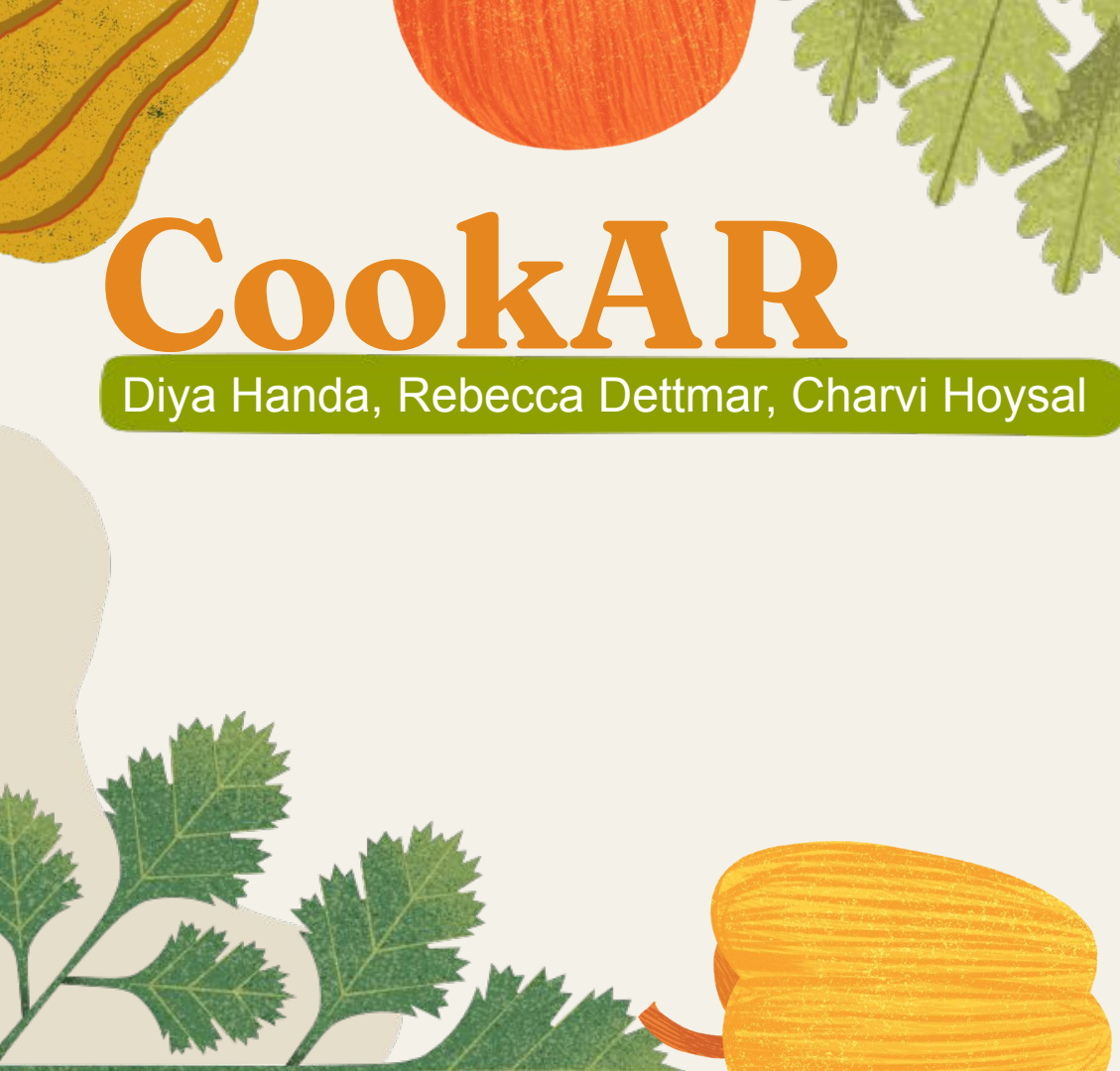
CookAR

Diya Handa, Rebecca Dettmar, Charvi Hoysal



A Use Case

B Use-Case Requirements

C Technical Challenges

D Solution Approach

E Testing, Verification, Metrics

F Tasks & Labour Division

G Schedule

USE CASE

Cooking can be overwhelming, requiring constant checking of screens or recipe books. Current solutions lack hands free guidance and accessibility.

Target Audience

- **A** Beginner cooks
- **B** Busy individuals
- **C** Individuals with accessibility needs

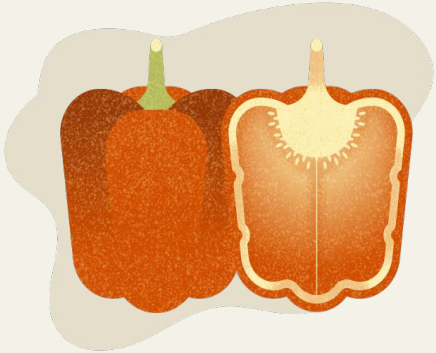
CookAR's Solution

AR based hands-free cooking assistant that displays step by step instructions. The system tracks progress through gesture recognition.

USE CASE REQUIREMENTS

Requirement	Target Value	Justification & Rationale
Gesture Recognition Accuracy	≥ 90% gesture recognition accuracy	Ensures hands-free navigation without errors. Based on Google Mediapipe, which achieves >90% real-time accuracy
AR Overlay Clarity	Readable at 30–50 cm (distance from glasses).	Matches focus range of AR devices (e.g., Microsoft HoloLens) for clarity and reduced eye strain.
Glasses Weight	≤ 150g	Matches current lightweight AR headsets on the market. Nreal Air(79g), Google Glass (50g). Comfortable for duration of cooking
Latency	< 200ms latency between steps	Maintains real time interaction. Industry standard for AR systems
Battery Life	≥ 2 hours or a continuous power option	Supports typical 1-2 hour cooking sessions without interruptions
Wireless Stability	10-meter Bluetooth range	BLE ensures low latency, reliable communication for typical kitchen distances
Step-by-Step	Recipes broken into ≤ 10 steps	Reduce cognitive load and simplify the cooking process. Supported by usability studies for task based guidance
Recipe Database	100 sanitized beginner recipes	Provides variety and is manageable for pre-processing and testing.

USER SAFETY REQUIREMENTS



A Physical Safety

- Glasses weight limited to ≤150g
- Materials used for 3D printing will be lightweight, skin safe

B Thermal Safety

- Active monitoring of temperature sensors in microcontroller
- Microcontroller can operate up to 85°C. Will have ventilation and low power components

C Electrical Safety

- Low voltage batteries
- Insulated devices to prevent short circuits or surges

TECHNICAL CHALLENGES

Gesture Recognition

Why

Reliable detection in cluttered or dim environments

Risk Mitigation Plan:

Use pre-trained models and diverse environment testing

AR Overlay Visibility

Why

Lighting and readability issues

Risk Mitigation Plan:

Adjustable brightness and high-contrast text.

Step Latency

Why

Disrupts cooking flow

Risk Mitigation Plan:

Optimize TensorFlow Lite and reduce processing load

Wireless Communication

Why

BLE interference and disconnections

Risk Mitigation Plan:

Automatic reconnection and local data storage

Weight of Glasses

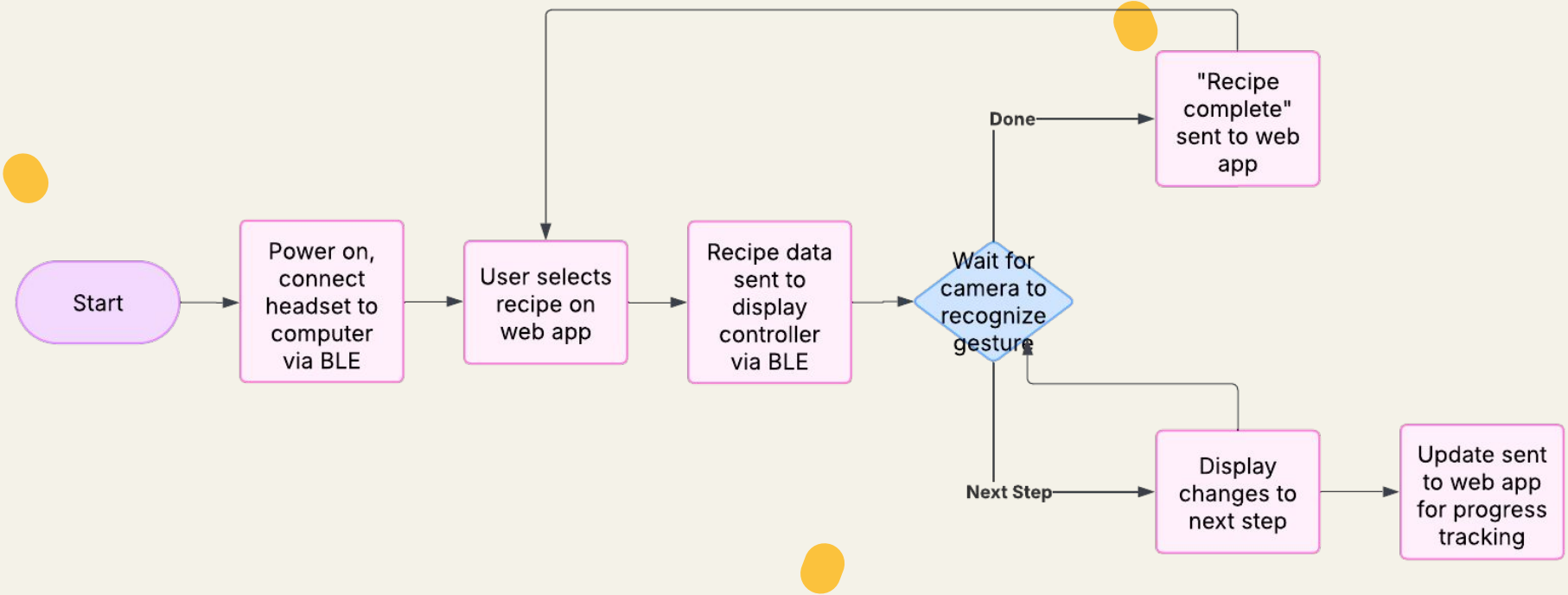
Why

Additional hardware risks making glasses heavy

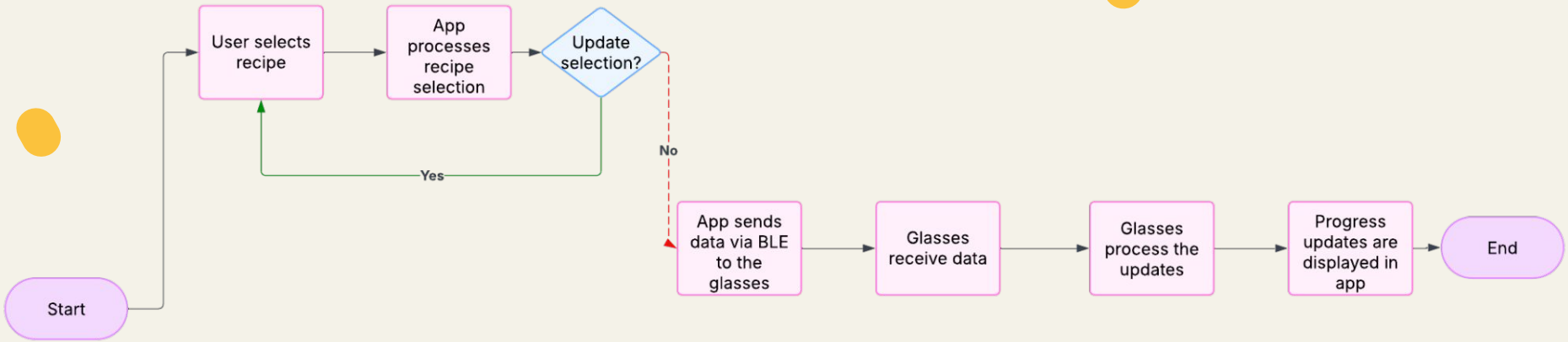
Risk Mitigation Plan:

Offload processing to the microcontroller and use lightweight materials

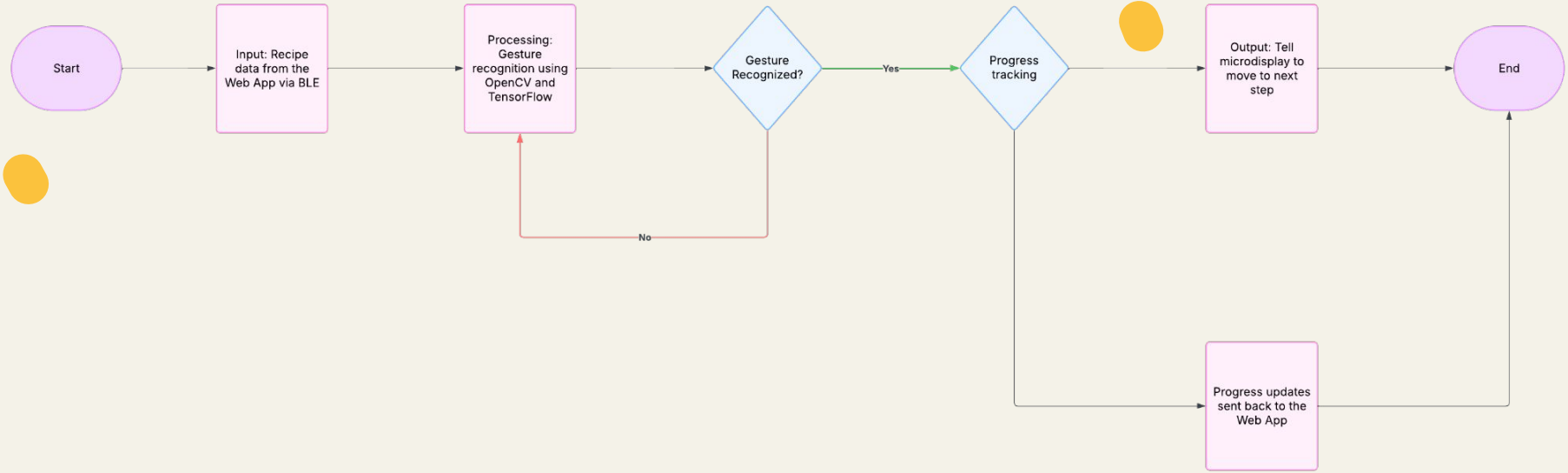
SOLUTION APPROACH



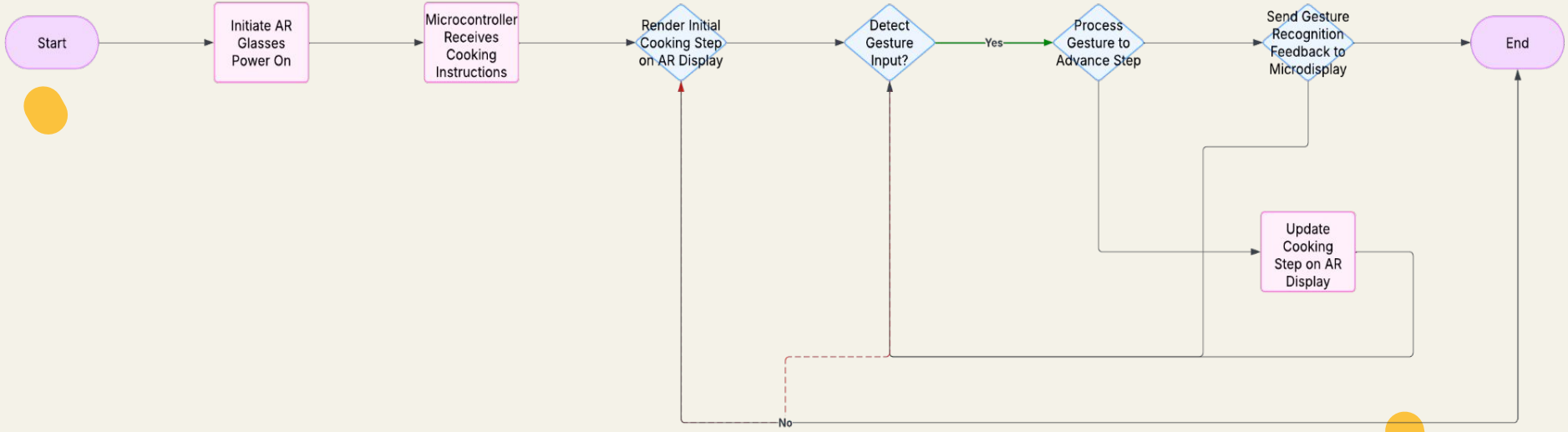
Web App Workflow



Camera Workflow



Microdisplay workflow



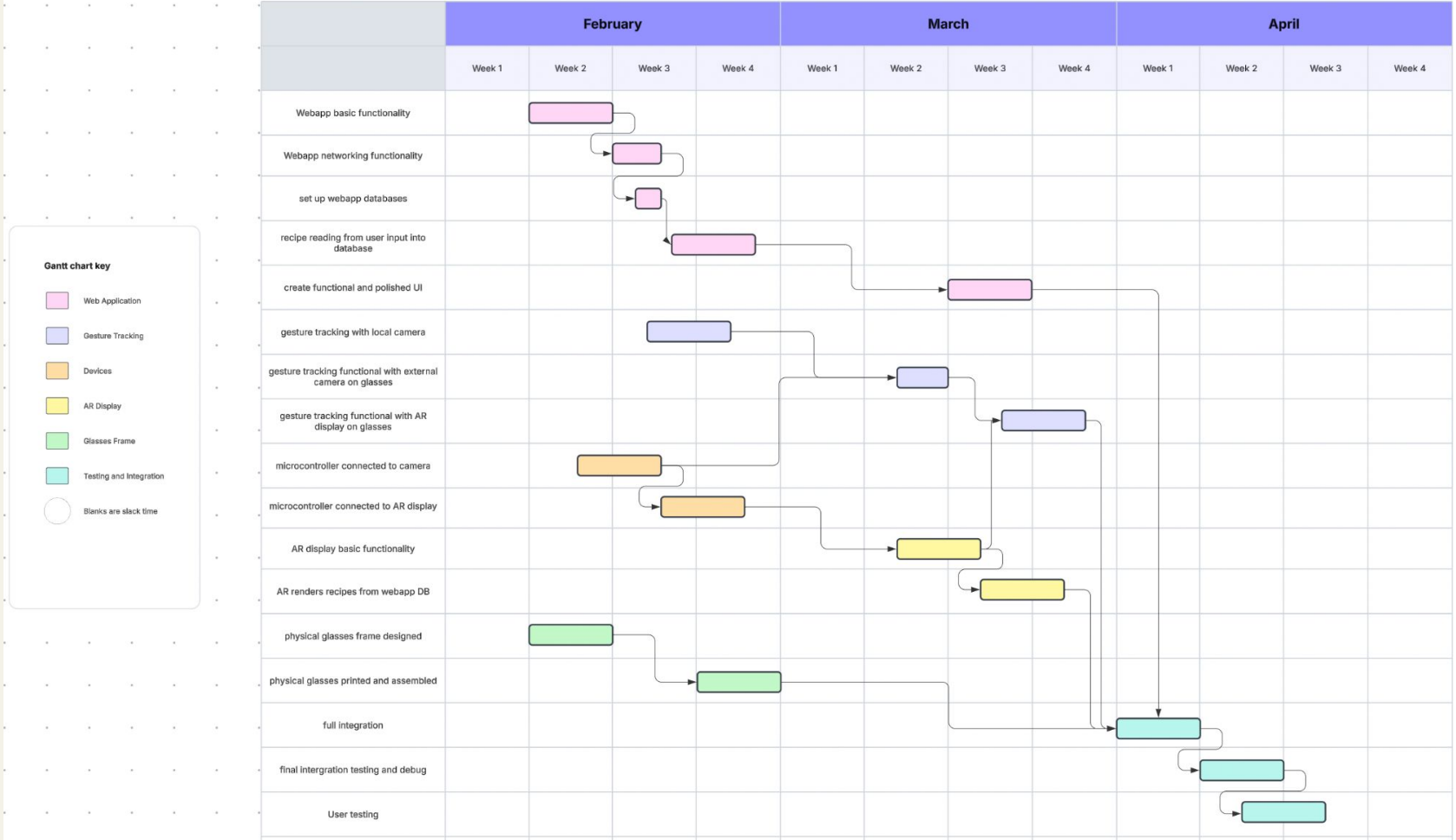
TESTING, VERIFICATIONS, METRICS

- **Gesture Tracking**
  - Functional with local camera
  - Functional with camera on glasses
  - Functional interaction with AR display
  - >= 90% accuracy
- **Web Application**
  - Functional basic application
  - Functional networking
  - Functional recipe input to database
  - User testing
- **AR Display**
  - Graphics properly rendering to AR Display
- **Physical Glasses**
  - User testing for wearability
  - < 150g weight
  - > 2hr battery life
- **Connectivity**
  - Functional bluetooth connection between glasses and webapp
  - Able to send correctly formatted data to webapp
  - Connection working in 10 meter range
  - Functional communication between camera and microcontroller
  - Functional communication between display and microcontroller
  - < 200ms latency for full interaction cycle between user gesture and AR display update
  - API response time <1 second on webapp
  - Successful recipe render to AR display

Tasks and Division of Labour

Task	Team Member
Construction of glasses - 3D printing frame - Physical part integration - Device microcontrollers	Rebecca
Web App - Backend framework - Frontend design - Integration with CV processing	Charvi and Diya
Hand Gesture Recognition - Integration of datasets/libraries - Integration with camera data	Diya
AR Overlay in Unity - Design overlay - Integration with web app - Mapping overlays to physical lens	Charvi

GANTT CHART


February
March
April
Week 1
Week 2
Week 3
Week 4
Webapp basic functionality
Webapp networking functionality
set up webapp databases
recipe reading from user input into database
create functional and polished UI
gesture tracking with local camera
gesture tracking functional with external camera on glasses
gesture tracking functional with AR display on glasses
microcontroller connected to camera
microcontroller connected to AR display
AR display basic functionality
AR renders recipes from webapp DB
physical glasses frame designed
physical glasses printed and assembled
full integration
final intergration testing and debug
User testing
Gantt chart key
Web Application
Gesture Tracking
Devices
AR Display
Glasses Frame
Testing and Integration
Blanks are slack time