Group project overview

What we accomplish:

For our group project, we decided to base our prototype on Jasmine's EDR, which uses the construction of nested HashMap that fetches past occupancy data on student tenants' usual timetable as to when they enter and leave the shared accommodation, in order to gives the students task-arrangement suggestions on the best time for them to do certain activities (e.g. using shared facilities or doing house chores). Our main objective is to optimise communal space usage and housework scheduling. To make the workflow clear and organised, we outlined a clear plan on how to approach the goal of the prototype step-by-step and split the workload in a manner that played to everyone's strengths. Saabarin was best with understanding HashMap and how they work, so she was responsible for building of the HashMap and implementing the JDBC connection whereas Asmaa was given the role of building the database using PostgreSQL and Jasmine was on calculation duties as she was best with the mathematical and logic side of the system. This method of having individual responsibilities that work to everyone's strengths and weaknesses helped ensure a smooth collaboration between the group.

How was it accomplished:

Plan:

Before setting out to do the project, we outlined the complete procedure in approaching the prototype construction stage-by-stage. We decided on the waterfall model for developing our prototype, as it would allow us to understand the exact tasks we need to complete, to meet our target of creating a well-functioning program. Thus, based on the different phases to follow and the estimated time required for fulfilling each of them, we as a group came up with a timetable that list out the deadlines for every stage of our project.

A Software Requirements Specification was produced during the requirement analysis stage, demonstrating our understanding on what our prototype should do (functional requirements) and the desired qualities of our program, like performance and accessibility (non-functional requirements). After the completion of the first stage, we started to convert the requirements into a software design. By referring to the EDR and considering carefully on optimizing the group's strength, we made the decision of coding our prototype in Java and using SQL as the foundation for our database, so that we could benefit from the simplicity and accessibility of building a JDBC connection in between the database and our Java program.

Before moving on to the implementation stage, we assigned each individual positions based on their strengths. We then created a git repository where members were able to commit their changes through pull requests and consistently update and make amendments to our code by using git commands and merging whilst allowing the other

members to track their progress. Furthermore, we also incorporated a kanban to be informed of the status of our work and see which tasks were in progress or are completed. We also regularly held meetings to ensure our prototype was progressing smoothly and to assist each other with any problems that may occur while programming.

Nested HashMap:
First, we had to establish the connection between PostgreSQL and the Java file had to be established. To do this, we installed the JDBC file from Postgres so it could be added into the external libraries for application. Then for the nested HashMap, we combined the day and hour columns into a single key that would be used for the outer layer and the username and present count was stored in the inner layer as its own HashMap. Due to the changes made in the database, the present count was modified into becoming a list of integers as the keys in the inner and outer layer are repetitive, this enables the same keys to have a list for the presence count that can be used later for the calculations.

Calculations:
The calculations for thresholds, which are required for setting the conditions for a suggestion to be given, alongside the computation of mean for each hourly time slot (needed later for comparison with thresholds to produce suggestions) is also a key component of our prototype. Multiple adjustments on the details of implementing the calculations were made, as we encountered potential problems with our original plan. To avoid the possibility of thresholds surpassing the boundary of [0,1], additional codes were added to clip upper thresholds to a maximum of 1 and lower thresholds to a minimum of 0. The original default values for scale factors K1 and K2 was 1, but according to a test run, we found it more appropriate to set them as 0.5 instead. Additionally, some planned mechanisms were simplified simplification to make programming more straightforward and achievable: Instead of including a dynamic algorithm to track the total number of weeks in our database, we chose to hardcode the value, which reduces the complexity of implementation while guaranteeing accuracy in calculation results.

Testing:
After we finished each section, we decided to implement unit testing. This would ensure each part of our project works as it should. By building tests using JUnit, we managed to implement them to each of our methods. Later in the stage of integration and system testing, we tested the performance of the entire product, which appeared to be efficient. If any of our tests failed, it allowed us to go back and check our work to identify any error that need to be fixed.

Build tools:
We also incorporated Gradle in our system to help process our code and allow our code to compile faster, resulting in our program becoming more efficient. It was also able to

detect problems that could exist in our software enabling us to be able to swiftly fix any errors, accelerating the development process.

Refections:

Reflection for Saabarin (from Yifan):

Saabarin is responsible for constructing a nested HashMap using Java programming, which involves establishing JDBC connections for building the link between our database and HashMap to be created. By flexibly applying the knowledge on HashMap methods like put() and get(), alongside a clear understanding of the EDR and the goal to achieve as a team, Saabarin successfully managed to code and build the nested HashMap as required (that has the structure of having 'date' as outer-layer key and 'username' as inner-layer key, with each pair of keys having a one-to-one correspondence to their respective list of data 'count').

Saabarin also helped Suhayr with setting up an SQL database and actively contributed the to error-tackling and troubleshooting the malfunctioning part of our program when we started the integrating and testing stage of our project.

Saabarin is a highly self-motivated and responsive individual that is a pleasure to work with. She actively attend group meetings and is willing to take responsibility, tackle unexpected problems and learn new skills, showing a high level of adaptability and resilience. She is able to manage her time wisely and balance the workload, especially when dealing with multiple tasks (e.g. when she need to work on both HashMap coding and research into the EDR while coping with other module assignments). Saabarin is also showing capability of comprehending complex logics (by having an in-depth understanding of the structure of our project via comprehending the explanation I gave for our project and the EDR description). Furthermore, Saabarin also showcased her problem solving skills and ability in giving clear and concise oral expressions, by constantly raising constructive suggestions during our group meetings.

I would suggest Saabarin to continue being self-disciplined and motivated in future projects / academic learning, with focus on persistently enhancing her professional skills on coding and project management.

Reflection for Asmaa (from Yifan):

Asmaa is responsible for building the SQL database for our project, which is a simulation of the SSH Cloud database that our program is dependent on. By utilizing the knowledge of SQL keywords like CREATE, INSERT, SET and VALUES, Asmaa demonstrated a solid understanding of basic SQL implementation. Asmaa also managed to manually add all data from an existing table to PostgreSQL using SQL queries, after building a .sql file. This illustrates her diligence and resilience in achieving set goals.

Furthermore, Asmaa shows patience and motivation when dealing with upcoming unexpected errors and new things to learn. With good communications and clear expressions for queries and issues she has been facing, we are able to have effective conversations that allow better understanding of her situation, enabling the rest of the group to offer support whenever needed.

To improve the productivity in the future, I would suggest Asmaa to focus on the following points:
1) Do regular revision on learnt modules (especially the highly useful and practical modules like Object Oriented Programming and Full Stack Application), so that there is a wealth of consolidated knowledge / skills available when faced with a challenging and highly-integrated project like this one.
2) Better time management is key to group projects; try to balance school projects / other module assignments and other job duties beyond curriculum, so that a reasonable amount of time would be given to each piece of task and the allocated work could be completed on time, leaving more time for project integration and further testing. Try to attend more group meetings, especially when you had agreed on the time of the meeting.

Jasmine (from saabarin):

Jasmine was a proactive member of the team, who oversaw the coding for the calculations that would be used in our prototype. As our prototype was formed from the basis of her EDR, she was able to explain her ideas about how she wanted the prototype to work, resulting in us being able to make an effective plan for developing the prototype. In addition, she was able to answer any questions we had about the prototype and explain where we went wrong, which was able to help minimise confusion and maximise use of time working on the project.

Jasmine attended all meetings, was able to complete all her work diligently and effectively and communicated well by keeping the group updated with any new changes that she may have added whether it was her coding or to the overall prototype

development. Her attention to detail is excellent and was able to identify any work that may have not been done well or cause an issue in the future and was able to provide solutions for those problems. In addition, she also displayed empathy and was accommodating to our needs, adapting to certain situations when all group members could not attend. She was also able to display her proficiency in java as her coding for the calculations were shown to be competent and was able to work well in our prototype.

Jasmine was a valuable member of our team, and I am looking forward to working with her again in the future.

Asmaa (from saabarin):

Asmaa took the role of building the database, through this she was able to demonstrate technical proficiency by using SQL to code for the database that will be utilised in our prototype. She was an active contributor and always communicated with the team about her role, her responsibilities in the team and was able to ask for assistance when it was needed. This proved to be effective as she was able to complete her work, and her database has shown to be compatible with the prototype we have created. In addition, she was very open to feedback and incorporated all the changes that were ask of her, always taking responsibility for her mistakes, learning from them and using them as a way to further improve herself in any future task she received. Moreover, she demonstrated flexibility as she was able to quickly adjust to any changes that had to be made over the course of our time working together which was an extremely beneficial to our team as we were constantly adapting our prototype and the database throughout our time together.

To improve, she could focus on working on her time management by making adjustments to her timetable, so she is able to finish her work in time and be more consistent with updating the team on her what she was doing.
Overall, she was a great asset to the team due to her ability to continuously adapt to feedback and to the changes that were made to the project.

Jasmine (from Asmaa):

Jasmine was very pleasant to work with as she understood the importance of collaboration. Her EDR was the forefront of our project, she had to spend a lot of her time answering questions from both me and Saabarin, her explanations were always concise and clear; her patience was the most important factor in understanding the criteria and implementation of our project.
Her calculations were based on the database I created, so it was important that I always fed back to her my progress; this was done through our WhatsApp group chat and our shared google doc.

Technical feedback I would give back to Jasmine would be to make her code more understandable using comments and general readability.
A non-technical feedback I would give to Jasmine is to make use of more collaboration tools such as Jira or Slack to keep a more detailed track of all work done, any upcoming deadlines, etc; collaborating tools would ensure a more thorough and clear communication is there between all members and would benefit Jasmine greatly.

Saabarin (from Asmaa):

Saabarin was tasked with building the Hashey for the database, this was a crucial part of our prototype to ensure data was read from the database correctly. She was a pleasure to work with as she was constantly available to ask questions and clarify any queries I had. I admired her resilience in troubleshooting technical errors we faced along the way.

Technical feedback I would give to Saabarin would be to try incorporate unit testing more with your code and to make use of frameworks such as JUnit (in Java) to execute this; this will help mitigate bugs and errors as they are identified during development.
A non-technical feedback I would give to Saabarin is to try to offer more feedback to your teammates along the way, constant communication on any issues she has would help foster a more cohesive unit within a team, as well as help identify any members who may be struggling with understanding of tasks; communication is always key.