ChatGPT UI/UX ENHANCEMENT – MILESTONE 2 REPORT

Project Overview

Team Members: Senthil Ganesh P (241110089), Khuswant Kaswan (241110035), Divyanshu (241110023)

Roles of Team Members

Senthil Ganesh P (241110089) - Design and analysis of the user survey. Will manage usability testing, analyze feedback, and ensure user needs are reflected in the final design.

Khuswant Kaswan (241110035) & Divyanshu (241110023) - Translate user insights into interface improvements. Responsible for wireframes, visual layout, and ensuring designs are user-friendly and accessible. Building interactive prototypes and managing task flows. Oversee implementation of design ideas into testable formats and ensure milestone deadlines are met.

Refined Scope: Our project aims to improve ChatGPT's web user interface and user experience by creating a *prototype* that addresses key usability gaps identified in its current design. In **Milestone 1**, we proposed to investigate ChatGPT's existing UI/UX for pain points using needfinding techniques. Based on initial findings, we refined our scope to focus on critical interaction issues rather than cosmetic changes. Specifically, we shifted attention toward features that enhance usability – such as better navigation in long chats, preventing accidental message sends, and allowing easy export of conversations – since users indicated these as high-priority needs. Minor aesthetic improvements (e.g. theme colors or font tweaks) were de-emphasized because the majority of users reported being satisfied with the default dark theme and font settings. This refined focus ensures that our efforts concentrate on improvements that will meaningfully enhance the user experience.

Needfinding Results

Methodology: For this milestone, we conducted a **user survey** as our primary needfinding method (with follow-up interviews planned in the next phase). The survey was designed to answer our research questions: *"What are the usability pain points in ChatGPT's current interface?"* and *"What new features or improvements do users most desire in ChatGPT's UI?"* We created an online questionnaire comprising both **closed-ended questions** (Likert-scale ratings and multiple-choice) and an **open-ended section** for suggestions. Key topics covered included: satisfaction with the **current interface design** (dark theme, font style/size), interest in potential new **features** (e.g. split-screen chat, navigation sidebar, export function, custom themes), and common **pain**

points during usage (such as accidentally sending messages). The google form link that was circulated among the users is given below:

https://docs.google.com/forms/d/e/1FAIpQLSdd7MukznPpxpoNVdl-62PQHrklj1SKFYP43sAp_LZUBNSiUQ/viewform?usp=header

Participants and Sampling: We collected responses from **159 ChatGPT users** over the last two weeks. Participants were recruited via **convenience sampling** – the survey was shared among student and professional networks known to use ChatGPT. The sample represents a mix of postgraduate students, MBBS students and early/mid level SDE career professionals who interact with ChatGPT almost everyday for various tasks (coding help, writing assistance, general Q&A). While not a strictly random sample of all ChatGPT users, this group provided diverse feedback across different use cases. The respondents are assumed to have familiarity with ChatGPT's interface, ensuring their feedback is grounded in real usage experience.

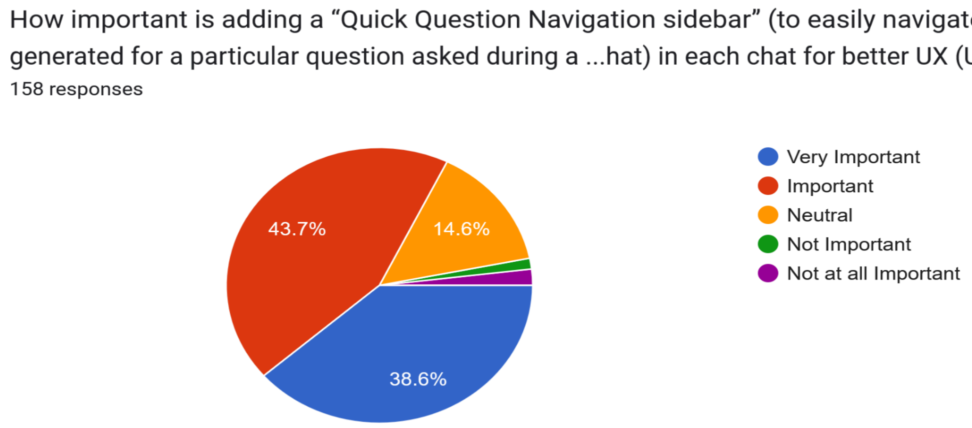
Data Collected: The survey gathered mostly **quantitative data** through rating-scale questions, yielding metrics on user satisfaction and feature importance, as well as some **qualitative insights** from free-text inputs. Example questions included: *"How satisfied are you with ChatGPT's dark theme?"* (5-point scale from Very Dissatisfied to Very Satisfied), *"How strongly do you recommend a split-screen feature?"* (5-point from Strongly Oppose to Strongly Recommend), and *"How frequently do you accidentally send a message by pressing Enter?"* (Never to Very Frequently). Additionally, a multi-select question asked users *"How would you improve ChatGPT's responses?"* offering options like adding colored text, adding images/emojis, etc., with an "Other (please specify)" for open-ended suggestions. This combination of question types allowed us to quantify the prevalence of certain needs and also capture **user-generated ideas** or specific complaints.

We ensured anonymity and encouraged honesty by not requiring personal identifiers (name was optional). The raw survey data (all responses) is provided in the project appendix for transparency. We plan to **triangulate** these survey findings with a handful of user interviews in the next phase, to dive deeper into the reasons behind the preferences and pain points identified. (As of Milestone 2, the needfinding results are drawn primarily from the survey; interviews are scheduled to be conducted before Milestone 3 to validate and enrich these findings.)

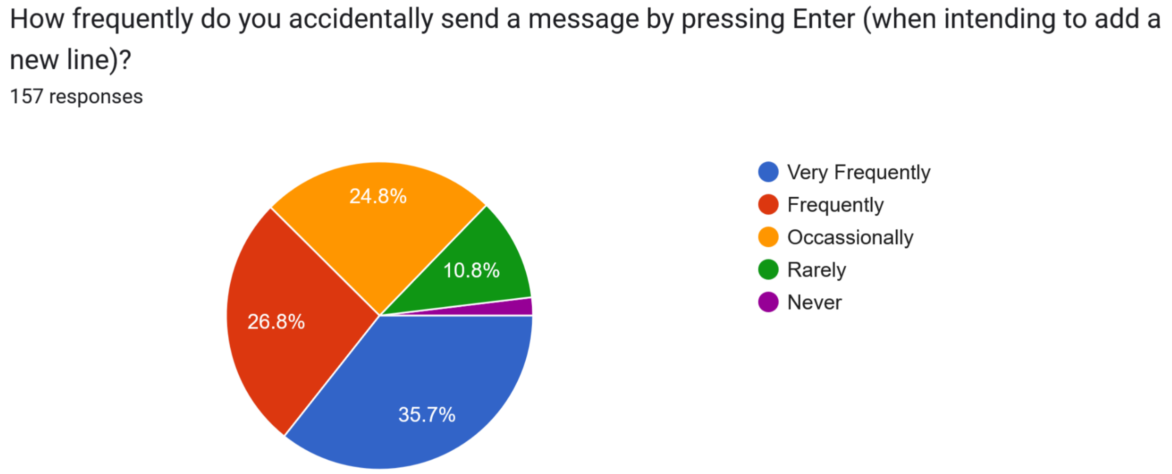
Key Findings – User Pain Points & Opportunities: The survey revealed several clear pain points and opportunities for enhancing ChatGPT's UI/UX. Below is a summary of the main insights supported by select statistics from the results:

1. **Difficult Navigation in Long Chats:** Users struggle to scroll through lengthy conversations to find past answers. **81%** of respondents rated a "quick question navigation

sidebar” as *Important* or *Very Important*, highlighting the opportunity for a feature to jump to specific questions/answers without manual scrolling.

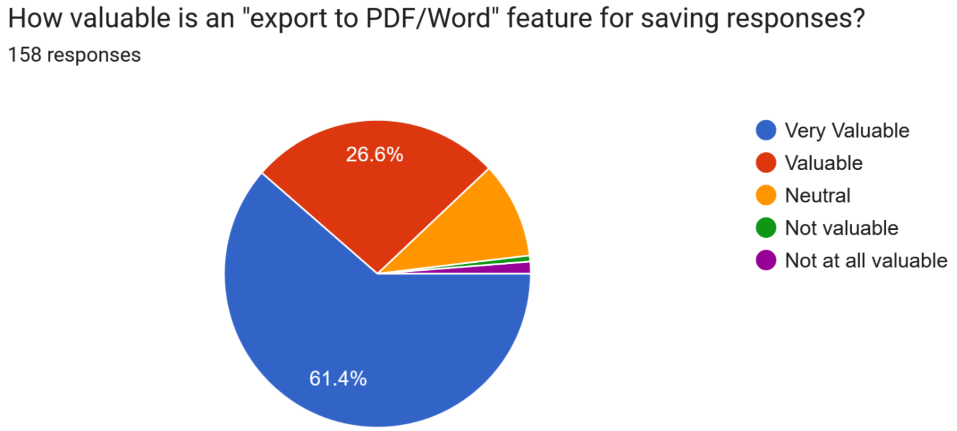

How important is adding a “Quick Question Navigation sidebar” (to easily navigate to the response generated for a particular question asked during a ...hat) in each chat for better UX (User Experience)?
158 responses

2. **Accidental Message Sending:** The current input box causes frequent mistakes where messages are sent unintentionally (e.g. pressing Enter trying to add a newline). About **63%** of users reported sending messages accidentally *“Frequently” or “Very Frequently”*, indicating widespread frustration with the message input design. This is a critical usability issue, as one participant noted wanting the ability to *“send multiple [lines] while texting… [without] pressing Enter”* prematurely (survey comment).

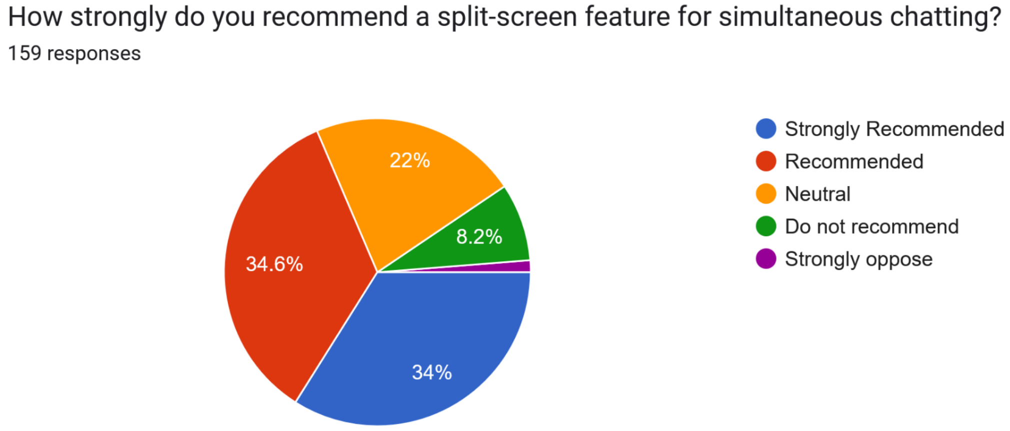

How frequently do you accidentally send a message by pressing Enter (when intending to add a new line)?
157 responses

3. **No Easy Way to Save or Export Chats:** ChatGPT currently lacks a built-in export function, which users find limiting for saving important conversations. An overwhelming **88%** of respondents consider an “export to PDF/Word” feature *Valuable* or *Very Valuable*.

This strong interest underscores an opportunity to implement an easy chat export/save option so users can keep records of useful dialogues.


How valuable is an "export to PDF/Word" feature for saving responses?
158 responses

4. **Desire for Multi-Tasking (Split-Screen):** Many users expressed interest in handling multiple chats or viewing information side-by-side. Roughly two-thirds of respondents (≈68%) *Recommended* or *Strongly Recommended* a **split-screen chat** feature for simultaneous chatting. This suggests users sometimes need to compare answers from two different chats or continue one conversation while referencing another, an experience currently not supported in the single-thread interface.


How strongly do you recommend a split-screen feature for simultaneous chatting?
159 responses

5. **UI Personalization and Visual Clarity:** There is moderate interest in more personalization of the interface. About 61% of users were in favor of a **custom theme/layout** option (with the remainder mostly neutral). Some users suggested adding visual elements to responses for clarity – for example, using *“colored fonts (green for*

positive, red for negative)" or including emojis/graphics to enhance understanding (multiple respondents selected these options in the survey). However, the core visual design is generally acceptable: most users are satisfied with the **dark theme** (73% satisfied or very satisfied) and **font style/size** (76% satisfied or very satisfied, with only ~6% expressing any dissatisfaction). This indicates that while themes and colors can be nice-to-have improvements, they are not pressing issues compared to the functional problems above.

In summary, the needfinding survey pinpointed **critical usability pain points** notably, the difficulty of navigating long chat histories, the high incidence of accidental message submissions, and the inability to easily save conversations — as well as **opportunities for new features** like multi-chat support. These insights ground our design direction, ensuring we address the real needs users voiced rather than peripheral aesthetic tweaks.The survey results are provided in the spreadsheet link given below:

Chat GPT User Experience Feedback Survey (Responses)

Preliminary Design Ideas

Based on the user needs identified, we brainstormed potential design solutions. The focus is on **four main design directions** that directly tackle the top pain points, with a couple of additional ideas noted for future consideration. The primary concepts under exploration are:

1. **Split-Screen Chat Interface:** Introduce an optional dual-pane view that allows users to view or interact with two chat sessions side by side. This split-screen mode would enable multitasking – for instance, a user could compare answers from two different chats or continue a conversation on one side while scrolling through a previous chat on the other. By allowing simultaneous viewing of multiple contexts, this design addresses the desire for multi-tasking and reduces the friction of switching between chats.

2. **Quick Navigation Sidebar:** Add a collapsible "question navigation" panel within each chat. This sidebar would list the questions or prompts asked in the conversation (possibly as a clickable outline or bookmarks), so the user can jump directly to a specific question-answer turn. Implementing this feature would make it much easier to navigate long chats, directly targeting the pain point of tedious scrolling. (This idea could be integrated with the split-screen interface – for example, one panel of the split screen could serve as a navigation/index view for the chat.)

3. **Export Conversation Feature:** Provide a one-click **"Export Chat"** function to save the entire conversation in a user-friendly format (such as PDF or Word document). The design would include an easily visible export button in the chat UI. When clicked, it

would generate a clean transcript of the chat that the user can download. This addresses the strong user demand for saving chat content, allowing users to archive important discussions or share them outside the ChatGPT interface without manual copy-pasting.

4. ·**Improved Message Input UI:** Redesign the message input box and send mechanism to prevent accidental submissions. Possible solutions include introducing a dedicated “Send” button (so that pressing Enter simply adds a newline by default) or requiring a key combination (e.g. Shift+Enter) to create a new line, clearly distinguishable from the action to send. We are considering visual cues (such as a multi-line expandable input area and a disabled-send state until the user explicitly clicks send) to make it obvious when a message will be sent. This design change aims to eliminate the frustration of unintentional message sends and give users more control when composing long queries.

Additionally, we took note of other suggestions like **UI theme customization** and incorporating **visual cues in responses**. For example, a future design could allow users to choose custom color themes or enable richer response formatting (with colored text or emojis) for better clarity. However, these ideas are currently secondary priorities. Given time and scope constraints, our immediate design efforts will concentrate on the four key areas above that promise the most impact on usability.

Finally, we acknowledge some suggestions from the needfinding that are *beyond UI scope* (e.g., improving the AI’s reasoning or accuracy). While our design can’t directly change the model’s intelligence, we will consider if UI can indirectly help (such as providing a way for users to give feedback on answers or request clarifications). However, our primary focus remains on interface improvements as listed above. These preliminary design directions will be refined by creating sketches and low-fidelity prototypes, and we will be gathering feedback on these ideas (from peers, mentor, or via the planned interviews) before we proceed to building the high-fidelity prototype in the next milestone.

Additional Requirements for Engineering focused Teams

Functional Requirements

We have provided the details of functional implementation under the assumption that we will be able to implement all of them. For the functionalities that we are unable to implement/facing issues, we will provide wireframe/prototypes.

1. **Custom Color Palettes.** This Provide options for users to select or define custom color themes for the UI. It provides the following features:

   (a) A palette selection UI element (e.g., color picker or dropdown menu).
   (b) Persistence of user-selected colors across sessions.
   (c) Application of CSS variables or styles that override default styles.

The acceptance Criteria would be as follows:

(a) Users can choose and preview color changes.

(b) Custom settings are saved (using localStorage or similar) and reloaded on page refresh.

2. **Split-Screen Mode.** It enables a split-screen layout that displays two independent Chat/GPT instances side-by-side. It provides the following features:

(a) A toggle or button to activate split-screen mode.

(b) Each screen should function independently, allowing separate conversations.

(c) Both screens should share the session (login state) to maintain consistency.

The acceptance Criteria is mentioned below:

(a) When enabled, the UI divides into two equal panels.

(b) Each panel loads a separate instance of the ChatGPT chat interface without significant performance degradation.

3. **Export to PDF/HTML.** This allows users to export the current chat content as a PDF or HTML document. It comes up with the following features:

(a) Export button(s) for PDF and HTML.

(b) Correct formatting and styling in the exported documents.

(c) Option to include timestamps and metadata (optional).

The corresponding acceptance Criteria are as follows:

(a) Clicking the export button generates a downloadable file in the chosen format.

(b) The file accurately reflects the chat content and applied styles.

4. **Quick Navigation Right Sidebar.** It Incorporate a collapsible sidebar on the right that allows for quick navigation within the chat. It will show the starting few letters of prompts written by the users in the current chat screen. This comes up with the features listed below:

(a) Sidebar items representing last k prompt input by the user in the current chat.

(b) Smooth scrolling and active state indication.

(c) Option to hide or collapse the sidebar.

The acceptance Criteria are as follows:

(a) The sidebar is accessible from any point in the conversation.

(b) Clicking a sidebar item scrolls the chat to the corresponding section.

(c) The sidebar does not obstruct the main chat content.

Non-Functional Requirements

1. **Usability**. The enhanced UI should be intuitive and easy to use as follows:

(a) Clear labeling of controls.

(b) Minimal learning curve for toggling modes and using customization features.
(c) Responsive design to accommodate various screen sizes.

2. **Performance**. The enhancements must not significantly degrade page performance. Following are the points to be considered:
(a) Quick loading of split-screen instances.
(b) Efficient memory usage even when both chat windows are active.
(c) Export functionality should complete within a reasonable timeframe.

3. **Compatibility**. To ensure that the scripts are compatible with major modern browsers (Chrome, Firefox, Edge).
(a) Thorough testing on supported browsers.
(b) Use of standard web technologies (HTML5, CSS3, JavaScript).

4. **Maintainability**. The code must be clean, modular, and well-documented. Use of modular functions and clear comments to facilitate future enhancements.

5. **Security**. The scripts must not introduce any new vulnerabilities.Majority of code will execute on client side browser only with no/minimal server side communication.

Technology Stack

1. **Userscripts.** Userscripts are small JavaScript programs that can be used to add new features or modify existing ones on web pages. With Tampermonkey/Violentmonkey, we can easily create, manage, and run these userscripts on any website we visit very easily. If time permits, we would create a browser extension with the same functionalities.

2. **Frontend.** *HTML/CSS/JavaScript*:Standard web technologies form the basis of the Tampermonkey scripts. CSS is used for layout (flexbox/grid) and theming via CSS variables.

3. **Framework Consideration.** While not strictly necessary for a Tampermonkey script, if the UI complexity increases, frameworks such as Vue.js or React may be considered for managing UI state. For this project, vanilla JavaScript seems to be sufficient to manipulate the DOM directly.

4. **Libraries**. jsPDF or html2pdf: For implementing the PDF export functionality.

5. **Backend.** This is not required as the enhancements are applied via Tampermonkey on the client side, there is no backend component. All processing is done within the browser.

6. **APIs.** Mainly Browser APIs. Leverage native browser APIs for DOM manipulation, event handling, and local storage for persistent settings like saving custom color preferences.

Timeline and Next Steps

With needfinding complete, our team is now transitioning into the prototyping phase. We have outlined an updated timeline and action plan for the upcoming milestones to implement and evaluate the proposed designs:

Milestone 3 – Prototyping (due April 15): In the next two weeks, we will develop low-fidelity and mid-fidelity prototypes for the main design ideas (split-screen chat view, navigation sidebar, export function, and improved input box). This will include creating wireframes and interactive mock-ups illustrating how these features integrate into the ChatGPT interface. We plan to conduct quick heuristic evaluations and possibly brief informal user walk-throughs with the prototypes to catch any glaring issues early. By mid-April, our goal is to have a functioning interactive prototype that embodies the enhanced UI/UX features identified.

Final Milestone – Evaluation & Refinement (due May 1): After prototyping, we will conduct **usability testing** with target users to gather feedback on the new interface features. This will likely involve scenario-based tasks (e.g. navigating a long chat using the sidebar, exporting a chat, etc.) to observe how users interact with the prototype. We will collect both qualitative feedback and quantitative usability metrics. Using these insights, we will iterate on the design to fix any usability problems and polish the UI. The final two weeks of April are reserved for refining the prototype and preparing the final project report and presentation. By the final deadline, we aim to deliver a validated design for an enhanced ChatGPT UI, along with analysis of how the changes improve the user experience.

Through these next steps, our team will ensure that the design ideas inspired by the needfinding results are effectively translated into a tangible prototype. We remain focused on addressing the core user needs identified, and we will use the upcoming prototype evaluations to verify that our solutions indeed alleviate the pain points (e.g., easier navigation, no accidental sends, successful chat exporting). This iterative design process will guide us toward a final refined solution that significantly enhances ChatGPT's overall UX.