Ian Rule, Biswas Gauli, Kaleb Pittman, Nikita Feoktistov
Professor Alan Paradise, TA Anna Glass
CSCI 4308-800

Unpluq Requirements Document

Functional Requirements (Ian Completed)

The numbered requirements below reflect "what" we want the browser extension we develop to contain in terms of features from the user's viewpoint.

MVP 1

1. A scheduling page where users can create new productivity schedules for blocking websites. (Use Case Example: an Unpluq premium subscriber goes to the scheduling page to create a new blocking schedule to block 7 websites on Monday through Friday from 9:00 AM to 5:00 PM)
    a. Include a creation form on the page for users to fill out.
        i. "Name of Schedule" input field.
        ii. "Which Days" input field.
        iii. "What Time" input field.
        iv. "Websites to Block" input field.
2. A "blocked" screen to display for users when they try to access a blocked website. (Use Case Example: an Unpluq subscriber tries to access YouTube in Google Chrome while they have a blocking schedule active, but instead of loading YouTube the "blocked" screen is displayed)
    a. Include the "blocked screen" visual.
    b. Include the name of the website the user tried to access.
    c. Include an unblock button for the user to utilize if they wish to unblock the site.

MVP 2

1. A toolbar menu page where users can access Unpluq features when clicking on the browser extension toolbar icon. (Use Case Example: an Unpluq subscriber wishes to quickly unblock a website that is currently blocked, so they do so by clicking the Unpluq toolbar icon and hitting the Unblock button beside that website)
    a. Include a currently blocked card.
    b. Include an unblock button near the currently blocked card.
    c. Include a schedule icon.
    d. Include a settings/more icon.

2. A second version of the scheduling page where users can see an overview of their current schedule(s) and edit their current schedule(s). (Use Case Example: an Unpluq premium subscriber goes to the scheduling page to edit see the two schedules they currently have active, edits one of the schedules, and turns the other schedule off)
    a. Include a “schedule overview” module.
    b. Include a “schedule editing” module.
3. A “more” page where users can access browser extension settings. (Use Case Example: an Unpluq subscriber wishes to edit the duration of their app block timer)
    a. Include a “settings” module.
        i. Include an “app block timer” option inside of the settings module.

MVP 3

1. An “onboarding” page where users can login in to their Unpluq account or set up a new account. (Use Case Example: a new user comes across the Unpluq browser extension for the first time, so the onboarding page is displayed with an option for them to sign up for Unpluq)
    a. Include a functional login flow.
    b. Include a functional onboarding/registration flow.
2. A second version of the toolbar menu page where users can quickly block the current website they are on by clicking the browser extension toolbar icon. (Use Case Example: an Unpluq subscriber wishes to block the current website they are on, so they click the Unpluq toolbar icon and select the new “block this website” button)
    a. Include a “block this website” button.
3. A second version of the “more” page where users can access even more browser extension settings, profile and subscription settings, and support resources. (Use Case Example: an Unpluq subscriber believes their account login information may have been compromised, so they navigate to the “more” page where they can change their password in profile and subscription settings and send feedback to the Unpluq team)
    a. A second version of the “settings” module.
        i. Include an “emergency mode” option inside of the new settings module.
    b. A “profile and subscription” module.
        i. Include an “account settings” opinion inside of the module.
        ii. Include an “Unpluq premium” option inside of the module.
    c. A “help” module.
        i. Include a “send feedback” option inside of the module.
        ii. Include a “FAQ” option inside of the module.
        iii. Include a “tutorial” option inside of the module.
        iv. Include a “privacy policy” option inside of the module.
        v. Include an “about Unpluq” option inside of the module.

Non-Functional Requirements (Kaleb Completed)

The numbered requirements below reflect "how" we want the final browser extension we develop to function and be maintainable from our viewpoint (the developers).

1. Branching/Project Organization
    a. GitHub repository branches will be labeled by the secion of development the branch will fall under.
        i. Bug fixing branches will be titled "bug-fix/…".
        ii. Feature branches will be titled "feature/…".
        iii. UI update branches will be titled "ui/…".
        iv. Refactoring branches will be titles "refactor/…".
    b. Weekly status reports will record ongoing project efforts and conflicts.
    c. Project charter will record the project's scope and its potential revisions.
2. Scalability
    a. The extension will be able to accept increasing numbers of users and user requests without a decline in performance.
        i. The infrastructure selected for handling app-extension communications will meet Unpluq's reliability standards, ensuring the extension will be able to handle many users.
    b. Schedule creation will be standardized and stored in a database capable of handling a large amount of user's personalized schedules.
    c. Extension will be available as a Google Chrome extension (and Apple Safari, time permitting).
        i. As a Chrome extension, the application will look and function similarly for all users (and the devices they happen to be running Chrome on).
3. Interfacing Requirements
    a. Inputted blocking times are internally stored as 24 hour time strings, e.g. "17:00" for 5 PM.
    b. Requests to unblock will be processed at the user's input.
        i. These requests will traverse the app-extension infrastructure selected by the Unpluq team, and are expected to use AWS for two way communication between the app and the extension.
    c. Upon creation, schedules are stored in the following way:
        i. Schedule data will include start and end time for blocking, stored in military time "hh:mm" format, e.g. "09:00" or "17:00".
        ii. Schedule data will include a barrier 'type', which will default to requiring an Unpluq Tag to be scanned on the application.

iii. Schedule data will include the barrier ‘difficulty’, referring to how hard it is to unblock websites.
iv. Schedule data will include a list of websites to block.
v. Days to block.
vi. Time to block.

4. User Compliance
    a. User will be prompted to log in to their Unpluq account.
        i. This will allow the user to unblock websites using only instances of the Unpluq app that are logged into their personal account.
        ii. This ensures that user schedules and data are private and only editable by the associated account.
        iii. This will also allow a user to preserve schedules (e.g. if the user logs in from a different device, their schedules can automatically be loaded)

Change Control Process (Ian Completed, Kaleb Completed)

<u>Steps for Project Requirement Changes</u>

1. The desired changes to the project requirements document must be formalized in writing, and sent to the project team for review over Slack or email.
2. The project team is given up to 48 hours to review the desired changes and reserves the right to accept, reject, or negotiate the desired changes.
3. Once the project team has come to a decision regarding the proposed changes, they will formalize a response in writing and send it to all associated parties.
4. If necessary, revisions to the requirements document will be completed by the project team and approved by Unpluq.
    a. The previous version of the requirements document is to be stored in the project team’s shared repository, as is the new requirements document.

Sponsor Sign-Off

Jorn Rigter - (15 Nov 2023)