Login with MFA & reCAPTCHA User Flow

This document describes the user flow for logging in with Multi-Factor Authentication (MFA) enabled, specifically focusing on the differences between the local development environment (using test credentials) and the production environment.

Overview

The login process involves an initial authentication with email and password, followed by a secondary verification step using an SMS code. This process is secured by reCAPTCHA to prevent abuse.

---

1. Local Development Environment (Test Credentials)

When running on `localhost`:

1. **Initial Login**: The user enters their email and password and clicks "Login".
2. **MFA Trigger**: The system detects that MFA is enabled for the account. The UI switches to the "2-Step Verification" screen.
3. **reCAPTCHA (Invisible)**: Since test phone numbers are whitelist-protected, the reCAPTCHA step is typically bypassed or invisible, automatically verifying the request.
4. **SMS Code (Simulated)**: No actual SMS is sent. The system accepts the pre-configured test verification code (e.g., `123456`).
5. **Verification**: The user enters the test code `123456` into the input field and clicks "Login".
6. **Success**: The system verifies the code and completes the login process.

---

2. Production Environment (Live Deployment)

When running on the live site (`ssdcp-react.web.app`):

1. **Initial Login**: The user enters their email and password and clicks "Login".
2. **MFA Trigger**: The system detects that MFA is enabled. The UI switches to the "2-Step Verification" screen.
3. **reCAPTCHA (Visible)**: The "I'm not a robot" reCAPTCHA widget appears. The user must manually check the box (and potentially solve a puzzle) to prove they are human.
4. **SMS Code (Actual)**: Upon successful reCAPTCHA verification, Firebase sends a real SMS with a 6-digit code to the registered phone number.
5. **Verification**: The user checks their phone, enters the received 6-digit code into the input field, and clicks "Login".
6. **Success**: The system verifies the code and completes the login process.

---

3. Sequence Diagram

```
sequenceDiagram
    actor User
    participant UI as Login Page (Client)
    participant Auth as Firebase Auth
    participant SMS as SMS Provider / Phone

    User->>UI: Enter Email & Password
    UI->>Auth: signInWithEmailAndPassword()

    alt MFA Required
        Auth-->>UI: Error: auth/multi-factor-auth-required
        UI->>UI: Switch to MFA Screen (isMfaStep = true)

        rect rgb(240, 248, 255)
            note right of UI: reCAPTCHA & Code Sending

            alt Localhost (Test Phone)
                UI->>Auth: Identify user & sendMfaSignInCode()
                Auth-->>UI: Auto-verified (No visible reCAPTCHA)
                Auth-->>UI: Session ID returned (No real SMS)
            else Production (Real Phone)
                UI->>Auth: Identify user & sendMfaSignInCode()
                User->>UI: Solve reCAPTCHA ("I'm not a robot")
                UI->>Auth: Verify reCAPTCHA Token
                Auth-->>SMS: Send Real SMS Code
                SMS-->>User: Receive 6-digit Code
                Auth-->>UI: Session ID returned
            end
        end

        User->>UI: Enter 6-digit Code
        UI->>Auth: resolveMfaSignIn(code, sessionID)

        alt Valid Code
            Auth-->>UI: Login Success (User Object)
            UI-->>User: Redirect to Dashboard / Logged In State
        else Invalid Code
            Auth-->>UI: Error (Invalid Verification Code)
            UI-->>User: Show Error Message
        end
    else MFA Not Enabled
        Auth-->>UI: Login Success
        UI-->>User: Redirect to Dashboard
    end
```