

MICHIGAN STATE

UNIVERSITY

Project Plan Presentation

Voice Transcription API

The Capstone Experience

Team RPM

Blake Garvin

Baran Manti

Karthik Kandikonda

Joel Nataren Moran

Al Al-Hassan

Department of Computer Science and Engineering
Michigan State University

Spring 2024



*From Students...
...to Professionals*

Project Sponsor Overview

- International logistics and supply chain solutions
- Focused on freight transportation logistics
- Connects clients and carriers
- Non-asset based



Project Functional Specifications

- Call centers are experiencing high volume of logistics questions
- Reduce workload and increase operational efficiency
- Interactive voice assistant with multilingual support
- View and modify personal and logistics information
- Admin interface



Project Design Specifications

- Display recent call logs
- Manage outgoing calls
- View call transcripts
- View call summary and metrics



Screen Mockup: Call logs

RPM

Call Logs

Outgoing Calls

Call Logs

Call No	Date	Contact	
123123	01/22/2024	John Doe	View Log
123123	01/22/2024	John Doe	View Log
123123	01/22/2024	John Doe	View Log
123123	01/22/2024	John Doe	View Log
123123	01/22/2024	John Doe	View Log
123123	01/22/2024	John Doe	View Log
123123	01/22/2024	John Doe	View Log



Screen Mockup: Outgoing calls

RPM

Call Logs

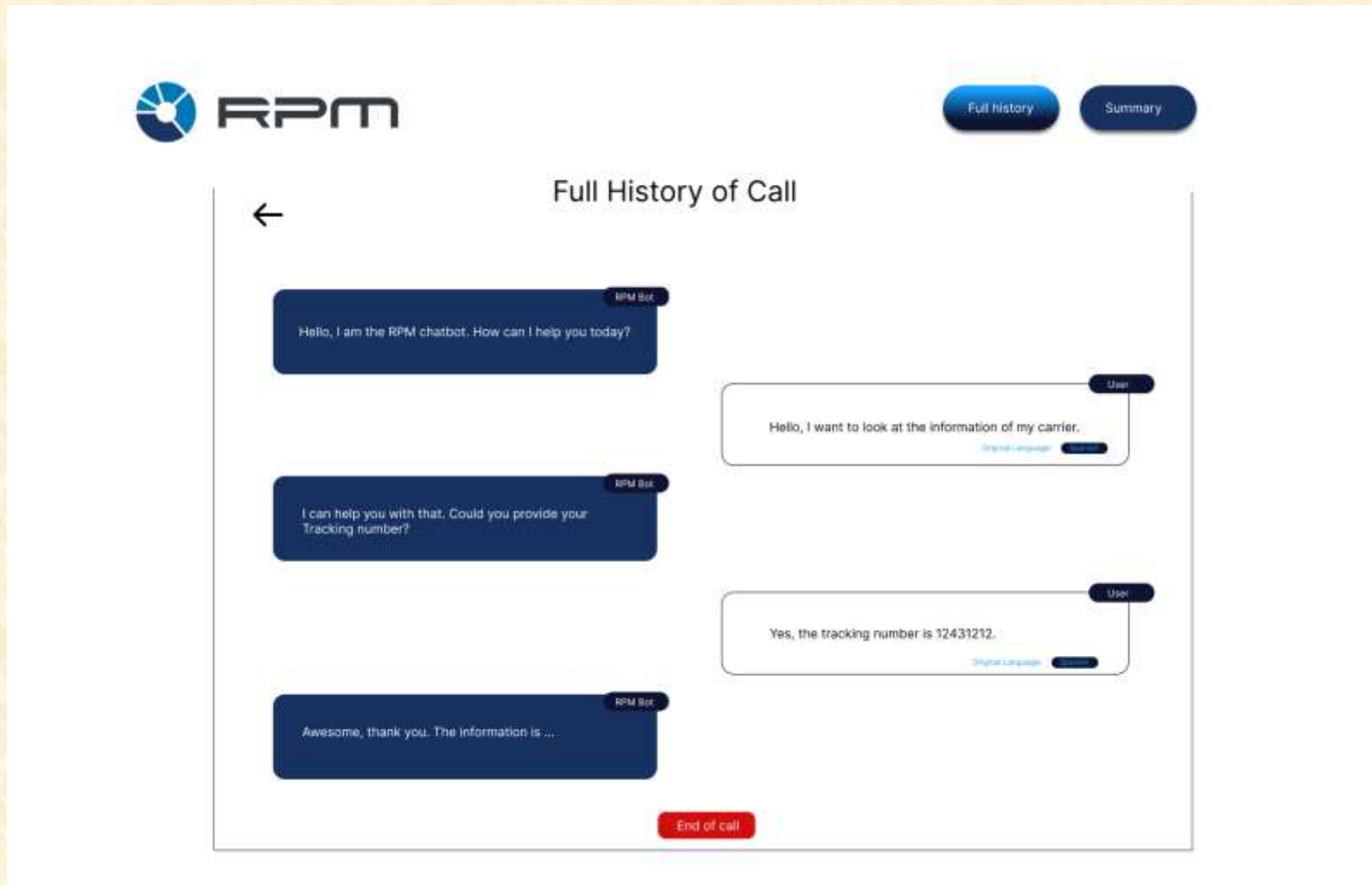
Outgoing Calls

Outgoing Calls

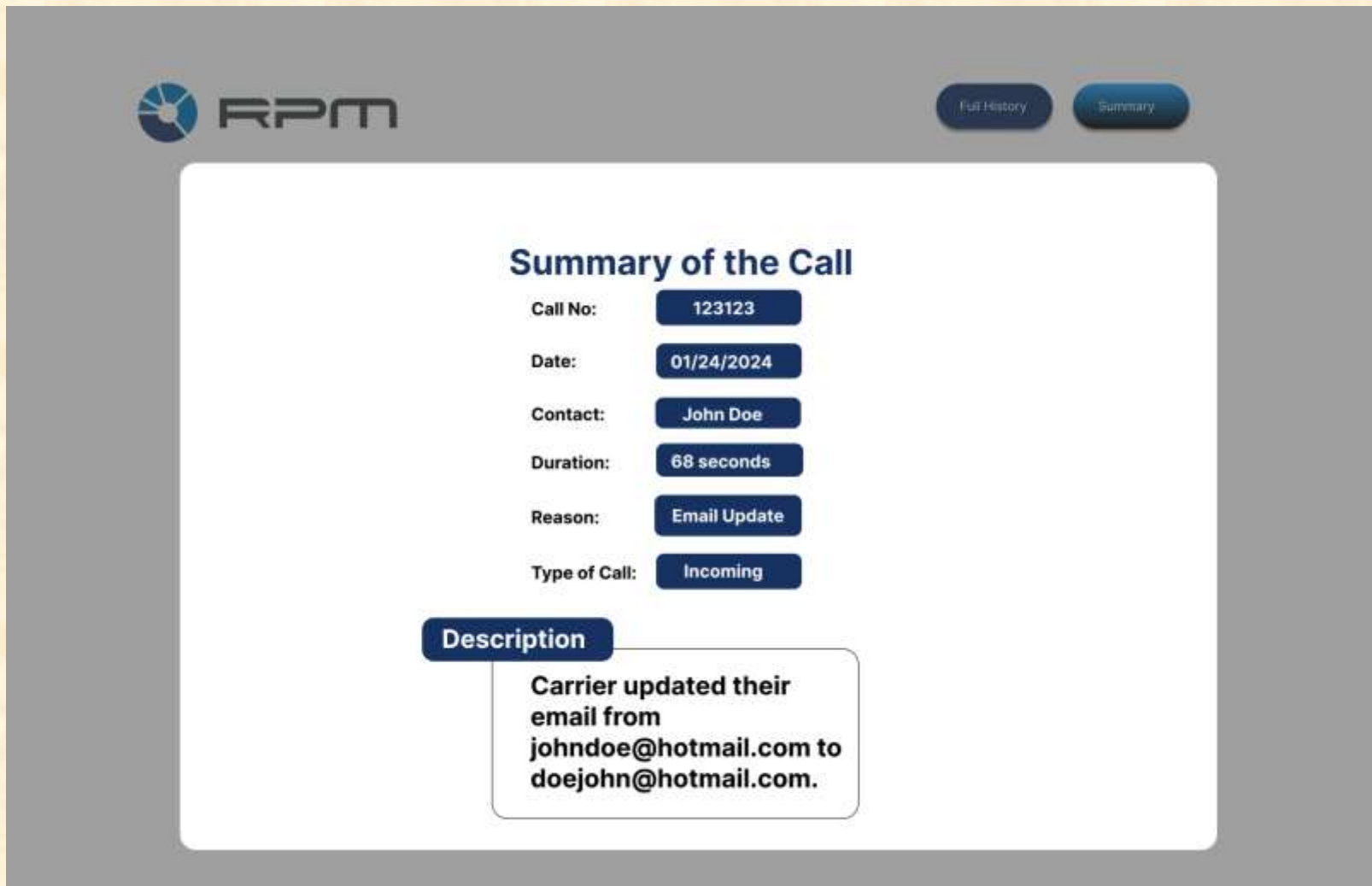
Phone Number	Contact	
+1 517 123 1212	John Doe	Call
+1 517 123 1212	John Doe	Call
+1 517 123 1212	John Doe	Call
+1 517 123 1212	John Doe	Call
+1 517 123 1212	John Doe	Call
+1 517 123 1212	John Doe	Call
+1 517 123 1212	John Doe	Call



Screen Mockup: Call history



Screen Mockup: Call summary

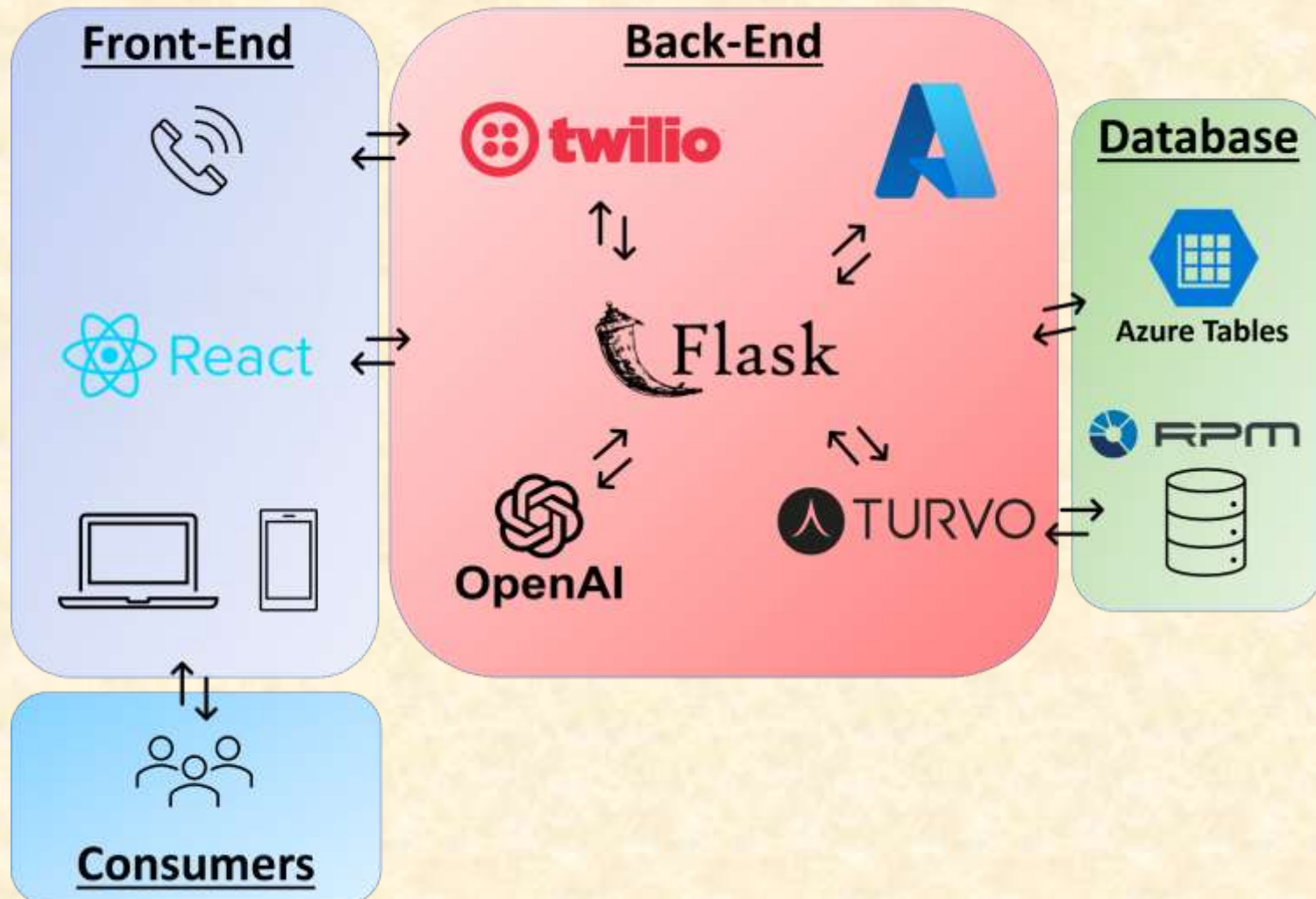


Project Technical Specifications

- Flask
- Twilio
- OpenAI
- Azure AI
- Azure Table Storage
- Turvo
- React



Project System Architecture



Project System Components

- Hardware Platforms
 - Phone
- Software Platforms / Technologies
 - Flask
 - Turvo
 - OpenAI
 - Azure resources
 - React



Project Risks

- Scalability and cost management
 - Running multiple calls at once while keeping costs down
 - Design backend to maximize token efficiency
- Voice command interpretation and execution
 - Understanding the needs of the caller and executing commands accordingly
 - Leverage OpenAI functions to take action accordingly
- Voice recognition in noisy environments
 - Identifying the caller even in noisy environments
 - Utilize Azure AI to recognize caller's voice effectively
- Data security and authorization
 - Avoiding leaking confidential information to unauthorized callers
 - Incorporate an efficient structure of OpenAI functions



Questions?

