

Final Report

CM3070 – Final Project

ProductiviD

AN EFFICIENT TASK PLANNER FOCUSING ON HIGH-LEVERAGE ACTIVITIES

Sébastien Lavoie

March 26, 2023

Code repository:

<https://github.com/sglavoie/cm3070-final-project>

Goldsmiths
UNIVERSITY OF LONDON

Contents

1	Introduction	3
1.1	Inspiration: the main contenders	3
1.2	Drawbacks of existing solutions	3
1.3	Market overview	4
1.4	Motivation	4
2	Literature review	5
2.1	Previous work	5
2.1.1	Todoist	5
2.1.2	Trello	5
2.2	Contributions from the literature and gaps being filled	5
2.2.1	Initial design justifications	5
2.2.2	Implementation intentions	7
2.2.3	Calendars, planning & scheduling	7
2.3	The four pillars forming a new approach: evaluating their effectiveness	8
2.3.1	Inscribe	8
2.3.2	Prioritize	8
2.3.3	Focus	9
2.3.4	Review	9
3	Design	10
3.1	Domain and target audience	10
3.2	Design decisions and principles	10
3.3	Structure of the application	11
3.4	Libraries and services	12
3.5	Plan of work	13
3.6	Testing and evaluation	13
4	Implementation	14
4.1	Overview	14
4.2	Features implemented: a visual tour	14
4.3	Techniques and methods	16
5	Evaluation: results and discussion	17
5.1	Successes	17
5.1.1	Planning and execution	17
5.1.2	Functionality achieved	17
5.2	Failures and limitations	18
5.2.1	Google Calendar integration	18
5.2.2	Unit testing	19
5.2.3	Advanced features powered by AI	19
6	Conclusion	20
6.1	Salient takeaways	20
6.2	Improvements and future research	20

7 References	22
8 Further Reading	30
9 List of Figures	34
10 List of Tables	42
A Project management tools as task managers	44
A.1 Airtable	44
A.2 Asana	44
A.3 ClickUp	44
B Wireframes	46
C Screenshots of the final application	53
D Technical highlights: code snippets	56
D.1 Unit testing Google Calendar integration with Jest	56
D.2 Modularizing functional React components	56
D.3 Interacting with the SQLite database	57
D.4 Leveraging TypeScript types	57
D.5 Consuming React Context, persisted Redux store and theme	58
D.6 Using nested contexts and custom hooks	58
E SQLite database schema	59
F Interface design with Figma	60

1 Introduction

The world needs a new kind of task manager, just like **ProductiviDo**, which comes to life through the “*task manager mobile app*” template provided for the Mobile Development module taught at Goldsmiths, University of London.

1.1 Inspiration: the main contenders

There are a few wonderful products that deliver a good experience¹. Desirable features range from filters, labels and reminders to custom views and breadcrumbs. With **Todoist** [73], tasks can be quickly captured and will be added to an “*Inbox*” section. Once categorized, tasks are organized into projects, sections and sub-tasks. Although Todoist recently added a board view, **ClickUp** takes it to the next level by labelling columns [152]. In ClickUp, tasks are organized into multiple levels of hierarchies, down to task dependency, where sub-tasks depend on parent tasks. **Trello** takes an entirely different approach [106], where planning happens by moving cards onto different columns horizontally and re-ordering them vertically. Even though it has a concept of workspaces which is a group of boards, all the organization of a project usually takes place within a single board which contains multiple columns, each containing multiple cards. Planning in **Airtable** is much more freestyle with a spreadsheet-like system [102]. There is a Gantt view in the paid plan to visualize tasks over time. Using Airtable feels like interacting directly with a SQL database with its heavily grid-focused appearance. Finally, **Asana** shines mostly within a team setting and mainly works with the core concept of projects, which themselves can contain tasks, sub-tasks and even sub-projects [88].

1.2 Drawbacks of existing solutions

There are usability issues across all these mobile apps (and others) for different reasons. Most popular task managers have a limited search functionality, making it hard to find what one needs quickly unless a good organizational system has been consciously put in place beforehand. Crucially, such applications do not address the fact that unimportant tasks accumulate over time and provide no system to focus on what matters to avoid the fate of having “*41% of to-do items [...] never completed*” [103].

A second point worth mentioning is that a calendar integration is not a core feature deeply integrated into these apps. For instance, Todoist offers a 2-way sync with Google Calendar, but tags, labels, priorities and areas of responsibilities are not shown nor taken into account in the calendar [129]. Some task managers will provide their own built-in set of calendar features that partially solve the problem of being able to navigate a complete schedule in a breeze [71, 78, 84, 124, 126, 131]. However, none of these applications really allow efficient scheduling within the calendar. Some allow the user to add or modify events in the calendar, but this remains a slow process working with reduced functionality [79, 85, 125].

With regards to organization, all these applications are well-suited to manage tasks but when it comes to “*time boxing*” or “*time blocking*”, they will show every single task in the calendar view. This is impractical at scale as it quickly becomes overwhelming with dense information, a setup encouraging fragmented work and multitasking. Fragmented work caused by context switching has been shown to have a negative impact on productivity and on output quality [23] while multitasking will lead to “*experiencing more stress, higher frustration, time pressure and effort*” [59].

¹Cf. Table 1 and Table 2 for a summary of the strengths and weaknesses of the main contenders.

1.3 Market overview

ProductiviDo will target in its first iteration Google Calendar because this product can be consumed from a plethora of desktop and mobile applications [24] and there are about 4.3 billion Google users [121]. Once data is integrated into Google Calendar, it can be consumed by different applications such as Apple Calendar [97], Zoom [89, 154], Slack [123, 151] or other Google products, including Google Docs [98, 150].

The number of smartphone users will increase by almost 23% over the next 5 years, forecasting more than a doubling over a 10-year period (2016–2026) [66]. Task management software is going upwards of USD\$5 billion in market size, the CAGR —*compound annual growth rate*— is hovering around 14% and continuously growing [61, 62, 64, 67]. Combining supply and demand, we can see that both are expected to increase drastically over the coming years and that task managers fall into a category of growth market, which is desirable as it clearly indicates there is room for new rivals in the field: this is distinctly the case given that the task management world is still dominated by a few big players [65], many products have retired over the years (which unsaturates the market) [138, 141, 143, 144, 146] and considering that ProductiviDo solves a real problem —that of staying on top of a growing list of commitments— with an innovative solution not yet implemented elsewhere.

1.4 Motivation

One reason why ProductiviDo will shine is because calendars are still under-appreciated in existing task managers, yet all major task manager applications integrate them in one way or another [71, 78, 84, 129, 131]. It has been shown through many studies that calendars save time [16], improve productivity [2], reduce stress [35] and make it simpler to batch similar activities [11]. Plus, they can be consumed in a myriad of ways with their Application Programming Interfaces (APIs) [75, 100]. The problem with using only a calendar without a task manager is that they are inconvenient when dealing with many events or tasks [16]. On the other hand, relying only on task managers makes it difficult to appreciate how useful calendars are when it comes to visualizing one's schedule from different perspectives such as a weekly or monthly view [48]. Regarding ProductiviDo as a task manager and not only a calendar integration, it will distinguish itself by allowing users to quickly and efficiently record their routine activities. This is because it has been shown that these occupy a good amount of time even for productive people and there is currently no application that really implements a direct solution to this [101].

Intelligent prioritization is essential, yet difficult to achieve with existing products [33, 57]. Beyond capturing tasks and categorizing them, this application will feature a prioritization system that will help users focus on their most high-leverage activities with the help of custom sorting algorithms as advocated by Edmond Lau in his book *The Effective Engineer* [44]. These will take into consideration aspects such as urgency and importance — in the spirit of the Eisenhower Matrix [18, 57] — as well as deadlines to calculate the impact that completing a task will have. Importantly, the priority matrix is adapted to account for other needed facets, such as when other people are involved in the task dependency chain (delegatees), an adaptation of the “*fit*” variable from the Sung Diagram, which is an extension of the Eisenhower Matrix [54].

The application will benefit users by providing a built-in planning system that will make it easier to follow a successful schedule. This goes hand in hand with findings illustrating how, in the absence of a mindful review process, users will want to focus on small or unimportant tasks to release dopamine even though they know what would bring them the best outcomes [26], a real productivity threat leading to a decrease in output and an increase in completion time [9]. Therefore, by offering a system that will encourage users to take better decisions and visualize evidently their highest-leverage tasks in a familiar and useful fashion, ProductiviDo will carve its place in the market by solving unmet needs (e.g., [72, 87, 114, 117, 148, 149]).

2 Literature review

2.1 Previous work

As alluded to in section 1.1, the current work found inspiration in well-established task management software. There are numerous alternatives available on the market (e.g., [104, 110, 111, 134, 153]): only the most well-known and relevant ones for this project will be covered. Evaluating such products in depth goes beyond the scope of this report: key points will be highlighted to justify the need for a new solution².

2.1.1 Todoist

Todoist is one of the most popular task management software, with over 20 million users [127]. It is a web-based application that allows users to create tasks, organize them into projects, and assign them to specific dates. It also features a priority system that allows users to sort tasks by importance, and a reminder system that will notify users of upcoming tasks. While Todoist is a very popular solution, it has some shortcomings. Its design is arguably polished, yet its dark theme lacks contrast (cf. Figure 2), recurring tasks don't show immediately on which schedule they recur (cf. Figure 3) and tasks with comments or attachments are difficult to locate (cf. Figure 4). Other functional deficiencies include the fact that the "Upcoming" view shows all tasks from all projects as a large, single list of items without allowing filtering (cf. Figure 5) or that the app has a concept of "due date" with no "start date" such that duration is not taken into account nor can be visually represented meaningfully in a calendar. Equally relevant when comparing to what ProductiviDo will offer is the fact that there is no way to apply "time boxing" as there is no calendar view and only a functionally limited Google Calendar integration, task filtering is not intuitive due to the custom query language syntax used by the app and there is no easy way to find recurring or routine tasks without setting up a custom labelling system on one's own.

2.1.2 Trello

Trello lies on the other end of the feature creep spectrum when compared to ClickUp and can be seen as possessing an even more basic set of capabilities than Todoist [128, 133]. With its simple and uncluttered drag & drop interface, Trello is much more intuitive than a more sophisticated and flexible option such as `Todo.txt` which can be technically extended at will and can be operated from the command-line on a desktop computer or from a multitude of third-party applications on mobile devices [96]. Trello's simplicity comes at a cost, however: it is not uncommon to lose track of "cards" because one must swipe horizontally in order to navigate the UI; and very little information fits on the screen at once, where only a fraction of a "board" with up to a few "cards" is shown (cf. Figure 7). There is a copious amount of calendar integrations available for Trello [130], yet the core of the issue that ProductiviDo aims to address remains unsolved: the app's metadata is not accessible from Google Calendar (e.g., priority, area of work, task status, etc.) and the app does not provide a way to visualize the user's tasks in a convenient calendar view as it only "displays all cards with due dates by month" and only when using one of its paid feature, called "Power Ups" [132].

2.2 Contributions from the literature and gaps being filled

2.2.1 Initial design justifications

Amongst a panoply of scientific papers on the subject of task management, the quintessential work of Bellotti et al. serves as an excellent entrypoint listing a set of design requirements to be expected in an

²See Appendix A for additional insights on the evaluation of Airtable, Asana and ClickUp.

efficiently conceived task manager [53]. Of particular relevance to the creation of ProductiviDo are the needs for the app to be “*instantly on*” with the option to have “*no formal task description*” and a “*mechanism for handling stale to-dos of low importance*”. The authors highlight that the main issue is “*making sure that the important tasks get done*”, which led to further research being exposed on the topic of prioritization matrices. The authors also advocate for task states, location, social relations, notes and time constraints to be captured, all of which are taken into account in the design. In a more recent study looking at the importance of prioritizing tasks, further evidence suggested that a prioritization model should be adopted (e.g., HI/LO, CARVER or Carpenter) in a task management system, pointing at the Eisenhower Matrix as a useful framework to use for this very purpose [57]. Other less formal alternatives such as the RICE or ICE frameworks have also been reported to be used with success and could be adapted to this project with their scoring model [115, 116].

Another aspect which served to inform the overall direction to take is the well-known fact that humans do not work optimally when multi-tasking as this leads to a high cognitive load in keeping track of different areas of focus or projects [55]. Fragmentation of information in one’s work—which occurs when multi-tasking—is a problem because people won’t remember where to find the information [43]. Applying this idea of fragmentation to the digital world, it only makes sense that offering too many ways to store and retrieve data (such as with ClickUp) is counterproductive. Furthermore, the ability to customize the UI is a double-edged sword because “*users make mistakes and create building blocks with unintended consequences*” [56]. Indeed, if the user interface is cluttered or offers too many bells and whistles, managing the global context of one’s tasks becomes difficult and one inevitably encounters obstacles when trying to juggle with multiple actions at once, because “*prospective memory is fallible*” [51]. Hence, simplicity is key to a good task manager (which certainly helps to explain Todoist’s wild success). While simplification is part of the gamified model—Habitica being one example of a product incorporating many elements of gamification [134]—this model suffers from the “*undifferentiated use of rewards*”, leads to “*punishments for productivity*”, to “*feeling of not being taken seriously*” and cause “*negative anticipation*” [10]. Moreover, “*people found their standard task manager simpler to use*” and the integration of extrinsic motivational factors such as a reward system “*would not excite them to continue being immersed in a gamification application*” [58]. While this form of motivation does not appear to be effective in goal attainment, the literature concludes that “*valuing extrinsic goals [...] does not seem to increase our happiness, but attaining those goals does*”, adding that giving weight to intrinsic goals (such as personal growth, loving relationships and physical health) and achieving them is a better way to increase happiness [31]. As long as a task manager nourishes intrinsic motivation, it is likely to be effective in helping people to achieve their goals while maintaining optimal psychological health.

Other approaches were considered, notably the “Binary Priority List” due to its simplicity when it comes to comparing two elements (such as tasks) [52]. However, it lacks flexibility and becomes inefficient as more and more tasks are added because—unlike the famous binary search algorithm—this algorithm requires human intervention to classify tasks, which becomes burdensome faster than $\mathcal{O}(\log n)$. The literature on gamification also linked to sources that revealed solutions to solving procrastination (e.g., [15, 17, 50]), a self-regulation failure that gamified systems sought to minimize to no avail, which led to identifying a desirable core component of the application: implementation intentions. These can be integrated into the “mental flexibility” facet of “planfulness”—through a technique known as “mental contrasting with implementation intentions” (MCII) [25]—along with temporal orientation and cognitive strategies, all of which improve goal outcomes [20].

2.2.2 Implementation intentions

Implementation intentions have shown their effectiveness in helping people to achieve their goals [1, 14, 41], the real purpose behind using task managers in the first place. The idea is to create a plan of action for a specific goal, which is then followed by a specific time and place. The plan must be broken down into a series of steps, which are written down in a way that is easy to remember using so-called “if–then plans”. The specific formulation of such plans “produce much more favorable and reliable outcomes than broad intentions to pursue a goal” [22]. Findings from Bieleke et al. show that implementation intentions are also useful when planning one’s work because they help “evaluate information from a certain perspective”—which can be verbally recommended to users so they adopt a “certain processing style (e.g., deliberative thought) or perspective (e.g., neutral observer)” [7]. Their research also highlight the high potential of implementation intentions in reviewing tasks because “participants can strategically adopt a reflective mode of information processing that helps them make more sophisticated decisions when facing uncertainty”. Other studies show that such “plans to deliberate can be used to increase the likelihood of deliberation and thereby the effective processing of newly available information”, which further supports the benefits of implementation intentions with regards to planning [12]. To make the planning stage more effective, research also “indicates that having plans that focus on overcoming obstacles are important for maintaining progress and staying on track with one’s goals” [36], so this aspect will also be featured when creating tasks in the “**Inscribe**” step described in section 2.3.1. Once planning has been done, implementation intentions will also prove useful to take action towards accomplishing one’s goals because they lead to “automatic action initiation without further conscious intent” [4]—surprisingly and encouragingly, even in participants suffering from mild to moderate depression [13]. While this is obviously beneficial during the initial stages of a project or task to overcome procrastination, it is equally important to re-evaluate one’s course of action because sticking to a plan stubbornly “can hinder overall goal performance by causing one to overlook alternative opportunities for achievement”, amongst other negative consequences [21]. This explains why ProductiviDo encompasses a “**Review**” step which deals with this particular problem and more, described in section 2.3.4.

While this self-regulation technique can be applied successfully in the context of a task manager, it bears mentioning that it is not a panacea as it suffers from some limitations, one of which being that “formulating multiple implementation intentions is ineffective when changing unwanted behavior [...] due to interference in the enacting phase of the planning process” [34]. In other words, multiple implementation intentions can compete for attention, which makes one lose focus. To drastically mitigate this issue, ProductiviDo dedicates an entire step to “**Focus**”, which is described in more detail in section 2.3.3. The “**Review**” step will help with reducing the number of tasks one has to juggle with while there will be a “**Prioritize**” step (described in section 2.3.2) to help users decide which tasks to focus on before they get conveniently extracted in the “**Focus**” step.

2.2.3 Calendars, planning & scheduling

Section 1.4 introduced some of the core benefits of using calendars for planning and scheduling, yet scientific examination supplies a treasure trove of additional findings justifying their use alongside a task manager. A study exploring the use of a personal calendar discovered that most interviewees prefer a weekly view to aid with “opportunistic rehearsals” [32], a fact that sits well with the need for a reviewing system such as the one proposed in section 2.3.4 as well as the realization that a simple list or Kanban presentation is helpful but not sufficient to get a clear overview of one’s schedule at a glance. The same study made it clear that most users rely on their calendar to store efficiently certain types of information such as the date (97%), time (96%), location (93%) and purpose (69%). Quite convincingly, participants also used their calendar for “tentative event scheduling”, revealing the need to empty their mind to stay on top of their duties, a need addressed with the “**Inscribe**” step referred to earlier. This study also emphasized the fact that calendars

can be viewed on the web from most devices or even printed for offline reference³. Finally, this paper also found that a majority of users (63%) reported using reminders and alarms with the help of their calendar.

Even though almost two thirds of working professionals consider using a calendar application to be very important in their workflow, it was also found in another study that “*it is inconvenient to schedule on calendar apps*” [16], which supports the use of task managers for that specific necessity since they can alter tasks brilliantly and speedily. Calendars are not necessarily the best tactic when it comes to scheduling all kinds of activities either. Indeed, “*when consumers schedule their leisure, they may inadvertently reduce their utility for the activity*” [30]. With this type of activity, the authors divulge that “*roughly scheduling (i.e., without pre-specified times)*” constitutes a practical alternative. This results in effective scheduling when tasks are entered via a task manager and synced with Google Calendar since blocks of activities can still be displayed in the calendar view without the added burden of specifying a strict duration for each event.

There are some use cases that simply are not well suited for task managers, such as synchronizing events from Gmail (i.e., capturing tasks from different sources), setting reminders on appointments scheduled by third parties (where the primary source of truth must remain the calendar itself) or sharing a schedule with groups across the G suite applications with the help of smart suggestions coming from Google users’ data which is not entirely available to external applications [16].

2.3 The four pillars forming a new approach: evaluating their effectiveness

2.3.1 Inscribe

ProductiviDo features four core components to help tackle one’s work skillfully, the first of which being all about jotting down tasks, a demonstration of the “generation effect” which assists with remembering information better [5]. At this stage of the process, the idea is to record any potential future action that is required while being fully aware of the impact that such action will have on one’s life, benefitting at once from reducing one’s cognitive load by storing or discarding thoughts that do not require immediate response [19]. “Inscribing” in this application is an intentional step demanding a certain level of mindfulness and acts as a precursor to the use of implementation intentions as it similarly “*serves an important self-regulatory function*” according to a study on mindfulness done at the University of Rochester [8]. A 2020 study from Bieleke and Keller found that “*opportunity-focused plans are especially important for initial progress towards the goal*” [6], therefore this step ensures —through questioning the user’s aspirations— that the task is indeed a priority and that it is not just a distraction by evaluating its “leverage”, a notion introduced in section 1.4.

2.3.2 Prioritize

There is substantial evidence to establish that the concept of a “master list” has been popularized and has found multiple concrete applications (e.g., as a list of relevant jobs held by an applicant [135], as an Excel spreadsheet keeping track of crucial information at a high-level [86], in some jurisdictions in the legal world [109], etc.), including as a way to manage tasks ([80, 108, 113]), an idea also sprouting from Edmond Lau’s work [44]. In the context of task management systems, a master list manifests itself as a centralized location where all tasks are accessible once they have been “inscribed”. With ProductiviDo, “prioritizing” is a transient state: it is about handling a database of data to be referenced later when progress must be undertaken on specific tasks. It is a digital space separate from the current areas of focus (presented in

³It is worth noting that this work from 2008 was still in the early days of Google Calendar which was created in 2006 and left its beta state in 2009, which has since then offered offline viewing of calendars [140].

the following step in section 2.3.3) used to find the next priorities to work on, where filters are available to that end (see section 3 for the technical details). As defended by Bellotti's research as a set of desirable features, the "**Prioritize**" step offers the "*ability to sort by importance and deadlines*" as well as a "*smart prioritization*" of tasks by default, where the most important ones will flow to the top of the list, ready to be acted upon [53]. Because tasks listed in the "**Prioritize**" step are forcibly distinct from those listed in the "**Focus**" step as they are mutually exclusive, this intermediate step helps to reduce the number of tasks shown in the schedule, improving one's perceived control of time, busyness and stress [16].

2.3.3 Focus

Once there are prioritized tasks in the application defined in terms of implementation intentions, science dictates that there is a need to "*support the viewing of entire task vistas, but also allow different perspectives for different kinds of planning*" and for "*top priority items to be made apparent*" [53]. It is also known that setting clear goals eliminates intrusive thoughts and that "*commitment to a later behavior was linked to reduced rather increased strain on mental resources*" [22]. For these reasons, the "**Focus**" step is a crucial component of the application. This component has the additional benefit of mitigating the problem uncovered in section 2.2.2 about the lesser efficacy of having multiple implementation intentions at once because it helps to recall all the high-leverage tasks in a filtered version of the master list (i.e., the current working list), leading to the observation of the Zeigarnik Effect which states that "*an activity that has been interrupted may be more readily recalled*" [147]. In addition, seeing a list of promises to oneself in the form of MCII's "*seems to facilitate behavior change even when there is an initial reluctance to engage in the targeted behavior*" [14]. Even more benefits from MCII's include "*sustained task value and higher persistence*" [38], two aspects that contribute to goal achievement and, consequently, to turning a task manager into an efficacious product.

2.3.4 Review

At this point, a working list of tasks exists to "**Focus**" and a backlog of work remains to "**Prioritize**", yet not all labor should be brought to fruition as initially planned. Life circumstances change and "*plans to deliberate can be used to increase the likelihood of deliberation and thereby the effective processing of newly available information*" [12], which can help eliminate or update tasks that are no longer considered important or relevant enough given an ever-changing environmental context. A paper on making informed scheduling decisions also noted that "*self-monitoring and self-reflection often affect behavior, and this change typically goes in the desired direction of improvement*" [3]. These findings can be applied practically by creating plans focusing on overcoming obstacles to make sustained progress, allowing one to ponder every major decision during the journey [28]. A 2021 study on task management tools proposed to break complex tasks into smaller chunks (using a sophisticated approach with dependency graphs) [60], an idea that can auspiciously be employed in a simpler way by breaking down a larger unit of work into smaller ones, which can be done during the information-gathering phase (i.e. the "**Inscribe**" step) as well as during the reviewing phase with existing units of work, which is known as "*compartmentalization*" [47]. This self-reflection process can also contribute to one's well-being by providing written cues to recollect positive images from past accomplishments [37]. Looking towards the future, cueing the users to imagine positive outcomes and encouraging them to restructure their priorities can equally evoke optimism and improve life satisfaction [27] while inspiring them to live according to their personal values [29]. After all, a "*person who sets up life and its routines to avoid inner conflict between goals is better off in the sense that he or she ends up feeling fewer bad emotional states and is generally happier*" [15].

3 Design

3.1 Domain and target audience

ProductiviDo is a task management system staying clear of project management and time management, geared towards being a capture and review tool leaving notifications and scheduling to external systems, primarily to Google Calendar in these first product iterations. As such, the target audience is represented by conscientious people desiring to optimize their productivity with a tool discouraging multitasking, who don't mind hand holding to achieve their goals — willing to trade off flexibility in exchange for a more rigid system backed by science —, expecting a simple set of essential features for a distraction-free environment to get things done and wanting to manage their own set of personal tasks (as this product is not designed with teams or businesses in mind).

Explicitly supported users include people who want to follow a useful system without having to build it from scratch (e.g., setting up projects, tags, custom priorities, etc.) as well as those who need to keep track of tasks involving others with basic functionality (via Google Calendar and filters in the application). The application will not cater to those who want to rely mainly on desktop applications (because the core of the application is mobile only), nor to those who want to use a calendar other than Google Calendar (which is the only calendar undeniably supported in this version) or those who are looking for a solution integrating into other applications, since there will be no publicly accessible API at first to make this possible.

3.2 Design decisions and principles

Section 2 abstractly laid out the foundation for the core ideas being implemented in the application meeting the needs of the target audience. Concretely, a few design principles listed in “The Pocket Universal Principles of Design” inform how the application has been thought out [45], markedly:

- The 80/20 rule: “*80 percent of a product’s usage involves 20 percent of its features*”, which is true with ProductiviDo where users will spend most of their time focusing on the work they have to accomplish (cf. section 2.3.3);
- Chunking: “*Chunk information when people are required to recall and retain information*”, where each task will display the most important information to contextualize work;
- Constraint: “*Limiting the actions that can be performed to simplify use and prevent error*”, where each screen will have a single function with a premeditated lack of customization;
- Five Hat Racks: the five ways in which information can be organized (i.e., category, time, location, alphabet, and continuum), all of which are supported by ProductiviDo;
- Form follows function: “*Aesthetic considerations should be secondary to functional considerations*”, where the application maximizes the functionality available on each screen while taking into consideration the primary interactions needed at each step, regardless of the visual impact these decisions have;
- Garbage in, garbage out: “*Make it impossible to store bad data types*”, where the application will not accept invalid inputs on any field that could jeopardize the integrity of the data;
- Nudges: “*smart defaults, clear feedback, aligned incentives, structured choices, and visible goals*”, where the app makes the purpose of each interaction crystal clear;
- Progressive disclosure: “*A method of managing complexity, in which only necessary or requested information is displayed*”, where filters are selectively shown to match the intent of each screen and

where optional fields are hidden away by default until they are needed.

It is noteworthy that calendar integration will be optional and that the application will remain fully functional without relying on external calendars for maximum convenience, especially while users are offline. Even though this work underlined the usefulness of calendars when used as companions for task managers, one of the primary goals of a task manager is to be available at all times regardless of internet connectivity so that users can access their list of priorities and get work done: requiring an active connection and access to a calendar via sign-on before being able to do any work would break the principle of least astonishment in this type of application [49].

3.3 Structure of the application

The application is structured in such a way that each screen has a single function and that the user is guided through the process of adding, editing, and reviewing tasks (cf. Appendix B for a visual tour of the application with early wireframes and Appendix C to take a look at the final product)⁴. The application is designed to be as simple as possible, with a minimalist interface and a limited number of interactions to keep the user focused on the task at hand. Technically, it is arranged according to a pyramid schematic architecture, where the “hub” is the hamburger menu which links to all other pages and where all the sub-pages can go back to the “hub” [42]. In this architecture, the tab navigation at the bottom of the screen lists the core sub-pages which are always accessible and can lead back to the hub as needed so that the user does not need to remember how to get from one screen to another, since they are all identified with both a recognizable icon and text label.

The application is structured in a way that makes it easy to add, edit, and review tasks, with the following screens:

1. **Inscribe:** Looking at the tab bar (bottom part of the wireframes) from left to right, this is the first screen meant to be easily accessible and also the first step that is part of the process of managing tasks by carefully adding them to the system;
2. **Prioritize:** this is the screen responsible for displaying the list of tasks that have been added to ProductiviDo in the first step (**Inscribe**) but still require planning (i.e., they represent the tasks to prioritize);
3. **Focus:** Once the tasks have been prioritized from the previous screen (**Prioritize**), they are ready to be scheduled and worked on, which is the purpose of this screen — it displays the current working list of tasks in a digestible format;
4. **Review:** Periodically (daily, weekly, etc.), the application will ask the user to review the tasks that remain to be completed to ensure that they are still relevant and that they are still a priority.

These core screens are accessible at all times from the tab bar to allow for easy navigation between them. The application will also have a few other screens that are not part of the core workflow but that are still important for the user experience:

1. **About:** This screen will display information about the application, such as the purpose behind each aspect as well as the version number;
2. **Help:** There will be a help system embedded in this screen to help users understand how to use the application and how to get the most out of it, which will be fully searchable;

⁴To take a peek at some representative code snippets highlighted from the implementation, please refer to Appendix D.

3. **Recycle bin:** Sometimes, tasks will be added to ProductiviDo by mistake or they will be completed before they are scheduled. This screen will allow the user to review and restore these tasks if they are still relevant;
4. **Accomplishments:** This screen will display a list of tasks that have been completed and will allow the user to review them and reflect on their accomplishments;
5. **Settings:** This screen will allow the user to customize the application to their liking, such as changing the theme, determining default values on new tasks, etc.

These secondary screens are accessible from each core screen in the top-right corner to toggle the presence of a “drawer” menu. The menu can also be revealed by swiping from right to left on the screen. The menu will be hidden by default to keep the interface as clean as possible and to avoid cluttering it with unnecessary information. The menu will be revealed by default on the first launch of the application to help users discover the features available to them.

3.4 Libraries and services

At a high level, the application will be built using React Native to support both Android and iOS platforms. In terms of software user-facing dependencies, the application will use the following libraries:

- **react-native-async-storage/async-storage:** The application will use this library to store data locally on the device;
- **react-native-community/datetimepicker:** This library will be used to display the date and time pickers;
- **React Navigation:** This library will be used to implement the navigation system;
- **Redux:** This library will be used to manage the state of the application;
- **NativeBase:** it will be used to implement the user interface;
- **redux-persist:** This library will be used to persist the state of the application locally on the device;
- **redux-thunk:** This library will be used to implement asynchronous actions in the application;
- **react-native-collapsible:** This library will be used to implement all the collapsible sections to provide progressive disclosure;
- **react-native-extended-stylesheet:** This library will be used to implement the theming system;
- **nandorojo/anchor:** This library will be used to implement the anchor system so that users can jump from one section to another in the help system.

The application will also use the following libraries to support the development process:

- **Eslint:** This library will be used to enforce a consistent coding style;
- **Babel:** This library will be used to transpile the code to support older versions of JavaScript;
- **Jest:** This library will be used to implement unit tests;
- **react-native-testing-library:** This library will be used to implement integration tests;
- **react-native-dotenv:** This library will be used to manage environment variables;
- **react-test-renderer:** This library will be used to implement snapshot tests;

- **TypeScript**: It will ensure type checking is available app-wide and reduce the number of bugs.

On the backend side, the application will make use mainly of the following services:

- **Firebase**: This service will be used to implement the authentication system needed to deal with Google accounts as an optional add-on;
- **Google Calendar**: This service will be used to implement the calendar system;
- **SQLite**: It will be used to implement the database system locally and will be backed up to the cloud.

3.5 Plan of work

In a nutshell, the plan of work will be followed as shown in Figure 1.

3.6 Testing and evaluation

The project will be tested primarily using the following methods:

- **Unit tests**: Unit tests will be implemented to ensure that each component of the application works as expected. These tests will be implemented using `Jest` and `react-test-renderer`.
- **Integration tests**: Integration tests will be implemented to ensure that the application works as expected when all the components are combined. These tests will be implemented using the apt `react-native-testing-library`.
- **User and usability testing**: User testing will be implemented to ensure that the application is easy to use and that it meets the needs of the users. These tests will be implemented using a combination of user interviews and surveys.
- **Performance, load and stress testing**: Performance testing will be implemented to ensure that the application is fast and responsive. These tests will be performed by simulating the presence of a large number of tasks and events.
- **Security testing**: Security testing will be implemented to ensure that the application is secure, most definitely when it comes to protecting user's data in the cloud to avoid data breaches.
- **Regression testing**: Regression testing will be implemented to ensure that the application does not break when new features are added, using a mixture of automated `git bisect` scripts to locate anomalies and by automatically running test suites with Git pre-commit hooks [120].
- **Compatibility testing**: Compatibility testing will be implemented to ensure that the application works on all the supported platforms.
- **Accessibility testing**: Accessibility testing will be implemented to ensure that the application is accessible to all users.

4 Implementation

4.1 Overview

At its core, ProductiviDo aims to provide a system that facilitates “*getting things done*” in the manner of the famously successful productivity system GTD, created and shared with millions of people by David Allen [39]. The system is based on the following principles, worth expanding on as it bears strong resemblance with the final design of the application:

- **Capture:** All tasks and events should be captured in a single place, so that they can be easily accessed and managed. This is achieved with the “**Inscribe**” screen, the first step in the ProductiviDo system.
- **Clarify:** All tasks and events should be clarified, so that they can be easily understood and prioritized. This is achieved in the “**Inscribe**” screen, where the user can add a description to the task or event along with necessary metadata to help classify and prioritize the data being added.
- **Organize:** All tasks and events should be organized, so that they can be easily accessed and managed. While this GTD step does not map directly to a step in ProductiviDo, the “**Focus**” screen serves a similar purpose in that it allows the user to organize tasks and events into a list of immediately actionable items that are easy to reference.
- **Reflect:** All tasks and events should be reflected upon, so that they can be easily understood and prioritized. With ProductiviDo, this is done in the “**Review**” screen, which invites users to review tasks logically according to their deadline, priority, and other metadata.
- **Engage:** All tasks and events should be engaged with, so that they can be easily accessed and managed. ProductiviDo deals with this GTD step in a simpler way via the “**Prioritize**” screen and the “**Focus**” screen, which allow users to prioritize tasks and events and then focus on the most important ones.

4.2 Features implemented: a visual tour

As can be observed in Appendix C, the application has the following features:

- **Add a task** (see figure 10a): The user can add tasks by navigating to the “**Inscribe**” screen and filling in the required fields. The user can also add a deadline date and time, a duration, a location, a priority to the task and so on. By choosing the type of task as “**Calendar**” and filling in the required fields, the task will be added to the user’s Google Calendar. This screen hides any navbar and tab bar to maximize the screen real estate.
- **Prioritize a task** (see figure 10b): The user can prioritize tasks by navigating to the “**Prioritize**”, where filtering options are available to help the user find the task to prioritize. The main task action that can be performed on this screen is to “focus” a task, effectively moving it to the “**Focus**” screen.
- **Focus on a task** (see figure 10c): The user can focus on a task by navigating to the “**Focus**” screen, where the user can see a list of tasks that are immediately actionable. The user can also filter the tasks to focus on. The main action that can be performed on this screen is to “complete” a task, effectively moving it to the “**Accomplishments**” screen.
- **Review tasks** (see figure 10d): The user can review tasks by navigating to the “**Review**” screen, where the user can see a list of tasks that are overdue, due soon (e.g., today, tomorrow, this week), due later (e.g., next week, next month, next year) as well as tasks that are “old” (i.e., tasks that have been added to the system more than a month ago relative to today’s date but have not been worked

on yet). This screen features “*quick actions*” that are relevant to the tasks being reviewed. These actions provide the ability to reschedule a task, update its current state (i.e., move it from and to the **Prioritize** and **Focus** screens) and to delete a task without having to open the task details screen to edit it.

- **Filter tasks to review** (see figure 10e): The user can filter tasks to review by navigating to the “**Review**” screen and selecting the filter options. The user can filter tasks by one of the categories “**Overdue**”, “**Soon**”, “**Later**”, “**Old**” and by all of these at once when no filters are applied.
- **Edit task details** (see figure 10f): Once a task has been added to the system, it can be edited from any of the main screens (i.e., “**Prioritize**”, “**Focus**”, “**Review**”). The user can edit all the same properties as when adding a task as well as add new ones not present when the task was added. When a task is of type “**Calendar**”, its related details will also be updated on the user’s Google Calendar.
- **About screen** (see figure 10h): while the **About** screen does not provide any functionality related to tasks, it allows users to learn more about the application. The screen includes a link to the application’s website, a link to a dynamic feedback form to help ProductiviDo become a better fit for its users, an email address to contact the developers (which opens the default email client on the user’s device) and other less frequently accessed information, including a link to the GitHub repository of the project and a list of open source libraries used to create the product.
- **Help screen** (see figure 11i): the application has been designed to avoid relying on documentation in order to be used. However, the **Help** screen is a useful resource for users who want to learn more about the concepts involved in the application. It describes possible workflows and guides users through the application’s main screens so they will be able to use it even more effectively.
- **Recycle bin screen** (see figure 11j): ProductiviDo makes it very easy to access and manage tasks that have been deleted by the user. The **Recycle bin** screen allows users to restore deleted tasks (which are sent back to the **Prioritize** screen for the next round of prioritization) or to permanently delete them by emptying the recycle bin.
- **Accomplishments screen** (see figure 11k): While the **Recycle bin** holds all deleted tasks, the **Accomplishments** screen holds all tasks that have been completed by the user from the **Focus** screen. The **Accomplishments** screen is a great way to keep track of the user’s progress and to celebrate the user’s accomplishments, yet it also allows removing completed tasks from the system if the user wishes to do so.
- **Settings screen** (see figure 11l): The **Settings** provided by ProductiviDo allow users to customize the application to their needs in a basic way — and all settings are persisted across reboots of the application or device. There is a dark mode available to protect the users’ eyes from screen brightness. Other settings include the ability to set default behaviours on new tasks, set the increment in minutes to be shown in the time picker, set a different locale to represent dates and times and some more options to toggle the display of certain elements in the application.
- **Filters** (see figure 11o): What ProductiviDo lacks in customizations, it makes up for in filters. The application allows users to filter tasks by type, status, priority, deadline date, deadline time, duration, location and more. The filters are applied in real-time, so the user can see the results of the filters as they are being applied. The filters can be applied to any of the main screens (i.e., “**Prioritize**”, “**Focus**”, “**Review**”) and can be combined to create more precise filters.
- **Google Calendar** (see figure 11p): One of the key tenets of ProductiviDo is to make it easy for users to manage their tasks and to have a visual window into their schedule, which is achieved by

integrating with the user’s primary calendar on Google Calendar. The application allows users to connect their Google Calendar account to ProductiviDo and to create calendar events from new or existing tasks. At the same time, this functionality remains entirely optional: users can choose to never opt in to the Google Calendar integration and can disconnect their Google account at any time, which will prevent any synchronization from happening between ProductiviDo and Google Calendar.

4.3 Techniques and methods

Beyond the libraries mentioned in section 3.4, the actual implementation of the application happened in the following way:

- **React Native bare workflow:** The application was developed using the React Native bare workflow, which allows for the use of native code in the application. This was necessary to implement the Google Calendar integration, which was not possible using the Expo workflow [91].
- **State management:** The application was architected in such a way that React Context was used to avoid “*props drilling*” and to provide a global state to the application, using data coming from the SQLite database loaded at launch [94]. All context that does not require persistence is managed by React Context, while the rest of the state (e.g., settings) is managed by Redux Toolkit [119].
- **TypeScript:** The app was developed using TypeScript, a typed superset of JavaScript that compiles to plain JavaScript [112]. TypeScript was used to ensure type safety and to provide a better developer experience with autocomplete and other features that help developers write code with fewer errors.
- **React Navigation** [118]: The navigation between screens uses a combination of stack and tab navigation, with the main screens being accessible through the tab navigation and the rest of the screens being accessible through the stack navigation. Additionally, a drawer navigation is used to provide access to all secondary screens via the traditional “hamburger” menu.
- **NativeBase:** This library was used to provide a consistent look and feel to the application and get access to components that are not available in React Native [95]. Many of its components were put to use in the application, including `ActionSheet`, `AlertDialog`, `Badge`, `FlatList`, `Modal`, `Switch` and `Transitions` to name a few.
- **Google Calendar API:** The application uses the Google Calendar API to allow users to connect their Google Calendar account to ProductiviDo and to create calendar events from new or existing tasks [99]. The API is used to create new events, to update existing events and to delete events. The API is also used to retrieve the user’s primary calendar and to retrieve the list of calendars available to the user.
- **SQLite:** The application uses SQLite to store all data locally on the device. The application uses the `expo-sqlite` library to interact with the SQLite database [90]. The database is initialized at launch and is updated whenever the user makes changes to the data.
- **Form validation:** The application uses the `yup` library to validate forms and to provide error messages to the user [105]. Coupled with `formik`, the library provides a great way to validate forms and to provide a consistent experience to the user [93].
- **Custom hooks and functional components:** This project explicitly embraces new React features such as custom hooks and functional components. The application uses custom hooks to encapsulate logic for all the main screens. The application also uses functional components to avoid the use of class components, which are considered legacy in React [92].

5 Evaluation: results and discussion

5.1 Successes

5.1.1 Planning and execution

Extensive research and preparation went into the project before writing a single line of production code. A multitude of potential implementations and concepts were explored in great details and the idea of the final product was refined from a very early stage, leading to a more focused development of the todo application. Because there existed a tangible vision of the desired outcome, it made it much simpler to define the database schema and the React context needed throughout the application to pass adequate data structures around. On a related note, doing all this work upfront helped to avoid wasting time learning libraries that would not entirely meet the needs of the application and to stop contemplating features that would not fit within the constraints of the chosen concept.

Another decision that proved to be quite useful was to postpone dealing with complicated services till much later during development. For instance, storing mock data as simple JSON allowed for quick iterations during development without having to manually add tasks one by one by using the application itself to do so. Once it came time to integrate database functionality, data was first stored as a temporary in-memory database for fast read/write speeds (because the task view, filtering and sorting was not optimized yet) and it was later persisted to disk. Likewise, external services (e.g., Firebase and Google Calendar) were only integrated until the internal application design was set in stone. Performance optimizations were favorably delayed until most of the logic was implemented, which gave ample time to test and refine the design, productively steering clear of premature optimization due to the changing nature of the application still being validated and tested by beta users.

5.1.2 Functionality achieved

One concept that served the final design well was a direct application of Fitt's law, which states that "*the time required to touch a target is a function of the target size and the distance to the target*", meaning that it is important to keep controls close and large when speed or accuracy matters [45]. As a concrete demonstration of this law, tasks can be added with the touch of a single button in the "**Inscribe**" button located in the tab bar, making it frictionless for users to add tasks and events to the application. This meets the need of being "*always on*", especially since all data is accessible locally without the need for an internet connection, allowing users to quickly get their priorities out of their mind and onto a digital device to allow for proper referencing later on. Other critical interactions such as focusing tasks and completing them can be done by tapping on a large icon next to each task.

The application allows visualizing all tasks at once in the "**Prioritize**" screen, which contains all necessary filters in order to make filtering and sorting through a growing list of incoming data not only feasible but also simple and practical. Indeed, the lack of hierarchical organization in the information architecture is absolutely intentional. This design decision allows flexibility with the use of filters without incurring a cognitive cost to understand how information is organized. As long as users are diligent with the metadata they input in the application (as they should if they wish to accomplish their goals), they will be able to easily find the information they need in the "**Prioritize**" screen. Excluding the hierarchical access to information, all other aspects of information architecture (location, alphabet, time, category) are available as filters to the users [46].

The "**Focus**" screen intuitively lets users know which tasks are most important to work on at any given time, sorting them by deadline, priority and other metadata, making the data "*always actionable*". Since

focused tasks are isolated from the rest of the data, users are able to focus on the task at hand without being distracted by other tasks. On this screen, users are able to complete tasks, which effectively disappear from the task list and are kept in the “**Accomplishments**” screen for later retrieval. This is a way to keep track of tasks that have been completed, which is a feature that is not available in the Google Calendar application and which complements its functionality very well. Additionally, this screen allows users to “restore” tasks, which performs the opposite action. Notably, it will prevent mistakes by letting users undo their actions; it will also allow users to review tasks and events that have been deleted and to permanently delete them as needed. This feature is not available in the Google Calendar application and contributes to making ProductiviDo a useful ally in this case so that events can be restored to the calendar view.

The final active step required to use ProductiviDo effectively, found in the “**Review**” screen, provides complimentary functionality to that found in most modern calendar applications. While a calendar is useful to get a sense of upcoming events, a review system is needed to stay on top of one’s priorities to make sure that time-insensitive yet important events do not slip through the cracks. The “**Review**” screen is a way to make sure that users are not only aware of the tasks and events they have scheduled, but also of the tasks and events that they have not scheduled but are important to them. Furthermore, reviewing one’s work periodically in an automatic way allows for better planning and will make sure that longstanding tasks are either updated, prioritized, delegated or removed.

5.2 Failures and limitations

The following points of failures are discussed in more details in section 6.2, where solutions for brighter times ahead are pictured.

5.2.1 Google Calendar integration

ProductiviDo solves real-world issues related to task management and scheduling in a robust manner, yet it does so partially. One major caveat to the current implementation is that it supports only Google Calendar, which severely limits the potential market the application can tap into. Because only Google Calendar was taken into account during this project, it is highly likely that the application will not be able to support other calendar applications without a major overhaul of the codebase. The application was thought to be a companion to Google Calendar from the ground up, which means that while it integrates seamlessly with it, it creates tight coupling between the two applications. There are in fact more issues to delve into when it comes to Google Calendar.

In particular, full synchronization of tasks and events is not supported in a scalable way. For one, Google Calendar API’s daily and monthly rates are limited: instead of calling that API on every little detail being updated in a task, the application should really try its best to batch requests so that heavy users will not be penalized if they happen to trigger many updates in their calendar or if Google decides to lower the number of requests that can be processed in a given amount of time. Another pitfall is that the application does not support recurring tasks as originally intended due to lack of human resources, which is a feature that is available in the Google Calendar application. While it is technically possible to implement this feature, more context would need to be stored in the application to avoid duplication of tasks in the database and to allow recurring tasks to be reset to their original state or updated as desired, all of which would need to communicate with the Google Calendar API in an efficient way.

There is, at the moment, no task tracking facility: a completed task is simply removed from the list of tasks by updating its status in the database. This is a considerable limitation, since it prevents users from creating tasks that are recurring, such as weekly meetings or monthly bills.

5.2.2 Unit testing

One clear oversight in the development process was the lack of unit testing. While the application was profusely tested in a manual way, it would have been much more efficient to have had a suite of unit tests that would have allowed for a more sturdy and scalable application. ProductiviDo comprises a few essential unit tests regarding the Google Calendar integration, but it is nowhere near the level of testing that would be required to ensure that the application is robust and that it will not break in the future. Amongst some of the types of tests that should have been implemented are unit tests for the database, unit tests for the application's business logic, snapshot testing and testing of asynchronous API calls through mocking.

In hindsight, the reason more tests did not make it into the final application is that the development process was not agile enough to allow for the time required to write unit tests and there have been too many side experiments that have taken priority over the core functionality of the application. Of equal relevance is the fact that unanticipated issues have arisen during the development process, which have required more time and effort to be spent on fixing them rather than on writing unit tests. It is also important to keep in mind that unit testing is not a one-time activity: it should be done continuously throughout the development process, more akin to the test-driven development (TDD) workflow [145]. Failing in that department was a major setback for the application, since it will make it harder to maintain and to scale in the future and it likely contributed to the fact that more time was spent on debugging than was necessary.

5.2.3 Advanced features powered by AI

Besides the issues already raised in this section, there are a few other features that would have optimized the user experience, but which were not implemented due to lack of time as otherwise the scope of the project would have fallen appreciably out of reasonable proportions. For instance, the application does not support batch editing of tasks, which is a feature that is available in many other task managers in one form or another. It also does not support location-based notifications, which would have been useful to send reminders to users based on their location.

The application does not make use of advancements in machine learning: learning users' preferences over time to provide helpful shortcuts when creating new tasks could have been a fantastic time saver. ProductiviDo could have stored task templates such that creating recurring tasks would be faster and more intuitive by providing logical default values based on what the algorithm thinks is going to be needed next. Similarly, the app could add locations, people, and other properties to tasks once it figures out that certain actions only happen in given locations or with the same people repeatedly.

Yet another feature that would have proven handy would be to provide time-based notifications based on predictions from past data derived from tracking user movements and time taken to complete specific types of tasks. For instance, if a user has a task that is due at 9:00 AM, the app could send a notification to the user's phone at 8:45 AM if the user is at home, but at 8:30 AM if the user is at work once the travel time could reliably be estimated from prior activities.

6 Conclusion

6.1 Salient takeaways

ProductiviDo set out to fathom the needs of the modern-day, demanding user and to provide a novel solution to the problem of task management and scheduling. The application was designed from the start to work alongside Google Calendar, which is a popular calendar application that is used by many millions of people around the world. The application was designed to be a simple and intuitive task manager that would allow users to create tasks and events, to schedule them, and to track their progress. While there is room for improvement, ProductiviDo goes far beyond a proof of concept or “*Goldilocks Quality*” and shows how it can be used by anyone who wants to manage their tasks and schedule more efficiently, as it relies on strong scientific foundations that were dispersed throughout this work as convincing evidence backing up such a claim [40, 139].

One of the key takeaways from this work is that despite a crowded task management digital market, innovation can take place by taking inspiration from different fields and by applying them to the problem at hand. In the case of ProductiviDo, the needs of business users and productivity seekers alike were distilled into an almost mechanical process, where the application flow is extremely simple and strives to put forward in a visually sensible manner all the required functionality to get work done without friction: this meant taking a few assumptions about users for granted, knowingly reducing the user base to better serve those who will resonate with the offering.

There are a few dominant players in the industry, albeit niches are still well underserved. Almost every software application out there in this space endeavors to serve customers in a very generic way, but it could be flourishingly argued that a task manager could be designed to serve a specific industry. For instance, a task manager for a construction company could be designed to be more intuitive for construction workers, who are not necessarily computer savvy, and could be designed to be more efficient for project managers, who are likely to be more tech-savvy. The same could be said for a task manager for a law firm, a task manager for a restaurant, or a task manager for a school. The possibilities are endless, and the key is to understand the needs of the users and to design the application to be as intuitive as possible. Even though “task templates” can, up to a point, be used to achieve this, nothing competes with a well-designed application that is tailored to the needs of a well-known target audience. Such applications will by definition be adopted within a given niche and will dispense unique value to the users. A generic task manager may be suitable for simple projects and may require fiddling around with the settings to get it to work for a specific use case, but a task manager dedicated to culinary chefs will natively understand and care for concepts such as “ingredients” and “recipes”, which will be baked into the product (no pun intended).

6.2 Improvements and future research

Even though Google users abound, using exclusively Google Calendar as an calendar integration is not a universal solution. In the future, it would be interesting to explore the possibility of integrating other calendar applications such as Apple Calendar, Outlook, and others. This would allow users to use the application with their preferred calendar application and would also allow for a more seamless integration with the calendar application, which would be a great benefit to users.

Expanding on the notion of “delegatees” explored in this project, a task manager with built-in collaboration would be welcome to further satisfy the needs of business users. This is because a calendar can become social and collaborative by itself, and it is not uncommon for colleagues or friends to share their calendars with others. This is especially true for business users, who often need to coordinate their activities with other members of their team. It would certainly not be far-fetched to have business teams be formed

and given access to their shared task assignments and calendars. This would allow for a more efficient and transparent way of working together, and it would also allow for a more efficient way of tracking the progress of a project and would pave the way to generating insightful analytics dashboards, another place where machine learning would shine by finding patterns and making it possible to optimize overlapping schedules and projects based on people's availability, area of expertise and so on.

Recent reports have shown that job creation and entrepreneurship is at its highest peak and that the United States of America is leading the way in terms of new business creation, with literally millions of companies being born each year [63]. This is a great sign for the economy and it shows numerous opportunities in the business world when considering task managers embarking on a journey to become a collaborative platform for teams and businesses within a specific line of work. In the meantime, ProductiviDo comes to the rescue of the millions of people who are looking for simplicity and intuitiveness in a task manager to get their work done with less technological grinding.

7 References

Articles

- [1] A. Achtziger, P. M. Gollwitzer, and P. Sheeran. Implementation intentions and shielding goal striving from unwanted thoughts and feelings. *Personality and Social Psychology Bulletin*, 34(3):381–393, 2008.
- [2] Amponsah-Tawiah. Time management: presenteeism versus management-by-objectives. *International Journal of Law and Management*, 60(6):1470–1484, 2018. <https://doi.org/10.1108/IJLMA-01-2018-0010>, Accessed: 2022-12-24.
- [3] M. Ancona, M. Beyeler, M. Gross, and T. Günther. Minetime insight: visualizing meeting habits to promote informed scheduling decisions. *IEEE Transactions on Visualization and Computer Graphics*, PP:1–1, Sept. 2019. DOI: 10.1109/TVCG.2019.2941208.
- [4] U. C. Bayer, A. Achtziger, P. M. Gollwitzer, and G. B. Moskowitz. Responding to subliminal cues: do if-then plans facilitate action preparation and initiation without conscious intent? *Social Cognition*, 27(2):183–201, 2009.
- [5] S. Bertsch, B. J. Pesta, R. Wiscott, and M. A. McDaniel. The generation effect: a meta-analytic review. *Memory & Cognition*, 35(2):201–210, 2007. DOI: 10.3758/BF03193441. URL: <https://doi.org/10.3758/BF03193441>.
- [6] M. Bieleke and L. Keller. Individual differences in if-then planning: insights from the development and application of the if-then planning scale (itps), Nov. 2020. DOI: 10.31234/osf.io/2efnv.
- [7] M. Bieleke, L. Keller, and P. M. Gollwitzer. If-then planning. *European Review of Social Psychology*, 32(1):88–122, 2021.
- [8] K. W. Brown and R. M. Ryan. The benefits of being present: mindfulness and its role in psychological well-being. *Journal of personality and social psychology*, 84(4):822, 2003.
- [9] D. Coviello, A. Ichino, and N. Persico. Time allocation and task juggling. *American Economic Review*, 104(2):609–23, Feb. 2014. DOI: 10.1257/aer.104.2.609. URL: <https://www.aeaweb.org/articles?id=10.1257/aer.104.2.609>.
- [10] S. Diefenbach and A. Müssig. Counterproductive effects of gamification: an analysis on the example of the gamified task manager habitica. *International Journal of Human-Computer Studies*, 127:190–210, 2019.
- [11] R. Dobbins. Manage your time well. *Management Research News*, 20(5):15–24, 1997. <https://doi.org/10.1108/eb028561>, Accessed: 2022-12-24.
- [12] J. T. Doerflinger, T. Martiny-Huenger, and P. M. Gollwitzer. Planning to deliberate thoroughly: if-then planned deliberation increases the adjustment of decisions to newly available information. *Journal of Experimental Social Psychology*, 69:1–12, 2017.
- [13] A. Fritzsche, B. Schlier, G. Oettingen, and T. Lincoln. Mental contrasting with implementation intentions increases goal-attainment in individuals with mild to moderate depression. *Cognitive Therapy and Research*, 40, Aug. 2016. DOI: 10.1007/s10608-015-9749-6.
- [14] P. Gollwitzer and G. Oettingen. Planning promotes goal striving. *Handbook of Self-Regulation: Research, Theory, and Applications*, Jan. 2011.
- [15] W. Hofmann, M. Luhmann, R. R. Fisher, K. D. Vohs, and R. F. Baumeister. Yes, but are they happy? effects of trait self-control on affective well-being and life satisfaction. *Journal of personality*, 82(4):265–277, 2014.
- [16] R. Huang. The effects of calendar apps on students' perceived stress. *Hanken School of Economics Helsinki*, 2020. URL: https://helda.helsinki.fi/dhanken/bitstream/handle/10227/402133/Huang_Ruolin.pdf. Accessed: 2022-12-24.

- [17] S.-C. Huang and J. Aaker. It's the journey, not the destination: how metaphor drives growth after goal attainment. *Journal of personality and social psychology*, 117(4):697, 2019.
- [18] D. R. Kennedy and A. L. Porter. The illusion of urgency. *American Journal of Pharmaceutical Education*, 86(7), 2022. ISSN: 0002-9459. DOI: 10.5688/ajpe8914. eprint: <https://www.ajpe.org/content/86/7/8914.full.pdf>. URL: <https://www.ajpe.org/content/86/7/8914>.
- [19] D. Kirsh. A few thoughts on cognitive overload. *Intellectica*, 1:19–51, Jan. 2000. DOI: 10.3406/intel.2000.1592.
- [20] R. Ludwig, S. Srivastava, E. Berkman, and B. Donnellan. Planfulness: a process-focused construct of individual differences in goal achievement. *Collabra: Psychology*, 4, July 2018. DOI: 10.1525/collabra.136.
- [21] E. Masicampo and R. F. Baumeister. Committed but closed-minded: when making a specific plan for a goal hinders success. *Social Cognition*, 30(1):37, 2012.
- [22] E. Masicampo and R. F. Baumeister. Consider it done! plan making can eliminate the cognitive effects of unfulfilled goals. *Journal of personality and social psychology*, 101(4):667, 2011.
- [23] A. N. Meyer, L. E. Barton, G. C. Murphy, T. Zimmermann, and T. Fritz. The work life of developers: activities, switches and perceived productivity. *IEEE Transactions on Software Engineering*, 43(12):1178–1193, 2017. DOI: 10.1109/TSE.2017.2656886.
- [24] J. Pérez-Quiñones Manuel A. Rode. You've got mail! calendar, weather and more: customizable phone access to personal information. *Department of Computer Science, Virginia Tech*, 2004. URL: <http://hdl.handle.net/10919/20178>. Accessed: 2022-12-24.
- [25] H. Riddell, C. Sedikides, F. Gucciardi, B. Jackson, C. Thøgersen-Ntoumani, and N. Ntoumanis. Goal motives and mental contrasting with implementation intentions facilitate strategic goal persistence and disengagement. *Journal of Applied Social Psychology*, 52, Aug. 2022. DOI: 10.1111/jasp.12915.
- [26] Z. Rusou. The psychology of task management: the smaller tasks trap. *Judgment and Decision Making, APA PsycInfo*, 15(4):586–599, 2020. DOI: 10.1257/aer.104.2.609. URL: <https://www.aeaweb.org/articles?id=10.1257/aer.104.2.609>.
- [27] K. M. Sheldon and S. Lyubomirsky. How to increase and sustain positive emotion: the effects of expressing gratitude and visualizing best possible selves. *The journal of positive psychology*, 1(2):73–82, 2006.
- [28] F. F. Sniehotta, R. Schwarzer, U. Scholz, and B. Schüiz. Action planning and coping planning for long-term lifestyle change: theory and assessment. *European Journal of Social Psychology*, 35(4):565–576, 2005.
- [29] A. M. Sools, T. Tromp, and J. H. Mooren. Mapping letters from the future: exploring narrative processes of imagining the future. *Journal of health psychology*, 20(3):350–364, 2015.
- [30] G. N. Tonietto and S. A. Malkoc. The calendar mindset: scheduling takes the fun out and puts the work in. *Journal of Marketing Research*, 53(6):922–936, 2016.
- [31] A. Toth, B. Wisse, and A. Uatkan. The impact of goal attainment and goal importance on satisfaction with life – a polynomial regression and response surface analysis. *Mentálhigiéné és Pszichoszomatika*, 19:80–101, Mar. 2018. DOI: 10.1556/0406.19.2018.004.
- [32] M. Tungare, M. Pérez-Quiñones, and A. Sams. An exploratory study of personal calendar use. *arXiv preprint arXiv:0809.3447*, 2008.
- [33] E. Vassena, J. Deraeve, and W. H. Alexander. Task-specific prioritization of reward and effort information: novel insights from behavior and computational modeling. *Cognitive, Affective, & Behavioral Neuroscience*, 19(3):619–636, 2019. DOI: 10.3758/s13415-018-00685-w. URL: <https://doi.org/10.3758/s13415-018-00685-w>.
- [34] A. A. Verhoeven, M. A. Adriaanse, D. T. De Ridder, E. De Vet, and B. M. Fennis. Less is more: the effect of multiple implementation intentions targeting unhealthy snacking habits. *European Journal of Social Psychology*, 43(5):344–354, 2013.

- [35] J. Wajcman. How silicon valley sets time. *New Media & Society*, 21(6):1272–1289, 2019. DOI: 10.1177/1461444818820073. eprint: <https://doi.org/10.1177/1461444818820073>. URL: <https://doi.org/10.1177/1461444818820073>.
- [36] K. M. Werner, H. Sjåstad, M. Milyavskaya, and W. Hofmann. Planning for success: trait self-control predicts goal attainment through the use of implementation intentions, 2020.
- [37] G. Westerhof and E. Bohlmeijer. Life stories and mental health: the role of identification processes in theory and interventions. *Narrative Matters*, 2(1):106–128, 2012.
- [38] J. Wong, M. Baars, M. He, B. de Koning, and F. Paas. Facilitating goal setting and planning to enhance online self-regulation of learning. *Computers in Human Behavior*, 124:106913, June 2021. DOI: 10.1016/j.chb.2021.106913.

Books

- [39] D. Allen. *Getting things done: The art of stress-free productivity*. Penguin, 2015.
- [40] B. K. Jake Knapp John Zeratsky. *Sprint: How to Solve Big Problems and Test New Ideas in Just Five Days*. Simon & Schuster, Illustrated edition, 2016. ISBN: 0593076117.
- [41] J. P. T. James E. Maddux. *Social Psychological Foundations of Clinical Psychology*. The Guilford Press, 2010. ISBN: 1606236792.
- [42] A. V. Jenifer Tidwell Charles Brewer. *Designing Interfaces — Patterns for Effective Interaction Design*. O'Reilly Media, Third edition, 2020. ISBN: 1492051969.
- [43] W. P. Jones and J. Teevan. *Personal information management*, volume 14. University of Washington Press Seattle, 2007. ISBN: 0274715406.
- [44] E. Lau. *The Effective Engineer*. Effective Bookshelf, 2015. ISBN: 0996128107. <https://effectiveengineer.com/>.
- [45] W. Lidwell. *The Pocket Universal Principles of Design: 150 Essential Tools for Architects, Artists, Designers, Developers, Engineers, Inventors, and Makers*. Rockport Publishers, Illustrated edition, 2015. ISBN: 1631590405.
- [46] E. N. McKay. *UI is Communication: How to Design Intuitive, User Centered Interfaces by Focusing on Effective Communication*. Morgan Kaufmann, First edition, 2013. ISBN: 0123969808.
- [47] C. A. H. Michael D. Mumford. *Leader Thinking Skills – Capacities for Contemporary Leadership*. Routledge, 2020. ISBN: 9781138284333.
- [48] C. Newport. *The Time-Block Planner: A Daily Method for Deep Work in a Distracted World*. Penguin Business, New York, NY, 2020. ISBN: 0593192052.
- [49] P. G. Pablo Perea. *UX Design for Mobile*. Packt Publishing Ltd, 2017. ISBN: 1787283429.
- [50] T. A. Pychyl. *Solving the procrastination puzzle: a concise guide to strategies for change*. Prince Frederick, MD, 2015. ISBN: 0399168125.
- [51] D. L. Schacter. *The Seven Sins of Memory: How the Mind Forgets and Remembers*. Houghton Mifflin Harcourt, 2001. ISBN: 0618219196.

Conference proceedings

- [52] T. Bebensee, I. van de Weerd, and S. Brinkkemper. Binary priority list for prioritizing software requirements. In *International working conference on requirements engineering: foundation for software quality*, pages 67–78. Springer, 2010.
- [53] V. Bellotti, B. Dalal, N. Good, P. Flynn, D. G. Bobrow, and N. Ducheneaut. What a to-do: studies of task management towards the design of a personal task list manager. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, pages 735–742, Vienna, Austria.

- Association for Computing Machinery, 2004. ISBN: 1581137028. DOI: 10.1145/985692.985785. URL: <https://doi.org/10.1145/985692.985785>.
- [54] H. Bratterud, M. Burgess, B. T. Fasy, D. L. Millman, T. Oster, and E. (Sung. The sung diagram: revitalizing the eisenhower matrix. In A.-V. Pietarinen, P. Chapman, L. Bosveld-de Smet, V. Giardino, J. Corder, and S. Linker, editors, *Diagrammatic Representation and Inference*, pages 498–502, Cham. Springer International Publishing, 2020. ISBN: 978-3-030-54249-8.
- [55] V. M. González and G. Mark. Managing currents of work: multi-tasking among multiple collaborations. In H. Gellersen, K. Schmidt, M. Beaudouin-Lafon, and W. Mackay, editors, *ECSCW 2005*, pages 143–162, Dordrecht. Springer Netherlands, 2005. ISBN: 978-1-4020-4023-8.
- [56] M. Haraty and J. McGrenere. Designing for advanced personalization in personal task management. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems*, DIS '16, pages 239–250, Brisbane, QLD, Australia. Association for Computing Machinery, 2016. ISBN: 9781450340311. DOI: 10.1145/2901790.2901805. URL: <https://doi.org/10.1145/2901790.2901805>.
- [57] N. S. Jyothi and A. Parkavi. A study on task management system. In *2016 International Conference on Research Advances in Integrated Navigation Systems (RAINS)*, pages 1–6, 2016. DOI: 10.1109/RAINS.2016.7764421.
- [58] D. L. Kappen, J. Johannsmeier, and L. E. Nacke. Deconstructing 'gamified' task-management applications. In *Proceedings of the First International Conference on Gameful Design, Research, and Applications*, pages 139–142, 2013.
- [59] G. Mark, D. Gudith, and U. Klocke. The cost of interrupted work: more speed and stress. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 107–110, Florence, Italy. Association for Computing Machinery, 2008. ISBN: 9781605580111. DOI: 10.1145/1357054.1357072. URL: <https://doi.org/10.1145/1357054.1357072>.
- [60] Y. Zhang, S. Jauhar, J. Kiseleva, R. White, and D. Roth. Learning to decompose and organize complex tasks. In pages 2726–2735, Jan. 2021. DOI: 10.18653/v1/2021.nacl-main.217.

Institutional publications

- [61] Fortune Business Insights. Task Management Software Market Size, Share & Industry Analysis, By Component (Software, Services), By Organization Size (SMEs and Large Enterprises), By Deployment Type (Cloud and On-Premises), By End-User (Banking, Financial Services, and Insurance (BFSI), IT and Telecommunications, Retail, Healthcare, Manufacturing, Travel and Hospitality, Media and Entertainment, and Others) and Regional Forecast, 2019-2026, Fortune Business Insights, 2020. URL: <https://www.fortunebusinessinsights.com/task-management-software-market-102249>. Accessed: 2022-12-24.
- [62] IndustryARC. Task Management Software Market - Industry Analysis, Market Size, Share, Trends, Application Analysis, Growth and Forecast 2021 - 2026, IndustryARC, 2018. URL: <https://www.industryarc.com/Research/Task-Management-Software-Market-Research-500596>. Accessed: 2022-12-24.
- [63] Kauffman Indicators of Entrepreneurship. Job Creation by Firm Age: Recent Trends in the United States, Kauffman Indicators of Entrepreneurship, 2022. URL: <https://indicators.kauffman.org/reports>. Accessed: 2023-03-25.
- [64] Mordor Intelligence. Task management software market - growth, trends, covid-19 impact, and forecasts (2022 - 2027), Mordor Intelligence, 2022. URL: <https://www.mordorintelligence.com/industry-reports/task-management-software-market>. Accessed: 2022-12-24.
- [65] Reclaim.ai. Task Management Trends Report: +200 Stats on Managers vs. Individual Contributors, Reclaim.ai, 2020. URL: <https://reclaim.ai/blog/task-management-trends-report>. Accessed: 2022-12-24.

- [66] Statista Research Department. Number of smartphone subscriptions worldwide from 2016 to 2021, with forecasts from 2022 to 2027, Statista Research Department, 2022. URL: <https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/>. Accessed: 2022-12-23.
- [67] Verified Market Research. Task Management Software Market Size And Forecast, Verified Market Research, 2022. URL: <https://www.verifiedmarketresearch.com/product/task-management-software-market/>. Accessed: 2022-12-24.

Online

- [68] Airtable. Connect at scale with airtable enterprise. 2022. URL: <https://www.airtable.com/enterprise>. Accessed: 2022-12-26.
- [69] Airtable. Creating recurring records using automations. 2022. URL: <https://support.airtable.com/docs/creating-recurring-records-using-automations>. Accessed: 2022-12-26.
- [70] Airtable. Guide to filtering records using conditions. 2022. URL: <https://support.airtable.com/docs/guide-to-filtering-records-using-conditions>. Accessed: 2022-12-26.
- [71] Airtable. Integrating airtable with external calendar applications. 2022. URL: <https://support.airtable.com/docs/integrating-airtable-with-external-calendar-applications>. Accessed: 2022-12-23.
- [72] Alberto N G. Alberto n g.'s review of google calendar. 2022. URL: <https://www.capterra.com/p/213340/Google-Calendar/reviews/3574851/>. Accessed: 2022-12-24.
- [73] Amir Salihefendic. Todoist. 2007. URL: <https://todoist.com/>. Accessed: 2022-12-22.
- [74] Annie C., Director for Organization Development Enterprise(> 1000 emp.) It's okay but the user needs to further study the platform to understand the dynamics of clickup. 2022. URL: <https://www.g2.com/products/clickup/reviews/clickup-review-6636506>. Accessed: 2022-12-26.
- [75] Apple. Eventkit. 2022. URL: <https://developer.apple.com/documentation/eventkit>. Accessed: 2022-12-24.
- [76] Asana. A better online to-do list app for work. 2022. URL: <https://asana.com/uses/to-do-list>. Accessed: 2022-12-26.
- [77] Asana. Asana's tech stack: how we build a collaborative app for teams of all sizes. 2022. URL: <https://blog.asana.com/2022/02/asana-tech-stack/>. Accessed: 2022-12-26.
- [78] Asana. Google calendar asana. 2022. URL: <https://asana.com/apps/google-calendar>. Accessed: 2022-12-23.
- [79] Asana. Planning with asana calendar. 2022. URL: <https://asana.com/guide/team/onboard/asana-calendar>. Accessed: 2022-12-23.
- [80] Claire Haidar. The power of the master list. 2022. URL: <https://www.lifehack.org/articles/productivity/the-power-of-the-master-list.html>. Accessed: 2023-01-01.
- [81] ClickUp. Clickapps. 2022. URL: <https://clickup.com/features>. Accessed: 2022-12-26.
- [82] ClickUp. Clickup features. 2022. URL: <https://clickup.com/features>. Accessed: 2022-12-26.
- [83] ClickUp. Clickup feedback. 2022. URL: <https://clickup.canny.io>. Accessed: 2022-12-26.
- [84] ClickUp. Google calendar syncing. 2022. URL: <https://help.clickup.com/hc/en-us/articles/6336507264663-Google-Calendar-Syncing>. Accessed: 2022-12-23.
- [85] ClickUp. Keep work on schedule with calendar view. 2022. URL: <https://clickup.com/features/calendar-view>. Accessed: 2022-12-23.
- [86] Colleaga. The master list's purpose. 2022. URL: <https://www.colleaga.org/article/master-lists-purpose>. Accessed: 2023-01-01.
- [87] Diane K. Diane k.'s review of asana. 2021. URL: <https://www.capterra.com/p/184581/Asana-PM/reviews/2890300/>. Accessed: 2022-12-24.
- [88] Dustin Moskovitz, Justin Rosenstein. Asana. 2008. URL: <https://asana.com/>. Accessed: 2022-12-22.

- [89] EasyCalendar. Create zoom meeting links for your appointments. 2022. URL: <https://easycalendar.com/appointment-scheduler-zoom>. Accessed: 2022-12-24.
- [90] Expo. Expo sqlite. 2023. URL: <https://docs.expo.dev/versions/latest/sdk/sqlite/>. Accessed: 2023-03-26.
- [91] Expo. Workflows. 2023. URL: <https://docs.expo.dev/introduction/managed-vs-bare/>. Accessed: 2023-03-26.
- [92] Facebook. Migrating a simple component from a class to a function. 2023. URL: <https://react.dev/reference/react/Component#migrating-a-simple-component-from-a-class-to-a-function>. Accessed: 2023-03-26.
- [93] Formium. Formik. 2023. URL: <https://formik.org/>. Accessed: 2023-03-26.
- [94] freeCodeCamp. How to avoid prop drilling with the react context api. 2023. URL: <https://www.freecodecamp.org/news/avoid-prop-drilling-with-react-context-api/>. Accessed: 2023-03-26.
- [95] GeekyAnts. Nativebase. 2023. URL: <https://nativebase.io/>. Accessed: 2023-03-26.
- [96] Gina Trapani. Todo.txt. 2020. URL: <http://todotxt.org/>. Accessed: 2022-12-26.
- [97] Google. Add google calendar events to apple calendar. 2022. URL: <https://support.google.com/calendar/answer/99358?hl=en>. Accessed: 2022-12-24.
- [98] Google. Add meeting notes to google calendar events. 2022. URL: <https://support.google.com/docs/answer/11324079?hl=en>. Accessed: 2022-12-24.
- [99] Google. Google calendar api overview. 2023. URL: <https://developers.google.com/calendar/api/guides/overview>. Accessed: 2023-03-26.
- [100] Google. Google calendar api overview. 2022. URL: <https://developers.google.com/calendar/api/guides/overview>. Accessed: 2022-12-24.
- [101] Harvard Business Review. How ceos manage time. 2018. URL: <https://hbr.org/2018/07/how-ceos-manage-time>. Accessed: 2022-12-24.
- [102] Howie Liu, Emmett Nicholas, Andrew Ofstad. Airtable. 2012. URL: <https://www.airtable.com/>. Accessed: 2022-12-22.
- [103] iDoneThis. How to finally make peace with your to-do list. 2014. URL: <http://blog.idonethis.com/unplanned-tasks/>. Accessed: 2022-12-24.
- [104] Ivan Zhao. Notion. 2013. URL: <https://www.notion.so/>. Accessed: 2022-12-25.
- [105] Jason Quense. Yup. 2023. URL: <https://github.com/jquense/yup>. Accessed: 2023-03-26.
- [106] Joel Spolsky. Trello. 2011. URL: <https://trello.com/>. Accessed: 2022-12-22.
- [107] Joshua W., Lead Videographer, Small-Business (50 or fewer emp.) Incredible product. incredibly buggy. 2022. URL: <https://www.g2.com/products/clickup/reviews/clickup-review-4257118>. Accessed: 2022-12-26.
- [108] Katie Azevedo, M.Ed. Master list vs. to-do list, and why you need both. 2022. URL: <https://schoolhabits.com/master-list-vs-to-do-list-and-why-you-need-both/>. Accessed: 2023-01-01.
- [109] Law Insider. Master list definition. 2022. URL: <https://www.lawinsider.com/dictionary/master-list>. Accessed: 2023-01-01.
- [110] Leo Tumwattana, Adam Lindemann. Sorted³. 2015. URL: <https://www.sortedapp.com>. Accessed: 2022-12-25.
- [111] Microsoft. Microsoft to do. 2017. URL: <https://todo.microsoft.com/>. Accessed: 2022-12-25.
- [112] Microsoft. Typescript. 2023. URL: <https://www.typescriptlang.org/>. Accessed: 2023-03-26.
- [113] My Time Management. How to achieve more with your master list. 2022. URL: <https://www.mytimemanagement.com/master-list.html>. Accessed: 2023-01-01.
- [114] Patia M. Patia m.'s review of google calendar. 2022. URL: <https://www.capterra.com/p/213340/Google-Calendar/reviews/3412359/>. Accessed: 2022-12-24.

- [115] ProductPlan. Ice scoring model. 2022. URL: <https://www.productplan.com/glossary/ice-scoring-model/>. Accessed: 2023-01-06.
- [116] ProductPlan. Rice scoring model. 2022. URL: <https://www.productplan.com/glossary/rice-scoring-model/>. Accessed: 2023-01-06.
- [117] Rafael P. Rafael p.'s review of google calendar. 2021. URL: <https://www.capterra.com/p/213340/Google-Calendar/reviews/2954867/>. Accessed: 2022-12-24.
- [118] React Navigation. React navigation 6. 2023. URL: <https://reactnavigation.org/>. Accessed: 2023-03-26.
- [119] ReduxJS. Redux toolkit. 2023. URL: <https://redux-toolkit.js.org/>. Accessed: 2023-03-26.
- [120] Scott Chacon, Ben Straub. 8.3 customizing git - git hooks. 2022. URL: <https://git-scm.com/book/en/v2/Customizing-Git-Git-Hooks>. Accessed: 2023-01-08.
- [121] Semrush. 50 google search statistics & facts. 2021. URL: <https://www.semrush.com/blog/google-search-statistics/>. Accessed: 2022-12-20.
- [122] Simone G., Rigger, Animator. Nice environment but frustrating to use. 2022. URL: <https://www.g2.com/products/clickup/reviews/clickup-review-6756303>. Accessed: 2022-12-26.
- [123] Slack Technologies. Google calendar for slack. 2022. URL: <https://slack.com/intl/en-in/help/articles/206329808-Google-Calendar-for-Slack>. Accessed: 2022-12-24.
- [124] Sorted³. Can i add calendars like google calendar, outlook calendar, etc.? 2021. URL: <https://support.sortedapp.com/articles/78805-can-i-add-calendars-like-google-calendar-outlook-calendar-etc>. Accessed: 2022-12-23.
- [125] Sorted³. Scheduling doesn't have to be stressful. 2022. URL: <https://www.sortedapp.com/hyper-scheduling>. Accessed: 2022-12-23.
- [126] TickTick. Subscribe ticktick tasks in your calendar app. 2013. URL: <https://blog.tictick.com/2013/11/12/subscribe-ticktick-tasks-in-your-calendar-app/>. Accessed: 2022-12-23.
- [127] Todoist. Community, todoist. 2022. URL: <https://todoist.com/achievements>. Accessed: 2022-12-25.
- [128] Todoist. From overwhelmed to on top of it. 2022. URL: <https://todoist.com/features>. Accessed: 2022-12-26.
- [129] Todoist. Use todoist with your calendar. 2022. URL: <https://todoist.com/help/articles/use-todoist-with-your-calendar>. Accessed: 2022-12-23.
- [130] Trello. Connect trello to everything. 2022. URL: <https://trello.com/integrations>. Accessed: 2022-12-26.
- [131] Trello. How to sync a trello and google calendar integration for better meetings. 2022. URL: <https://blog.trello.com/enterprise/trello-google-calendar-integration>. Accessed: 2022-12-23.
- [132] Trello. Power-ups for trello. 2022. URL: <https://trello.com/power-ups>. Accessed: 2022-12-26.
- [133] trello. Give work a new look with trello views. 2022. URL: <https://trello.com/views>. Accessed: 2022-12-26.
- [134] Tyler Renelle, Siena Leslie, Vicky Hsu. Habitica. 2013. URL: <https://habitica.com/static/home>. Accessed: 2022-12-28.
- [135] University of California, Santa Cruz. Master list template. 2022. URL: <https://careers.ucsc.edu/grad/resources/blank-master-list-pdf.pdf>. Accessed: 2023-01-01.
- [136] Verified User in Biotechnology. Good starting point for project/task management, very inexpensive. 2022. URL: <https://www.g2.com/products/clickup/reviews/clickup-review-6751281>. Accessed: 2022-12-26.
- [137] Verified User in Computer Software, Enterprise (> 1000 emp.) Complicated (but ok, powerful) and confusing mobile app. 2022. URL: <https://www.g2.com/products/clickup/reviews/clickup-review-5169254>. Accessed: 2022-12-26.

- [138] Wikipedia. Astrid. 2013. URL: [https://en.wikipedia.org/wiki/Astrid_\(application\)](https://en.wikipedia.org/wiki/Astrid_(application)). Accessed: 2022-12-24.
- [139] Wikipedia. Goldilocks principle. 2023. URL: https://en.wikipedia.org/wiki/Goldilocks_principle. Accessed: 2023-03-26.
- [140] Wikipedia. Google calendar. 2022. URL: https://en.wikipedia.org/wiki/Google_Calendar. Accessed: 2022-12-31.
- [141] Wikipedia. Iqtell. 2017. URL: <https://en.wikipedia.org/wiki/IQTELL>. Accessed: 2022-12-24.
- [142] Wikipedia. Kanban. 2022. URL: [https://en.wikipedia.org/wiki/Kanban_\(development\)](https://en.wikipedia.org/wiki/Kanban_(development)). Accessed: 2022-12-26.
- [143] Wikipedia. Springpad. 2014. URL: <https://en.wikipedia.org/wiki/Springpad>. Accessed: 2022-12-24.
- [144] Wikipedia. Sunrise calendar. 2016. URL: https://en.wikipedia.org/wiki/Sunrise_Calendar. Accessed: 2022-12-24.
- [145] Wikipedia. Test-driven development. 2023. URL: https://en.wikipedia.org/wiki/Test-driven_development. Accessed: 2023-03-26.
- [146] Wikipedia. Wunderlist. 2020. URL: <https://en.wikipedia.org/wiki/Wunderlist>. Accessed: 2022-12-24.
- [147] Wikipedia. Zeigarnik effect. 2022. URL: https://en.wikipedia.org/wiki/Zeigarnik_effect. Accessed: 2023-01-01.
- [148] Yasar Sri W. Yasar sri w.'s review of clickup. 2022. URL: <https://www.capterra.com/p/158833/ClickUp/reviews/3710136/>. Accessed: 2022-12-24.
- [149] Yoana B. Yoana b.'s review of todoist. 2020. URL: <https://www.capterra.com/p/149339/Todoist-for-Business/reviews/2360405/>. Accessed: 2022-12-24.
- [150] Zapier. How to connect google calendar + google docs. 2022. URL: <https://zapier.com/apps/google-calendar/integrations/google-docs>. Accessed: 2022-12-24.
- [151] Zapier. How to connect google calendar + slack. 2022. URL: <https://zapier.com/apps/google-calendar/integrations/slack>. Accessed: 2022-12-24.
- [152] Zeb Evans, Alex Yurkowski. Clickup. 2017. URL: <https://clickup.com/>. Accessed: 2022-12-22.
- [153] Zhimin Chen. Ticktick. 2010. URL: https://ticktick.com/?language=en_US. Accessed: 2022-12-25.
- [154] Zoom Video Communications. Using the zoom for google workspace add-on. 2022. URL: <https://support.zoom.us/hc/en-us/articles/360020187492-Using-the-Zoom-for-Google-Workspace-add-on>. Accessed: 2022-12-24.

8 Further Reading

Articles

- [155] S. Abd-El-Fatta. The effect of achievement goals profiles on learning approaches and academic achievement: a multiple-goals perspective. *International Journal of School and Cognitive Psychology*, 5(3):214–229, 2018.
- [156] A. Achtziger, U. Bayer, and P. Gollwitzer. Committing to implementation intentions: attention and memory effects for selected situational cues. *Motivation and Emotion*, 36:1–14, Sept. 2012. DOI: 10.1007/s11031-011-9261-6.
- [157] M. M. Al Qmase and M. R. J. Qureshi. Evaluation of the cost estimation models: case study of task manager application. *International Journal of Modern Education and Computer Science*, 5(8):1, 2013.
- [158] S. Bagdadli and M. Gianecchini. Organizational career management practices and objective career success: a systematic review and framework. *Human Resource Management Review*, 29, Aug. 2018. DOI: 10.1016/j.hrmr.2018.08.001.
- [159] J. Bargh. The four horsemen of automaticity: awareness, Dec. 1994.
- [160] T. Behrens, M. Woolrich, M. Walton, and M. Rushworth. Learning the value of information in an uncertain world. *nature neuroscience*, 10(9), 1214-1221. *Nature neuroscience*, 10:1214–21, Oct. 2007. DOI: 10.1038/nn1954.
- [161] F. Birren. Color psychology and color therapy: a factual study of the influence of color on human life. *Academic Medicine*, 25:303, July 1950.
- [162] S. Braem. Conditioning task switching behavior. *Cognition*, 166:272–276, 2017.
- [163] J.-A. Chase, R. Topp, C. Smith, M. Cohen, N. Fahrenwald, J. Zerwic, L. Benefield, C. Anderson, and V. Conn. Time management strategies for research productivity. *Western journal of nursing research*, 35, Aug. 2012. DOI: 10.1177/0193945912451163.
- [164] A. J. Crum, J. P. Jamieson, and M. Akinola. Optimizing stress: an integrated intervention for regulating stress responses. *Emotion*, 20(1):120, 2020.
- [165] J. de la Fuente, P. Paoloni, D. Kauffman, M. Yilmaz Soylu, P. Sander, and L. Zapata. Big five, self-regulation, and coping strategies as predictors of achievement emotions in undergraduate students. *International Journal of Environmental Research and Public Health*, 17(10):3602, 2020.
- [166] D. Drewery, R. Sproule, and T. Pretti. Lifelong learning mindset and career success: evidence from the field of accounting and finance. *Higher Education, Skills and Work-Based Learning*, 10:567–580, Feb. 2020. DOI: 10.1108/HESWBL-03-2019-0041.
- [167] B. George, R. M. Walker, and J. Monster. Does strategic planning improve organizational performance? a meta-analysis. *Public Administration Review*, 79(6):810–819, 2019.
- [168] M. Gillebaart and D. T. de Ridder. Effortless self-control: a novel perspective on response conflict strategies in trait self-control. *Social and Personality Psychology Compass*, 9(2):88–99, 2015.
- [169] D. Gutt, T. von Rechenberg, and D. Kundisch. Goal achievement, subsequent user effort and the moderating role of goal difficulty. *Journal of Business Research*, 106:277–287, 2020.
- [170] L. Heracleous. Strategic thinking or strategic planning? *Long range planning*, 31(3):481–487, 1998.
- [171] Y. Huang and J. Hutchinson. The roles of planning, learning, and mental models in repeated dynamic decision making. *Organizational Behavior and Human Decision Processes*, 122:163–176, Nov. 2013. DOI: 10.1016/j.obhdp.2013.07.001.
- [172] M. Inzlicht, K. M. Werner, J. L. Briskin, and B. W. Roberts. Integrating models of self-regulation. *Annual review of psychology*, 72:319–345, 2021.

- [173] M. Janczyk, M. Dambacher, M. Bieleke, and P. Gollwitzer. The benefit of no choice: goal-directed plans enhance perceptual processing. *Psychological research*, 79, Mar. 2014. doi: 10.1007/s00426-014-0549-5.
- [174] I. Karpovich. Self-regulation as a basic element of the professional culture of engineers. *Education Sciences*, 9, July 2019. doi: 10.3390/educsci9030200.
- [175] S. Kejriwal, V. Vishal, A. Gulati, and G. Gambhir. A review of daily productivity growth using todo manager.
- [176] J. W. Lee, S. R. Daly, A. Huang-Saad, G. Rodriguez, and C. M. Seifert. Cognitive strategies in solution mapping: how engineering designers identify problems for technological solutions. *Design studies*, 71:100967, 2020.
- [177] T. Martiny-Huenger, S. E. Martiny, E. J. Parks-Stamm, E. Pfeiffer, and P. M. Gollwitzer. From conscious thought to automatic action: a simulation account of action planning. *Journal of Experimental Psychology: General*, 146(10):1513, 2017.
- [178] J. Pencavel. The productivity of working hours. *The Economic Journal*, 125, June 2014. doi: 10.1111/eco.12166.
- [179] A. Scholer, D. B. Miele, K. Murayama, and K. Fujita. New directions in self-regulation: the role of metamotivational beliefs. *Current Directions in Psychological Science*, 27:096372141879054, Nov. 2018. doi: 10.1177/0963721418790549.
- [180] J. Schwabach, J. Bartley, and K. Polk. Sorting it out: a framework for increasing mental flexibility and valued action in athletes using the act matrix. *Journal of Sport Psychology in Action*, 10:1–6, Dec. 2018. doi: 10.1080/21520704.2018.1549638.
- [181] K. Spiliotopoulos, M. Rigou, and S. Sirmakessis. A comparative study of skeuomorphic and flat design from a ux perspective. *Multimodal Technologies and Interaction*, 2:31, June 2018. doi: 10.3390/mti2020031.
- [182] K. Thorsteinsen and J. Vittersø. Striving for wellbeing: the different roles of hedonia and eudaimonia in goal pursuit and goal achievement. *International Journal of Wellbeing*, 8(2), 2018.
- [183] C. Vogelsang, C. Kulgemeyer, and J. Riese. Learning to plan by learning to reflect?—exploring relations between professional knowledge, reflection skills, and planning skills of preservice physics teachers in a one-semester field experience. *Education Sciences*, 12(7):479, 2022.
- [184] L. C. Walsh, J. K. Boehm, and S. Lyubomirsky. Does happiness promote career success? revisiting the evidence. *Journal of Career Assessment*, 26(2):199–219, 2018.
- [185] F. Wieber, L. A. Sezer, and P. M. Gollwitzer. Asking “why” helps action control by goals but not plans. *Motivation and Emotion*, 38(1):65–78, 2014.

Articles from collections

- [186] P. G. Zimbardo and J. N. Boyd. Putting time in perspective: a valid, reliable individual-differences metric. In *Time perspective theory; review, research and application*, pages 17–55. Springer, 2015.

Books

- [187] F. Birren. *Color Psychology And Color Therapy; A Factual Study Of The Influence of Color On Human Life*. Hauraki Publishing, 2016. ISBN: 9781786258663. URL: <https://books.google.com.mx/books?id=B6D2CwAAQBAJ>.
- [188] L. Buley. *The User Experience Team of One: A Research and Design Survival Guide*. Rosenfield Media, 2019. ISBN: 1933820187.
- [189] E. Canziba. *Hands-On UX Design for Developers*. Packt, 2018. ISBN: 9781788626699.

- [190] N. Eyal. *Hooked: How to Build Habit-Forming Products*. Portfolio, Illustrated edition, 2014. ISBN: 1591847788.
- [191] H. G. Halvorson. *Succeed — How We Can Reach Our Goals*. Hudson Street Press, 2010. ISBN: 1594630739.
- [192] J. Johnson. *Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Guidelines*. Morgan Kaufmann, Second edition, 2014. ISBN: 9780124079144.
- [193] D. Kadaav. *Design for Hackers*. John Wiley & Sons, First edition, 2011. ISBN: 1119998956.
- [194] L. Klein. *Build Better Products: A Modern Approach to Building Successful User-Centered Products*. Rosenfeld Media, First edition, 2016. ISBN: 1933820586.
- [195] A. Kleon. *Steal Like an Artist: 10 Things Nobody Told You about Being Creative*. Workman Publishing, 2012. ISBN: 9780761169253.
- [196] S. Krug. *Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability*. New Riders, Third edition, 2013. ISBN: 9780321965516.
- [197] L. Maioli. *Fixing Bad UX Designs: Master Proven Approaches, Tools, and Techniques to Make Your User Experience Great Again*. Packt Publishing, 2018. ISBN: 1787120554.
- [198] D. Rock. *Your Brain at Work: Strategies for Overcoming Distraction, Regaining Focus, and Working Smarter All Day Long*. Harper Business, 2009. ISBN: 0061771295.
- [199] R. Thaler and C. Sunstein. *NUUDGE: Improving Decisions About Health, Wealth, and Happiness*, volume 47. June 2009. ISBN: 9780143115267.

Conference proceedings

- [200] S. Jolaoso and M. Perez-Quinones. Taskambient: a study in personal task management visualization. In pages 1797–1802, Apr. 2015. DOI: 10.1525/collabra.136.
- [201] Y.-H. Kim, E. K. Choe, B. Lee, and J. Seo. Understanding personal productivity: how knowledge workers define, evaluate, and reflect on their productivity. In May 2019. DOI: 10.1145/3290605.3300845.
- [202] T. Kreifelts, E. Hinrichs, and G. Woetzel. Sharing to-do lists with a distributed task manager. In *Proceedings of the Third European Conference on Computer-Supported Cooperative Work 13–17 September 1993, Milan, Italy ECSCW'93*, pages 31–46. Springer, 1993.

Theses

- [203] H. Eisfeld and F. Kristalovich. *The rise of dark mode: A qualitative study of an emerging user interface design trend*, 2020.
- [204] A. Kühn. *The Different Effects of Temporal Orientation on Optimism and Life Satisfaction in the Relationship between a Best Possible Self Intervention and Well-Being*, University of Twente, 2021.
- [205] T. Y. Lin. *A study of how emotional design of a digital task management tool impacts individual productivity*, Massachusetts Institute of Technology, 2018.
- [206] N. Stauner. *Personal Goal Attainment, Psychological Well-Being Change, and Meaning in Life*, Mar. 2013.

Online

- [207] Goldsmiths. College. 1891. URL: <https://www.gold.ac.uk>. Accessed: 2022-12-22.
- [208] Joe McAlister. Cm3050 mobile development. 2021. URL: <https://www.coursera.org/learn/uol-cm3050-mobile-development/>. Accessed: 2022-12-22.
- [209] Ludwig, Srivastava, Berkman. Planfulness scale. 2020. URL: <https://mysocialbrain.org/plan-info.html>. Accessed: 2022-12-30.

- [210] Marc Zao-Sanders, Harvard Business Review. How timeboxing works and why it will make you more productive. 2018. URL: <https://hbr.org/2018/12/how-timeboxing-works-and-why-it-will-make-you-more-productive>. Accessed: 2022-12-25.
- [211] Sébastien Lavoie. Productivido. 2022. URL: <http://productivido.tech>. Accessed: 2022-12-22.
- [212] University of London. University. 1836. URL: <https://www.london.ac.uk>. Accessed: 2022-12-22.

9 List of Figures

1	Gantt chart: Adopting a strict, weekly cadence to deliver the product on time.	35
2	Todoist's dark theme shows a lack of contrast on some elements such as the "hover" color on some links, highlighted by a red arrow.	36
3	Todoist's recurring task feature not displaying the recurring schedule by default, highlighted by a red arrow.	36
4	Todoist's attachments are found under the "Comments" section of the search results, which seems unintuitive.	36
5	Todoist's "Upcoming" view is used to visualize all future tasks at once without having the ability to use filters.	37
6	ClickUp boasts about a large number of features, but it is up to the user to become knowledgeable enough to take advantage of them. The desktop version is shown here because the mobile application, as of this writing, is still a largely incomplete product that does not permit users to reach most of the features found in the desktop counterpart.	38
7	Trello is dead simple to use, but it is also very limited in terms of features.	39
8	Airtable is a powerful tool when used as a desktop application, but it is difficult to navigate on mobile devices as fields are constantly cut off the screen regardless of the device orientation.	40
9	Asana's simple calendar works well, but it does not offer any custom views, such as weekly or monthly.	41
13	Showing the database design behind ProductiviDo.	59
14	Showing the design of the app's interface in Figma.	60

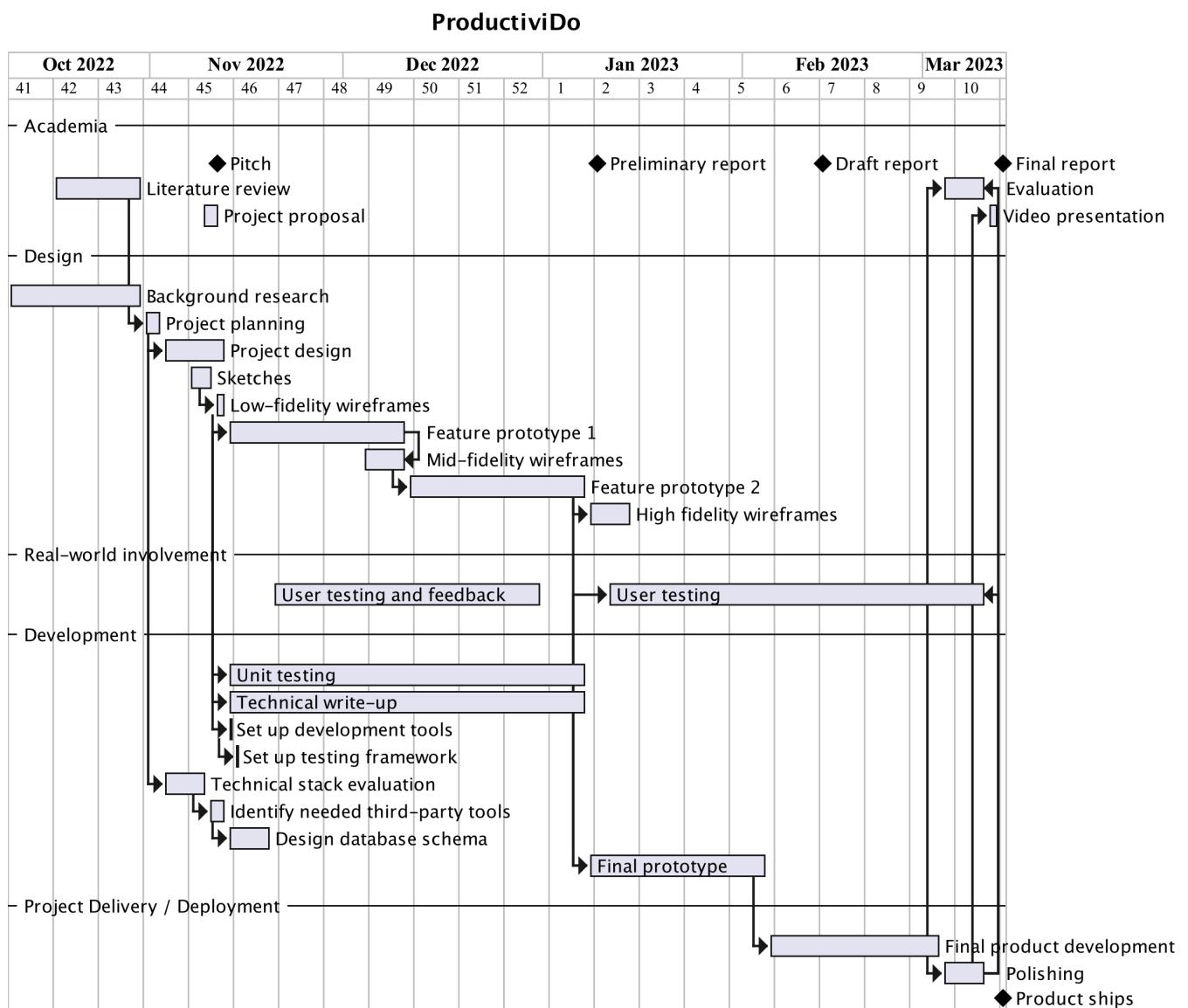


Figure 1: Gantt chart: Adopting a strict, weekly cadence to deliver the product on time.

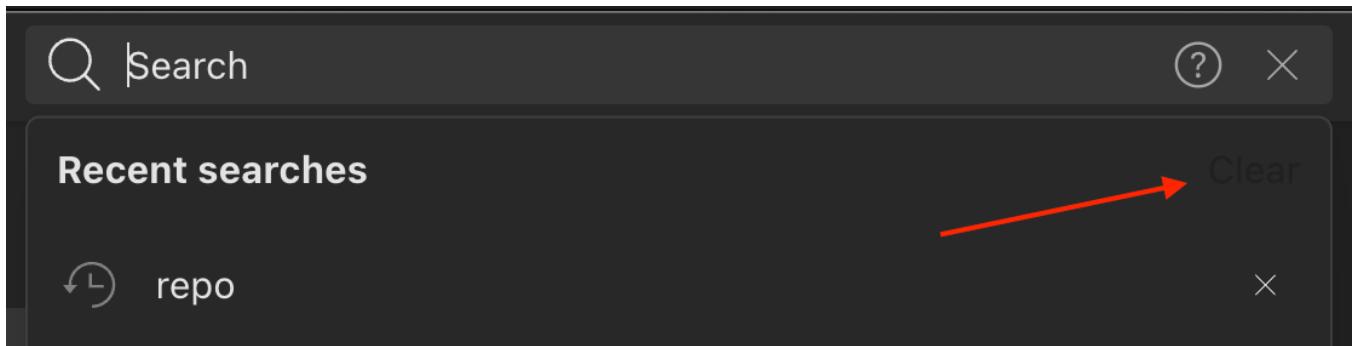


Figure 2: Todoist's dark theme shows a lack of contrast on some elements such as the "hover" color on some links, highlighted by a red arrow.

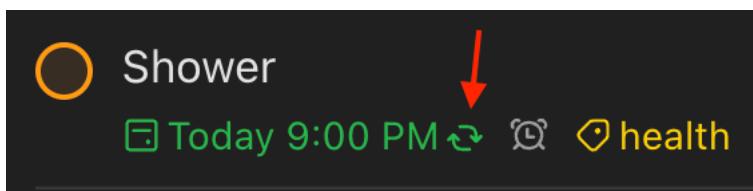


Figure 3: Todoist's recurring task feature not displaying the recurring schedule by default, highlighted by a red arrow.

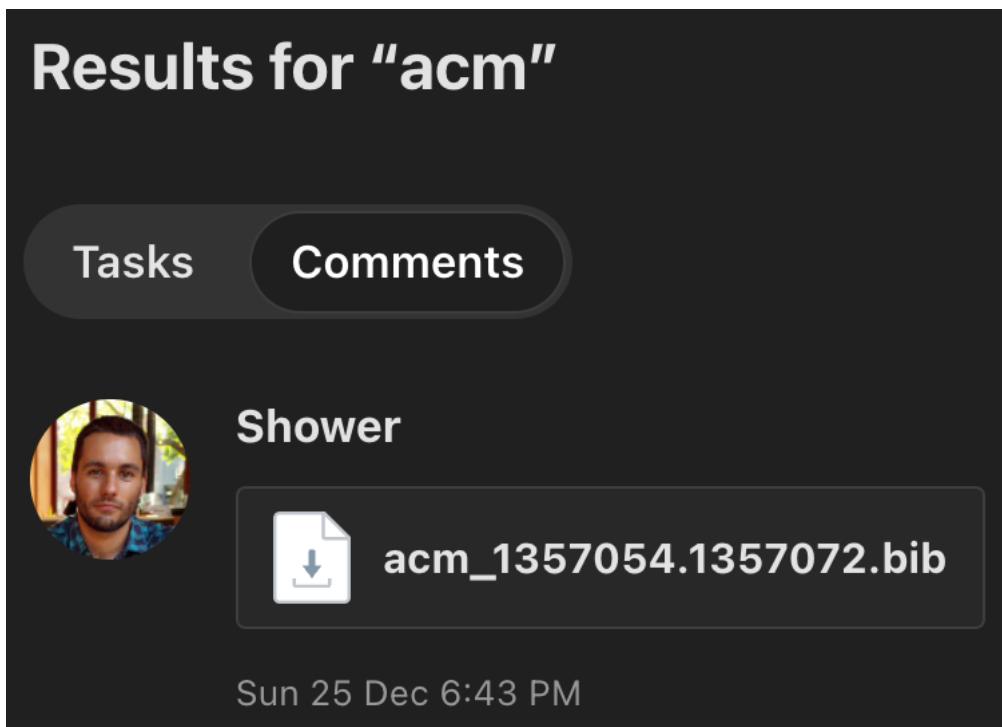


Figure 4: Todoist's attachments are found under the "Comments" section of the search results, which seems unintuitive.

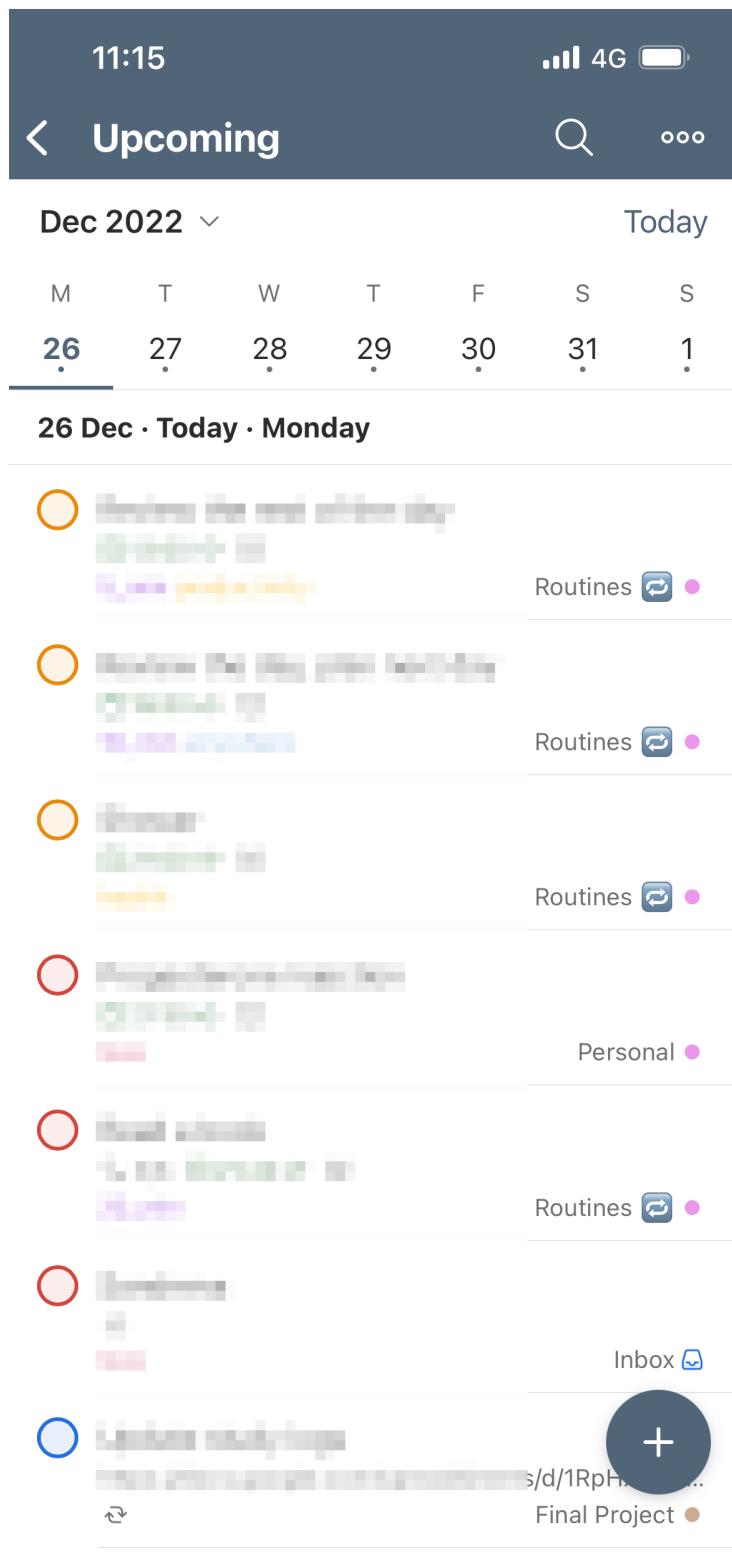


Figure 5: Todoist's "Upcoming" view is used to visualize all future tasks at once without having the ability to use filters.

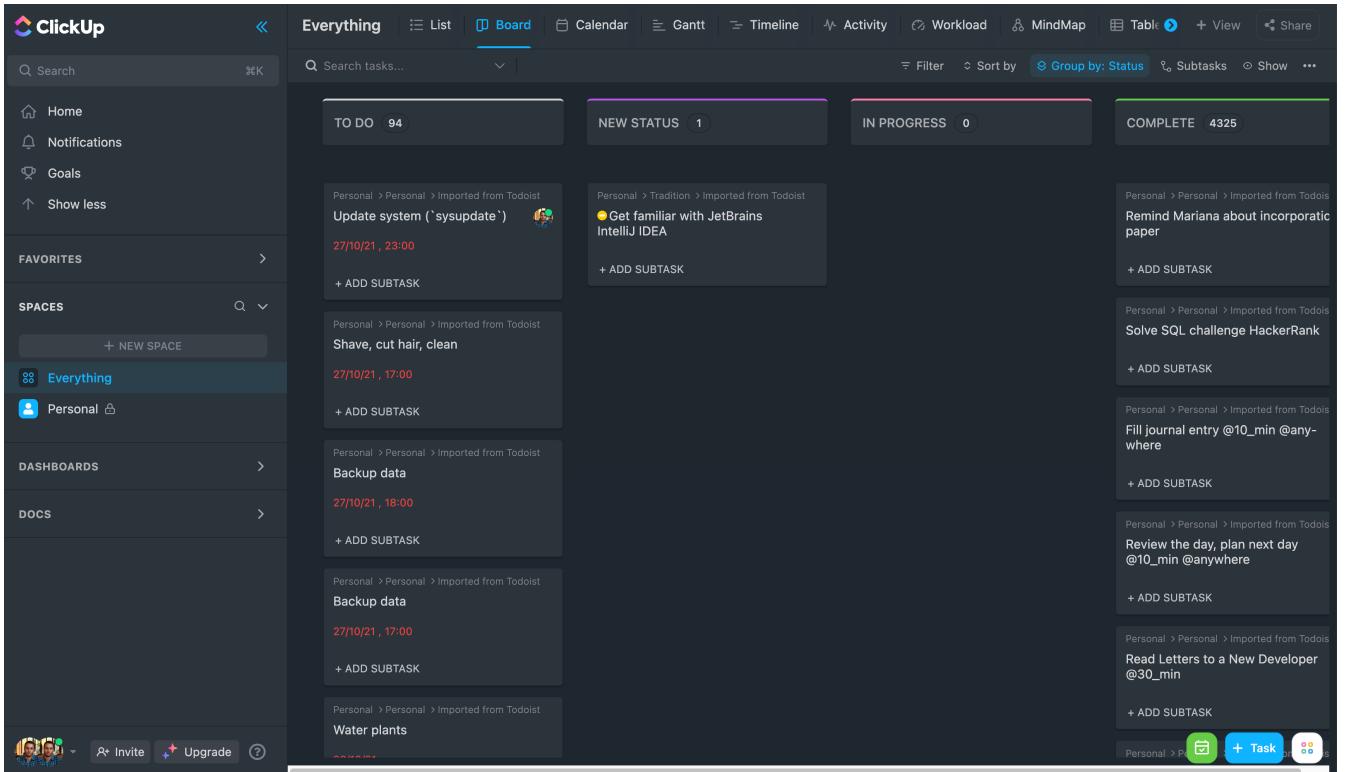


Figure 6: ClickUp boasts about a large number of features, but it is up to the user to become knowledgeable enough to take advantage of them. The desktop version is shown here because the mobile application, as of this writing, is still a largely incomplete product that does not permit users to reach most of the features found in the desktop counterpart.

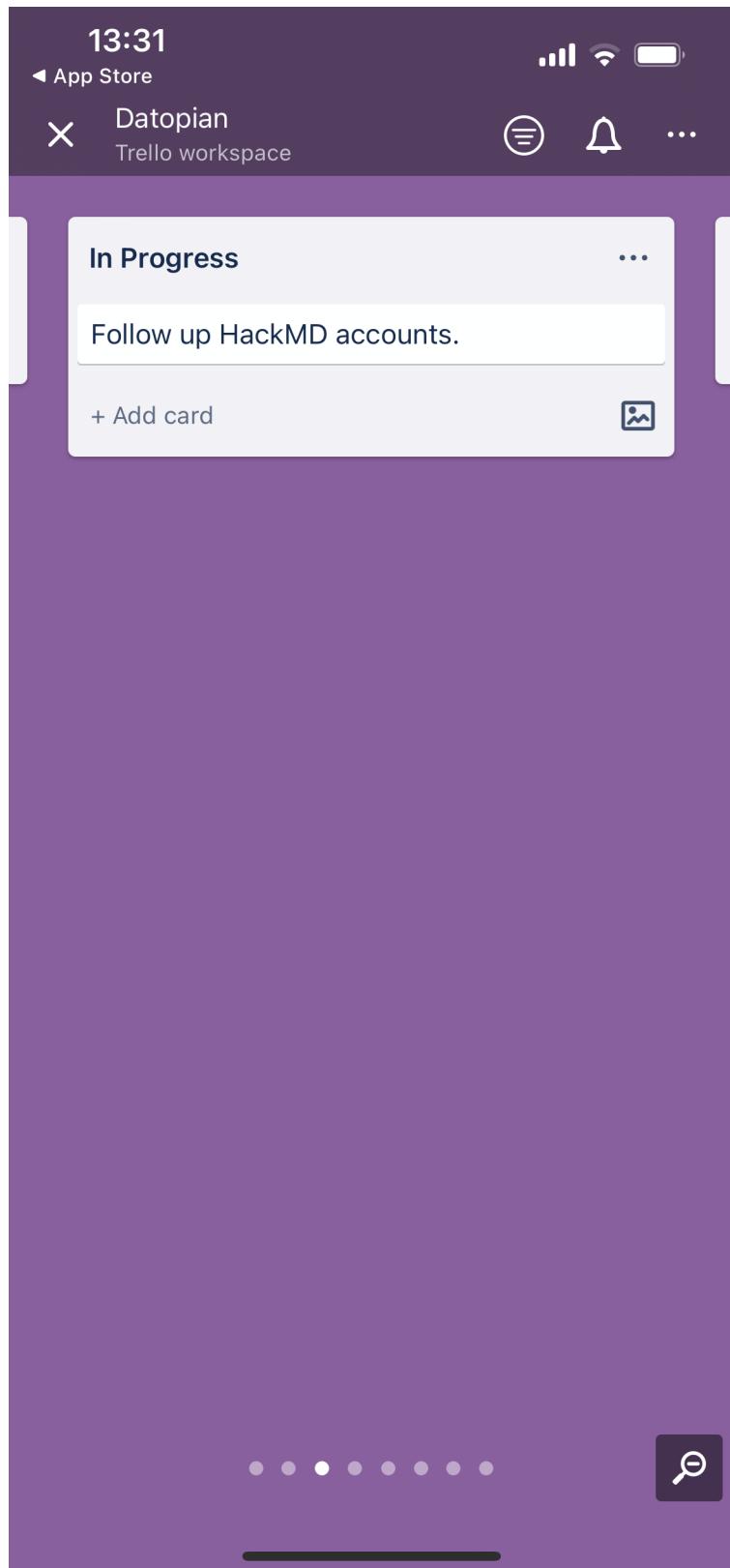


Figure 7: Trello is dead simple to use, but it is also very limited in terms of features.

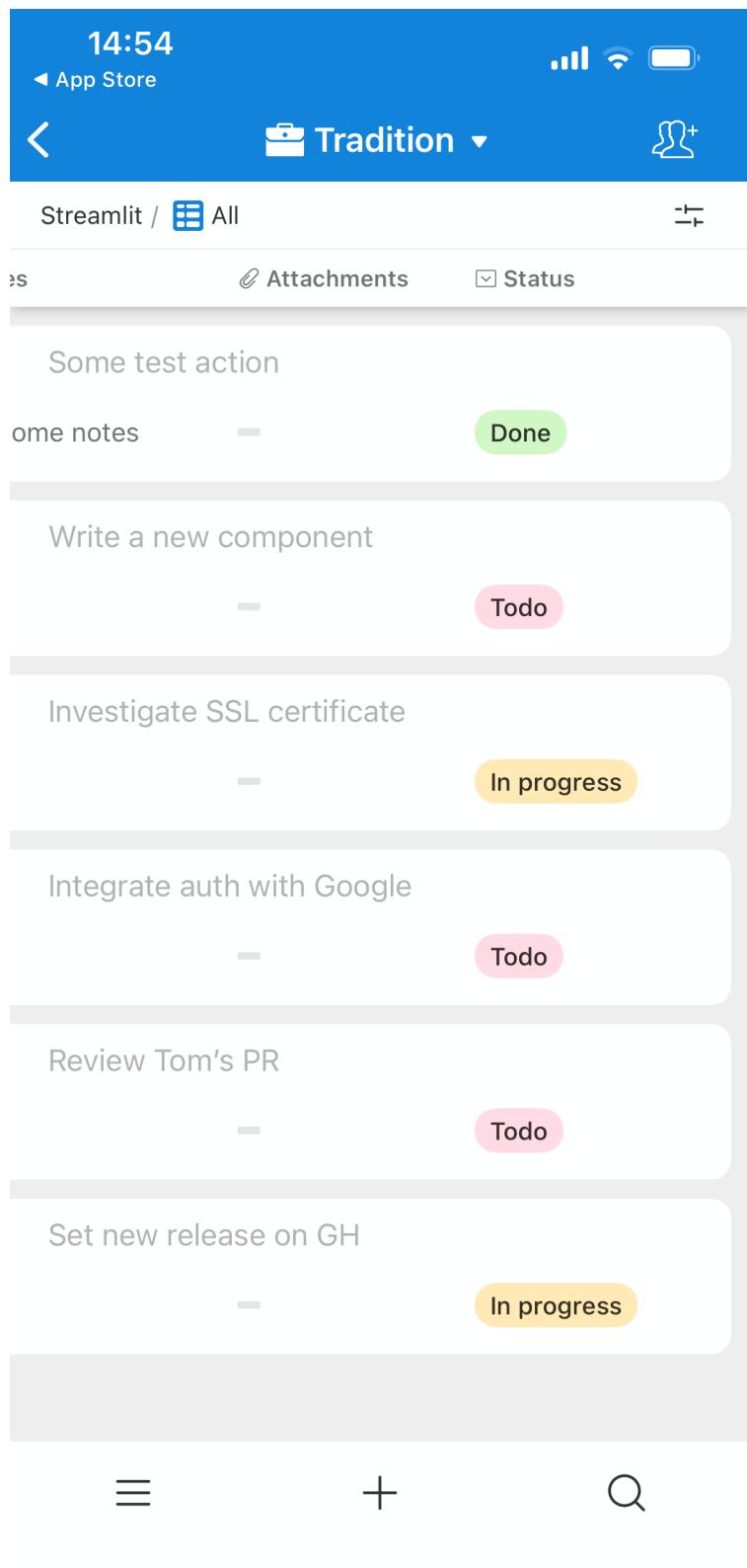


Figure 8: Airtable is a powerful tool when used as a desktop application, but it is difficult to navigate on mobile devices as fields are constantly cut off the screen regardless of the device orientation.

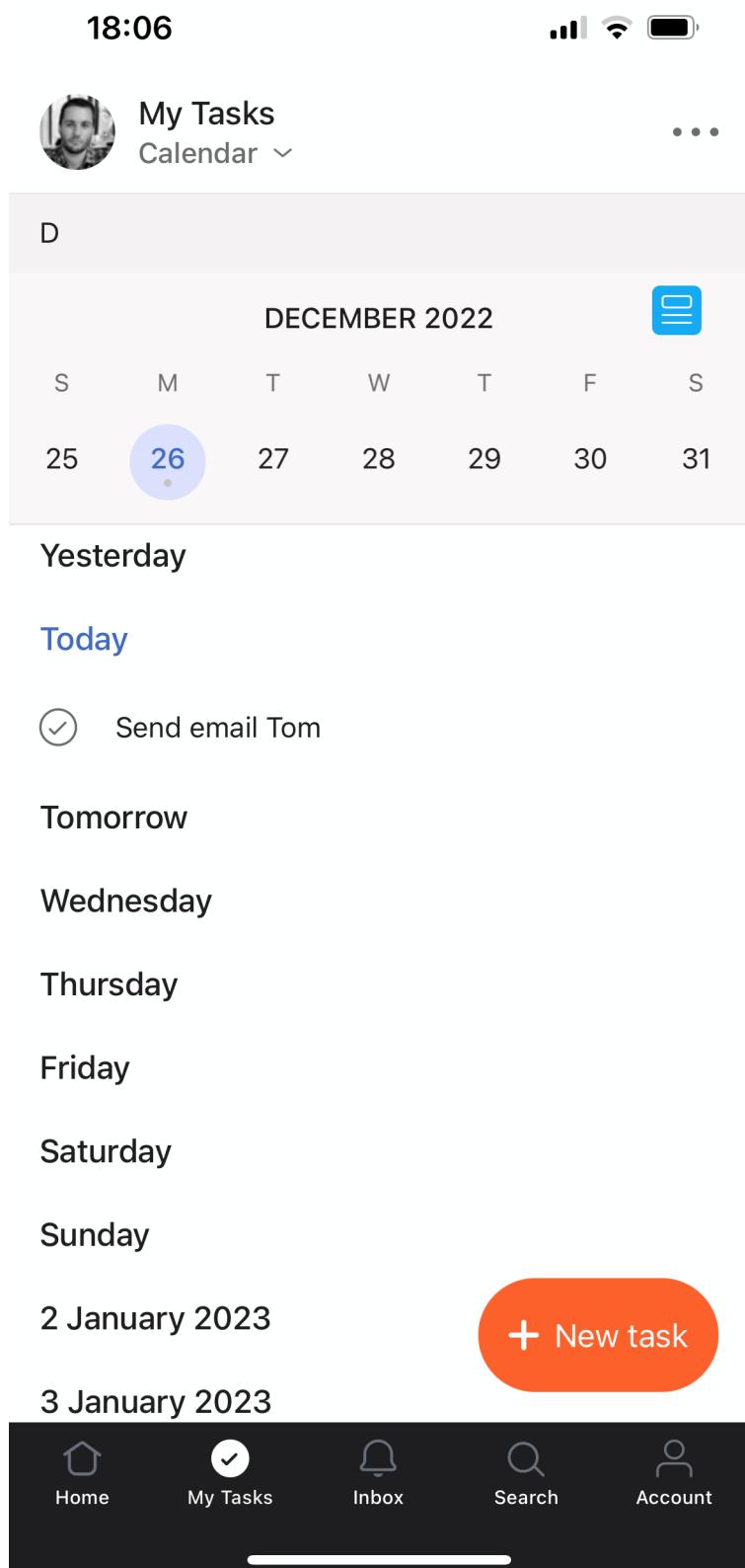


Figure 9: Asana's simple calendar works well, but it does not offer any custom views, such as weekly or monthly.

10 List of Tables

1	Highlighting the good side of task managers.	43
2	Emphasizing a lack of digital productivity with current solutions.	43

	 Todoist	 ClickUp	 Trello	 Airtable	 Asana
 Design	Clean Minimal Accessible	Customizable Modern Standard	Easy to understand Compact Familiar	Colorful Help-oriented Complete	Readable Business style Intuitive
 Functionality	Filters / Labels Reminders Fast task input	Time tracker Reminders Scratchpads	Drag & drop "Power-Ups" Quick checklist	Integrated tutorials Numerous templates SQL-like storage	Calendar view Custom swipe actions Task history
 Navigation	Straightforward Simple calendar Multiple views	Nested menus Quick add button Task breadcrumbs	Card-based Free-flowing Compartmentalized	Spreadsheet-like Custom views Tabbed UI	Nested lists Breadcrumbs Calendar support
 Planning	Concept of inbox Upcoming view Today view	Categorical Task dependency Agile-like	Cards with due date Cards drag & drop Notifications	Column status Gantt view Freestyle	Project-based Complete vs. incomplete Assigned tasks
 Organization	Hierarchical Fine-grained Efficient	Lists / Sub-lists Boards Spaces	Kanban-focused Workspaces / Boards Item ordering	Grid-focused Database mindset "Base" / Workspaces	Kanban support Team-oriented "Weekly focus"

Table 1: Highlighting the good side of task managers.

	 Todoist	 ClickUp	 Trello	 Airtable	 Asana
 Design	Low contrast Limited themes How is it recurring?	Subpar (vs. desktop) Cluttered Mixed content	Inflexible Too simplistic Arguably inconsistent	Flawed on mobile Unusable "Gallery" view High complexity	Lots of wasted space Somewhat inaccessible Truncated views
 Functionality	Limited search† Limited task ordering Bad synchronization	Limited search† Subpar (vs. desktop) Useless features	Limited search† Unexpressive filters Minimal in free plan	Confusing search† Hidden content Unusable without work	Very limited search† Limited for individuals Minimal sorts/filters
 Navigation	Long lists of items Unsortable items Too much list-oriented	Impractical calendar Too team-focused Limited speed dial	Disjointed views Lists/Boards only Difficult on large projects	Needed zoom levels Best for large screens Small touch points	Requires multi-taps Needs horizontal scroll Hidden search functions
 Planning	Single "Upcoming" view Difficult prioritization Partial GTD†† support	Limited recurring options Tasks are dispersed Priorities get lost	Only sorted by due date Behind paywall No review process	Complicated "bases" Requires preparation Difficult calendar	Undoable from calendar Dull home page Hard to find by priority
 Organization	Complicated filters Only filters and projects Sub-tasks clutter	Confusing layouts Hidden track time Limiting home view	Cards only Kanban only Missing sorts/filters	Too much content Segmented Appropriate for desktop	Confusing hierarchy Unrelated menu items Limited task moves

Table 2: Emphasizing a lack of digital productivity with current solutions.

Appendices

A Project management tools as task managers

There are countless solutions for task management. Many options do not qualify strictly as task managers, but rather as project management tools as they are built to accommodate teams and collaboration, which is not the focus of this project. Nevertheless, some of these tools find their way into the personal productivity space, and they are worth mentioning here.

A.1 Airtable

Although it focuses on teams and collaboration features, Airtable can do task management at the personal level and accommodates tech-savvy people very well [68]. The whole concept revolves around using tables and even the use of “bases” (workspaces) to navigate different spaces feels like switching between different relational databases, albeit with a clean frontend. This product is mainly built to scale with large teams and requires more work upfront to set it up properly for individuals. Without optimizing tables and columns for one’s needs (which requires a significant amount of side-scrolling on mobile devices), the UI is very space inefficient on smaller screens (cf Figure 8). While some features have inspired ProductiviDo’s design, such as the ability to restore items from trash, there are other concerns that help rationalize a need for a newcomer: the built-in calendar view is limited in terms of functionality (e.g., many more applications communicate and sync with Google Calendar); recurring tasks are complicated to set up with “automations” [69]; and the filters are over-engineered for an audience using the mobile version of the software where one is expected to add “conditions” looking very much like SQL queries [70].

A.2 Asana

Asana, like Trello, focuses primarily on displaying tasks in the “Kanban” style with columns of data [142]. It is a very stable and reactive product, the backend being built in part with Asana’s own LunaDb for fast data loading, its web components being put together with TypeScript and React while its mobile applications are solidly relying on Swift (iOS) and Kotlin (Android) and are accompanied by an Electron-based offering for desktop users [77]. Asana falls into the category of project management systems and is not limited to task management, but it can serve the same purpose for individuals as the previous options [76]. Like the rivals discussed above, its built-in calendar view suffers from the limitations already mentioned and in no way competes with Google Calendar’s capabilities (cf. Figure 9). Asana’s technical prowess is executed at the expense of some important features that are missing from the product, notably a lack of task filters: only “assignees” and “due dates” are accounted for, which is particularly limiting for those working alone. Its collapsible list of items make good use of the screen real estate, but search can only be performed on keywords, which makes finding existing tasks a strenuous activity inside this nested hierarchy of information. Once a task is accessed, though, the options presented are straightforward and efficient (e.g., moving a task from one project to another). As a product focusing specifically on team work, Asana maximizes its potential with collaborative tools including comments, assignments and notifications across teams.

A.3 ClickUp

ClickUp touts itself as “*One app to replace them all*” [152] with “*hundreds of powerful tools*” [82] and as such ends up being a bloated piece of software — a state of affairs contributing to making it the worst app amongst the five task management platforms analyzed in **Reclaim.ai**’s report when looking at the “*percent of teams that accomplish less than 40% of planned tasks each week*” [65]. ClickUp implements an impressive

number of features generally well received by its users when singled out [83], while at the same time being the recipient of many critics vouching for a simpler interface as its complexity is a major drawback for the average user as reported on G2 [74, 107, 122, 136, 137], “*the world’s largest and most trusted software marketplace*”⁵. In the context of ProductiviDo’s objectives, it is clear that ClickUp does not provide a similarly frictionless experience where users won’t feel overwhelmed by the UI (User Interface) and will be able to take actions immediately towards their goals (cf. Figure 6) as it suggests to “*tailor your tasks for any work need with 35+ ClickApps that each add different functionalities*” [81].

⁵The author remains fully aware of the monetary incentives and biases involved in such companies and certifies having manually verified the veracity of all user reviews referenced in this work — all mobile applications listed having some form of free trial available to make this pursuit economically viable.

B Wireframes

Focus

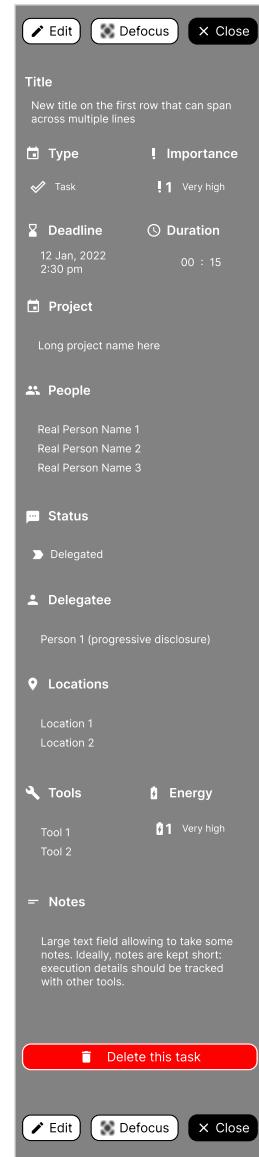
Viewing Focus list



Completing a task



Viewing single task



Inscribe

Prioritizing

Mandatory section

Title
New title on the first row that can span across multiple lines

Type **Importance**

Task **!1** Very high

Status
Delegated

Delegatee
Person 1 (progressive disclosure)

Optional section

Deadline 12 Jan, 2022 2:30 pm **Duration** 00 : 15

Project
Project 2

People
Person 1
Person 2
Person 3

Locations
Location 1
Location 2

Tools **Energy**

Tool 1
Tool 2 **!1** Very high

Notes
Large text field allowing to take some notes. Ideally, notes are kept short: execution details should be tracked with other tools.

Discard changes **Add to priorities**

Prioritizing

Mandatory section

Title
New title on the first row that can span across multiple lines

Type **Importance**

Task **!1** Very high

Status
Delegated

Delegatee
Person 1 (progressive disclosure)

Optional section

Deadline 12 Jan, 2022 2:30 pm **Duration** 00 : 15

Project
Project 2

People
Person 1
Person 2
Person 3

Locations
Location 1
Location 2

Tools **Energy**

Tool 1
Tool 2 **!1** Very high

Notes
Large text field allowing to take some notes. Ideally, notes are kept short: execution details should be tracked with other tools.

Discard changes **Add to priorities**

Prioritize

Prioritize Send to Focus from task list Send to Focus from task view



Send to Focus from task view



Filtering

Main view



V1

Switch



V2

Popovers



Expanded



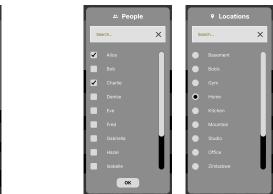
Small modal



Collapsed



Large modals



Searching



Sorting

Main view



V4

Popovers



Review

Active

Delete **Reschedule** **≡**

-  ! 1 **Complete task with long title spanning across two lines**
NEW Jan 12 2023 2:30pm ⌚ 02:30 📅 Long project name here
-  ! 1 **Complete task with long title spanning across two lines**
NEW Jan 12 2023 2:30pm ⌚ 02:30 📅 Long project name here
-  ! 1 **Complete task with long title spanning across two lines**
NEW Jan 12 2023 2:30pm ⌚ 02:30 📅 Long project name here
-  ! 1 **Complete task with long title spanning across two lines**
NEW Jan 12 2023 2:30pm ⌚ 02:30 📅 Long project name here
-  ! 1 **Complete task with long title spanning across two lines**
NEW Jan 12 2023 2:30pm ⌚ 02:30 📅 Long project name here
-  ! 1 **Complete task with long title spanning across two lines**
NEW Jan 12 2023 2:30pm ⌚ 02:30 📅 Long project name here
-  ! 1 **Complete task with long title spanning across two lines**
NEW Jan 12 2023 ⌚ 02:30 📅 Long project

Inactive

Delete **Reschedule** **≡**

-  **! 1** **Complete task with long title spanning across two lines**
NEW
Jan 12 2023 2:30pm ⌚ 02:30 ⌚ Long project name here
-  **! 1** **Complete task with long title spanning across two lines**
NEW
Jan 12 2023 2:30pm ⌚ 02:30 ⌚ Long project name here
-  **! 1** **Complete task with long title spanning across two lines**
NEW
Jan 12 2023 2:30pm ⌚ 02:30 ⌚ Long project name here
-  **! 1** **Complete task with long title spanning across two lines**
NEW
Jan 12 2023 2:30pm ⌚ 02:30 ⌚ Long project name here
-  **! 1** **Complete task with long title spanning across two lines**
NEW
Jan 12 2023 2:30pm ⌚ 02:30 ⌚ Long project name here
-  **! 1** **Complete task with long title spanning across two lines**
NEW
Jan 12 2023 2:30pm ⌚ 02:30 ⌚ Long project name here
-  **! 1** **Complete task with long title spanning across two lines**
NEW
Jan 12 2023 ⌚ 02:30 ⌚ Long project name here

About

i About X

Introduction text Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco...

Sections

- Section 1
- Section 2
- Section 3
- Section 4

Section 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi...

Section 2

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi...

Help

?

Help

X

Sections

- Section 1
- Section 2
- Section 3
- Section 4

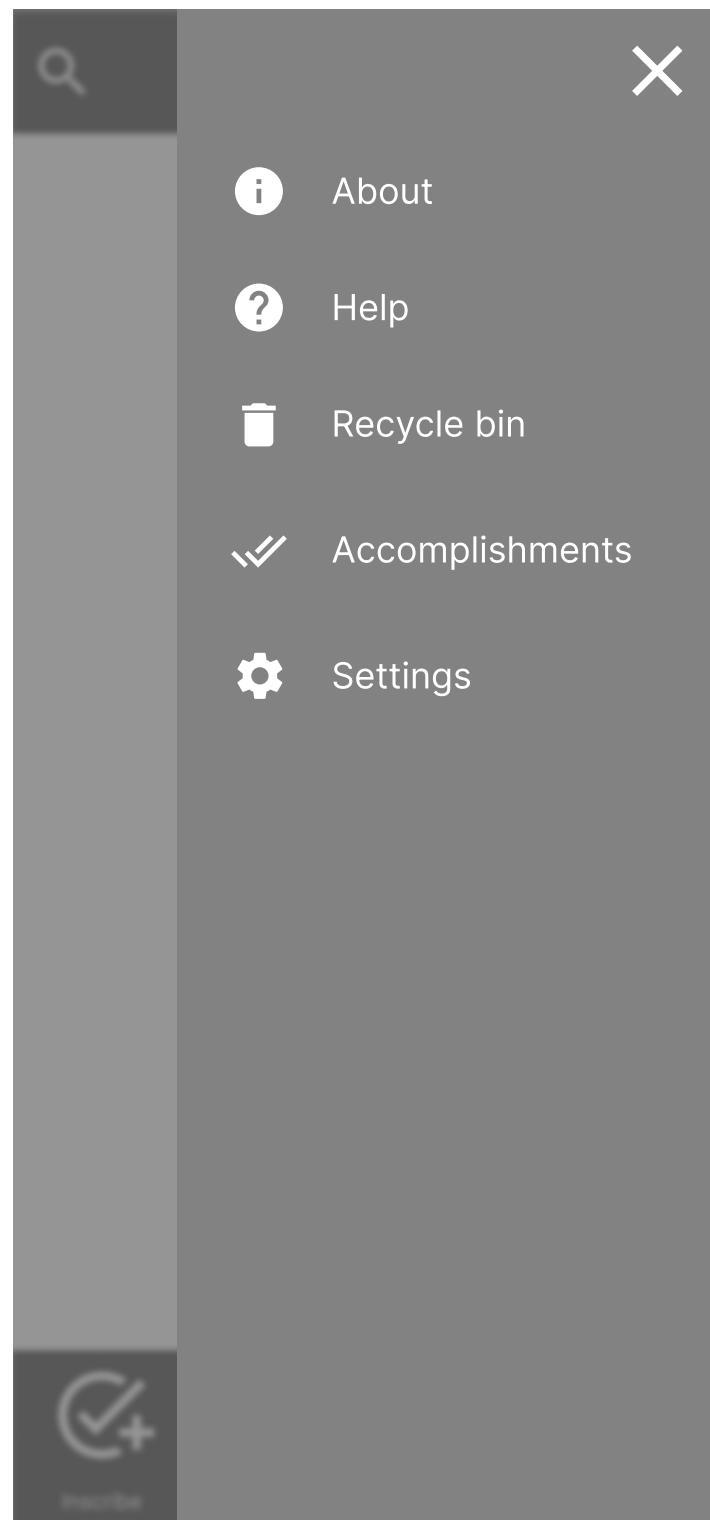
Section 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi...

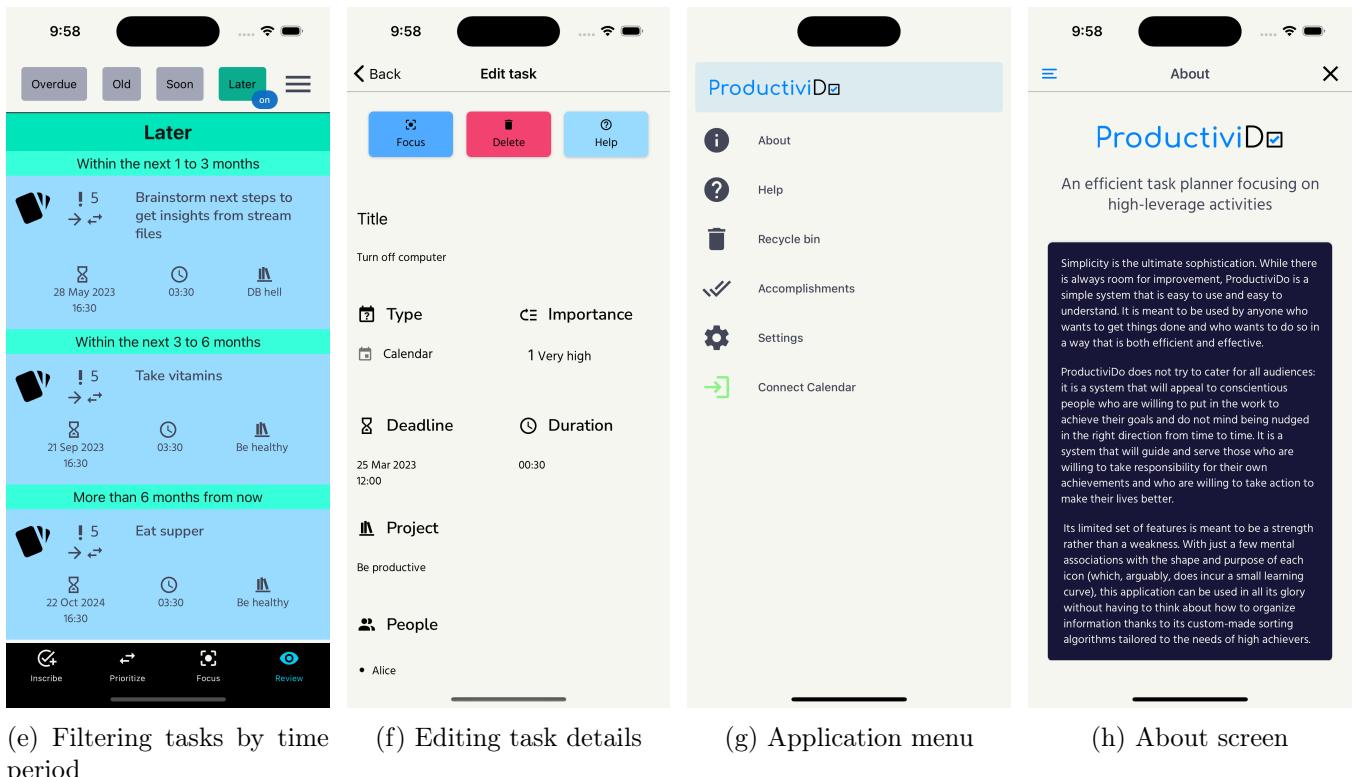
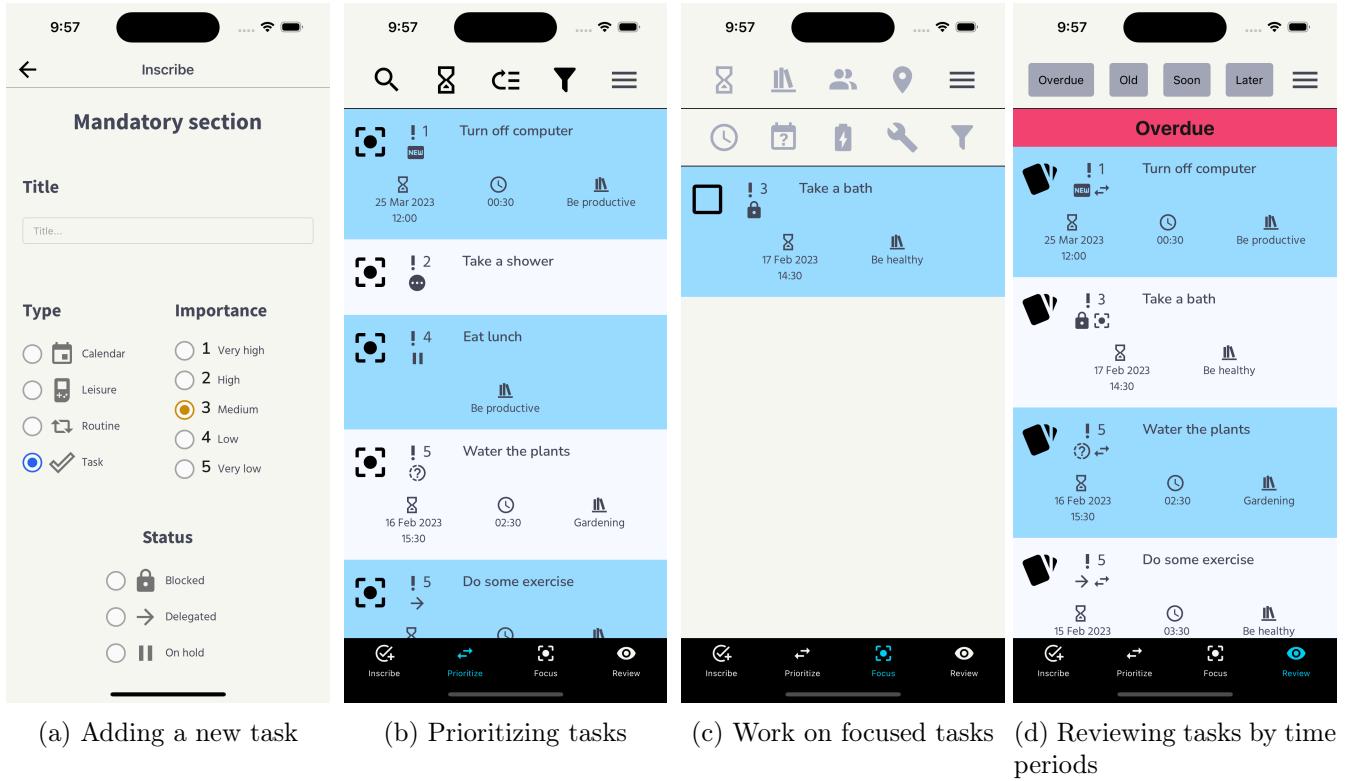
Section 2

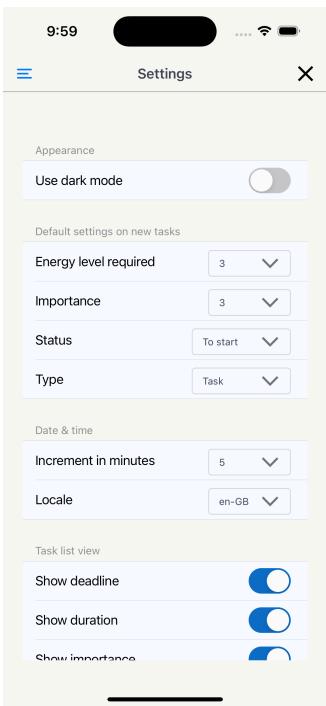
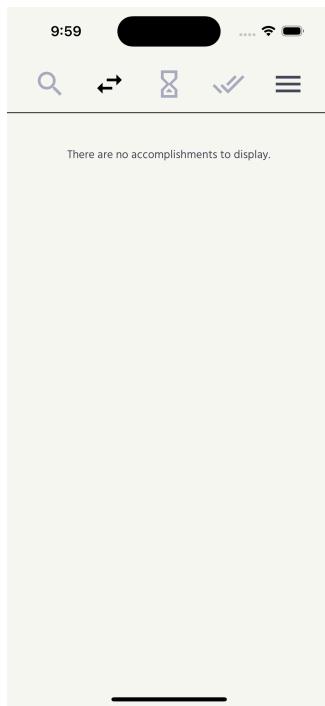
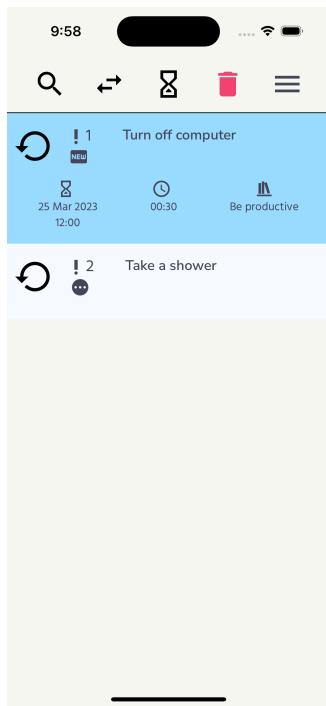
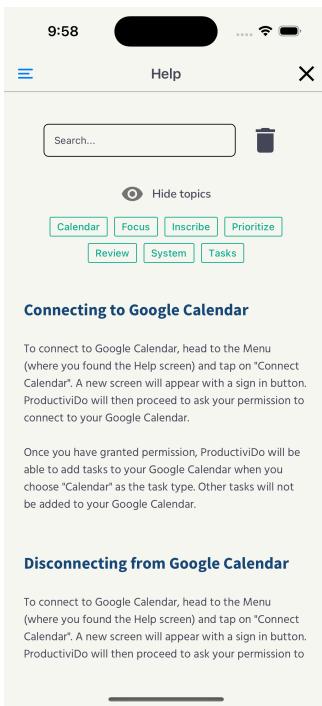
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi...

Menu open



C Screenshots of the final application



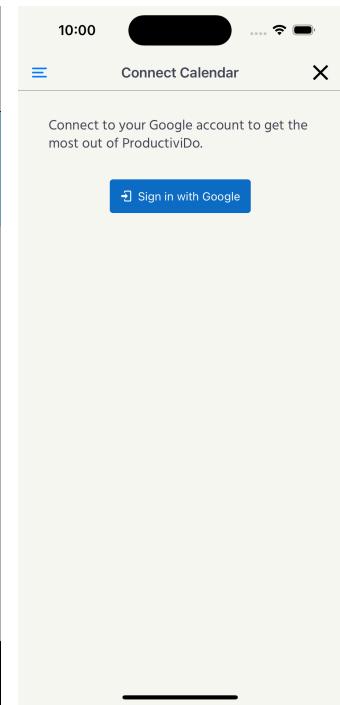
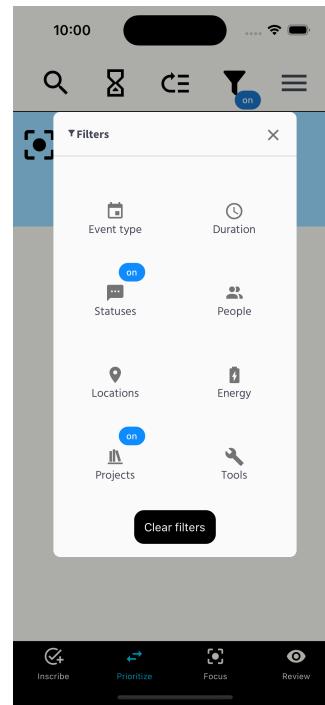
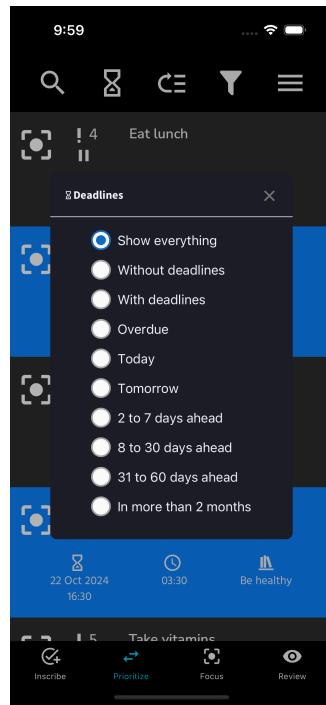
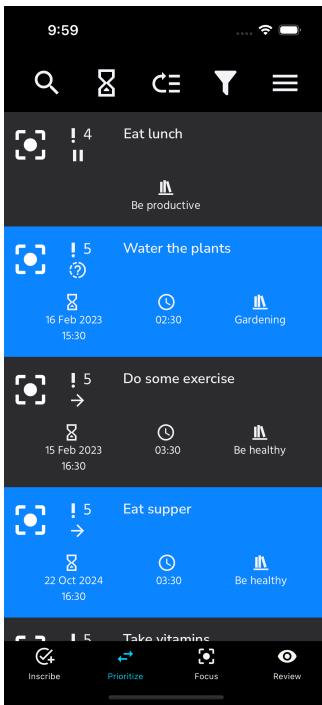


(i) Help and documentation screen

(j) Recycle bin screen

(k) Accomplishments screen

(l) Settings screen

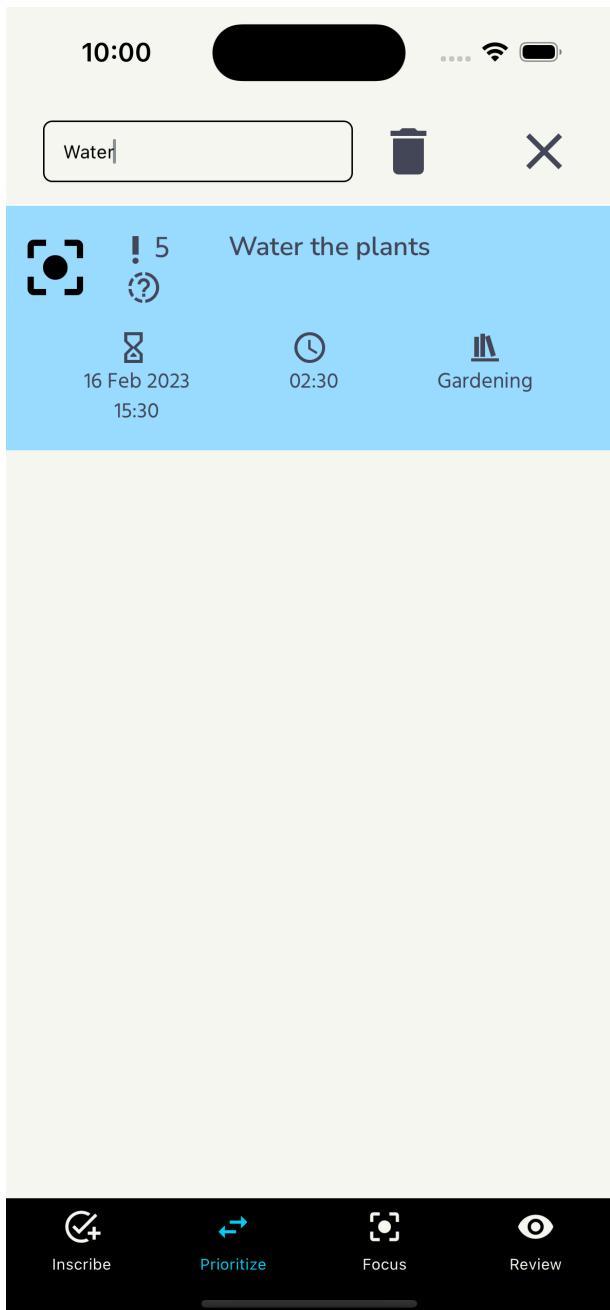


(m) Showcase of the Dark Mode

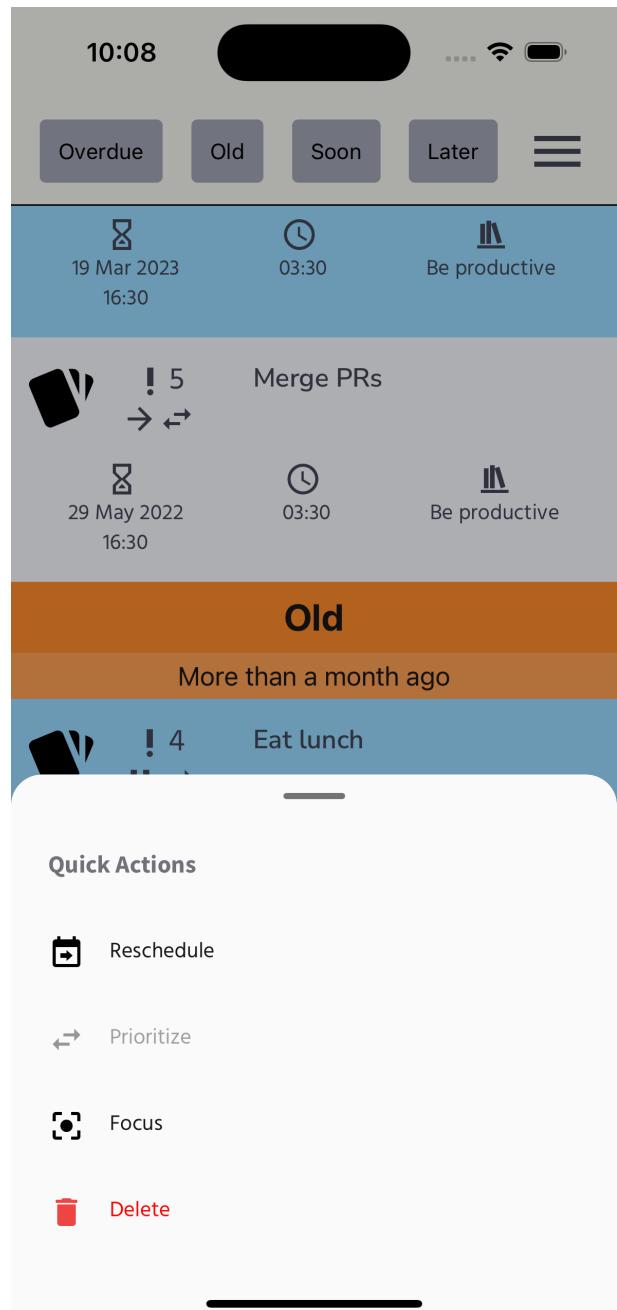
(n) Applying a filter in Dark Mode

(o) Applying multiple filters

(p) Connecting to Google Calendar



(q) Finding tasks by keyword(s)



(r) Applying quick actions on tasks

D Technical highlights: code snippets

D.1 Unit testing Google Calendar integration with Jest

```
test('parsePrimaryCalendarId returns null if no primary calendar could be retrieved', () => {
  expect(
    parsePrimaryCalendarId({
      kind: 'calendar#calendarList',
      etag: 'etag',
      nextSyncToken: 'CICog001nv0CEhNzZ2RsYXZvaWVAZ21haWwuY29t',
      items: [
        {
          kind: 'calendar#calendarListEntry',
          etag: 'etag',
          id: 'en.usa#holiday@group.v.calendar.google.com',
          summary: 'Holidays in United States',
          description: 'Holidays and Observances in United States',
          timeZone: 'America/Mexico_City',
          colorId: '15',
          backgroundColor: '#9fc6e7',
          foregroundColor: '#000000',
          selected: true,
          accessRole: 'reader',
          defaultReminders: [],
          conferenceProperties: {
            allowedConferenceSolutionTypes: ['hangoutsMeet']
          }
        }
      ]
    })
  ).toBe(null);
});
```

D.2 Modularizing functional React components

```
// Shows an error message when the input field is invalid
const InputFieldError = (props: InputFieldErrorProps) => {
  const { colors } = useTheme();
  return (
    <FadeView data={props.errorState}>
      <Text
        style={[
          globalElements.errorInput,
          { color: colors.textError, fontFamily: colors.bodyFontBold }
        ]}
      >
        {props.value}
      </Text>
    </FadeView>
  );
};
```

D.3 Interacting with the SQLite database

```
const getFocusedLocations = (
  setFocusedLocationsFunc: React.Dispatch<React.SetStateAction<IdName[]>>
) => {
  db.transaction(
    (tx) => {
      tx.executeSql(
        `SELECT DISTINCT locations.id, locations.name
        FROM locations
        INNER JOIN tasks ON tasks.id = locationsTasks.taskId
        INNER JOIN locationsTasks ON locations.id = locationsTasks.locationId
        WHERE tasks.taskState = ${taskStatesFromString.focused}`,
        [],
        (_, { rows: { _array } }) => {
          setFocusedLocationsFunc(_array);
        }
      );
    },
    (error) => {
      console.log('db error load focused locations');
      console.log(error);
    },
    () => {
      console.log('loaded focused locations');
    }
  );
};
```

D.4 Leveraging TypeScript types

```
export type TaskWithComputedProperties = {
  locations: number[];
  people: number[];
  tools: number[];
} & Task;

export const valueIsTaskType = (
  value: any
): value is keyof taskTypesFromNumberType => {
  const taskTypes = Object.values(taskTypesFromNumber);
  return taskTypes.includes(value);
};

// Create a type that is a subset of T with all optional properties, but with at least
// the properties K being required
export type AtLeast<T, K extends keyof T> = Partial<T> & Pick<T, K>;

export type PartialTaskGoogleCalendar = AtLeast<
  Task,
  'title' | 'deadlineAppDate'
>;
```

D.5 Consuming React Context, persisted Redux store and theme

```
if (isDBLoadingComplete) {
  return (
    <StrictMode>
      {/* Provide a global context while the app is in use to access data
       from the database */}
      <TasksContextProvider>
        {/* Provide the Redux store to the app */}
        <Provider store={store}>
          {/* Persist the Redux store to the device's local storage */}
          <PersistGate loading={null} persistor={persistor}>
            {/* Provide the NativeBase UI library to the app */}
            <NativeBaseProvider theme={theme}>
              <ThemeProvider>
                {/* Render the app's navigation */}
                <NavigationAndStatusBar />
              </ThemeProvider>
            </NativeBaseProvider>
          </PersistGate>
        </Provider>
      </TasksContextProvider>
    </StrictMode>
  );
}
```

D.6 Using nested contexts and custom hooks

```
const useInscribeManager = (navigation: any) => {
  // Redux state
  const calendarPrimaryId = useSelector(selectCalendarPrimaryId);

  // Auth token
  const { accessToken } = useAuthentication();

  const {
    projects,
    locations: allLocations,
    people: allPeople,
    tools: allTools,
    addNewTask,
    addDelegateeToTask,
    addLocationToTask,
    addPersonToTask,
    addToolToTask
  } = useContext(TasksContext);

  // ...
}
```

E SQLite database schema

Tables (10)	
>  delegatesTasks	CREATE TABLE delegatesTasks (id INTEGER PRIMARY KEY AUTOINCREMENT,delegateeId INT NOT NULL,taskId INT NOT NULL)
>  locations	CREATE TABLE locations (id INTEGER PRIMARY KEY AUTOINCREMENT,name TEXT NOT NULL)
>  locationsTasks	CREATE TABLE locationsTasks (id INTEGER PRIMARY KEY AUTOINCREMENT,locationId INT NOT NULL,taskId INT NOT NULL)
>  people	CREATE TABLE people (id INTEGER PRIMARY KEY AUTOINCREMENT,name TEXT NOT NULL)
>  peopleTasks	CREATE TABLE peopleTasks (id INTEGER PRIMARY KEY AUTOINCREMENT,personId INT NOT NULL,taskId INT NOT NULL)
>  projects	CREATE TABLE projects (id INTEGER PRIMARY KEY AUTOINCREMENT,name TEXT NOT NULL)
>  sqlite_sequence	CREATE TABLE sqlite_sequence(name,seq)
>  tasks	CREATE TABLE tasks (id INTEGER PRIMARY KEY AUTOINCREMENT,title TEXT NOT NULL,type INT NOT NULL,importance INT NOT NULL,
 id	"id" INTEGER
 title	"title" TEXT NOT NULL
 type	"type" INT NOT NULL
 importance	"importance" INT NOT NULL
 status	"status" INT NOT NULL
 deadlineAppDate	"deadlineAppDate" TEXT
 deadlineAppTime	"deadlineAppTime" TEXT
 duration	"duration" TEXT
 projectId	"projectId" INT
 taskState	"taskState" INT
 energy	"energy" INT
 notes	"notes" TEXT
 created	"created" INT NOT NULL
 gcalendarCalendarId	"gcalendarCalendarId" INT
 gcalendarEventId	"gcalendarEventId" INT
>  tools	CREATE TABLE tools (id INTEGER PRIMARY KEY AUTOINCREMENT,name TEXT NOT NULL)
>  toolsTasks	CREATE TABLE toolsTasks (id INTEGER PRIMARY KEY AUTOINCREMENT,toolId INT NOT NULL,taskId INT NOT NULL)

Figure 13: Showing the database design behind ProductiviDo.

F Interface design with Figma

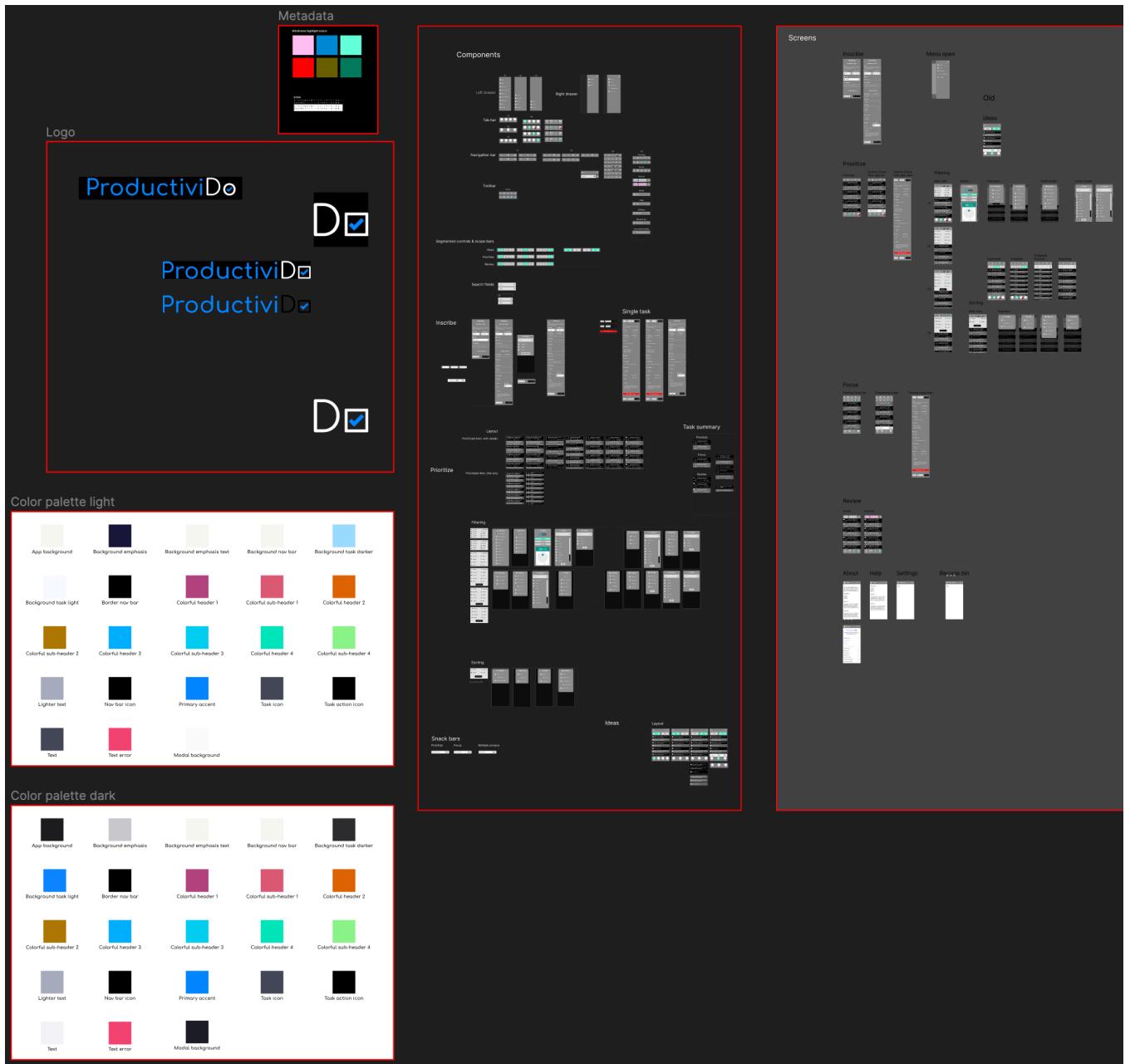


Figure 14: Showing the design of the app's interface in Figma.