Alyssa Dang (626006043)
Elizabeth Park(825006152)
Hard Patel(425004549)
Eric Anderson (425005256)

Survey

	Was the project fun and interesting?	**Did it provide opportunities to innovate?**	**What went well? What didn't go well?**	**What lessons did you learn that you would share with your team?**	**What one topic do you want to make sure we address in the retrospective meeting?**
Alyssa Dang	I think learning something new was interesting since we haven't been introduced to databases yet, but the problems we experienced were not interesting.	There were some opportunities to innovate with the tables we chose for the databases and how we implemented the GUI, but time and space constraints hindered some things we could improve.	Phases 1-3 went well. Phase 4 was a struggle to finish because of how hard it was to work on it together. We need to start using GitHub for version control.	We need to test our program better and start working on it earlier. We need to use GitHub so that we're all working on the current version of the code. We need more time for group meetings.	We should establish a time for group meetings. I don't think what we've been doing, just meeting between class and lab is sustainable. We should also do better at dividing work and communicating our tasks.
Elizabeth Park	It was interesting to work with database and sql commands.	Yes, I got to know more about databases and how they are managed by working on the project.	Phase 1 to 3 went well and Phase 4 was complicated. We had to put in a lot of time to finish the codes.	I learned that we should divide our roles more beneficially. We kind of worked on the codes independently then shared with others rather than working together from the start.	I think it would have been better if we had more knowledge about sql commands. Sql examples on the presentation was too basic to use in the project.
Hard Patel	The project at first sounded very interesting. Phase 1 to 3 was fun to work on; although phase 4 was a little bit stressful due to some technical difficulties.	It provided me with a great opportunity to innovate and create algorithms using Java and SQL. It also helped me familiarize myself with new	Phase 1 to 3 was the best part of the project and finishing phase 4 was the most stressful and was hoping to have more time to complete and	I learned that we should have used GIT to share and work on our code. We should have also given more time to group meetings and brainstorming	There should be a brief review of the programming language we will be using in the project. Also, it would be helpful if there was more

		coding techniques.	perfect it.	ideas. However, I am happy with the work done by the team.	information related to the project.
Eric Anderson	The project was interesting in setting up a database and was a struggle when the server crashed resulting in a lot of our testing to be done on AWS.	The project allowed us to try new things and figure out ways around obstacles especially when the server crashed. Alyssa built an AWS server for us to test our code on.	We worked well with the time given although we all had very busy schedules during the harder phases which was very difficult. We worked well together in helping each other where we could.	I learned the benefit of version control. We used Google Docs as our storage method. If we had used Github we would have a better method of seeing what has been changed and the current up to date version instead of guessing which is the newest name of our java file.	Having maybe a week and a half for due dates would have been beneficial because we would be assigned something new on tuesday and then the assignment wouldn't be fully explained until Thursday so having a bit more time once we fully understand the assignment would be great. I understand why it was only a week per phase though.

Satisfaction:

The team consensus about the IMDb database project is that while the information we learned and the type of work that this deals with is interesting, the implementation was often tedious and many problems arose that could not be helped. It would be more satisfying if the database was more stable and we had a little more knowledge about what was happening beyond the commands that were given to access and manipulate the database. Information like what databases are, how they work, how they're created, and more context would've made the project more impactful. That said, we also learned a lot and this is a good introduction to PostgreSQL and working with a database in general. The problems that we witnessed and experienced also provided some perspective on trying to create a product that we're working in a team to deliver.

Opportunities to Innovate:

Since this project was pretty open-ended, there were many opportunities to innovate and customize our application to how we want it. Places where innovation comes to mind is the implementation of the GUI and its use of PostgreSQL. There wasn't any instructions or requirements about how to implement our product, just that it delivered what the stakeholder wanted so there was a lot of leeway and flexibility there. Our GUI ended up being a simple

window with a search bar and a search button and a drop-down menu for the categories to search from, but if we wanted and if time allowed, we could've expanded the design without cost to the requirements given. There were also many ways we could've implemented the tables in the database. As long as the data was kept, we likely could have found many innovative ways to organize the database to make our queries more efficient.

What Went Well and Not Well:

Things that went well includes adding our data to our database, creating our GUI, connecting the two, and testing that portion of the deliverable. We were able to do simple queries and have some basic functions like searching for TV shows, movies, people, and so on. However, after that, we experienced problems implementing the degrees of separation problem in part four of the project. Some of the members in our group were able to complete parts of the questions they worked on, but this was the hardest part of implementing the application. In the end, the database went down before we were able to finish so we have partial algorithms and code for deliverable 4 part 1, but the outlook for completing what we set out to do does not look good as of the code due date October 15, 2019.

Lessons Learned:

The lessons learned are primarily related to working in a team. There were some stressful moments during the project because the deadlines were typically one week from the assignment of the problems and it was difficult to find good meeting times or times to work together within that one week since we all have full schedules with work and other responsibilities on top. This did not hinder our progress however, since members worked on the project when they had time and communicated what they were doing in the GroupMe. On some weeks, a team member would be particularly more busy so other members would take on more work and the next week, the member would take on more work so that effort would be distributed fairly.

Topic That Should Be Addressed:

One topic we should discuss more is if we should change how we approach the assignments. As of now, we rarely meet up physically to work together. If the problem or tasks are big enough, how should we approach it so that one person is not doing the bulk of the problem or stuck for a long time trying to figure something out until the next person comes to try to work on it. Should we all meet up before tackling the problems ourselves? Should we do all help on each task or divide it completely to each member? How do we work more efficiently so that when a big deliverable comes up, we can complete it in a timely manner?

Retrospective

Project Overview:

For the IMDb database project, we created a database that contains six tables. These tables include information like movies, TV show episodes, their crew, the people in the movies and shows, the ratings, and other information. Using these tables, we created a Java application that accessed the database from our computers and returned information based on what the user searches. The current version of our application can search for movies, TV shows, people, ratings, the degree of separation for two degrees, and partial non-tested features that were ideally supposed to be implemented in deliverable four, but the database is unavailable.

Engagement Analysis:

Usually, when the team is meeting, members are engaged, but the assignment wasn't exciting in the way that encourages extra work and engagement outside of meeting times and the required work time. Overall, everyone is satisfied, but effort was more based on the deadline and the tasks requested rather than members' enthusiasm for the database and the product. The opportunities to innovate were largely overshadowed by limitations such as time, the reliability of the database, the struggle of implementing the tasks, and general confusion about the product and what everyone was doing and how to tackle problems at times.

Product Analysis:

We got most of the functionalities working and our GUI works well for what we have.

Key Points To Remember:

We need to do better at dividing tasks and setting goals of when to complete them by so that we do not fall behind if multiple people attempt the same thing. Our team also needs to start using github. Thus far, we've uploaded our code onto a shared drive, which doesn't allow much for documentation and version control. Additionally, we need to start meeting earlier if possible so that we do not need to have an emergency session right before the deadline when something doesn't work. Although our team members all contribute to the project, we can vastly improve our progress if we organize better.

Work Effort Analysis:

The most progress was done during group meetings. It was sometimes hard to complete tasks individually because some things were dependent on each other and when members ran into problems, they would have to troubleshoot it themselves.

Tasks:

Alyssa Dang:

1. Updating shared documents such as class notes (SQL commands, Instructions, etc.), design documents, and deliverables documents
2. Designing the GUI
3. Create the first part of the Java GUI (Connecting it to the database via VPN, adding a JFrame window, search bar, and getting input when the user clicks the search button)
4. Creating a new database using AWS to test the code and help with deliverable 4 when the school's database crashed
5. Added searching the database from a given rating
6. Helped debug added code related to the GUI
7. Bonus question 4 of deliverable 4 and helping other team members

Elizabeth Park:

1. Helped adding the data tables and data to the database
2. Designing the GUI
3. Create the second part of the Java GUI (getting queries from the database and displaying it, searching the database for a certain movie, TV show, and person) and adding a combination box
4. Helped debug code in relation to the GUI and passing SQL commands
5. Adding all the tables and data into the new AWS database to be used for deliverable 4
6. Worked on questions 1 and 2 while helping other team members
7. Added to the deliverables documents

Eric Anderson:

1. Helped adding the data tables and data to the database
2. Verifying queries and sharing the document with the team
3. Added to the Appendix document and the updated document with the new tables design
4. Documenting the changes from design to implementation with the data tables and the GUI
5. Added to the deliverables documents
6. Worked on question 3 while helping other team members
7. Helped debug SQL commands and members deliverable 4 code

Hard Patel:

1. Created the database (with his credentials)
2. Figuring out how to add data to the created tables and adding the data tables and data to the database
3. Helped members connect to the database and use SQL commands
4. Added to the Appendix document
5. Added to the deliverables documents
6. Worked on question 1 from deliverable 4 (and got the furthest out of all the attempts with it partially working)
7. Worked on question 2 from deliverable 4

Agenda
Meeting 10/17/2019 5:30pm

Discussion on what we should continue doing:

1. Keep talking on groupMe and being available after class on Tuesdays and Thursdays.
2. Working together and helping one another to achieve the set goals
3. Flexibility to help more when group members have difficult and busy weeks so our overall progress continues at the same pace.

Discussion on issues we had:

1. Ensuring we use GitHub for code improvement and a centralized location we can all work from.
2. Making sure to communicate on what we are doing and what we need help with
3. Finding an available time where we could all meet probably just staying after class on Tuesday and Thursday again

Overall improvements for next project:

1. Use Github
2. Make sure we all stay on the same page
3. Ensure we stay organized and work effectively towards the projects deliverables