Project Name: Adding Two-Factor Authentication (2FA) Using Google Authenticator
Document Status: In Progress

Target Release Date: End of Q3 2018

---

1. Introduction

The purpose of this document is to outline the requirements for adding Two-Factor Authentication (2FA) to our platform using Google Authenticator. This feature aims to enhance account security by providing an additional layer of authentication for users accessing their accounts through our mobile, web, and desktop portals.

2. Goals and Objectives

- **Enhance Security:** Provide clients and users the option to add an additional layer of security when accessing their accounts.
- **Ensure Integrity:** Maintain the integrity of the client-server connection when users utilize Google Authenticator for authentication.
- **User Experience:** Implement 2FA enablement with a seamless and user-friendly interface consistent with our product standards.
- **Communication:** Develop a marketing campaign to inform customers about the new 2FA feature.
- **Recovery Path:** Offer a clear recovery process for users who lose access to their authentication device.

3. Background and Context

Two-Factor Authentication has become a standard security measure across major technology platforms. As our company often handles highly sensitive documents and information, it is imperative to offer this additional security layer to our customers who request it. Additionally, for user onboarding and high-risk transactions, we should consider making 2FA a mandatory part of the process due to the higher probability of fraud in these areas.

Google Authenticator provides a time-based one-time password (TOTP) that changes every 30 seconds, requiring users to have access to their device for authentication in addition to their username and password.

4. Scope

4.1 In-Scope

- Implementation of 2FA using Google Authenticator across mobile, web, and desktop platforms.
- Development of user interfaces for enabling and managing 2FA within user accounts.
- Integration with existing authentication systems to support 2FA during login.
- Creation of a recovery mechanism for users who lose access to their authentication device.
- Development of marketing materials and campaigns to promote the new feature.

4.2 Out-of-Scope

- Support for alternative 2FA methods (e.g., SMS, email-based codes).
- Overhauling the entire authentication system beyond integrating 2FA.
- Changes to user onboarding processes unrelated to 2FA implementation.

5. User Types

- **All Registered Users:** The option to enable 2FA will be available to all users with an account, without distinction between different user types or roles.

6. Assumptions

- Users will access 2FA features through mobile apps, web browsers, and desktop applications.
- Users do not need a Google account to use Google Authenticator.
- All devices with Google Authenticator installed will use Network Time Protocol (NTP) for clock synchronization.
- Users have access to devices compatible with Google Authenticator.

7. Use Cases

1. **Enabling 2FA:**
    - As a user, I want to enable 2FA on my account to enhance security.
    - As a user, I should be guided through the setup process seamlessly.
2. **Logging In with 2FA Enabled:**
    - As a user with 2FA enabled, I want to be prompted for a verification code after entering my username and password.
    - As a user, I expect the login process to be consistent across all platforms.
3. **Recovery from Lost Device:**
    - As a user, I want a clear recovery path if I lose access to my authentication device.
    - As a user, I should be able to use backup codes or contact support for assistance.
4. **Disabling 2FA:**
    - As a user, I want the ability to disable 2FA if I choose to.
5. **Mandatory 2FA for High-Risk Transactions:**

    - As a company, we may require users to enable 2FA for high-risk transactions to reduce fraud.

8. Functional Requirements

8.1 Enabling Two-Factor Authentication

- **FR1:** Users can enable 2FA from their account settings.
- **FR2:** The system will display a QR code and a manual key for Google Authenticator setup.
- **FR3:** Users must verify 2FA setup by entering a code generated by Google Authenticator.

8.2 Logging In with 2FA

- **FR4:** After entering valid username and password credentials, users with 2FA enabled are prompted to enter a verification code.
- **FR5:** The system validates the TOTP code before granting access.
- **FR6:** If the code is invalid, users are prompted to try again, with a limit on the number of attempts.

8.3 Recovery Mechanism

- **FR7:** Upon enabling 2FA, users receive backup codes to store securely.
- **FR8:** Users can use backup codes to access their account if they lose their authentication device.
- **FR9:** Users can request assistance from customer support if backup codes are unavailable.

8.4 Disabling Two-Factor Authentication

- **FR10:** Users can disable 2FA from their account settings after re-authenticating.
- **FR11:** The system confirms the action before disabling 2FA.

8.5 Cross-Platform Consistency

- **FR12:** The 2FA feature must function consistently across mobile apps (iOS and Android), web browsers, and desktop applications.
- **FR13:** User experience and interface elements should be similar across platforms.

8.6 Notifications and Communication

- **FR14:** Users receive confirmation notifications when 2FA is enabled or disabled.
- **FR15:** Marketing communications are sent to inform users about the availability and benefits of 2FA.

9. Non-Functional Requirements

- **NFR1 (Security):** All data related to 2FA must be stored securely and comply with industry security standards.
- **NFR2 (Usability):** The 2FA setup and login processes should be intuitive and user-friendly.
- **NFR3 (Performance):** 2FA verification should not noticeably delay the login process.
- **NFR4 (Reliability):** The 2FA system must be highly available with minimal downtime.
- **NFR5 (Scalability):** The system should handle an increasing number of users enabling 2FA without performance degradation.

10. Risks and Mitigations

- **Risk 1:** Users may find the 2FA process inconvenient, leading to decreased adoption.
    - *Mitigation:* Provide clear instructions and emphasize the security benefits during setup.
- **Risk 2:** Users might lose access to their authentication device and backup codes.
    - *Mitigation:* Implement a secure but user-friendly recovery process through customer support.
- **Risk 3:** Time synchronization issues may cause valid codes to be rejected.
    - *Mitigation:* Ensure devices use NTP for clock synchronization and provide troubleshooting tips to users.

11. Dependencies

- **D1:** Integration with Google Authenticator APIs and adherence to their guidelines.
- **D2:** Coordination with the design team for UI/UX elements.
- **D3:** Collaboration with the marketing team for the promotional campaign.
- **D4:** Updating customer support protocols to handle 2FA-related inquiries.

13. Open Questions

- **Q1:** Should 2FA be mandatory for certain high-risk user groups or transactions?
- **Q2:** What is the policy for users who repeatedly fail 2FA verification? Will there be account lockouts?
- **Q3:** How will we measure the adoption rate and success of the 2FA feature post-launch?

14. Approvals

- **Product Manager:** [Document Owner's Name]
- **Design Lead:** [Designer's Name]
- **Development Lead:** [SDM's Name]
- **QA Lead:** [QA Engineer's Name]
- **Marketing Lead:** [Marketing Manager's Name]

Note: This PRD serves as a foundational document for the development and implementation of the Two-Factor Authentication feature using Google Authenticator. It is intended to align all stakeholders on the project's objectives, requirements, and expectations.

Please review the document and provide feedback or approval so we can proceed to the next phase of the project. If there are any additional details or adjustments needed, let me know, and I'll be happy to incorporate them.