Milestone 3

Group 4

Sam Vensel, Matthew Higgins, Keven Fuentes, Michael Culleton

4/2/2017

CPSC 4140, Spring 2017

Usability Specifications

Quantitative benchmark tasks:

- Recording the time it takes for people to understand what to do within the application
    - How long does it take them to find a specific recipe? (30 seconds is too long)

  How long does it take them to find the action buttons that lead them to the next step?
- Record number of negative unsolicited comments (all are important)
- Record number of positive unsolicited comments (all are important)
- Record the number of times people need to go back to a previous screen or instruction (3 is too much)
- Record the number of times people clicked an un-clickable area (3 is too much)

These tests should illustrate the success of the design goals. We tried to make each step as streamlined as possible so that the screen would not be cluttered with useless information, but we also did not hide information from the user. We simply placed the buttons in strategic points that would make the most logical progression. If the user cannot find specific items in the menus or proceed steps in a reasonable time, the system design will need to be re-evaluated. Negative and positive unsolicited comments can help to show what was designed well and what needs more attention. If a user is frequently having to go back to previous steps in the process, this could indicate that the user was misunderstanding the goal of each step. An unreasonable number of "backs" could show that the goals of various steps are not clear and need to be re-evaluated. Recording the number of times a user tried to click something that is not clickable could help us identify if there needs to be a presentation change to the way items on the screen look.

Qualitative benchmark tasks:

- Give surveys at the end of the user's experience
    - Ask for their thoughts on various steps
    - Have them identify anything that frustrated them
    - Have them identify anything that they liked
    - Ask for any additional feedback that they might be willing to give to improve the application
    - Ask if they felt like they understood the goals of the various tasks and screens
- Record unsolicited comments
- Record facial expressions of the users

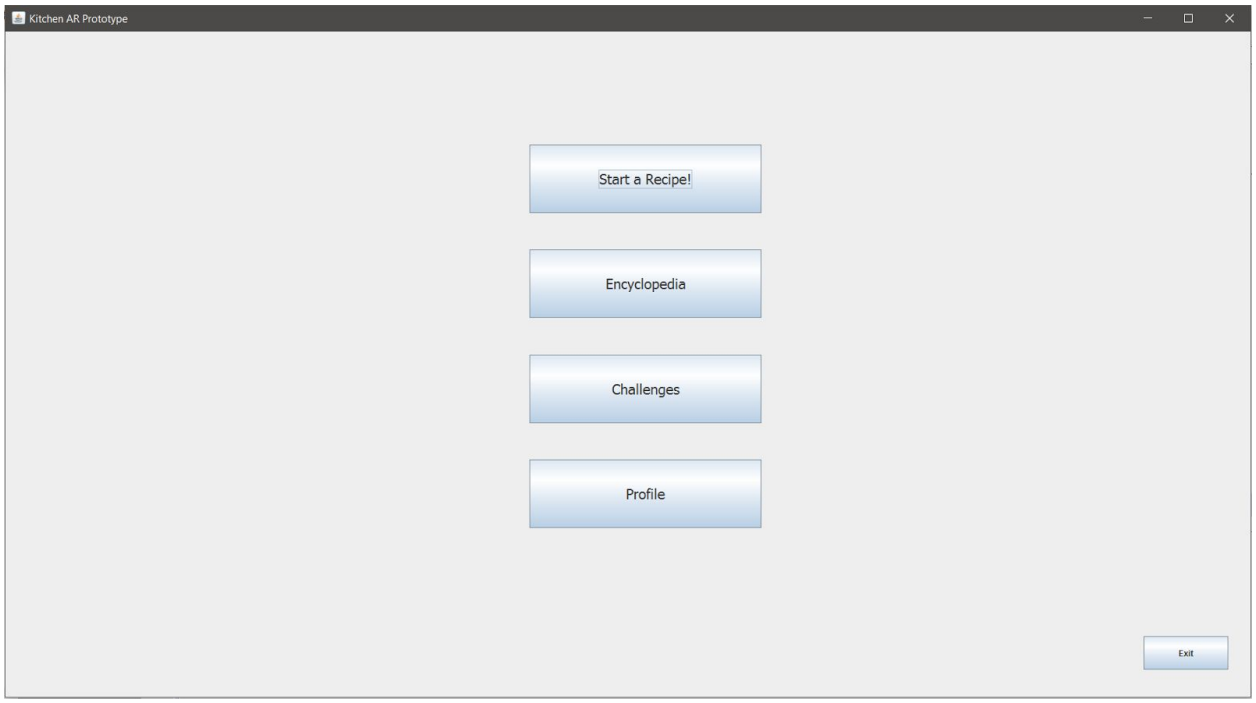
User feedback could help to point out the best and worst of the application. Getting an idea of what people liked and did not like can help us identify what we can do to make the poorly received components better. Identifying what users found most frustrating can help us determine if the feature could be better implemented or determine if it needs to be a feature at all. Asking

for additional feedback could help us get information on the application that we did not think to ask about. Also, it could help us get new ideas for new features or improved features in the future. The user needs to understand the goals of the various screens and steps, so confirming the user's understanding can show us what screens/steps need to be improved.

Prototype Overview

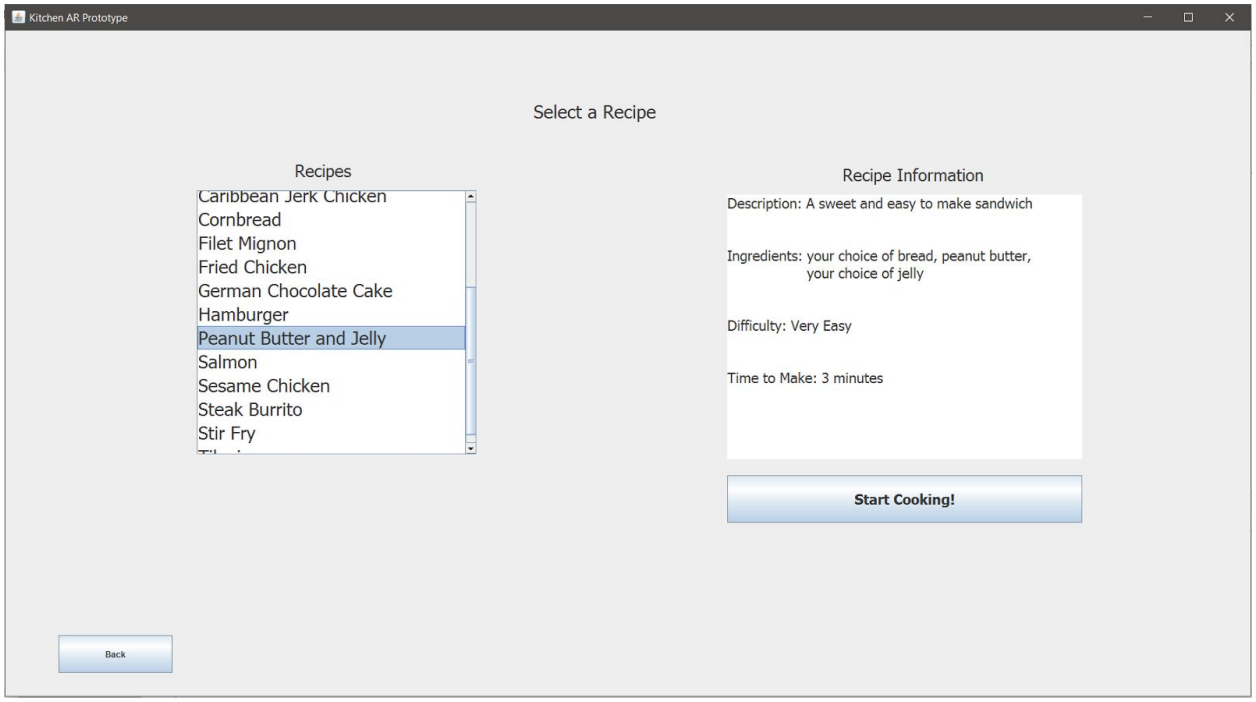
This prototype for an augmented reality cooking aid walks the user through the various steps of making a peanut butter and jelly sandwich, offering tips and tutorial videos along the way. As we did not have access to AR glasses or the knowledge required to develop in that space, it is a desktop application that simply serves as a proof of concept rather than a fully functional system.

Walkthrough:



When the user starts the application, they will be greeted with the screen shown above. In this prototype, only the "Start a Recipe!" button is functional, but the others represent functionality we would like to see in a final version. The prototype also lacks all of the theming we would eventually like to see.

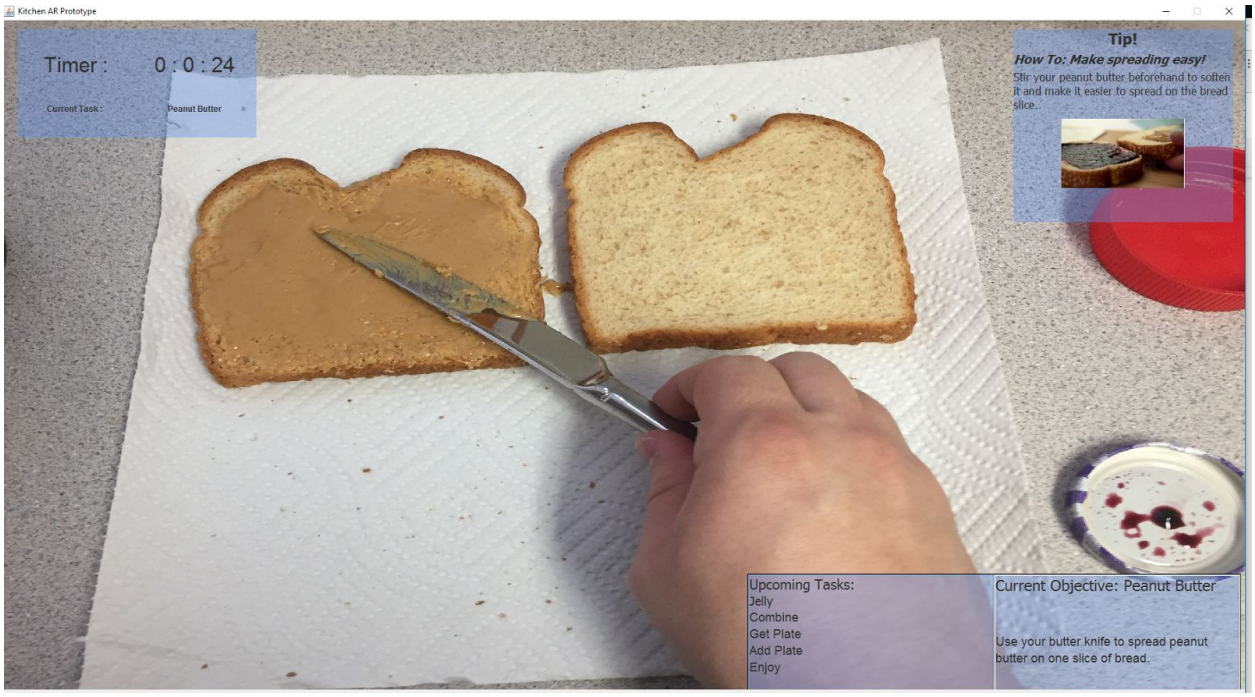
This menu would also not take place in the AR glasses, but on a companion phone app, as navigating menus and lists can be a challenge when just using AR glasses.



After clicking “Start a Recipe” on the previous menu, the user will then be taken to this screen. This is where they can choose a recipe from a varied list and view the information for that list like: difficulty, ingredients and time to make. The Peanut Butter and Jelly recipe is the only working recipe for this prototype. The others are placeholders. Once the user has selected the Peanut Butter and Jelly recipe, they will then select “Start Cooking!”. From here, the experience moves to the AR glasses instead of the companion phone app.

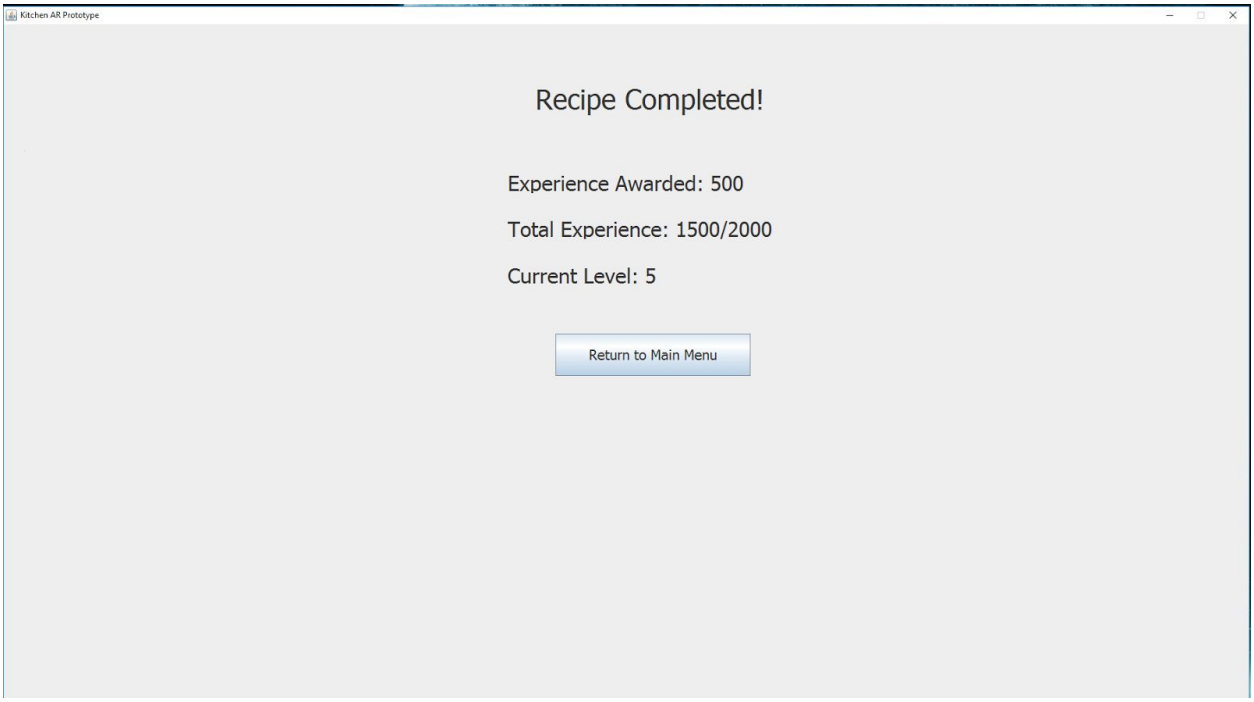
Once the cooking portion starts, the user will get this HUD shown above in their AR glasses. However, again, this is simply a desktop mockup so we simulated this by taking a picture for each step of the sandwich being made, using that as a background for the screen.

Each step will have a “current objective” panel in the bottom right, showing the future tasks as well as the current task and a brief description of that task. Some steps will also have a tip for the current task in the upper right corner. For the steps where the user spreads the peanut butter and the jelly, the tips window will also contain a tutorial video for extra hints.



Some steps will also have a timer, shown in the upper left corner. Naturally, time isn't much of a factor when making a PB&J, but the functionality is important for things that may have to cook for a certain amount of time. Likewise with the tips, spreading peanut butter is not exactly a difficult task, but this shows how useful this could be for more difficult tasks and recipes.

In order to progress to the next task, the user must click the center of the screen. To go back a step, the user must click the far left of the screen. Naturally, when using AR glasses, clicking is not possible so it is likely that this functionality would be controlled through voice commands.



Once the last step has been completed, the user will see this screen in the companion app. This shows their overall experience gain from the recipe and will allow them to return to the main menu (the first screen).

Implementation Challenges

For our initial prototype we did not have the requisite capabilities to create an actual augmented reality device. Augmented reality technology is still in early stages of development with no mainstream head mounted solutions being available. Similar effects are capable with smartphone cameras or stereoscopic cameras like those found on the Nintendo 3DS, however developing for these platforms using augmented reality APIs was still outside of our scope for quickly developing an initial prototype.

In order to simulate overlaying our user interface onto the user's view, we took pictures of each step in a recipe from a similar perspective to what a user would see while executing the recipe. We then placed these images behind a mock-up of our user interface created using Java Swing and JavaFX. This allows us to evaluate the usefulness of the information we are presenting in our interface before investing in developing an AR device that may or may not be useful for users.

We also wanted to be able to present users with helpful tips. Some of these tips would be in text format, but things like techniques benefit from being presented visually. However, we did not find a reasonable way to embed streaming video within the technology stack we utilized. To simulate embedding streamed video we stored tip videos locally and then used JavaFX to play the video. This solution allows us to test the usefulness of having video based tips before investing the time required to find or develop a solution to streaming video into our interface.

Finally, we decided that the challenges, profile, and encyclopedia modalities required a lot of back-end development that we lacked the skill and time for, so we focused our attention on the main feature of the app which was the cooking experience. Also our completed augmented reality application would include a back end for the sharing of recipes. This backend is not currently implemented, as the functionality it encompasses is not necessary for testing the user interface. We simulate the existence of a back-end by simply populating one recipe within the application's code. This approach allows us to test the user interface without investing time in developing extraneous infrastructure.

Design Justification

In the last milestone, we came up with three ideas: One of them was a mobile app, the second was a virtual reality solution, and third was a voice assistant.

What we liked about the mobile application was that most people have smartphones, and therefore it would not require any additional hardware to be bought by the user. In the first milestone, we found that the budget for entertainment was not very large, and this app would definitely be the least expensive to develop and the least expensive for the customer. However, having a phone while cooking can be a problem. Users will need to wash their hands after touching the phone --which has been found to be full of germs and bacteria --and they will need to wash their hands before touching the phone because they won't want to get their phone filled with residue that may be on their hands after handling an ingredient such as chicken, meat, or even tomato juice after dicing. The fact that this design needed to be hands-on during the cooking process was its biggest downfall.

The second solution removed the need to actually be in the kitchen because instead of an assistant it was more of a learning game and tutorial resource that would help the user learn how to cook and how to perform certain techniques that would be required in the kitchen for certain recipes. This knowledge would then be applied to the kitchen. A problem with this design, however, was the cost of a VR system which would make the product quite inaccessible to a large number of people. A VR system would be a massive expense that an average user may not want to spend for one application that they are unsure that they actually want. Another problem with this idea is that techniques and tutorials in virtual reality may not translate to the real-life kitchen, and therefore the idea would be purely education on terminology and technique steps and minimal learning on how to actually perform techniques in practice.

The last solution of a voice assistant removed the need to interact with an application as the first idea required; however, it would also require new hardware to be bought, and it may not help the user learn how to cook or guide them in a way that makes cooking easier. Only interacting with the device through voice would not allow the user to know the current state that they are in in an efficient way. Is the chicken almost cooked? What step am I on? What are my next objectives so that I can prepare for them or get them started early? These types of questions and more would require constant communication with the device throughout the entire process of cooking, which could begin to get cumbersome and annoying to the user.

The decision that we came up with took parts of each of these idea to create an Augmented Reality solution for the kitchen. An AR solution would not require the hands-on interaction, which was the main downfall of the app, but it could provide a similar interface that would still allow for similar interactions just with motion of the hands and voice, eliminating the need to touch a bacteria filled device without removing much of the functionality that could be in the app. The AR solution also could provide the learning and tutorial aspects of the VR device; however, it would have eliminated the fear that skills and learned material would not translate to

the real-life kitchen because the learning is being done in a real life kitchen. Removing the simulation and instead augmenting the user's view and teaching with hands on experience would allow the user to do much of the same type of learning that would happen in a VR solution but under the real conditions of the kitchen and cooking of a recipe. With an augmented reality solution, the virtual portion of the VR solution could still be utilized if the user would prefer not to work with actual ingredients. Since it is augmenting the user's view and keeping track of the location of physical objects, the objects that are projected could be interacted with in the same way that physical objects are, and therefore a simulation of a recipe could still be done in the kitchen environment with the use of AR. Such a simulation may still have the problem of translation to real life skills, but in this case, unlike the VR solution, the user's movements are their actual, precise movements in the real world. The only simulation that is occurring is with the objects themselves; therefore, the learning goals that the VR solution failed to accomplish may be possible with this type of simulation. As previously stated, this solution will allow for voice interaction between the user and the system which will eliminate the need to touch a phone. But the device will no longer suffer the major downfall of the voice assistant. Since the view of the user is augmented, state information such as a timer, current objectives, and next objectives can be seen without the need for constant interaction with the system. Voice communication would then only be needed a few times during the cooking of a recipe where the user may want to move on to another step or go back a step, no longer needing constant interaction for trivial tasks.

We decided to create and choose this solution because it allowed us to utilize the features that we liked across the other three solutions while eliminating some of their problems. It did require a complete rethinking in order to design the solution since AR was a completely different way of interaction, but it solved the problem most effectively. Many individuals find the creative aspect of cooking to be enjoyable; however, there are some individuals that simply do not know how to cook or do not see the activity of cooking as an enjoyable activity to do because it is difficult or time-consuming or both. In fact, one of the most significant pieces of information that we discovered in our previous research was that many did not cook because they did not have time, and they thought that it was difficult. That is why there was a rise in microwavable and prepacked meals in recent years; therefore, the design needed to make the process easier and less time-consuming. Through teaching, whether in real life or a simulation, the AR solution gives the users the ability to practice which can make the process of cooking more enjoyable and easy to do for cooking novices. The users get time and difficulty estimates for each of the recipes so that they can search based on the time that they have available or the skill level that they are ready to tackle, giving novices easy recipes that they can do and learn from and people that have limited time the ability to partake in the activity of cooking. Users could also ask for recipe recommendations that could be made with a list of certain ingredients or recommendations for supplemental ingredients, which would increase creativity in the process of cooking and make it more enjoyable to make brand new recipes. In order to attempt to decrease the amount of time

cooking, the device will also suggest multitasking so that the time of a recipe can be further reduced. The main feature of the AR solution, however, is the ability to give a heads-up display to them that can give them an idea of the state that they are currently in. State information on the HUD would include current step, a preview of the next steps, tip information, and a timer of when the step needs to be completed for tasks such as pasta that needs to cook for 20 minutes. This can keep the user up-to-date on what they need to do and how they need to do it through the use of tips and videos that the user can ask to watch in the tip window on their HUD.

Overall, the AR idea solves the problem in the best way compared to the other designs because it utilizes the features that were described in the other designs that make cooking easier and more enjoyable to novices and experts, but the way that a user interacts with the device is much less intrusive to the process of cooking, allowing the user to complete a recipe and interact with the system without needing to check another device or ask constant questions to find out where they are in the recipes.