Landing page (when not signed up)

The first website I looked at was Monday.com. Upon accessing the landing page, it revealed its product features, positive reviews from past users and companies they're affiliated with. These together uplift the image of the company. The description of features do a great job of letting the user know whether what they offer can match their needs

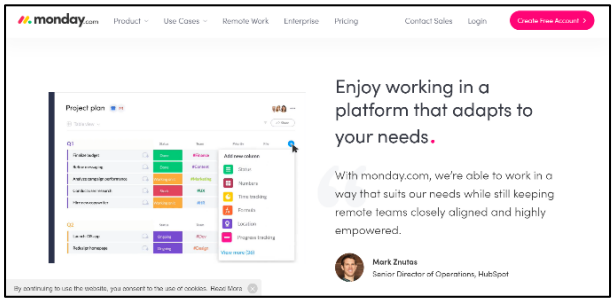


Figure 1: Explaining the features on the homepage

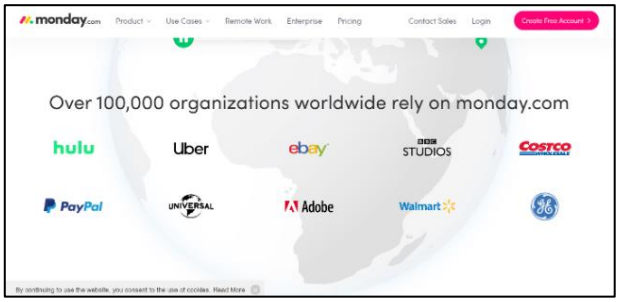


Figure 2: Showing affiliations on the homepage

What I liked about this landing page was its visually appealing design; dark font colour in a white background to increase contrast and therefore readability; professional fonts; an even distribution of headers and paragraphs to reduce large blobs of text and well cropped/rendered and laid out images to provide another way of conveying information. Along with the information presented on the landing page, the visually appealing design increase the professionalism of the website and therefore build user trust.

Its landing page is also a sign up/log in page. Unlike the many other buttons in the website, the sign up button is the only button that is coloured. This is to give the user direction, as the website relies on the user to sign up first before their roster system can be used.

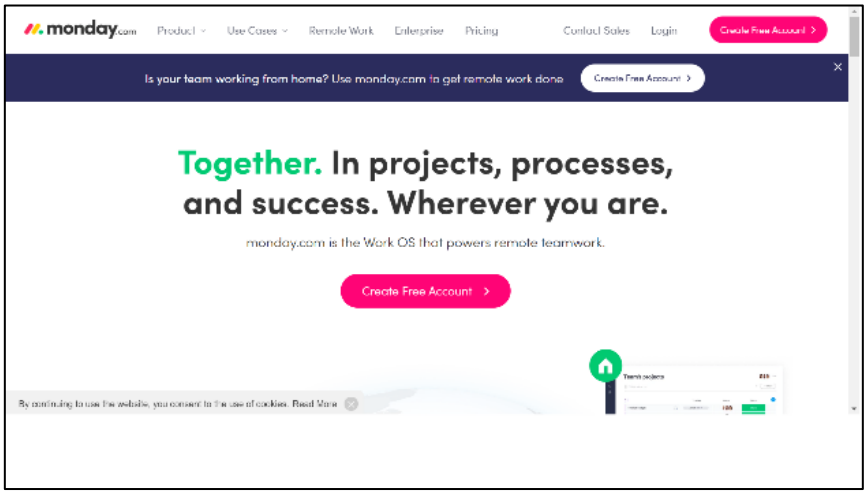


Figure 3: Sign up interface on the homepage

The landing page being a sign up page makes a lot of sense because in order to view a homepage that is a roster, the user would have to be known first so the website knows which roster to display (There is even a sign up button at the bottom of the page, visible after the user has read all of what the website offers). The sign up process is smooth in that it asks for a few pieces of information at a time. This prevents overwhelming the user. In my website, as the sign up will be through social media, these will transform into avaialability, tasks that can be performed and more.

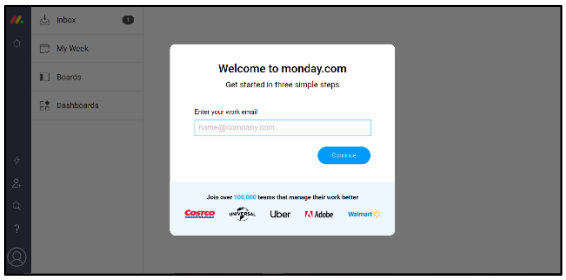


Figure 4: Asking for email only first when signing up

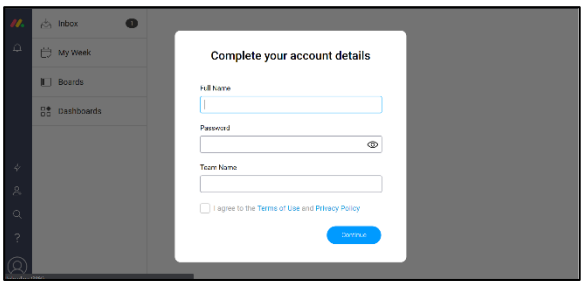
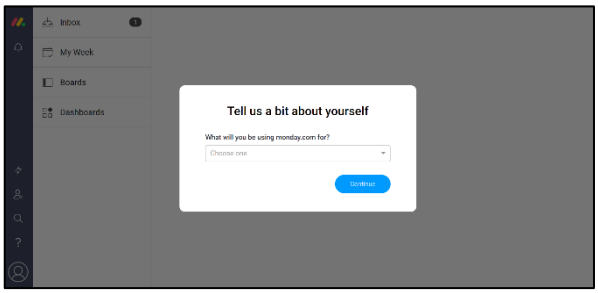


Figure 5: Asking for name only when signing up



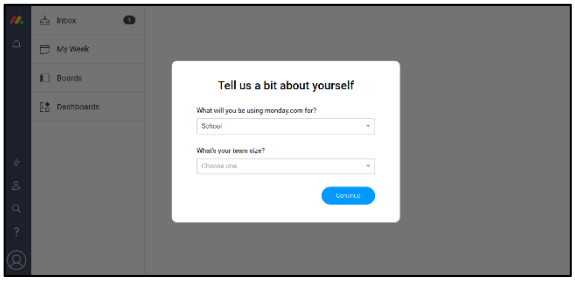


Figure 6 and 7: The second question will not show up unless the first question was answered.

The second website I looked at was teamwork.com. This website's homepage is almost a copy of Monday.com. Like the former, it has a professional font, a dark font colour on a white background, an even distribution of headers and text to showcase its features and well rendered and laid out images. Its signup button, titled 'Try now for free' is also an eyecatcher of the homepage of the website. It further builds on this by making the header remain visible as the page is scrolled through. The presence of these similarities show that the elements aforementioned are major elements in order for roster websites to succeed.

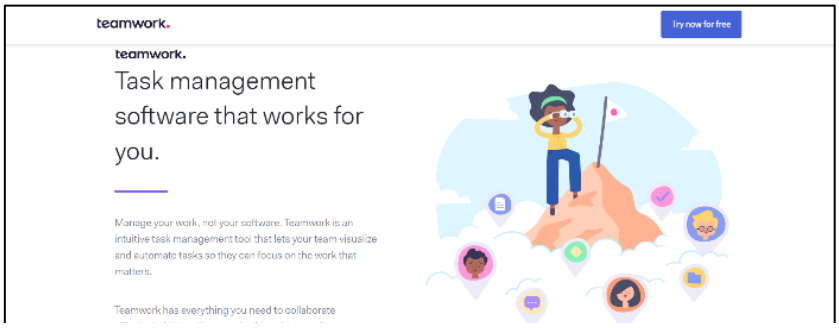


Figure 8: The clean homepage of teamwork.com includes a navigation bar that stays in place, a neat heading and text for information, a prominent signup button and a visually appealing design

The third website I looked at was Atlassian.com's Jira. Its landing page is visually appealing, containing all elements aforementioned in the first two websites. It uses themes similar to teamwork.com: cartoon images and a modern looking palette. Like the first two websites, Jira's landing page showcases its features and its sign up button is also prominent.

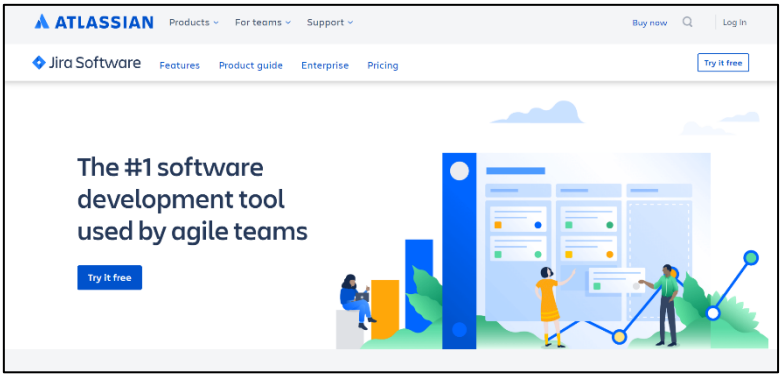


Figure 9: Professionally laid out homepage of Jira

Task system

The design of the Monday.com's roster system is modular. It consists of boards, projects, and tasks. A user can have many boards, boards can contain many projects, and projects have tasks. The boards are presented as the main page, containing projects represented as tables, and tasks are presented as rows of the projects table. I like it because its easy to navigate, comprehend and makes sense as I use it.

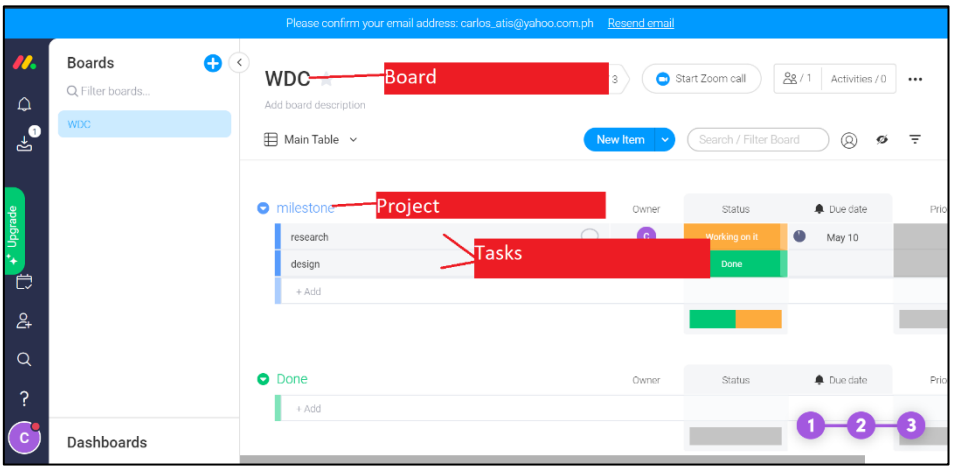
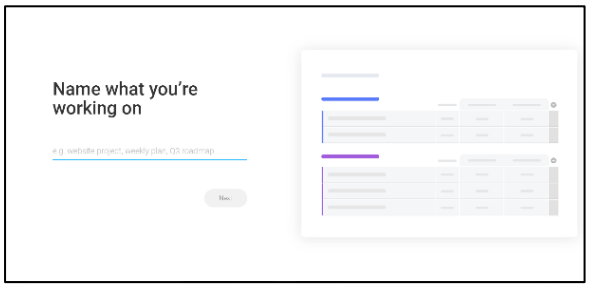


Figure 10: Homepage of Monday.com showing board, project and tasks

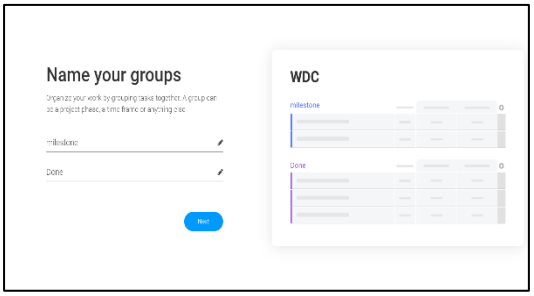
This modularity creates a good and flexible roster system which can be used anywhere. The user may have a board for his hotel environment (managing employees) and another for his personal environment (managing exercises for fitness). An example is provided below: A board named "WDC", inside "WDC" I have a project called "milestone", and inside milestone I have tasks called "research" and "design".

Like the sign up, this was also done in step by step (or page by page) to prevent overwhelming the user.

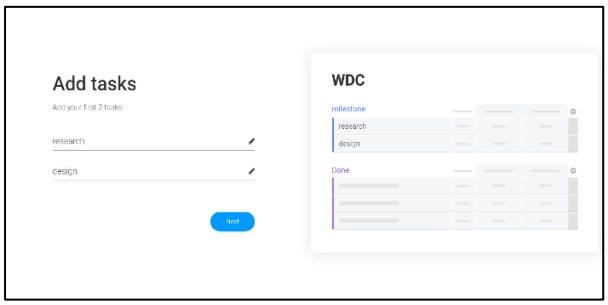
1. Creating the first board (The board was named "WDC")



2. Creating the first project in the board (The project was named "milestone")



3. Creating the first two task in the project (The tasks were named "research" and "design")



The tasks have attributes that are capable of describing the task to a great detail. These include progress, person responsible, location and due date. The user can also add customised attributes, one such attribute can be "priority" and it can have values such as "high" "low" "urgent". These attributes are presented in columns of the project table. The website does a good job of customisability by allowing the user to change the size and order of these columns.

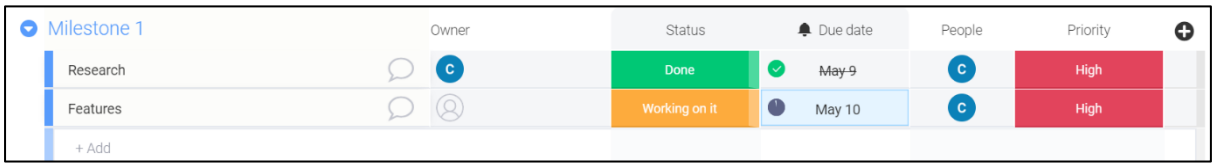


Figure 11: Project table named Milestone 1 showing tasks and their attributes which are represented as columns

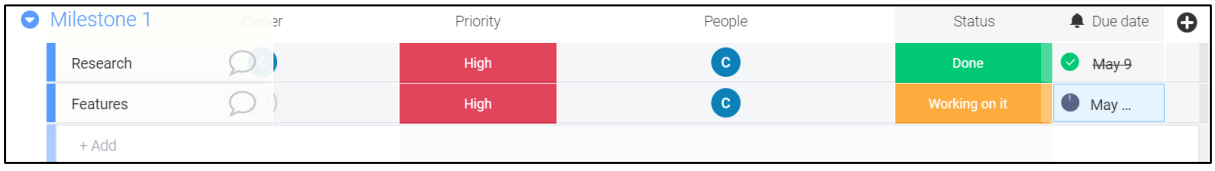


Figure 12: The people column is wider and has changed position to the 4th column

The second website I looked at was teamwork.com. This website usefully provides many settings however it confused me a lot. This main page of a project has a lot of attributes and buttons

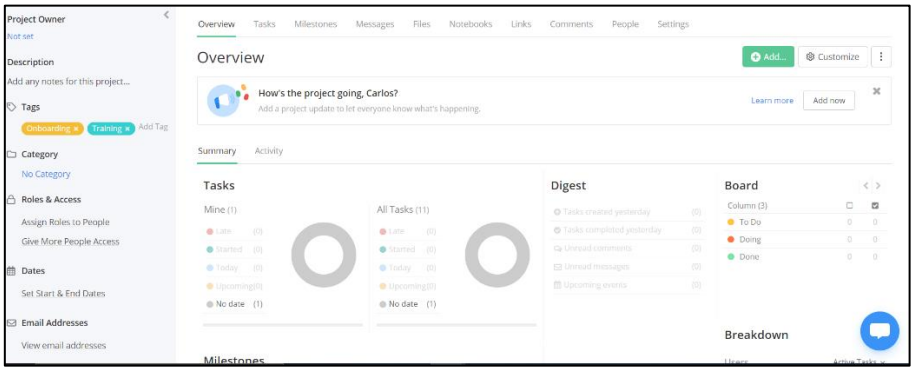


Figure 13: The main page of a project. Navigation buttons on the top and left side of the page and information showcase in the middle

Its modularity is different from monday.com. It has no board, the second element down is projects, then below projects is task list and then tasks. From what I gather, this will be useful in dividing and therefore organising the project. In an example of a time based project, task lists can serve as stage 1, stage 2 and so on. And each tasks are the 'tasks' required to complete that stage. In another example, a project that is divided by departments. One task list for the accounting department, one for coding and so on. Below is an example I made of project WDC, task list named documentation and task named research.

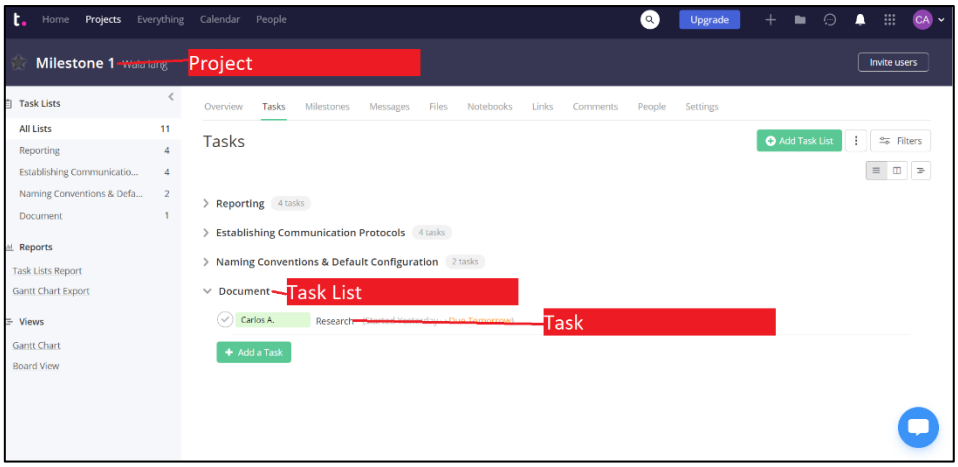


Figure 14: The 'tasks' tab inside the project "milestone 1".

Each task has a person who takes responsibility. In the above example, I took responsibility of the research task. I can click the tick button, signifying the task was finished and visually, the task name would simply be crossed out. If I decide the task was unfinished, the line that crosses out disappears and the task is more visible again.

The third website I looked at was Atlassian.com's Jira. This website is specific for software development. It does a good job of simulating a working environment of a software developer, seen by the buttons on the side of the page and the attributes linked to creating a task such as issue type, linked issue and versions.

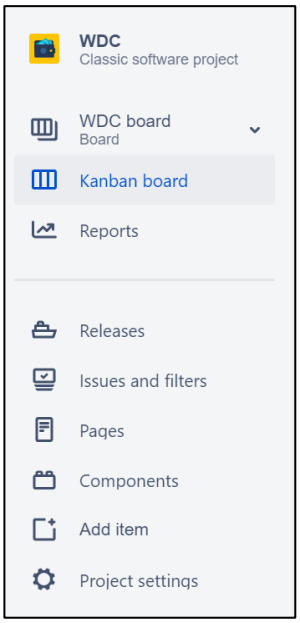


Figure 15: Side bar showing development related options

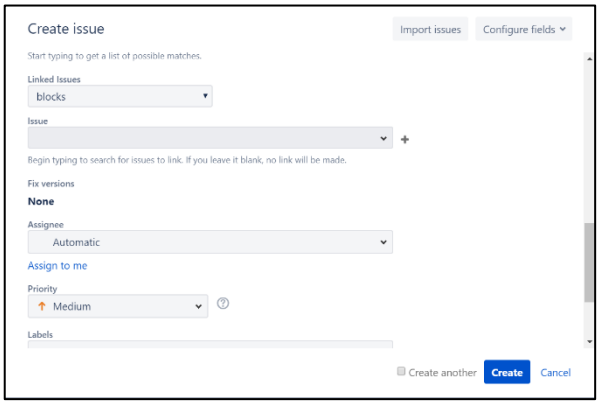


Figure 16: Task creation window showing software development related options

Unlike the first two websites, the tasks UI for Jira can be expanded and be represented as cards. This emphasises details on the task. It also improves communication as it makes commenting on the task easier. The tasks weight/usefulness can be increased as users can upload files onto it.

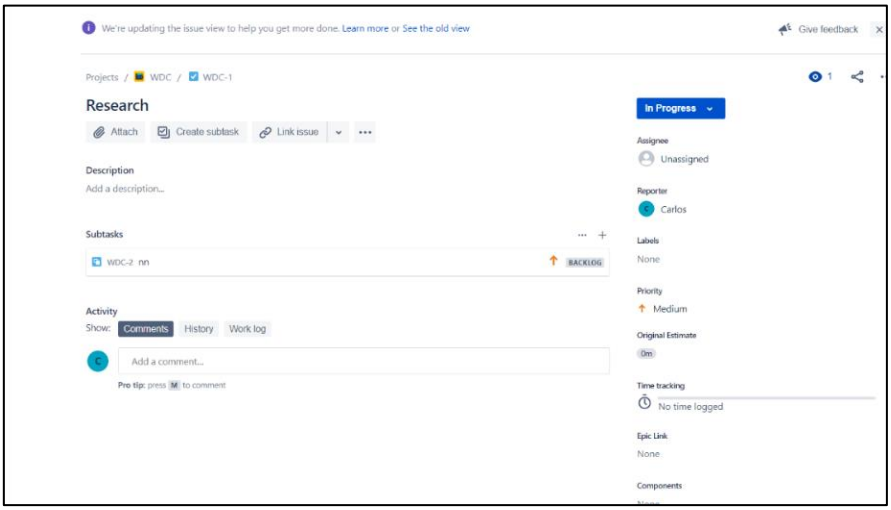


Figure 17: Task called 'Research' represented as card showcasing its attributes on the right side, comments on the bottom side and file options underneath the name