Abstract

Project Name
CookAR

Team Members:
Diya Handa, Charvi Hoysal, Rebecca Dettmar

Scope
Cooking is often perceived as a challenging task because of several reasons: lack of knowledge about recipes or techniques, difficulty in managing time and resources, or simply the overwhelming and uninspiring nature of the process. This can oftentimes discourage individuals from exploring diverse cuisines or engaging in cooking altogether. Additionally, opportunities for culture exchange and shared experiences through cooking are limited in a digital environment, especially for long-distance friends and family.

CookAR is an augmented reality (AR) wearable glasses cooking assistance designed to make cooking more engaging, accessible, interactive and fun by gamifying the experience. Our project aims to address these challenges by offering features such as:

1. Interactive, Step-by-Step Recipe WalkThroughs. We aim to provide a visual overlay in an AR setting to guide users through recipes
2. Curate multicultural recipes with historical or cultural context
3. Enable real-time collaborative cooking sessions for users in different locations
4. Incorporate gamification elements such as unlocking badges and milestones for completing recipes, trying new cuisines, or cooking within a set time. Some other elements we can add is a leaderboard element where users can compete with friends or family in a fun and friendly way and also a cooking challenges element with time-based or skill-based tasks.

Our initial implementation will focus on integrating CookAR on a 3D printed set of glasses with a companion web app. The main functionality will focus on recipe guidance, customizable cooking pace with user feedback, live connection for collaborative cooking with synchronized recipe steps and gamification elements to enhance user engagement. We are still working on how we want to create customizable user feedback, whether that be manually through the app, buttons on the glasses, or voice recognition. By combining AR technology with gamification, CookAR aims to transform cooking into an engaging, rewarding, and socially connected experience while fostering cross-cultural connections.

Requirements
As mentioned above, our audience generally includes people that struggle with cooking easily. This could be people that are first starting to cook, people that struggle with recipe ideas, or

simply people that get overwhelmed with long cooking processes. We also aim to provide opportunities for long distance friends and family to connect with these experiences, and to make cooking more fun and engaging.

Thus, our usability requirements include:

1) Intuitive controls. This could be in the form of voice command, physical buttons, or web app based controls: but it's imperative that this not be disruptive from the cooking process and flows seamlessly.
2) Clear and easy to follow AR overlay for recipe steps, including readable and accessible text.
3) Glasses must be lightweight, with a weight limit of less than 150g
4) Control box should be "pocketable", at most 0.5-1lb
5) Real time recipe guidance should be synchronized with user actions as much as possible - response to a button press should be less than 200ms

Some possible technical challenges and unknowns for the project include:

1) Feasibility of managing microdisplay on a microcontroller versus passing too much information over Bluetooth. Looking ahead, we might have trouble balancing the amount of computation happening on the microcontroller / processor versus on the cloud.
2) We are unsure of the exact power consumption for the final battery size. We expect a 5V supply, but cannot finalize actual power demand until the project is further developed.
3) We are also unsure of how fast all of our connections (wireless, cable, microcontroller, etc) are as this will determine specifics in distribution of computation load.
4) We might also need to develop some sort of effective fallback communication methods in environments with poor connectivity, as for now our requirements involve bluetooth / wifi connection.

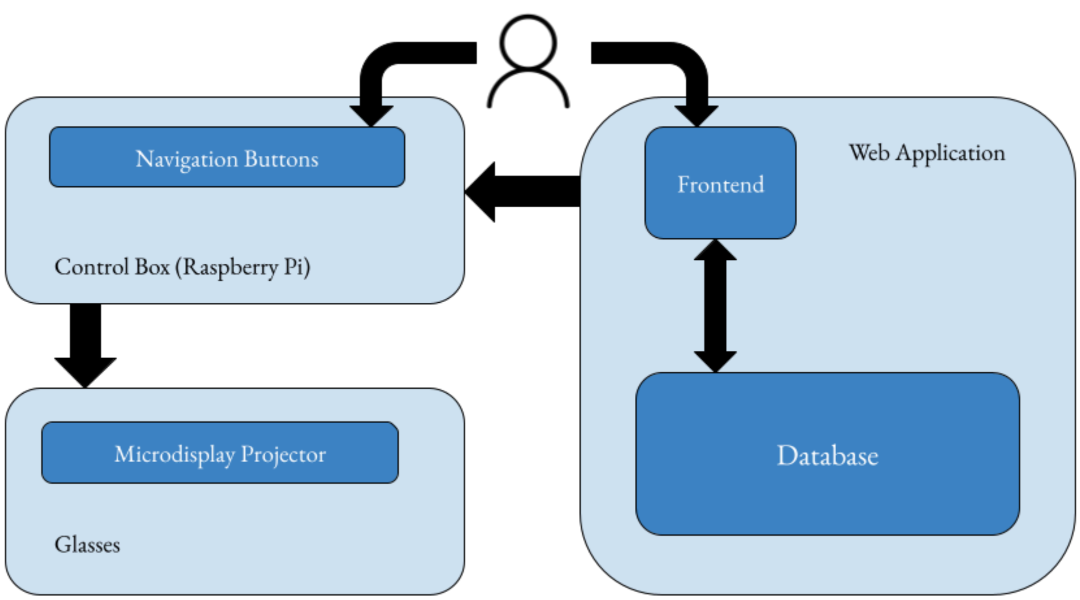
Implementation

Our hardware components will include custom AR glasses with a printed 3D frame that contain a microprocessor in the form of a Raspberry Pi. The AR glasses will have a plastic lens with a microdisplay (probably flcos), and a board.
We will also have a separate controller box to store the controls in the form of buttons to interact with the UX in the AR glasses - namely, two buttons for control over the step progression. The control box will also store the microprocessor as well as any peripheral boards needed, as these would be too heavy to store on the glasses on the face. The glasses and controller will be connected via cable.

The software components will include a webapp for most of the user input control - such as picking recipes, updating profiles, and interacting with others for the personal aspect. The Web Application will be developed using the Django framework, which has networking support. We will also deploy the Web Application on AWS EC2, as this is relatively cheap and scalable for the requirements of our project. Additionally, we will also need to store data such as user and recipe data in databases. For the scope of this project, the built in Django SQLite database will be used.

The general integration of the software and hardware components of this project will be as follows:

1) The user will use the Webapp to create a profile, add friend connections, and keep track of their recipes and achievements. They will use this app to input what recipe they want to start, and who they want to start it with.
2) We will store inputted recipe and user data in the designated DB.
3) We will send necessary user/recipe data to the microprocessor in the controller component via bluetooth.
4) The microprocessor will take the data and process it, and will then render the appropriate text and visuals.
5) The overlay will get projected via AR glasses into the space for the viewer to see.
6) When the viewer wants to control the overlay or advance the recipe, they will use the buttons on the controller.
7) When a recipe is finished or a specific achievement is reached, the microprocessor will take the appropriate control inputs and send them via bluetooth to the application on the cloud, which will process the data appropriately to then be reflected on the Webapp for the viewer to interact with.



Testing Plan

In order to verify that CookAR meets all functional and non-functional requirements, including usability, performance, reliability, and integration between hardware and software components.

Testing Area	Component	Testing Details	Metrics and Targets	Possible Challenges and Mitigations
Hardware Testing	AR glasses	Weigh the glasses to make sure that they aren't too heavy and actually wearable.	Weight <= 150g	Optimize print fill percentage to minimize the weight without compromising durability
		Testing projection clarity in various lighting conditions.	Clear projections in low, medium, and bright lighting.	Allow display brightness control as a manual user input
	Microprocessor (Raspberry Pi)	Measure power consumption to determine energy efficiency.	Battery life >= 1 hour under normal use.	Optimize hardware and firmware for energy efficiency
	Compute Box	Test the interaction between the compute box and AR glasses	<200 ms latency from user input to visual response	Ensure that there are optimized data transfer protocols.
Software Testing	Frontend	Do usability tests to make sure that users can easily navigate and there is clear instructions	Conduct a user satisfaction score. Aim for an average score of >= 7/10	Gather iterative feedback during playtests.
	Backend	Test API performance for fast and accurate data retrieval	API response time <= 1 second.	Optimize API structure and use caching wherever possible

		Perform load testing for scalability under multiple concurrent users	Handle at least 10 concurrent users without performance degradation	Use a scalable infrastructure like an AWS EC2 instance and load balancers.
	Computer Vision	Test step recognition accuracy using AR overlays	>= 90% accuracy in identifying completed steps.	Implement comprehensive training and testing of computer vision algorithms
System Testing	Integration Testing	Test the communication between glasses, compute box, and web app.	Ensure that there is seamless integration with minimal communication errors.	Perform thorough end-to-end testing and loggings
	Performance Testing	Test AR functionality for latency and responsiveness	<200ms latency for AR updates	Make sure that both hardware and software components are optimized
	AR Projection	Verify alignment and clarity of AR overlays across diverse kitchen environments	AR overlays correctly align with environment objects	Introduce some sort of calibration step during the setup.
Playtesting	Usability Testing	Conduct playtests with different users, including cooking enthusiasts and beginners.	Positive feedback on gamification and recipe guidance.	Iterate based on user feedback to refine features.
Quantitative Requirements	Latency	Measure delay in AR updates and web app responses	<= 200 ms for AR updates and web app response	Use low latency communication and protocols and also optimize

				backend services on the EC2 instance.
	Accuracy	Test the success rate of the step recognition	>= 90% accuracy in detecting completed steps and moving onto the next step.	Train computer vision algorithms on diverse datasets.

Some unknowns and future goals for CookAR include determining the exact power consumption to finalize the appropriate battery size for extended usability. We are also focused on refining the compute box design by testing the feasibility of integrating all AR computation directly into the control box. This involves researching the performance trade-offs between edge computing and relying on an external compute box to optimize efficiency and responsiveness. Additionally, we aim to investigate fallback communication methods or alternative protocols to ensure stable connectivity in environments with poor Wi-Fi or interference. Our ultimate goal is to guarantee reliable performance and seamless functionality, even in low-connectivity scenarios. By addressing these challenges, we want to enhance the robustness and user experience of CookAR in our future iterations.

MVP

Our MVP will be a functional prototype of CookAR, consisting of lightweight, 3D-printed augmented reality (AR) glasses paired with a companion web application. The glasses will feature basic AR capabilities, providing interactive, step-by-step visual overlays to guide users through cooking recipes. Users will be able to select recipes from the companion app, which will sync seamlessly with the glasses to display instructions, ingredients lists and times. The MVP will also include a customizable cooking pace, allowing users to control the flow of instructions through simple inputs, such as physical buttons on the glasses, compute box, or voice commands. Additionally, the glasses will support real-time collaborative cooking, enabling two users to connect via the app and synchronize their recipe steps for a shared experience. While the MVP will not yet incorporate advanced features like gamification or edge computing, it will create a foundation for these enhancements by first focusing on the usability, connective and clear recipe guidance in order to ensure a smooth and engaging user experience.