**Project Requirement Analysis**

**Project: Lutron**

**FEATURE:  Providing The Troubleshooting Feature on Login Page**

**Objective**

To improve user experience, the email password troubleshooting feature on the login page aims to facilitate a streamlined and user-friendly resolution process for individuals encountering challenges with their login credentials. By implementing this feature, we intend to empower users to efficiently troubleshoot. The primary goal is to enhance overall user satisfaction and engagement by offering a reliable and intuitive mechanism for Troubleshooting login-related issues.

**Functional Requirements:**

**Link for Troubleshooting**

* A **clickable troubleshooting link** will be added to the login page.
* The link will be **positioned just below** the login button.
* On Clicking the troubleshooting link will **open a popup**.

**New popup Content**

* The **popup** will provide suggestions regarding the correctness of **login credentials** syntax, as demanded by the system.
* It will display a suggestion prompting users to verify the reachability and proper functioning of **backend services**.
* A user-friendly message will advise checking the **internet connection** to ensure it is functioning properly.
* The **popup** will also highlight the importance of **reviewing password policies** for compliance.
* Add a button to the **customer support** interface for users if the problem has **not been resolved** yet.
* Include a close button at top right corner of the **popup**.

**Non-Functional Requirements:**

* **User Interface:** Troubleshooting with a design that **fits app's style**, avoiding unnecessary animations.
* **Performance:** Set specific goals for how **fast the Troubleshooting page** should work. This could include how quickly it loads or how fast it responds to user actions.
* **Cross-browser Compatibility:** Ensure that the Troubleshooting page functions effectively on **the most used web browsers** (chrome, Microsoft Edge etc) by application users. Instead of claiming universal compatibility.

**Technical considerations**

**Cross-Browser Compatibility:** Ensure that the Troubleshooting page functions effectively on the **most used web browsers** (chrome, Microsoft Edge etc) by application users. Instead of claiming universal compatibility.

**Selecting the Framework:** The implementation Part can be done with **any frontend framework**.

**Secure Communication (HTTPS):** Ensuring secure communication **through HTTPS** is paramount to protect user credentials during the troubleshooting process. It encrypts data in transit, preventing unauthorized access.

**Risks and Mitigations**

**Risk**:

* Unauthorized access or data breaches during the troubleshooting process.
* Delay to open Troubleshooting Page

**Mitigation:**

* Implement rate limiting to prevent brute force attacks. Provide clear instructions for users on password recovery.
* Optimize code and perform testing.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Signature** | **Date** |
| Varun Bhatia | Reviewer |  |  |
| Varun Bhatia | Lead |  |  |
| Prabhash Sharma | Developer | Ptabhash | 24-01-2024 |