

Global Relay - Technical test

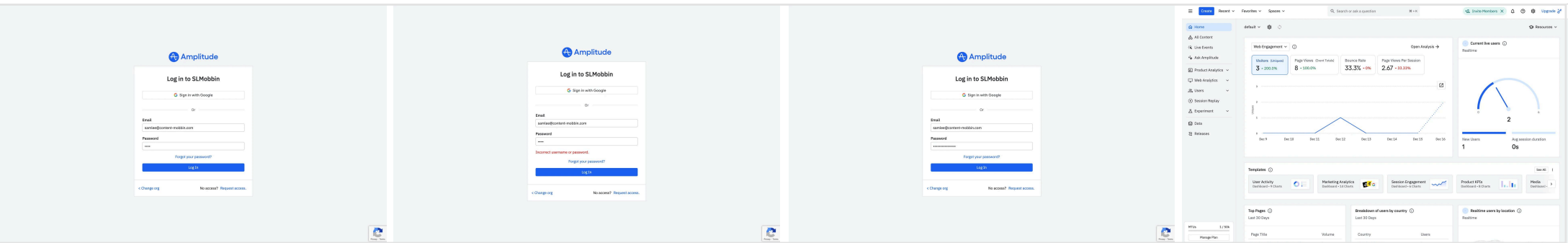
By Jago Livingstone



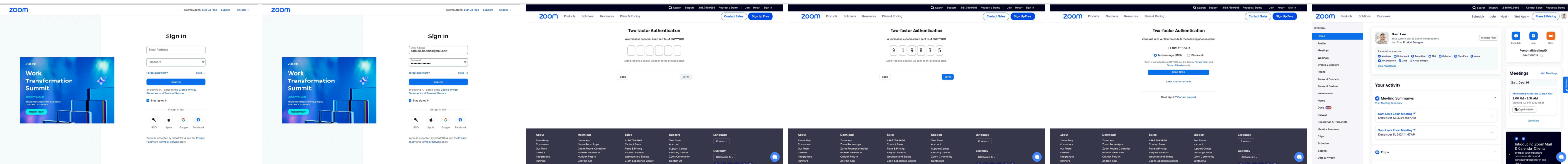
Competitor Analysis

Studied login flows from Miro, Amplitude, Zoom, Stripe, Adobe and coinbase

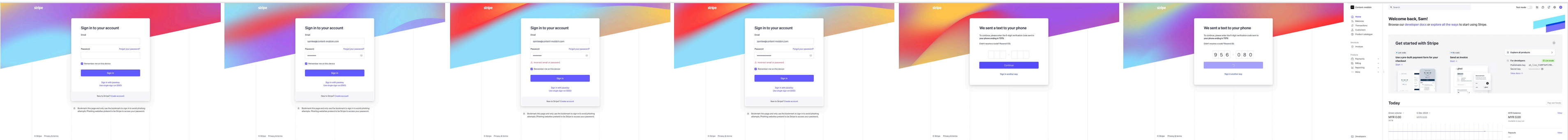
Amplitude



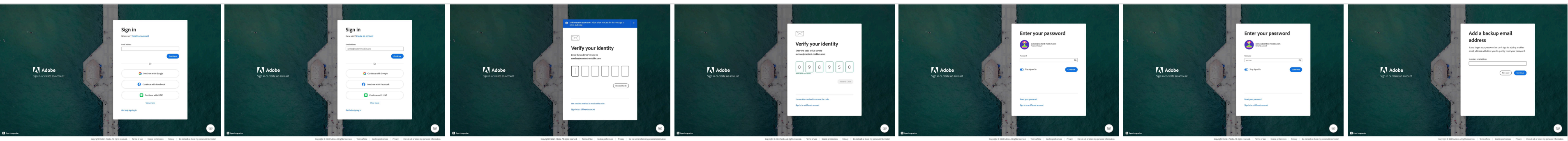
Zoom



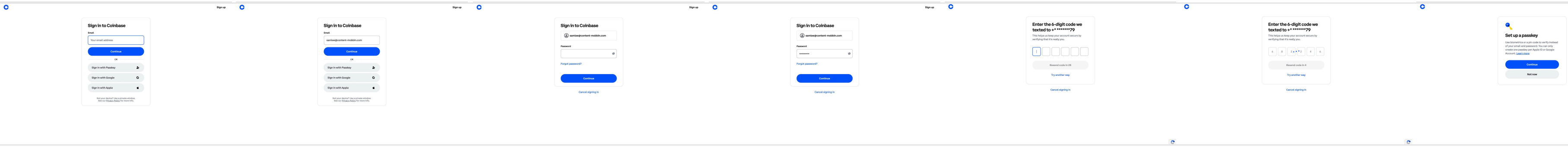
Stripe



Adobe

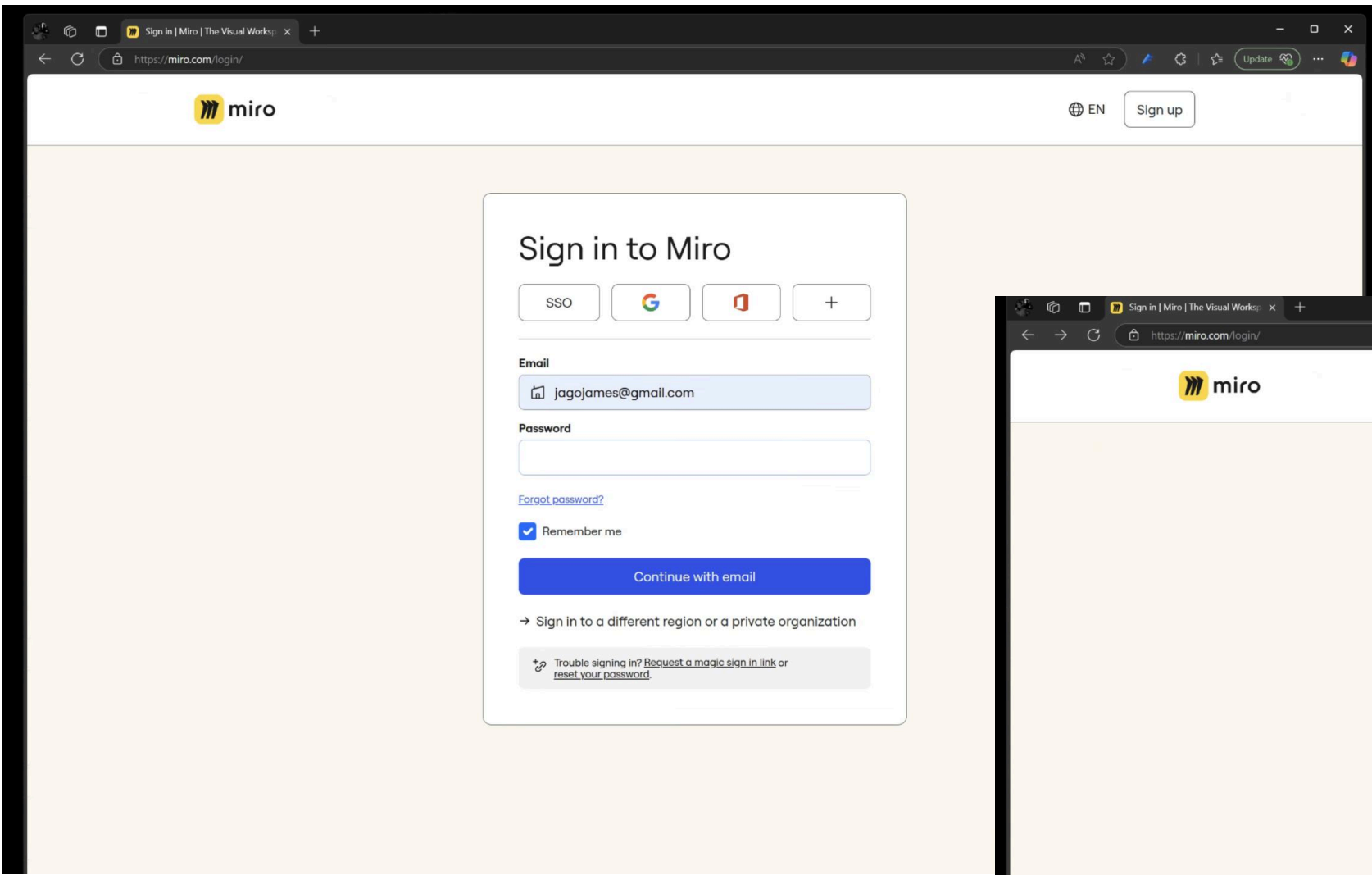


coinbase



Competitor Analysis

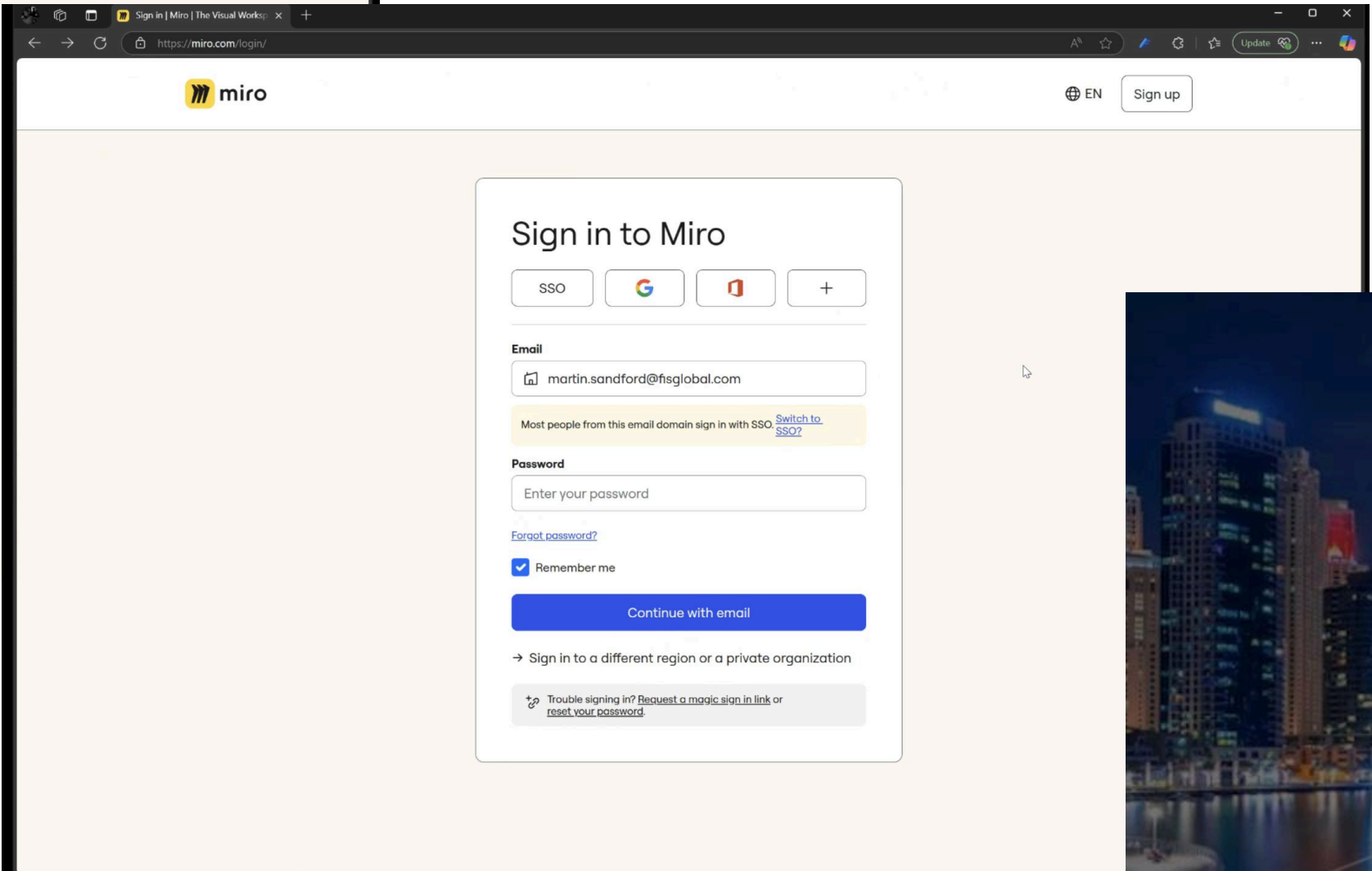
MIRO - Closer look at the flow



User Flow

Step 1: Pretty standard login interface. With three main options: 3rd party auth, SSO options or manually type in password and username

- SSO (Single Sign-On)
- Google Authentication
- Microsoft Authentication
- Custom Authentication ('+' button for more options)
- Manually enter in both username and password

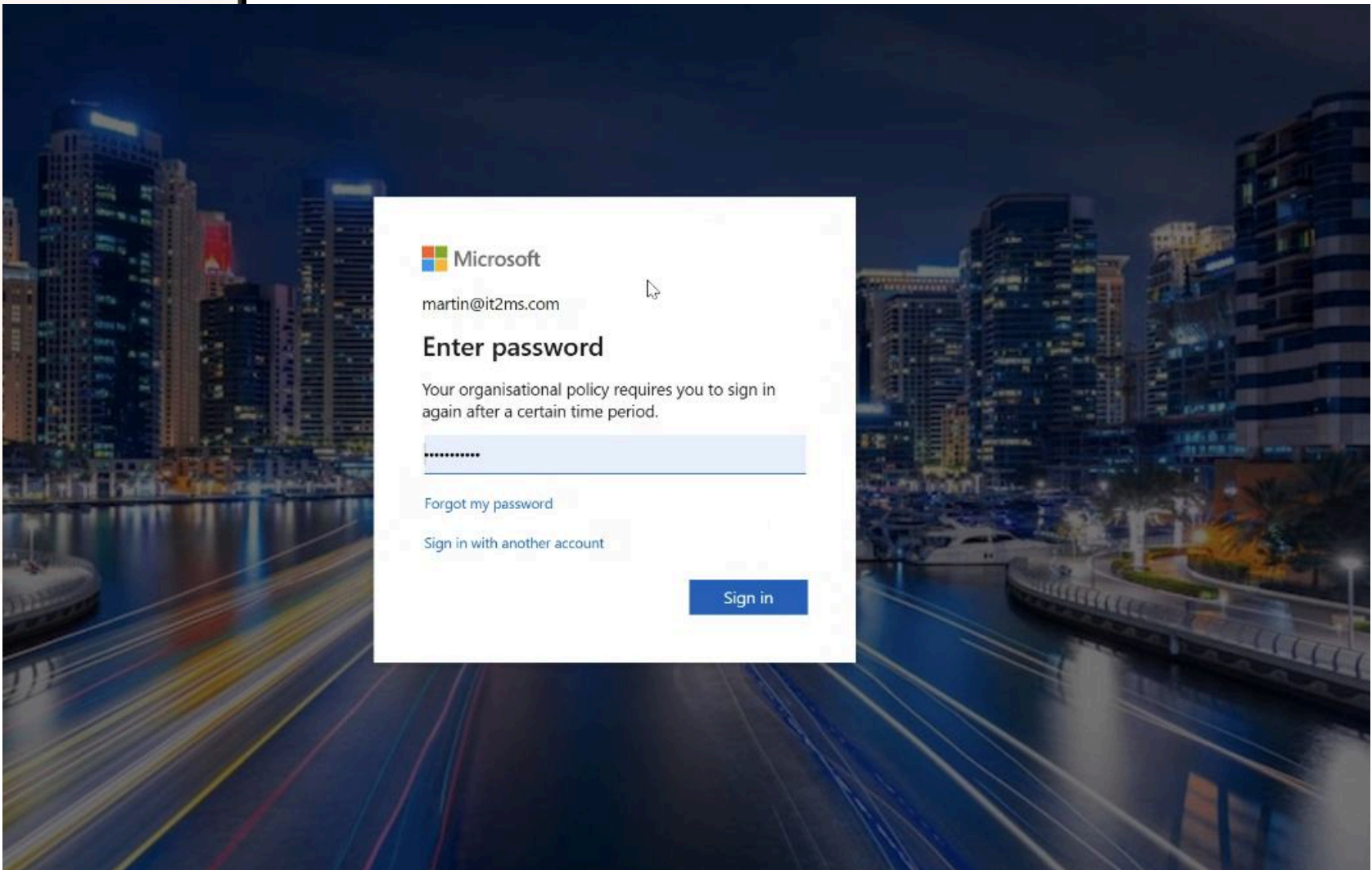


Step 2: SSO Identification Based on Email Domain

Once the user enters an email address. If the domain is associated with an organisation that uses SSO (@fsglobal.com), a yellow notification banner appears, prompting users to "Switch to SSO?"

This nudges users towards using SSO instead of manually entering a password.

If the email domain is not linked to SSO, the form behaves like a traditional login.



Step 3: SSO Redirect

If the user clicks "Switch to SSO?", they are redirected to their organisation's identity provider (e.g., Okta, Azure AD, Google Workspace).

Authentication happens externally before users are returned to Miro, already signed in.

Security Research

Reviewed authentication standards (NIST, OWASP, SAML, SSO).

Information Technology Laboratory

SMALL BUSINESS CYBERSECURITY CORNER

Cybersecurity Basics

NIST Cybersecurity Framework

Quick Start Guides

Events

Guidance by Sector

Guidance by Topic

Telework

Training

Videos

Join a Community of Interest

About

Cybersecurity @ NIST

CONNECT WITH US

Multi-Factor Authentication

You can download a pdf version of this page [here](#).

Protecting Your Small Business: Multi-Factor Authentication

You've recently set up a travel management portal for your small business that requires everyone to log in using a username and password. The portal also requires a user to have an authentication app on their phone for verification of their identity. That way, a one-time code will be accessed in the authentication app and entered into the portal to confirm their identity. This scenario depicts the use and benefits of multi-factor authentication, an increasingly common method to add multiple layers of security to internet-enabled services.

What is Multi-Factor Authentication (MFA)?

Passwords alone are not effective in securing your most sensitive business assets, as they have become too easy for threat actors to access. MFA is an important security enhancement that requires a user to verify their identity by providing more than just a username and password. It requires a user to provide a combination of two or more of the following:

• something you know (like a password or PIN)

• something you have (like a smart card or security key)

• something you are (like your fingerprint or face)

Protecting Your Business from a Common Cyber Threat: Phishing

Due to their effectiveness and simplicity to carry out, phishing attacks have rapidly become the tool of choice for cyber criminals. But what is phishing? Phishing refers to a variety of attacks that are intended to convince you to hand over sensitive information to an imposter. These attacks can come in many forms—most commonly in the form of a convincing email, text message, or social media message. What are they seeking? They're looking for financial gain and your account credentials, such as your password, pin, or one-time passcodes.

How does MFA Protect My Business From this Threat?

If a password is compromised, MFA creates a second barrier that makes it much harder for the threat actor to access your systems and data.

Example: Unfortunately, you received a convincing phishing email from what you thought was your accounting software provider. You entered your credentials into the fake website, giving the imposter your username and password. Thankfully, you have MFA enabled on this account. In addition to a username and password, a user also needs a security key to be granted access. Because the criminal did not have access to this security key, you were able to avert the crisis.

Taking MFA to the Next Level: Phishing-Resistant Authentication

Enabling MFA on all accounts that offer it is essential for reducing the cybersecurity risks to your business. However, some forms of MFA are more secure than others— as some forms of MFA can be susceptible to phishing threats such as One Time Pass (OTP) and SMS based codes.

• FIDO authenticators paired with WAC's Web Authentication API are the most common form of phishing-resistant authentication widely available today. These can take the form of separate hardware keys or be embedded directly into platforms (for example your phone or laptop). Availability, practicality, and security of these "platform authenticators" increasingly puts strong, phishing-resistant authenticators into user's hands without the need for additional devices or dongles.

• Not every transaction requires phishing-resistant authentication. However, for applications that protect sensitive information (such as health information or PII data) or for users that have elevated privileges (such as admins or security personnel) organizations should be enforcing, or at least offering, phishing-resistant authenticators.

• Individuals should explore the security settings for their more sensitive online accounts to see if phishing-resistant authenticators are available and make use of them if they are. In reality, these tools are often easier, faster, and more convenient than the MFA—such as SMS text codes— they may currently be using.

Learn more here: <https://www.nist.gov/blogs/cybersecurity-insights/phishing-resistance-protecting-keys-your-kindom>

Questions to Consider

• Have we completed an inventory of all our systems to determine which ones offer multi-factor authentication?

• Have we enabled MFA on our most sensitive accounts? Are phishing-resistant options available to us for use on our most sensitive applications?

• Do employees understand how to enable MFA and its importance in protecting the business?

• Do we have a policy for requiring use of MFA and phishing-resistant MFA?

• Beyond MFA, are we considering other ways to manage access to our systems and data, such as:

◦ Access to systems and data is limited to only those who need it to do their jobs.

◦ Access is removed when needs change or when employees leave the business.

◦ Administrative privileges to systems and devices are limited to only certain employees.

◦ Using a password manager to create and store strong passwords.

OWASP Cheat Sheet Series

OWASP Cheat Sheet Series

Authentication Cheat Sheet

Table of contents

Introduction

Authentication General Guidelines

User IDs

Authentication Solution and Sensitive Accounts

Implement Proper Password Strength Controls

Implement Secure Password Recovery Mechanism

Store Passwords in a Secure Fashion

Compare Password Hashes Using Safe Functions

Change Password Feature

Transmit Passwords Only Over TLS or Other Strong Transport

Require Re-authentication for Sensitive Features

Consider Strong Transaction Authentication

TLS Client Authentication

Authentication and Error Messages

Authentication Responses

Incorrect and correct response examples

Login

Password recovery

Account creation

Error Codes and URLs

Protect Against Automated Attacks

Multi-Factor Authentication

Login Throttling

Account Lockout

CAPTCHA

Security Questions and Memorable Words

Logging and Monitoring

Use of authentication protocols that require no password

OAuth

Openid

SAML

FIDO

Password Managers

Changing A User's Registered Email Address

Recommended Process If the User HAS Multifactor Authentication Enabled

Recommended Process If the User DOES NOT HAVE Multifactor Authentication Enabled

Notes on the Above Processes

Introduction

Authentication (AuthN) is the process of verifying that an individual, entity, or website is who or what it claims to be by determining the validity of one or more authenticators (like passwords, fingerprints, or security tokens) that are used to back up this claim.

Digital Identity is the unique representation of a subject engaged in an online transaction. A digital identity is always unique in the context of a digital service but does not necessarily need to be traceable back to a specific real-life subject.

Identity Proofing establishes that a subject is actually who they claim to be. This concept is related to KYC concepts and it aims to bind a digital identity with a real person.

Session Management is a process by which a server maintains the state of an entity interacting with it. This is required for a server to remember how to react to subsequent requests throughout a transaction. Sessions are maintained on the server by a session identifier which can be passed back and forth between the client and server when transmitting and receiving requests. Sessions should be unique per user and computationally very difficult to predict. The Session Management Cheat Sheet contains further guidance on the best practices in this area.

Authentication General Guidelines

User IDs

The primary function of a User ID is to uniquely identify a user within a system. Ideally, User IDs should be randomly generated to prevent the creation of predictable or sequential IDs, which could pose a security risk, especially in systems where User IDs might be exposed or inferred from external sources.

Usernames

Usernames are easy-to-remember identifiers chosen by the user and used for identifying themselves when logging into a system or service. The terms User ID and username might be used interchangeably if the username chosen by the user also serves as their unique identifier within the system.

Users should be permitted to use their email address as a username, provided the email is verified during signup. Additionally, they should have the option to choose a username other than an email address. For information on validating email addresses, please visit the [input validation cheatsheet](#) email discussion.

Authentication Solution and Sensitive Accounts

• Do **NOT** allow login with sensitive accounts (i.e. accounts that can be used internally within the solution such as to a back-end / middle-ware / DB) to any front-end user interface

• Do **NOT** use the same authentication solution (e.g. IDP / AD) used internally for unsecured access (e.g. public access / DMZ)

Implement Proper Password Strength Controls

A key concern when using passwords for authentication is password strength. A "strong" password policy makes it difficult or even improbable for one to guess the password through either manual or automated means. The following characteristics define a strong password:

• Password Length

- **Minimum** length of the passwords should be **enforced** by the application. Passwords **shorter than 8 characters** are considered to be weak (NIST SP800-63B).
- **Maximum** password length should be **at least 64 characters** to allow passphrases (NIST SP800-63B). Note that certain implementations of hashing algorithms may cause long password denial of service.

• Do not silently truncate passwords. The Password Storage Cheat Sheet provides further guidance on how to handle passwords that are longer than the maximum length.

• Allow usage of all characters including unicode and whitespace. There should be no password composition rules limiting the type of characters permitted. There should be no requirement for upper or lower case or numbers or special characters.

• Ensure credential rotation when a password leak occurs, at the time of compromise identification or when authenticator technology changes. Avoid requiring periodic password changes; instead, encourage users to pick strong passwords and enable Multifactor Authentication Cheat Sheet (MFA). According to NIST guidelines, verifiers should not mandate arbitrary password changes (e.g., periodically).

• Include a password strength meter to help users create a more complex password and block common and previously breached passwords

- `zxcvbn-ts` library can be used for this purpose.
- `Pwned Passwords` is a service where passwords can be checked against previously breached passwords. You can host it yourself or use the API.

Learn

Discover

Product documentation

Development languages

Topics

Microsoft Entra

Microsoft Entra ID

External ID

Global Secure Access

ID Governance

Permissions Management

Microsoft Security documentation

Troubleshooting

Filter by title

Microsoft identity platform

Overview

What is the Microsoft identity platform?

What's new in docs?

Samples

Concepts

Single-page app (SPA)

Web app

Web API

Desktop

Mobile

Service, daemon, script

Command-line interface (CLI) app

Scenarios

Reference

Token claims reference

Microsoft Authentication Library (MSAL) reference

Protocol reference (OAuth, OIDC, SAML)

OAuth 2.0 application types

OAuth 2.0 and OpenID Connect (OIDC)

SAML 2.0

How Microsoft Entra ID uses the SAML protocol

SAML single sign-on (SSO)

Sign out (SAML)

Token exchange scenarios

WS-Federation

App registration reference

Custom authentication extensions reference

Native authentication API reference

Endpoint reference

API reference

Microsoft auth libraries by app type

AADSTS error code reference

Configured permissions limits troubleshooting

Resources

Learn / Microsoft Entra / Microsoft identity platform /

Single sign-on SAML protocol

Article • 11/29/2024 • 33 contributors

Feedback

In this article

AuthnRequest

Response

This article covers the SAML 2.0 authentication requests and responses that Microsoft Entra ID supports for single sign-on (SSO).

The following protocol diagram describes the single sign-on sequence. The cloud service (the service provider) uses an HTTP Redirect binding to pass an `AuthnRequest` (authentication request) element to Microsoft Entra ID (the identity provider). Microsoft Entra ID then uses an HTTP post binding to post a `Response` element to the cloud service.

Microsoft Entra ID (Identity provider)

User

Application (Service provider)

1. User tries to access the application

2. Application finds the identity provider to authenticate the user

3. Application generates a SAML 2.0 `AuthnRequest` and redirects the user's browser to the Microsoft Entra ID SAML single sign-on URL

4. Application completes user sign-in

5. If the user is not signed in, Microsoft Entra ID authenticates the user and generates a SAML token

6. Microsoft Entra ID posts the SAML response to the application via the user's browser

Note

This article discusses using SAML for single sign-on. For more information on other ways to handle single sign-on (for example, by using OpenID Connect or integrated Windows authentication), see [Single sign-on to applications in Microsoft Entra ID](#).

AuthnRequest

To request a user authentication, cloud services send an `AuthnRequest` element to Microsoft Entra ID. A sample SAML 2.0 `AuthnRequest` could look like the following example:

XML

```
<?xml version='1.0'>
<samlp:AuthnRequest
  xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol"
  ID="C2dE3FH4J5K6LmN7PqR9tBuV1w"
  Version="2.0"
  IssueInstant="2013-03-18T03:28:54.1839884Z"
  xmlns:saml="urn:oasis:names:tc:SAML:2.0:protocol">
  <Issuer xmlns="urn:oasis:names:tc:SAML:2.0:assertion">https://www.contoso.com/</Issuer>
</samlp:AuthnRequest>
```

Copy

Expand table

Parameter	Type	Description
ID	Required	Microsoft Entra ID uses this attribute to populate the <code>IsResponseTo</code> attribute of the returned response. ID must not begin with a number, so a common strategy is to prepend a string like "ID" to the string representation of a GUID. For example, <code>8f8c1c378c366548686870c43a75e482708</code> is a valid ID.
Version	Required	This parameter should be set to 2.0.
IssueInstant	Required	This is a <code>DateTime</code> string with a UTC value and round-trip format ("u"). Microsoft Entra ID expects a <code>DateTime</code> value of this type, but doesn't evaluate or use the value.
AssertionConsumerServiceURL	Optional	If provided, this parameter must match the <code>RelayStateURL</code> of the cloud service in Microsoft Entra ID. Entra ID will honor the ACS URL if it is present in the SAML Request.
ForceAuthn	Optional	This is a boolean value. If true, it means that the user will be forced to reauthenticate, even if they have a valid session with Microsoft Entra ID.
IsPassive	Optional	This is a boolean value that specifies whether Microsoft Entra ID should authenticate the user silently, without user interaction, using the session cookie if one exists. If this is true, Microsoft Entra ID attempts to authenticate the user using the session cookie.

All other `AuthnRequest` attributes, such as `Consent`, `Destination`, and `ProviderName` are ignored.

Microsoft Entra ID also ignores the `Conditions` element in `AuthnRequest`.

Issuer

The `Issuer` element in an `AuthnRequest` must exactly match one of the `ServicePrincipalNames` in the cloud service in Microsoft Entra ID. Typically, this is set to the **App ID URI** that is specified during application registration.

A SAML excerpt containing the `Issuer` element looks like the following sample:

Learn

Discover

Product documentation

Development languages

Topics

Microsoft 365

Solutions and architecture

Apps and services

Training

Resources

Version

Microsoft 365

Filter by title

Microsoft 365 admin center help

Small business help & learning?

YouTube - Microsoft 365 help for small businesses?

Get started

Overview of the Microsoft 365 admin center

Manage users, groups, and passwords

Users

Remove a former employee

Groups

Passwords

Reset passwords

Let users reset passwords

Set passwords to never expire

Password policy recommendations

Turn off strong passwords

Set the password expiration policy

Manage email and calendars

Manage domains

Manage your data and services

Manage subscriptions and billing

Secure your organization

Manage devices and app data

Switch from Google Workspace to M365

Work with customers

Microsoft 365 feature descriptions

Engineering Direct Portal

Troubleshoot

Contact support

Find support phone numbers?

Navigation guide

Learn / Microsoft 365 /

Password policy recommendations for Microsoft 365 passwords

Article • 05/28/2024 • 19 contributors

Feedback

In this article

Understanding password recommendations

Password guidelines for administrators

Some common approaches and their negative impacts

Successful Patterns

Show 2 more

Check out all of our small business content on [Small business help & learning?](#).

As the admin of an organization, you're responsible for setting the password policy for users in your organization. Setting the password policy can be complicated and confusing, and this article provides recommendations to make your organization more secure against password attacks.

Microsoft cloud-only accounts have a predefined password policy that can't be changed. The only items you can change are the number of days until a password expires and whether or not passwords expire at all.

To determine how often Microsoft 365 passwords expire in your organization, see [Set password expiration policy for Microsoft 365](#).

For more information about Microsoft 365 passwords, see:

Reset passwords (article)

Set an individual user's password to never expire (article)

Let users reset their own passwords (article)

Reset a user's password (article)

Time to rethink mandatory password changes?

Understanding password recommendations

Good password practices fall into a few broad categories:

• **Resisting common attacks** This involves the choice of where users enter passwords (known and trusted devices with good malware detection, validated sites), and the choice of what password to choose (length and uniqueness).

• **Containing successful attacks** Containing successful hacker attacks is about limiting exposure to a specific service, or preventing that damage altogether. If a user's password gets stolen. For example, ensuring that a breach of your social networking credentials doesn't make your bank account vulnerable, or not letting a poorly guarded account accept reset links for an important account.

• **Understanding human nature** Many valid password practices fail in the face of natural human behaviors. Understanding human nature is critical because research shows that almost every rule you impose on your users results in a weakening of password quality. Length requirements, special character requirements, and password change requirements all result in normalization of passwords, which makes it easier for attackers to guess or crack passwords.

Password guidelines for administrators

The primary goal of a more secure password system is password diversity. You want your password policy to contain lots of different and hard to guess passwords. Here are a few recommendations for keeping your organization as secure as possible.

• Maintain an fourteen-character minimum length requirement

• Don't require character composition requirements. For example, `*&(%$`

• Don't require mandatory periodic password resets for user accounts

• Ban common passwords, to keep the most vulnerable passwords out of your system

• Educate your users to not reuse their organization passwords for nonwork-related purposes

• Enforce registration for [multi-factor authentication](#)

• Enable risk based multi-factor authentication challenges

Password guidance for your users

Here's some password guidance for users in your organization. Make sure to let your users know about these recommendations and enforce the recommended password policies at the organizational level.

• Don't use a password that is the same or similar to one you use on any other websites

• Don't use a single word, for example, `password`, or a commonly used phrase like `I love you`

• Make passwords hard to guess, even by people who know a lot about you, such as the names and birthdays of your friends and family, your favorite bands, and phrases you like to use

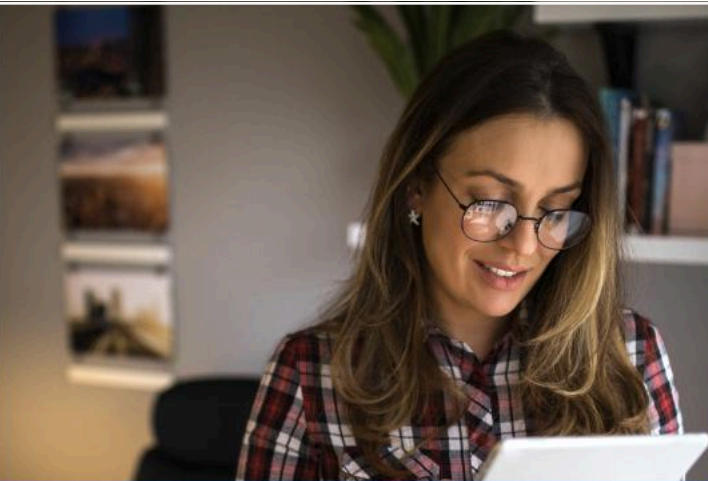
Some common approaches and their negative impacts

They're some of the most commonly used password management practices, but research warns us about their negative impacts.

Password expiration requirements for users

User Needs & Pain Points

Created a persona and then looked at common pain points



Sharon Smith

Modern office worker

"I just want to get to my work without wasting time on login issues. Security is important, but it shouldn't slow me down"

AGE

38

STATUS

Married

LOCATION

UK

#SSOUSER

#FOCUS ON EFFICIENCY, SECURITY,

#MULTIDEVICEUSER

#APRODUCTIVITYDRIVEN

DAY IN THE LIFE

Sharon works as a financial analyst at a global investment firm. Her day is packed with reviewing market data, preparing reports, and collaborating with teams across different regions. Since she frequently switches between devices—laptop in the office, tablet during commutes, and mobile for quick checks—she requires seamless and secure access to hir company’s systems.

Logging in to multiple applications daily is a necessary but frustrating part of her workflow. Security is important, but he expects a smooth and efficient login experience that doesn't disrupt productivity.

GOALS

- Quickly access applications/files with minimal friction.
- Maintain security and compliance without dealing with unnecessary authentication steps.
- Bypass multi-factor authentication & encryption protocols
- Gain persistence within corporate networks for long-term access
- Ensure password recovery is fast and stress-free

NEEDS

- A quick login process that minimises wait times
- Consistent Experience Across Devices
- Protected login information and data through robust authentication
- Intuitive navigation with clear instructions

PAIN POINTS

- Users often struggle with remembering passwords, and the recovery process can be cumbersome
- Variability in the login process across different devices or platforms can create confusion
- Poorly Communicated Error messages and Lack of Guidance
- Users have mental strain if the login systems are not unified

DEVICES & USAGE

Mobile	
Tablet	
Desktop/Laptop	

TECHNICAL KNOWLEDGE

Super User

Intermediate

Basic user

Notes/Suggestions/Insights

- Prefers SSO authentication to avoid multiple login attempts.
- Wants fewer interruptions while working
- Expects high security but also convenience (balance between security & usability).

Persona Overview

- User Type: Modern office workers across various roles (e.g., analysts, managers, IT admins, executives).
- Device Usage: Desktop/laptop primary, mobile secondary.
- Work Environment: Corporate setting with security protocols.
- Authentication Needs: Seamless access with strong security measures (SSO, MFA, password recovery).

Common Pain Points & Frustrations

- ▶ Frequent Password Resets Due to Forgotten Credentials
 - Users often forget passwords and find the reset process slow or confusing.
 - Pain Point: "I reset my password but didn't get the email, now I'm locked out."
- ▶ Multi-Device Authentication Hassles
 - Users get locked out when switching between office, home, and mobile devices.
 - Pain Point: "I logged in from my phone and now my desktop session is expired."
- ▶ Confusing SSO vs. Standard Login Experience
 - Some users don't know if their organisation uses SSO and attempt to log in with a password instead.
 - Pain Point: "I entered my company email and password, but I was required to use SSO which was not clear to me so I struggled to login."
- ▶ Time-Sensitive Logins Disrupted by MFA Delays
 - Multi-factor authentication codes sometimes expire too quickly or don't arrive instantly.
 - Pain Point: "I didn't get my verification code in time, and now I have to start over."
- ▶ Session Expirations & Auto-Logouts
 - Users get logged out unexpectedly and have to re-authenticate frequently.
 - Pain Point: "Why do I have to log in again after just a few hours? It's annoying."
- ▶ *Unclear Error Messages & Lack of Guidance
 - Error messages aren't specific enough, leaving users unsure how to fix login issues.
 - Pain Point: "It just says 'Login unsuccessful'—but I am confused to why. Was it the wrong password or an SSO issue?"
- ▶ IT/Admins Struggle to Support Authentication Issues
 - Helpdesk teams deal with high volumes of login-related tickets.
 - Pain Point: "I have to waste loads of time contacting IT because I don't know how to reset my password."

Defining the Problem

Main Challenge

Design a login panel that accommodates two different login flows (ID/Password & SSO), maintains security, and provides a seamless experience for all users.

Key Considerations

User Clarity – Users don't always know their authentication method.

Security Compliance – Prevent brute force attacks, phishing, and credential leaks.

Consistency – Design should work across desktop, mobile browsers, and shared/public computers.

Scalability – The system should allow for future localisation and additional security features (MFA, passkeys, etc.).

User Flow & Information Architecture

High-Level Login Flow

User enters their Login ID (can be email or another identifier).

System detects login method (Password-based OR SSO).

If Password Login → Prompt for password and authenticate.

If SSO Login → Redirect to customer's authentication page.

Upon successful login → Redirect to Global Relay service.

- ✔ Designed a basic flow diagram.
- ✔ Created wireframes for each scenario (Mobile & Desktop).
- ✔ Included error handling screens for common login failures.

User Flow Enhancements

Streamlined Authentication – Ensures a clear, intuitive path for both SSO and password-based login users.

Optimised Error Handling – Prevents login friction by providing meaningful error messages with actionable next steps.

Security Considerations – Prevents enumeration attacks by not revealing if an account exists during login failures.

Information Architecture Considerations

Logical Step Progression – Users move from login identification → authentication → service access without unnecessary steps.

Contextual Guidance – Provides tooltips, prompts, and inline messages for users unfamiliar with SSO or password reset flows.

Scalability – Designed to accommodate future enhancements, such as biometric authentication.

Accessibility & Usability Improvements

Mobile & Desktop Optimisation – Ensures a consistent experience across devices.

Colour Contrast & Readability – Designed with WCAG compliance in mind for better visibility.

Keyboard & Screen Reader Support – Allows smooth navigation for users with disabilities.

Future Iterations & Testing

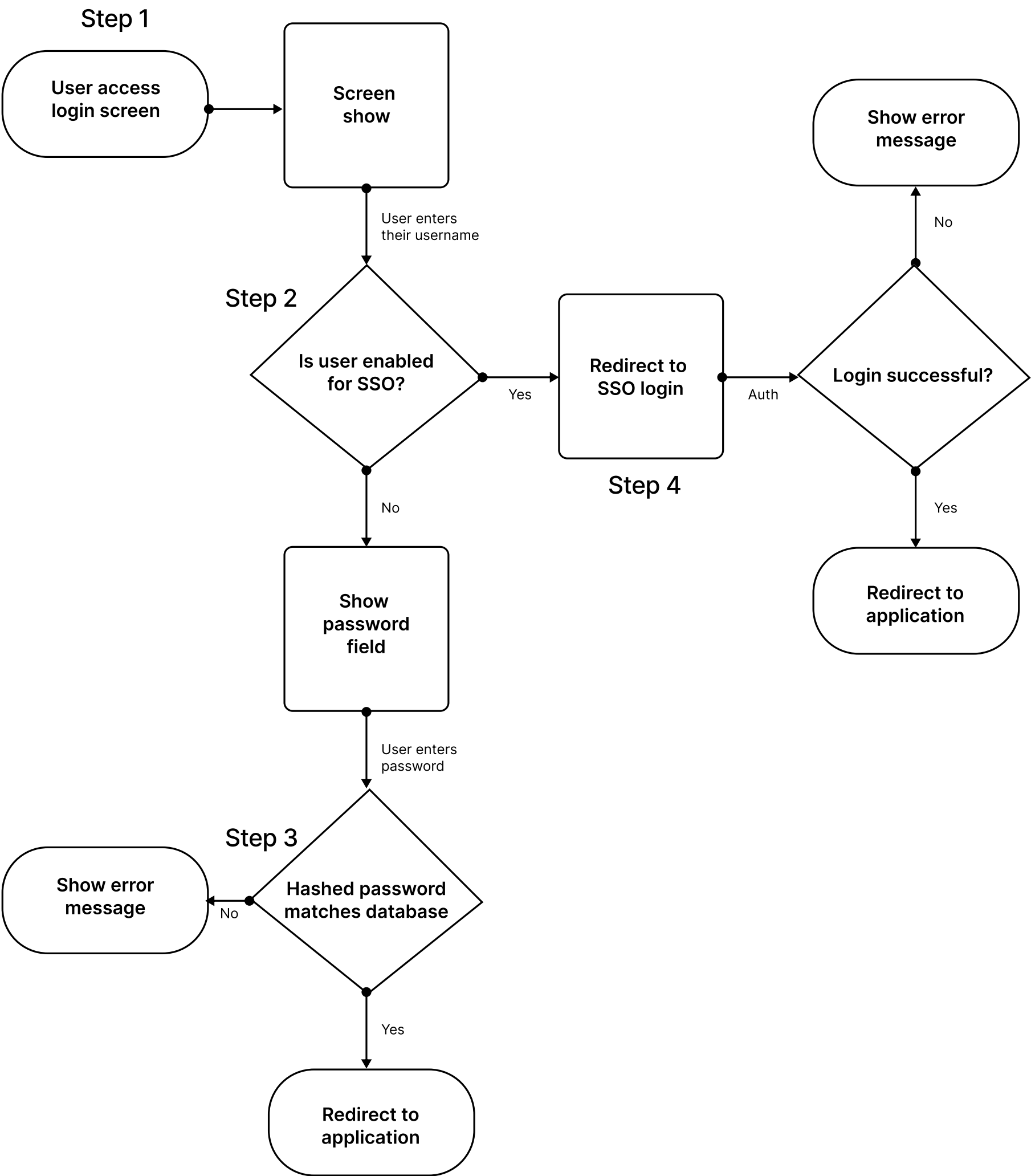
A/B Testing for Login Options – Test variations of SSO vs. password login prominence to reduce friction.

Session Handling Improvements – Evaluate user preferences for "Remember Me" functionality without compromising security.

Performance Tracking – Analyse login drop-off rates and optimise flow based on user behaviour data.

Flow

SSO & Non-SSO Log in process



Step 1: User Accesses the Login Screen

- The user lands on the login page and is prompted to enter their Login ID (email or username).
- The system needs to determine the correct authentication method based on their ID.

Step 2: Determining the Authentication Method

- A decision point checks whether the user is enabled for SSO (Single Sign-On).
- If YES → The user is redirected to their company's SSO provider (e.g., Microsoft, Okta).
- If NO → The system prompts the user to enter their password for manual authentication.

Step 3: Password-Based Authentication (For Non-SSO Users)

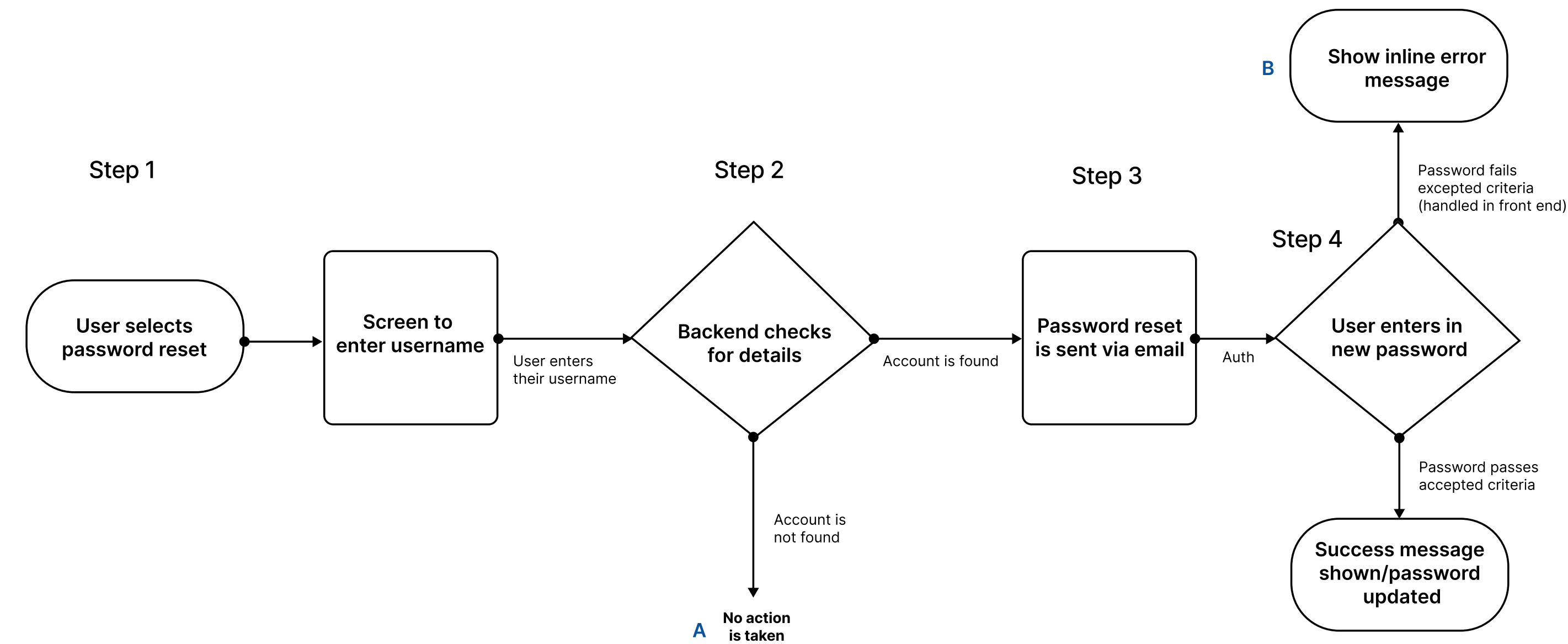
- The user enters their password, and the system checks if it matches the hashed credentials in the database.
- If the password is incorrect → An inline error message appears, guiding the user to try again or reset their password.
- If the password is correct → The user is redirected to the application they were trying to access.

Step 4: SSO Authentication Pathway

- If the user is SSO-enabled, they are redirected to their organization's authentication page.
- After successful authentication, the SSO provider notifies Global Relay that the user is logged in.
- If authentication is successful → The user is redirected to the application.
- If authentication fails → An error message appears, guiding the user to retry or contact support.

Flow

Password reset



A If the entered username does not match an existing account, no confirmation or error message is shown. This prevents potential attackers like Ivan* from discovering valid usernames by testing different inputs. Instead, the system proceeds as if the request was successful, ensuring security through obscurity and allowing the user to progress along their journey.

B The inline errors are simple. More examples are on the next screen.

Step 1: User Initiates Password Reset

- The user selects "Forgot Password" from the login screen.
- They are redirected to a password recovery page, where they must enter their username or email address.

Step 2: Backend Checks for User Account

- The system verifies whether the provided email/username exists in the database.
- If an account is found → The user receives an email with a password reset link.
- If no account is found → No indication is given ("No action is taken"), preventing login enumeration attacks (where attackers try to identify valid accounts).

Step 3: User Receives Password Reset Email & Enters a New Password

- The user follows the link in their email to reach the password reset screen.
- They enter a new password, which is validated on the frontend (e.g., checking length, special characters, and strength).

Step 4: Validation & Success Message

- If the password does not meet criteria → An inline error message appears explaining what needs to be fixed.
- If the password meets criteria → The system updates the password and displays a success message, confirming the reset.

A lean UX design process

Discovery

Uncovering business and consumer needs in context of emerging markets and technology Exploring new opportunities to add real value through collaborative creative design sessions

Definition

Fail fast, fail forward. Testing and iterating possibilities through tailored experiments that help us pivot quickly

Delivery

Deliver iterative releases to the target audiences and learn from measurement and user testing.

Software Development Life Cycle



- 1 Analysis – Define requirements and objectives.
- 2 Design – Plan system architecture and UX/UI.
- 3 Development – Write and build the software
- 4 Testing – Identify and fix bugs.
- 5 Implementation – Deploy the software to users.
- 6 Maintenance – Monitor, update, and improve.

Laws of UX

By JON YABLONSKI 2022

HEURISTIC

★ Aesthetic-Usability Effect

Users often perceive aesthetically pleasing design as design that's more usable.

Fitts's Law

The time to acquire a target is a function of the distance to and size of the target.

Goal-Gradient Effect

The tendency to approach a goal increases with proximity to the goal.

★ Hick's Law

The time it takes to make a decision increases with the number and complexity of choices.

★ Jakob's Law

Users spend most of their time on other sites. This means that users prefer your site to work the same way as all the other sites they already know.

★ Miller's Law

The average person can only keep 7 (plus or minus 2) items in their working memory.

★ Parkinson's Law

Any task will inflate until all of the available time is spent.

GESTAL

★ Law of Common Region

Elements tend to be perceived into groups if they are sharing an area with a clearly defined boundary.

★ Law of Proximity

Objects that are near, or proximate to each other, tend to be grouped together.

★ Law of Prägnanz

People will perceive and interpret ambiguous or complex images as the simplest form possible, because it is the interpretation that requires the least cognitive effort of us.

Law of Similarity

The human eye tends to perceive similar elements in a design as a complete picture, shape, or group, even if those elements are separated.

★ Law of Uniform Connectedness

Elements that are visually connected are perceived as more related than elements with no connection.

PRINCIPLE

Doherty Threshold

Productivity soars when a computer and its users interact at a pace (<400ms) that ensures that neither has to wait on the other.

Occam's Razor

Among competing hypotheses that predict equally well, the one with the fewest assumptions should be selected.

★ Pareto Principle

The Pareto principle states that, for many events, roughly **80%** of the effects come from **20%** of the causes.

Postel's Law

Be liberal in what you accept, and conservative in what you send.

★ Tesler's Law

Tesler's Law, also known as The Law of Conservation of Complexity, states that for any system there is a certain amount of complexity which cannot be reduced.

COGNITIVE BIAS

Peak-End Rule

People judge an experience largely based on how they felt at its peak and at its end, rather than the total sum or average of every moment of the experience.

Serial Position Effect

Users have a propensity to best remember the first and last items in a series.

Von Restorff Effect

The Von Restorff effect, also known as The Isolation Effect, predicts that when multiple similar objects are present, the one that differs from the rest is most likely to be remembered.

Zeigarnik Effect

People remember uncompleted or interrupted tasks better than completed tasks.

The stated laws are the ones I tend to use the most

UX/UI Design Principles

Login, authentication, security, and system errors

Clarity & Guidance

Smart Detection of Login Method – If the user enters an SSO-based email, provide a prompt to redirect them to SSO before they type a password.
Inline Validation & Error Messages – Clearly communicate issues (“Invalid password”- considered and “This email uses SSO” - used).
Step-by-Step Process – Progressive disclosure to reduce cognitive load. Used for the ‘Global Relay Cookie Policy’ and screens load step by step

Security & Compliance

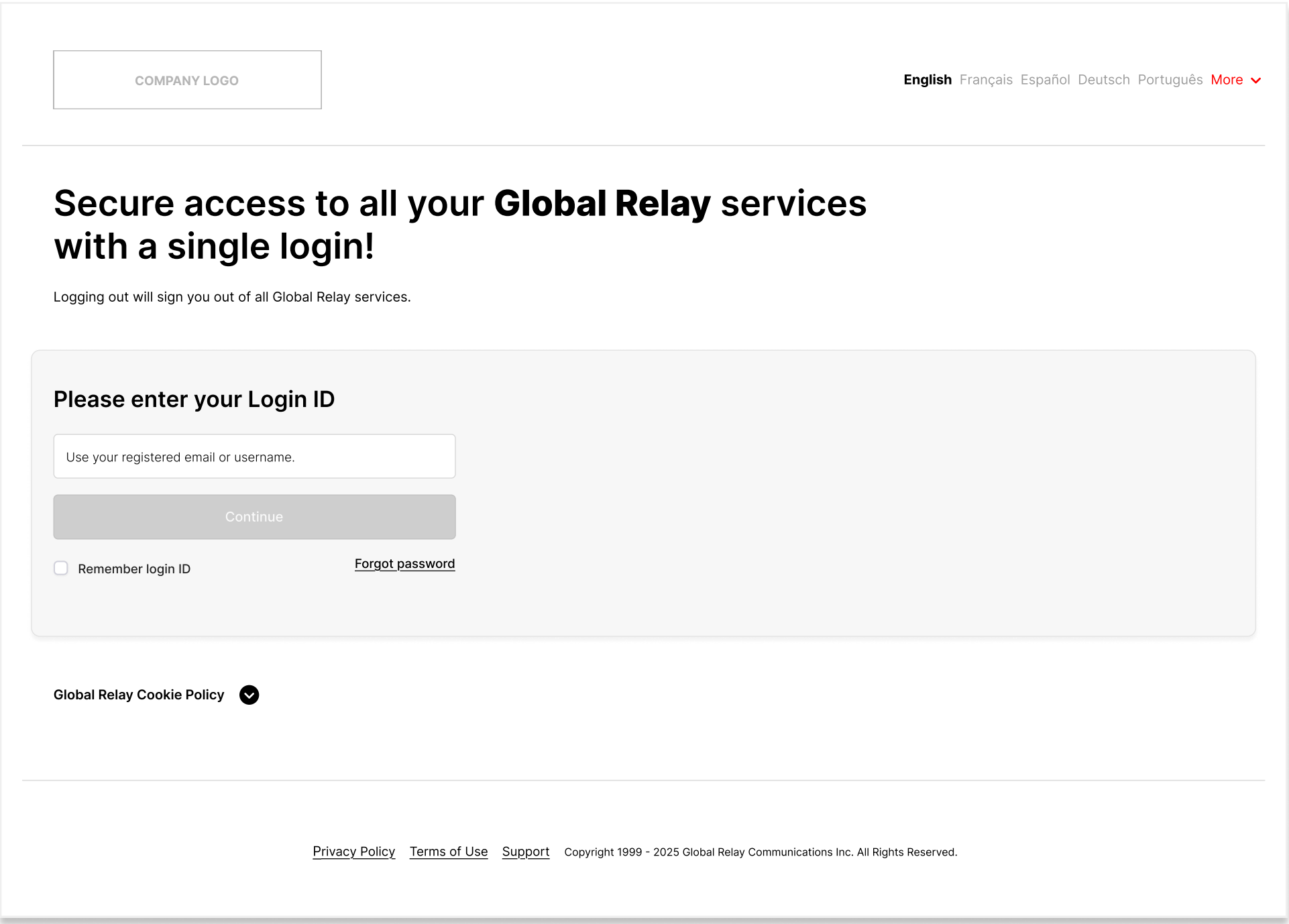
Protect against brute-force attacks - Limit failed attempts and CAPTCHAs have been included.
Prevent valid Login ID enumeration - to avoid revealing whether an email exists or not.
I would also enforce session expiration with warnings e.g. “You’ve been logged out due to inactivity”.
Also have secure password reset flows with verification links expiring quickly.

Accessibility & Performance

Keyboard & Screen Reader Compatibility have been considered – ARIA labels for assistive tech.
I would take into account colour Contrast & Readability and make sure the designs meets WCAG 2.1 AA standards.
After testing we would know how fast the Authentication Process is. We would try to optimise the backend to avoid long wait times and add loading indicators if necessary.

Deliverables - Happy path

Login Landing Page (Desktop)



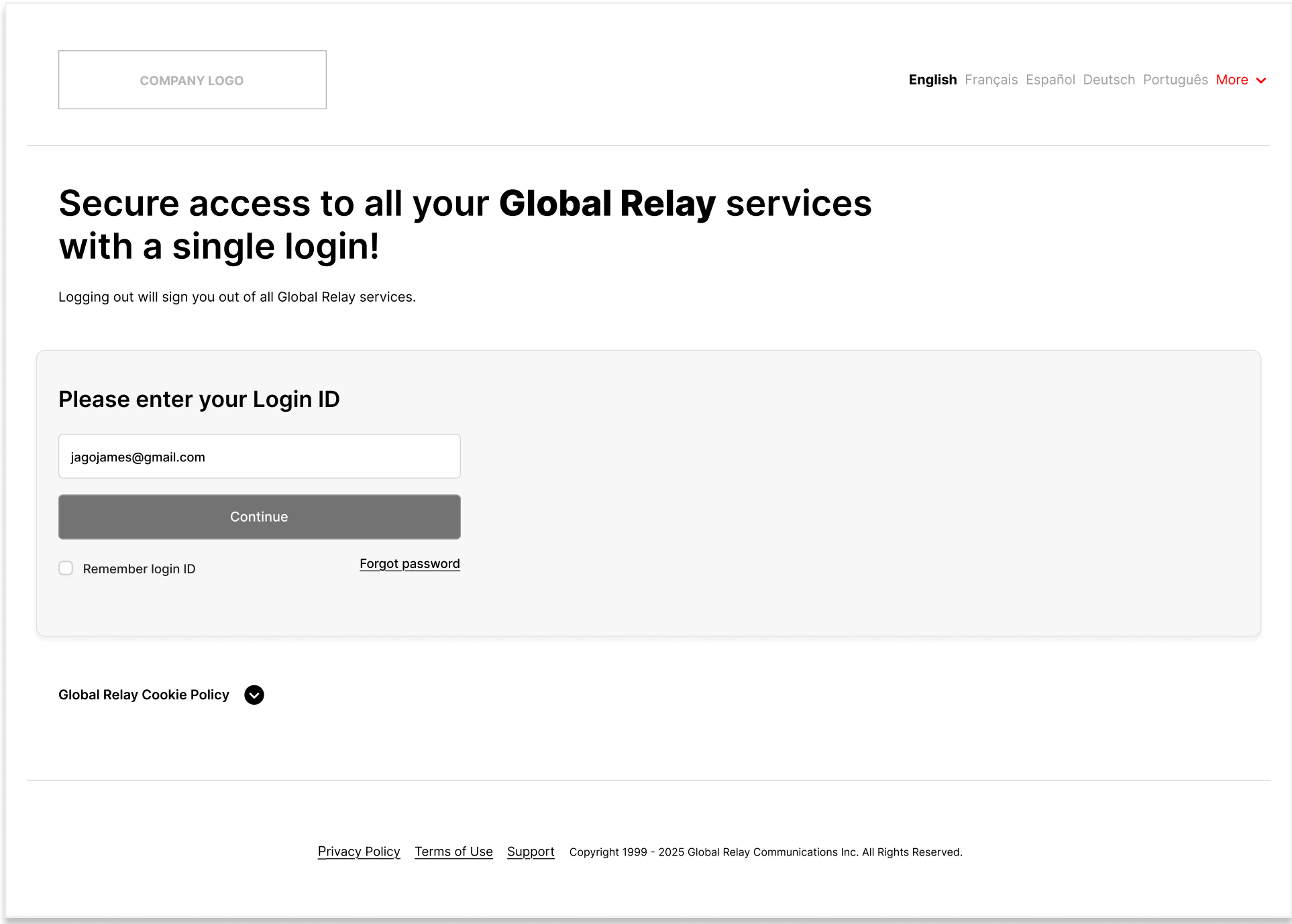
Interaction & Behaviour

User enters their Login ID:

- If the system detects that the user is configured for SSO, they will be redirected to their company's authentication page.
- If the user needs a password-based login, the system will reveal the password field.

Progressive disclosure used (password field only appears if needed and cookie policy).

I have also provided what I think are clear instructions about single login & logout behaviour.



Interaction & Behaviour

The "Continue" button activates as soon as any character is entered in the Login ID field.

This serves two key purposes:

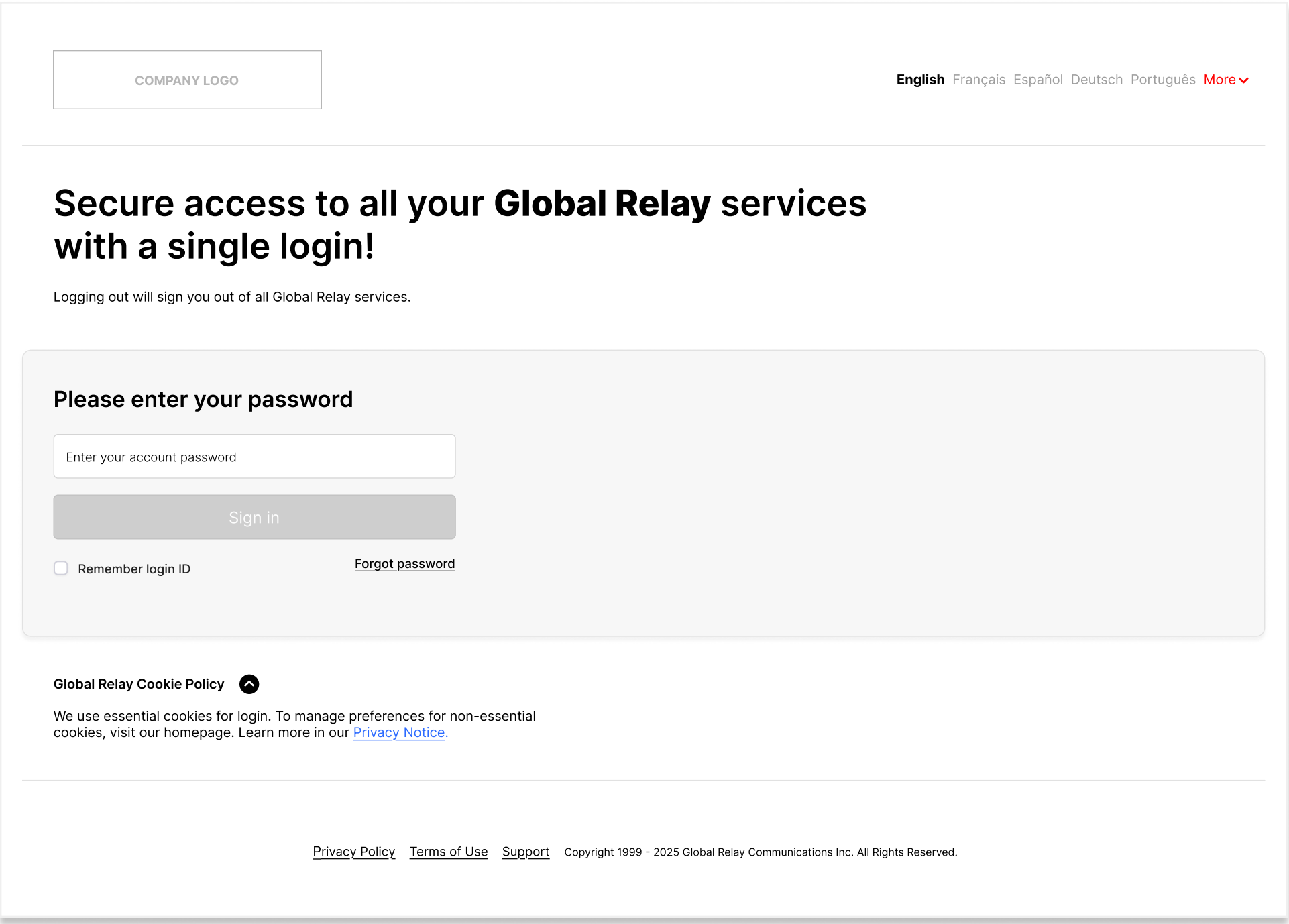
Accessibility: Helps partially sighted and colourblind users identify the active state of the button more easily.

Security: Prevents exposing whether a login ID exists in the system, reducing the risk of username enumeration attacks (where attackers use error messages to determine valid accounts).

You can view tablet layouts in the Figma file

Deliverables - Happy path

Login Landing Page (Desktop)

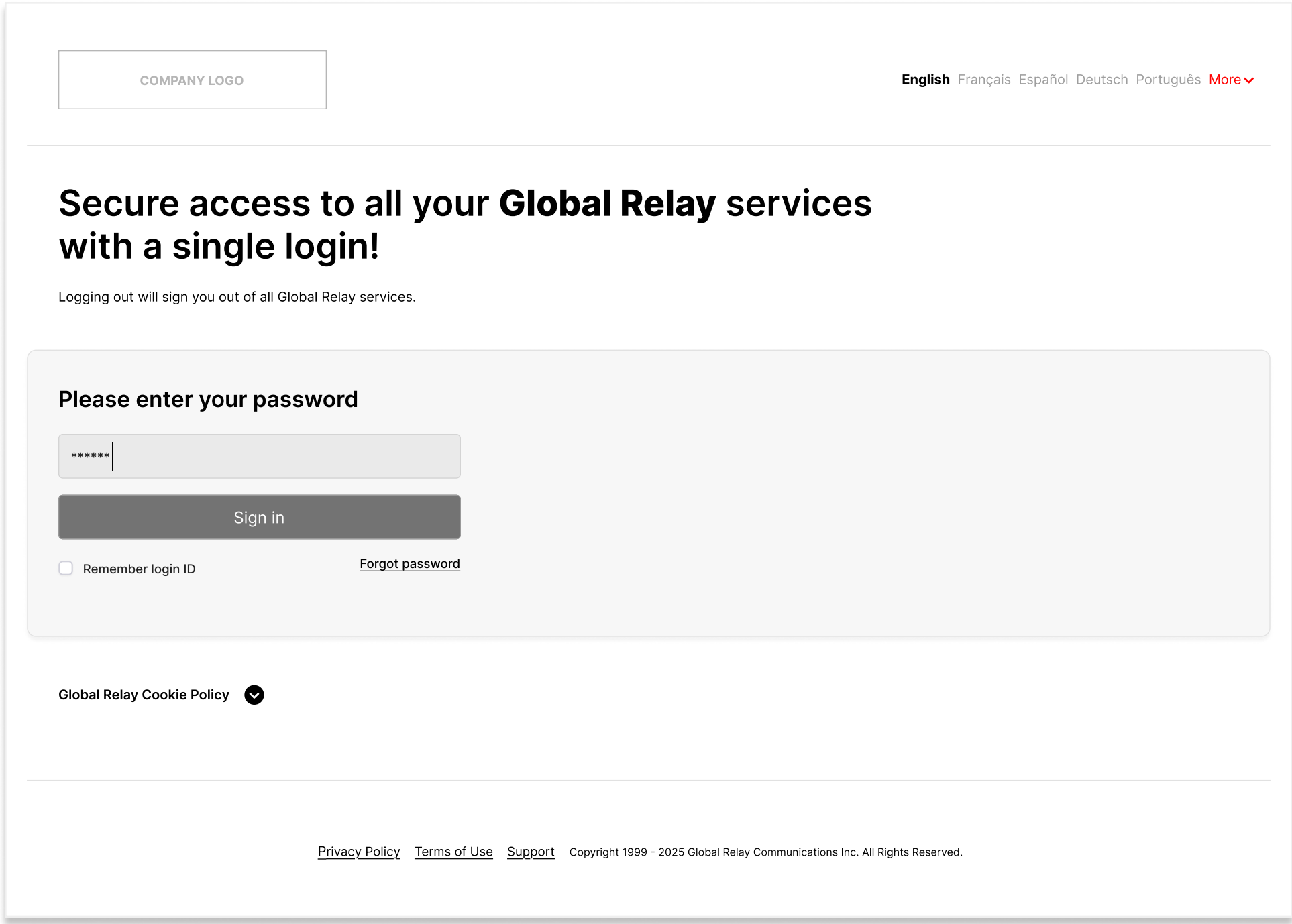


Interaction & Behaviour

In this case, the Login ID was correct, but regardless of its validity, this screen loads to ensure no information is exposed that could help attackers determine valid accounts.

The user now enters their password. I considered adding a progress indicator (such as a step counter or visual markers) to show how many steps remain in the login process. However, I opted for a clean and minimal design to keep the experience focused and distraction-free.

Note: In this wireframe, I have displayed the cookie policy in its expanded state for transparency.

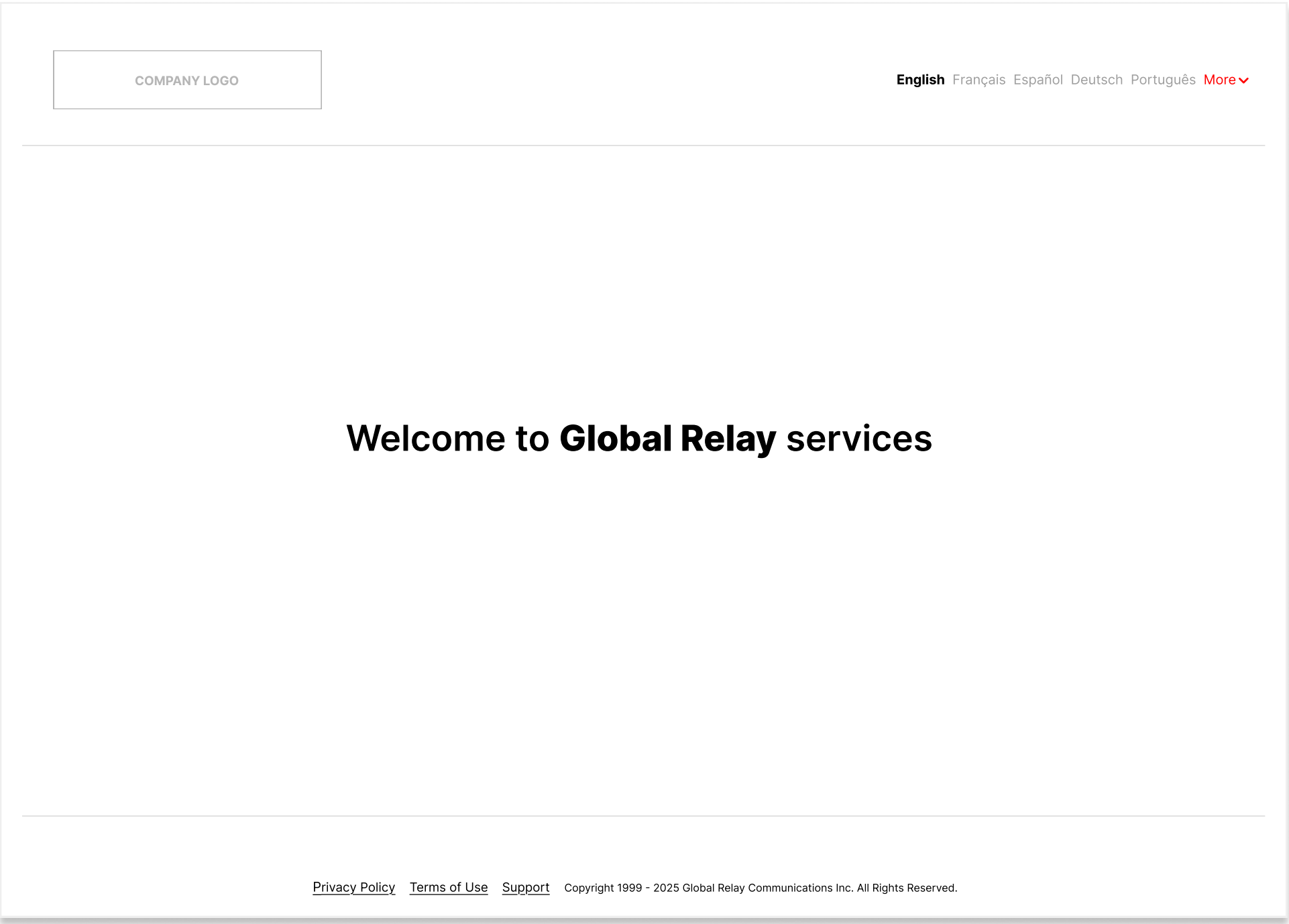


Interaction & Behaviour

Password is being entered and box shows focus.

Deliverables - Happy path

Login Landing Page (Desktop)



Overview

At this stage, the user has successfully authenticated and is now inside the Global Relay environment. This marks the completion of the login process and shows a password-based login journey.

User is Redirected to Their Target Service

- Since Global Relay uses a single login system, the user is now signed in to all associated services.
- The system automatically redirects them to the specific Global Relay service they intended to access.

Persistent Authentication (if enabled)

- If the user selected “Remember Login ID”, their Login ID is pre-filled for future logins.
- Session persistence rules apply based on security policies (e.g., auto-logout after inactivity).

Single Logout Behaviour

- If the user logs out from one Global Relay service, they will be logged out from all services, maintaining security and consistency.

Characters to be restricted

(Login ID only)

Certain characters should not be permitted in user name login field. They should be restricted as they can be used to run arbitrary scripts or cause security vulnerabilities, such as SQL injection or cross-site scripting (XSS). As SQL injection can be used to bypass password validation. However the password should accept special characters.

The following characters should typically be restricted:

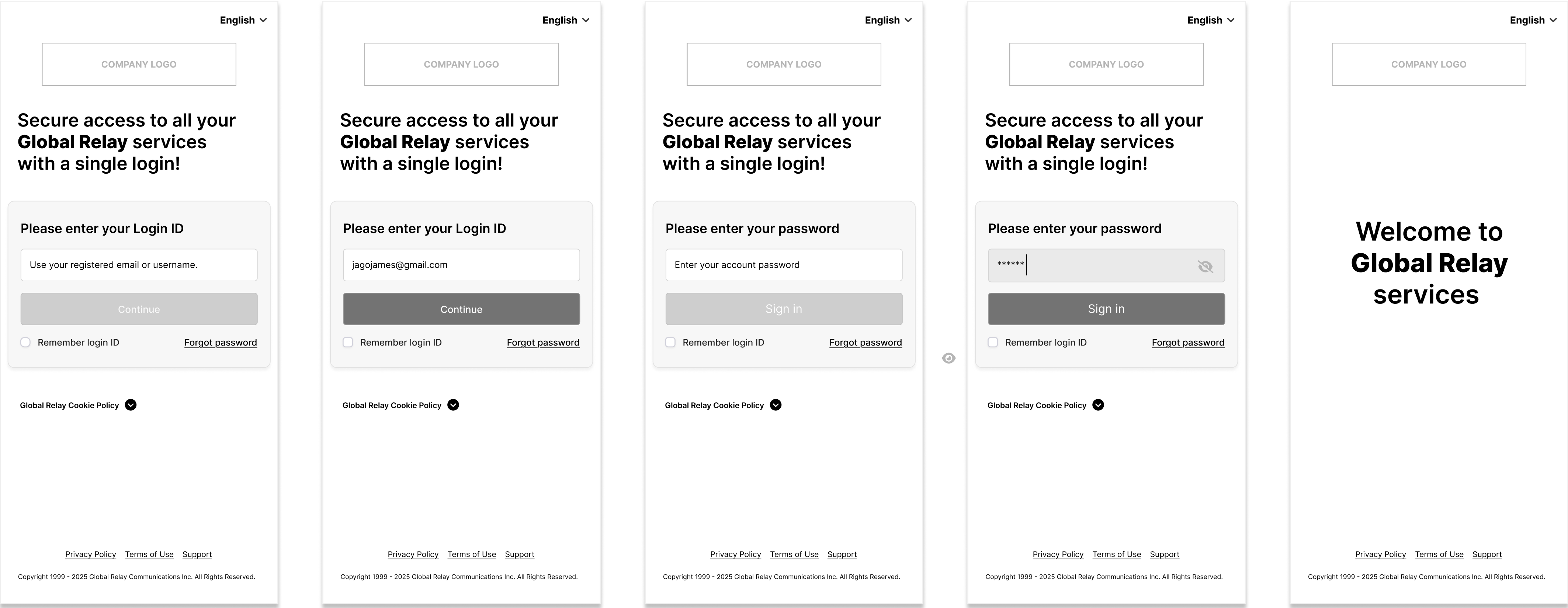
- < (less than)
- > (greater than)
- " (double quote)
- ' (single quote/apostrophe)
- ; (semicolon)
- (double hyphen, used in SQL comments)
- /* and */ (used in SQL and CSS comments)
- \ (backslash)
- | (pipe)
- & (ampersand)
- # (hash, can be used in URLs to manipulate navigation)
- () (parentheses, can be used in script execution)
- { } (curly braces, used in JavaScript and other scripting languages)
- [] (square brackets, used in some programming languages and JSON)

Implementing input validation will help prevent malicious use of these characters.

More info here: <https://owasp.org/www-community/attacks/xss/>

Deliverables - Happy path

Login Landing Page (mobile)

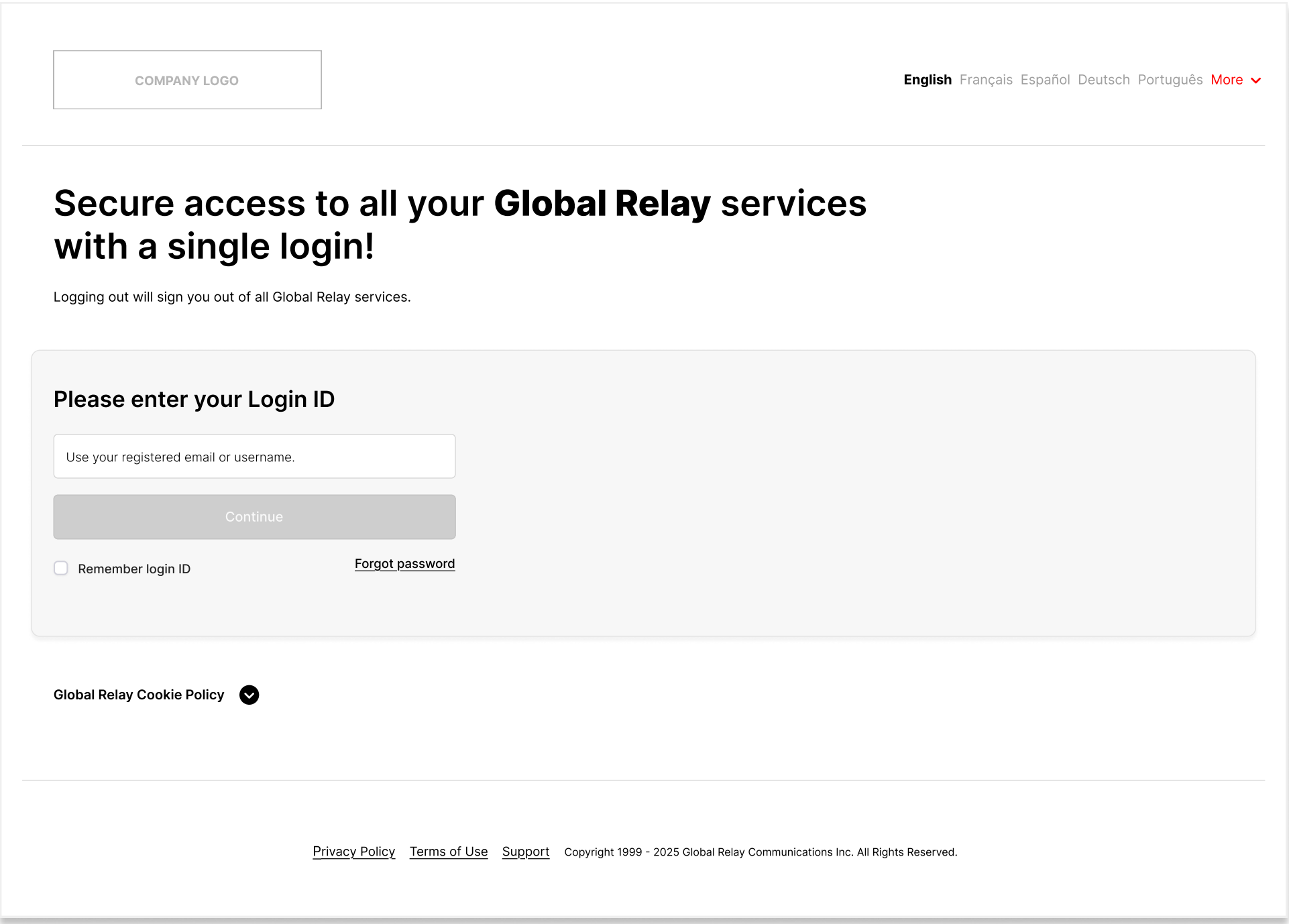


Summary: Key Mobile-Specific Enhancements

- Responsive layout for smaller screens with full-width elements.
- Larger tap targets for touch-friendly interactions.
- Mobile keyboard optimisations for smooth form completion.
- Password visibility toggle for easier password entry.

Deliverables - SSO Happy path

Login Landing Page (Desktop)



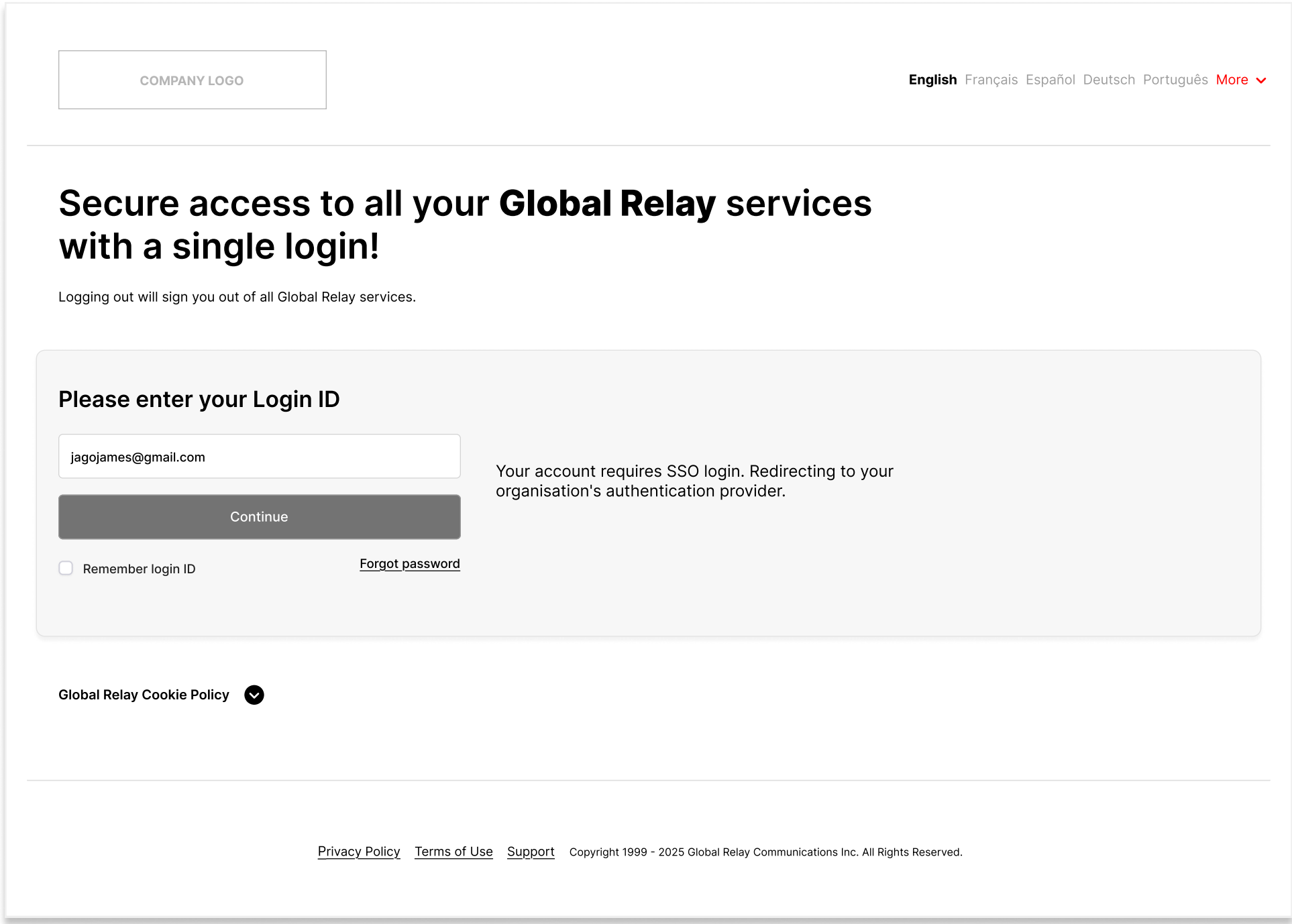
Initial Login Page (Same as Password Flow)

New UI Behaviour:

If the system detects that the user's organisation uses SSO, instead of showing the password field, a notification appears informing them:

'Your account requires SSO login. Redirecting to your organisation's authentication provider.'

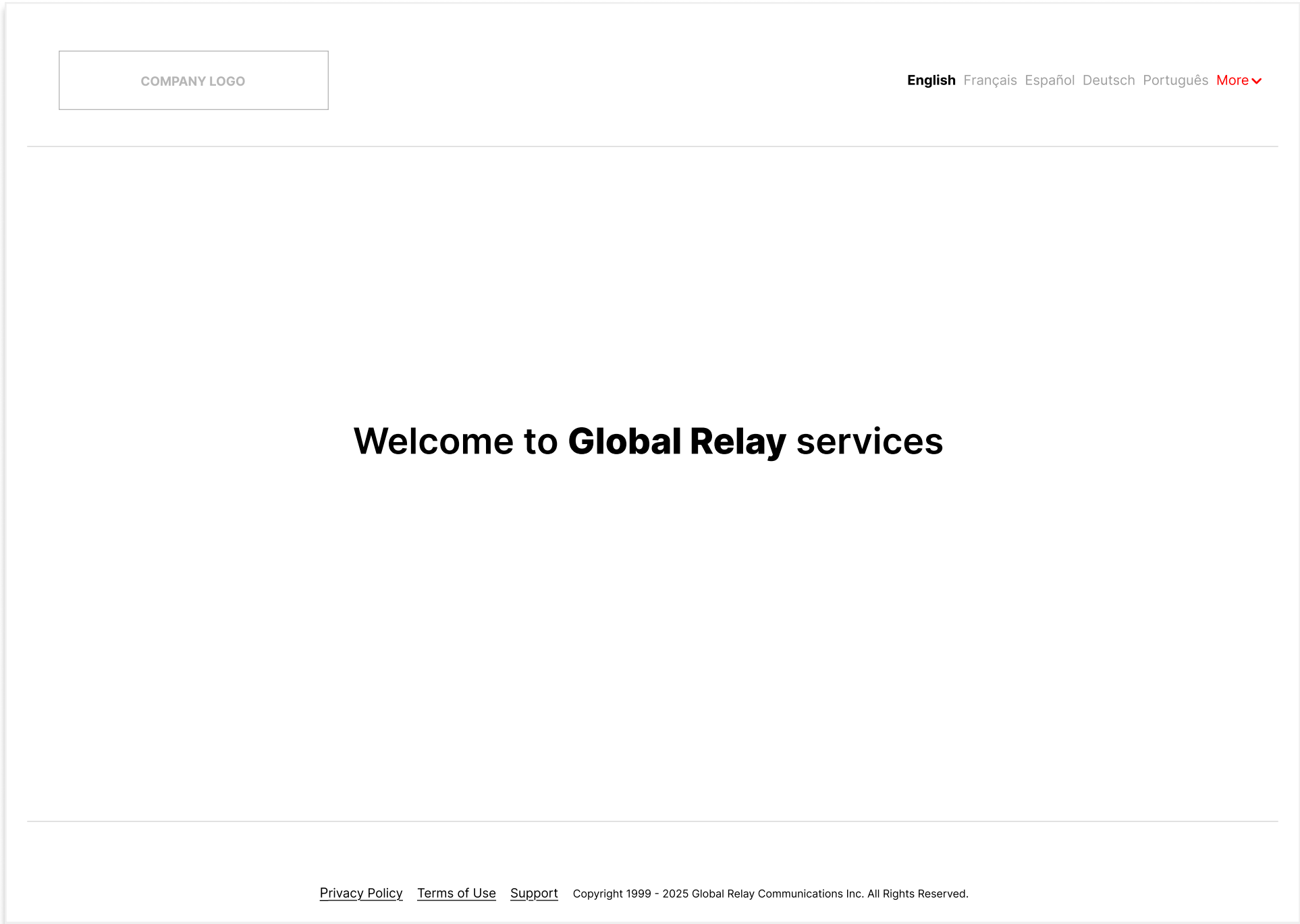
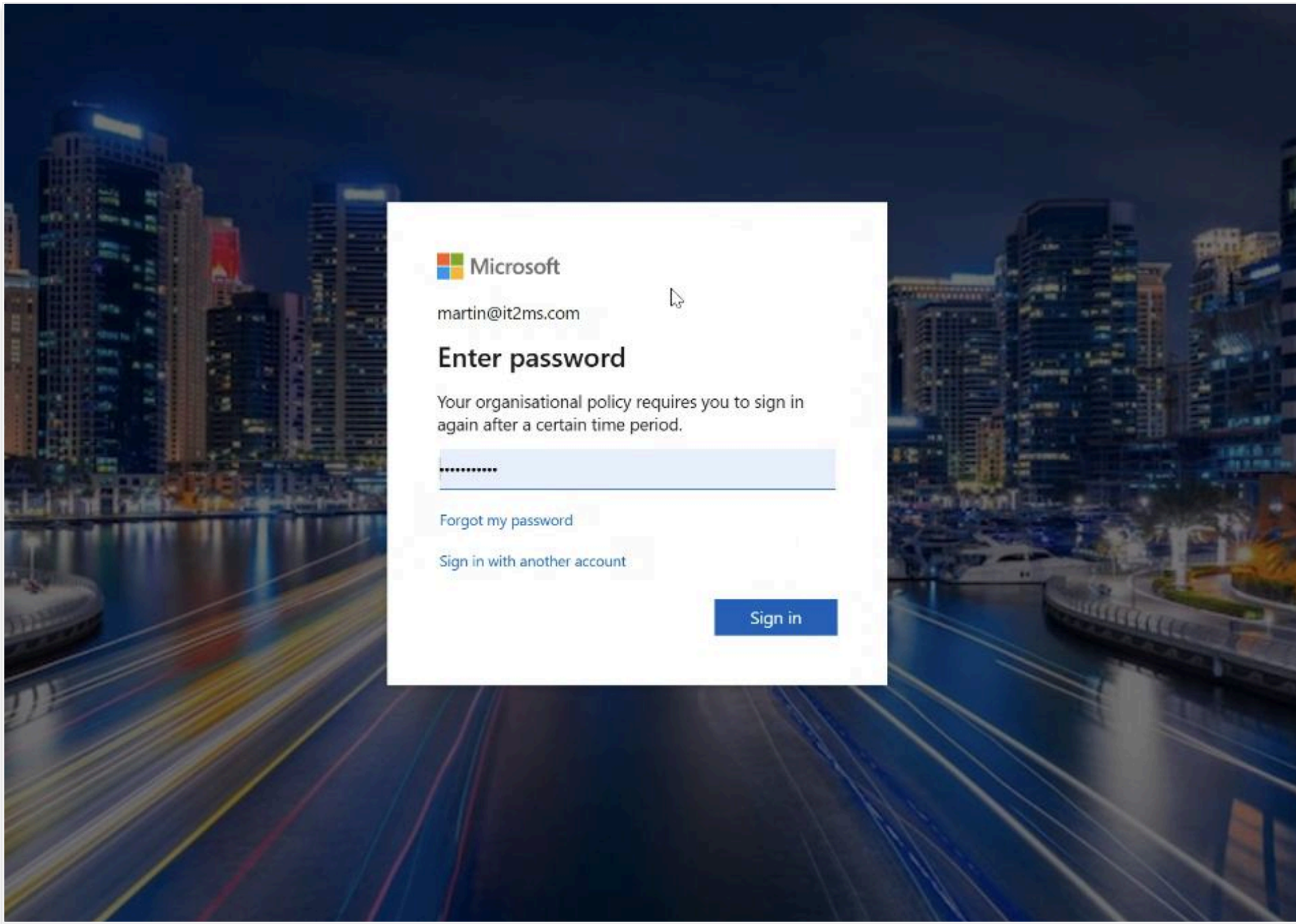
The Continue button redirects the user to the external SSO login provider.



Redirected to
customer SSO

Deliverables - SSO Happy path

Login Landing Page (Desktop)



Redirection to Customer's SSO Provider

The user is taken outside of Global Relay to their company's authentication page (e.g., Microsoft, Okta, Google Workspace).

They enter their password (or use passkeys, MFA, or biometric login if required by their company).

Authentication Completion & Return to Global Relay

Once successfully authenticated, the SSO provider notifies Global Relay, and the user is redirected back to the service they were originally trying to access.

Welcome Screen

- Unlike the password login flow, the user never sees a password input field within Global Relay.
- After authentication, they are automatically signed into all Global Relay services due to the single-login paradigm.

Deliverables - SSO Happy path

Login Landing Page (mobile)

English ▾

COMPANY LOGO

Secure access to all your
Global Relay services
with a single login!

Please enter your Login ID

Use your registered email or username.

Continue

☐ Remember login ID [Forgot password](#)

Global Relay Cookie Policy ▾

[Privacy Policy](#) [Terms of Use](#) [Support](#)

Copyright 1999 - 2025 Global Relay Communications Inc. All Rights Reserved.

English ▾

COMPANY LOGO

Secure access to all your
Global Relay services
with a single login!

Please enter your Login ID

jagojames@gmail.com

Continue

☐ Remember login ID [Forgot password](#)

Your account requires SSO login. Redirecting to your organisation's authentication provider.

Global Relay Cookie Policy ▾

[Privacy Policy](#) [Terms of Use](#) [Support](#)

Copyright 1999 - 2025 Global Relay Communications Inc. All Rights Reserved.

18:03 32m

Microsoft Azure

Microsoft

← jagojames@gmail.com

Enter password

Password

[Forgot my password](#)

Sign in

login.microsoftonline.com

English ▾

COMPANY LOGO

Welcome to
Global Relay
services

[Privacy Policy](#) [Terms of Use](#) [Support](#)

Copyright 1999 - 2025 Global Relay Communications Inc. All Rights Reserved.

Error handling

Failed Login & Account locked

COMPANY LOGO

EnglishFrançaisEspañolDeutschPortuguêsMore

Secure access to all your Global Relay services with a single login!

Logging out will sign you out of all Global Relay services.

Please enter your password

Login failed

Sign in

☐ Remember login ID

[Forgot password](#)

Global Relay Cookie Policy

[Privacy Policy](#) [Terms of Use](#) [Support](#) Copyright 1999 - 2025 Global Relay Communications Inc. All Rights Reserved.

If the user enters an email that is not registered, we hide this information for security reasons, we need to balance security and user experience (UX).

COMPANY LOGO

EnglishFrançaisEspañolDeutschPortuguêsMore

Secure access to all your Global Relay services with a single login!

Logging out will sign you out of all Global Relay services.

Please enter your password

Locked account due to multiple failed attempts.

Contact your administrator for help [Learn more](#)

Sign in

☐ Remember login ID

[Forgot password](#)

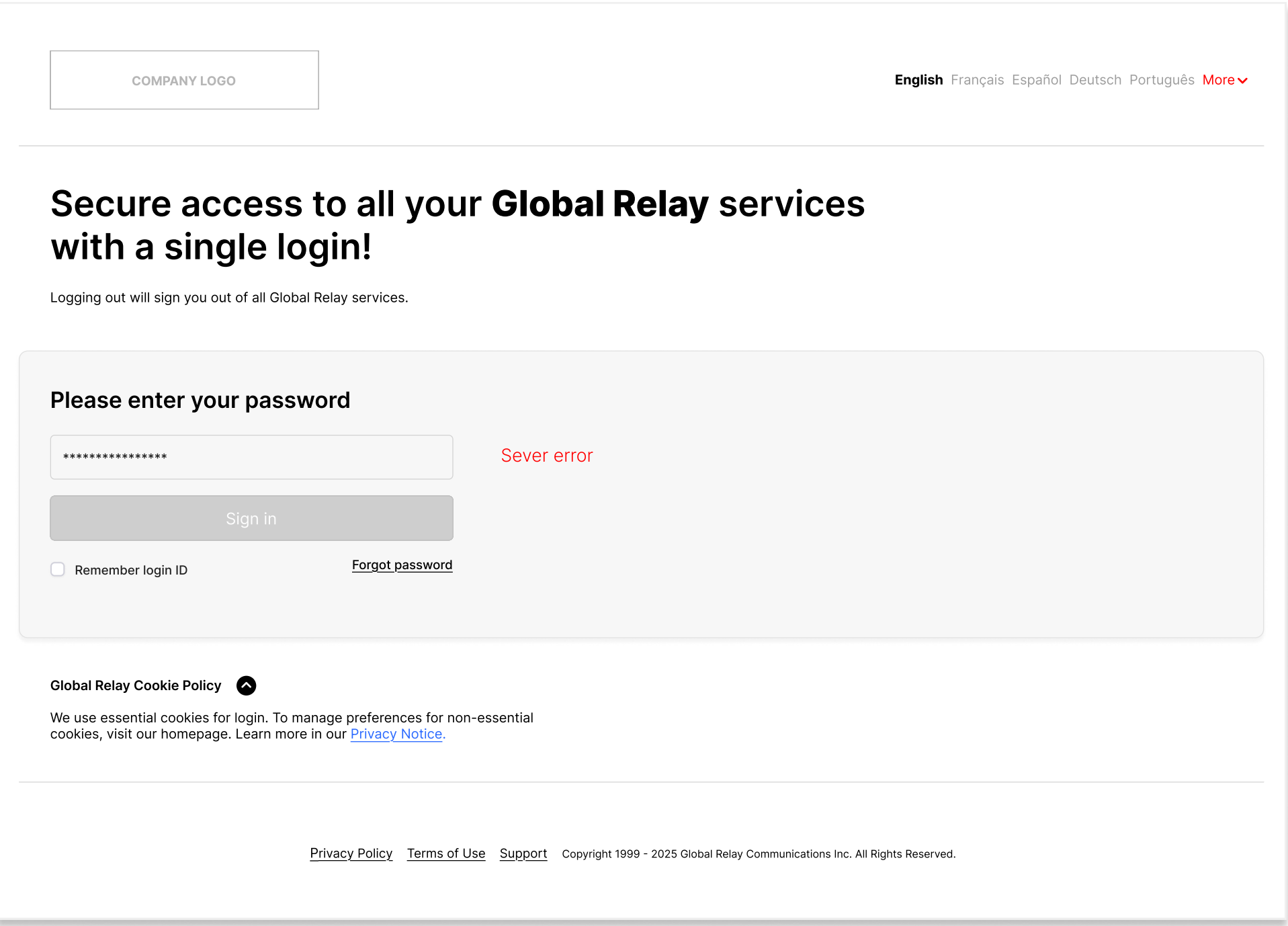
Global Relay Cookie Policy

[Privacy Policy](#) [Terms of Use](#) [Support](#) Copyright 1999 - 2025 Global Relay Communications Inc. All Rights Reserved.

To keep accounts secure, we temporarily lock them after multiple failed login attempts. This helps prevent attackers from guessing passwords, protects user data, and alerts account holders to any suspicious activity.

Other Errors

Sever error, login, authentication, security, and system errors



SSO login works - But Global Replay have revoked the license

Technical issue on the auth side (Only show user login failed to hide information that could be useful to a hacker).

Login & Authentication Errors

- Invalid username or password. Please try again.
- Your account has been temporarily locked due to multiple failed login attempts. Try again later or reset your password.
- Multi-factor authentication (MFA) required. Please verify your identity.
- MFA verification failed. Please enter the correct code or try another method.
- Your session has expired. Please log in again.

Security & Access Errors

- Unauthorised access attempt detected. Your session has been terminated.
- Your account does not have the required permissions to access this resource.
- Suspicious activity detected. Additional verification is required.
- Your IP address has been blocked due to repeated failed login attempts.
- Access denied. Please contact your administrator.
- Your account has been disabled. Contact support for assistance.

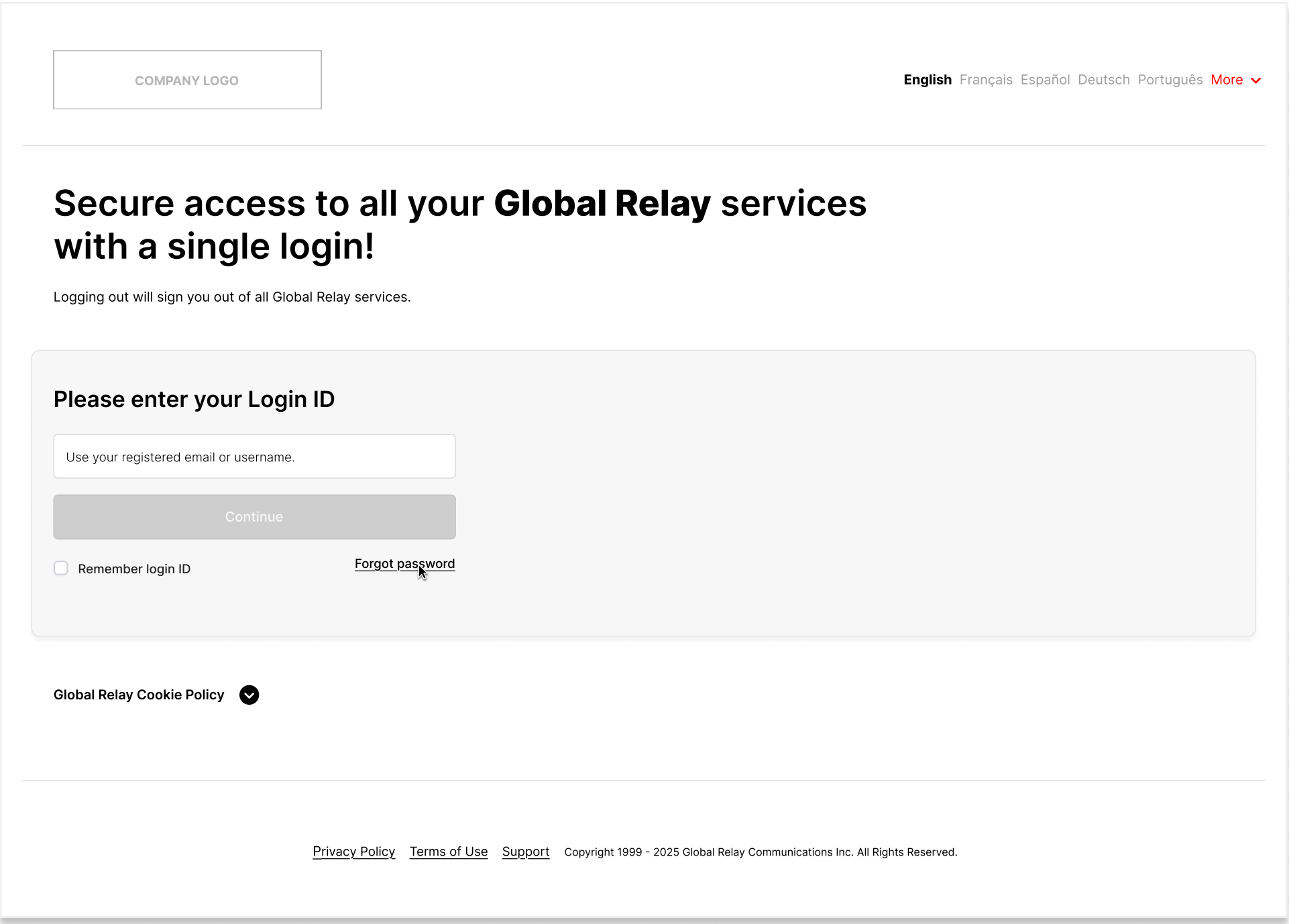
Data & System Errors

- Error retrieving data. Please try again later.
- Connection timeout. Unable to fetch requested information.
- An unexpected error occurred. Please refresh the page or contact support.
- System maintenance in progress. Some services may be unavailable.
- File upload failed. Please check the file format and try again.
- Data integrity check failed. Possible corruption detected.

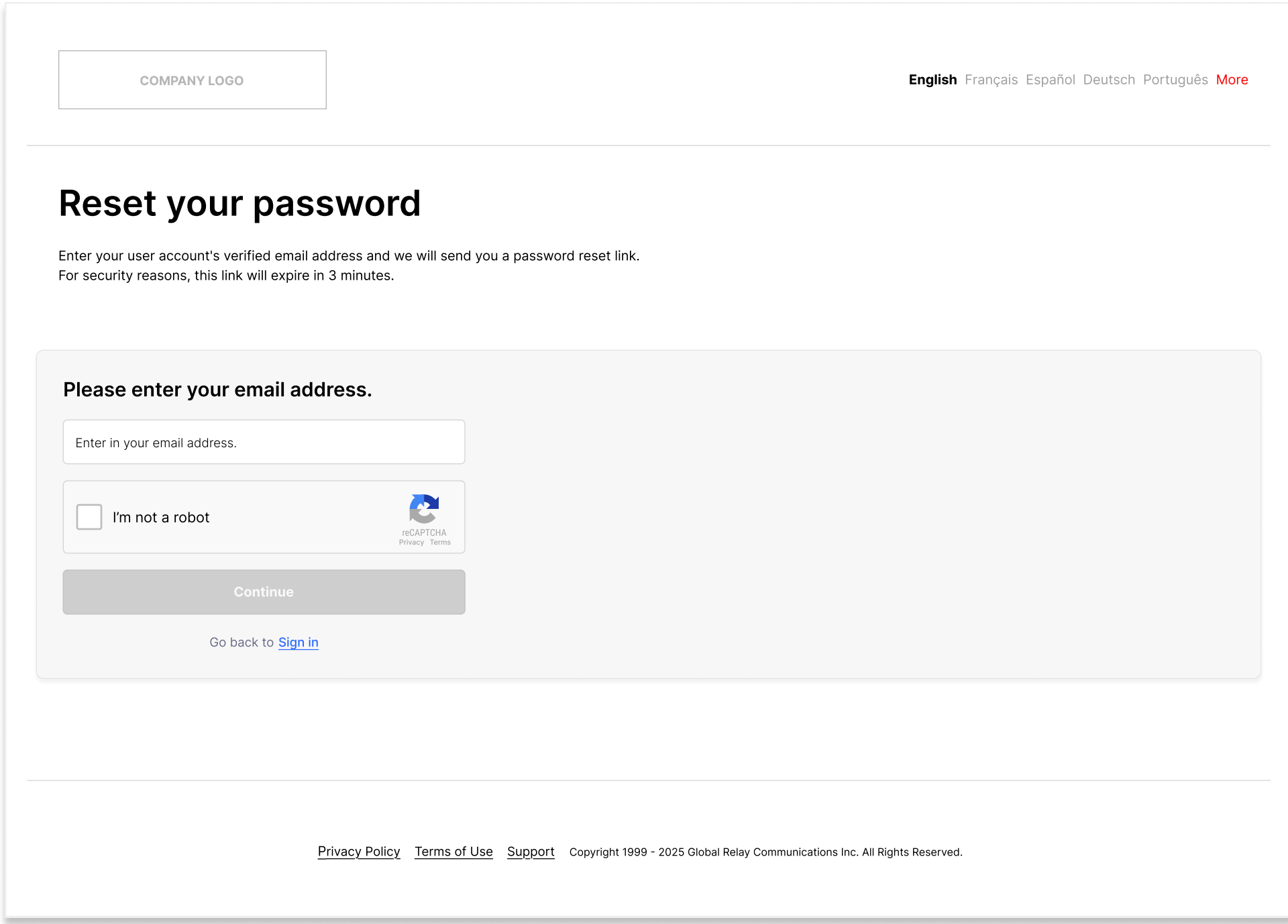
Synchronisation & Session Errors

- Your session has timed out due to inactivity.
- You have been logged out because your account was accessed from another device.
- System synchronisation error. Please refresh and try again.
- Session expired. Please log in again to continue.

Password reset



The user starts on the Login Page and clicks “Forgot password” instead of entering credentials. This action redirects them to the password reset flow. This keeps the design consistent and allows for progressive disclosure. For a clean UI the form only shows necessary fields, reducing cognitive load.



The user is prompted to enter their email address to receive a reset link. A CAPTCHA verification is included to prevent automated bots from abusing the password reset function. Clicking “Continue” (once activated) sends the request.

A "Go back to Sign In" link provides users with an easy way to return to the login page if they selected "Forgot Password" by mistake.

Password reset

COMPANY LOGO

EnglishFrançaisEspañolDeutschPortuguêsMore

Reset your password

Enter your user account's verified email address and we will send you a password reset link.
For security reasons, this link will expire in 3 minutes.

Please enter your email address.

☒ I'm not a robot


reCAPTCHA
[Privacy](#) [Terms](#)

Continue

Go back to [Sign in](#)

[Privacy Policy](#) [Terms of Use](#) [Support](#)

Copyright 1999 - 2025 Global Relay Communications Inc. All Rights Reserved.

User Behaviour
The "Continue" button is disabled until both fields are filled to prevent accidental submissions.

COMPANY LOGO

EnglishFrançaisEspañolDeutschPortuguêsMore

Check Your Email

If an account with this email exists, we've sent a password reset link.
Please check your inbox and follow the instructions.
The link will expire in 3 minutes

Didn't receive the email? [Resend link](#)

Go back to [Sign in](#)

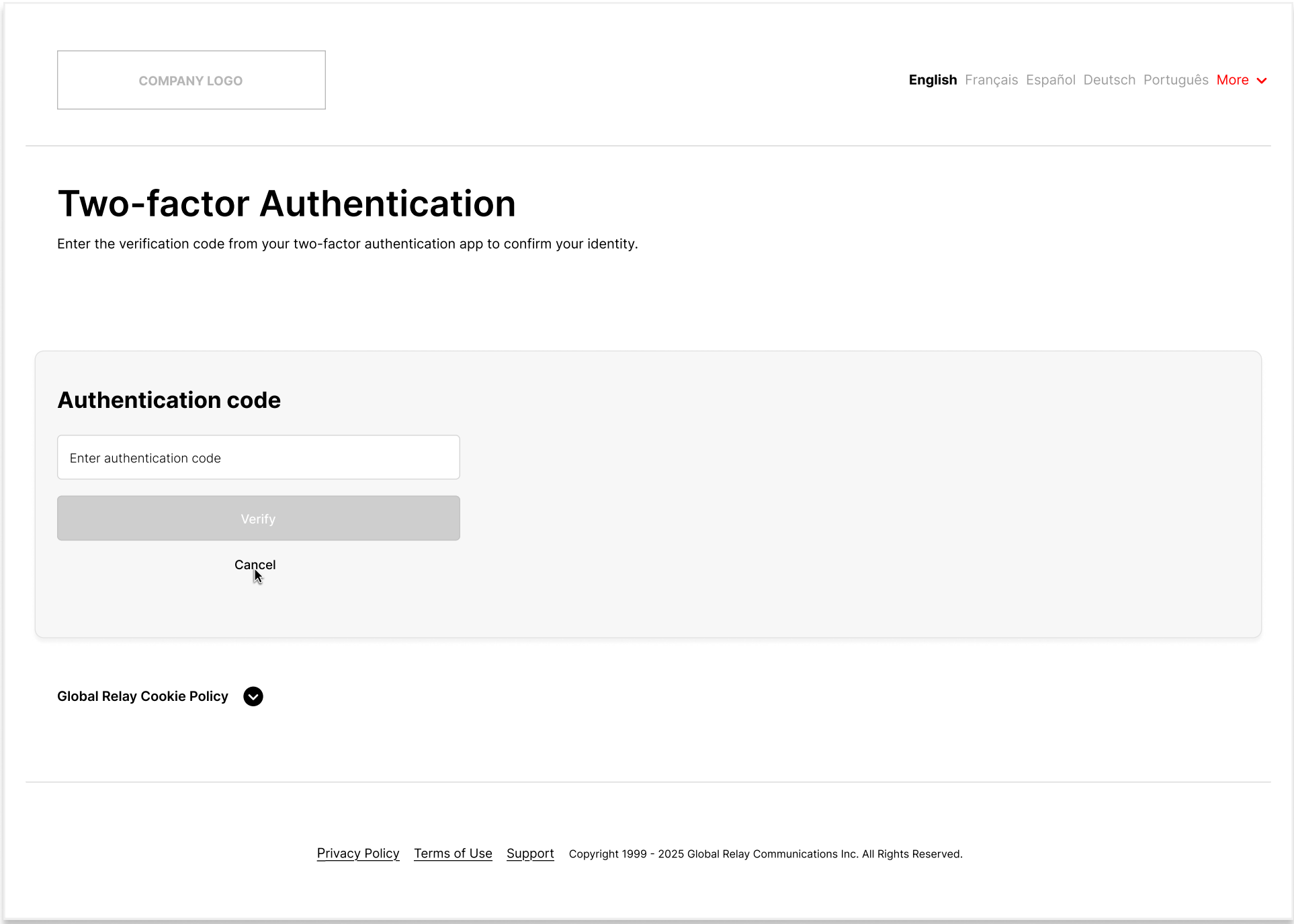
[Privacy Policy](#) [Terms of Use](#) [Support](#)

Copyright 1999 - 2025 Global Relay Communications Inc. All Rights Reserved.

The user is shown a confirmation message informing them that if their email is registered, they will receive a reset link.

The reset link expires in 3 minutes, reinforcing security and urgency.
A “Resend Link” option is provided if the user does not receive the email.

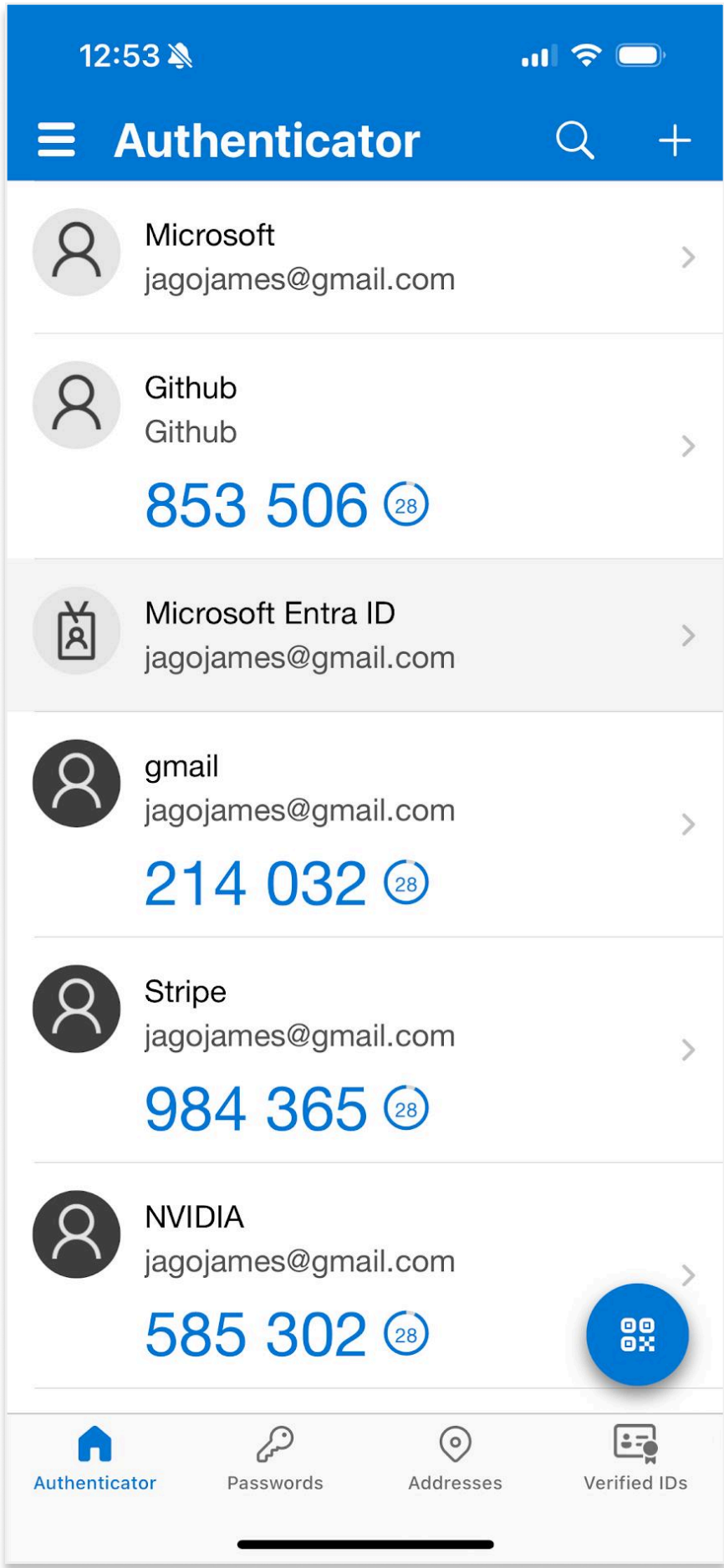
Two-factor Authentication



The (2FA) process enhances the login security by requiring a secondary verification step. The user opens their authenticator app on their phone.

A time-sensitive, rotating 6-digit code is displayed for each service linked to the user’s account.

The user copies or manually enters the code into the Global Relay 2FA screen.



After successfully entering their Login ID and Password, the user reaches the Two-Factor Authentication (2FA) screen.

- They are prompted to enter the authentication code from their authenticator app (e.g., Microsoft Authenticator, Google Authenticator).
- The “Verify” button remains disabled until a valid 6-digit code is entered.
- A “Cancel” button allows users to return to the login screen if needed.

Key UX & Security Considerations

Security: Stronger Authentication

- Prevents unauthorised access, even if a password is compromised.
- Ensures compliance with security best practices in enterprise environments.
- Codes expire quickly (typically within 30 seconds) to prevent reuse.

Usability & Accessibility

- Simple interface with clear instructions to reduce friction.
- Disabling the "Verify" button until input is valid prevents errors.
- “Cancel” button provides an escape route if the user is unable to retrieve the code.

Error Handling & Edge Cases

- If an incorrect code is entered, display an inline error message rather than clearing the input field.
- If the user loses access to their authenticator app, provide an alternative login recovery option (e.g., backup codes, SMS 2FA).

Tone of voice - UX writing

Tone of voice - UX writing

Digital experiences are made clear, simple, and frustration-free by UX writing. It guarantees uniformity across apps, enhances accessibility, and provides user guidance. Confusion is avoided, a reliable tone is maintained, and users can easily recover from mistakes when UX writing is done well. Clear microcopy in confirmations, tooltips, and buttons encourages action and improves usability for all users, including those with disabilities.

Formal & Clear

Enter the verification code from your two-factor authentication app to confirm your identity.
To proceed, input the code from your authentication app.

Concise & Direct

Enter the 2FA code from your app.
Input your authentication code to continue.
Check your 2FA app and enter the code provided.

Friendly & Supportive

We need to verify it's you! Enter the code from your authentication app.
For added security, please enter the code from your 2FA app.
Your verification code is in your authentication app. Type it here.
Would you like a version specific to SMS-based 2FA or backup codes?

These three example showcases three different tones for 2FA instructions, each shaping user experience differently. Formal & Clear is professional and precise, ideal for corporate settings. Concise & Direct keeps it minimal and actionable for experienced users. Friendly & Supportive is warm and reassuring, great for consumer-friendly apps. The choice of tone influences perception, aligning with brand identity and user needs.

Usability Testing

(Before Launch)

Goal: Identify usability friction points and refine the login experience.

Methods:

Initial feedback

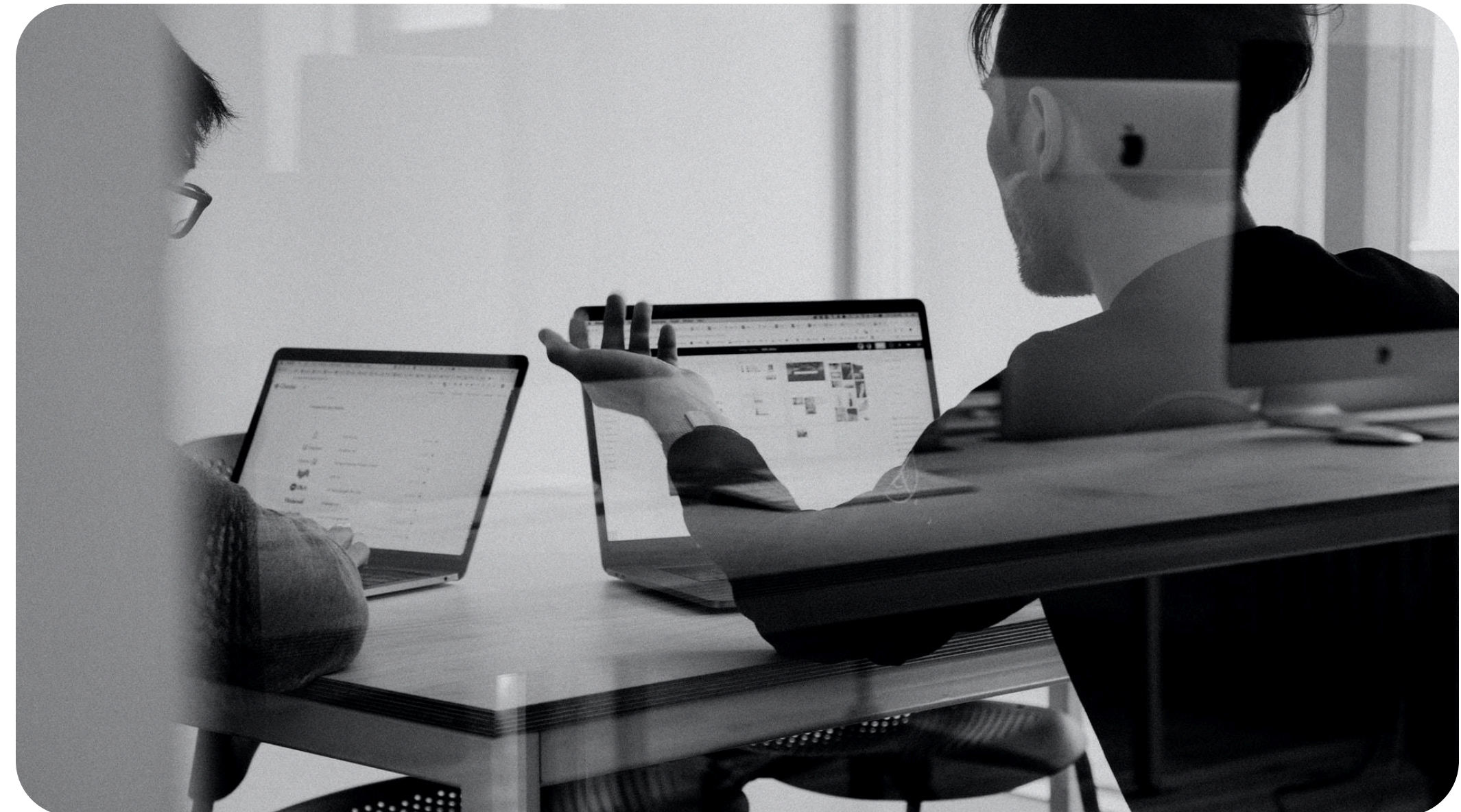
- Gain internal feedback from customer support team and other stakeholders.

Moderated User Testing

- Conduct live testing sessions with a mix of new and returning users.
- Observe their reaction to different login flows (e.g, standard login vs. SSO).
- Ask participants to verbalise thoughts while logging in.

Unmoderated User Testing

- Use platforms like Maze or UserTesting.com to collect feedback asynchronously.
- Give users specific tasks (e.g., “Reset your password” or “Log in with SSO”).
- Track time taken, errors, and confusion points.



Usability Testing

(Before Launch)

Goal: Compare different login flow variations to determine the best-performing UX.

Example Tests

A/B Test 1: Passwordless Login Options

- Version A: Default password login.
- Version B: Magic link + Passkeys (WebAuthn).
- Metric: Reduction in failed login attempts, password reset requests.
(Not wired this, but worth a thought)

A/B Test 2: Error Message Clarity

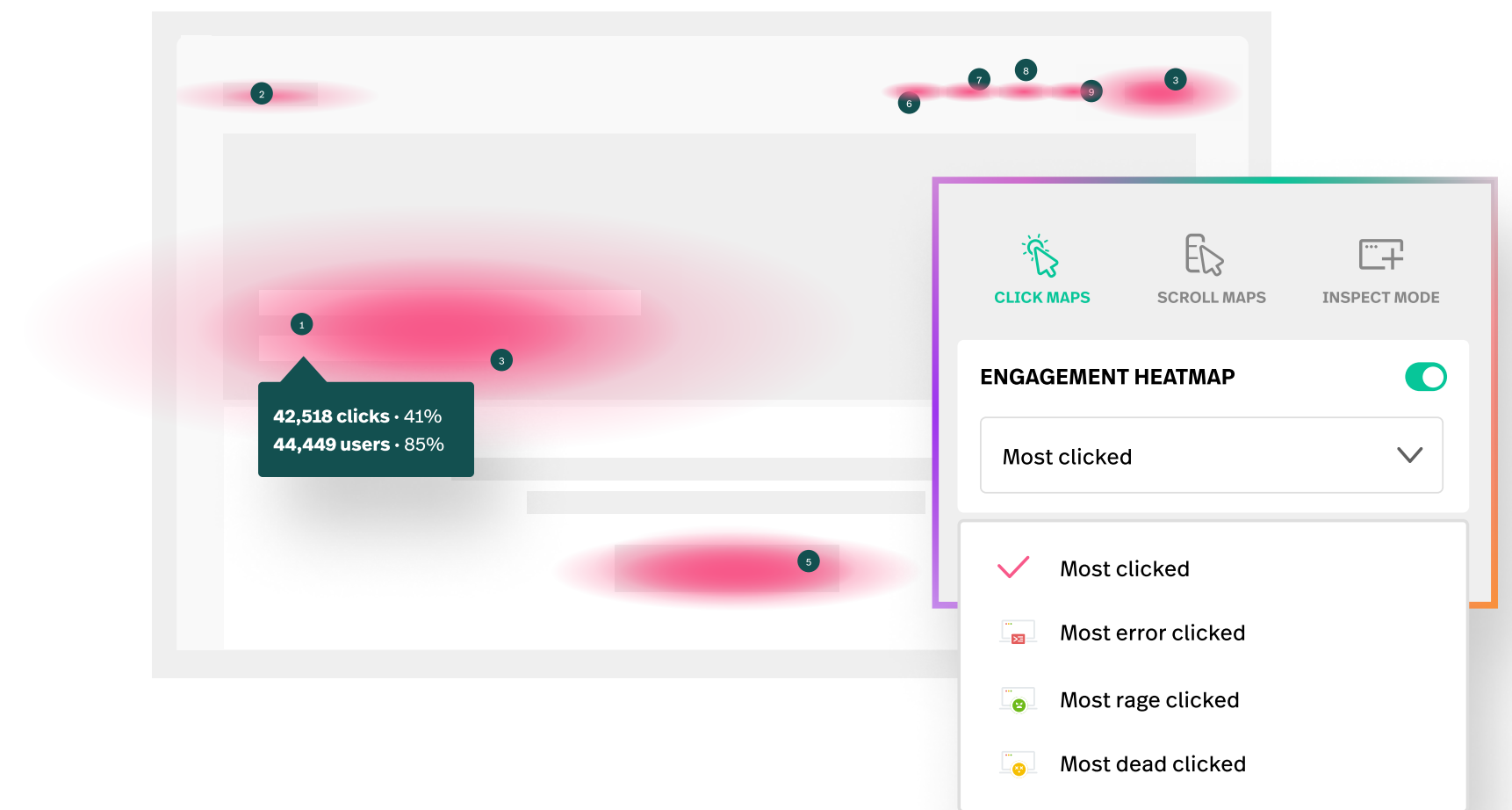
- Version A: Generic error message (“Login unsuccessful.”).
- Version B: Specific guidance (“Incorrect password. Try again or reset it.”).
- Metric: Reduction in repeated failed login attempts.

A/B Test 3: Session Expiry Experience

- Version A: Immediate logout after timeout.
- Version B: Warning modal before auto-logout (“Your session will expire in 2 minutes”).
- Metric: Drop-off rate after session expiry.

Session Recordings & Heatmaps

- Use tools FullStory to capture real user interactions.
- Identify where users hesitate, drop off, or struggle.



Accessibility Considerations

Clear Text Labels & Instructions

- Each step in the flow has concise and clear instructions, reducing confusion.
- Example: “Enter your user account’s verified email address and we will send you a password reset link.”

Readable, High-Contrast Text

- The black text on a white background provides strong contrast for readability.
- Ensures compliance with WCAG contrast ratio requirements (4.5:1 for normal text).

Logical Flow & Progressive Disclosure

- The flow is broken into clear steps, preventing cognitive overload.
- Users only see what they need at each step rather than being overwhelmed with multiple inputs at once.

Captcha with Checkbox Instead of Image Challenges

- Uses reCAPTCHA with a simple checkbox, which is more accessible than text/image-based captchas.
- This reduces barriers for users with low vision or cognitive disabilities.

Keyboard Navigability

- The email input field, captcha checkbox, and submit button appear in a logical sequence.
- Likely supports tab navigation for users who rely on keyboards instead of a mouse.

Alternative Contact Path (Resend Link)

- If users don’t receive an email, there’s a “Resend link” option.
- Helps users who may have missed the email or have filtering issues in their inbox.

Screen Reader Optimisation

- Ensure proper ARIA labels and live region announcements for status updates.
- Example: When users submit an email, a screen reader should announce: “If an account with this email exists, a password reset link has been sent.”

Multiple Reset Options

Passwordless authentication like a magic link is great for users who struggle to type or have dyslexia
Also good for users who are locked out of their email

Visual Indicators for Actions

- Make the “Continue” button visually change when active (e.g., background colour darkens).
- Helps users with low/poor vision or colour blindness recognise when they can progress though the journey.

Session Timeout Messaging

- If the password reset link expires in 3 minutes, proactively warn users before they submit their request.
- Example: “You will have 3 minutes to reset your password once you receive the email. Need more time? Request a new link after submission.”



That's all Folks!