Hey CHEF!

<u>Alena Lu,</u> Kayla Lei, Michelle Heo Team B2 Use Case and Design Requirements

Solution Approach

System Specification

Implementation Plan

Project Management

Use Case + Requirements

Problem:

- Finding recipes and referencing while cooking is difficult \rightarrow 2 hand task. Solution:
 - Hands-free recipe guidance via smart glasses with voice assistance and web app
 - Recipe recommendations based on available ingredients

Requirements:

Smart Glasses	Voice Commands	Web App		
 Weight < 150g Battery life ~1 hour Display current recipe step as text in user's line of vision 	 Voice input process time < 4 seconds Audio output delay < 2 seconds >= 95% accuracy of audio and text instructions 	 Latency < 3 seconds 95% of recipe recommendations require only ingredients provided by user 		

Technical Design Requirements

<u>Smart Glasses:</u>

- Weight (<150g)
 - Expected glasses weight (~50g)
 - Expected weight of components (~50g)
 - PCB weight: 100mm x 25mm \rightarrow ~7.4g + 40g leeway for potential counterweight
- Battery Life (~1 hour)
 - Average Pico Current Draw: 40 mA
 - Battery: 500 mAh → 500/40 = 12.5 hours (with more components attached, this is needed)
- Ease of Display:
 - Ordered foldable, adjustable OLED that bends to fit comfortable eye view



Technical Design Requirements (cont.)

Voice Commands:

- Input Process time (< 4 seconds)
 - 0.5 to 2 seconds of latency for audio segments up to 4 seconds (Google API) + 400 ms network latency
 - Gives 2 seconds leeway for algorithmic processing
- Output Process time (< 2 seconds)
 - \circ 200–500 milliseconds for the speech to start → typical TTS latency for long instructions, 1.5 second leeway for algorithmic processing
- 95% Accuracy
 - Using Google Web Speech API or Microsoft Azure, both are 95%+ accuracy

Web App:

- Latency (< 3 seconds)
 - 100-300ms to serve content over local area network, <= 2 seconds for algorithmic processing
- 95% Accuracy
 - Up to 5% of recommendations may include omitted spices or condiments that users may have on hand

Solution Approach

Solution feature	Impact				
Hands-free guidance with voice assistance	 People who have a hard time looking at screens Reduce contamination between foods, handheld devices, and other surfaces 				
Recipe library recommendations	Allows user to explore recipes from different cultural backgrounds				
Lightweight, cost-friendly, and modular design	• Simplifies production on a large scale, making it an economically viable solution				
Recipe filters and preferences	• Cater to different cultural cuisines and dietary preferences, encouraging inclusivity				
Friendly user-interface	• Allow a variety of people of different skill levels to learn how to cook				
Mic and speaker on glasses	Easier to hear instructions and convey requests				

System Specification: Physical Device







System Specification: Web Application



Implementation Plan



Testing

Test:

- Run 3 user tests by experience level
 - Beginner cooks < 2 meals per week
 - Intermediate cooks < 6 meals per week
 - Advanced cooks >=6 meals per week
- From user tests, gather the following data:
 - Battery life
 - Voice command + web app latency
 - Voice command + LCD + web app accuracy

Physical Design Tests	Criteria	Failure Control		
Weight	Weight measured is <150g	Decrease % infill on glasses or remove counterweight		
Battery Life	Device runs for at least one hour in each test run	Increase battery (mAh) and decrease weight elsewhere		



Latency Tests	Criteria	Failure Control	Accuracy Tests	Criteria	Failure Control	
Voice Commands	No input from the glasses shall take more than 4 seconds to process	Change Python module used for voice input	Voice Commands	Of all commands inputted by the user (via voice), >= 95% of generated responses shall be relevant	Change voice command hardware or algorithm	
	No output from the glasses shall take more than 4 seconds	Optimize voice command processing algorithm	LCD	>=95% of the text shown on the LCD shall display steps and ingredients generated by the recipe algorithm	No control needed - simply fix LCD implementation	
Web App	No input/output from the webapp shall take more than 3 seconds	Upgrade hardware to decrease latency between physical device and web app	Web App	Of all recipes requested by the user, >=95% of them will include all relevant ingredients and preferences	Change recipe generation algorithm	

Project Management + Schedule

							-		
1 1	Research + Design	- -	-	Ŧ	Ŧ		FFFF leam		Lask
1.1	Voice Commands	Alena	9/9/24	9/11/24	2	92%	i Curri		TUSK
1.1.1	Find libraries	Alena	9/9/24	9/11/24		100%			
1.1.2	Test libraries	Alena	9/9/24	9/11/24	2	75%	Momh	or	
1.1.3	Microphone research	Alena	9/9/24	9/11/24	2	100%			
1.2	Web App	Kayla	9/9/24	9/11/24	2	88%			
1.2.1	Web app-RPi communication	Kayla	9/9/24	9/11/24	2				
1.2.2	Web scrapers	Kayla	9/9/24	9/11/24	2				
1.2.0	Recommendation also	Kayla	9/9/24	9/11/24	2	505			
13	Physical Design	Michelle	9/9/24	9/11/24	2	88%	Micho		Develoal Design
1.3.1	Initial sketch finalization	Michelle	9/9/24	9/11/24	2	100%		ווכ	FIIVSICAL DESIGN
1.3.2	Speaker, LCD, battery	Michelle	9/9/24	9/11/24	2	50%			, 6
1.3.3	Initial CAD model	Michelle	9/9/24	9/12/24	3	100%			
1.4	Microcontroller research	Michelle + Alena	9/9/24	9/11/24	2	100%			Dovico Intogration
2	Prototyping								
2.1	Parte	Michelle + Alena	9/15/24	9/25/24	10	55%			0
2.1.1	Finalize parts list	Michelle + Alena	9/15/24	9/16/24	1	100%			
2.1.2	Create BOM	Michelle + Alena	9/15/24	9/16/24	1	100%			Sub-Device Communication
2.1.3	Order parts	Michelle + Alena	9/15/24	9/18/24	3	75%			
2.1.4	Schematic for parts	Michelle + Alena	10/1/24	10/4/24	3	0%			
2.1.5	Layout	Michelle + Alena	10/5/24	10/9/24	4	0%			
2.1.0	Order POB	Michelle + Alena	0/22/24	0/28/24	2	0%	and the second		
221	CAD Model for housing	Michelle	9/22/24	9/25/24	3	105			
222	3D print prototype housing	Michelle	9/25/24	9/26/24	1	0%			Voice Commande
2.3	Design web app UI	Kayla	9/15/24	9/18/24	3	0%	I Alena		Voice Commands
2.4	Initial voice commands prototype	Alena	9/15/24	9/20/24	5	0%			
2.4.2	SpeechRecognition library> STT	Alena	9/15/24	9/18/24	3	100%			
2.4.2	Generate responses with Pyttsx3	Alena	9/22/24	9/25/24	3	0%			Dovico Intogration
3	Implementation								
31	Web app frontend	Kavla	9/25/24	9/30/24	5	05			
3.1.1	Set up hosting infrastructure	Kayla	9/25/24	9/26/24	1	0%			
3.1.2	Replicate UI mockups	Kayla	9/25/24	9/30/24	5	0%			Sub-Device (ommunication
3.2	Web app backend	Kayla + Alena	9/30/24	10/11/24	11	0%			
3.2.1	Database queries	Kayla	10/7/24	10/9/24	2	0%			
3.2.2	Recipe recommendation algorithm	Kayla	10/4/24	10/8/24	4	0%			
324	Web acrapera	Kayla + Alena Kayla	9/30/24	10/3/24	3	0%			
3.2.5	Authentication	Kayla + Alena	9/30/24	10/1/24	1	0%			
3.3	Recipe database	Kayla	10/4/24	10/7/24	3	0%			Meh Ann
3.3.1	Set up database	Kayla	10/4/24	10/5/24	1	0%	Ιταγια		MACD WAD
3.3.2	Data structures	Kayla	10/5/24	10/7/24	2	0%			
3.4	Microcontroller Code	Michelle & Alege	9/20/24	0/20/24	10	0%			
2.4.2	Pring up microphone	Michelle + Alega	9/20/24	10/2/24	2	05			Recipe Algorithm
343	Bring up speaker	Michelle + Alena	10/2/24	10/4/24	2	0%			
3.4.4	Bring up LCD	Michelle	10/4/24	10/6/24	2	0%			
3.5	Voice commands	Alena	9/22/24	10/2/24	10	0%			
3.5.1	Finalize library and code for voice	Alena	9/22/24	9/28/24	6	0%			VVeb Scraping
3.5.2	Sentence splicing for keywords	Alena	9/22/24	9/28/24	6	0%			the octuping
353	Generate matching responses with	Alana	9/27/24	10/2/24	5	035			
	Pyttax3 -> TTS		Jacob						 Database
4	Hardware Testing	Michelle	10/7/2/	10/11/24		02			
3.1.1	Verify device connection	Michelle	10/7/24	10/7/24	0	0%			
3.1.2	Verify microphone functions	Michelle + Alena	10/7/24	10/8/24	1	0%			
3.1.3	Verify speaker functions	Michelle + Alena	10/9/24	10/10/24	1	0%			
3.1.4	Verify LCD functions	Michelle	10/11/24	10/12/24	1	0%			
3.2	Software Testing	Kayla	10/14/24	10/17/24	3	0%			Classes I web app integration
5	Fall Break						All		Glasses + web abb integration