PUrfect Dining

TEAM 23 : Product Backlog

Team Members: Aashir Aumir, Arjun Srinivasan, Eric Baeza, Kailasam Sriram, Tung Dao

Problem Statement:

Currently, Purdue doesn't have a system for students, faculty, and the general public to rate their experience at dining courts, or to make informed decisions about where they wish to dine on campus based on real-time statistics of other users' dining experiences. Additionally, dining courts are also unaware of their customers' opinions about their experience. PUrfect Dining gives users the ability to rate their dining experience at a specific dining court after completing their meal effortlessly. It also provides them with the opportunity to provide insightful comments to help dining courts improve their service via our website and Android application.

Background Information:

Audience:

Purdue's dining courts are used on a daily basis by students and faculty. On special occasions such as Thanksgiving, the general public of Lafayette and West Lafayette also eat at the dining courts. Additionally, visiting families of Purdue students also dine at the dining courts during their stay. With such a large volume of people from various places visiting dining courts, a centralized rating system will be a useful tool to aid in making informed decisions.

Existing platforms:

Existing platforms such as Yelp and Zomato that are similar to PUrfect Dining allow users to rate and comment on their experience at various establishments and restaurants across the world. However, they do not support Purdue's dining courts on their applications. These applications provide users in Purdue and the Lafayette area with information about restaurants on campus and around the world.

Limitations of similar existing platforms:

- Services such as Yelp and Zomato require users to create accounts to rate on various establishments
  - Our goal is to make rating as simple and effortless as possible for diners while also collecting data efficiently. All diners will be able to vote on their experience while exiting dining courts with our 3-button system. However, to account for reliability of specific comments, users will have to create accounts.
- None of existing services provide information about the dining courts at Purdue
  - PUrfect Dining provides real time and historic information about the dining courts at Purdue. Users will also have the option to elaborate by commenting on our website or app.
- The Purdue Menus app provides personalizes a user's experience by allowing them to choose favorite items at dining courts. However, users cannot rate on dining courts or comment on their experience.
  - With PUrfect dining, users will be able to rate and comment their experience. They will also be able to view statistics based on other users' ratings. With access to this and the comments made by users, they can make informed decisions.

Functional Requirements:

As a user, I would like to:

1. Rate my dining experience in an effortless manner after a meal.
2. Receive audio feedback when the button is pressed to confirm that my rating has been recorded.
3. To view the different dining courts.
4. To view the different meal times of the dining courts.
5. To view the menus of the dining courts.
6. To have a simple guidance that shows up when I open the app for the first time.
7. To see top trending dining courts that day.
8. To view statistics in the form of graphs.
9. The graphs to displayed in an easily understandable manner.
10. The graphs to distinguish dining courts by color to make it more appealing.
11. To choose my favorite type of graph to view (bar or pie chart).
12. To view real time statistics of dining court ratings.
13. To view historic statistics of dining court ratings.
14. To view trends in dining court ratings over different periods of time.
15. To view the top 3 dining courts overall at the end of the month.
16. To know what the best dish from the previous week was.
17. An Android app version of PUrfect Dining.
18. To view the statistics through my phone in an easy-to-access manner.
19. To view other user’s opinions on items.
20. To be able to register for PUrfect Dining account.
21. To be able to login.
22. To be able to logout.
23. To manage my PUrfect Dining account.
24. To be able to recover my PUrfect Dining account’s password.
25. To provide further comments on a website or app which help elaborate on my rating.
26. To receive a visual confirmation of my comment submission.
27. To view comments that I have made in the past.
28. To like other users’ comments.
29. To reply to other user’s comments.
30. To be able to report inappropriate comments.
31. Comments with obscene language to not be displayed.
32. The freedom to view the data available to me in different ways.
33. To customize which features I want to be notified about.
34. To set preferences for my favorite dining court(s) and receive a notification about the current ratings for that dining court (breakfast, lunch, and dinner).
35. To set favorite meal time and receive a notification that tells me the best dining court at that time.
36. To share statistics of dining courts with friends via social apps.
37. To share comments of other diners on social media platform.
38. To upvote/downvote individual food on the menu.
39. To utilize the functionalities that navigation bar provides.
40. To view the dining courts sorted by those that are open and closed respectively.

41. To view the open dining courts sorted by ratings for the current meal time.
42. To be able to report bugs/suggest features for the app.

As a dining court representative, I would like

40. To receive and view the feedback from diners.
41. To improve my service based on feedback of users.
42. To view which individual food the diners prefer the most and which one they would like improvements in.

Non-functional Requirements:

Architecture

This project has mainly two divisions: software and hardware.

The software division is divided into three sub divisions: Web, Android and Pi. Each of these will act as a client for communication with our backend. Our backend, Backendless, provides APIs, both RESTful and native, for receiving and handling requests from any client. Also it provides query-based search which enables us to present a data analysis in many forms.

The web client is divided into two further divisions: frontend and backend. Keeping both independent of each other will allow us to manage tasks easily and efficiently. For the frontend, we’re making use of Bootstrap to make the website responsive and the Angular framework to retrieve data from the backend. As for the backend itself, we are using NodeJS as a server along with the Backendless integrated database.

The Android client will make use of this API for synchronizing data with the backend and Realm, an OO database, for storing data offline. The app will make use of the Model-View-Presenter framework for easily testable and readable code, keeping the UI and data separate. The app will be built using Reactive Programming (RXJava), performing all network and data operations asynchronously. Both of these design choices allow the app to run on low-end devices as well as it would on high-end devices.

The 3-button system will be built using a Raspberry Pi and hardware accessories and extensions that include: a wifi modem, three buttons, a button interface, a speaker, a few LEDs and wires for putting it all together. On the software side, we will be using the Backendless Java SDK for communication with the server and a local Realm instance (similar to the Android setup) to store ratings to sync with the server in the event that internet is unavailable.

Security

For security, all hardware devices built will have a kill switch in the event a device is lost or stolen so our collected data is not damaged or exploited. Moreover the device(s) will allow new input only when the dining court that they are placed in is open and serving food. This will prevent unwanted interferences that may affect the data. On the backend side, some global objects and all tables will maintain an Access Control List (ACL) which determines read/write permissions for each type of user. To ensure that all reviews posted on the website/app are reliable, users will have to create accounts to comment. These accounts will be password protected to ensure security.

Usability

Both the website and the Android application interfaces will be simple so that the users can easily navigate through the app and understand the features. Both of these will be flexible enough to provide the users with some customization, such as choosing their favorite type of graph (bar or pie chart) or changing some of the notification options. These graphs will be color Moreover, the platforms interfaces between diners and dining representatives should be similar so that both parties can be familiar with the system and access the information in the same place, though some user permissions are different for each group. Furthermore, we want the users can access PUrfect Dining on all screen sizes, and that is the reason why we are going to make the website responsive for all popular devices. Last but not least, on the hardware side, since we are giving the users three rating options to choose from and setting the button system right next to the dining court's exit door, the diners can rate their dining experience effortlessly.

Data Integrity

This project has two points of data entry: using the 3-button hardware and leaving comments on dining courts. The Raspberry Pi handling these inputs will be responsible for filtering subsequent intentional button presses by the same person, or in other words "fake ratings". Using a time delay between each button press and visually alerting the diner/user that we're ready for their rating and that their rating has been registered.

Availability

This project's focus is currently on the dining courts at Purdue. But the idea of data collection, data analytics and real-time data tracking has many different applications: restaurants at PMU, clubs, audience poll, live event statistics (PUDM, NASHA etc). This will be expanded upon towards the end of the project cycle.

Portability

This goes hand-in-hand with the previous mention of Availability. The hardware to be created is to be portable such that it is light hence easy to move while remaining sturdy enough to withstand falls and other types of damage. That it should be able to function if there is no ethernet cable available, moreover that it should function even when there is no internet connection available and save ratings offline and sync with the backend once a connection is established so no data for a specific time or day is lost.

Hosting/Deployment

We will use Heroku as the platform to host our website. Its free custom domain provides us with the freedom to design our website. As for the hardware, after prototyping the button system, we can try setting it up at a dining court, next to the exit door.