Comparative Heuristics Report

BCIT DTC 13 ‘STACK’ - Google Calendar

Andrew Solomko
Donovan Larkin Newcombe
Kelsen Alloway

Summary

This report is all about the usability of STACK, which is a productivity and time-management application that is meant for students to be successful in the aforementioned skills. It is designed with the thought in mind of managing things such as assignments, tracking required study hours, and visualizing their weekly schedule. Comparing STACK to Google Calendar, a widely known and used productivity system is chosen as it is a direct competitor and a useful means of comparing how usability heuristics are used.

Three of Nielsen's usability heuristics were used for the analysis. The first is Aesthetic and Minimalist Design, Visibility of System Status and lastly, Recognition Rather than Recall. Each of these play a major part in productivity tools as clarity, efficiency, and rapid interaction will be needed for the users of the app. The stated heuristics are used to be able to tell how systems are able to communicate information to the user, reducing cognitive loads, and supporting decision-making for users.

The analysis of the competitor had taught that while Google Calendar is thought-out and useful, it suffers from issues such as visual clutter, inconsistent iconography and an interface that forces the user to use their own recall rather than their recognition. This greatly inhibits the user from being able to easily guide themselves through the application. On the other hand, with the application STACK, it benefits from a new and simple design. The choice naturally reduces the clutter and encourages intuitive navigation. There are several areas that were identified to still need improvements. Some aspects such as reinforcing the system feedback, simplifying certain visual elements and removing unnecessary text would be useful. There was also the fact that ensuring an interface that follows consistent interaction patterns would also improve the heuristics. All of this solidifies that STACK is not perfect and has room to grow.

The usability heuristics remain highly relevant in modern development. They are able to give a framework on how strengths and weaknesses are diagnosed in interface design. Though, it still ensures that the decisions remain centered on user needs rather than developer assumptions as that is important.

As future developers, it should be noted that the importance of designing the product is not based on ourselves. It should be known that there are a variety of things that can be important. Things such as minimizing the cognitive load through familiar patterns, ensuring an immediate and clear system action/feedback for the user and consistently applying recognizable visuals is quite important. Alongside that, using interaction metaphors are all highlighted as important to be thought of for the users instead of as a developer. All of which guide the future iterations of STACK and server as foundational guidelines for creating an accessible, intuitive and user-centered application

Introduction

The app we created is called STACK. It is a time management application designed to help students keep track of their assignments. The app in its current state has all the basic necessary features the users would need. This report analyses the usage of three usability heuristics for user interface design in STACK's user interface.

The competitor we chose to compare with our app is Google Calendar. We found that Google Calendar, a popular time management application, came up a lot when gathering survey responses for our app, and it has some features that are similar to the main features in STACK. This report will analyse the usage of the same three heuristics in the user interface of Google Calendar.

The three selected heuristics are as follows. Aesthetic and Minimalist Design, Visibility of System Status, and Recognition over Recall. These were selected because they were the most relevant to our application, which is still in an early stage of development. For each heuristic this report provides an explanation of the heuristic, followed by an analysis of Google Calendar, and finally an analysis of STACK. Both app analyses will have the following structure; examples in the form of screenshots of the app, analysis of the usage of the heuristic, and improvements that should be made in order to better meet the heuristic. The order in which the heuristics are presented is not relevant and has no exact purpose.

Aesthetic and minimalist design

Explanation & Relevence

Aesthetic and minimalist design covers the importance of both the familiarity and noise reduction, in any UI. The first half, aesthetic design has two parts. First, the design must seem familiar to the user on their first glance. This includes using universally accepted symbols and illustrations, such as a gear representing settings. And second, the aesthetics of the design must be tailored to the desired audience or persona, which is a personality created by developers to represent the typical target user for their product. Note that these two can conflict. Therefore the design should adapt a mixture of the two that makes sense to the target audience or persona (Fessenden & Moran, 2021).

The second part, minimalist design is also made up of two main parts. First(developers should), minimize the amount of information given to the user. This is done by removing any unnecessary information, also called noise, that the user doesn't need to complete their desired task. Second, make sure that when fulfilling the first step, you don't remove any necessary information. A minimalist design can look crowded, as long as there is no noise, and the information that is given is strictly necessary (Fessenden & Moran, 2021).

This heuristic is relevant because it makes navigation more intuitive for the user, helps the user complete their desired task, and provides them with all the necessary information. Intentional design that avoids unnecessary noise keeps the user interested in the app, and maintains their confidence in the system.

Competitor Analysis: Google Calendar

Google Calendar is a scheduling and time management application. It uses aesthetic and minimalist design to ensure the user has access to many important time management features.

For example Figure 1 shows the task tab of Google calendar.

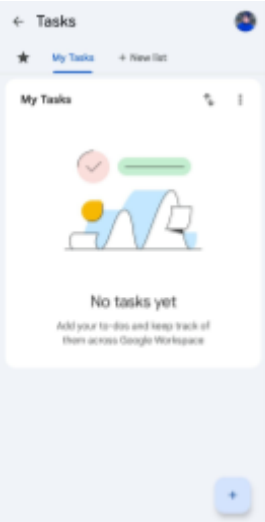

Figure 1. Task Page, Google Calendar, 2025.

The tasks page has a plus sign with a colored background near the bottom. This is easily recognizable as the action of adding a task. Near the top left, is a star icon which is commonly used to represent the starred, or most important, items in a list. An arrow near the heading on the top left points left, signifying a ‘return to previous page’ action. Two opposing arrows can be seen in the top right. Their function is to choose the sorting method, but this is not immediately obvious. So a more recognizable symbol should be used to improve the aesthetic design of this page. Finally, the three dots beside the arrows signify more options. Everything except the sort is universal and recognizable to any user. Showing that the aesthetic design is mostly good, with just one issue

In this view of Google Calendar the user is shown some necessary information. The heading, a **My Tasks** tab, a **New list** tab, a no tasks notice, and a short instruction to add tasks. This is everything the user needs to know at this moment. However, one big piece of noise is present on this page. The illustration in the middle doesn’t give additional information to the user, and is therefore a piece of noise that should be removed to meet the minimalist design part of this heuristic.

Figure 2 shows the main page of Google Calendar.

Figure 2. Calendar Page, Google Calendar, 2025.

This page contains several important pieces of aesthetic design. In the top left corner sits a hamburger menu symbol, and the top right has three different symbols as well. Of these, the middle one is unclear. It is meant to display today's date, but is not obvious to a user, as it sits among two clickable action symbols.

On this page, the user can see a calendar of the current month, and each day a user has created tasks for, will show the user the titles of those tasks. All necessary information has been provided to the user. The biggest issue with the minimalist design of this page is the fact that the current date is both shown in the top row, as well as highlighted in the calendar. One of these should be removed to reduce unnecessary noise.

Competitor improvements

Google Calendar has been around for many years and is a powerful time management tool. However it can struggle to meet this usability heuristic. Future improvements to this app should include:

- Switching the sorting button out for a more universal symbol (Fig. 1).
- Removing the illustration from the tasks page (Fig. 1).
- Removing date from top navigation (Fig. 2).

Our App: STACK

STACK uses aesthetic and minimalist design to help the user navigate to and complete their desired task. Figure 3 shows the home tab of STACK, that the user would see the first time they use the app.

Figure 3. Home Tab, STACK, 2025.

As far as aesthetics go, the main processes the user will need to complete are all marked by easy to recognize icons. The plus sign, much like in Google Calendar, will perform the action of adding a task. The calendar, house, and gear symbols can be recognized as tabs. This perfectly reflects the aesthetic design part of this heuristic.

This tab mostly meets the criteria of minimalist design. The title, a short instruction, the plus symbol, and the tabs are all the user needs to see in this mode. However, this tab does contain some noise. The toggleable dashboard is something the user doesn't need until they add tasks. As well as the text beside the tab symbols. This text is not necessary, as the symbols are already obvious enough for the user to understand each tab's contents. The navigation tabs are the same in every view, and therefore won't be discussed in the following figures.

Figure 4 shows the home page after two tasks have been added.

Figure 4. Task Cards, STACK, 2025.

Two pieces of universal aesthetics are present on each task card. The coloring should be easy to recognize for any user. The background of each card represents the time the user has to complete it. Red being the more urgent task than the one with green as the background. The two buttons, **Complete** and **Delete**, are also colored in green and red. This can conflict with the colors of the cards, and should instead be symbols. For example a cross for delete, and a checkmark for complete. Making them universally recognizable.

This design has several flaws. Now that the user has added tasks to the home tab, most information is displayed on each task card. The title, importance, due date, study hours remaining before the due date, a complete button, and a delete button. The only necessary piece of information that isn't displayed is the time estimate for each task that was inputted by the user. This should be added to meet this usability heuristic. Each card contains a piece of noise. This being the slack ratio, which was used for testing, and isn't relevant to the user. Therefore, it must be removed to make this design more minimalist.

Figure 5 shows the calendar tab, which the user can use to communicate their study time to STACK.

Figure 5. Calendar Page, STACK, 2025.

Two main pieces of aesthetic design are present in this tab. The calendar grid would be familiar to any student who opens this app as it will remind them of their class timetable layout. The arrows on either side signal motion towards the left and right. This being a calendar page is universal for moving to the next or previous part of the calendar, in this case the next or previous week.

On this tab the user is shown the calendar grid for this week, the days and month of the week in view, and a short instruction for this tab. An obvious missing piece of necessary information is the year, which is not visible or selectable by the user. This, along with the unnecessary study time text on each block, shows that the minimalist design of this tab needs to be further improved.

STACK Improvements

STACK is in its early stages, but it nevertheless partially meets this usability heuristic in especially minimalist design, as this app hasn't been in development long enough for it to contain unnecessary information. But many improvements must still be made, such as:

- Removing the dashboard when no tasks have been added (Fig. 3).
- Removing text from navigation tabs (Fig. 3).
- Replacing **Complete** and **Delete** with universal symbols (Fig. 4).
- Removing slack ratio from task cards (Fig. 4).
- Adding time estimate to task cards (Fig. 4).
- Adding the current year to the calendar (Fig. 5).
- Removing **Study time** text from study blocks (Fig. 5).

Takeaways as a Future Developer

This heuristic enforces the theory that an application should be made for a persona, and their use cases. It's important for a developer to remember that they aren't necessarily the same as the persona they developed. This means that just because the developer can see what they need and nothing else, doesn't mean that is the case for anyone who will use their product.

It's important for them to use things that are universally accepted to make their app easy to navigate, and show all important information to make sure the persona can carry out their own use cases. As well as limiting the noise of anything the persona doesn't need. If the application doesn't contain enough necessary information or contains too much noise, the persona will simply move on to a different application.

A developer should remember to always come back to the use cases for their persona when developing their app. And if they modify the original use cases, they must make sure the persona is modified accordingly, otherwise their app will go in the wrong direction in terms of aesthetics and noise, and users won't use it.

Visibility of System Status

Explanation & Relevance

Productivity tools are essential to help with the flow of information for time-management. Users rely on accurate feedback to be able to schedule their tasks, adjust workloads and for managing many of their deadlines. If a user cannot visibly see or know if something was saved or added, it creates a moment of uncertainty. This can cause the trust between the system and the user to plummet. Therefore, that effort should be known and there to provide an insight into how well a system supports a user's decision-making.

This principle is all about the effort of maintaining a clear communication point for the user and system. Visibility enables a user to understand the outcome of their actions. The user should see the changes in the system they are interacting with at any given moment. This allows for the individual users to see progress, confirm inputs and acknowledge changes. This in turn lets the user feel confident in navigating complex tasks (Harley, 2018).

This provides insight for the user's knowledge to see and know that the page is showing them what has happened. There should not be unnecessary confusion from the functionality of the system. Without a way to visibly see what might be happening, there's less room for deciding what to do. The visibility of the system status should guide the way for the next decision to accomplish the goals of the webpage (Harley, 2018).

Visibility of System Status refers to how effectively the interface a system has can keep users informed about ongoing processes. There is always a need for users to be informed on what is going on. The user should never be misinformed in what they do as the system should always lead them back to seeing an action on the screen within a reasonable amount of time or they might guess the next action to-do. (Nielsen, 1994).

If a user uses time-management tools, the heuristic is especially important. Many users depend on accurate and immediate feedback on their screen for planning. If there is a delay, missing confirmation or even if something fails in the background then that can disrupt how the user will proceed and their confidence in the system.

Competitor Analysis: Google Calendar

A **loading screen** from the Google Calendar mobile app is shown in Figure 1, displaying the app's initial state before the user logs in.

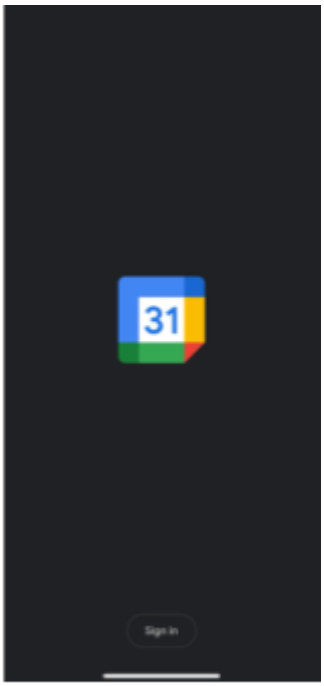

Figure 1. Loading Page, Google Calendar, 2025

In Figure 1, the Google Calendar logo appears centered on a dark background. This then has a "Sign in" button that is positioned near the bottom of the screen. There is only the sign in button for the user to click on that has interactivity and does not require more to ensure there is total focus on signing in.

This loading screen clearly communicates the system's current status as it shows the app being initialized, preparing itself for the **user's input**. The sign-in button serves as a cue to proceed, which reinforces any predictable navigation and reduces cognitive effort on the user. Though, the button is seemingly blending in with the background, lacking emphasis, which some users may not immediately take notice, especially with the logo taking some of the focus away.

Figure 2 presents a system alert displayed in Google Calendar when the user signs out of their Google account. In Figure 3, the task-creation interface can be seen in Google Calendar after the user selects the option to add a new task.

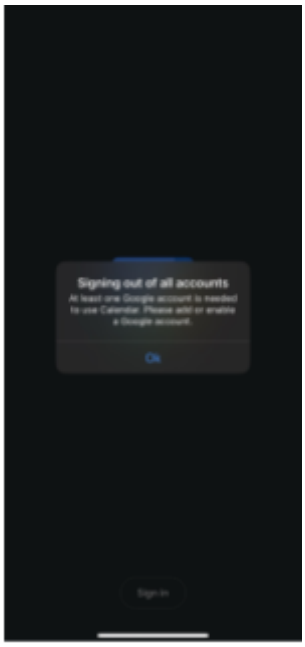

Figure 2. Sign Out, Google Calendar, 2025

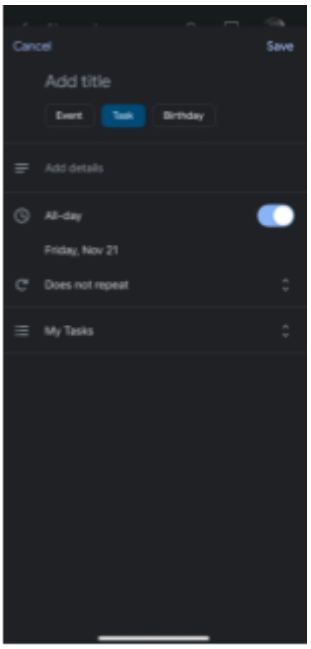

Figure 3. Add Task, Google Calendar, 2025

In Figure 2, a pop-up appears at the center of a dimmed background due to the alert. The only available action is an “Ok” button, which dismisses the alert and goes into the sign-in page. This makes the message the singular **focal point** of the screen for the user. Google Calendar instantly communicates the system state to the user and explains what must happen **after**. Though, it would probably be better for the Visibility of System Status to have a button that gives a call-to-action to add another account.

In Figure 3, the interface displays several interactive settings, with the “Task” category highlighted in blue, an “All-day” option to enable or disable, a date button, repetition options and a list selection, “My Tasks”. The “Cancel” and “Save” actions are clearly visible on the top, which allow the user to finalize or discard the task that they are creating. Every component has a visible reaction when it is tapped which indicates an updated state. Though, with that, there is no way for the user to know whether what they do on the screen will conflict with other parts of the system.

In Figure 4, the task options menu can be seen in Google Calendar, which is selected by clicking on the task and then clicking the three dots.

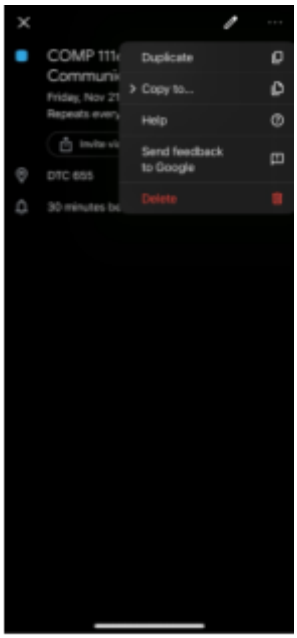

Figure 4. Task Options, Google Calendar, 2025

The screenshot shows a slide-up panel that contains a variety of contextual actions for the user. There is also the overlay of the event details within the background which remain visible but darkened. Each option reacts visually and “Delete” is highlighted in red to signal an action that is destructive. Which has no indication that it will be permanent or reversible, and no confirmation between it.

The icons beside the text allows for each action to be reinforced by the image, though, there can be more room for improvement with further tooltips to explain each action.

Figure 5 shows an insight to the main Google Calendar view, and the events after completing a schedule by marking it.

Figure 5. Day-to-day schedule, Google Calendar, 2025

In the screenshot, the completed event is at the top of the day's schedule which distinguishes it. There is a strike through it to imply it is not active anymore and completed. The rest of the events are in solid blue and are differentiated from the event that is at the top that is already completed.

Visual cues are a key part of Google Calendar to communicate real-time status changes. When the user marks an event as complete, the system will update accordingly and instantly. Though, the sudden immediate finishing that is silent, can cause uncertainty on whether the tap was actually registered. On top of that, the completed tasks don't stand out too clearly, in which the colored tone is relatively the same as the normal tasks, which makes it blend in a bit easier to users who might not know the difference.

Competitor Improvements

Despite Google Calendar's strength as a reliable and widely adopted time-management tool, the mobile interface shows several areas where the Visibility of System Status needs to be adequately supported. Google Calendar would benefit from:

- Strengthening the sign-in button and reducing the space taken by the logo that might distract users.
- Adding an actionable step through the sign-out button to add another account.
- Adding confirmations to tasks to know that nothing will conflict.
- Explanations to each action.
- Increasing visual contrast between completed and active events.

Our App: STACK

In Figure 6, the empty state of the Tasks page can be seen, which displays sorting options and a call-to-action to add a task. In Figure 7, the task-creation is illustrated with a pop-up that appears once the user taps the add-task button. As for Figure 8, it displays the weekly scheduling grid used to view and manage study blocks throughout the week.

Figure 6. Task Menu, STACK, 2025

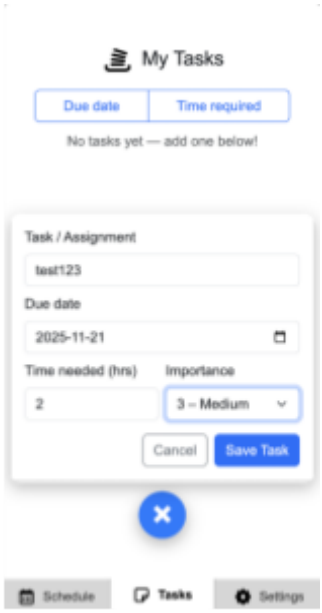

Figure 7. Adding Task, STACK, 2025

Figure 8. Adding Study Hours, STACK, 2025

Figure 6 has an interface that presents two sorting tabs, "Due date" and "Time required" at the top, followed by the message indicating no tasks were added. This clearly is communicating for the user to go to their next step, and the floating buttons are immediately recognised as action control. There is also a "+" symbol that is easily noticeable to indicate that the user should use this button for creating a task.

In Figure 7, it shows the “+” being clicked on, and then it turns to a “x”, and a task-creation pop-up can be seen. There is the “Save Task” and “Cancel” that is easily visible with it taking up most of the screen to show that you should be looking at the pop-up. The input fields are visually distinct, and give a cue for the user to know what to immediately enter. Though, there isn’t an actual dimming around the task-creation that can be used to help further the user into looking at the task-creation section.

As for Figure 8, a calendar grid can be seen. The grid shows days across the top and hourly times on the left. Existing study blocks are displayed as blue rectangles positioned within their respective time slots. Navigation arrows can be seen on the sides, which allows the user to move through the week. Scheduling studying time can be seen as immediately apparent to the user, the color-coded blocks compared to the white background allows for the user to know they’re input study hours into their time slot. Though, there is nothing to further indicate that these blocks have been saved to the user, which could further the Visibility of System Status.

In Figure 9, the main task-list view can be seen within STACK, where users can see all of their upcoming assignments sorted by time. Figure 15 shows the account creation pop-up that appears when a new user signs up for the app. As for Figure 16, the settings page is clearly visible where users can manage their notifications, theme preferences, and account actions.

Figure 9. Time Required Sort Feature, STACK, 2025

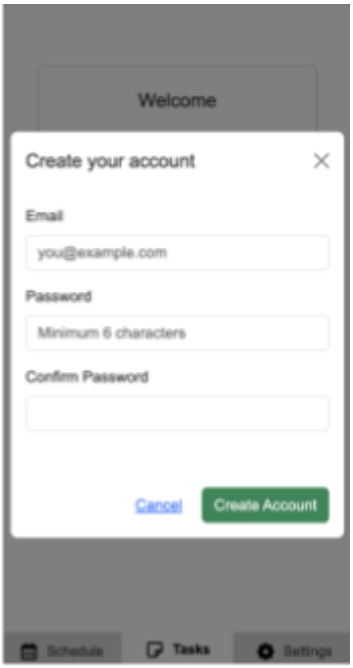

Figure 10. Create Account pop-up, STACK, 2025

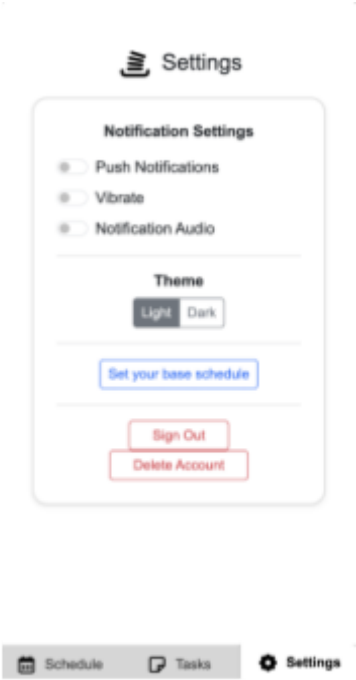

Figure 11. Settings Page, STACK, 2025

In Figure 9, a vertical list of tasks cards, each containing the data of the card can be seen. Each card has two clear actions to choose from, “Complete” and “Delete”. Complete is in green and Delete is in red to indicate that they are important choices. A “+” symbol is at the bottom a t all times to keep creation visually accessible. Users are able to see what they have to do, or how much time they have to put in and which of them are a higher priority. Though, the screen can become visually dense with a larger card padding, making it harder to manage long lists.

Figure 10 shows the screen that contains all the necessary fields for an email, password and password confirmation. Alongside that, two buttons show up, one in green that says “Create Account” and the other with a highlight in blue and underline saying “Cancel”. In conjunction with the dimmed background, it signals that the user should either complete or cancel this action. The user can easily know their next action based on what can easily be seen.

Figure 11 has an interface that groups different settings with each other. There are toggles for notifications, a theme selector with two options being light and dark, a button for setting the base schedule and a sign out and delete account option in red. The visual spacing and rounded container helps distinguish this page from task-oriented ones.

In Figure 12, it shows the delete account information page within the Settings tab.

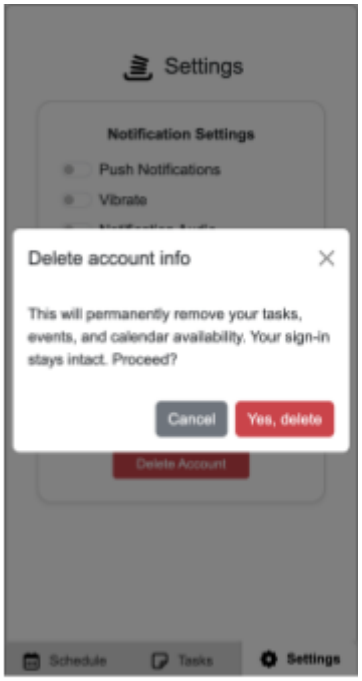

Figure 12. Delete Account pop-up, STACK, 2025

There is a clear system-state visibility that is provided and each button has a toggle status that is immediately evident. For example, in Figure 12, when you click on the Delete Account button, a confirmation will appear. This confirmation dimmens the background and allows for the user to focus on either cancelling or deleting, which is again in red. All of this to give the user a clear showing of the Visibility of System Status.

STACK Improvements

Despite STACK's strong use of a simplistic design, color, clear pop-up layouts and predictable patterns, several areas can still be improved upon. The areas for improvement are as follows:

- Dimming around the task-creation to help the user to follow more clearly.
- Adding a confirmation for saving a block to reinforce the system's response.
- Easier card management for long task lists.
- More animations like loading indicators and transitions between actions.

Takeaway as a Future Developer

Visibility of System Status is a communication contract between the system and the user. As a developer, it means designing an interface that can respond properly, continually showing users what is happening, what has changed, and what they can expect. Clearly showing this feedback reduces uncertainty, prevents mistakes and builds a long-term trust within the product. If the users must question if their action succeeded or registered an input, the product has failed at its purpose. This is all to allow users to always feel informed, in control and confident in their decisions (Harley, 2018) (Nielsen, 1994).

Recognition over Recall

Recognition over recall focuses on reducing cognitive effort, allowing users to act based on familiar cues rather than memorizing instructions. Interfaces that leverage recognizable icons, consistent patterns, and predictable workflows, help users immediately understand how to navigate an app. For a time-management tool, this is especially important: users must be able to interpret their schedule quickly and perform small tasks with minimal friction. (Budiu, 2024) (Nielsen, 1994).

Explanation & Relevance

Recognition over recall works by triggering a user's built-in pattern recognition. Familiar shapes (such as a hamburger menu), common metaphors (such as tabs for navigation), and consistent colour usage, effectively activate existing memory pathways. When an interface communicates

through intuitive, recognisable elements, the user doesn't need training or language-heavy guidance.

For productivity apps, the stakes are high: if a user cannot immediately accomplish their goal, they typically leave for a simpler competitor. Recognizable design patterns reduce the learning curve, increase efficiency, and broaden accessibility; All particularly important for tools aimed at students. (Budiu, 2024) (Nielsen, 1994).

Competitor Analysis: Google Calendar

Google Calendar uses colour effectively to distinguish event types; however, as seen in Figure 1, the interface quickly becomes crowded in the main calendar view. Low contrast in the base UI and rounded shapes contribute to visual blending, making it harder to recognise where one event ends and another begins. As a result, the user must consciously parse information- moving from recognition to recall.

While some icons (such as the hamburger menu, magnifying glass, and user profile) are instantly recognizable, others create ambiguity. The checkmark and the calendar icon provide no immediate mental association given the context (Fig. 1). Users must interpret, remember, or tap experimentally to understand their function.

Figure 1
Google Calendar Homepage.
Source: Google Calendar, 2025.

In Figure 2 you can see that Google Calendar's month dropdown offers a familiar layout, but the interaction adds vertical clutter by pushing the interface downward and revealing extra buttons. The open hamburger menu, shown in Figure 3, presents a dense cluster of icons and toggles, many of which are not standard visual metaphors.
Although the functions are powerful, they require experimentation and recall: the user must determine then remember what each toggle means, instead of recognizing it.

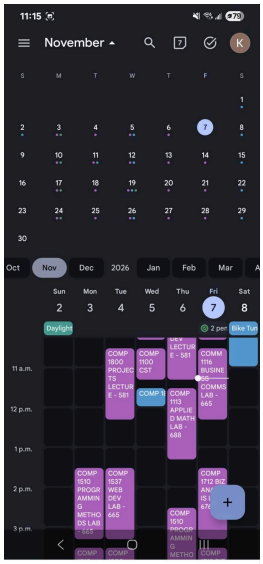

Figure 2
Google Calendar Drop-down Menu.
Source: Google Calendar, 2025.

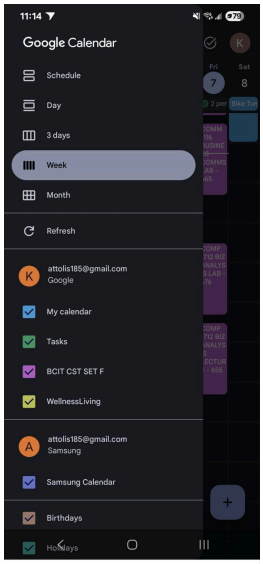

Figure 3
Google Calendar Hamburger Menu.
Source: Google Calendar, 2025.

The "+" button is familiar and well- placed (Fig. 1, Fig. 2). However, the menu that opens from it presents event types ("Event," "Task," "Birthday") (Fig. 4) that appear inconsistent with the selections later in the event creation window (Fig. 5). The presence of unrelated Google services and low-use options take priority space. The pop-up's temporary dimming of the background is highly effective- yet it is inconsistently applied across the app, weakening recognition patterns.

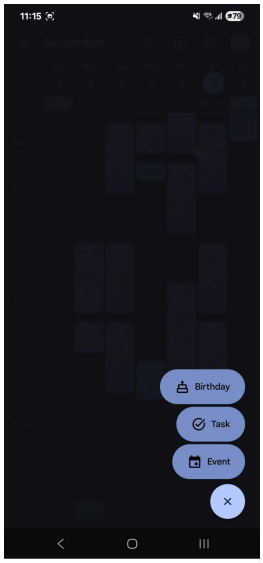

Figure 4
Google Calendar Add Task.
Source: Google Calendar, 2025.

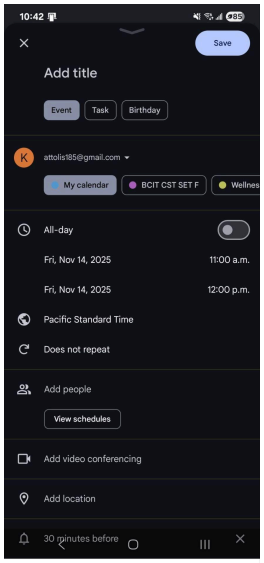

Figure 5
Google Calendar Event Menu.
Source: Google Calendar, 2025.

The add-event interface in Google Calendar, presented in figure 5, is notably dense and places competing elements at the same visual priority. The options for “Event,” “Task,” and “Birthday” appear again despite having been presented differently on the previous screen. This introduces inconsistency, forcing the user to recall the difference between these categories, which as of yet hasn’t been made clear. Sections dedicated to features such as inviting people, adding video conferencing, or attaching a location are functional but rarely used by the average user (Based on our user preferences survey conducted at the beginning of the term).Their prominence signals a misleading hierarchy of importance, increasing cognitive load and making it harder to identify the core fields required to create a simple event. As a result, the user must parse the interface rather than immediately understand it. This weakens recognition-based navigation.

Competitor Improvements

To better support recognition over recall, Google Calendar would benefit from:

- Greater icon standardization, removing ambiguous or proprietary visual marks
- Reduced density in menus and clearer grouping of related actions
- Consistent pop-up behaviour, reinforcing the same visual hierarchy across all screens
- Clearer differentiation between event types so the user does not need to guess which menu affects which function

Our App: Stack

As shown in Figure 6, Stack intentionally avoids overwhelming the user with information. Tabs replicate the familiar browser metaphor, creating immediate navigation recognition. Colour is used to emphasise meaning and interactability. Avoiding decorative highlighting keeps attention anchored on usable elements.

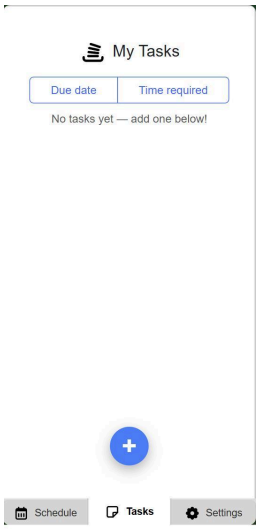

Figure 6
STACK Landing Page.
Source: STACK, 2025.

The “+” button follows a standard convention and opens a clean, minimal task form, shown in Figure 7. Autofill reduces effort, and switching the “+” to an “×” when the pop-up is open reinforces recognition by signaling a clear exit action.

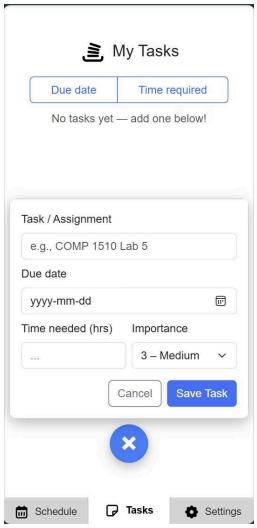

Figure 7
STACK Add Task Menu.
Source: STACK, 2025.

In Figure 8, you can see that, once created, tasks appear as cards with consistent colours, spacing, and metadata placement. This allows users to instantly identify priority, deadlines, and completion states without reading detailed text. We use familiar and established colours(Blue, Red, and Green) to associate element use. With Red highlights drawing user attention to priority or severity. The shape, shadow, and colour selection, all support recognition over recall.

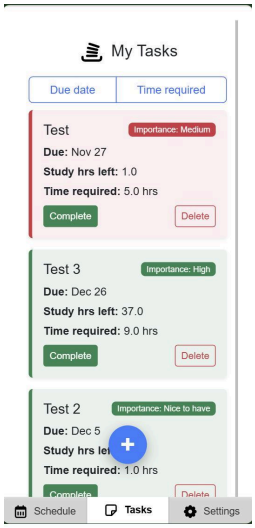

Figure 8
STACK Task Display.
Source: STACK, 2025.

The Settings tab in Figure 9 uses our same colour palette and associations. Red for destructive actions, and blue for interactivity. Users do not have to interpret meaning; the association is automatic and widely recognized

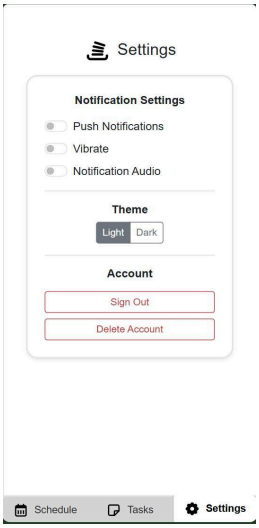

Figure 9
STACK Settings Menu.
Source: STACK, 2025.

The Base Schedule pop-up in Figure 10 is straightforward and familiar in design, but is an inherently unique feature. To compensate, we provide a small text description for the user ensuring understanding without search or experimentation.

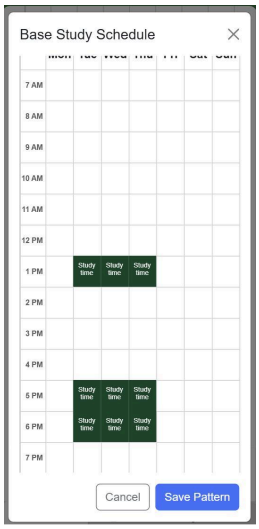

Figure 10
STACK Base Schedule M
Source: STACK, 2025.

The Calendar tab in Figure 11 uses weekly views with distinguishable colours for recurring blocks (green) vs user-modified blocks (blue). Arrows on the sides (highlighted in red) are designed to imitate swipe functionality (a feature we unfortunately do not have due to program constraints), following a universally-recognized “next/previous” pattern, reinforcing intuitive navigation.

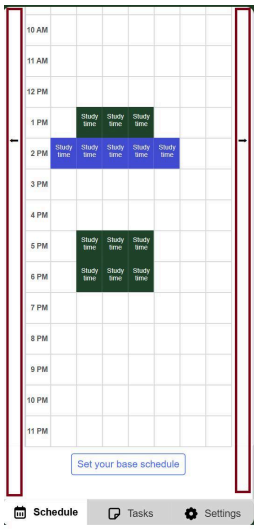

Figure 11
STACK Calendar Navigat
Source: STACK, 2025.

STACK Improvements

Although Stack performs well in recognition, future development should include:

- Even clearer iconography for first-time users
- Expanded tooltips for users who need additional context (without cluttering the UI)
- A more distinctive visual anchor for blocking vs editing schedule items
- Animation cues (small transitions) to help visually guide the user through changes

Takeaway as a Future Developer

Recognition over recall is not just a design principle, it is an efficiency principle. As a developer, it means creating interfaces where users can act instantly without thinking, reading, or recalling instructions. The more our designs leverage patterns people already understand, the more accessible and successful our products will be. This heuristic reinforces a key truth for future projects: if your user must pause to figure out what a button does, you've already lost them.

Conclusion

Evaluating Google Calendar through the lenses of Aesthetic and Minimalist Design, Visibility of System Status, and Recognition over Recall reveals a tool that is powerful but often burdened by its own complexity. Google Calendar uses familiar icons, colour cues, and quick system responses. However, many of its interfaces suffer from visual crowding, inconsistent iconography, and interactions that require users to interpret or remember meanings rather than recognize them. This creates cognitive load and reduces overall usability. Google Calendar meets each heuristic in parts, but not consistently or cleanly across its workflows.

Several improvements would strengthen Google Calendar's alignment with these principles. Reducing unnecessary illustrations, removing duplicated information, and standardizing icon usage would support both aesthetic, and minimalist design. Increasing contrast and providing clearer confirmations, especially for completed actions, would improve visibility of system status. Finally, reorganizing dense menus, clarifying event-type distinctions, and ensuring consistent pop-up behaviour, would reinforce recognition over recall and reduce effort for the user.

In contrast, our project STACK demonstrates strong initial alignment with these heuristics, largely due to its intentional simplicity. The interface emphasizes familiar metaphors, predictable patterns, and restrained colour use; helping users recognize actions without needing explanation. System feedback is generally clear, with most views avoiding unnecessary noise. However, as a developing product, STACK still requires refinement to fully meet these principles. These improvements include adding dimming effects and confirmations to reinforce system status, enhancing icon clarity, distinguishing interactive elements more visibly, and reducing residual testing artifacts or dense card layouts that could distract users.

Overall, conducting this heuristic analysis has reinforced several key lessons for us as developers. First, usability depends not on the number of features but on how easily users can understand and act on them. Every interaction should be recognizable, every change visible, and every element necessary. Second, the interface must always serve the user's mental model, not the developer's assumptions. Practicing empathy and staying connected to user-stories is key to competitive product design. Finally, meeting these principles is an ongoing process: as features grow, so does the responsibility to preserve clarity, familiarity, and trust. This analysis will guide future iterations of STACK, and future projects; ensuring that we develop tools that are not only functional, but genuinely usable.

References

Fessenden, T., & Moran, K. (2021, January 24). Aesthetic and Minimalist Design (Usability Heuristic #8) - NN/G. Nielsen Norman Group. Retrieved November 21, 2025, from https://www.nngroup.com/articles/aesthetic-minimalist-design/

Harley, A. (2018). Visibility of System Status (Usability Heuristic #1). Retrieved November 12, 2025, from https://www.nngroup.com/articles/visibility-system-status/
Nielsen, J. (1994, April 24). 10 Usability Heuristics for User Interface Design. Nielsen Norman Group. Retrieved November 12, 2025, from https://www.nngroup.com/articles/ten-usability-heuristics/#toc-1-visibility-of-system-status-1

Budiu, R. (2024, January 15). Memory Recognition and Recall in User Interfaces. Nielsen Norman Group. Retrieved November 20, 2025, from https://www.nngroup.com/articles/recognition-and-recall/
Nielsen, J. (1994, April 24). 10 Usability Heuristics for User Interface Design - NN/G. Nielsen Norman Group. Retrieved November 20, 2025, from https://www.nngroup.com/articles/ten-usability-heuristics/#toc-6-recognition-rather-than-recall-6

Images sourced from Google Calendar Mobile App, 2025 and Our App STACK, 2025