ALEP: Advanced Loan Eligibility Predictor

# System Design Document

Version 1.0

2024-02-27

Table of Contents

[1. Introduction 1](#_Toc160112726)

[**1.1** Purpose of the SDD 1](#_Toc160112727)

[2. General Overview and Design Guidelines/Approach 2](#_Toc160112728)

[**2.1** General Overview 2](#_Toc160112729)

[**2.2** Assumptions/Constraints/Risks 2](#_Toc160112730)

[2.2.1 Assumptions 2](#_Toc160112731)

[2.2.2 Constraints 3](#_Toc160112732)

[2.2.3 Risks 3](#_Toc160112733)

[**2.3** Alignment with an Enterprise Architecture framework 3](#_Toc160112734)

[3. Design Considerations 5](#_Toc160112735)

[**3.1** Goals and Guidelines 5](#_Toc160112736)

[**3.2** Development Methods & Contingencies 5](#_Toc160112737)

[**3.3** Architectural Strategies 5](#_Toc160112738)

[**3.4** Performance Engineering 6](#_Toc160112739)

[4. System Architecture and Architecture Design 8](#_Toc160112740)

[**4.1** Logical View 8](#_Toc160112741)

[**4.2** Hardware Architecture 8](#_Toc160112742)

[4.2.1 Security Hardware Architecture 8](#_Toc160112743)

[4.2.2 Performance Hardware Architecture 8](#_Toc160112744)

[**4.3** Software Architecture 8](#_Toc160112745)

[4.3.1 Security Software Architecture 9](#_Toc160112746)

[4.3.2 Performance Software Architecture 9](#_Toc160112747)

[**4.4** Information Architecture 9](#_Toc160112748)

[4.4.1 Records Management 9](#_Toc160112749)

[**4.5** Internal Communications Architecture 10](#_Toc160112750)

[**4.6** Security Architecture 10](#_Toc160112751)

[**4.7** Performance 10](#_Toc160112752)

[**4.8** System Architecture Diagram 11](#_Toc160112753)

[5. System Design 12](#_Toc160112754)

[**5.1** Business Requirements 12](#_Toc160112755)

[**5.2** Database Design 12](#_Toc160112756)

[5.2.1 Data Objects and Resultant Data Structures 12](#_Toc160112757)

[5.2.2 File and Database Structures 12](#_Toc160112758)

[**5.3** Data Conversion 14](#_Toc160112759)

[**5.4** User Machine-Readable Interface 14](#_Toc160112760)

[5.4.1 Inputs 14](#_Toc160112761)

[5.4.2 Outputs 15](#_Toc160112762)

[**5.5** User Interface Design 16](#_Toc160112763)

[5.5.1 Section 508 Compliance 16](#_Toc160112764)

[6. Operational Scenarios 17](#_Toc160112765)

[7. Detailed Design 19](#_Toc160112766)

[**7.1** Hardware Detailed Design 19](#_Toc160112767)

[**7.2** Software Detailed Design 19](#_Toc160112768)

[**7.3** Security Detailed Design 20](#_Toc160112769)

[**7.4** Performance Detailed Design 20](#_Toc160112770)

[**7.5** Internal Communications Detailed Design 20](#_Toc160112771)

[8. System Integrity Controls 21](#_Toc160112772)

[9. External Interfaces 22](#_Toc160112773)

[**9.1** Interface Architecture 22](#_Toc160112774)

[**9.2** Interface Detailed Design 22](#_Toc160112775)

[Appendix A: Record of Changes 23](#_Toc160112776)

[Appendix B: Acronyms 24](#_Toc160112777)

[Appendix C: Glossary 25](#_Toc160112778)

[Appendix D: Referenced Documents 26](#_Toc160112779)

[Appendix E: Approvals 27](#_Toc160112780)

[Appendix F: Additional Appendices 28](#_Toc160112781)

[Appendix G: Database Design Document – Sample Template 30](#_Toc160112782)

[5.2. Functional Design Decisions 32](#_Toc160112783)

[**9.3** 6.1. Data Software Objects and Resultant Data Structures 32](#_Toc160112784)

[**9.4** 6.2. Database Management System Files 33](#_Toc160112785)

[**9.5** 7.1. Roles and Responsibilities 33](#_Toc160112786)

[**9.6** 7.2. System Information 34](#_Toc160112787)

[**9.7** 7.2.1. Database Management System Configuration 34](#_Toc160112788)

[7.2.2. Database Support Software 35](#_Toc160112789)

[7.2.3. Security and Privacy 35](#_Toc160112790)

[7.3. Performance Monitoring and Database Efficiency 35](#_Toc160112791)

[7.3.1. Operational Implications 35](#_Toc160112792)

[7.3.2. Data Transfer Requirements 36](#_Toc160112793)

[7.3.3. Data Formats 36](#_Toc160112794)

[**9.8** 7.4. Backup and Recovery 36](#_Toc160112795)

[Appendix H: Interface Control Document 37](#_Toc160112796)

[**1.1** Interface Overview 37](#_Toc160112797)

[1.1.1 Interface Controls 37](#_Toc160112798)

[**1.2** Functional Allocation 38](#_Toc160112799)

[**1.3** Data Transfer 38](#_Toc160112800)

[**1.4** Transactions 38](#_Toc160112801)

[**1.5** Security and Integrity 39](#_Toc160112802)

[**2.** Detailed Interface Requirements 40](#_Toc160112803)

[**2.1** Requirements for Interface with an External System 40](#_Toc160112804)

[2.1.1 Assumptions 40](#_Toc160112805)

[2.1.2 Technical Interface Requirements 40](#_Toc160112806)

[2.1.3 General Processing Steps 40](#_Toc160112807)

[2.1.4 Interface Processing Time Requirements 40](#_Toc160112808)

[2.1.5 Message Format (or Record Layout) and Required Protocols 40](#_Toc160112809)

[2.1.6 Communication Methods 41](#_Toc160112810)

[2.1.7 Security Requirements 41](#_Toc160112811)

[**3.** Quality Assurance 41](#_Toc160112812)

List of Figures

[Figure 1 – System Architecture Diagram 11](#_Toc160120519)

[Figure 2 – Database Relational Schema 13](#_Toc160120520)

[Figure 3 – UI Interface Diagram 16](#_Toc160120521)

[Figure 4 – Security Detailed Design 20](#_Toc160120522)

List of Tables

[Table 1 - Record of Changes 23](#_Toc160119796)

[Table 2 - Acronyms 24](#_Toc160119797)

[Table 3 - Glossary 25](#_Toc160119798)

[Table 4 - Referenced Documents 26](#_Toc160119799)

[Table 5 - Approvals 27](#_Toc160119800)

[Table 6 - OSI Application Layer 37](#_Toc160119801)

[Table 7 - OSI Presentation Layer 37](#_Toc160119802)

## Introduction

This System Design Document (SDD), which will be referred to as SDD, applies to the development of the Automated Loan Eligibility Prediction (ALEP) software, which will be referred to as ALEP. This SDD describes the technical specifications and architectural blueprint necessary for the implementation of the ALEP system. The SDD is expected to change throughout the development of ALEP, reflecting any changes or refinements in the technical design specifications as the project progresses. As ALEP will be handling financial information, sufficient security and privacy measures will need to be considered. Measures will be implemented to safeguard the sensitive applicant information and ensure compliance with regulatory standards.

### Purpose of the SDD

The purpose of the SDD is to document the necessary information required to define the architecture and system design. It is to serve as a guide for the development team, providing guidance on the technical infrastructure and design considerations essential for the development of the ALEP system, so that they may understand the goals, functionality, and constraints.

## General Overview and Design Guidelines/Approach

This section describes the principles and strategies to be used as guidelines when designing and implementing the system.

### General Overview

This software is intended for financial institutions, including banks and credit unions, to automate the process of evaluating loan eligibility for applicants. End-users include loan officers, financial analysts, and other staff involved in the loan approval process. The ALEP software aims to streamline the loan approval process by analyzing applicant data that was submitted by the application form and determining the applicant’s eligibility for the loan. The scope of the ALEP software includes analyzing applicant data from the loan application form to assess the applicant’s eligibility for a loan. The main goals are to automate the loan approval process, reduce manual errors, and enhance the overall efficiency of the lending process.

### Assumptions/Constraints/Risks

#### Assumptions

There are a few assumptions concerning the ALEP system. The system is assumed to run effectively on hardware configurations meeting the following minimum requirements:

* an Intel Core i5 or higher and equivalents
* RAM that is 8GB or higher
* At least 100GB of available disk space for storage

The system is assumed to be compatible with the following operating systems:

* Windows 10 or higher
* macOS Catalina (version 10.15) or higher
* Linux 20.04 LTS or higher

The system is assumed to be compatible with libraries Scikit-Learn version 1.4 and XGBoost version 2.0.3, which are the current versions as of February 2024. Any updates to these libraries may impact the system’s functionality and require corresponding updates to maintain compatibility. Additional assumptions are made regarding the stability and reliability of the libraries that are used in the system. It is assumed that proper version control and dependency management practices will be followed to mitigate any potential issues stemming from updates or changes to the libraries.

#### Constraints

The ALEP system should be compatible with a variety of hardware environments, including desktop computers and laptops, to accommodate different end-user preferences and accessibility requirements. Compatibility with various operating systems (Windows, macOS, Linux) is essential to ensure widespread usability and accessibility across different platforms. The ALEP system must comply with industry standards and financial regulations related to data privacy and security, such as the Gramm-Leach-Bliley Act (GLBA). Security is a critical constraint for the system, requiring access controls and data protection measures to safeguard sensitive financial information and ensure compliance with regulatory standards. The ALEP system must meet performance requirements to provide real-time loan eligibility predictions and ensure a responsive user experience. Rigorous verification and validation processes are essential for ensuring the accuracy and reliability of the system’s predictions. This includes testing procedures to validate data inputs, evaluate model accuracy, and assess system performance under various scenarios. The system may be subject to licensing requirements for third-party software libraries.

#### Risks

In the design of the ALEP system, there are risks that could potentially impact the successful development and deployment of the system. It is necessary to identify these risks and implement appropriate mitigation strategies to address them.

* Data Privacy and Security Risks:
  + Risk: The ALEP system deals with sensitive financial data, including income, credit history, and employment status, which poses a risk of data breaches or unauthorized access.
  + Mitigation: Implement encryption techniques, access controls, and data anonymization to protect sensitive information. Conduct regular security audits and compliance assessments to ensure adherence to data privacy regulations.
* Model Accuracy Risks:
  + Risk: The accuracy of loan eligibility predictions generated by the ML model may be affected by biases in the training data or changes to lending criteria over time.
  + Mitigation: Continuously monitor and evaluate the performance of the ML model using historical data and feedback from end-users. Implement bias detection and mitigation techniques, such as dataset preprocessing, threshold tuning and fairness constraints, to minimize inaccuracies in predictions.

### Alignment with an Enterprise Architecture framework

The ALEP system, being a standalone, self-contained, web-based application, is not explicitly designed with the intent to be deployed in an existing environment or network. The system is designed to function independently without extensive dependencies on external structures. However, it is meant to be used as an application for loan officers in financial institutions. While the ALEP system is standalone, the financial institutions’ frameworks can provide standardized guidelines and principles for system development. To not align with a particular enterprise architecture framework, the system will remain consistent with the design goals, emphasizing autonomy, streamlined functionality, and ease of integration within the existing infrastructure of financial institutions. The Software Requirements Specification document does not mandate alignment with a specific enterprise framework, allowing ALEP to maintain flexibility and adaptability to diverse financial environments.

## Design Considerations

### Goals and Guidelines

* **Performance Optimization:** ALEP prioritizes swift response times to ensure users receive loan decisions promptly, enhancing overall user experience and operational efficiency.
* **Precision in Predictions:** The foremost objective is to ensure ALEP's loan eligibility predictions are highly accurate, fostering trust and reliability among users.
* **Scalability Planning:** ALEP is designed to seamlessly accommodate increasing user volumes and data loads, facilitating smooth scalability without compromising performance.
* **User-Centric Design:** ALEP is designed with a focus on user-friendliness and aesthetic appeal, employing intuitive interfaces and visually pleasing layouts to enhance user engagement.
* **Security Measures:** Safeguarding user data integrity is paramount, necessitating stringent security measures such as robust authentication protocols and data encryption mechanisms.

### Development Methods & Contingencies

* **Iterative Development Approach:** ALEP adopts an iterative development methodology, enabling incremental enhancements based on user feedback and evolving requirements.
* **Architectural Adaptability:** ALEP's architecture is designed to be flexible, facilitating easy integration of new features and adaptability to changing technological landscapes.

### Architectural Strategies

In designing the ALEP system, several key architectural strategies have been employed to ensure efficient organization and functionality. These strategies focus on the overall system architecture and the mechanisms used to achieve its objectives. Below are the main design decisions and strategies:

* **Programming Language:** The system is developed using Python for its versatility, ease of development, and extensive libraries for machine learning tasks. Python provides a robust environment for implementing the machine learning model and integrating it with the web application.
* **Machine Learning Model:** A decision was made to leverage an existing machine learning model for loan eligibility prediction rather than developing one from scratch. This decision was based on the availability of pre-trained models that offer high accuracy and reliability.
* **Web Application Framework:** Flask, a lightweight and flexible web framework for Python, is used to build the backend of the web application. Flask provides the necessary tools for handling user authentication and serving dynamic content.
* **User Interface Design:** The frontend of the application is developed using Angular, HTML, and CSS for responsive design and improved user experience. The user interface follows modern design principles to ensure intuitive navigation and accessibility.

### Performance Engineering

Incorporating the performance requirements defined in the Requirements Document, the ALEP system's design prioritizes scalability and responsiveness to meet the demands of processing loan applications efficiently. Performance requirements play a crucial role in shaping the system's design and ensuring optimal performance under various workloads. Here's how performance considerations are integrated into the system's design:

* **Scalability:** The system architecture is designed to be horizontally scalable, allowing it to handle increasing numbers of concurrent users and loan application requests. This scalability is achieved through modular design principles, where components can be independently scaled based on demand.
* **Response Time:** To meet the performance requirement of providing instant eligibility decisions, the system optimizes response time by minimizing processing overhead and leveraging caching mechanisms where applicable. This ensures that users receive timely feedback on their loan applications.
* **Resource Optimization:** The system efficiently utilizes computational resources to maximize throughput and minimize resource contention. Techniques such as lazy loading and efficient memory management are employed to optimize resource utilization and improve overall system performance.
* **Load Testing:** Performance testing is conducted to validate the system's ability to handle anticipated workloads and identify potential bottlenecks. Load testing scenarios simulate various usage scenarios to assess the system's performance under different conditions and identify areas for optimization.
* **Continuous Monitoring:** Once deployed, the system undergoes continuous monitoring to track key performance metrics such as response time, throughput, and resource utilization. This allows for proactive identification of performance degradation and timely intervention to maintain optimal system performance.

By integrating performance engineering principles into the system's design, ALEP aims to deliver a highly responsive and scalable loan eligibility prediction platform that meets the performance expectations of users and stakeholders.

## System Architecture and Architecture Design

### Logical View

### Hardware Architecture

Not Applicable.

#### Security Hardware Architecture

Not Applicable.

#### Performance Hardware Architecture

Not Applicable.

### Software Architecture

The software architecture for the ALEP system comprises various components distributed across different layers to support its functionality. Below is an overview of the software components needed to support the system:

Presentation Layer

* Angular Framework
* HTML, CSS, and JavaScript

Application Layer

* Flask Framework
* Python
* RESTful APIs

Data Layer

* PostgreSQL Database
* SQL

Operating Systems

* Windows 10, macOS Catalina, Linux 20.04 LTS

The presentation layer interacts with the application layer via RESTful APIs, while the application layer interacts with the data layer through SQL queries executed on the PostgreSQL database. This layered architecture ensures separation of concerns and facilitates scalability, maintainability, and extensibility of the ALEP system.

#### Security Software Architecture

The security software architecture of ALEP encompasses various security controls and mechanisms implemented within the software stack. Role-based access control (RBAC) is enforced to restrict user permissions based on predefined roles and privileges. Data encryption algorithms are employed to protect sensitive information stored in the database, ensuring confidentiality and integrity.

#### Performance Software Architecture

The performance software architecture of ALEP focuses on optimizing resource utilization and minimizing latency in application processing. Caching mechanisms are employed to cache frequently accessed data and computation results, reducing database access latency.

### Information Architecture

The information architecture of ALEP defines the structure and organization of data stored within the system. It encompasses various types of information, including loan application details, user profiles, and audit logs. Personally identifiable information (PII) and sensitive data are encrypted at rest and during transit to maintain confidentiality and compliance with data protection regulations.

#### Records Management

The records management aspect of ALEP ensures proper handling and storage of data in compliance with legal and regulatory requirements. Data retention policies are defined to govern the lifecycle of data, including archival and deletion procedures. Access controls are enforced to restrict unauthorized access to sensitive information, and data encryption techniques are employed to protect data integrity and confidentiality.

##### Data

ALEP manages a variety of data types, including user demographics, financial information, and loan application documents. Data sources may include manual inputs from users, automated data feeds from external systems, and internally generated transactional data. Data formats range from structured database records to unstructured documents and multimedia files.

##### Manual/Electronic Inputs

Manual and electronic inputs submitted by users are processed and validated before being entered into the system's master database. Verification checks are performed to ensure data accuracy and consistency, and error handling mechanisms are implemented to handle invalid or incomplete inputs.

##### Master Files

The master database of ALEP stores comprehensive records of user profiles, loan applications, and system activities. Data stored in the master files are organized into relational tables with well-defined schemas, enabling efficient querying and retrieval of information. Database indexes and optimization techniques are employed to enhance data access performance and scalability.

### Internal Communications Architecture

The internal communications architecture of ALEP defines the network topology and communication protocols used for inter-component communication. The system relies on standard network protocols such as TCP/IP for data transmission between application servers, databases, and external interfaces. Secure communication channels are established using encryption and digital certificates to protect data in transit.

### Security Architecture

The security architecture of ALEP encompasses a comprehensive set of security controls and measures to protect the confidentiality, integrity, and availability of system resources and data. Access controls, encryption, intrusion detection, and logging mechanisms are implemented to mitigate security risks and vulnerabilities. Regular security assessments and audits are conducted to ensure compliance with industry standards and regulatory requirements.

### Performance

The performance of ALEP is optimized through efficient resource utilization, scalability, and responsiveness. Performance monitoring tools and metrics are employed to track system performance and identify areas for optimization. Load testing and capacity planning exercises are conducted to assess system scalability and determine hardware and software requirements for handling expected workloads.

### System Architecture Diagram

A detailed system architecture diagram illustrating the interconnected components of ALEP is provided. The diagram, referenced in Appendix D, showcases the interaction between software modules and communication components, emphasizing the flow of data and control within the system.

A diagram of a software process

Description automatically generated

Figure 1 – System Architecture Diagram

## System Design

### Business Requirements

Not Applicable.

### Database Design

#### Data Objects and Resultant Data Structures

1. **User Table**
   1. Data Structure:
      1. userID: Primary key. Unique identifier for each user.
      2. username: Name of the user.
      3. email: Email address of the user.
      4. password: Password for user authentication.
2. **LoanApplications Table**
   1. Data Structure:
      1. applicationID: Primary key. Unique identifier for each loan application.
      2. userID: Foreign key referencing the User Table. Identifies the user who submitted the loan application.
      3. loan\_amount: Amount of the loan requested.
      4. credit\_score: Credit score of the applicant.
      5. employment\_status: Status of employment of the applicant.
3. **LoanDecisions Table**
   1. Data Structure:
      1. decisionID: Primary key. Unique identifier for each decision.
      2. applicationID: Foreign key referencing the LoanApplications Table. Identifies the loan application associated with the decision.
      3. answer: Decision on the loan application (e.g., approved, rejected).
      4. reason: Explanation or reason for the decision.
      5. decision\_data: Additional data related to the decision, if applicable.

#### File and Database Structures

1. **File and Database Structures:**
   1. The database will be stored in PostgreSQL, with each table representing a file structure.
   2. PostgreSQL organizes data into tablespaces, which are physical locations where database objects can reside. In the context of ALEP, the default tablespace provided by PostgreSQL will be used.
   3. The database schema will consist of tables corresponding to each entity in the logical data model: User, LoanApplications, and LoanDecisions.
2. **Data Structure in PostgreSQL:**
   1. Each table in PostgreSQL corresponds to a relation in the relational model and stores rows of data organized by columns.
   2. Primary keys, foreign keys, and indexes defined in the logical data model will be implemented in PostgreSQL to enforce data integrity and optimize query performance.
3. **Changes to the LDM:**
   1. The LDM may require minor adjustments to accommodate PostgreSQL-specific features or optimizations, noted in Appendix A.

##### Database Management System Files

A diagram of a computer application

Description automatically generated with medium confidence

Figure 2 – Database Relational Schema

The detailed design of the DBMS files, including the physical description of schemas, records, and storage configurations, can be found in the Database Design Document (refer to Appendix G: Database Design Document - Detailed Database Design).

##### Non-Database Management System Files

Not Applicable.

### Data Conversion

For ALEP's PostgreSQL database, no data conversion is typically required for routine operations within the ALEP system, as it operates within a consistent data environment.

However, data conversion may be necessary for preparing data for ML decision models. ML models may require input data in a specific format or encoding. If the data stored in PostgreSQL does not match the expected format, conversion may be needed to prepare the data for use in the ML models. This may include preprocessing steps such as feature engineering, normalization, or encoding.

### User Machine-Readable Interface

**User Classes/Roles:**

1. **Operational Users:** These users are primarily responsible for interacting with the system on a day-to-day basis, submitting loan applications, and accessing decisions.
2. **System Operators:** System operators oversee the overall functioning of the ALEP system, monitor system health, and manage user accounts.
3. **Administrators:** Administrators have elevated privileges to configure system settings, manage security permissions, and generate reports. We expect 2 administrators who may access the system concurrently during administrative tasks.

#### Inputs

* **Data Entry Screens:** The primary input medium for users is web-based data entry screens accessible through modern browsers. These screens allow users to input loan application details, including loan amount, credit score, and employment status.
* **Input Data Screens/GUIs:** Data entry screens are designed with intuitive GUIs, including windows for entering application details. Each GUI component corresponds to specific data elements such as loan amount, credit score, and employment status.
* **Data Element Definitions:** Data elements include:
  + *Loan Amount:* Numeric field representing the requested loan amount.
  + *Credit Score:* Numeric field indicating the applicant's creditworthiness.
  + *Employment Status:* Dropdown menu for selecting employment status options.
* **Edit Criteria:** Mandatory fields include loan amount and employment status. Loan amount must fall within predefined ranges, while employment status requires selection from predefined options. Data entry controls enforce field validation to prevent bypassing edits.

#### Outputs

The system output design of ALEP encompasses various components tailored to meet the needs of users and operators. Key aspects of the output design include:

* **Reports:** ALEP generates reports to provide insights into loan application trends, approval rates, and portfolio performance. These reports are accessible to administrators and managers for decision-making purposes. Examples of reports include:
  + Loan Application Trends: Graphical representation of the number of loan applications received over different time intervals.
  + Loan Approval Rate Over Time: Line chart illustrating the trend of loan approval rates over a specified period.
  + Loan Portfolio Performance: Stacked bar chart showing the composition of the loan portfolio based on loan type, duration, and interest rate.
* **Data Display Screens and GUIs:** ALEP features intuitive GUIs for displaying application status, decision outcomes, and user dashboards. GUI components include:
  + Application Status Dashboard: Provides real-time updates on the status of loan applications, including pending, approved, and rejected.
  + Decision Outcome Screens: Display detailed information about loan decisions, including approval rationale and decision factors.
  + Query Results: Users can query the system to retrieve specific information related to loan applications, decisions, and user activities.

**Purpose of Outputs**

The primary purpose of system outputs is to facilitate informed decision-making, enhance operational efficiency, and provide transparency in the loan application process. Outputs are designed to cater to the needs of various stakeholders, including administrators, managers, and users.

**Access Restrictions and Security Considerations**

Access to system outputs is governed by role-based access control (RBAC) mechanisms to ensure data confidentiality and integrity. Only authorized users with appropriate permissions can access sensitive information such as decision outcomes and loan application details.

### User Interface Design

**A screenshot of a computer screen

Description automatically generated**

Figure 3 – UI Interface Diagram

#### Section 508 Compliance

* **Keyboard Navigation:** All functionalities within the user interface are accessible via keyboard shortcuts to accommodate users with mobility impairments.

## Operational Scenarios

* **SCENARIO 1: Initial Login**
  + Description: The loan officer accesses the ALEP system for the first time and needs to set up their account and credentials.
  + Steps:
    1. The loan officer receives initial access credentials (username and temporary password) from the system administrator.
    2. The loan officer opens the ALEP system login page on their computer.
    3. They enter the provided username and password into the login fields.
    4. The ALEP system prompts the officer to create a new password for their account to enhance security.
    5. The loan officer enters a new password and confirms it to proceed with the account setup.
    6. The ALEP system verifies the password complexity and accepts the new password.
    7. The ALEP system confirms successful account setup and redirects the loan officer to the login page.
    8. The loan officer will then log into the system with their credentials and be sent to the home page.
    9. The loan officer explores the system interface and available features to familiarize themselves with the ALEP system’s functionality.
* **SCENARIO 2: Loan Application Submission**
  + Description: The loan officer receives a loan application from a customer and needs to assess the customer’s eligibility for a loan using the ALEP system.
  + Steps:
    1. The loan officer logs into the ALEP system using their credentials.
    2. They access the loan application form within the system.
    3. They enter the customer’s financial information into the system’s application form.
    4. They submit the application form to the system.
    5. The ALEP system processes the customer’s financial data using the ML prediction model.
    6. The system generates a loan eligibility prediction based on the customer’s financial information.
    7. The loan officer reviews the eligibility prediction and decides whether to approve or deny the loan application.
    8. The loan officer communicates the decision to the customer.
* **SCENARIO 3: Reporting Loan Approval Statistics**
  + Description: The loan officer needs to generate reports on loan approval statistics and trends using the ALEP system.
  + Steps:
    1. The loan officer logs into the ALEP system with their credentials and is taken to the home page.
    2. The loan officer accesses the reporting dashboard within the ALEP system.
    3. The loan officer selects the desired parameters for generating loan approval statistics.
    4. The ALEP system retrieves relevant data from the database.
    5. The ALEP system processes the data to generate reports on loan approval statistics and trends.
    6. The loan officer reviews the generated reports and analyzes loan approval trends over time.
    7. The loan officer uses the insights from the reports to make informed decisions about lending practices.

## Detailed Design

### Hardware Detailed Design

Not Applicable.

### Software Detailed Design

* **Service Identifier:** LoanEligibilityService
* **Classification:** Application Service
* **Definition:** Determines the eligibility of loan applications based on predefined criteria and machine learning models.
* **Requirements:** Satisfies functional requirements related to loan application processing and eligibility determination.
* **Internal Data Structures:** Utilizes internal data structures for storing loan application data and model predictions.
* **Constraints:** Assumes availability of trained machine learning models and sufficient computational resources for prediction.
* **Composition:** Integrates with data storage services, and model prediction services.
* **Users/Interactions:** Interacts with the loan application submission interface and communicates eligibility decisions to users.
* **Processing:** Implements algorithms for feature extraction, model prediction, and decision making based on loan criteria.

### Security Detailed Design

A group of colorful rectangular boxes with text

Description automatically generated

Figure 4 – Security Detailed Design

### Performance Detailed Design

Not Applicable.

### Internal Communications Detailed Design

Not Applicable.

## System Integrity Controls

The ALEP system incorporates various control specifications to ensure data security, auditability, and integrity. The following minimum levels of control are implemented within the system:

* Internal Security:
  + RBAC mechanisms are implemented to restrict access to critical data items based on user roles. Only authorized users, such as loan officers and system administrators, are granted access to specific data elements required for their respective roles.
* Audit Procedures:
  + Reporting Requirements: Audit procedures are established to meet control, reporting, and retention period requirements for operational and management reports. Logs of system activities, user interactions, and data modifications are maintained for auditing purposes.
  + Retention Period: Audit logs are retained according to regulatory requirements and organizational policies to ensure compliance with data retention regulations.
* Application Audit Trails:
  + The system maintains application audit trails to dynamically audit retrieval access to designated critical data. Every access attempt to critical data elements is logged, including details such as user identification, date, time, and the data accessed.
* Standard Tables for Data Validation:
  + Standard tables are predefined and utilized for validating data fields entered by users. These tables contain validated values for specific data fields, ensuring data accuracy and consistency throughout the system.
* Verification Processes for Data Modification:
  + Verification processes are established for additions, deletions, or updates of critical data within the ALEP system. Changes to critical data are logged and audited to maintain data integrity.
* Identification of Audit Information:
  + All audit information within the ALEP system is identified by user identification, network terminal identification, date, time, and the specific data accessed or changed. This ensures traceability and accountability for all system activities and data modifications. Audit logs capture comprehensive details of user actions and data access events for forensic analysis and compliance reporting.

## External Interfaces

Not Applicable.

### Interface Architecture

Not Applicable.

### Interface Detailed Design

Not Applicable.

Appendix A: Record of Changes

Table 1 - Record of Changes

| Version Number | Date | Author/Owner | Description of Change |
| --- | --- | --- | --- |
| 1 | 02/28/2024 | Awwal Ahmed  Sarah Freidel | Initial Document |

Appendix B: Acronyms

Table 2 - Acronyms

| Acronym | Literal Translation |
| --- | --- |
| ALEP | Automated Loan Eligibility Predictor |
| GLBA | Gram-Leach-Bliley Act |
| GUI | Graphical User Interface |
| LDM | Logical Data Model |
| ML | Machine Learning |
| PII | Personally Identifiable Information |
| RBAC | Role-Based Access Control |
| SDD | Software Design Document |
| TCP/IP | Transmission Control Protocol/ Internet Protocol |
| DBA | Database Administrator |
| DBMS | Database Management System |
| I/O | Input/Output |

Appendix C: Glossary

Table 3 - Glossary

| Term | Acronym | Definition |
| --- | --- | --- |
| Gramm-Leach-Bliley Act | GLBA | This is a federal law that protects consumer financial privacy and requires financial institutions to explain their information-sharing practices to their customers and to safeguard sensitive data. |
| Role-Based Access Control | RBAC | An access control method that assigns permissions to end-users based on their role within the organization. |
| <Term> | <Acronym> | <Definition> |

Appendix D: Referenced Documents

Table 4 - Referenced Documents

| Document Name | Document Location and/or URL | Issuance Date |
| --- | --- | --- |
| System Architecture Diagram | https://imgur.com/a/GjZKTGF | 02/27/2024 |
| UI Design Diagram | https://imgur.com/a/TXF1uan | 02/27/2024 |
| Security Detailed Design | https://imgur.com/a/T02gE8D | 02/27/2024 |

Appendix E: Approvals

The undersigned acknowledge that they have reviewed the SDD and agree with the information presented within this document. Changes to this SDD will be coordinated with, and approved by, the undersigned, or their designated representatives.

Table 5 - Approvals

| Document Approved By | Date Approved |
| --- | --- |
| A black and white logo  Description automatically generatedName: Awwal Ahmed, ML Engineer – Loan Logic Solutions Inc. | Date |
| Name: Sarah Freidel, Software Engineer – Loan Logic Solutions Inc. | Date |

Appendix F: Additional Appendices

# DATA DICTIONARY

**1- PROCESSES**

* + createUser:
    - Description: Process to create a new user record in the Users table.
  + retrieveUser:
    - Description: Process to retrieve user information based on userID.
  + updateUser:
    - Description: Process to update user information based on userID.
  + deleteUser:
    - Description: Process to delete a user record based on userID.
  + createLoanApplication:
    - Description: Process to create a new loan application record in the LoanApplications table.
  + retrieveLoanApplication:
    - Description: Process to retrieve loan application information based on applicationID.
  + updateLoanApplication:
    - Description: Process to update loan application information based on applicationID.
  + deleteLoanApplication:
    - Description: Process to delete a loan application record based on applicationID.
  + createLoanDecision:
    - Description: Process to create a new loan decision record in the LoanDecisions table.
  + retrieveLoanDecision:
    - Description: Process to retrieve loan decision information based on decisionID.
  + updateLoanDecision:
    - Description: Process to update loan decision information based on decisionID.
  + deleteLoanDecision:
    - Description: Process to delete a loan decision record based on decisionID.

**2- DATA FLOWS**

* User Data Flow:
  + Description: Flow of user-related data between the application and the Users table in the database.
* Loan Application Data Flow:
  + Description: Flow of loan application data between the application and the LoanApplications table in the database.
* Loan Decision Data Flow:
  + Description: Flow of loan decision data between the application and the LoanDecisions table in the database.

**3- DATA ELEMENTS**

* User Table:
  + userID (Primary key): Unique identifier for each user.
  + username: Name of the user.
  + email: Email address of the user.
  + password: Password for user authentication.
* LoanApplications Table:
  + applicationID (Primary key): Unique identifier for each loan application.
  + userID (Foreign key): Identifies the user who submitted the loan application.
  + loan\_amount: Amount of the loan requested.
  + credit\_score: Credit score of the applicant.
  + employment\_status: Status of employment of the applicant.
* LoanDecisions Table:
  + decisionID (Primary key): Unique identifier for each decision.
  + applicationID (Foreign key): Identifies the loan application associated with the decision.
  + answer: Decision on the loan application (e.g., approved, rejected).
  + reason: Explanation or reason for the decision.
  + decision\_data: Additional data related to the decision, if applicable.

**4- DATA STORES**

* Users Table:
  + Description: Stores information about users registered in the system.
* LoanApplications Table:
  + Description: Stores information about loan applications submitted by users.
* LoanDecisions Table:
  + Description: Stores information about decisions made on loan applications.

Appendix G: Database Design Document

# ASSUMPTIONS/CONSTRAINTS/RISKS

4.1. Assumptions

* The database design assumes that ALEP will primarily interact with users through a web-based interface.
* It is assumed that users will have internet access and will be using modern web browsers to access the application.
* The database design assumes that ALEP will be hosted on a cloud platform, providing scalability and accessibility.
* Assumption that PostgreSQL will continue to be the chosen database management system for ALEP throughout its lifecycle.
* It is assumed that users will have appropriate permissions and credentials to access different parts of the system based on their roles.

4.2. Constraints

* The database design must adhere to regulatory compliance standards regarding the handling of personal and financial data.
* Constraints regarding data privacy and security necessitate the implementation of encryption mechanisms for sensitive data stored in the database.
* Constraints related to data consistency and integrity require the implementation of appropriate database constraints, such as foreign key constraints and unique constraints.
* Constraints on system performance and scalability require efficient indexing strategies and query optimization techniques.
* Budget constraints may limit the resources available for database scaling and maintenance.

4.3. Risks

* Risk of data breaches or unauthorized access to sensitive information stored in the database.
* Mitigation: Implement robust authentication mechanisms, role-based access control (RBAC), and encryption for data at rest and in transit.
* Risk of data corruption or loss due to system failures or hardware malfunctions.
* Mitigation: Implement regular backups, redundancy, and failover mechanisms to ensure data durability and availability.
* Risk of performance degradation under high load or concurrent user access.
* Mitigation: Perform thorough performance testing, optimize database queries, and scale resources as needed to handle increased demand.
* Risk of compatibility issues or conflicts with third-party software components or libraries.
* Mitigation: Conduct thorough compatibility testing, maintain up-to-date documentation, and stay informed about software updates and patches.

# DESIGN DECISIONS

5.1. Key Factors Influencing Design

* ALEP's database design is influenced by critical requirements related to performance, availability, security, and privacy.
* Performance: The database design aims to optimize query performance and ensure responsiveness under varying workloads.
* Availability: High availability is ensured through redundant database instances, automated failover mechanisms, and regular backups.
* Security: The database will enforce stringent access controls, encryption of sensitive data, and logging mechanisms to monitor and detect unauthorized access.
* Privacy: Compliance with data privacy regulations such as GDPR will be ensured through anonymization of personal data and restricted access to sensitive information.

## 5.2. Functional Design Decisions

* **Inputs and Outputs:** The database will accept loan applications submitted through the web interface and provide eligibility decisions as outputs.
* **Processing:** Sequential processing is favored for inserts and updates to maintain data consistency, while random access is optimized for queries to retrieve application details and decision outcomes.

5.3. Database Management System Decisions

* PostgreSQL is selected as the initial DBMS for ALEP due to its robust features, compatibility with Python, and support for ACID transactions.
* The database design incorporates flexibility to adapt to changing requirements through schema modifications and query optimizations.

5.4. Security and Privacy Design Decisions

* **User Classifications:** Users will be classified into roles based on their privileges, such as administrators, loan officers, and applicants, with granular access control enforced through role-based authentication.
* **Access Rights:** Access to sensitive data will be restricted based on the principle of least privilege, with encryption applied to data at rest and in transit to safeguard privacy.

5.5. Performance and Maintenance Design Decisions

* **Backup and Restoration:** Regular backups will be scheduled and automated to safeguard against data loss and ensure business continuity. Automated processes for data restoration will minimize downtime in the event of a system failure or data corruption, thereby enhancing data integrity and reliability.
* **Concurrency Control:** To manage concurrent access and minimize contention, ALEP will implement data partitioning strategies. By partitioning data into smaller segments based on defined criteria (e.g., geographical region, loan type), the system can reduce locking conflicts and enhance parallelism, thereby improving throughput and reducing latency.

# DETAILED DATABASE DESIGN

### 6.1. Data Software Objects and Resultant Data Structures

For each functional data object in the ALEP system, the following data storage and processing procedures will be utilized:

**User Table:**

* Database Objects: Table (Users)
* Stored Procedures:
  + createUser: Creates a new user record in the Users table.
  + retrieveUser: Retrieves user information based on userID.
  + updateUser: Updates user information based on userID.
  + deleteUser: Deletes a user record based on userID.
* Functions: None

**LoanApplications Table:**

* Database Objects: Table (LoanApplications)
* Stored Procedures:
  + createLoanApplication: Creates a new loan application record in the LoanApplications table.
  + retrieveLoanApplication: Retrieves loan application information based on applicationID.
  + updateLoanApplication: Updates loan application information based on applicationID.
  + deleteLoanApplication: Deletes a loan application record based on applicationID.
* Functions:
  + submit\_loan\_application: Handles the submission of loan applications, including data validation and insertion into the database.
  + get\_loan\_application\_details: Retrieves loan application details based on applicationID.

**LoanDecisions Table:**

* Database Objects: Table (LoanDecisions)
* Stored Procedures:
  + createLoanDecision: Creates a new loan decision record in the LoanDecisions table.
  + retrieveLoanDecision: Retrieves loan decision information based on decisionID.
  + updateLoanDecision: Updates loan decision information based on decisionID.
  + deleteLoanDecision: Deletes a loan decision record based on decisionID.
* Functions:
  + get\_eligibility\_decision: Retrieves eligibility decision based on applicationID.
  + update\_loan\_application\_status: Updates the status of a loan application based on decisionID.

### 6.2. Database Management System Files

Physical Database Design

* The database management system (DBMS) files will be structured according to the PostgreSQL relational database model.
* Objects created to support access methods, such as indexed access for efficient data retrieval and manipulation.
* Distribution, partitioning, or other compartmentalization of the data will be implemented to optimize performance and scalability.
* The estimated DBMS file size and volume of data within the file will be monitored and adjusted as needed to accommodate data growth and optimize performance.
* Definition of the update frequency of the database tables, views, files, areas, records, sets, and data pages will be determined based on transaction volume and system requirements.

These updates ensure that the database functions and design decisions align with the requirements and objectives of the ALEP system.

# DATABASE ADMINISTRATION AND MONITORING

### 7.1. Roles and Responsibilities

* **Database Administrator:** Responsible for overall management and maintenance of the database system, including schema design, performance tuning, backup, and recovery procedures.
* **System Administrator:** Responsible for managing the hardware and software infrastructure supporting the database system, ensuring optimal system performance and availability.
* **Security Administrator:** Responsible for implementing and maintaining security measures to protect the integrity and confidentiality of the database, including access control policies and encryption mechanisms.

### 7.2. System Information

The ALEP system is assumed to run effectively on hardware configurations meeting the following minimum requirements:

* CPU: Intel Core i5 or higher, or equivalent
* RAM: 8GB or higher
* Storage: At least 100GB of available disk space

The system is assumed to be compatible with the following operating systems:

* Windows 10 or higher
* macOS Catalina (version 10.15) or higher
* Linux 20.04 LTS or higher

Since the ALEP system is currently in the development stage, the PostgreSQL database will be stored in a local development environment, primarily on developers' machines. This environment will be utilized for testing, debugging, and local development purposes. This hardware configuration and usage of local development environments deviate from the organization's standard architecture. However, as of 2/28/2024, a waiver has been granted for these deviations.

### 7.2.1. Database Management System Configuration

The Database Management System (DBMS) chosen for the initial implementation of the ALEP system is PostgreSQL. The current version of PostgreSQL being used is 13.5, which was released on September 9, 2021. PostgreSQL is targeted to run on standard hardware configurations meeting the minimum requirements outlined for the ALEP system. These hardware requirements include an Intel Core i5 processor or higher, 8GB of RAM, and at least 100GB of available disk space for storage. The storage device utilized for PostgreSQL will typically be a solid-state drive (SSD) to ensure optimal performance. Sizing formulas will be employed to estimate the internal and peripheral storage requirements based on the anticipated volume of data and the expected growth rate of the database.

## 7.2.2. Database Support Software

Database support software for PostgreSQL includes a variety of utilities and tools designed to aid in database management, maintenance, and monitoring. These utilities include pgAdmin, a comprehensive database design and management tool for PostgreSQL, which provides a graphical interface for database administration tasks such as creating and managing databases, executing queries, and monitoring server performance. Additionally, PostgreSQL provides built-in utilities such as pg\_dump and pg\_restore for database backup and restoration, as well as pg\_stat\_statements for query performance analysis. Documentation for these utilities can be found in the PostgreSQL official documentation, including titles, version numbers, and relevant sections.

#### 7.2.3. Security and Privacy

Security and privacy measures for PostgreSQL will be enforced through the use of integrity and access controls implemented at various levels of the database. These controls apply to database components such as schemas, tables, records, and data elements. PostgreSQL provides robust authentication and authorization mechanisms, including role-based access control (RBAC), which allows administrators to define granular permissions for database users and roles. Additionally, PostgreSQL supports encryption at rest and in transit to ensure data confidentiality. Tools such as pgcrypto may be utilized to implement data encryption within the database. Detailed documentation on PostgreSQL security features and best practices can be found in the official PostgreSQL documentation, providing guidance on configuring and managing security settings to meet specific requirements and compliance standards.

#### 7.3. Performance Monitoring and Database Efficiency

Performance monitoring for the ALEP system will be the joint responsibility of the database administrator (DBA), system administrator, and application development team. The DBA will primarily focus on monitoring database-specific metrics such as space utilization, query performance, and index efficiency. The system administrator will monitor overall system resource consumption, including CPU usage, memory utilization, and disk I/O. The application development team will collaborate with the DBA and system administrator to identify performance bottlenecks within the application code and optimize database queries for efficiency.

To facilitate performance monitoring, the ALEP system will leverage various monitoring tools and utilities. These tools include DBMS Utilities such as PostgreSQL's built-in utilities for monitoring database performance, such as pg\_stat\_activity for tracking active database sessions, pg\_stat\_statements for analyzing query performance, and pg\_buffercache for monitoring buffer cache usage. Additionally, third-party monitoring tools like Datadog, New Relic, and Prometheus may be employed to enhance performance monitoring capabilities, providing comprehensive monitoring of system and application metrics.

Interfaces with other systems may impact maintenance activities and performance monitoring. These interfaces include integration with external services for credit scoring, machine learning model inference, and data enrichment. Monitoring interfaces with these external services will be essential to ensure optimal performance and reliability. Furthermore, data exchange with partner systems or external databases for data synchronization and consistency will require monitoring interfaces to maintain efficient operations. Additionally, monitoring the interface between the database and front-end applications developed using Angular will be critical to identify performance issues related to data retrieval and rendering.

#### 7.3.1. Operational Implications

No operational implications have been identified at this stage.

#### 7.3.2. Data Transfer Requirements

No specific data transfer requirements have been identified for the ALEP system.

#### 7.3.3. Data Formats

Data formats will adhere to standard conventions for interoperability and compatibility with other system components. No specific formats have been identified as of now.

### 7.4. Backup and Recovery

Backup and recovery strategies for the ALEP system will include the following:

* Regular Backups: Periodic backups of the PostgreSQL database will be scheduled to ensure data integrity and availability.
* Differential Backups: Incremental or differential backups will be implemented to minimize data loss and optimize backup storage usage.
* Recovery Procedures: Procedures for restoring the database from backups will be documented and tested regularly to ensure effectiveness in case of data loss or system failures.
* Backup Storage: Backup data will be stored securely in offsite locations to mitigate risks associated with hardware failures or disasters.
* Backup Retention: Policies for backup retention will be defined to manage storage space effectively while ensuring compliance with data retention requirements.

Appendix H: Interface Control Document

### Interface Overview

As the ALEP system does not interact with any third-party service or external interface, there are no external systems to describe in this section. The ALEP system operates as a standalone application within the financial institution’s infrastructure. The system is a self-contained web-based application designed to automate the loan eligibility process for financial institutions. It incorporates a ML model to analyze consumer financial data and predict loan eligibility. The system architecture primarily consists of internal components hosted within the financial institution’s network.

The ALEP system provides loan officers with a user-friendly interface to input consumer financial data directly into the application. Key functionalities include:

* Data Input: Loan officers can input financial data using the application’s user interface.
* Data Analysis: The application processes the entered financial data using the ML model to predict loan eligibility.
* Result Display: The predicted loan eligibility results are displayed within the application’s home page for loan officers to review and make informed lending decisions.

The architecture of the ALEP system is to be primarily comprised of internal components hosted within the financial institution’s infrastructure, including:

* Frontend: Developed using Angular framework, providing an intuitive and interactive user interface for loan officers.
* Backend: Implemented using Python with Flask framework, responsible for server-side processing, data validation, loan eligibility prediction, and database interactions.
* Database: Utilizes SQLite for storing consumer financial data and prediction results.

No external interfaces or third-party interactions are involved in the ALEP architecture.

#### Interface Controls

Table 6 - OSI Application Layer

| Interface Type | Interface From | Interface To | Description of Interface | Other Information |
| --- | --- | --- | --- | --- |
| User Interface | User Input | Application | Enables users to input financial data | N/A |
| Prediction Engine | Application | Application | Analyzes financial data to predict loan eligibility | Involves ML algorithm model |
| Reporting Dashboard | Application | User Output | Displays loan eligibility results and statistics | May include interactive charts and graphs |

Table 7 - OSI Presentation Layer

| Interface Type | Interface From | Interface To | Description of Interface | Other Information |
| --- | --- | --- | --- | --- |
| Data Formatting | Application | Application | Formats data for consistency and compatibility | Ensures proper display of information |

### Functional Allocation

End User Interaction:

* End-users, such as loan officers, interact directly with the ALEP system through a user-friendly web interface. They input consumer financial data into the system by filling out an application form provided by the ALEP app. This data includes information such as income, credit history, and other relevant financial details.

Data Processing:

* Upon submission of the application form, the ALEP system processes the inputted data using its loan eligibility prediction engine. This engine analyzes the financial data provided by the end-user to determine the consumer’s eligibility for a loan. The prediction engine considers factors such as credit score, income, and other relevant metrics.

Eligibility Determination:

* Based on the analysis performed by the prediction engine, the ALEP system generates loan eligibility predictions. These predictions are then presented to the end-user through the web interface. The end-user can view the results, including whether the consumer is approved for a loan.

Reporting:

* The ALEP system has reporting capabilities to generate reports on loan approval statistics and trends. These reports provide insights into the lending process and help financial institutions make informed decisions regarding loan approvals.

### Data Transfer

In the ALEP system, data movement among components is primarily facilitated through internal data processing and communication within the application itself. As the system is self-contained, there are no external component systems involved in data transfer. The ALEP system operates as a cohesive unit, where data is processed internally within the application’s architecture. Upon submission of the loan application form by the loan officer, the data is directly received and processed within the ALEP system’s backend. The loan eligibility prediction engine within the ALEP system analyzes the received data to determine loan eligibility and generate predictions. The eligibility results and any associated data are then presented to the loan officer through the web interface. The reporting features of the ALEP system also generate reports on loan approval statistics and trends based on the processed data within the application.

Connectivity among the components within the ALEP application is implemented using internal communication protocols and mechanisms. The web-based interface allows for seamless interaction between the loan officers and the ALEP system, facilitating data input, retrieval of eligibility results, and access to reporting features. Data transfer within the ALEP system utilizes standard web communication protocols such as HTTP(S) for transmitting data between the frontend and backend components. The type of messaging or packaging of data used within the ALEP system is based on industry-standard practices for web-based applications, ensuring secure and efficient data transfer.

### Transactions

Data Submission Transaction:

* Type: One-way transaction
* Description: This transaction involves the submission of consumer’s loan application data by the loan officer through the web-based interface.
* Process: The loan officer fills out the loan application form with relevant data and submits the form to the ALEP system for analysis.
* Interface: The data submission transaction occurs between the frontend web interface and the backend server of the ALEP system.

Data Processing Transaction:

* Type: Internal Transaction
* Description: This transaction involves the internal processing of submitted loan application data within the ALEP system.
* Process: Upon receiving the loan application data, the ALEP system’s backend processes the data using the loan eligibility prediction engine.
* Interface: The data processing transaction occurs entirely within the backend server of the ALEP system.

Eligibility Prediction Transaction:

* One-way transaction
* Description: This transaction involves the generation of loan eligibility predictions based on the processed application data.
* Process: The ALEP system’s eligibility prediction engine analyzes the submitted data to determine the consumer’s eligibility for the loan.
* Interface: The eligibility prediction transaction occurs within the backend server of the ALEP system and results are communicated to the frontend web interface for display to the loan officer.

Reporting Transaction:

* Type: One-way transaction
* Description: This transaction involves the generation of reports on loan approval statistics and trends based on the processed data.
* Process: The ALEP system’s reporting features generate comprehensive reports summarizing loan approval statistics, trends, and other relevant information.
* Interface: The reporting transaction occurs within the backend server of the ALEP system and results are presented to the loan officer through the frontend web interface.

### Security and Integrity

Security and integrity measures are implemented to ensure the confidentiality, authenticity, and integrity of data transmission within the system. Given that the ALEP system operates within a secure environment, with no external interfaces or connections to external networks, the interface between its components does not require elaborate security measures beyond standard internal protocols. Security and integrity measure for the ALEP system include:

Access Security:

* Access to the ALEP system is restricted to authorized users only, such as loan officers, who are authenticated through secure login credentials.
* RBAC is implemented to ensure that users can only access data and functionalities relevant to their roles and responsibilities within the system.

Data Transmission Security:

* As the ALEP is self-contained, data transmission occurs within the internal network infrastructure.
* The transmission medium used for data transfer within the system is a secure local network, ensuring that data remains within the controlled environment of the financial institution’s premises.
* Although data encryption is not mandated for internal data transmission, data integrity is ensured through standard network security measures and protocols.

Data Protection and Integrity:

* Data integrity is maintained through internal data validation and verification mechanisms within the ALEP system.
* Data protection measures are implemented to safeguard sensitive information stored within the system’s databases, including encryption of stored data and regular data backups.

Authentication and Auditing:

* User actions within the ALEP system are audited and logged to track system activities and ensure accountability.
* User authentication and session management mechanisms are employed to verify the identity of users accessing the system and track their interactions with the application.

### Detailed Interface Requirements

Given that the ALEP system is a standalone, self-contained, web-based application, there are no requirements for interfacing with an external system. As a result, no specific interface specifications, technical requirements, processing steps, message formats, communication methods, or security measures are needed for interfacing with external systems.

### Requirements for Interface with an External System

Not Applicable.

#### Assumptions

1. Not Applicable.

#### Technical Interface Requirements

Not Applicable.

#### General Processing Steps

Not Applicable.

#### Interface Processing Time Requirements

Not Applicable.

#### Message Format (or Record Layout) and Required Protocols

Not Applicable.

##### File Layout

Not Applicable.

##### Data Assembly Characteristics

1. Not Applicable.

##### Field/Element Definition

Not Applicable.

#### Communication Methods

Not Applicable.

##### Interface Initiation

Not Applicable.

##### Flow Control

Not Applicable.

#### Security Requirements

Not Applicable.

### Quality Assurance

Not Applicable.