6.170 – Software Studio
Fall 2014
Abdi-Hakin Dirie

Project 3 Problem Analysis: FoodbackMIT

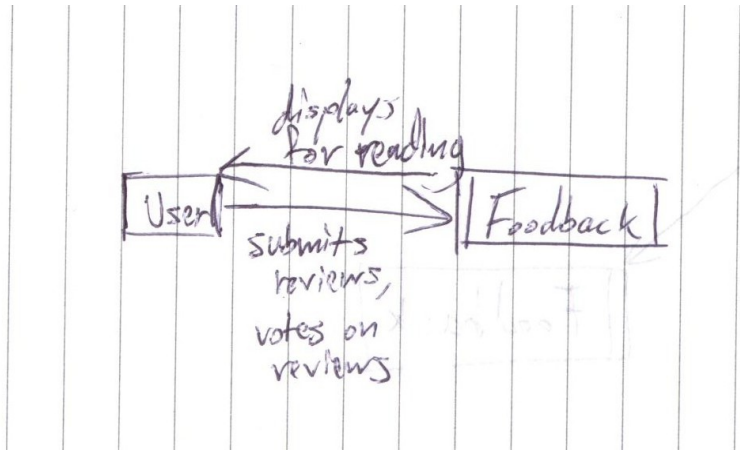
Description

FoodbackMIT is an online app for posting and reading reviews on MIT's dining hall services. Users of the app can post reviews on specific dishes, dining halls, or meal hours, as well as agree or disagree with the reviews made by fellow users.

Purposes

- **Post reviews on MIT's dining halls.** The app aims to create a platform that wasn't previously available by make their opinions heard to a wider audience. Dining halls have small physical cards for writing short comments, but the primary audience is the dining staff. We believe the audience should be the diners, not the cooks.
- **Read reviews.** Diners can already read the comments of other diners, which are written on the aforementioned cards. However, this system requires the diner to be at a dining hall.. This is troublesome because it would mean a diner has used a meal swipe to enter the hall, which is especially annoying if you you are deciding to go to another, infrequently visited dining hall. The app allows for reading reviews regardless of location.
- **Confirm reviews.** The app allows for users to agree or disagree with reviews that they have read. This allows for certain reviews to be seen as more credible than others. For example, a review accusing a dining hall of labeling a non-vegan item as vegan should be expected to be corroborated by other users to be considered a legitimate complaint.
- **Recommend a dining hall or specific dish.** The app also serves as a way for users to laud certain dishes or services the encountered at dining halls. The app can display menus with dishes highlighted or dulled based on their popularity among diners.

Context Diagram



6.170 – Software Studio
Fall 2014
Abdi-Hakin Dirie

Concepts

Review: A textual piece written by users that applauds or criticizes an element of MIT dining, whether it be the food, service, dining hall, meal hour, atmosphere, etc. Unlike the physical comment cards that are posted on the wall in dining halls, reviews are meant to be more substantial in their content.

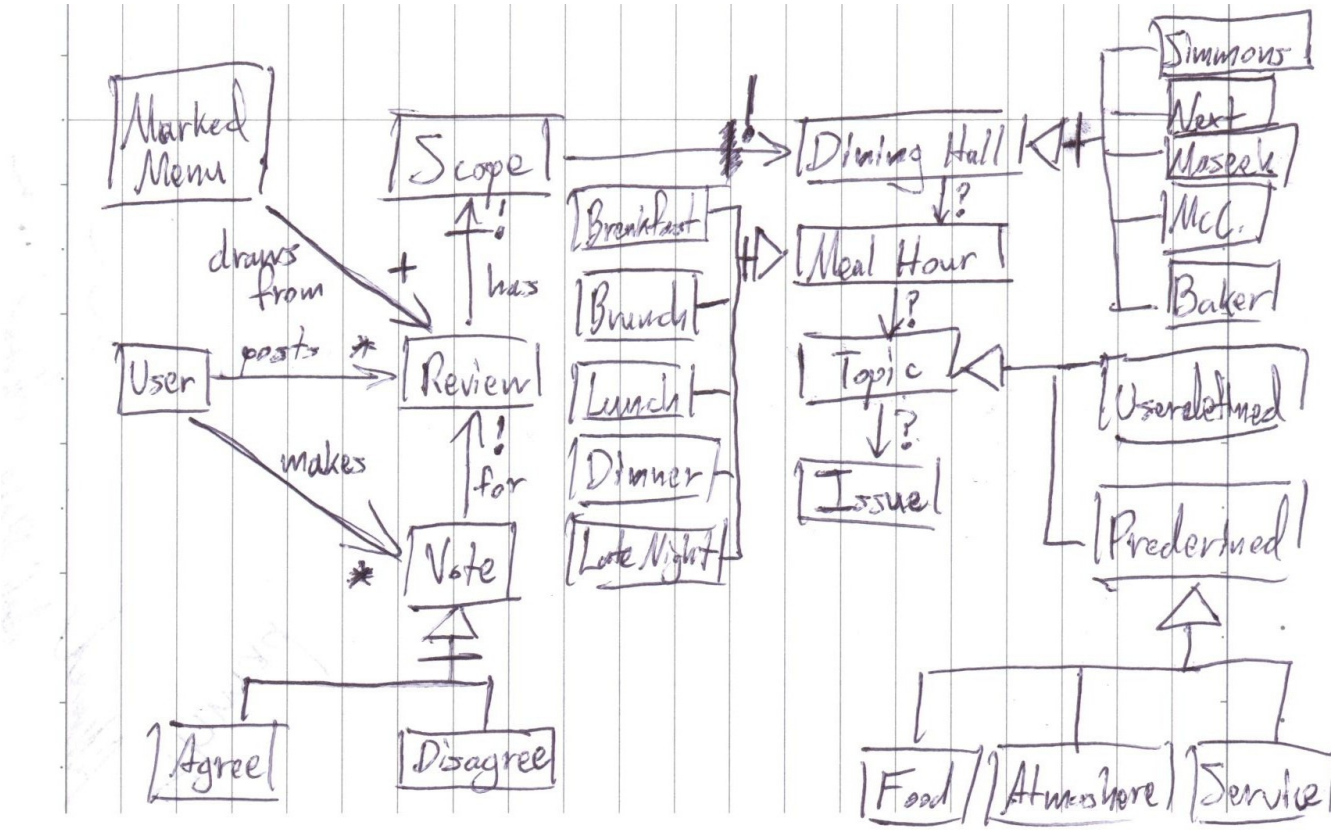
Agreements/Disagreements: Users can up-vote or down-vote a review depending if they feel the review is accurate. This concept adds a democratic element to the app, making collective community perception more important to the user than a single review. The app highlights accurate portrayals of dining while weeding out inaccurate ones.

Scoping: Reviews are given within a *scope*, which is a way to set different levels of specificity in a review. The top level, which is required, is the dining hall. The next level is designating the meal hour (breakfast, brunch, lunch, or dinner). Next is the topic. Some are predefined – food, atmosphere, or service – while others can be user defined. Finally, the user can take another step to specify exactly what they wish to review. This is dependent on what the previous scope was. For example, if it is food, then the user can choose to highlight a specific dish. If service, then the user can comment on the hospitality of the staff. By using scoping, users can narrow what they read to specific subsets of reviews. It also helps to prevent incredibly vague reviews from cluttering the app. The level of scoping a review uses is up to the author.

Marked Menus: A menu for a specific dining hall on a specific day at a specific meal hour that is annotated with data drawn from reviews. The data can be used to highlight especially good dishes (useful for recurring offerings), provide a rating on the service, and can even be used using aggregate reviews from certain time intervals to see how community perception changes over time.

6.170 – Software Studio
Fall 2014
Abdi-Hakin Dirie

Data Model



Design Challenges

How are reviews organized?

For reviews to be useful to other users, there must be a way for users to find them in a structured, non-convoluted way. Using simple categories intuitively seemed the obvious way to go, but it became clear that reviews might focus on a whole dining hall or a very specific dish. That is, some categories would inevitably have a large share reviews. Thus, using multiple categories seemed like the next best thing. However this was also troublesome. For example, most dining halls operate independently. They have different chefs, equipment, etc. Thus, a negative review on food without specifying which dining hall the meal was had in would negatively affect other dining halls, an unwanted side effect. Thus the idea of hierarchal categories, a.k.a. scoping, would solve the problem. Now a review on service would be forced to specify which meal (breakfast, brunch, etc.) the service was encountered, and which dining hall.

6.170 – Software Studio
Fall 2014
Abdi-Hakin Dirie

How are reviews responded to?

There were two routes for this: comment thread or a voting system. Each had their pros and cons. A comment thread would allow the reader to get more descriptive perceptions on how the community feels about the review. Votes are purely quantitative, aside from the positive/negative connotation attributed to each vote. However, for a community the size of MIT dining, any one user could find themselves reading a long comment thread, yet be indecisive on the issue. Votes on the other hand are easily decisive. The user only has to take into account what the review says and the net reaction from the community (represented as a number) to make a decision, as opposed to sifting through as whole thread.