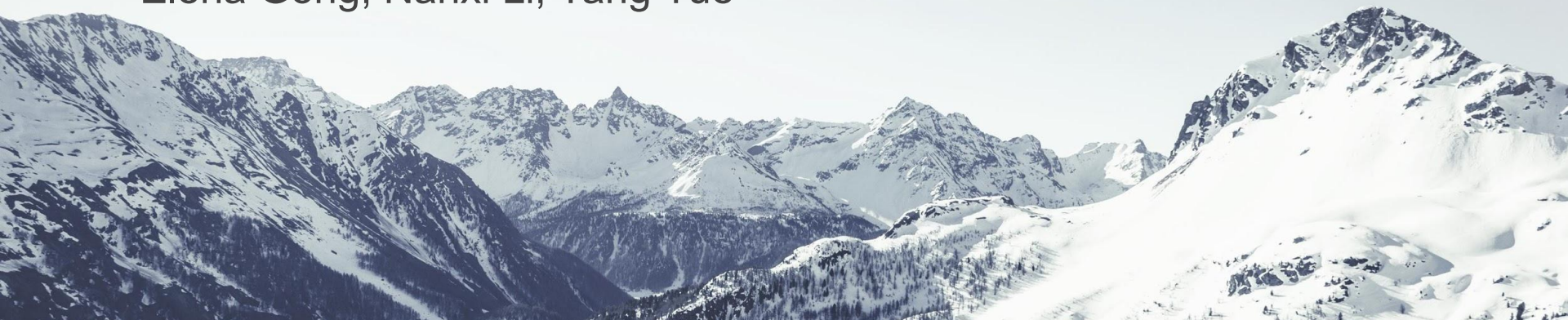
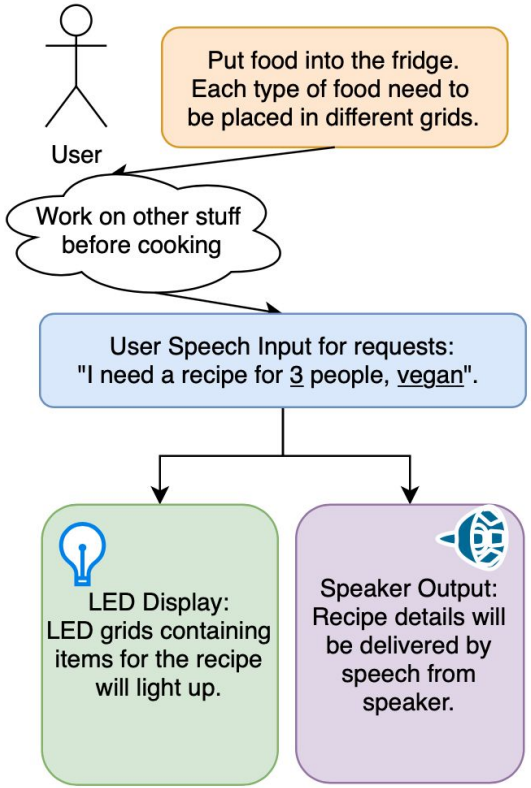
SmolKat:
A Smart Kitchen Assistant

Elena Gong, Nanxi Li, Yang Yue



Use Case

- **Goal: Reduce food waste by improving refrigerator throughput**
- Track ingredients in refrigerator
  - Use OpenCV + desktop camera
- Recommend Recipes
  - Use a personalized database to recommend recipes via text-to-speech
- Highlight required ingredients
  - Use LEDs to guide users to the locations of key ingredients
- **ECE Areas: Software, Hardware**



Requirements: Latency of System, Responsiveness

1. End-to-end System Latency: 100ms
    a. Selected as the boundary for humans to perceive as instantaneous
    b. Total allotted time between the customer request, to speech processing, recipe selection, text-to-speech generation, and ingredient highlighting
2. Ingredient Recognition Latency: 2 seconds
    a. Image processing and ingredient recognition latency do not directly impact customer experience.

Requirements: Accuracy of Recognition Systems

- Speech Processing Accuracy: 90%
  - Selected set of 5 keywords for proof of concept
  - Accuracy bound before user experience degrades
- Food Recognition Accuracy: 95%
  - Selected set of 10 common food items for proof of concept
  - High accuracy bound, rest of system depends on accuracy of food recognition.

Requirements: Quality of Recipe Recommendations

- Assumed that common pantry items (salt, pepper, flour) are available
- Required dataset of diverse recipes such that the user will not be tired of repetitive meals
    - Lower bound target of 30 recipes
- Able to meet special requirements for the end user (vegetarian, lunch/dinner, high in protein)
- Measured success by providing recipes to sample population, recording satisfaction
    - Targeted: 90% satisfaction

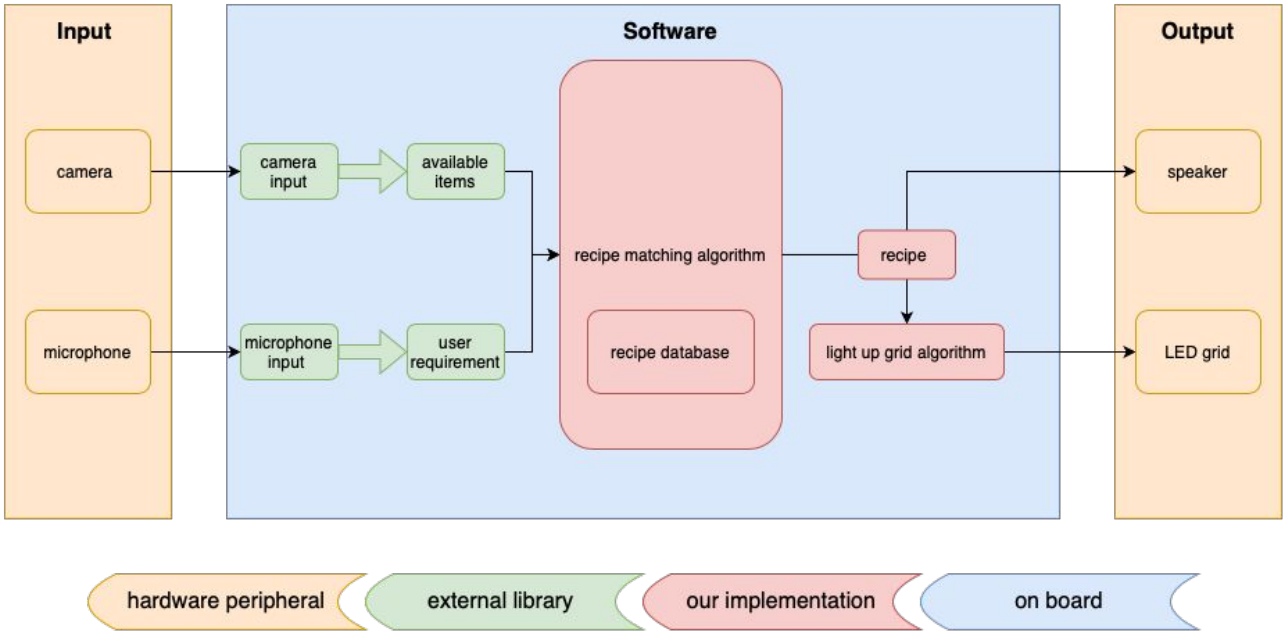
Technical Challenges: Meeting Latency

- End-to-end latency requirement: 100ms
  a. Speech Recognition to parse user requirements
  b. Recipe Selection from internal database
  c. Text-To-Speech processing of audio output
  d. LED highlighting for physical ingredient guidance
- Mitigation Plan: Use more compute heavy / AI focused processing board
  a. Swap from using Raspberry Pi 4 to AI-ML workload focused boards such as Jetson Nano
  b. Speed up the processing for the speech recognition, text to speech processing
  c. Speed up the processing for ingredient recognition
  d. Costs: Power Consumption, Unit Cost

Technical Challenges: Meeting Recognition Accuracy

- Accuracy bounds for Speech Processing, and Ingredient Recognition
- Use various ML/AI frameworks to accomplish our Speech Processing and Ingredient Recognition
  - System will run directly on hardware platform, will not use cloud processing
- Risk Mitigation:
  - Do more processing with denser ML models -- tradeoff latency for accuracy
  - Use cloud ML services such as Amazon Comprehend NLP, Google Vision AI -- tradeoff latency for accuracy

Solution Design: Overview



Solution Design: Components

- **Software:**
  - **Speech Processing**: Pretrained Libraries -- Jetson.Inference, SpeechRecognition python library
  - **Recipe Recommendation System**: Custom software database
  - **Image Recognition:** Jetson.Inference, OpenCV, Google Vision AI
- **Hardware Peripherals:**
  - **LED grids:** GPIO pins + Jetson.GPIO python library to control chained LEDs
  - **Audio Input/Output:** Adafruit Mini USB Microphone/Speaker
  - **Image Input:** Logitech Desktop Camera with USB communication
- **Hardware Platform:** NVIDIA Jetson Nano

Testing, Verification, and Metrics

Description	Goal	Verification Method
System Latency	100ms	Record time from user input to system output. Full integration test.
Image Processing Latency	2 seconds	Record time from image input to result output of ML model
Speech Recognition	90% accuracy	Accuracy on test dataset of audio recorded with selected microphone
Food Recognition	95% accuracy	Accuracy on test dataset of images taken with video hardware

Tasks and Division of Labor

Hardware System: Nanxi

- Connect LED grids with RPi
- Microphone, Camera, Speaker with RPi's I/O
- Choose packages

Recommendation System: Elena

- Find appropriate datasets
- Feature extraction
- Build baseline models
- Build models for recommendation
- Test & Improve performance

Image & Speech Processing: Yang, Nanxi

- User speech request to text
- Speech from recipes text
- Process photos to inputs
- Run CNN on PC for food recognition
- Migrate model to Hardware
- Test on image/speech recognition accuracy

Proposed Schedule