Introduction

The pet grooming industry is a fast-growing sector, and pet grooming businesses are increasingly looking for ways to improve their services and enhance customer satisfaction. A pet grooming shop requires a comprehensive system that can efficiently store, process, and manage customer and manager data to ensure that it can provide top-quality services. The current paper-based system employed by the pet grooming shop is disorganized, making it challenging to maintain accurate records of pets and groomers. Additionally, the absence of a digital system has made it impossible for customers to access their records. As the shop's reputation continues to grow, an increase in demand is expected, and it was essential to implement a modern system to manage customer and pet information more efficiently.

Our team's solution was a pet grooming appointment webpage named Petgrm. This webpage considered the users to have 2 main characteristics. One is the administrator (shop manager), who maintains and updates the system, and the customers who want their pets to be groomed. The base programs used to create the back-end system were Java17 and Spring Boot. Additional programs for the database and front-end programming includes HTML, MySQL, JavaScript, CSS, and jQuery.

The main functions required of the system were that it is a web-based system that allows the booking of pet grooming appointments as well as help maintain the flow of data such as user info and appointment schedules that a paper-based system might struggle in as the business expands. Additional requirements include customers having the choice to choose their pet type, and a groomer (which has a ranking system) from the venture and the allowing of cross and upselling strategies based on the past user data.

In my perspective our project PetGrm accomplishes the basic functions that is required of an appointment webpage but has room for improvement. To start off, in comparison to other pet grooming webpages, it is very easy to attain information that a user might want. This is due to the fact that it is very easy for the user to ask questions on the webpage if the info they are looking for is not uploaded as well as the Q&A and FAQ sections of the webpage are viewable as soon as the user logs into the site. Another strength of Petgrm is that the system is user friendly and has a simple process of making appointments for their pets. To add on, there are a vast amount of breeds that can be chosen when choosing the dog type without having to manually type in extra information regarding the breed of dog.

However, I believe that the system lacks some major features at the period of its release such as the ability to find IDs and passwords if a user has forgotten the information or the ability to move back to the homepage during the appointment creation stage. Furthermore, data regarding the groomers are not shared as much as they should be.

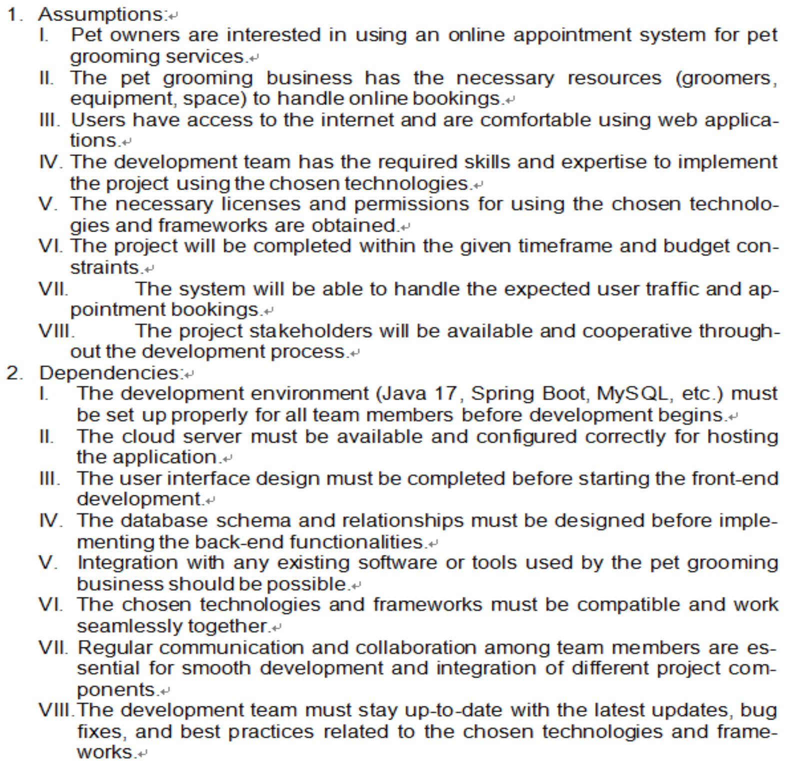
Conclusion

The project on making a webpage with a devoted team has allowed me to understand how the scrum process works on a more hands down level. Understanding and engaging in time distribution for creating the required features within the sprints that were set was also a valuable experience because it is a needed skill in future projects. The notion of a semester long project in itself was also new and challenging in regards to time distribution as the team and I had to set goals and deadlines by ourselves and had to push each other to control progress that was made. Also, I was able to encounter and learn the fundamentals of many development tools that I had no experience with in the past such as CSS, HTML, and Spring Boot. Overall, the project was very fruitful in both software development skill improvement and in personal discipline.

An improvement that I would make in future works is in making a more systematic work ethic in meeting certain task deadlines. This is because the time use in the project was tending to get sporadic mid-semester and I think better time management would have enabled our team to create more features to add onto our project in the long run.

Appendix

Screenshot6. Later version of assumptions & dependencies



1. Assumptions:
   - I. Pet owners are interested in using an online appointment system for pet grooming services.
   - II. The pet grooming business has the necessary resources (groomers, equipment, space) to handle online bookings.
   - III. Users have access to the internet and are comfortable using web applications.
   - IV. The development team has the required skills and expertise to implement the project using the chosen technologies.
   - V. The necessary licenses and permissions for using the chosen technologies and frameworks are obtained.
   - VI. The project will be completed within the given timeframe and budget constraints.
   - VII. The system will be able to handle the expected user traffic and appointment bookings.
   - VIII. The project stakeholders will be available and cooperative throughout the development process.
2. Dependencies:
   - I. The development environment (Java 17, Spring Boot, MySQL, etc.) must be set up properly for all team members before development begins.
   - II. The cloud server must be available and configured correctly for hosting the application.
   - III. The user interface design must be completed before starting the front-end development.
   - IV. The database schema and relationships must be designed before implementing the back-end functionalities.
   - V. Integration with any existing software or tools used by the pet grooming business should be possible.
   - VI. The chosen technologies and frameworks must be compatible and work seamlessly together.
   - VII. Regular communication and collaboration among team members are essential for smooth development and integration of different project components.
   - VIII. The development team must stay up-to-date with the latest updates, bug fixes, and best practices related to the chosen technologies and frameworks.