Carrot Critic - “Carrot over Stick”

Software model: incremental – We chose this model because our product needs to be flexible with changing user requirements as we gather user feedback and make updates over time. However, it is very difficult to prototype a review app and we don’t have much need for risk management since our system is rather simple, so we didn’t want to use the prototyping or spiral models. The incremental model lets us create the product effectively without any unnecessary overhead.

Feedback address: Unlike most current review services C.C. will be anonymous for users. This feature will help our users avoid backlash, embarrassment, or bribery as a result of using the application. Furthermore, our reviews will be verified using two-factor authentication which will help keep out fraudulent reviews and help make our service much more reliable and trustworthy than its competitors.

Functional Requirements:

1. User can register an account once
2. User can add a review
3. User can search information for nearby businesses
4. User can search reviews for nearby businesses based on cost, distance, etc.
5. User can edit their previous reviews

Nonfunctional Requirements:

Product requirements:

1. The system will protect the integrity of user data (location, IP address, etc.) by storing user data on user devices
2. The system would be down only on Tuesdays 3am CST to 4am CST. Users cannot create new reviews during the maintenance period but can access database during this period.
3. Accessibility – We want to have a lot of visual and usability aids such as: Narrator to read all text on screen, Magnifier capability that will increase size of text and button as well as decrease if necessary, Inverting colors, optional dumbed-down UI with bigger buttons and less of the complex issues, visual tutorials,
4. The system will send push notifications weekly to the user and ask them if they would like to update their local database. The database itself will be tailored to their search history.

Organizational requirements:

1. Development – We will be using MVC, so the product will be organized and separate the inner workings of the app from the UI development.
2. Must work on both Android (6.0 and higher) and iOS (v10 and higher)
3. Application must request access to location services.

External requirements:

1. Write privacy protection technicalities into terms & conditions agreement for user
2. Data shall not be influenced by businesses outside of consumer input.
3. Account details will not be accessible by anyone but the account holder.

Software Architecture Model: MVC

