**ÇANKAYA UNIVERSITY**

**Software Engineering Department**

**Computer Engineering Department**

**SENG 491-492**

**SOFTWARE DESIGN DOCUMENT**

**APPHASIA**

**Hikmetcan Altaş**

**Emre Buğrahan Yaşar**

**Mesude Gökpınar**

**Emin Barış Portakal**

[1. Introduction 1](#_Toc226825613)

[1.1. Purpose 1](#_Toc1228570944)

[1.2. Scope 1](#_Toc1631194500)

[1.3. Definitions, Acronyms and Abbreviations 2](#_Toc1379130767)

[1.4. References 2](#_Toc788473980)

[1.5. Overview 3](#_Toc2128147843)

[2. Architecture 4](#_Toc326072653)

[2.1. Clients 4](#_Toc596053891)

[2.1.1. Mobile Phones 4](#_Toc1737699222)

[2.1.2. Tablets 4](#_Toc2108427007)

[2.1.3. Personal Computers 4](#_Toc2095886454)

[2.2. Application Server 5](#_Toc1332550000)

[2.3. Database Server 5](#_Toc1726461309)

[3. User Interfaces 5](#_Toc592518074)

[3.1 Greetings Screen 6](#_Toc705356881)

[3.2 Create Profile 6](#_Toc406199982)

[3.2 Simple Main Page 7](#_Toc1921755192)

[3.3 Drawing Screen 9](#_Toc596811528)

[3.4 Patiente Routine Screen 9](#_Toc917015644)

[3.5 Routine Detail Screen 11](#_Toc218448503)

[3.6 Routine Detail Screen 2 12](#_Toc326000355)

[3.7 Routine Add/Delete 13](#_Toc1513572828)

[3.8 Routine Deleted Screen 14](#_Toc1072220195)

[3.9 Routine Added Screen 15](#_Toc2105914208)

[3.10 Send Notification to Caretaker 16](#_Toc2091889972)

[3.11 Notification Screen On Caretaker’s Screen 16](#_Toc1715171370)

[3.12 Patiente Login 17](#_Toc1831250851)

[3.13 Caretaker Settings 18](#_Toc1277588740)

[3.14 Show Patience for Caretaker 19](#_Toc1971885639)

[3.15 Adding Routine 20](#_Toc202258219)

[4. Architectural Design 21](#_Toc635487117)

[5. Database and Deployment 23](#_Toc10169498)

[5.1 Database 23](#_Toc134976781)

[5.2 Deployment 23](#_Toc1505465988)

# Introduction

## Purpose

The purpose of this Software Design Document (SDD) is to provide a comprehensive overview of the architectural and design decisions made for the Aphasia Communication Aid Application. This document serves as a detailed reference to guide the development team, ensuring that the functional and non-functional requirements are effectively addressed in the system design. It outlines the strategies and components used to create an accessible and efficient application tailored to the needs of individuals with aphasia, taking into account their physical and cognitive limitations. The SDD is intended for the development team to ensure accurate implementation, project stakeholders to validate the alignment of the design with project goals, the quality assurance team to facilitate effective testing, and future developers to support maintenance and further development.

## Scope

This Software Design Document (SDD) covers the following aspects:

* System Architecture: Describes the overall architecture of the application, including components such as clients, application server, and database server.
* User Interface Design: Details the design of screens and interface components.
* High-Level Design: Defines the core modules of the application and their interactions.
* Low-Level Design: Provides technical details and the internal workings of each module.
* Database Design: Includes the data model, E-R diagram, and the structure of database tables.
* References: Lists the resources and materials used in the development and design process.

This document does not include specific implementation details such as coding practices, third-party libraries, or deployment instructions.

## Definitions, Acronyms and Abbreviations

This section provides definitions of key terms, acronyms, and abbreviations used in the Software Design Document (SDD) to ensure clarity and proper interpretation.

* SDD: Software Design Document - A document outlining the design and architecture of a software system.
* UI: User Interface - The visual and interactive components through which users interact with the application.
* DB: Database - A structured collection of data stored and managed electronically.
* E-R Diagram: Entity-Relationship Diagram - A graphical representation of entities and their relationships in a database.
* API: Application Programming Interface - A set of protocols and tools for building and interacting with software applications.
* Offline Mode: A feature allowing the application to function without an active internet connection.
* Aphasia: A language disorder affecting a person's ability to communicate, often caused by brain injury.

Further terms and acronyms may be detailed in an appendix or referenced from external documentation as necessary.

## References

This section lists all documents and resources referenced in this Software Design Document (SDD). It provides the necessary details for identifying and obtaining these references.

1. IEEE Standard 1016-2009
   * Title: IEEE Standard for Information Technology - Systems Design - Software Design Descriptions
   * Date: June 25, 2009
   * Publishing Organization: IEEE
   * Source: https://ieeexplore.ieee.org/Xplore/home.jsp
2. Aphasia Research Report
   * Title: Communication Challenges and Solutions for Individuals with Aphasia
   * Report Number: N/A
   * Date: 2023
   * Publishing Organization: Aphasia Institute
   * Source: https://www.aphasia.ca
3. Database Design Principles
   * Title: Database System Concepts (7th Edition)
   * Report Number: N/A
   * Date: 2020
   * Publishing Organization: McGraw-Hill Education
   * Source: https://www.mheducation.com
4. UI/UX Design Guidelines
   * Title: Material Design Guidelines
   * Report Number: N/A
   * Date: 2024
   * Publishing Organization: Google

## Overview

This Software Design Document (SDD) provides a comprehensive description of the architectural and design elements of the Aphasia Communication Aid Application. It serves as a detailed guide for developers, stakeholders, and quality assurance teams, ensuring alignment on system implementation and functionality. The SDD is organized to facilitate ease of understanding and navigation.

The document is structured as follows:

* Introduction: Defines the purpose, scope, intended audience, and references used in the document.
* Architecture: Describes the high-level system architecture, including components such as clients, application servers, and database servers.
* User Interface Design: Details the design and layout of application screens, focusing on usability and accessibility.
* High-Level Design: Provides an overview of the core modules and their interactions within the application.
* Low-Level Design: Explains the technical specifics of each module, including detailed workflows and interactions.
* Database Design: Includes the entity-relationship diagram and a detailed description of database tables.
* References: Lists all documents and resources referenced throughout the SDD for further reading and validation.

The document is organized into clearly defined sections to ensure logical flow and readability, enabling the audience to easily locate specific details and understand the system's design comprehensively.

# Architecture

# Clients

# Mobile Phones

Minimum requirements:

* Operating System (OS): Android 9.0 (Pie) or higher, iOS 13 or higher
* Memory (RAM): At least 2 GB
* Storage: At least 100 MB of free space
* Screen: 5 inches or larger, 720x1280 resolution
* Internet Connection: Not required (optimized for offline use)
* Additional Hardware: Touchscreen support

# Tablets

Minimum requirements:

* Operating System (OS): Android 9.0 (Pie) or higher, iPadOS 13 or higher
* Memory (RAM): At least 3 GB
* Storage: At least 150 MB of free space
* Screen: 7 inches or larger, 1280x800 resolution
* Internet Connection: Not required (optimized for offline use)

# Application Server

Minimum requirements:

* Operating System (OS): Ubuntu 20.04 LTS, CentOS 8, or Windows Server 2019
* Memory (RAM): At least 8 GB
* Storage: At least 50 GB SSD
* Processor: Dual-core 2.5 GHz or higher
* Network: 1 Gbps network connection
* Additional Software: Node.js (v18 or higher), Nginx or Apache

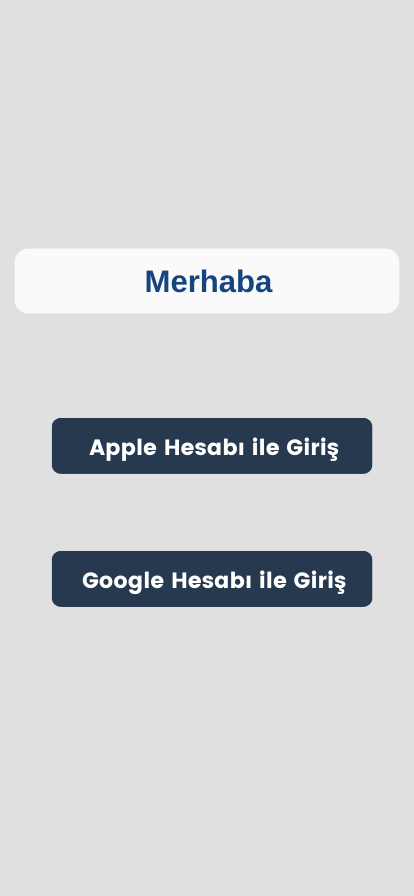
# Database Server

Minimum requirements:

* Operating System (OS): Ubuntu 20.04 LTS, CentOS 8, or Windows Server 2019
* Memory (RAM): At least 16 GB
* Storage: At least 100 GB SSD (scalable)
* Processor: Quad-core 3.0 GHz or higher
* Network: 1 Gbps network connection
* Additional Software: PostgreSQL 14 or higher, or MySQL 8.0 as an alternative

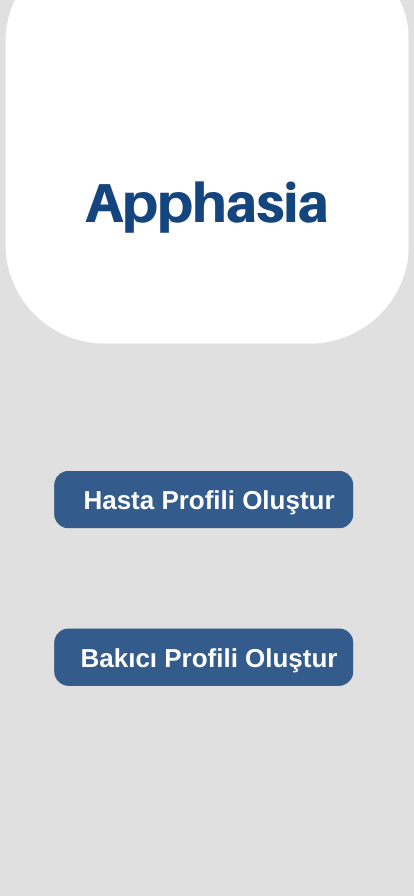
# 3. User Interfaces

## 3.1 Greetings Screen



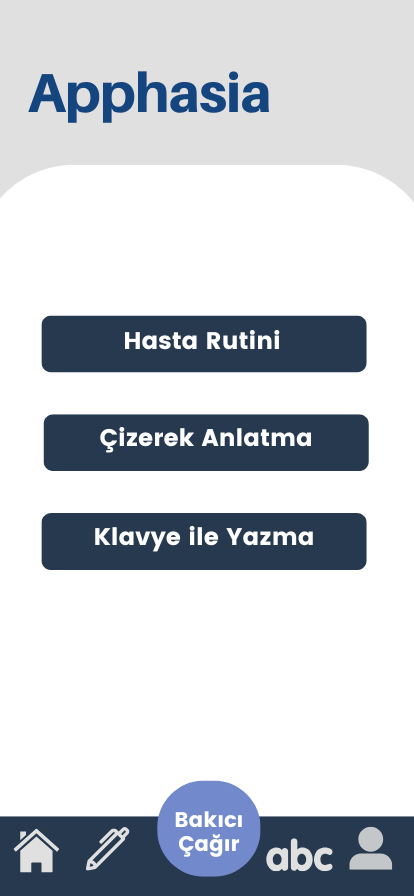
|  |  |
| --- | --- |
| **Greeting Text** | Displays "Merhaba" as a welcoming text at the top of the screen. |
| **Apple Login Button** | Button labeled "Apple Hesabı ile Giriş" to log in using an Apple account. |
| **Google Login Button** | Button labeled "Google Hesabı ile Giriş" to log in using a Google account. |

## 3.2 Create Profile



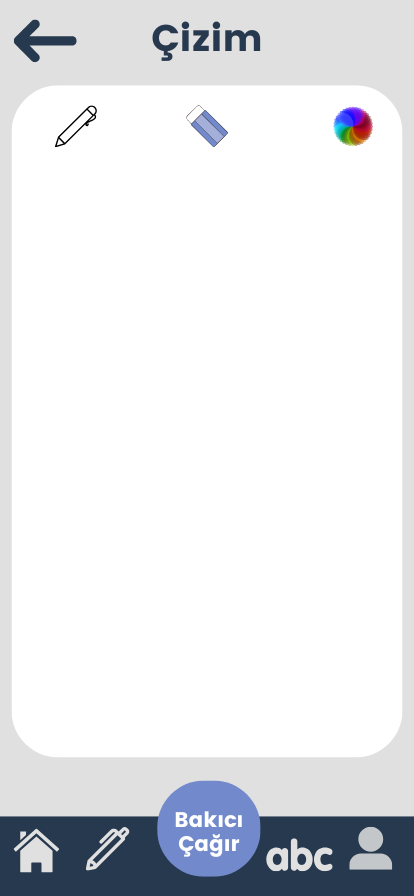
|  |  |
| --- | --- |
| **Application Title** | Displays "Apphasia" at the top of the screen. |
| **Patient Profile Button** | Button labeled "Hasta Profili Oluştur" to create a new patient profile. |
| **Caretaker Profile Button** | Button labeled "Bakıcı Profili Oluştur" to create a new caretaker profile. |

## 3.3 Simple Main Page



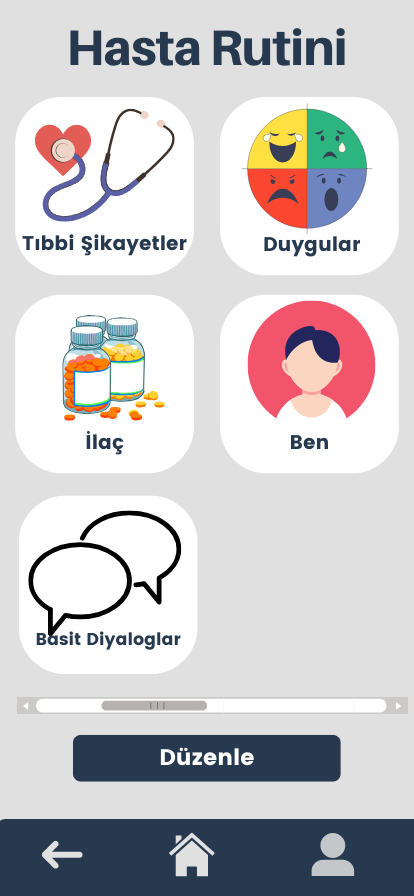
|  |  |
| --- | --- |
| Patient Routine Button | Allows the user to access the patient’s daily routine. |
| Drawing Mode Button | Enables the user to express themselves through drawing. |
| Typing Mode Button | Allows the user to type text using the keyboard for communication. |
| **Home Button** | Icon representing the home screen navigation. |
| **Write Button** | Icon for accessing text or note-based tools. |
| **Call Caretaker Button** | Central circular button labeled "Bakıcı Çağır" to call the caretaker. |
| ABC Icon | Switches to the typing mode. |

## 3.4 Drawing Screen



|  |  |
| --- | --- |
| **Screen Title** | Displays "Çizim" at the top of the screen. |
| **Pen Icon** | Allows the user to draw on the screen. |
| **Eraser Icon** | Allows the user to erase parts of the drawing. |
| **Color Picker Icon** | Enables the user to select different colors for drawing. |
| **Drawing Canvas** | A blank area where the user can create drawings. |
| ABC Icon | Switches to the typing mode. |
| Home Button | Icon representing the home screen navigation |

## 3.5 Patiente Routine Screen



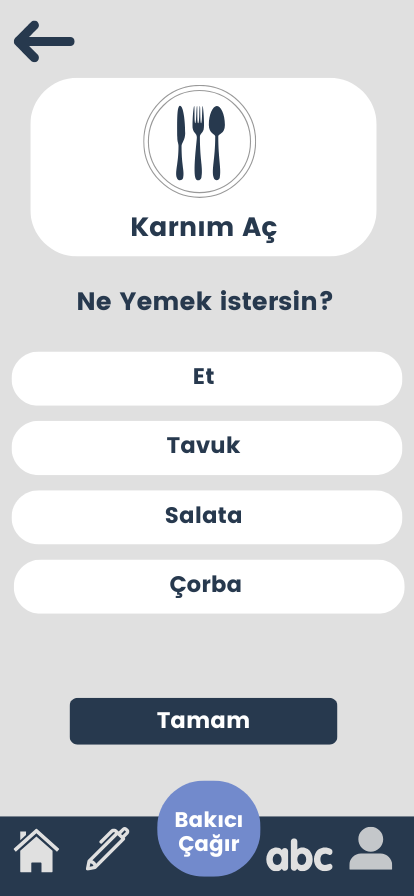
|  |  |
| --- | --- |
| **Icons and Buttons** | Provides options for navigating through patient-related features: |
| **Tıbbi Şikayetler** | Button with an icon of a stethoscope to log or view medical complaints. |
| **Duygular** | Button with an emotion chart icon to express or select feelings. |
| **İlaç** | Button with a medication icon to manage and track medicines. |
| **Ben** | Button with a personal avatar icon to access self-related routines. |
| **Basit Diyaloglar** | Button with speech bubbles icon to navigate to simple dialogues for communication. |
| **Scroll Bar** | Horizontal scroll bar at the bottom for accessing additional categories if needed. |

## 3.6 Routine Detail Screen



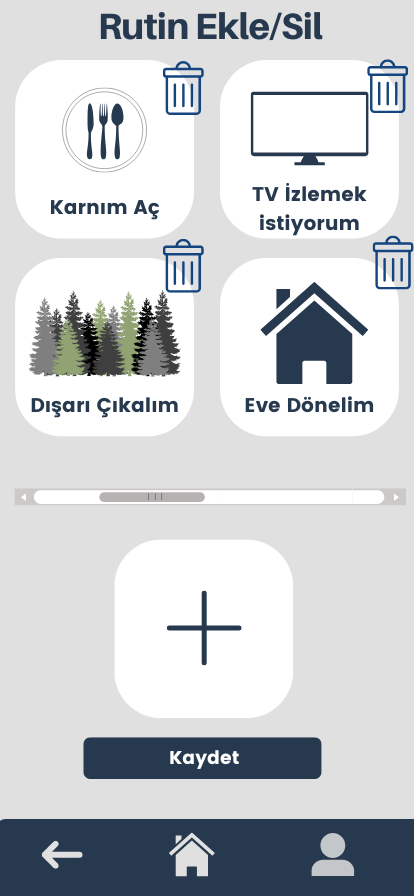
|  |  |
| --- | --- |
| **Option Buttons** | Provides three options for the user to select: |
| **Deniz Kenarı** | Button for choosing "Beach." |
| **Şehir Merkezi** | Button for choosing "City Center." |
| **Orman / Doğa** | Button for choosing "Forest / Nature." |
| **Confirm Button** | Button labeled "Tamam" to confirm the user's selection. |
| **Back Button** | An arrow icon at the top-left corner to navigate back to the previous screen. |
| **ABC Icon** | Switches to the typing mode. |
| **Home Button** | Icon representing the home screen navigation |

## 3.7 Routine Detail Screen 2



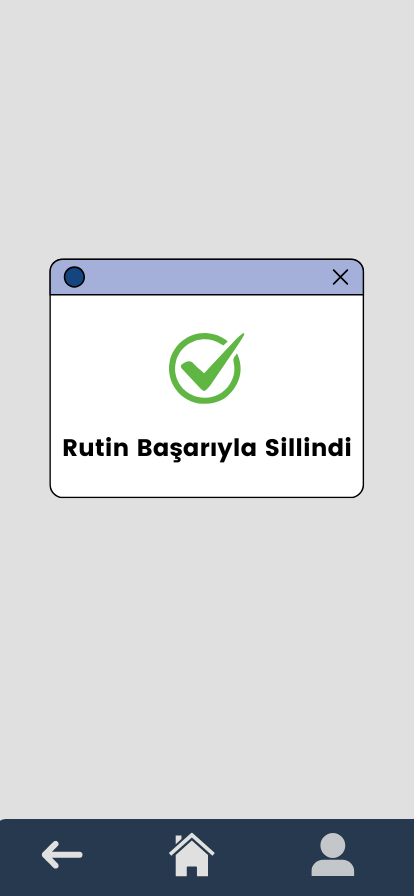
|  |  |
| --- | --- |
| **Question Text** | Displays the question "Ne Yemek İstersin?" to ask the user what they would like to eat. |
| **Option Buttons** | Provides four options for the user to select: |
| - **Et (etc.)** | Button for choosing "Meat." |
| **Confirm Button** | Button labeled "Tamam" to confirm the user's selection. |
| **Back**  **Button** | An arrow icon at the top-left corner to navigate back to the previous screen. |
| **ABC Icon** | Switches to the typing mode. |

## 3.8 Routine Add/Delete



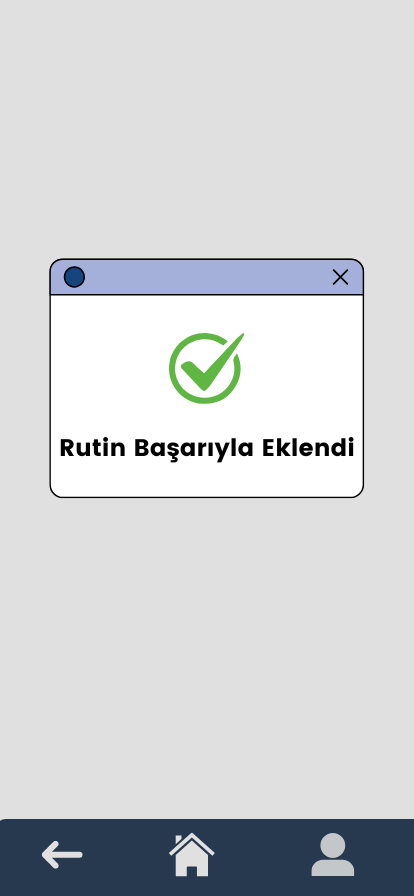
|  |  |
| --- | --- |
| **Add Routine Button** | A large "+" icon to add a new routine to the list. |
| **Save Button** | Button labeled "Kaydet" to save the changes made to routines. |
| **Trash Icons** | Allows the user to delete individual routines by clicking on the trash icon next to them. |
| Routine Cards | Displays a grid of existing routines with icons and labels. |
| Home Button | Icon representing the home screen navigation. |
| Back Button | An arrow icon at the top-left corner to navigate back to the previous screen. |

## 3.9 Routine Deleted Screen



|  |  |
| --- | --- |
| **Popup Window** | Displays a confirmation message in a modal-style window. |
| **Message** | Displays "Rutin Başarıyla Silindi" confirming the successful deletion of a routine. |
| **Icon** | Green checkmark icon to indicate success. |
| **Close Button** | An "X" button at the top-right corner to close the popup. |
| **Back Button** | An arrow icon at the top-left corner to navigate back to the previous screen. |
| **Home Button** | Icon representing the home screen navigation. |
| **Profile Icon** | To go to profile page. |

## 3.10 Routine Added Screen



|  |  |
| --- | --- |
| **Popup Window** | Displays a confirmation message in a modal-style window. |
| **Message** | Displays "Rutin Başarıyla Eklendi" confirming the successful addition of a routine. |
| **Icon** | Green checkmark icon to indicate success. |
| **Close Button** | An "X" button at the top-right corner to close the popup. |
| **Back Button** | An arrow icon at the top-left corner to navigate back to the previous screen. |
| **Home Button** | Icon representing the home screen navigation. |

## 3.11 Send Notification to Caretaker



|  |  |
| --- | --- |
| **Popup Window** | Displays a confirmation message in a modal-style window. |
| **Message** | Displays "Bakıcıya bildirim gönderildi" confirming that a notification has been sent to the caretaker. |
| **Close Button** | An "X" button at the top-right corner to close the popup. |
| **Home Button** | Icon representing the home screen navigation. |
| **ABC Icon** | Switches to the typing mode. |
| **Pen Icon** | To go to drawing page. |

## 3.12 Notification Screen On Caretaker’s Screen



|  |  |
| --- | --- |
| **Popup Window** | Displays an emergency notification in a modal-style window. |
| **Message** | Displays "Hastanızın Size İhtiyacı Var!" indicating that the patient needs help. |
| **Icon** | A red flashing emergency icon to emphasize the urgency of the situation. |
| **Close Button** | An "X" button at the top-right corner to close the popup. |
| **Home Button** | Icon representing the home screen navigation. |
| **Back Button** | An arrow icon at the top-left corner to navigate back to the previous screen. |

## 3.13 Patiente User Page



|  |  |
| --- | --- |
| **Profile Picture** | Displays a placeholder profile image with the option to "Fotoğraf Ekle" (Add Photo). |
| **Greeting Text** | Displays "Merhaba X," where "X" is the patient's name. |
| **Patient Email** | Shows the patient's email address, e.g., "[hasta@adres.com](mailto:hasta@adres.com)". |
| **Caretaker Info Section** | Displays a label "Bakıcı Bilgileri" to indicate caretaker details. |
| **Caretaker Email** | Shows the caretaker's email address, e.g., "[bakıcı@adres.com](mailto:bakıcı@adres.com)". |
| **ABC Icon** | Switches to the typing mode. |

## 3.14 Caretaker Settings



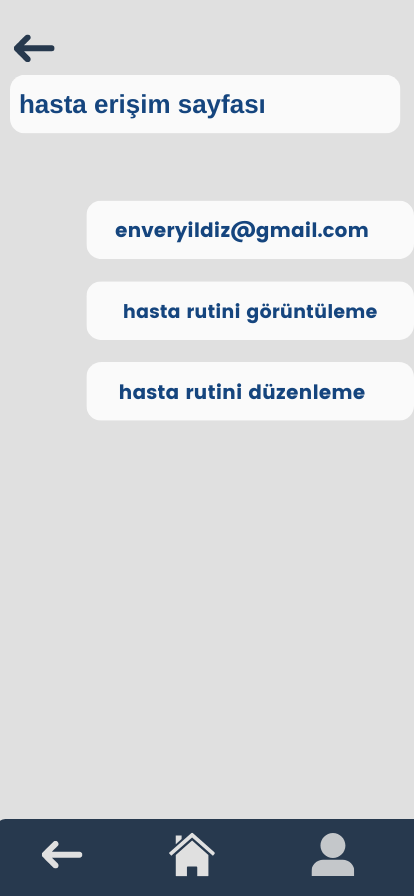
|  |  |
| --- | --- |
| **Profile Picture** | Displays a placeholder profile image with the option to "Fotoğraf Ekle" (Add Photo). |
| **Greeting Text** | Displays "Merhaba X," where "X" is the caretaker's name. |
| **Email Display** | Shows the caretaker's email address, e.g., "[bakıcı@adres.com](mailto:bakıcı@adres.com)". |
| **Patient Info Button** | Button labeled "hasta bilgilerim" to navigate to patient information. |
| **Bottom Navigation Bar** | Contains the buttons to go other pages. |
| **Home Button** | Icon representing the home screen navigation. |
| **Back Button** | An arrow icon at the top-left corner to navigate back to the previous screen. |

## 3.15 Keyboard Screen



|  |  |
| --- | --- |
| **Back Arrow Icon** | Allows the user to navigate back to the previous screen. |
| **Text Input Field** | Allows the user to type messages or phrases for communication. |
| **Send Icon (Arrow)** | Sends the typed message to be displayed or processed further. |
| **Suggested Words Bar** | Provides predictive text suggestions based on what the user is typing. |
| **Keyboard** | Standard virtual keyboard for typing messages. |
| **Home Icon** | Redirects the user to the home screen. |
| **Pen Icon** | Switches to the drawing mode. |
| ABC Icon | Indicates the user is currently in typing mode. |

## 3.16 Show Patience for Caretaker



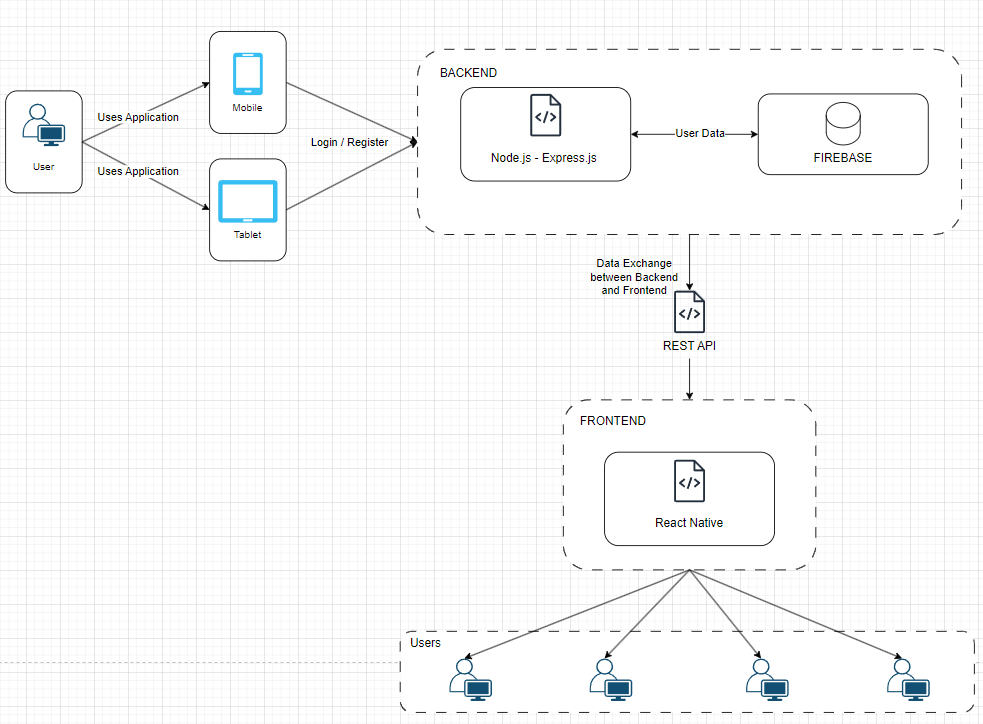
|  |  |
| --- | --- |
| **Screen Title** | Displays "hasta erişim sayfası" at the top of the screen. |
| **Email Display Button** | Button labeled "hasta mail adresi" to show the patient's email address. |
| **View Routine Button** | Button labeled "hasta rutini görüntüleme" to view the patient's routine. |
| **Edit Routine Button** | Button labeled "hasta rutini düzenleme" to edit the patient's routine. |
| **Back Button** | Icon representing the home screen navigation. |
| **Home Button** | Icon representing the home screen navigation. |
| **Profile Icon** | To go to profile button. |

## 3.17 Adding Routine



|  |  |
| --- | --- |
| **Image Display** | Displays an image at the top, representing the request theme. |
| **Question Text** | Displays the question "Ne istersin?" to ask the user to make a selection. |
| **Option Buttons** | Provides two options for the user: |
| - **Öneri-1** | Button labeled "Öneri-1" as the first suggestion or option. |
| Home Button | Icon representing the home screen navigation. |
| **Save Button** | Button labeled "Kaydet" to confirm and save the user's selection. |
| **Back Button** | Icon representing the home screen navigation. |

# 4. Architectural Design



1. User Interaction:
   1. The user interacts with the application via mobile or tablet devices.
   2. The application provides the ability to log in or register for accessing features.
2. Backend (Server-Side):
   1. The backend is built using Node.js and Express.js to handle server-side operations.
   2. User data, such as authentication and profile information, is stored and managed in Firebase.
   3. The backend ensures secure and efficient communication between the user and the database.
3. REST API:
   1. A REST API acts as the intermediary for data exchange between the backend and the frontend.
   2. This ensures that the application can send and receive data reliably and efficiently.
4. Frontend (Client-Side):
   1. The frontend is developed using React Native, providing a smooth and responsive user interface for mobile and tablet users.
   2. It consumes the REST API to fetch and send data to the backend.
5. Users:
   1. The final users of the application interact with the frontend on their devices to utilize the app's features like communication, data access, or customization.

# 5. Database and Deployment

### **5.1 Database**

For our application, we have chosen **Firebase Realtime Database** as our primary database solution. Firebase offers a cloud-hosted NoSQL database that is highly suitable for mobile applications requiring real-time data synchronization. Its ability to provide seamless data updates across devices ensures that users' custom communication templates and preferences are always accessible, regardless of the device they use.

Firebase Realtime Database supports offline capabilities, allowing users to interact with the application even without an active internet connection. Data is stored locally on the device and synchronized with the cloud once the connection is re-established, ensuring continuity and reliability.

Additionally, Firebase provides robust security features, such as user-based authentication and access rules, to protect sensitive user data. Its scalability and global infrastructure also ensure that our application can efficiently handle increasing user demand and data volume as the project grows.

By leveraging Firebase, we ensure a simplified yet powerful database solution that aligns with the accessibility and functionality requirements of our target users.

### 5.2 Deployment

To host and manage our backend services, we chose **Firebase** for its simplicity and integration with mobile platforms. Firebase provides a robust infrastructure, enabling seamless hosting, database management, and analytics within a single platform.

For mobile app deployment, the application is distributed through:

* **Google Play Store** (for Android users).
* **Apple App Store** (for iOS users).

The deployment process includes:

* **Version Control:** Managed via **Git** to ensure traceability and collaboration.
* **Continuous Integration (CI):** Automated builds and tests using **GitHub Actions**, ensuring high-quality releases.
* **Monitoring and Analytics:** Leveraging Firebase Analytics to track user engagement and identify potential issues in real-time.

This deployment strategy ensures a smooth and scalable delivery of our application, meeting the needs of our target users while allowing future updates to be implemented seamlessly.