BLIND ENHANCED AND ASSISTIVE CAP WITH OPTICAL NAVIGATION

A Project Report

Submitted to the APJ Abdul Kalam Technological University in partial fulfillment of requirements for the award of degree

Bachelor of Technology

in

Computer Science and Engineering

by

BASHAAR ABDULLAH (AWH22CS016)

MADHAV M P (AWH22CS029)

NIVED T (AWH22CS044)

SABARISH A V (AWH22CS048)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

AWH ENGINEERING COLLEGE KUTTIKKATTOOR

KOZHIKODE KERALA

MARCH 2026

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
AWH ENGINEERING COLLEGE, KOZHIKODE

CERTIFICATE

This is to certify that the report entitled **BEACON BLIND ENHANCED AND ASSISTIVE CAP WITH OPTICAL NAVIGATION** submitted by **BASHAAR ABDULLAH, MADHAV M P, NIVED T, SABARISH A V** to the APJ Abdul Kalam Technological University in partial fulfillment of the B.Tech. degree in Computer Science and Engineering is a bonafide record of the seminar work carried out by him under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

INTERNAL SUPERVISOR	PROJECT COORDINATOR	HEAD OF THE DEPARTMENT
Ms. Rahana Sulthana	**Ms. Anjana P**	**Dr. Gireesh TK**
Assistant Professor	Assistant Professor	Associate Professor
Dept.of CSE	Dept.of CSE	Dept. Of CSE
AWH Engineering College	AWH Engineering College	AWH Engineering College
Kozhikode	Kozhikode	Kozhikode

DECLARATION

We hereby declare that the seminar report **BEACON BLIND ENHANCED AND ASSISTIVE CAP WITH OPTICAL NAVIGATION**, submitted for partial fulfillment of the requirements for the award of degree of Bachelor of Technology of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under supervision of Ms. Rahana Sulthana. This submission represents my ideas in my own words and where ideas or words of others have been included, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

Kozhikode
23-10-2025

BASHAAR ABDULLAH (AWH22CS016)
MADHAV M P (AWH22CS029)
NIVED T (AWH22CS044)
SABARISH A V (AWH22CS048)

ABSTRACT

The proposed B.E.A.C.O.N (Blind Enhanced and Assistive Cap with Optical Navigation) system is driven by the critical need to enhance mobility, safety, and independence for visually impaired individuals, who continue to face significant challenges in perceiving and navigating dynamic environments. Traditional assistive tools such as white canes and guide dogs provide limited environmental awareness and lack the ability to interpret complex surroundings in real time. These limitations often result in reduced confidence, increased dependency, and a higher risk of accidents. Therefore, there is a pressing need for an intelligent, wearable, and integrated assistive solution that can provide real-time situational awareness and adaptive guidance.

The system integrates a camera-mounted smart cap with smartphone-based processing to enable continuous monitoring of the user's surroundings. The visual data captured by the camera is transmitted to the smartphone, where multiple artificial intelligence models operate in parallel to perform obstacle detection, crosswalk recognition, and currency identification. This multi-model approach allows the system to provide comprehensive environmental understanding, enabling users to avoid obstacles, identify safe crossing zones, and recognize currency independently. The use of vision-based intelligence significantly enhances the system's ability to interpret real-world scenarios compared to traditional sensor-based methods.

Furthermore, the system employs edge-based processing through the smartphone, ensuring low latency, real-time responsiveness, and reduced dependency on internet connectivity. This approach enhances reliability in outdoor environments while preserving user privacy by avoiding cloud-based data transmission. The detected information is converted into audio feedback, providing intuitive and immediate guidance to the user.

In addition to the AI-driven perception module, the system incorporates a feature-rich mobile application designed specifically for accessibility. The application includes functionalities such as fall detection with automatic alert generation, manual emergency alert triggering, customizable emergency contact management, weather condition analysis to advise outdoor movement, voice-based time and location announcements, and a reminder-based alarm system for daily activities. A voice-assisted interface with a confirmation-based interaction mechanism ensures ease of use and prevents accidental operations.

The integration of AI-based perception with a responsive mobile application backend creates a unified assistive ecosystem capable of both environmental awareness and personal safety management. By combining real-time object detection, navigation assistance, and intelligent user interaction within a lightweight wearable design, the proposed system offers a practical and scalable solution. The expected outcome is a significant improvement in user autonomy, safety, and confidence, demonstrating the effectiveness of the B.E.A.C.O.N system as an advanced assistive technology for visually impaired individuals.

ACKNOWLEDGEMENT

We would like to take this opportunity to express our sincere gratitude to all those who supported and guided us in the successful completion of this project.

We express our heartfelt thanks to **Dr. Gireesh T K**, Head of the Department of Computer Science and Engineering, AWH Engineering College, Kozhikode, for providing us with the necessary facilities, encouragement, and support throughout the course of this project.

We extend our sincere thanks to our project coordinator, **Ms. Anjana P**, Department of Computer Science and Engineering, AWH Engineering College, Kozhikode, for her valuable guidance and cooperation during the development of this project.

We would like to place on record our deep sense of gratitude to our project guide, **Ms. Rahana Sulthana**, Assistant Professor, Department of Computer Science and Engineering, AWH Engineering College, Kozhikode, for her constant support, insightful suggestions, and encouragement, which greatly contributed to the successful completion of this work.

We also express our sincere thanks to all faculty members of the Department of Computer Science and Engineering for their support and assistance.

Finally, we extend our heartfelt gratitude to our family and friends for their continuous encouragement, motivation, and support throughout the completion of this project.

BASHAAR ABDULLAH
MADHAV M P
NIVED T
SABARISH A V

CONTENTS

Content	Page No.
ABSTRACT	i
ACKNOWLEDGEMENT	ii
LIST OF FIGURES	v
LIST OF TABLES	vi
1. INTRODUCTION	1
2. LITERATURE SURVEY	3
2.1. Paper 1: Safety Helmet Detection Based on Improved YOLOv8	3
2.2. Paper 2: Smart Helmet 5.0 for Industrial Internet of Things Using Artificial Intelligence	5
2.3. Paper 3: Computer Vision Based Detection and Classification of Road Obstacles	7
2.4. Paper 4: Obstacle Detection and Warning System for Visually Impaired Using IoT Sensors	9
2.5. Paper 5: Personal Voice Assistant Security and Privacy	11
2.6. Paper 6: Smart Helmet Compliance Detection Using Enhanced YOLO-M	13
2.7. Paper 7: Visual Fall Detection from Activities of Daily Living for Assistive Living	15
2.8. Paper 8: A Deep Learning Framework for the Classification of Brazilian Coins	17
2.9. Paper 9: Human-Centered AI in Smart Farming: Toward Agriculture 5.0	19
2.10. Paper 10: Omni-Directional Obstacle Detection for Vehicles Based on Depth Camera	21
2.11. Comparison of Related Works	23
3. SYSTEM ANALYSIS	24
3.1. Existing System	24
3.2. Problem Statement	24
3.3. Proposed System	25
4. FEASIBILITY STUDY	26
4.1. Economic Feasibility	26
4.2. Technical Feasibility	26
4.3. Operational Feasibility	27
4.4. Implementation Feasibility	27
4.5. Time Complexity and Performance Feasibility	28
5. REQUIREMENT SPECIFICATIONS	29
5.1. Functional Requirements	29
5.2. Non-Functional Requirements	30
5.3. Usecase Diagram	31
5.4. Dataflow Diagram	31
5.5. Entity Relationship Diagram	34

6. SYSTEM DESIGN 35
6.1. Software Architecture 35
6.2. Working Principle 36
6.3. Database Design 37

7. IMPLEMENTATION AND TESTING 41
7.1. Minimum Hardware Requirements 41
7.2. Minimum Software Requirements 41
7.3. Types of Testing Conducted 41
7.4. Test Cases 43

8. RESULTS AND DISCUSSION 45
8.1. Screenshots 46

9. CONCLUSION 59

REFERENCES 60

LIST OF FIGURES

No.	Title	Page No.
2.1	Safety Helmet Detection Based on Improved YOLOv8	3
2.2	Smart Helmet 5.0 for Industrial Internet of Things Using Artificial Intelligence	5
2.3	Computer Vision Based Detection and Classification of Road Obstacles	7
2.4	Obstacle Detection and Warning System for Visually Impaired Using IoT Sensors	9
2.5	Personal Voice Assistant Security and Privacy	11
2.6	Smart Helmet Compliance Detection Using Enhanced YOLO-M	13
2.7	Visual Fall Detection from Activities of Daily Living for Assistive Living	15
2.8	A Deep Learning Framework for the Classification of Brazilian Coins	17
2.9	Human-Centered AI in Smart Farming: Toward Agriculture 5.0	19
2.10	Omni-Directional Obstacle Detection for Vehicles Based on Depth Camera	21
5.1	Data Flow Diagram – Level 0	32
5.2	Data Flow Diagram – Level 1	32
5.3	Data Flow Diagram – Level 2	33
5.4	Entity Relationship Diagram	34
6.1	System Architecture	36
8.1	Add Emergency Contact	46
8.2	Edit Emergency Contact	46
8.3	Detailed View of One Emergency Contact	47
8.4	List Emergency Contact	47
8.5	Create Task	48
8.6	Lists the Tasks	48
8.7	login.tsk	49
8.8	register.tsk	49
8.9	time.tsk	50
8.10	weather.tsk	50
8.11	where-am-i.tsk	51
8.12	Home Page	52
8.13	Tasks and Create Tasks Pages	53
8.14	Weather Page	54
8.15	Emergency Contact Pages	55
8.16	Where Am I Page	56
8.17	Time Page	57
8.18	Emergency Alerts	58

LIST OF TABLES

No.	Title	Page No.
2.1	Comparison of Related Works	23
6.1	User	38
6.2	Message Queue	38
6.3	Emergency Contacts	39
6.4	Tasks	40

CHAPTER 1
INTRODUCTION

With the rapid advancement of artificial intelligence, embedded systems, and mobile computing technologies, the development of intelligent assistive systems has gained significant attention in recent years. Among various application domains, assistive technologies for visually impaired individuals have emerged as a critical area of research due to the increasing need to enhance independent mobility, safety, and quality of life. Visually impaired individuals often face considerable challenges in navigating complex and dynamic environments, where the lack of visual perception limits their ability to detect obstacles, recognize objects, and make informed decisions in real time. Traditional assistive tools such as white canes and guide dogs provide only limited support, as they are primarily restricted to short-range obstacle detection and do not offer contextual or semantic understanding of the surroundings. As a result, there exists a growing demand for intelligent systems capable of providing real-time environmental awareness and adaptive guidance.

In recent years, various approaches have been proposed to address these challenges, including sensor-based systems, smartphone applications, and wearable devices. Sensor-based solutions, such as ultrasonic obstacle detectors, are effective in identifying nearby objects but lack the ability to classify or interpret them. Smartphone-based applications utilize computer vision techniques to provide navigation assistance; however, many of these systems rely on cloud-based processing, which introduces latency, increases dependency on internet connectivity, and raises concerns regarding data privacy and reliability. Furthermore, most existing solutions focus on a single functionality, such as obstacle detection or navigation guidance, resulting in fragmented systems that fail to provide comprehensive assistance. These limitations highlight the need for an integrated, efficient, and real-time assistive system capable of addressing multiple aspects of mobility and safety within a unified framework.

To overcome these limitations, the integration of computer vision, machine learning, and edge computing has emerged as a promising approach for developing next-generation assistive technologies. Computer vision techniques enable systems to interpret visual data captured from the environment, while machine learning models provide the capability to detect and classify objects with high accuracy. Edge computing further enhances system performance by enabling data processing directly on local devices, thereby reducing latency and eliminating the need for continuous cloud connectivity. Additionally, advancements in mobile application development have made it possible to design accessible and user-friendly interfaces tailored specifically for visually impaired users, incorporating features such as voice feedback and simplified interaction mechanisms.

In this context, the proposed system, B.E.A.C.O.N (Blind Enhanced and Assistive Cap with Optical Navigation), introduces a comprehensive assistive solution that combines wearable hardware, artificial intelligence, and mobile application technologies. The system employs an ESP32-CAM module mounted on a wearable cap to capture real-time visual data, which is transmitted through a CH340C interface to the processing unit. Artificial intelligence models developed using Python are utilized to perform key tasks such as obstacle detection, crosswalk recognition, and currency identification. These models operate in real time to analyze environmental data and generate appropriate alerts based on the detected conditions. The processed information is then communicated to a mobile application developed using React Native, while backend functionalities are managed using Node.js, ensuring efficient coordination between different system components.

In addition to environmental perception, the system incorporates multiple safety and utility features designed to enhance user experience and reliability. These include fall detection mechanisms with automatic alert generation, manual emergency alert triggering, customizable emergency contact management, weather condition analysis to assist in decision-making, real-time announcements of time and location, and a reminder-based alarm system for daily activities. The system operates using a trigger-based architecture, where specific events such as obstacle detection, user interaction, or abnormal motion initiate appropriate responses. This integrated approach enables the system to function as both a navigation aid and a personal safety assistant, providing comprehensive support to visually impaired users.

The primary contribution of this project lies in the development of a unified assistive system that integrates multiple functionalities within a single platform, combining real-time computer vision, edge-based processing, and accessible mobile interaction. By addressing the limitations of existing solutions and incorporating advanced technologies, the proposed system aims to improve environmental awareness, enhance navigation safety, and promote independence among visually impaired individuals. The subsequent chapters of this report provide a detailed discussion of the theoretical background, system analysis, design methodology, implementation details, and evaluation of the proposed system, along with an exploration of its potential applications and future enhancements.

CHAPTER 2
LITERATURE SURVEY

2.1 Paper 1: Safety Helmet Detection Based on Improved YOLOv8

2.1.1 Aim

This research aims to improve real-time safety helmet detection in complex construction environments where challenges such as small object size, cluttered backgrounds, and varying lighting conditions exist. The study focuses on enhancing the detection accuracy of helmets using an improved YOLOv8 model. The ultimate goal is to ensure worker safety by enabling reliable and fast identification of helmet usage, thereby reducing workplace accidents and enforcing safety compliance effectively.

2.1.2 Methodology

The study utilizes an enhanced version of the YOLOv8n model as the base architecture for object detection. To improve performance, Mosaic data augmentation is applied to generate diverse training samples, especially for small object detection. A Coordinate Attention (CA) mechanism is integrated into the network to help the model focus more accurately on helmet regions within complex scenes.

Additionally, a slim-neck architecture is introduced to optimize feature fusion while maintaining computational efficiency. A dedicated small-target detection layer is incorporated to improve detection of helmets that appear small or distant in images.

The model is trained on annotated datasets containing helmet and non-helmet images and is evaluated using real-time detection scenarios to validate accuracy, speed, and robustness.

2.1.3 Workflow

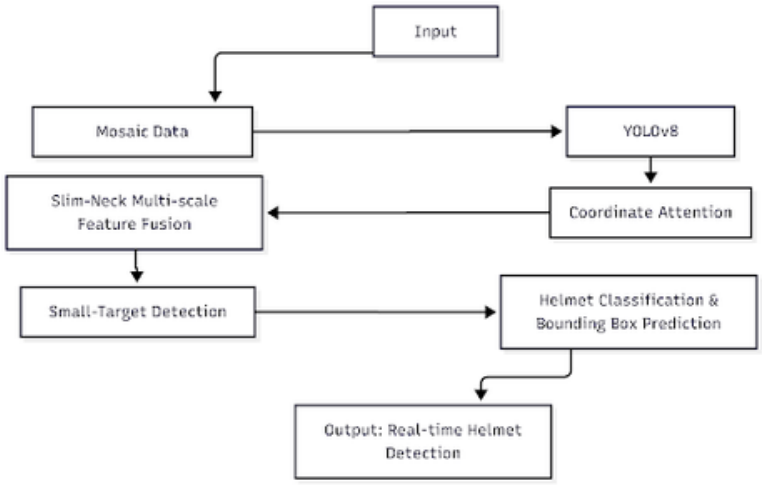


Fig.2.1 : Safety Helmet Detection Based on Improved YOLOv8

2.1.4 Advantages

- Provides high accuracy in detecting safety helmets, even in complex environments
- Effectively handles small object detection using enhanced architecture
- Real-time detection capability suitable for live monitoring systems
- Improved feature extraction using attention mechanisms
- Reduces workplace accidents by ensuring safety compliance
- Can be integrated with surveillance systems for automated monitoring

2.1.5 Disadvantages

- Requires high computational resources for training and deployment
- Performance may decrease in poor lighting or occluded conditions
- Needs a large, well-annotated dataset for optimal accuracy
- Complex model architecture increases implementation difficulty
- May produce false positives in highly cluttered backgrounds
- Deployment on low-power edge devices can be challenging

2.2 Paper 2: Smart Helmet 5.0 for Industrial Internet of Things Using Artificial Intelligence

2.2.1 Aim

This research aims to develop a smart helmet system integrated with Artificial Intelligence and Industrial Internet of Things (IIoT) technologies to enhance workplace safety in hazardous industrial environments. The system focuses on real-time monitoring of environmental conditions and worker activities to detect potential risks and prevent accidents. The primary goal is to create an intelligent, connected safety solution that provides timely alerts and improves overall occupational safety and efficiency.

2.2.2 Methodology

The system incorporates multiple sensors to collect real-time data related to environmental and user conditions, including temperature, humidity, gas levels (VOC), brightness, motion, and shock detection. These sensors are integrated with an ESP32 microcontroller, which processes and transmits the collected data to an IoT platform such as ThingsBoard for monitoring and analysis.

Various machine learning models, including Support Vector Machine (SVM), Naive Bayes, Static Neural Networks, and Convolutional Neural Networks (CNN), are evaluated to analyze the sensor data. Among these, CNN is selected due to its superior accuracy (approximately 92%).

The system continuously monitors incoming data and uses the trained AI model to detect abnormal conditions or potential hazards. When a risk is identified, real-time alerts are generated and sent through the IoT platform to ensure immediate response and preventive action.

2.2.3 Workflow

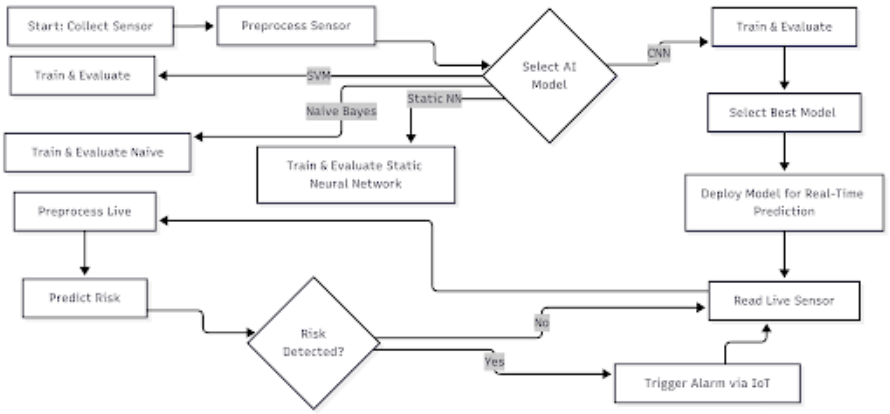


Fig.2.2 :Smart Helmet 5.0 for Industrial Internet of Things Using Artificial Intelligence

2.2.4 Advantages

- Provides real-time monitoring of both environmental and worker conditions
- High accuracy in risk detection using AI models (CNN)
- Multi-sensor integration enables comprehensive safety analysis
- Instant alerts through IoT platforms improve response time
- Suitable for hazardous industrial environments such as mining and construction
- Enhances worker safety and reduces accident risks

2.2.5 Disadvantages

- Limited processing capability of microcontrollers may restrict performance
- Relies heavily on IoT connectivity for real-time communication
- Uses only non-visual sensor data, limiting contextual understanding
- Requires proper calibration and maintenance of multiple sensors
- May need further real-world testing for reliability in diverse environments
- Power consumption can be a concern for continuous operation

2.3 Paper 3: Computer Vision Based Detection and Classification of Road Obstacles

2.3.1 Aim

This research aims to develop a computer vision-based system for detecting and classifying road obstacles to improve safety in both autonomous and assistive navigation systems. The system focuses on identifying various types of obstacles such as vehicles, pedestrians, and barriers in real-time. The primary objective is to enhance situational awareness and enable safer navigation by providing accurate and timely information about surrounding road conditions.

2.3.2 Methodology

The system uses camera-based input to continuously capture real-time road scenes. The captured images undergo preprocessing steps such as noise reduction, resizing, and normalization to improve image quality and prepare them for analysis.

Feature extraction techniques and machine learning or deep learning models are then applied to identify relevant patterns in the visual data. The model is trained on large datasets containing labeled images of different types of road obstacles.

Once trained, the system classifies detected objects into categories such as vehicles, pedestrians, or static obstacles. The detection and classification process operates in real-time, allowing the system to continuously monitor the environment and support navigation decisions.

2.3.3 Workflow

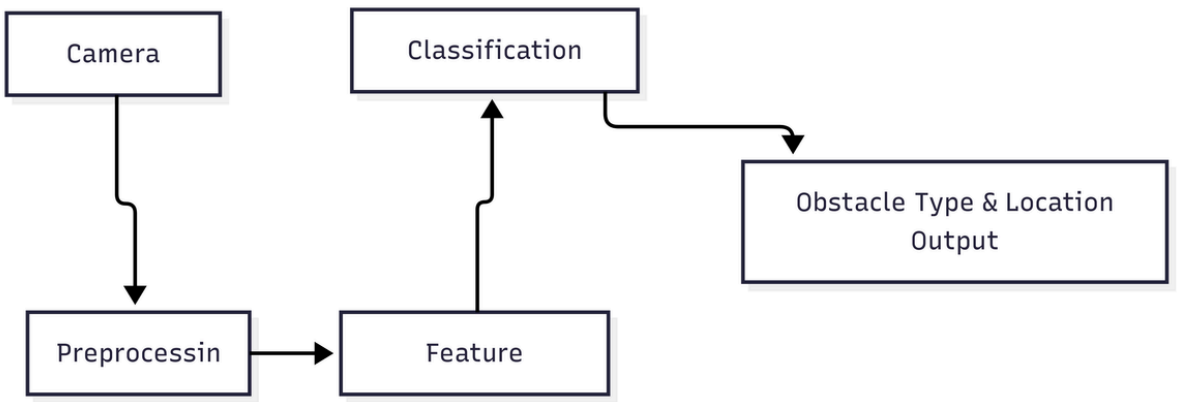


Fig.2.3 : Computer Vision Based Detection and Classification of Road Obstacles

2.3.4 Advantages

- Enhances road safety through real-time obstacle detection
- Capable of identifying and classifying multiple types of obstacles
- Provides continuous monitoring of dynamic environments
- Can be integrated with autonomous vehicles and assistive systems
- Improves decision-making through visual understanding of surroundings
- Scalable with advancements in deep learning models

2.3.5 Disadvantages

- Performance decreases in poor lighting or adverse weather conditions
- Computationally intensive, requiring powerful processing units
- Requires large and diverse datasets for accurate training
- May struggle with occluded or partially visible objects
- Real-time processing can introduce latency in low-resource systems
- Accuracy may vary depending on camera quality and positioning

2.4 Paper 4: Obstacle Detection and Warning System for Visually Impaired Using IoT Sensors

2.4.1 Aim

This research aims to develop a wearable obstacle detection and warning system for visually impaired individuals using IoT-based sensors. The system focuses on providing real-time assistance by detecting nearby obstacles and alerting users to enhance safe navigation. The primary objective is to improve mobility, independence, and safety for visually impaired users in both indoor and outdoor environments.

2.4.2 Methodology

The system utilizes ultrasonic and infrared sensors to detect obstacles and measure their distance from the user. These sensors are integrated with a microcontroller such as Arduino or Raspberry Pi, which processes the sensor data in real time.

When an obstacle is detected within a certain range, the system generates alerts through output devices such as buzzers, vibration motors, or voice modules. The intensity or type of alert may vary depending on the distance of the obstacle.

The entire system is designed as a wearable device, ensuring portability and ease of use. Continuous sensing and processing enable real-time feedback, allowing users to respond quickly to potential hazards.

2.4.3 Workflow

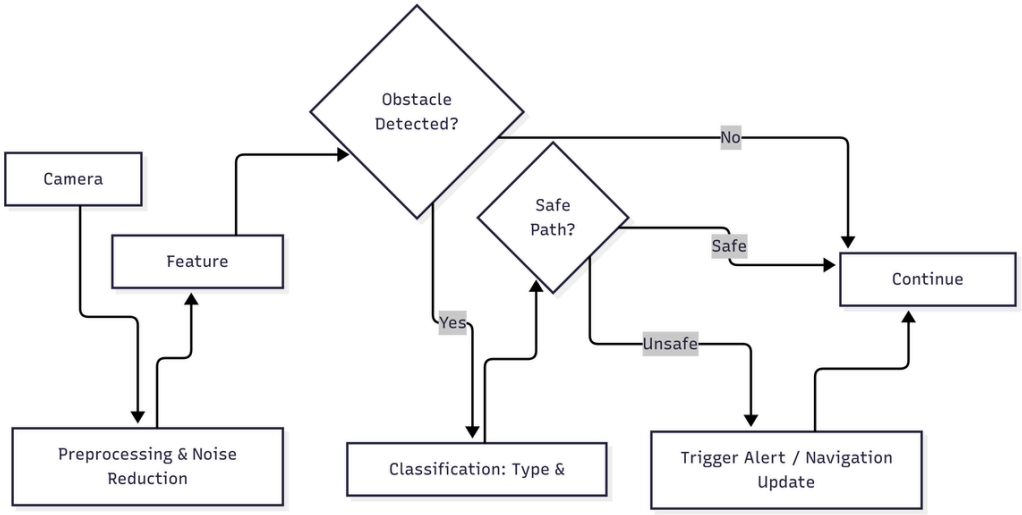


Fig.2.4 : Obstacle Detection and Warning System for Visually Impaired Using IoT Sensors

2.4.4 Advantages

- Affordable and cost-effective solution for assistive navigation
- Provides real-time obstacle detection and alerts
- Simple design and easy implementation
- Portable and suitable for daily use
- Improves independence and mobility of visually impaired users
- Low power consumption compared to vision-based systems

2.4.5 Disadvantages

- Limited detection range compared to advanced vision systems
- Cannot classify or identify types of obstacles
- Accuracy may be affected by environmental noise or interference
- Difficulty in detecting small or fast-moving objects
- Limited effectiveness in crowded or complex environments
- Provides only proximity information without contextual awareness

2.5 Paper 5: Personal Voice Assistant Security and Privacy

2.5.1 Aim

This research aims to analyze the security and privacy challenges associated with personal voice assistants. The study focuses on identifying potential threats related to data collection, storage, and user authentication in voice-enabled systems. The primary objective is to improve the safety, reliability, and trustworthiness of voice assistants by addressing vulnerabilities and enhancing privacy protection mechanisms.

2.5.2 Methodology

The study conducts a comprehensive review of existing voice assistant systems and their underlying technologies, particularly speech recognition and natural language processing. It analyzes how user data is collected, transmitted, and stored within these systems.

Various security threats, including spoofing attacks, unauthorized access, and data leakage, are examined to understand system vulnerabilities. The research also evaluates existing authentication methods such as voice recognition, multi-factor authentication, and encryption techniques.

Based on the analysis, the study identifies weaknesses in current systems and proposes improvements to enhance security and privacy in voice assistant technologies.

2.5.3 Workflow

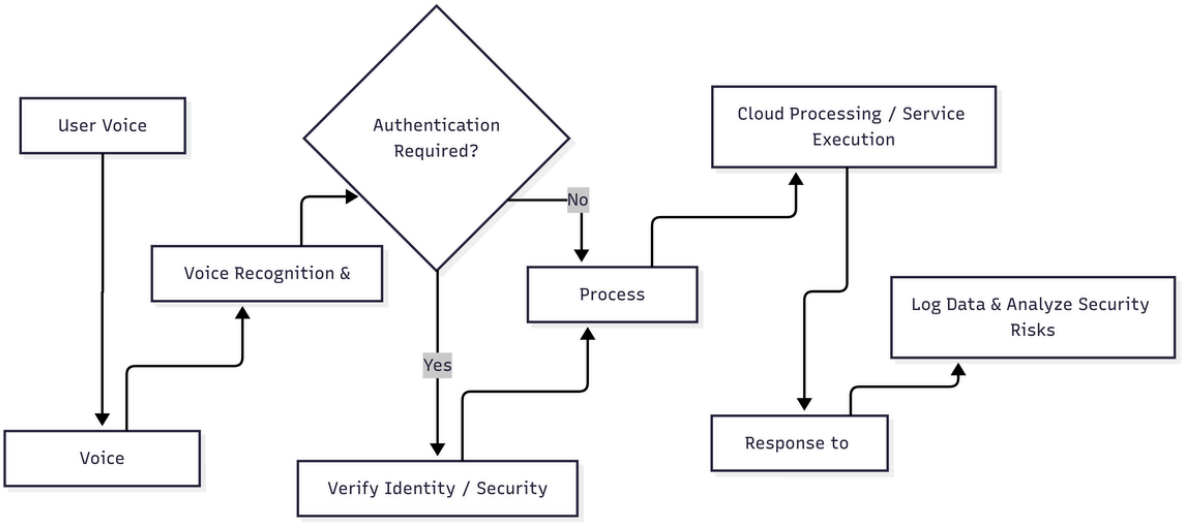


Fig.2.5: Personal Voice Assistant Security and Privacy

2.5.4 Advantages

- Enables hands-free interaction and improved accessibility
- Enhances user convenience in daily tasks
- Supports personalized user experiences
- Improves efficiency in device control and information retrieval
- Encourages adoption of smart technologies
- Provides insights for improving security in voice-based systems

2.5.5 Disadvantages

- Privacy risks due to continuous data collection and listening
- Vulnerable to spoofing and voice imitation attacks
- Sensitive data may be exposed through cloud storage
- Dependence on internet connectivity for full functionality
- Misinterpretation of commands in noisy environments
- Limited user awareness of data usage and permissions

2.6 Paper 6: Smart Helmet Compliance Detection Using Enhanced YOLO-M

2.6.1 Aim

This research aims to develop an automated system for detecting helmet compliance in industrial and construction environments using an enhanced YOLO-M deep learning model. The system focuses on identifying whether workers are wearing safety helmets in real time to reduce the risk of head injuries. The primary goal is to improve workplace safety by enabling continuous monitoring and instant alerts for non-compliance.

2.6.2 Methodology

The system utilizes the YOLO-M deep learning model, which is optimized for real-time object detection. A dataset containing images of workers with and without helmets is collected and augmented to improve model robustness.

The model is trained to distinguish between helmet and non-helmet cases using supervised learning techniques. Once trained, the system is deployed on edge AI devices to enable low-latency inference without relying on cloud processing.

The detection results are integrated with IoT or mobile-based systems to generate real-time alerts whenever a worker is detected without a helmet. This allows for immediate intervention and improved safety monitoring.

2.6.3 Workflow

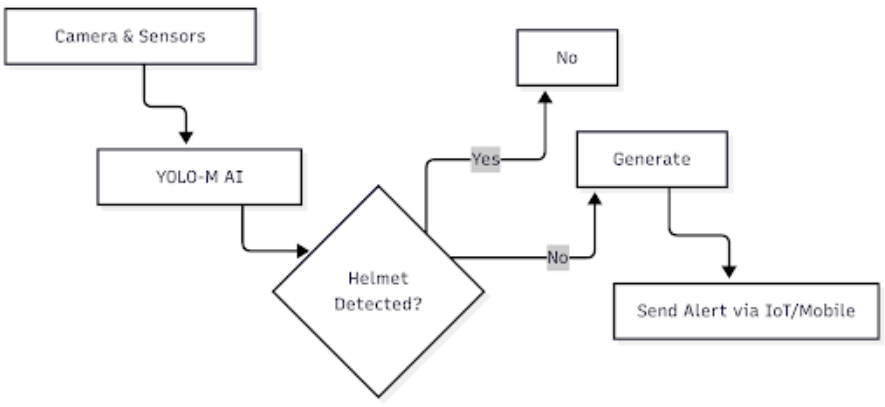


Fig.2.6: Smart Helmet Compliance Detection Using Enhanced YOLO-M

2.6.4 Advantages

- High accuracy in detecting helmet compliance
- Real-time monitoring and alert generation
- Low latency due to edge AI deployment
- Reduces need for manual supervision
- Improves workplace safety and regulatory compliance
- Can be integrated with existing surveillance systems

2.6.5 Disadvantages

- Limited to detecting helmets only, not other safety violations
- Performance may degrade in low-light or occluded environments
- Requires a large and diverse dataset for training
- Edge AI hardware can increase deployment cost
- May produce false detections in crowded scenes
- Needs periodic model updates for improved accuracy

2.7 Paper 7: Visual Fall Detection from Activities of Daily Living for Assistive Living

2.7.1 Aim

This research aims to develop a vision-based system for detecting falls in elderly and disabled individuals during their daily activities. The system focuses on distinguishing between normal activities and dangerous fall events in real time. The primary objective is to enhance safety and provide immediate assistance by enabling continuous monitoring in assistive living environments.

2.7.2 Methodology

The system uses video input captured through cameras installed in the environment to monitor user activities continuously. The captured video data is processed using deep learning models trained to recognize human actions and movements.

The model is trained on datasets containing various daily activities and fall scenarios to accurately differentiate between normal behavior and abnormal events such as falls. Feature extraction and classification techniques are applied to identify patterns associated with falls.

When a fall is detected, the system triggers alerts to caregivers or emergency services, enabling quick response and assistance. The system operates in real time to ensure immediate detection and notification.

2.7.3 Workflow

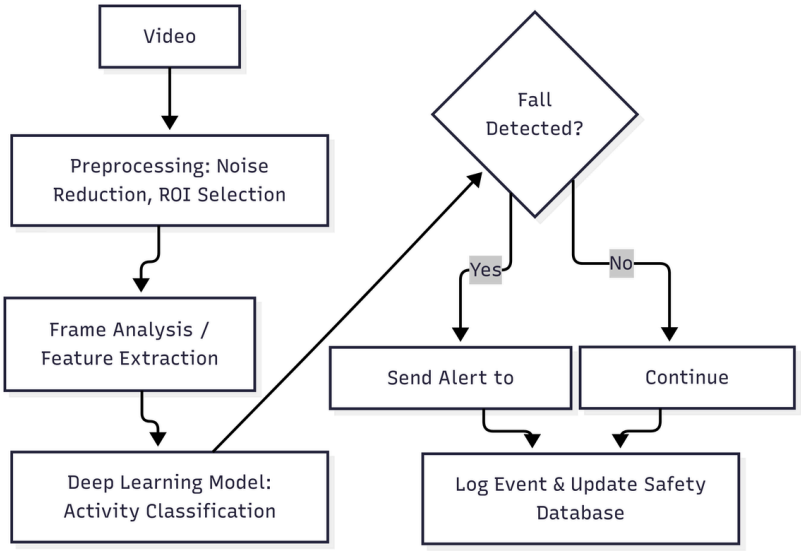


Fig.2.7: Visual Fall Detection from Activities of Daily Living for Assistive Living

2.7.4 Advantages

- Provides continuous monitoring for elderly and disabled individuals
- High accuracy in detecting fall events using deep learning
- Enables quick response through real-time alert systems
- Improves safety and reduces risk of unattended injuries
- Useful in healthcare and assisted living environments
- Can be integrated with smart home systems

2.7.5 Disadvantages

- Requires continuous video surveillance, raising privacy concerns
- High computational requirements for real-time processing
- Performance depends on camera placement and coverage
- May produce false alarms in complex activity scenarios
- Requires large labeled datasets for accurate training
- Not suitable for environments with limited visibility

2.8 Paper 8: A Deep Learning Framework for the Classification of Brazilian Coins

2.8.1 Aim

This research aims to develop a deep learning-based system for the automatic recognition and classification of Brazilian coins. The system focuses on identifying different coin denominations accurately using image processing techniques. The primary objective is to assist visually impaired individuals and general users in handling and recognizing currency efficiently, thereby improving accessibility and usability in everyday transactions.

2.8.2 Methodology

The system begins by capturing images of coins using a camera or image acquisition device. These images undergo preprocessing steps such as resizing, normalization, and noise reduction to enhance quality and consistency.

A Convolutional Neural Network (CNN) model is then trained on a labeled dataset of Brazilian coins, where each image corresponds to a specific denomination. The model learns distinguishing features such as size, texture, and patterns of the coins.

Once trained, the system classifies input coin images into their respective categories in real time. The output can be provided through audio or visual feedback, making it especially useful for visually impaired users.

2.8.3 Workflow

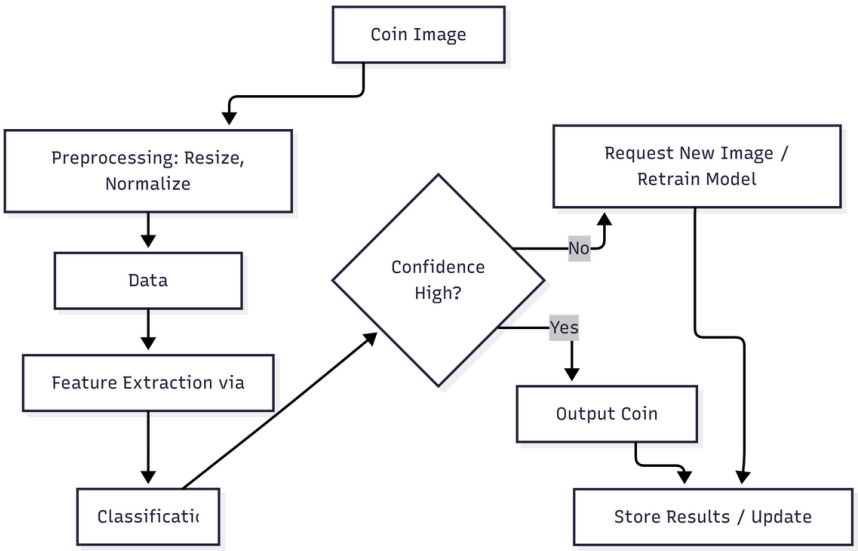


Fig.2.8: A Deep Learning Framework for the Classification of Brazilian Coins

2.8.4 Advantages

- Enables accurate and automated currency recognition
- Assists visually impaired users in daily transactions
- High classification accuracy using deep learning models
- Reduces human effort and errors in identifying coins
- Can be implemented in mobile or embedded systems
- Scalable to support multiple currencies

2.8.5 Disadvantages

- Limited to trained coin types and denominations
- Performance may be affected by poor image quality or lighting
- Requires a large and well-labeled dataset for training
- Computational requirements for model training and inference
- Difficulty in distinguishing worn or damaged coins
- May require frequent updates for new currency designs

2.9 Paper 9: Human-Centered AI in Smart Farming: Toward Agriculture 5.0

2.9.1 Aim

This research aims to explore the concept of human-centered artificial intelligence in smart farming environments, focusing on the integration of AI, IoT sensors, and human decision-making. The study emphasizes developing adaptive AI systems that support real-time decision-making while ensuring usability and interpretability for users. The primary objective is to create intelligent systems that effectively collaborate with humans to improve efficiency, productivity, and safety in complex real-world environments.

2.9.2 Methodology

The system integrates IoT sensors to collect real-time data related to environmental conditions such as soil quality, temperature, humidity, and crop status. This data is processed using advanced AI models to perform predictive analysis and generate actionable insights.

The approach follows a human-centered design, where AI outputs are structured in a way that is understandable and useful for human users. The system continuously adapts based on feedback and changing environmental conditions.

Simulation and real-world testing are conducted in agricultural settings to evaluate system performance, adaptability, and usability. The interaction between human users and AI systems is analyzed to improve collaboration and decision-making efficiency.

2.9.3 Workflow

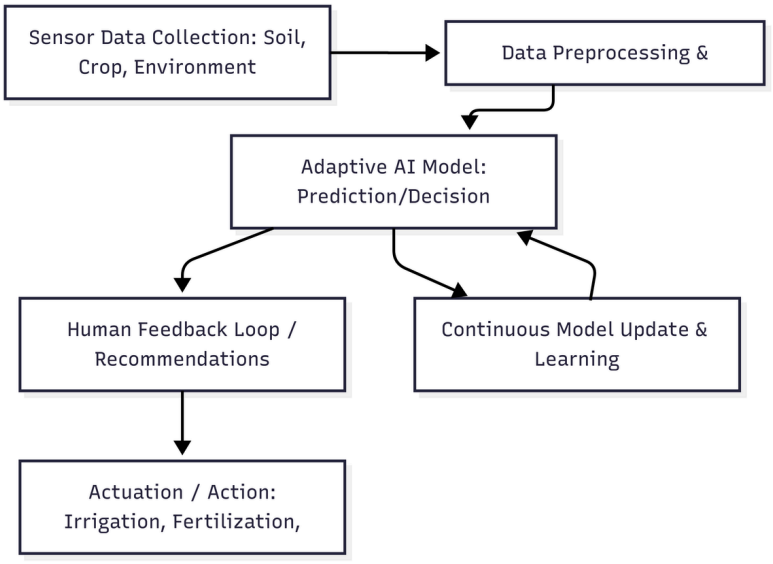


Fig.2.9: Human-Centered AI in Smart Farming: Toward Agriculture 5.0

2.9.4 Advantages

- Promotes human-centered AI design with better usability and interpretability
- Supports adaptive decision-making in dynamic environments
- Integrates IoT sensors with AI for real-time data-driven insights
- Enhances efficiency and productivity through intelligent recommendations
- Applicable to various real-world domains beyond agriculture
- Encourages effective collaboration between humans and AI systems

2.9.5 Disadvantages

- Primarily focused on agriculture, requiring adaptation for other domains
- Complex integration of multiple sensors and AI models
- High computational and implementation complexity
- Requires continuous user feedback and system tuning
- May be time-consuming to evaluate human-AI interaction effectiveness
- Deployment cost can be high for large-scale systems

2.10 Paper 10: Omni-Directional Obstacle Detection for Vehicles Based on Depth Camera

2.10.1 Aim

This research aims to develop an omni-directional obstacle detection system for vehicles using depth camera technology. The system focuses on detecting obstacles from all directions and accurately estimating their distance in real time. The primary objective is to enhance vehicle safety by providing comprehensive environmental awareness and enabling timely alerts or automated responses to prevent collisions.

2.10.2 Methodology

The system uses depth cameras to capture 3D spatial information of the surrounding environment. These cameras generate depth maps that provide detailed distance measurements of objects in all directions.

The captured depth data is processed using computer vision and AI algorithms to detect and identify obstacles. The system calculates the distance and position of objects relative to the vehicle.

Based on this information, real-time alerts are generated for drivers, or the data is fed into autonomous driving systems for decision-making. In advanced implementations, the system can trigger automatic actions such as braking or steering to avoid collisions.

2.10.3 Workflow

Fig.2.10: Human-Centered AI in Smart Farming: Toward Agriculture 5.0

2.10.4 Advantages

- Promotes human-centered AI design with better usability and interpretability
- Supports adaptive decision-making in dynamic environments
- Integrates IoT sensors with AI for real-time data-driven insights
- Enhances efficiency and productivity through intelligent recommendations
- Applicable to various real-world domains beyond agriculture
- Encourages effective collaboration between humans and AI systems

2.10.5 Disadvantages

- Primarily focused on agriculture, requiring adaptation for other domains
- Complex integration of multiple sensors and AI models
- High computational and implementation complexity
- Requires continuous user feedback and system tuning
- May be time-consuming to evaluate human-AI interaction effectiveness
- Deployment cost can be high for large-scale systems

2.11 Comparison of Related Works

Table 2.1: Comparison of Related Works

PAPER TITLE	ADVANTAGES	DISADVANTAGES
Safety Helmet Detection Based on Improved YOLOv8	• High accuracy in helmet detection • Works in real-time	• Needs high computational power • Performance affected by lighting
Smart Helmet 5.0 for Industrial IoT Using AI	• Multi-sensor monitoring • Real-time IoT alerts	• Depends on IoT connectivity • Limited processing capability
Computer Vision Based Detection and Classification of Road Obstacles	• Detects multiple obstacle type • Real-time performance	• Affected by weather/lighting • High computation required
Obstacle Detection for Visually Impaired Using IoT Sensors	• Low cost and portable • Real-time alerts	• Cannot classify objects • Limited detection range
Personal Voice Assistant Security and Privacy	• Hands-free interaction • Improves accessibility	• Privacy risks • Vulnerable to spoofing attacks
Smart Helmet Compliance Detection Using YOLO-M	• High detection accuracy; • Low-latency edge processing	• Limited to helmet detection • Cost of edge devices
Visual Fall Detection for Assistive Living	• Accurate fall detection • Useful for healthcare	• Privacy concerns • Requires continuous monitoring
Deep Learning Framework for Brazilian Coins	• Helps visually impaired users • High classification accuracy	• Limited to trained coins • Sensitive to image quality
Human-Centered AI in Smart Farming	• Adaptive AI decision-making • Human-friendly design	• Domain-specific • Complex implementation
Omni-Directional Obstacle Detection Using Depth Camera	• 360° detection • Accurate distance measurement	• High cost • Requires calibration

CHAPTER 3
SYSTEM ANALYSIS

3.1 Existing System

Existing assistive systems for visually impaired individuals primarily include traditional tools such as white canes and guide dogs, along with modern technological solutions like sensor-based devices and smartphone applications. White canes are widely used due to their simplicity and affordability; however, they are limited to detecting obstacles within a short range and cannot provide information about object type, direction, or severity of potential hazards. Guide dogs offer dynamic assistance but require extensive training, high maintenance costs, and are not accessible to all users.

In recent years, electronic assistive devices have been developed using ultrasonic sensors and infrared sensors to detect nearby obstacles. While these systems improve detection capability compared to traditional methods, they lack semantic understanding and are unable to classify objects or interpret complex environmental conditions. Smartphone-based applications using computer vision techniques have also been introduced, providing features such as object recognition and navigation assistance. However, many of these systems rely on cloud-based processing, which introduces latency, requires continuous internet connectivity, and raises concerns related to privacy and reliability.

Furthermore, most existing solutions focus on a single functionality, such as obstacle detection, navigation, or object recognition, rather than providing a comprehensive assistive system. This fragmented approach reduces usability and increases the cognitive load on users, as they must rely on multiple tools or applications to meet their daily needs

3.2 Problem Statement

Despite significant advancements in assistive technologies, visually impaired individuals still face considerable challenges in navigating real-world environments safely and independently. One of the major limitations of existing systems is the lack of real-time environmental awareness, as traditional tools and basic sensor-based devices are unable to interpret complex surroundings or provide contextual information about detected objects.

Another key challenge is the inability of current systems to detect and classify different types of obstacles and environmental features, such as pedestrian crosswalks or currency notes, which are essential for independent daily activities. Many existing solutions also fail to provide directional guidance, making it difficult for users to make informed navigation decisions.

In addition, reliance on cloud-based processing in some modern systems results in increased latency and dependency on internet connectivity, making them unreliable in outdoor or low-network environments. Privacy concerns further arise when user data is transmitted to external servers. The lack of accessible and intuitive user interfaces also poses a challenge, as visually impaired users require systems that provide clear and simple interaction mechanisms.

Safety is another critical concern, as most existing systems do not include features such as fall detection, emergency alerts, or real-time assistance during critical situations. The absence of an integrated solution that combines environmental perception, navigation assistance, and personal safety features highlights a significant gap in current assistive technologies.

Therefore, there is a need for a unified, intelligent, and real-time assistive system that can enhance environmental awareness, provide accurate navigation guidance, and ensure user safety through an accessible and reliable platform.

3.3 Proposed System

To address the limitations of existing systems, the proposed B.E.A.C.O.N system introduces a comprehensive assistive solution that integrates wearable hardware, artificial intelligence, and mobile application technologies. The system is designed to provide real-time environmental awareness, navigation assistance, and safety features within a single unified platform.

The hardware component consists of a smart cap equipped with an ESP32-CAM module, which continuously captures visual data from the user's surroundings. This data is transmitted through a CH340C interface to the processing system. Artificial intelligence models developed using Python are employed to analyze the captured images and perform tasks such as obstacle detection, crosswalk recognition, and currency identification. These models operate in real time to ensure accurate and timely detection of environmental features.

The processed information is then communicated to a mobile application developed using React Native, which serves as the primary user interface. The application backend, implemented using Node.js, manages system functionalities and ensures efficient communication between components. The application provides voice-based feedback, enabling users to receive real-time information about their surroundings in an accessible manner.

In addition to environmental perception, the system incorporates several advanced features to enhance user safety and usability. These include fall detection with automatic alert generation, manual emergency alert triggering, customizable emergency contact management, weather condition analysis, real-time announcements of time and location, and a reminder-based alarm system for daily activities. The system operates using a trigger-based mechanism.

CHAPTER 4
FEASIBILITY STUDY

Feasibility study is an important phase in system development that evaluates whether the proposed system is practical, implementable, and beneficial. It helps in analyzing various aspects such as cost, technical requirements, operational usability, implementation constraints, and performance considerations. The feasibility of the proposed B.E.A.C.O.N system is analyzed under the following categories.

4.1 Economic Feasibility

The proposed system is economically feasible as it utilizes cost-effective and widely available components. The primary hardware requirements include an ESP32-CAM module, CH340C interface chip, a mobile phone, and a power bank for portable power supply. These components are relatively inexpensive and easily accessible in the market, making the system affordable for large-scale deployment.

Additionally, the system leverages existing smartphones for processing and user interaction, eliminating the need for dedicated high-cost computing devices. The software components used in the system, such as React Native for mobile application development, Node.js for backend services, and Python for artificial intelligence model development, are open-source technologies. This significantly reduces development and licensing costs.

Overall, the low-cost hardware setup combined with open-source software tools ensures that the system is economically viable and suitable for real-world applications.

4.2 Technical Feasibility

The proposed system is technically feasible as it is built using established technologies and proven methodologies. The ESP32-CAM module is capable of capturing real-time images and transmitting data efficiently, while the CH340C interface facilitates communication between hardware components.

The artificial intelligence models developed using Python are capable of performing tasks such as obstacle detection, crosswalk recognition, and currency identification. These models can be optimized for real-time performance, ensuring minimal delay in processing.

The mobile application developed using React Native ensures cross-platform compatibility and provides an accessible user interface. The backend implemented using Node.js efficiently handles application logic and communication between system components.

Since all required technologies are well-supported, widely used, and compatible with each other, the system is technically feasible and can be successfully implemented.

4.3 Operational Feasibility

Operational feasibility evaluates how effectively the system can be used by end users. The proposed system is designed with a strong focus on usability and accessibility for visually impaired individuals.

The system provides voice-based feedback, enabling users to interact with the application without relying on visual input. The interface is simple and intuitive, ensuring that users can easily access various features such as navigation, alerts, and reminders. The double-tap interaction mechanism further enhances usability by preventing accidental operations.

The wearable design of the smart cap allows hands-free operation, making it convenient for daily use. Additional features such as fall detection, emergency alerts, and weather updates improve safety and reliability.

Overall, the system is highly user-friendly and meets the operational requirements of visually impaired users.

4.4 Implementation Feasibility

The implementation of the proposed system is feasible using standard development tools and readily available hardware components. The integration of hardware and software components can be achieved through well-defined communication protocols.

The ESP32-CAM module can be programmed to capture and transmit image data, which is then processed using Python-based AI models. The processed results are communicated to the mobile application, where further actions are triggered through the Node.js backend.

The use of modular design simplifies development and testing, allowing individual components such as AI models, mobile application, and backend services to be developed and integrated independently. This reduces complexity and improves system reliability.Since all components are compatible and development tools are widely available, the system can be implemented within the given project timeline

4.5 Time Complexity and Performance Feasibility

In the context of this project, time complexity refers to the system's ability to process data and generate responses in real time. The proposed system is designed to ensure low latency and efficient performance.

The use of optimized artificial intelligence models enables fast processing of image data, allowing real-time detection of obstacles, crosswalks, and currency. Edge-based processing through the smartphone reduces dependency on external servers, ensuring quicker response times.

The system operates in a continuous processing loop, where image capture, analysis, and feedback occur simultaneously with minimal delay. This ensures that users receive timely alerts and guidance, which is critical for safety and navigation.

Therefore, the system meets performance requirements and is suitable for real-time assistive applications.

CHAPTER 5
REQUIREMENT SPECIFICATIONS

Requirements specification is an important phase in system development that defines the functional and non-functional needs of the proposed system. It provides a clear understanding of what the system is expected to perform and the conditions under which it should operate. For the B.E.A.C.O.N system, the requirements are categorized into functional requirements, which describe the core features and operations, and non-functional requirements, which define the performance, usability, and reliability aspects of the system.

5.1 Functional Requirements

- Functional requirements describe the primary operations that the system must perform to assist visually impaired users effectively. The proposed system is designed to provide real-time environmental awareness, navigation assistance, and safety features through the integration of artificial intelligence and mobile technologies.

- The system must be capable of detecting both stationary and moving obstacles in real time using camera-based input. This ensures that users are alerted about objects in their path and can avoid potential collisions. The detection process should be continuous and responsive to changes in the environment.

- The system should provide navigation guidance through audio and haptic alerts. These alerts must be clear, timely, and easy to understand, enabling users to move safely in different environments. Audio feedback serves as the primary mode of communication, while haptic feedback acts as a supporting mechanism.

- The system must be able to recognize text, currency, and environmental signs. This feature allows users to perform daily tasks independently, such as identifying currency denominations and reading important information from their surroundings.

- The system must detect falls or collisions and automatically trigger alerts to emergency contacts. In addition, users should have the ability to manually trigger alerts when required. These safety features ensure immediate assistance during emergencies.

- The system may optionally support monitoring of vital signs, which can provide additional health-related information and enhance user safety.

- The system must process data locally using edge artificial intelligence techniques. This ensures faster response times, reduces latency, and minimizes dependency on internet connectivity.

- Finally, the system must communicate with the mobile application using wireless technologies such as Bluetooth or Wi-Fi. This communication enables data transfer, system coordination, and user interaction.

5.2 Non-Functional Requirements

- Non-functional requirements define the quality attributes and performance standards of the system. These requirements ensure that the system operates efficiently, reliably, and securely in real-world conditions.

- The system must provide real-time performance with minimal latency. Since the application involves navigation and safety, delays in processing or feedback must be minimized to ensure timely responses.

- The system should be lightweight, comfortable, and easy to use. The wearable design must allow users to use the system for extended periods without discomfort.

- The system must be reliable in different environments, including indoor and outdoor settings, as well as under varying lighting conditions such as low-light environments. This ensures consistent performance regardless of external factors.

- The system should be scalable, allowing future enhancements such as improved AI models, additional features, or extended functionalities to be integrated without major redesign.
- The system must be portable and should function with minimal dependency on external devices. While a smartphone is used for processing and interaction, the system should maintain efficiency and usability in a portable setup.

- Finally, the system must ensure data security and user privacy. Sensitive user data should be protected, and secure communication methods must be used to prevent unauthorized access.

5.3 Usecase Diagram

The use case diagram represents the interaction between the user and the proposed B.E.A.C.O.N system. It illustrates how different functionalities of the system are accessed and utilized by the user. The primary actor in the system is the visually impaired user, who interacts with the system through the mobile application and wearable device.

The major use cases include obstacle detection, navigation assistance, currency and text recognition, fall detection and emergency alert triggering, weather checking, time and location announcement, and alarm management. The system processes user inputs and environmental data to provide appropriate responses through audio and haptic feedback.

The use case diagram helps in understanding the functional flow of the system and the relationship between different system operations.

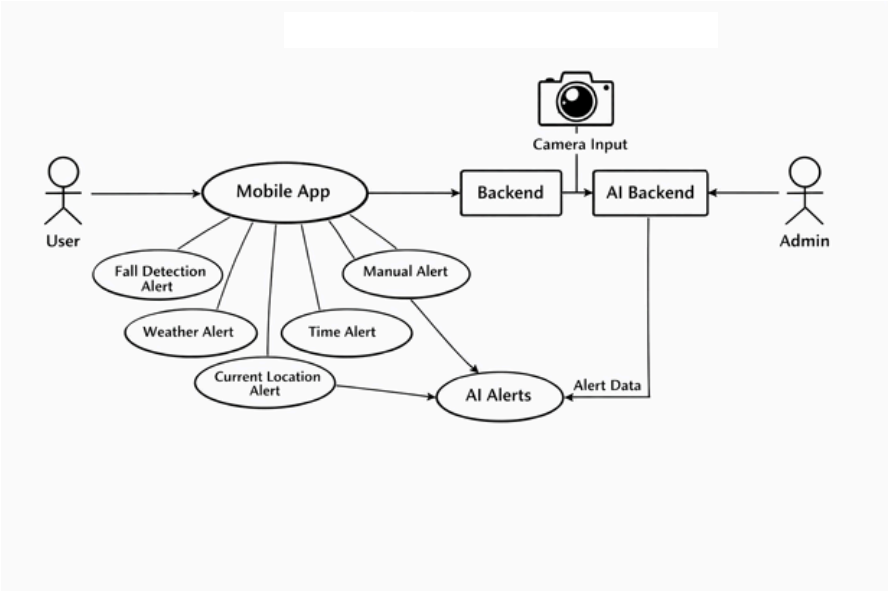


Figure 6.1: Use Case Diagram of the Proposed System

5.4 Dataflow Diagram

The data flow diagram (DFD) illustrates how data moves within the B.E.A.C.O.N system, from input acquisition to processing and output generation. It provides a clear understanding of how different components of the system interact with each other.

The system begins with data input from the camera mounted on the cap, which captures real-time environmental images. This data is transmitted to the processing unit, where artificial intelligence models analyze the input and detect relevant

level 0 : Context Diagram.

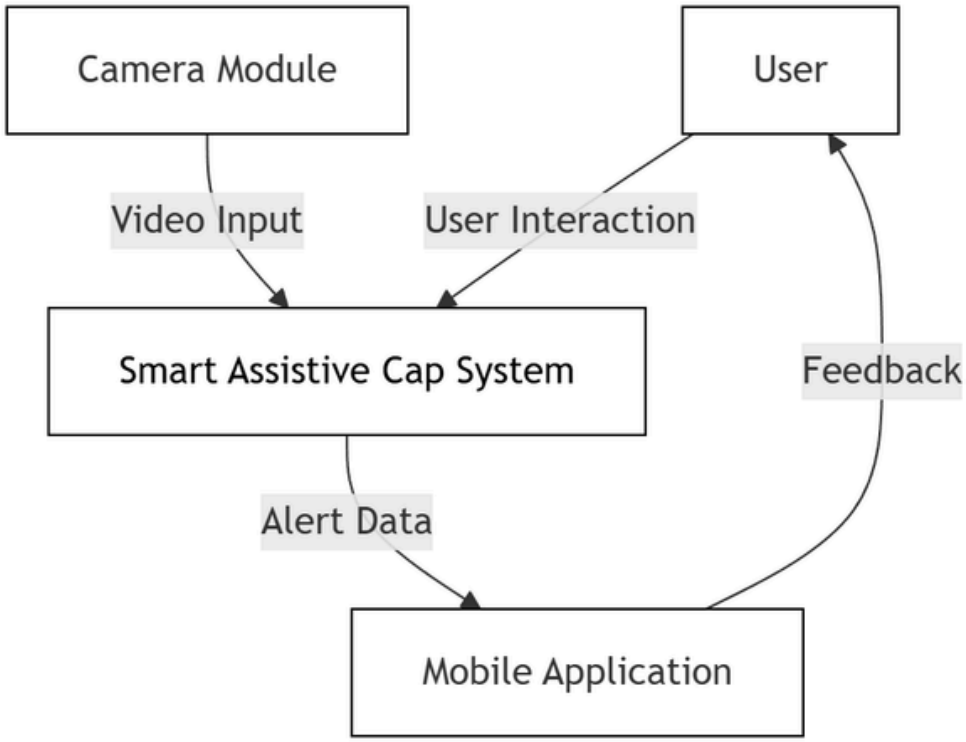


Figure 5.1: Data Flow Diagram – Level 0

level 1 : Main System Processes

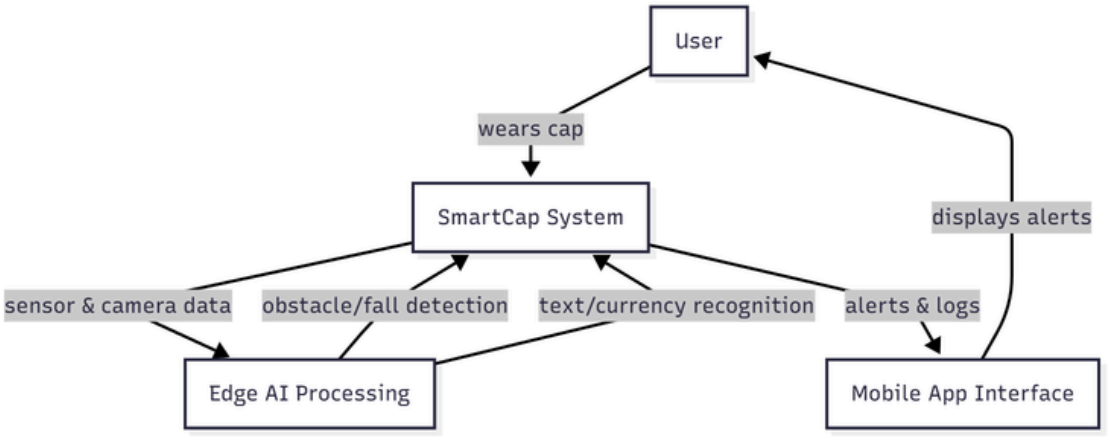


Figure 5.2: Data Flow Diagram – Level 1

level 2 : Detailed Route Computation.

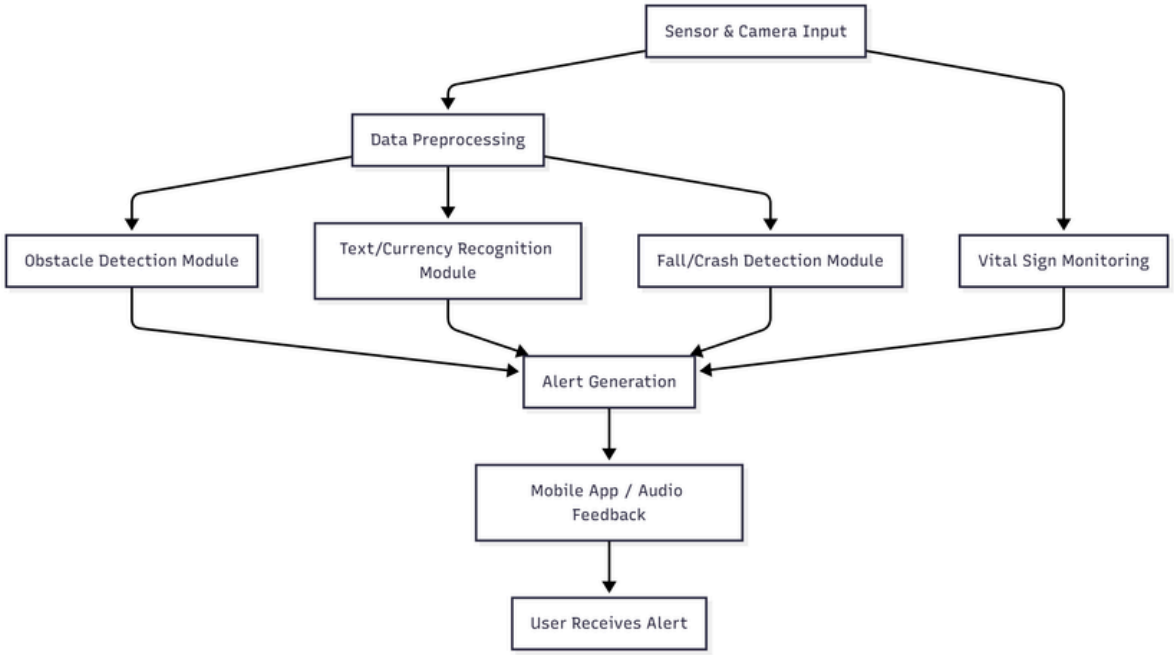


Figure 5.3: Data Flow Diagram – Level 2

5.5 Entity Relationship Diagram

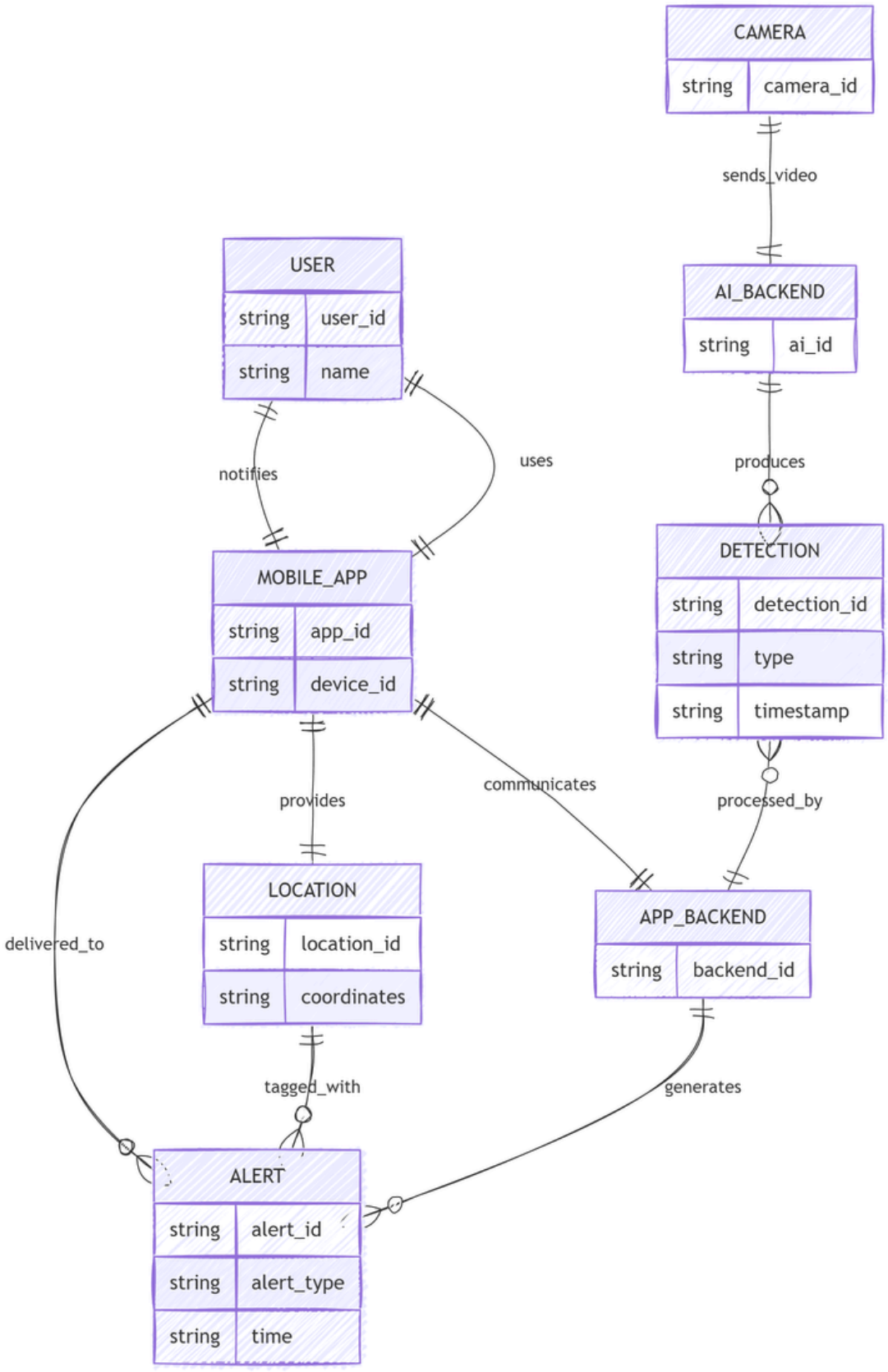


Fig 5.4 Entity Relationship Diagram

CHAPTER 6
SYSTEM DESIGN

6.1 Software Architecture

The system architecture of the proposed B.E.A.C.O.N system is designed as an integrated framework combining hardware components, artificial intelligence modules, and a mobile application. The architecture ensures efficient data flow, real-time processing, and seamless interaction between different system components.

The hardware layer consists of the ESP32-CAM module mounted on a wearable cap, which continuously captures visual data from the user's surroundings. The module is connected through a CH340C interface, enabling communication and data transmission to the processing system. A power bank is used to provide portable power supply, ensuring continuous operation.

The processing layer includes artificial intelligence models developed using Python. These models are responsible for performing tasks such as obstacle detection, crosswalk recognition, and currency identification. The AI models analyze incoming image data and generate appropriate outputs based on trained patterns.

The communication layer facilitates data transfer between the hardware and the mobile application using wireless technologies such as Bluetooth or Wi-Fi. This ensures smooth and real-time communication between components.

The application layer consists of a mobile application developed using React Native, with backend support provided by Node.js. The application acts as the central interface, receiving processed data and delivering feedback to the user through voice and haptic alerts. It also manages additional functionalities such as fall detection, emergency alerts, weather updates, and reminders.
This layered architecture ensures scalability, modularity, and efficient system performance.

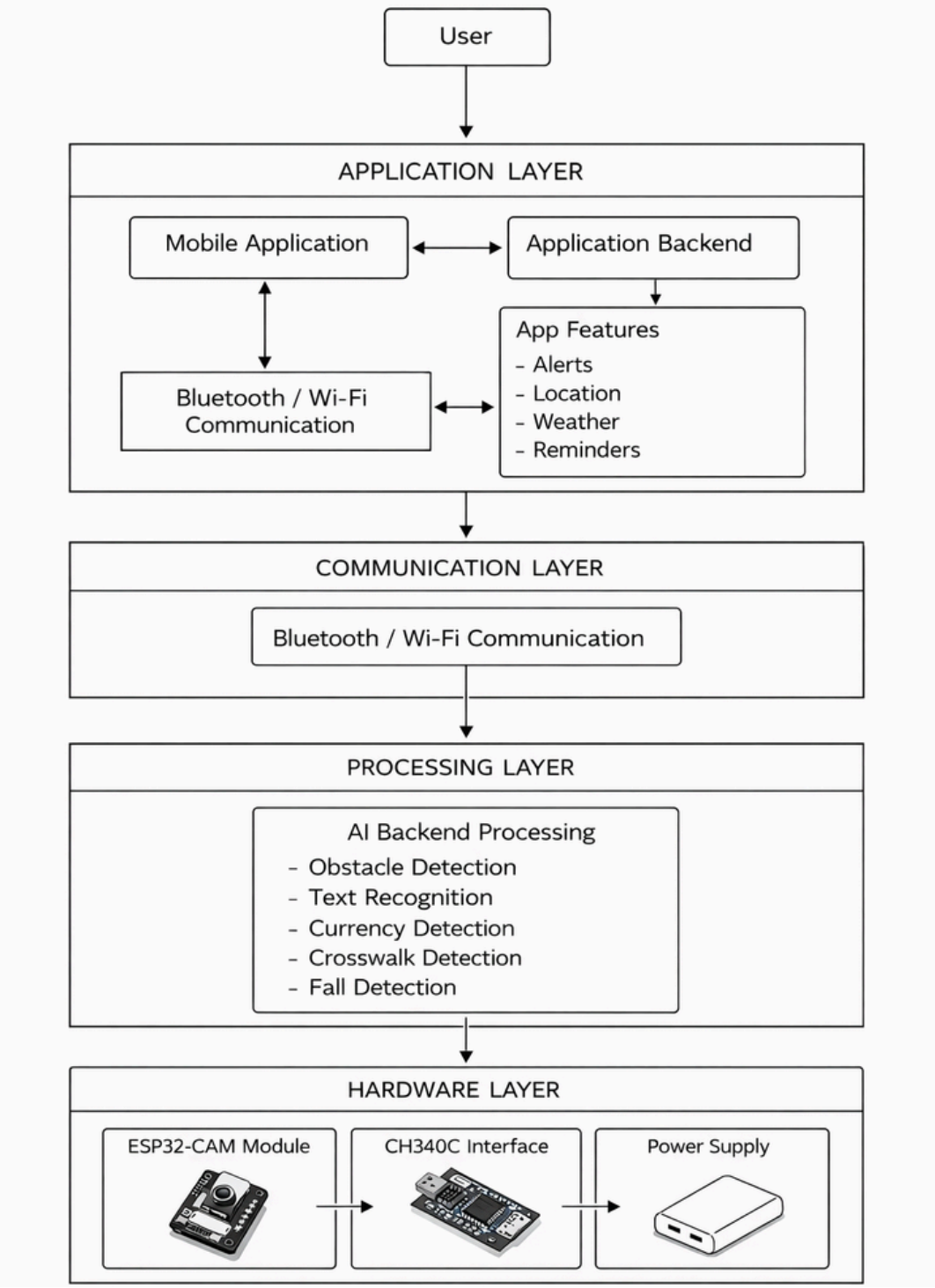


Figure 6.1: System Architecture

6.2 Working Principle

The working of the B.E.A.C.O.N system is based on continuous data acquisition, processing, and feedback generation. The system operates in real time to assist users in navigating their surroundings safely.

Initially, the camera mounted on the cap captures real-time images of the environment. These images are transmitted to the processing unit through the ESP32-CAM module. The captured data is then passed to the artificial intelligence models for analysis.

The AI models process the incoming frames and detect relevant objects such as obstacles, crosswalks, and currency. Based on the detection results, the system determines the appropriate response or alert. For example, if an obstacle is detected, the system generates a warning message indicating its presence and direction.

The processed information is then sent to the mobile application, which converts it into audio or haptic feedback. The user receives these alerts in real time, allowing them to make informed decisions while navigating.

Simultaneously, the mobile application monitors additional conditions such as fall detection and user-triggered alerts. If a fall or emergency is detected, the system automatically sends notifications to predefined contacts. Other features such as weather updates, time announcements, and reminders operate based on user requests or predefined triggers.

This continuous cycle of sensing, processing, and feedback ensures efficient and reliable system operation.

6.3 Database Design

The database design of the proposed B.E.A.C.O.N system is structured to efficiently manage user-related data, system-generated alerts, and application-level interactions. The database plays a crucial role in supporting functionalities such as user management, emergency communication, and system notifications. Each entity in the system is represented using structured tables with clearly defined attributes and relationships.

The primary table in the database is the user table, which stores essential information such as user ID, name, email, and mobile number. This table acts as the central entity, with other tables linked to it using foreign keys. Additional tables are used to manage alerts, messages, and reminders generated during system operation. Relationships between tables are maintained using constraints such as primary keys, foreign keys, and unique attributes to ensure data integrity.

The database is designed to be simple, scalable, and efficient, allowing easy integration of additional features in the future. Mechanisms such as referential integrity are used to maintain consistency when records are updated or deleted. Overall, the database structure ensures reliable data management and supports the smooth functioning of the B.E.A.C.O.N system.

Table 6.1: User

SL.NO	COLUMN NAME	TYPE	CONSTRAINTS	DESCRIPTION
1	ID	String	PK, Default: cuid()	Unique user ID
2	Email	String	Unique, Nullable	User email
3	Name	String		Name of the user
4	Password	String		User password
5	Created at	DateTime	Default: now()	Account creation time

Table 6.2: Message Queue

SL.NO	COLUMN NAME	TYPE	CONSTRAINTS	DESCRIPTION
1	ID	String	PK, Default: cuid()	Unique user ID
2	Email	String	Unique, Nullable	User email
3	Message	String	Default: now()	Account creation time

Table 6.3: Emergency Contacts

SL.NO	COLUMN NAME	TYPE	CONSTRAINTS	DESCRIPTION
1	ID	String	PK, Default: cuid()	Contact ID
2	UserID	String	FK→ User.id. Cascade Delete	Owner user
3	Email	String	Unique (with userld), Optional	Contact email
4	MobileNo	String	Unique (with userld), Optional	Contact phone
5	Name	String		Contact name
6	Position	Int		Priority
7	Created at	DateTime	Default: now()	Created time

Table 6.4: Tasks

SL.NO	COLUMN NAME	TYPE	CONSTRAINTS	DESCRIPTION
1	ID	String	PK, Default: cuid()	Task ID
2	Userid	String	FK→ User.id. Cascade Delete	Owner user
3	Name	String		Task name
4	Scheduledtime	String		Time
5	Specificdate	DateTime	Optional	One-time task date
6	Isactive	Boolean		Active status
7	Isresponding	Boolean		Repeating task
8	Repeatdays	WeekDays []	Default: []	Days of repeatition
9	Create at	DateTime	Default: now()	Created time
10	Last triggered	DateTime	Optional	Last run time

CHAPTER 7
IMPLEMENTATION AND TESTING

7.1 Minimum Hardware Requirements

The hardware requirements for the proposed system are minimal and cost-effective, making the system practical for real-world implementation.

The system requires an ESP32-CAM module, which is used to capture real-time visual data from the user's surroundings. A CH340C interface chip is used to establish communication between the hardware components. A mobile phone is required to run the application and perform processing tasks. Additionally, a power bank is used to provide portable power supply for continuous operation of the system.

These components are lightweight, affordable, and easily available, ensuring ease of implementation.

7.2 Minimum Software Requirements

The software components of the system are developed using widely used and open-source technologies.

The mobile application is developed using React Native, which allows cross-platform compatibility and efficient user interface design. The backend functionalities are implemented using Node.js, which handles communication, processing logic, and system coordination.

Artificial intelligence models are developed using Python, which provides powerful libraries for machine learning and image processing. These models are responsible for performing detection tasks such as obstacle detection, crosswalk recognition, and currency identification.

The use of these technologies ensures flexibility, scalability, and ease of development.

7.3 Types of Testing Conducted

Testing is a critical phase in the development of the proposed system to ensure that all components function correctly, reliably, and efficiently under different conditions. Various testing methods are conducted to validate individual modules, system integration, performance, and overall usability. The following types of testing are carried out:

1. Unit Testing

Unit testing is performed to verify the functionality of individual components of the system in isolation. Each module, such as the AI models for obstacle detection, crosswalk recognition, and currency identification, is tested separately to ensure correct operation. Similarly, individual features of the mobile application, including voice feedback, alert triggering, and reminder functionalities, are tested independently. This helps in identifying and fixing errors at an early stage of development.

2. Integration Testing

Integration testing is conducted to ensure that different modules of the system work together seamlessly. This includes testing the communication between the ESP32-CAM module, AI processing unit, and mobile application. Data flow from image capture to processing and feedback generation is verified. The interaction between frontend (React Native) and backend (Node.js) is also tested to ensure proper coordination. This testing ensures that combined components function as a unified system without errors.

3. Functional Testing

Functional testing is performed to verify that all system features operate according to the specified requirements. Key functionalities such as real-time obstacle detection, navigation guidance, currency recognition, fall detection, and emergency alert triggering are tested under various scenarios. The system is evaluated to ensure that it provides accurate outputs and appropriate responses for different inputs. This testing confirms that the system meets its intended objectives.

4. Performance Testing

Performance testing evaluates the efficiency and responsiveness of the system. The system is tested to ensure real-time processing with minimal latency, which is crucial for navigation and safety applications. The response time of AI models, data transmission speed, and feedback generation are analyzed. The system is also tested under continuous operation to ensure stability and consistent performance over time.

5. Security Testing

Security testing is carried out to ensure that user data is protected and that the system is safe from unauthorized access. The communication between system components, especially between the mobile application and backend, is tested for secure data transfer. Measures such as data encryption and secure access control are verified.

6. User Acceptance Testing

User acceptance testing (UAT) is conducted to evaluate the system from the end user's perspective. Visually impaired users or test participants interact with the system to assess its usability, accessibility, and effectiveness. The ease of use of the voice-based interface, clarity of alerts, and overall user experience are evaluated. Feedback collected during this phase helps in refining the system to better meet user needs.

7. Regression Testing

Regression testing is performed after system updates or modifications to ensure that existing functionalities continue to work correctly. Whenever changes are made to the system, such as updates to AI models or application features, regression testing is conducted to verify that no new issues are introduced. This helps maintain system stability and reliability throughout the development lifecycle.

7.4 Test Cases

Test cases are used to verify the correctness and reliability of the system by testing different functionalities under specific conditions.

• Test Case 1: Obstacle Detection

The system is tested by placing an object in front of the camera. The expected result is that the system detects the obstacle and provides an audio alert to the user.

• Test Case 2: Moving Object Detection

A moving object such as a person is introduced in front of the camera. The system should detect the movement and provide real-time alerts.

• Test Case 3: Crosswalk Detection

The system is tested in an environment with a crosswalk. The expected result is that the system recognizes the crosswalk and provides navigation guidance.

• Test Case 4: Currency Recognition

Different currency notes are shown to the camera. The system should correctly identify and announce the denomination.

• Test Case 5: Text Recognition

Text or signboards are provided as input. The system should read and announce the text accurately.

• Test Case 6: Fall Detection

A fall scenario is simulated. The system should detect the fall and automatically trigger an emergency alert.

• Test Case 7: Manual Alert

The user manually triggers an alert using the application. The system should send notifications to the predefined emergency contacts.

• Test Case 8: Navigation Guidance

The navigation feature is activated. The system should provide correct directional instructions such as left or right.

• Test Case 9: Voice Feedback

User interacts with the application. The system should provide accurate voice responses.

• Test Case 10: Communication Test

Data transmission between hardware and mobile application is tested. The system should successfully communicate without errors.

CHAPTER 8
RESULTS AND DISCUSSION

The proposed B.E.A.C.O.N system was successfully developed and tested to evaluate its performance in real-time scenarios. The system integrates hardware components, artificial intelligence models, and a mobile application to assist visually impaired users in navigating their surroundings effectively.

The obstacle detection module was tested in both indoor and outdoor environments. The system was able to detect stationary as well as moving objects with satisfactory accuracy. Real-time audio alerts were provided to the user, enabling safe navigation and avoiding potential hazards.

The crosswalk detection feature was evaluated in road-like environments, and the system successfully identified pedestrian crossing areas. Based on the detection, appropriate navigation guidance was provided, improving the user's ability to cross roads safely.

The currency recognition functionality was tested using different denominations of currency notes. The system accurately identified and announced the currency values, allowing users to perform transactions independently.

The mobile application functioned effectively as the central interface of the system. Voice feedback was clear and responsive, enabling users to interact with the system easily. Features such as time and location announcements, weather updates, and reminder alarms were tested and found to be working correctly.

Safety features such as fall detection and manual alert triggering were also tested. The system successfully detected simulated fall events and automatically sent alerts to predefined emergency contacts. Manual alerts were also triggered effectively through the application.

The system demonstrated reliable performance with minimal latency due to the use of edge-based processing. Communication between hardware components and the mobile application via wireless technologies such as Bluetooth or Wi-Fi was stable and efficient.

Overall, the results indicate that the system is capable of providing real-time assistance, improving navigation safety, and enhancing independence for visually impaired individuals.

8.1 Screenshots

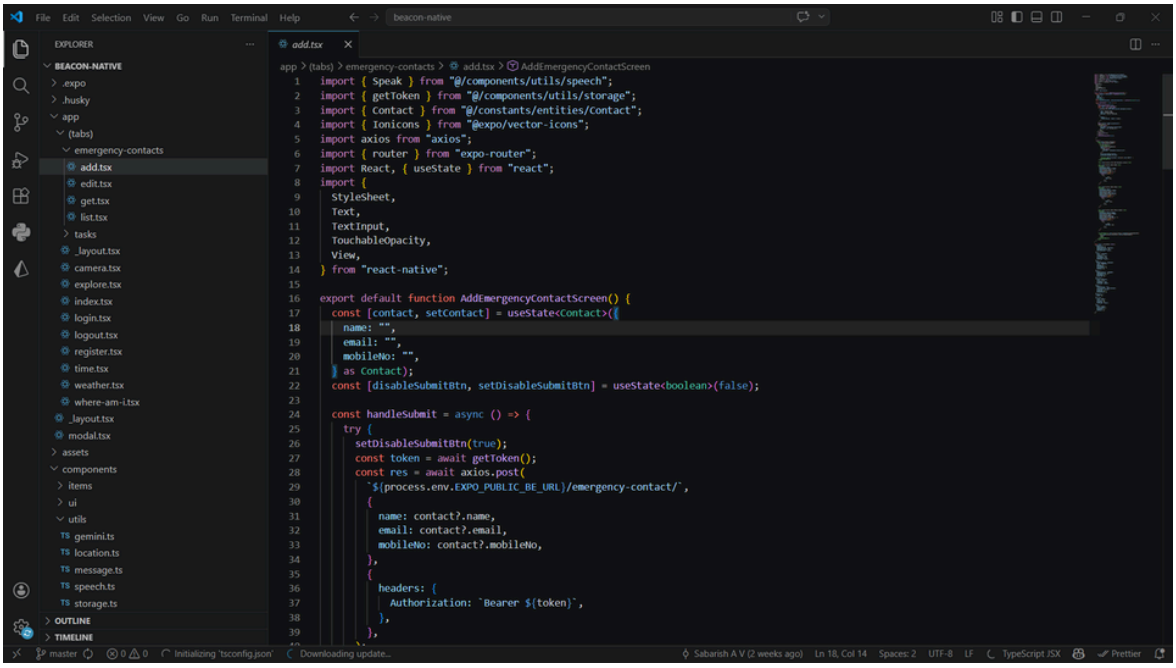


Figure 8.1: Add emergency contact

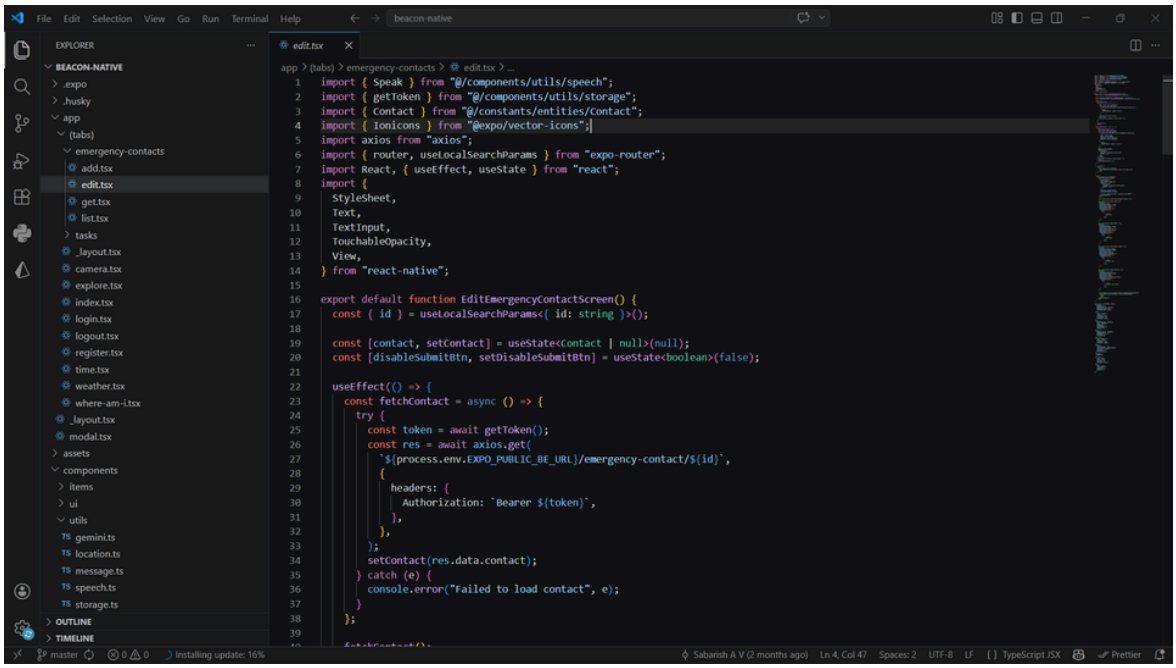


Figure 8.2: Edit emergency contact

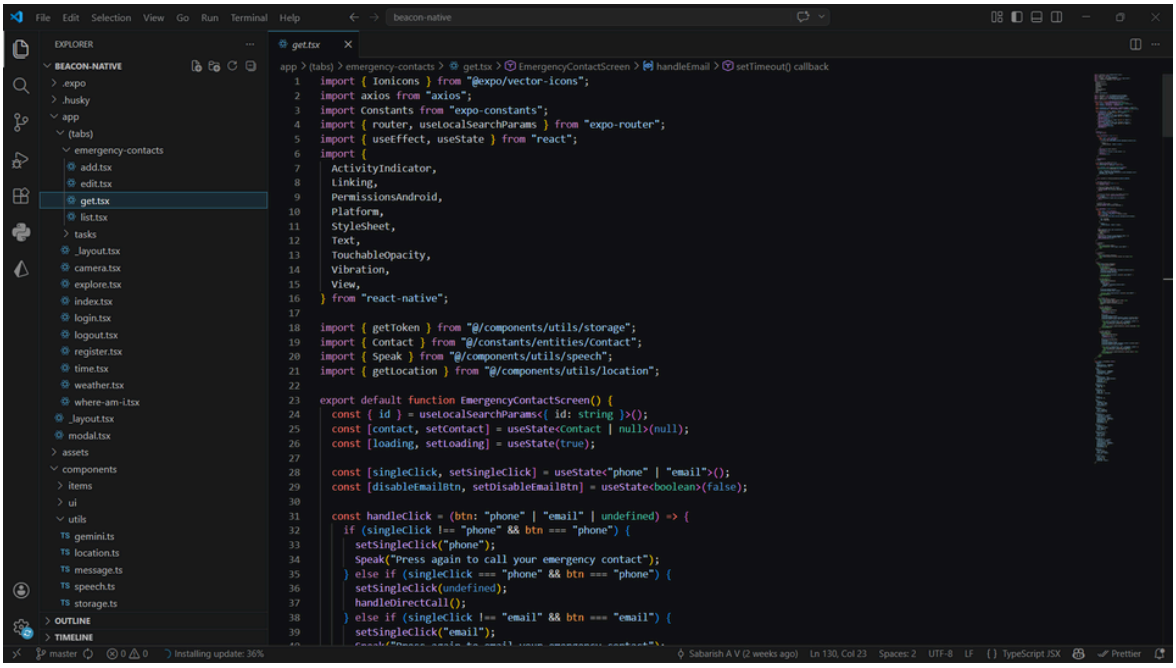


Figure 8.3: Detailed view of one emergency conatct

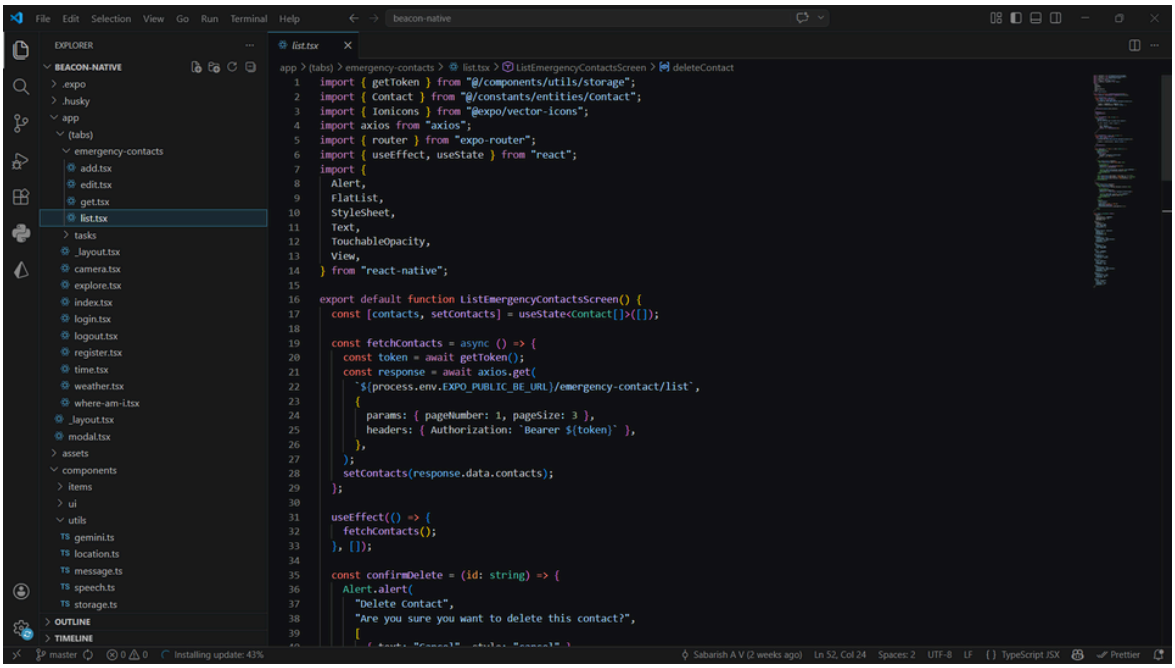
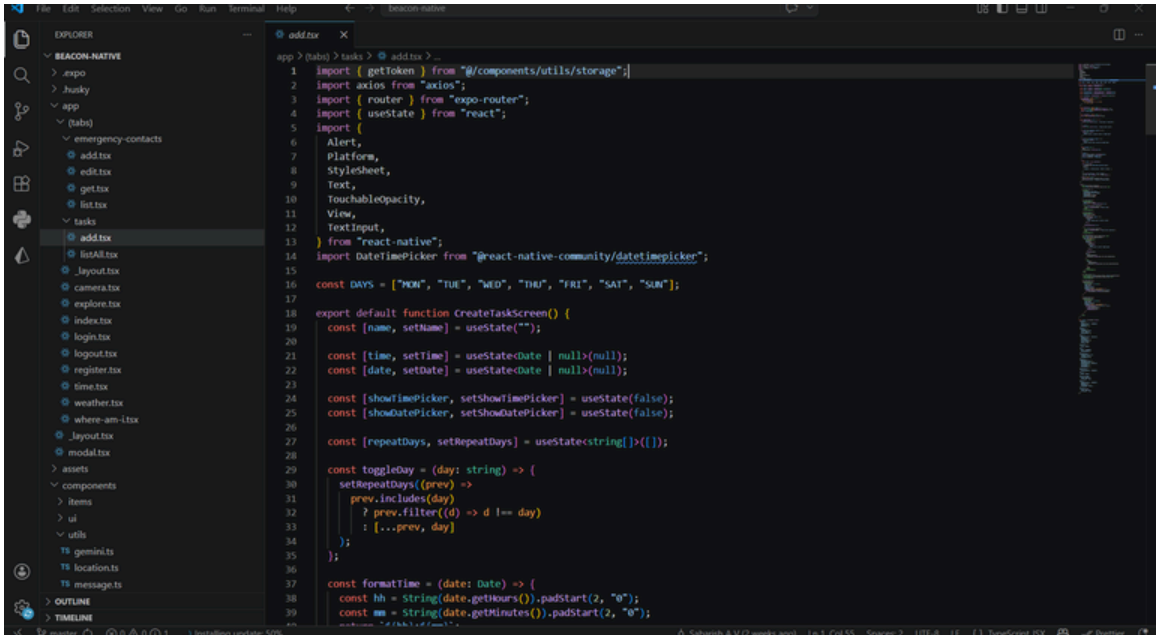
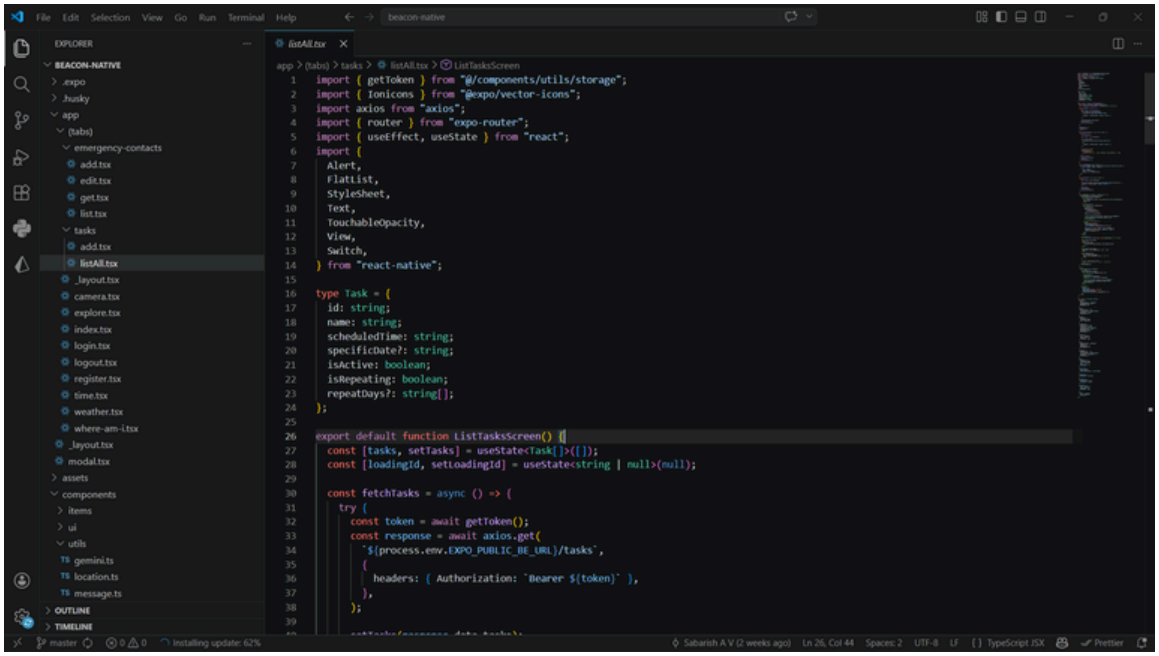


Figure 8.4: List emergency contact



```tsx
import { getToken } from "@/components/utils/storage";
import axios from "axios";
import { router } from "expo-router";
import { useState } from "react";
import {
  Alert,
  Platform,
  StyleSheet,
  Text,
  TouchableOpacity,
  View,
  TextInput,
} from "react-native";
import DateTimePicker from "@react-native-community/datetimepicker";

const DAYS = ["MON", "TUE", "WED", "THU", "FRI", "SAT", "SUN"];

export default function CreateTaskScreen() {
  const [name, setName] = useState("");

  const [time, setTime] = useState<Date | null>(null);
  const [date, setDate] = useState<Date | null>(null);

  const [showTimePicker, setShowTimePicker] = useState(false);
  const [showDatePicker, setShowDatePicker] = useState(false);

  const [repeatDays, setRepeatDays] = useState<string[]>([]);

  const toggleDay = (day: string) => {
    setRepeatDays((prev) =>
      prev.includes(day)
        ? prev.filter((d) => d !== day)
        : [...prev, day]
    );
  };

  const formatTime = (date: Date) => {
    const hh = String(date.getHours()).padStart(2, "0");
    const mm = String(date.getMinutes()).padStart(2, "0");
```

Figure 8.5: Create task



```tsx
import { getToken } from "@/components/utils/storage";
import { Ionicons } from "@expo/vector-icons";
import axios from "axios";
import { router } from "expo-router";
import { useEffect, useState } from "react";
import {
  Alert,
  FlatList,
  StyleSheet,
  Text,
  TouchableOpacity,
  View,
  Switch,
} from "react-native";

type Task = {
  id: string;
  name: string;
  scheduledTime: string;
  specificDate?: string;
  isActive: boolean;
  isRepeating: boolean;
  repeatDays?: string[];
};

export default function ListTasksScreen() {
  const [tasks, setTasks] = useState<Task[]>([]);
  const [loadingId, setLoadingId] = useState<string | null>(null);

  const fetchTasks = async () => {
    try {
      const token = await getToken();
      const response = await axios.get(
        `${process.env.EXPO_PUBLIC_BE_URL}/tasks`,
        {
          headers: { Authorization: `Bearer ${token}` },
        }
      );
```

Figure 8.6: Lists the tasks

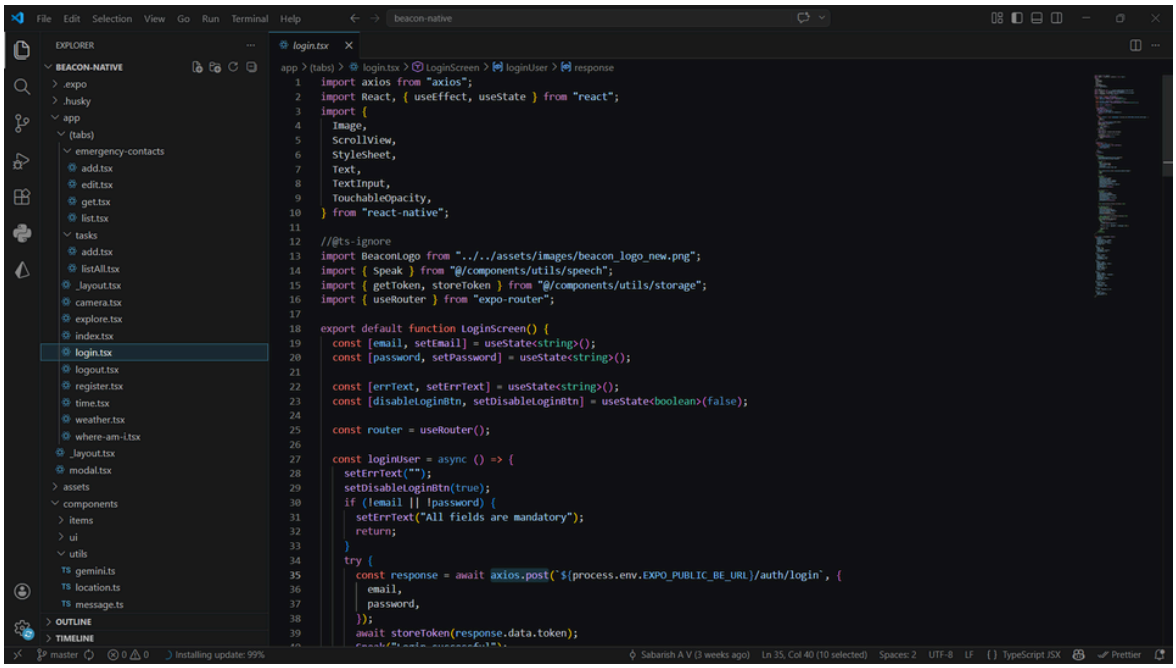


Figure 8.7: login.tsk

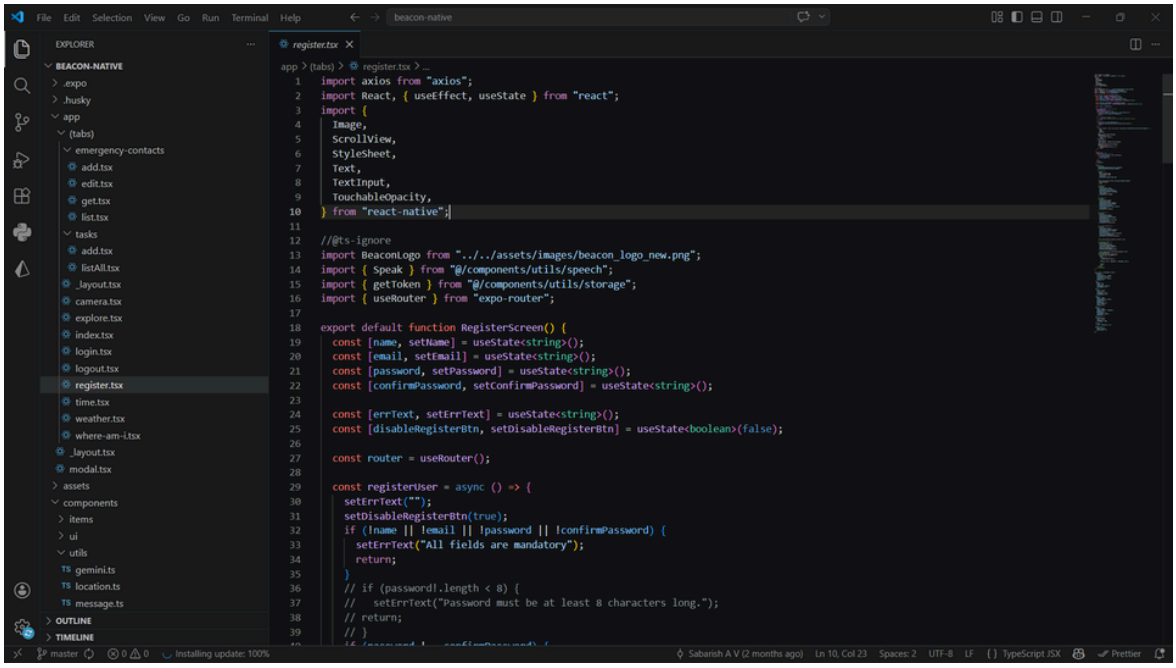


Figure 8.8: register.tsk

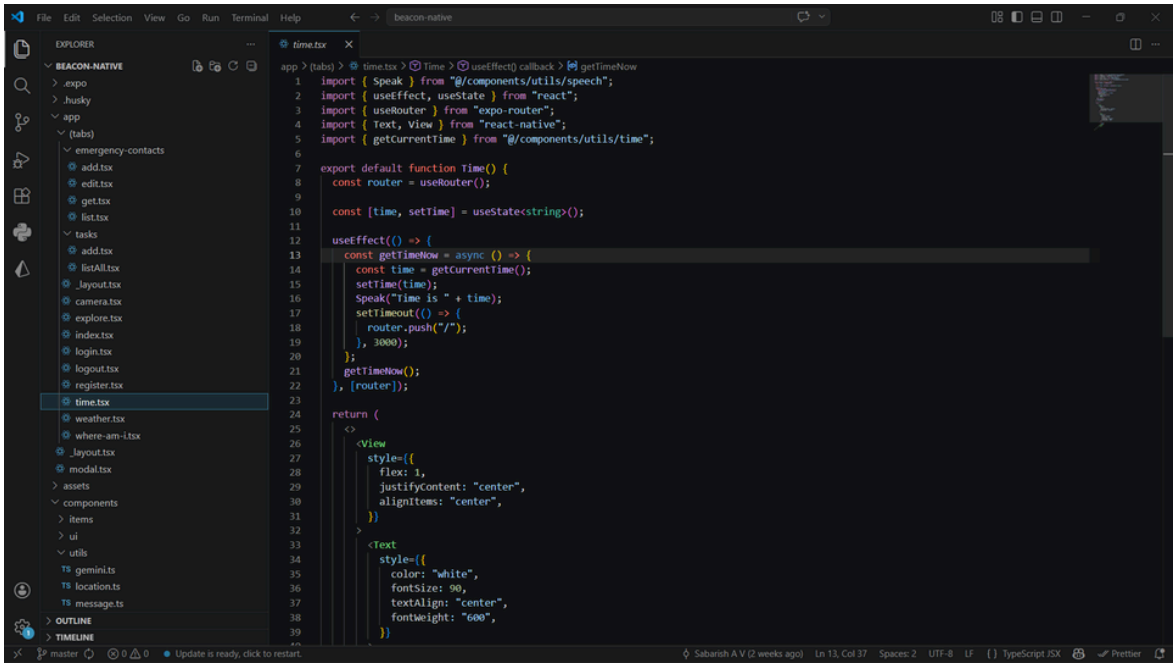


Figure 8.9: time.tsk

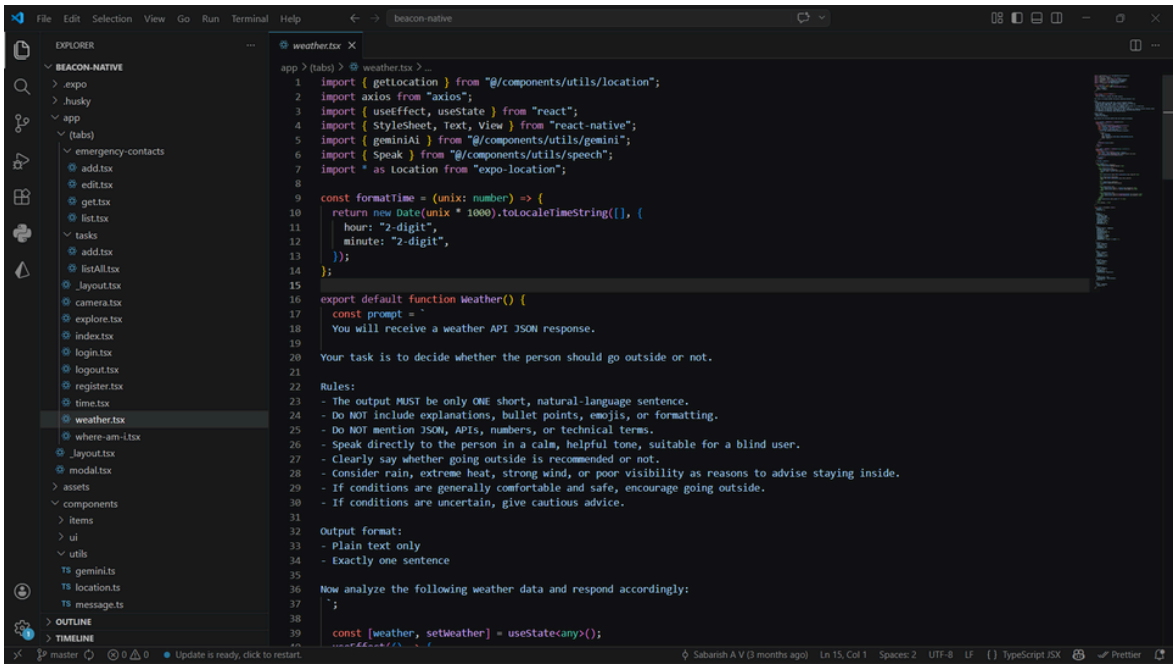


Figure 8.10: weather.tsk

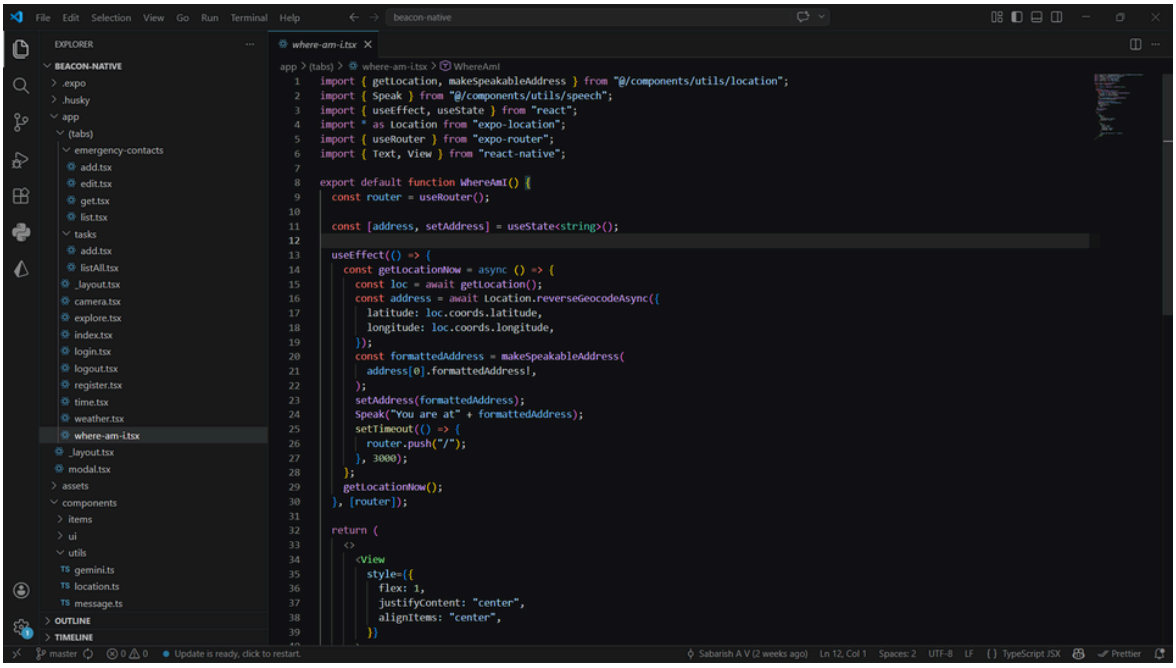


Figure 8.11: where-am-i.tsk

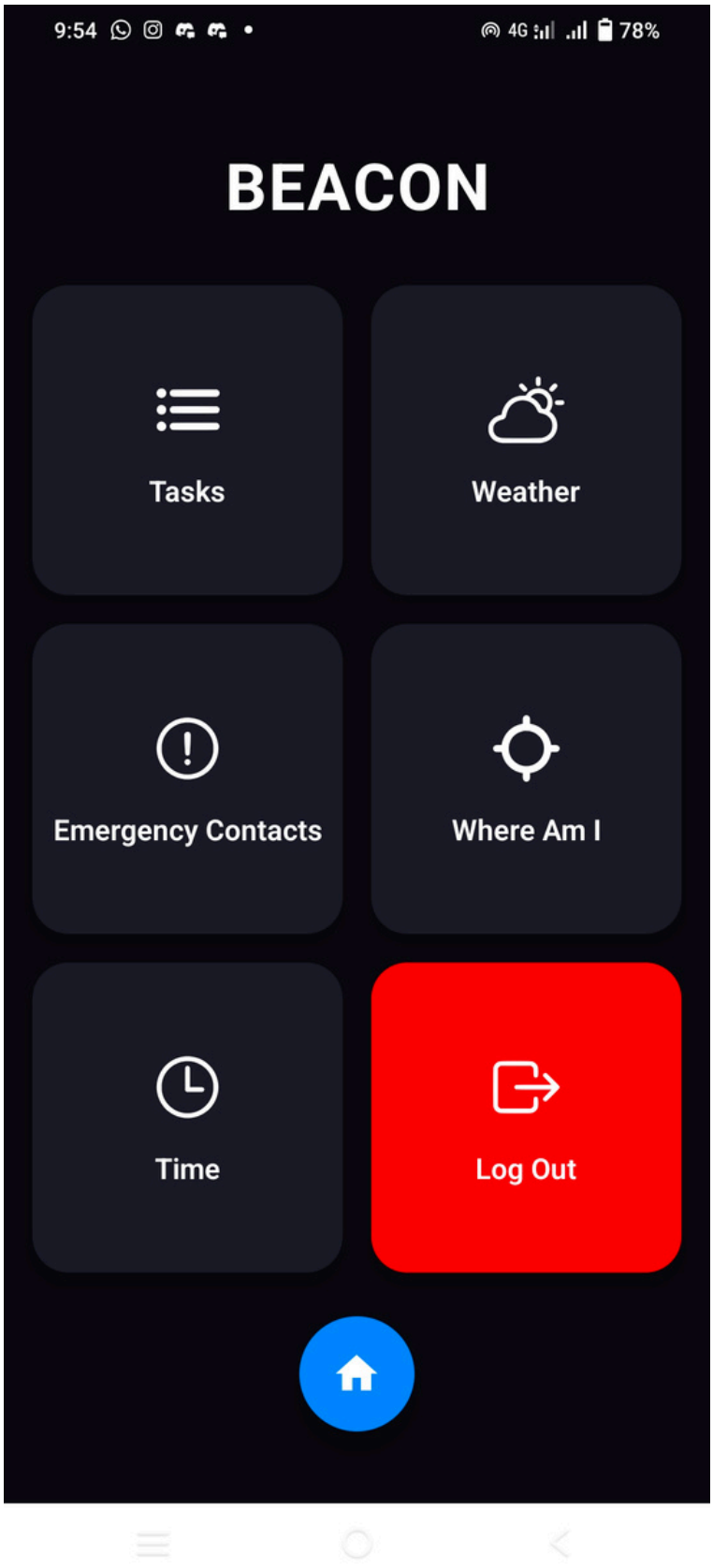


Figure 8.12: Home page

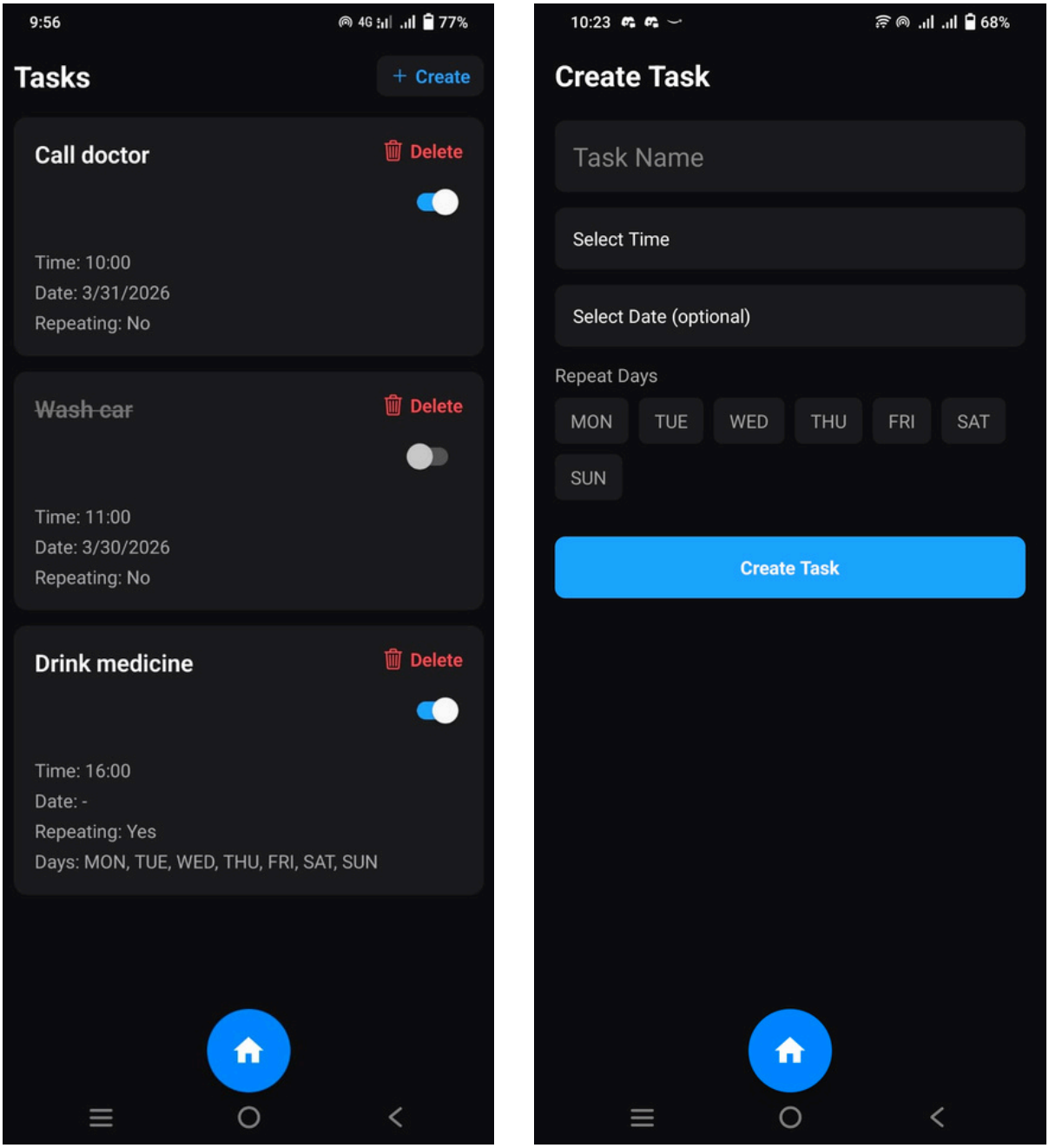


Figure 8.13: Tasks and Create Tasks pages

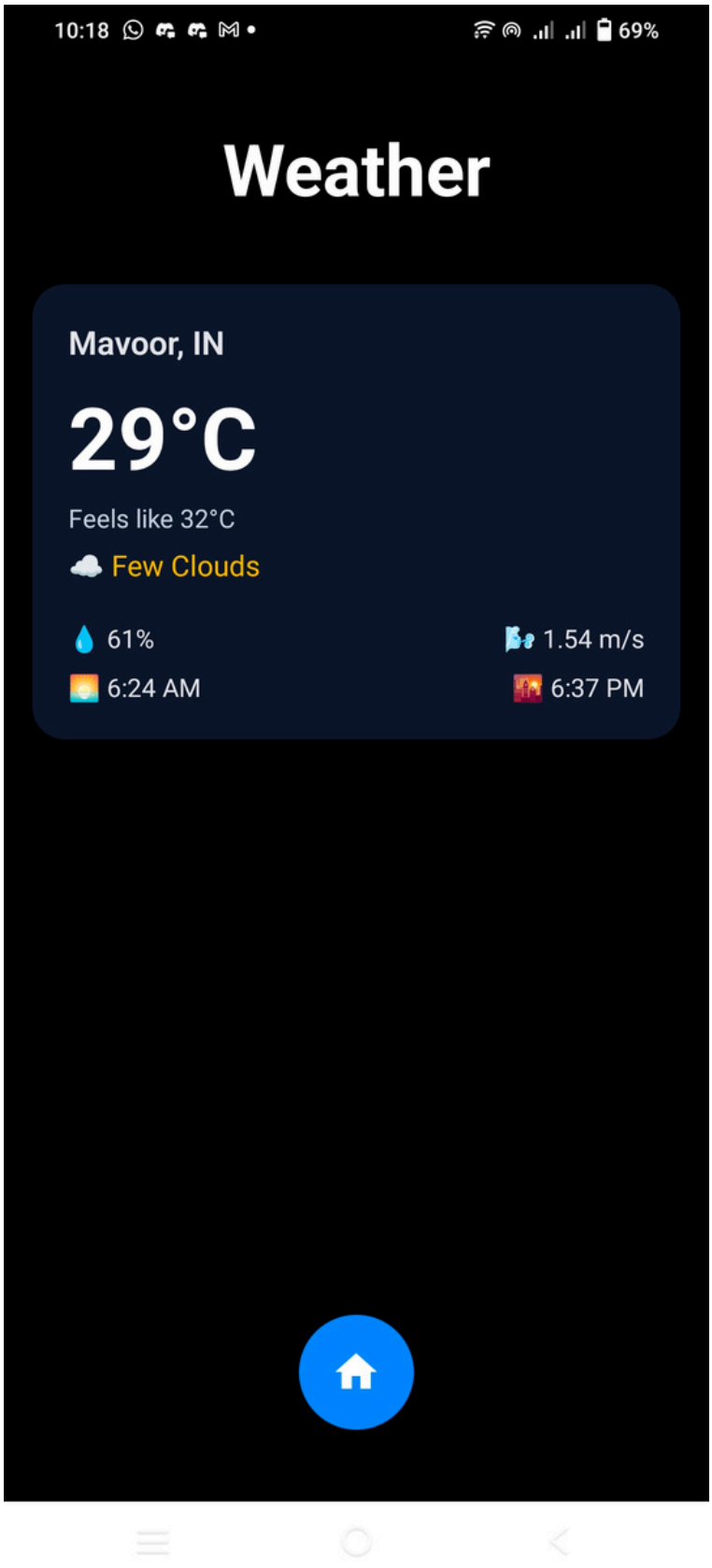


Figure 8.14: Weather page

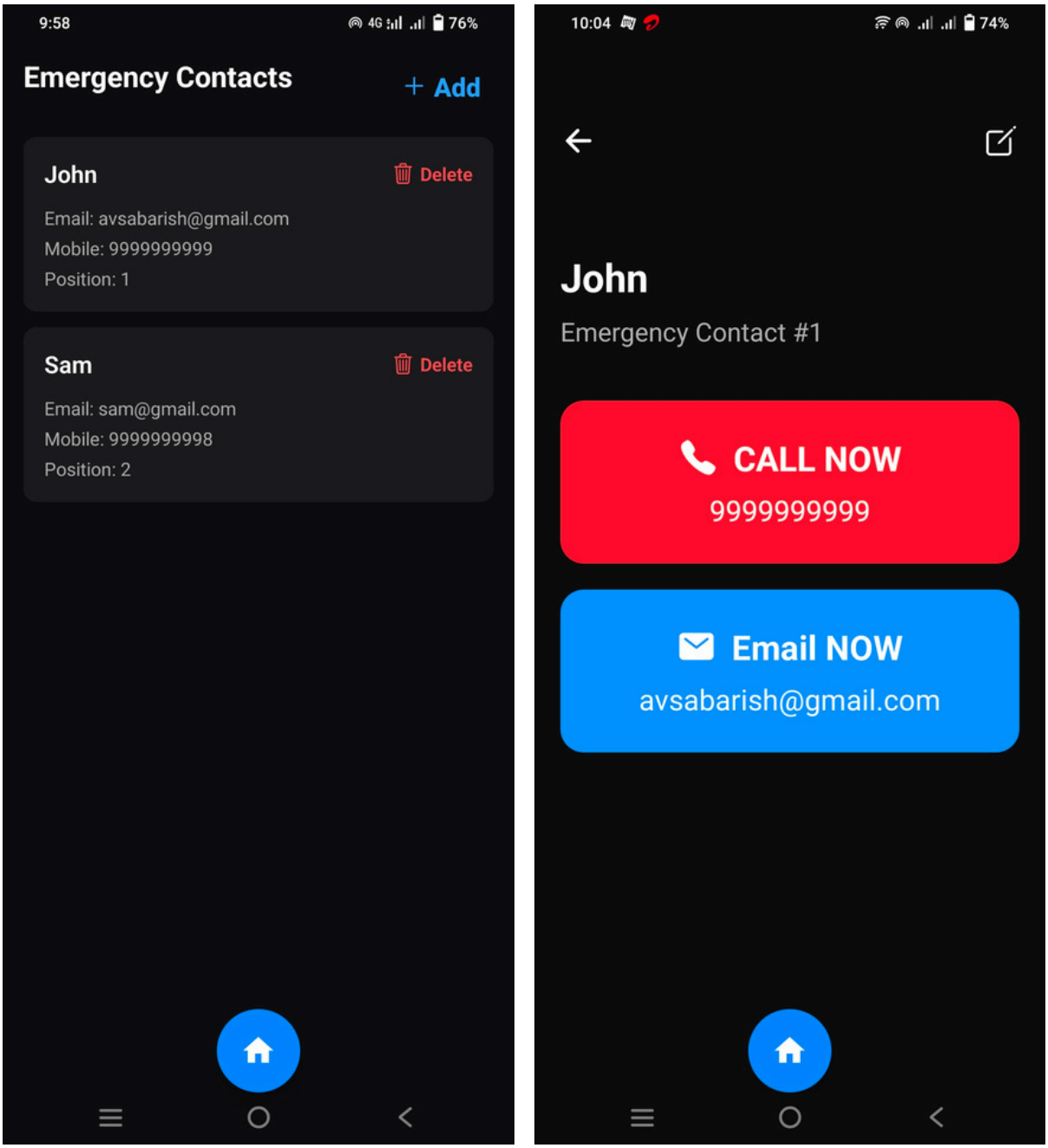


Figure 8.15: Emergency contact pages

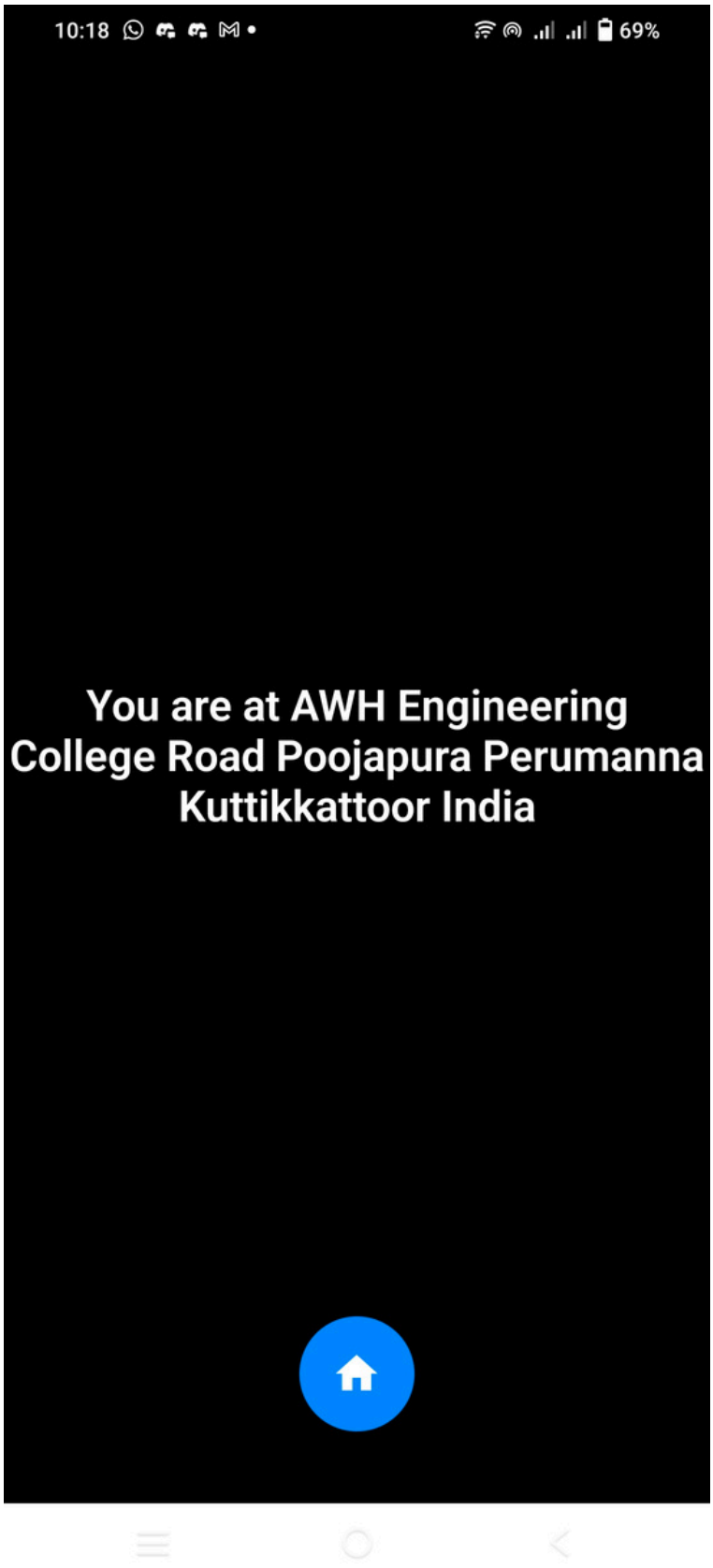


Figure 8.16: Where am i page

Figure 8.17: Time page

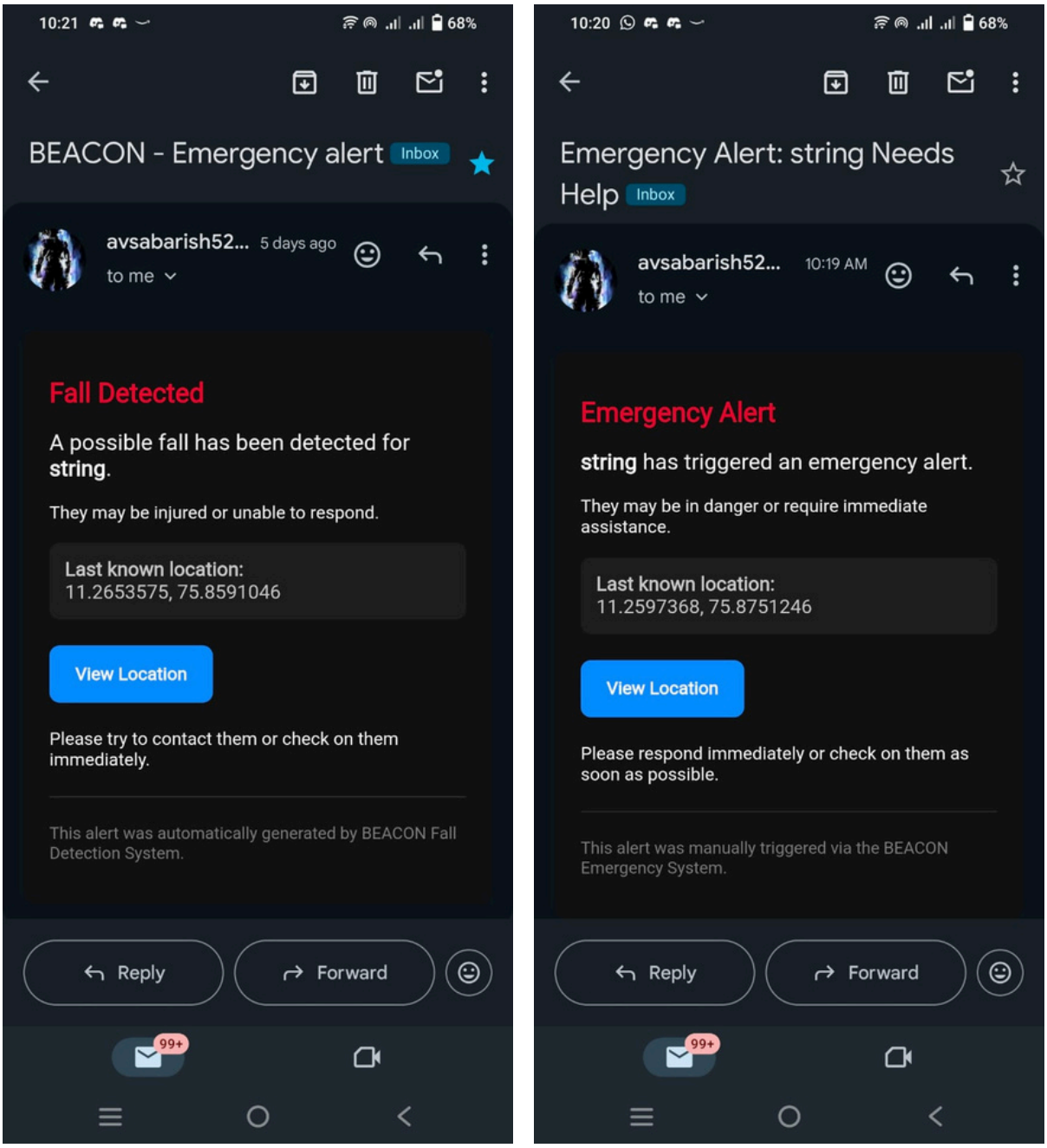


Figure 8.18: Emergency Alerts

CHAPTER 9
CONCLUSION

The B.E.A.C.O.N system presents an effective and innovative solution for assisting visually impaired individuals in performing their daily activities with greater ease and independence. By integrating artificial intelligence, wearable technology, and mobile application features, the system provides enhanced environmental awareness and real-time guidance, enabling users to navigate their surroundings more safely and confidently. The use of a smart cap equipped with a camera, combined with intelligent processing through AI models, allows the system to interpret real-world scenarios and respond appropriately.

The system successfully combines multiple functionalities such as obstacle detection, crosswalk recognition, currency identification, and emergency alert mechanisms into a single unified platform. This integration eliminates the need for multiple assistive tools and simplifies the user experience. The inclusion of additional features such as fall detection, manual alert triggering, weather updates, and reminder systems further enhances the practicality and usefulness of the application in real-life situations.

The implementation of edge-based processing ensures fast response times and reliable performance, as data is processed locally without dependency on cloud infrastructure. This significantly reduces latency and improves system efficiency, especially in environments with limited or no internet connectivity. The voice-based interface and simple interaction mechanisms make the system highly accessible and user-friendly, ensuring that visually impaired users can operate it with minimal difficulty.

Furthermore, the system demonstrates strong potential in improving mobility, safety, and independence for visually impaired individuals. It not only assists in navigation but also supports everyday tasks such as identifying currency and receiving important alerts, thereby increasing user confidence and reducing dependency on others. The modular and scalable design of the system also allows for future enhancements and integration of additional features.

In conclusion, the B.E.A.C.O.N system represents a significant step forward in the field of assistive technology. By combining real-time computer vision, intelligent processing, and accessible user interaction, the system offers a comprehensive and practical solution that can greatly enhance the quality of life for visually impaired individuals.

CHAPTER 9
REFERENCES

[1] Y. Yao, J. Li, and C. Zhang, "Safety Helmet Detection Based on Improved YOLOv8," IEEE Access, vol. 12, 2024, doi: 10.1109/ACCESS.2024.3368161.

[2] I. Campero-Jurado, S. Márquez-Sánchez, J. Quintanar-Gómez, S. Rodríguez, and J. M. Corchado, "Smart Helmet 5.0 for Industrial Internet of Things Using Artificial Intelligence," MDPI Sensors, vol. 20, no. 21, p. 6241, Nov. 2020, doi: 10.3390/s20216241.

[3] H. Assemlali et al., "Computer Vision-Based Detection and Classification of Road Obstacles: Systematic Literature Review," IEEE Access, 2025, doi: 10.1109/ACCESS.2025.11079598.

[4] S. Ikram et al., "Obstacle Detection and Warning System for Visually Impaired Using IoT Sensors," IEEE Access, vol. 13, pp. 35309–35321, 2025, doi: 10.1109/ACCESS.2025.3543299.

[5] P. Cheng and U. Roedig, "Personal Voice Assistant Security and Privacy—A Survey," Proceedings of the IEEE, vol. 110, no. 4, pp. 476–507, Apr. 2022, doi: 10.1109/JPROC.2022.3150789.

[6] A. Wang et al., "Safety Helmet Wearing Detection Model Based on Improved YOLO-M," IEEE Access, vol. 11, pp. 26247–26257, 2023, doi: 10.1109/ACCESS.2023.3257183.

[7] S. Q. Wahla and M. U. Ghani, "Visual Fall Detection from Activities of Daily Living for Assistive Living," IEEE Access, vol. 11, pp. 108876–108890, 2023, doi: 10.1109/ACCESS.2023.3321192.

[8] D. Swain, V. Rupapara, A. A. Nour, S. K. Satapathy, B. Acharya, S. Mishra, and A. Bostani, "A Deep Learning Framework for the Classification of Brazilian Coins," IEEE Access, vol. 11, pp. 109448–109461, 2023, doi: 10.1109/ACCESS.2023.3321428.

[9] A. Holzinger et al., "Human-Centered AI in Smart Farming: Toward Agriculture 5.0," IEEE Access, 2024, doi: 10.1109/ACCESS.2024.3395532.

[10] X. Zhao et al., "Omni-Directional Obstacle Detection for Vehicles Based on Depth Camera," IEEE Access, 2020, doi: 10.1109/ACCESS.2020.9091268.