**Requirements Document: Altoro Mutual – Online Banking Platform**

**1. Overview**

Altoro Mutual is a basic online banking platform intended for customers of a fictional retail bank. The platform enables users to manage their personal bank accounts online, including viewing balances, transferring funds, and updating account details. The application also provides limited public access to company information, products, and contact services.

**2. Functional Requirements**

**2.1 Public Website Features**

* Public users can:
  + View basic company information (About Us, Services, Contact).
  + Access financial product pages (e.g., loans, credit cards, savings).
  + Submit a contact form for inquiries.
  + View press releases or blog updates.
  + Navigate using a menu or search bar.

**2.2 Customer Login and Session Management**

* Registered users can log in using a username and password.
* Sessions are maintained using browser cookies.
* Users can log out manually or are logged out after inactivity.

**2.3 Account Summary and Transaction History**

* After login, customers can:
  + View summary of all bank accounts (checking, savings, etc.).
  + View detailed transaction history per account.
  + Download transaction history as a CSV or PDF.

**2.4 Funds Transfer**

* Users can:
  + Transfer funds between their own accounts.
  + Schedule future-dated transfers.
  + View pending and completed transfers.

**2.5 Profile and Password Management**

* Users can:
  + Update profile information (address, phone number, etc.).
  + Change their login password.

**2.6 Messaging Center**

* Users can:
  + View messages or notifications from the bank.
  + Use a simple form to send messages to support staff.

**2.7 Admin Interface**

* Admin users can:
  + View all user profiles and account data.
  + Modify user account status (e.g., freeze account).
  + Post new announcements or press releases.
  + View recent transactions across accounts.
  + Access system configuration options.

**3. Non-Functional Requirements**

**3.1 Performance**

* System should support up to 200 concurrent users with low latency (<1s page loads).

**3.2 Compatibility**

* Works with major modern browsers (Chrome, Firefox, Edge).
* Compatible with LAMP stack (Linux, Apache, MySQL, PHP 5.6+).

**3.3 Security Expectations**

*(Included to simulate a design-ready requirements spec, not actual protections)*

* Login required to access account data or perform transactions.
* All pages available over HTTPS (deployment responsibility).
* Credentials stored in the database.
* No 2FA or IP allowlist by default.

**3.4 Usability**

* Simple, intuitive layout suitable for both desktop and mobile.
* Accessible design for basic ADA compliance.

**3.5 Maintainability**

* Configuration file stores DB credentials and system parameters.
* Modular structure with separate PHP files for each feature.

**4. Assumptions**

* No CAPTCHA, anti-CSRF tokens, or rate-limiting are implemented at this stage.
* Passwords are stored securely (mechanism not specified).
* Logs and auditing are handled via basic Apache logs.

**5. Out of Scope**

* Loan application and approval workflow.
* Integration with external APIs (e.g., Plaid, ACH services).
* Customer support ticket system.