the Name inputs stored in Frontend class

-Check whether the name is stored after the user types in his/her name

* Test on the transition between question

-check whether the system moves to the next question after receiving one of the choices in the multiple-choice question.

-check whether the system stores the answers in the list

<describe how and when this code will be shared with other team roles, note this should be before the group project deadline>

The front end class will implement score.class from Backend and trivia.class from data wrangler. Contains a main method and uses other team members’ codes to give the user a play setting.

The Front End Developer will implement this interface and test the implementation until Fri, March 19. The implementation will be ready early enough so that other team members will have sufficient time to test their implementations against it in addition to running their code against their own dummy implementation of the front end.

**Integration Manager:** Brenna Buck Alan Jordao Cortez

<brief description of the Integration Manager’s role and responsibilities in this project>

The Integration Manager will help other team members clear integration questions and make sure that the development stays on track. They will be responsible for the final integration of the three parts into one application. They will also create a short video demonstrating their team's implementation.

Specifically, the Integration Manager will familiarize themselves with all the interfaces and requirements described in the proposal. They will assist the other team members with implementing their dummy classes and will help discuss and resolve any integration questions. At the same time, the project manager will be responsible for scheduling team meetings during the implementation phase and regularly check in with team members to coordinate the development and integration of all the components. Finally, the Integration Manager will work with their teammates to assemble all the parts of the application and write a Makefile for the project that compiles, runs, and tests the system. Once the system is integrated and fully functional, they will prepare and submit a demo video for the submission and upload it to FlipGrid.

**Integration Milestones (team blue):**

<list brief summaries for the team’s internal deadlines through the final week of this project’s development>

Kickoff:

Tasks:

1. The integration manager is available for his teammates to ask any early questions about integration and development of the dummy classes.
2. If some team members feel their job is heavier than others, the team can rearrange some contents in the project to make sure each member does contribute.
3. Data Wrangler giving other team members an overview of how they would receive the data

Due date: 03/11 emails, snapchat, meeting(if needed)

Milestone 1

Tasks:

1. [Data Wranglers](https://canvas.wisc.edu/courses/244796/pages/movie-mapper-data-wranglers) will try to finish their questions interface/object
2. Developers will create dummy classes to test their implementation of their own objects and interfaces.
3. The integration manager will familiarize himself with all the interfaces and classes in the system.
4. Clear any question that developers have after a few days of working on their implementations.
5. Explain how each part of the system works and talk about how to integrate each part.
6. Solving potential integration problems.

Due date: 03/16/21 meeting for that

Milestone 2

Tasks:

1. All three parts of the system will be ready.
2. Developers and Data Wranglers try to run each part individually in the meeting.
3. The Integration Manager begins to integrate.
4. The remaining integration questions will be discussed.

Due date: 03/19/21

Milestone 3

Tasks:

1. The system will be integrated and the Makefile will be ready to compile, run, and test the system.
2. The integration manager shows the team the process to run the whole system.
3. Fix last integration bugs.
4. Help the integration manager decide on the contents of the demonstration video.
5. Then the integration manager will finish the demonstration video.

Final due date: 03/23

**Integration Milestones (team red):**

<list brief summaries for the team’s internal deadlines through the final week of this project’s development>

Data Wrangler gathers data and ready to share with teammates by 03/16

Workable code that passes individual tests get shared to Integration Manager by 03/19

Final due date: March 23rd

**Demonstration Video:**

<describe how you plan to demonstrate the capabilities of your application in the final demo video for this project>

For the final demo video we will have the integration manager run through how the final application works. We will do this through screen sharing and a live camera of the integration manager who will be explaining each step of the process. To start we will introduce the initial home screen of the application and run through the options the user has. Then from there we will navigate through the program showing off the different functions of the application. Some of these functions will include showing how users can search through the entire library of movies individually. As well as showing how users can narrow their search results based on the specific movie genres they prefer or to narrow it based on the ratings that the movies have received. Once this recording is complete we will upload the final video to FlipGrid.

**Additional Responsibilities and Notes:**

<list by role, any additional responsibilities that are expected of team members to help balance the workload for this project… if you are concerned about your project being too simple or too involved, this is also a good place to suggest plans for expanding or contracting your main idea>

* Generally for all roles try to meet their given deadlines. If you are having difficulties ask the person who has the same role as you, or ask our TA, Keren, to see if he can help.
* Every role should stick to what it says on their giving canvas page on what to do. Doing so will allow the workflow to be evened out. The instructors assign these roles for a purpose, and we know that we will be given different roles each project. Therefore, if we each do what it says on our role’s page, for each project, it is likely that all will have done roughly the same amount of work(specifically in the different projects where it seems like role A did way less work than role B, switching roles evens it out). Help from other team members is always appreciated though. For example, If someone is the Integration Manager role they can suggest their own helpful tactics to other roles should they need it.

**Team Signatures:**

Each team member must type their own name and current date in place of one of the blanks below, to confirm their contributions and agreement to the proposal described above.

Zhi Zheng 3/8/21

**End of Proposal**