TED ÜNİVERSİTESİ

High-Level Design

GATE

(Guard and Access Technology for Estates)

Alperen Göyce 39028411222

Berkay Demirbilek 39203133116

Furkan Cerrahoğlu 70246051814

1. Introduction

1.1 Purpose of the System

The purpose of the GATE (Protection and Access to Estate Technology) system is to address management and security issues and challenges faced by residents and managers of large-scale residential complexes in particular. It aims to increase security, streamline management processes and improve communication within these communities. It uses advanced technologies such as AI-powered facial recognition and automated reporting. GATE provides a unified platform accessible via mobile and web interfaces. The system is designed to increase trust, simplify administrative tasks and provide a safe, user-friendly living environment.

1.2 Design Goals

- **Scalability :** The system is designed to handle increasing numbers of users and residential complexes without compromising performance or responsiveness.
  Usability : The platform provides user-friendly interfaces for mobile and web applications, ensuring comfortable and convenient use for users across different roles within the site.
- **Security :** Robust data protection measures such as encryption, secure access controls, and regulatory compliance such as KVKK are implemented to protect sensitive information.
- **Reliability :** The system is built for high availability and fault tolerance, ensuring consistent functionality and minimal downtime even during high usage or unexpected scenarios.
- **Integration :** Seamless integration with existing infrastructure such as security cameras, access control systems, and payment gateways increases the platform's operational efficiency.
- **Performance :** Critical operations such as notifications, payments, and security alerts are optimized to ensure fast processing and response times.
- **Adaptability :** The modular architecture allows for future updates, additions, and incorporation of emerging technologies without disrupting the existing system.
- **Accessibility :** The platform works efficiently across devices and browsers, ensuring cross-platform compatibility.
- **Sustainability :** The hardware and software components to be used in the system are state-of-the-art, environmentally friendly, and components that require little maintenance or constant replacement.
- **Ethical Design :** The system is designed to ensure transparency, obtain user approval, and provide security practices, eliminating privacy and ethical concerns.

1.3 Definitions, Acronyms, and Abbreviations

- **GATE**: Guard and Access Technology for Estates.
- **KVKK**: Turkey's data protection regulation.

1.4 Overview

The GATE (Protection and Site Access Technology) project is a comprehensive management and security platform designed to address the challenges and problems faced by large-scale residential complexes. It offers a unified solution that integrates advanced technologies such as AI-powered facial recognition, anomaly detection and automatic reporting to enhance security and streamline administrative processes.

The platform enables accessibility via mobile application and web interface, allowing residents, administrators and security personnel to interact seamlessly. Key features include a notification system for announcements, payment tracking and reminders, parking management and real-time security alerts. These features aim to improve communication, ensure security and simplify the management of shared spaces.

Designed with scalability, usability and sustainability in mind, GATE prioritizes data security and compliance with legal regulations such as KVKK, as well as user-friendly interfaces. Its modular architecture allows for future developments and makes it adaptable to emerging technologies and broader applications. GATE brings a new solution for residential management and security by ensuring trust, efficiency and efficiency.

2. Current Software Architecture

This section provides an overview of the current software architecture, highlighting its components, technologies, capabilities, and limitations.

2.1 Overview of the Existing System

- **General Functionality:** The current system provides essential functionalities such as:
  - Allowing residents to manage maintenance payments, receive notifications, and view community announcements.
  - Enabling security personnel to access camera feeds and monitor entries and exits.
- **Scope:** The system primarily serves residents, managers, and security personnel, addressing their unique needs within the estate management context.

2.2 Technology Stack

- **Frontend:**
    - Web Application: Built using React for an intuitive and responsive user interface.
    - Mobile Application: Developed using Flutter for cross-platform compatibility on both iOS and Android devices.
- **Backend:**
    - Built with C# and .NET to handle business logic, database operations, and integrations.
- **AI:**
    - AI models are developed using TensorFlow for features like facial recognition and anomaly detection.
- **Storage and Data Management:**
    - PostgreSQL is used as the relational database for structured data storage.
    - Cloud-based storage solutions manage large files such as video archives.

2.3 Current System Capabilities

- **User Management:**
    - Secure login and authentication for residents, managers, and security personnel.
    - Role-based access control to ensure data security and appropriate permissions.
- **Notifications and Communication:**
    - Community-wide announcements and individual notifications for important updates.
    - Messaging capabilities between residents, managers, and security staff.
- **Security:**
    - AI-powered facial recognition for monitoring authorized entries.
    - Anomaly detection to identify and alert on unusual activities.
- **Payment Processing:**
    - Online payment gateway integration for maintenance fees and dues.
    - Automated reminders for upcoming or overdue payments.

2.4 System Limitations

- **Performance Issues:**
    - The system experiences slowdowns during peak traffic.
- **Lack of Modularity:**
    - Components are tightly coupled, making it challenging to add new features.
- **Scalability Problems:**
    - The current architecture struggles to accommodate increasing users or new residential complexes.

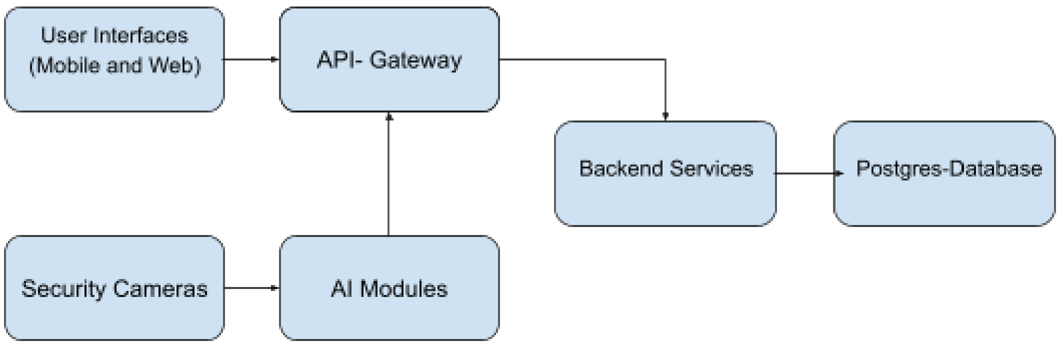
2.5 Current Data Flow

The data flow within the system can be summarized as follows:

1. A user logs into the mobile application and initiates a payment.
2. The payment request is sent to the backend via APIs.
3. The backend processes the request and updates the PostgreSQL database.
4. The confirmation of the payment is sent back to the mobile application.

2.6 Current System Diagram

A simplified diagram illustrates the interaction between the system components:



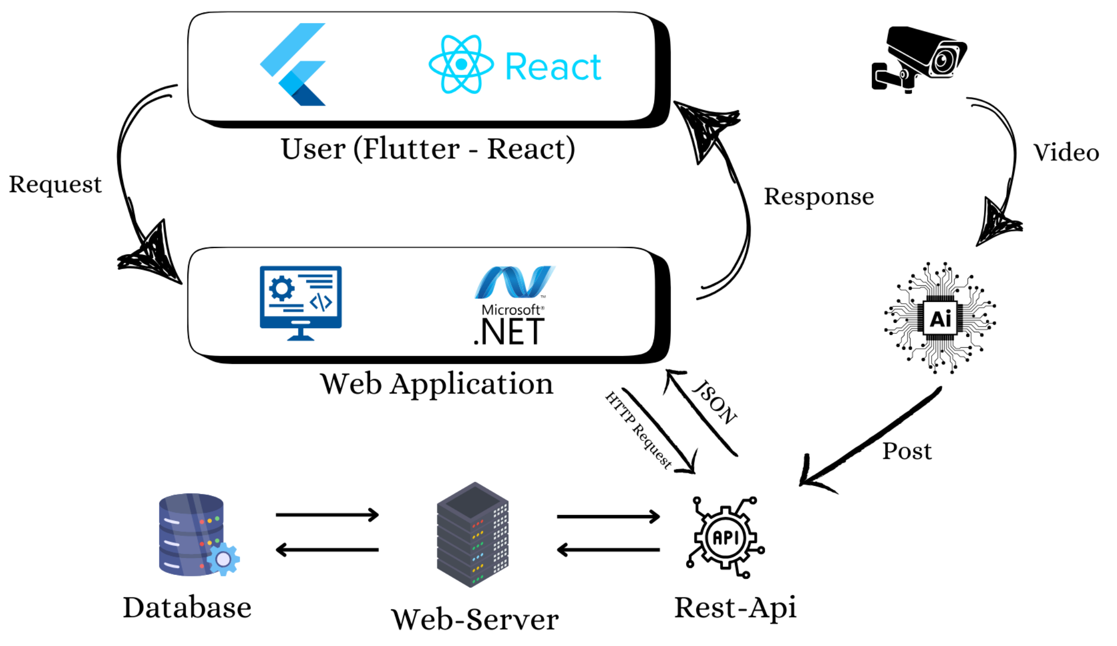
3. Proposed Software Architecture

3.1 Overview

The GATE platform integrates multiple technologies to deliver a comprehensive solution for estate management and security. The architecture is designed to ensure modularity, scalability, and performance. The key components are:

- **Mobile Application (Flutter):** Enables residents and managers to interact with the platform via a responsive and intuitive interface, ensuring accessibility across iOS and Android devices.
- **Web Application (React):** Provides a browser-based interface for residents, managers, and administrators, designed for ease of use and efficient task execution.
- **Backend Services (C# with .NET):** Acts as the backbone of the platform, managing data storage, business logic, and integration with external systems such as AI tools and payment gateways.
- **AI Models (Python Frameworks):** Implements advanced features like facial recognition and anomaly detection to enhance security. TensorFlow is used for training and deployment of these AI models. OpenCV is also integrated for image processing on facial recognition processes. In addition to that, CUDA can accelerate

processes such as facial recognition by supporting GPU usage with the parallel computing technology it provides.

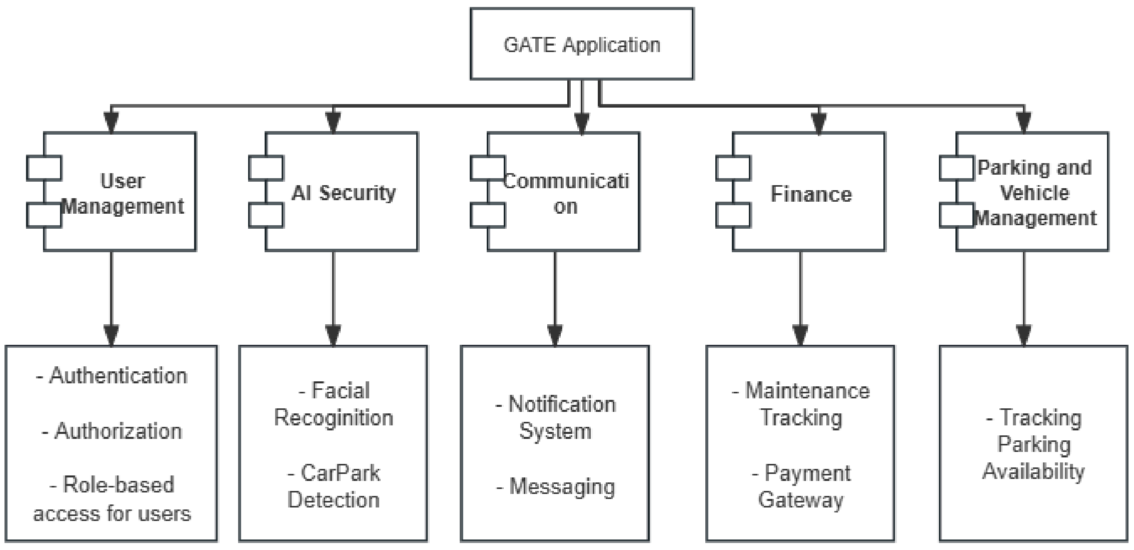


3.2 Subsystem Decomposition

The GATE system is divided into the following subsystems, each addressing specific functional areas:

1. **User Management:**
    - Provides role-based access control for residents, managers, security personnel, and administrators.
    - Includes authentication (secure login) and authorization mechanisms to protect sensitive data and actions.
2. **AI Security:**
    - **Facial Recognition:** Tracks and identifies individuals entering the estate, comparing them with a pre-approved list of residents and guests.
    - **Anomaly Detection:** Identifies unusual patterns in surveillance feeds, triggering real-time alerts for security personnel.
3. **Communication:**
    - **Notification System:** Delivers community-wide announcements, payment reminders, and emergency alerts.
    - **Direct Messaging:** Facilitates one-on-one communication between residents and managers or security personnel.

4. **Finance:**
    - Tracks maintenance fees, dues, and other financial obligations of residents.
    - Provides an online payment gateway and automated reminders to ensure timely payments.
5. **Parking and Vehicle Management:**
    - Tracks vehicle entry and exit using Artificial Intelligence.
    - Displays real-time parking availability and manages guest vehicle registrations.



3.3 Hardware/Software Mapping

The GATE system integrates a variety of hardware and software components to ensure a reliable and efficient platform. Below is a mapping of the key components:

Mobile Application

- **Software:** Developed using Flutter for cross-platform compatibility, enabling deployment on both Android and iOS devices.
- **Hardware:** Accessible on user devices, including smartphones and tablets with a minimum hardware requirement of 2 GB RAM and Android 8.0/iOS 12.0 or higher.

Web Application

- **Software:** Built using React for a responsive and intuitive user interface, optimized for both desktop and mobile browsers.
- **Hardware:** Designed to run on devices with a minimum of 4 GB RAM and a modern web browser (e.g., Chrome, Firefox, Edge).

Backend Services

- **Software:** Implemented using C# with .NET for efficient handling of business logic, integrations, and database interactions.
- **Hardware:** Hosted on cloud infrastructure, leveraging virtual machines or containerized environments for scalability and fault tolerance.

AI Subsystems

- **Software:** Powered by TensorFlow for facial recognition and anomaly detection.
- **Hardware:** Requires integration with surveillance cameras and local processing units (e.g., GPUs or AI accelerators) for real-time video analysis.

Security Infrastructure

- **Software:** Employs firewalls, authentication protocols, and secure API gateways.
- **Hardware:** Includes dedicated cloud-based servers with hardware security modules (HSMs) for encryption and secure data storage.

3.4 Persistent Data Management

The GATE platform's data management strategy focuses on maintaining data integrity, security, and accessibility. Below are the details:

Database

- **Type:** Relational database, such as PostgreSQL, for structured data storage.
- **Structure:** Optimized schema design includes tables for users, roles, notifications, financial transactions, and system logs.
- **Scalability:** Supports horizontal scaling to accommodate increasing user data and transaction volumes.

Security Measures

- **Data Encryption:** All sensitive data, such as user credentials and financial records, is encrypted both in transit and at rest using encryption.
- **Anonymization:** Personal data is anonymized wherever feasible to ensure privacy compliance.
- **Regular Backups:** Automatic daily backups are performed, stored securely in geographically redundant cloud locations.

Data Access

- **Access Control:** Implements role-based access to ensure only authorized users can view or manipulate sensitive data.
- **Audit Logging:** Maintains detailed logs of data access and modifications for accountability and troubleshooting.

Integration

- **AI Models:** Persistent storage for training data and AI model outputs, integrated seamlessly with the database for facial recognition and anomaly detection.
- **Cloud Storage:** Leverages cloud-based object storage solutions for large data files, such as surveillance video archives and backup snapshots.

3.5 Access Control and Security

The GATE system prioritizes robust access control and security measures to safeguard user data and ensure the integrity of platform operations. Below is a detailed description of the access control and security mechanisms:

Role-Based Permissions

The platform utilizes role-based access control (RBAC) to assign permissions based on user roles. Each role has a distinct set of capabilities tailored to its responsibilities:

1. **Residents:**
    - View personal account details, including payment history and notifications.
    - Submit maintenance requests, complaints, and emergency alerts.
    - Reserve shared facilities and track approvals.
2. **Managers:**
    - Monitor resident payments and generate financial reports.
    - Broadcast announcements and update community rules.
    - Manage maintenance requests and coordinate with security personnel.
3. **Admins:**
    - Oversee user management, including adding or removing users and assigning roles.
    - Configure system settings, such as notification preferences and access permissions.
    - Manage backups, updates, and overall platform maintenance.
4. **Security Personnel:**
    - Monitor camera feeds and respond to emergency alerts.
    - Log entry/exit details and report unauthorized access.
    - Track and address parking violations.

Authentication and Authorization

1. **Multi-Factor Authentication (MFA):**
    - Sensitive operations, such as making payments or accessing admin settings, require MFA for additional security.
    - Supported methods include: OTPs sent via email.

2. **Session Management:**
    - Secure session tokens are used to maintain authentication across the system.
    - Sessions expire automatically after a set period of inactivity to prevent unauthorized access.
3. **Authorization Policies:**
    - Permissions are enforced at both the application and API levels to ensure data and functionality are only accessible to authorized users.

Data Protection Measures

1. **Encryption:**
    - Data in transit is protected using Transport Layer Security.
    - Data at rest is encrypted using encryption standards, ensuring compliance with regulations like KVKK.
2. **Audit Logs:**
    - Detailed logs track all user actions and system changes, enabling administrators to detect unauthorized or suspicious activities.
    - Logs are stored securely and accessible only to authorized personnel.
3. **Firewall and Intrusion Detection:**
    - Cloud-based firewalls block unauthorized access attempts.
    - Intrusion detection systems monitor for anomalies or potential breaches.

Recovery and Redundancy

1. **Backup Strategy:**
    - Incremental backups are performed daily, and full backups are conducted weekly.
    - Backups are encrypted and stored in multiple geographic locations for disaster recovery.
2. **Failover Systems:**
    - Redundant servers ensure continuous system availability in the event of hardware failure or outages.

3.6 Global Software Control

The GATE platform employs a comprehensive software control strategy to ensure reliable operation, efficient handling of tasks, and seamless user experiences. The primary components of this strategy include:

Event-Driven Architecture

- **Purpose:** The platform adopts an event-driven architecture to manage real-time notifications and alerts efficiently.

- **Key Features:**
  - **Event Producers:** Subsystems like the notification system, AI security module, and financial management module act as event producers, generating events based on user interactions or system triggers.
  - **Event Handlers:** Dedicated handlers process these events, ensuring timely delivery of notifications, updates, or alerts.
  - **Example Use Case:** An anomaly detected by the AI triggers a security alert that is immediately sent to relevant personnel and logged in the system.

Centralized Logging and Monitoring

- **Purpose:** Ensures system-wide visibility and enables quick identification and resolution of issues.
- **Key Features:**
  - **Centralized Logging:** All subsystems, including mobile and web applications, backend services, and AI modules, log their activities to a central system.
  - **Real-Time Monitoring:** Tools like Prometheus or AWS CloudWatch monitor the health and performance of system components.
  - **Error Alerts:** Automated alerts notify administrators of critical errors or potential breaches.
  - **Audit Trails:** Logs are stored securely and maintain a comprehensive history of system operations for accountability and troubleshooting.

3.7 Boundary Conditions

The GATE platform is designed to handle boundary conditions effectively, ensuring robust performance under various scenarios.

Failure Handling

- **Fallback Mechanisms for AI Failures:**
    - **Fallback Mode:** The system defaults to manual validation by security personnel.
    - **Alert System:** Automatic notifications are sent to administrators for immediate investigation.
    - **Redundancy:** Pre-trained AI models stored in the cloud can be redeployed to minimize downtime.
    - **Grace Periods:** Residents are granted additional time for payment deadlines to accommodate technical issues.

Scalability

- **Modular Design for Additional Housing Complexes:**
    - The platform is built with a microservices architecture, enabling easy scaling and addition of new functionalities.
    - Each housing complex operates as a separate tenant in the system, with isolated data and configurations.
    - Infrastructure scaling strategies:
        - Horizontal scaling: Adding more instances of services as user load increases.
        - Vertical scaling: Upgrading server resources for intensive processes like AI anomaly detection.
    - **Example Use Case:** When a new housing complex adopts GATE, administrators can easily configure the system for the new location without impacting existing operations.

4. Subsystem Services

The GATE platform consists of several subsystem services, and each designed to meet the specific requirements of the users. These services ensure seamless operation, enhanced user experience, and robust security mechanisms. Below are the detailed services categorized by user roles:

4.1 Resident Services

Residents are the core user group of the GATE platform. The services designed for them focus on convenience, accessibility, and transparency.

- **Payment Management:**
    - View pending dues, payment history, and upcoming deadlines.
    - Process payments securely through integrated payment gateways.
    - Receive automated reminders and confirmations via notifications.
- **Notification Viewing:**
    - Access community-wide announcements such as upcoming events, maintenance schedules, or rule updates.
    - Receive personalized notifications, including payment reminders or package deliveries.
- **Facility Reservations:**
    - Book shared facilities like meeting rooms, sports facilities, or parking spots.
    - Track approval status and manage reservation changes.

- **Complaint and Maintenance Requests:**
    - Submit complaints or maintenance requests directly through the mobile application.
    - Track the status of submitted requests and communicate with assigned personnel.

4.2 Manager Services

Managers oversee the operational and financial aspects of the estate. Their services are designed to facilitate efficient task execution and community management.

- **Rule Setting and Updates:**
    - Configure and update community guidelines and policies.
    - Publish rule changes and notifications to all residents.
- **Communication:**
    - Broadcast announcements to specific groups or the entire community.
    - Directly message residents or security personnel to address issues.
- **Financial Oversight:**
    - Monitor maintenance fees, dues, and other financial transactions.
    - Generate detailed financial reports for audits and reviews.
- **Camera Monitoring:**
    - Access live feeds and recordings from surveillance cameras.
    - Coordinate with security personnel in case of identified anomalies.
- **Task Assignment:**
    - Assign and monitor maintenance tasks or complaint resolutions.
    - Track task progress and completion timelines.

4.3 Security Services

Security personnel are responsible for ensuring the safety and security of the estate. Their services integrate advanced AI tools for enhanced situational awareness.

- **Visitor Tracking:**
    - Log and verify visitor entries using AI-powered facial recognition or manual validation.
    - Issue and monitor temporary visitor passes for controlled access.
- **Anomaly Detection:**
    - Identify unusual activities through AI-driven surveillance analysis.
    - Respond to alerts generated by anomaly detection systems in real-time.
- **Parking Management:**
    - Monitor and track vehicles parking availability for parking regulations and resolve violations.
- **Incident Reporting:**
    - Document security incidents and generate detailed reports for managers.
    - Maintain logs of resolved incidents for future reference and analysis.

4.4 Integration with Other Subsystems

The subsystem services integrate seamlessly with other components of the GATE platform to provide a cohesive experience:

- **AI Integration:**
    - Facial recognition data is used for visitor tracking and resident authentication.
    - Anomaly detection integrates with notification systems to alert managers and security personnel.
- **Payment Gateway Integration:**
    - Financial transactions are securely processed and logged in the backend services.
    - Payment statuses are updated in real-time and accessible to both residents and managers.
- **Notification System:**
    - All subsystems feed data into the centralized notification system to ensure timely alerts and updates.

Each subsystem is designed with scalability and customization in mind; subsystems operate independently, allowing for easy upgrades or replacements.Additional features can be added without disrupting existing services.Each service is accessible based on user roles to maintain data security and streamline functionality.

By detailing the services provided by each subsystem, the GATE platform ensures a comprehensive and robust design that addresses all functional and nonfunctional requirements.

5. Glossary

- **Resident:** A user of the GATE platform who lives within the residential complex.
- **Manager:** A user responsible for overseeing estate operations, financials, and resident communications.
- **Security Personnel:** Users tasked with monitoring and ensuring the safety of the estate.
- **Role-Based Access Control (RBAC):** A security mechanism that grants permissions based on user roles.
- **Facial Recognition:** AI-powered technology used to identify individuals entering the estate.
- **KVKK:** Turkey's Data Protection Regulation, ensuring compliance with data security standards.
- **API Gateway:** A server that acts as an entry point for client requests, routing them to appropriate backend services.
- **Backend Services:** The core system managing data processing, logic execution, and integrations.
- **PostgreSQL:** A relational database management system used for structured data storage.
- **TensorFlow:** An open-source machine learning framework used to develop AI models for the platform.
- **Cloud Storage:** Remote storage of large files such as video archives and system backups, ensuring data accessibility and redundancy.
- **Mobile Application:** A software application designed for residents and managers to interact with the GATE platform via smartphones and tablets.
- **Web Application:** A browser-based platform providing access to GATE features for residents, managers, and security personnel.
- **Notifications:** Alerts or messages sent to users about important updates, reminders, or emergencies.
- **Payment Gateway:** A service that processes online payments for maintenance fees, dues, and other financial obligations.
- **Surveillance Cameras:** Hardware devices used to monitor and record activity within the estate.
- **Scalability:** The system's ability to handle increased load, users, or additional residential complexes without performance degradation.
- **Firewall:** A security system that monitors and controls incoming and outgoing network traffic based on predetermined rules.
- **Encryption:** The process of converting data into a secure format to prevent unauthorized access.
- **Real-Time Processing:** The ability to process data and provide responses instantaneously as events occur.

6. References

- Netsec Cloud. (n.d.). *How to create an effective high-level design document: A step-by-step guide*. Retrieved December 25, 2024, from https://netseccloud.com/how-to-create-an-effective-high-level-design-document-a-step-by-step-guide
- Vedmkw. (n.d.). *How to create a good high-level design (HLD)*. Medium. Retrieved December 25, 2024, from https://medium.com/@vedmkw/how-to-create-a-good-high-level-design-hld-fddba7f6ae18
- Vancouvering, D. (n.d.). *Writing a high-level design*. Medium. Retrieved December 25, 2024, from https://david-vancouvering.medium.com/writing-a-high-level-design-26280ee88480
- Ergun, O. (n.d.). *How to document your high-level design: A step-by-step guide*. Retrieved December 25, 2024, from https://orhanergun.net/how-to-document-your-high-level-design-a-step-by-step-guide
- KVKK. (n.d.). *Personal Data Protection Law No. 6698*. Retrieved December 25, 2024, from https://www.mevzuat.gov.tr/mevzuat?MevzuatNo=6698&MevzuatTur=1&MevzuatTertip=5
- Implementation of robust face recognition system using ... (n.d.-a). https://arxiv.org/pdf/1811.07339