The Problem

- **Students -** need to juggle assignments, exams, and extracurriculars with varying deadlines and importance.
- **Freelancers** - manage multiple clients and deliverables with overlapping deadlines and varying importance.
- **Individuals -** struggle to prioritize tasks across multiple life areas (e.g., work, personal, health) with varying urgency and importance.
- **Event planners -** need to prioritize tasks (e.g., venue booking, catering, promotion) based on urgency and impact.

User Stories

- As a user I want a secure login system to protect my data.
- As a user I want to:
- have clean simple navigation.
- create tasks,edit and, delete tasks.
- have tasks with a title and description.
- have none or many tags on each task.
- view all tasks in order based on tags.
- create, edit and delete tags.
- have tags with a title and description.
- As a user I want to change a tags importance.

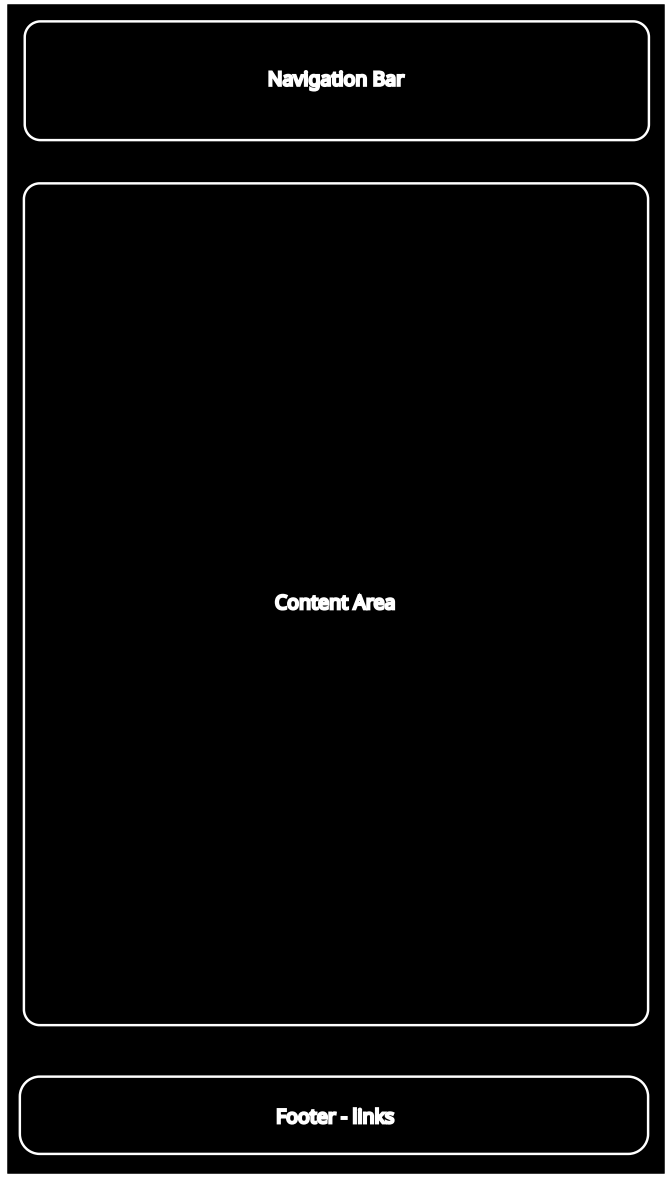
The Solution

- An app for your phone, laptop or desktop.
- Built with **next.js** - providing server side rendering.
- Using **pg** - providing PostgreSQL database access.
- Using **clerk** - providing user accounts and authentication.

Minimal clutter

- main navigation at the top
  -- Sign in/up, Tasks, Tags, New and About
- Content area to show:
  -- the list of tasks
  -- the list of tags
  -- Create edit Tasks and tags
- Footer, room for external links

All buttons, are routes with the exception of the editors forms.



Routes

- Tasks /
- Tags /tags
- New /new
- About /about
- Each Task has a dropdown section holding Edit Delete and Tags buttons
  -- Edit /edit/36
  -- Delete /delete/36
  -- Tag /tag/36

Data storage

- **PostgreSQL database** provided by Supabase.com
- **Three Tables**
  -- tt_tasks
  -- tt_tags
  -- tt_tag_links

Authentication

- The clerk ID provided by auth() is stored on each table record.
- Allows all accounts to the same tables.
- Ensures only logged in accounts can access the data.

Live Demo ...

Reflection

I used a markdown file to track my progress

I hate YouTube shorts ... but ... <details><summary> tags

I learned about Begin,Commit and Rollback

I learned about indexing

I learned that SQL is way more powerful than I first thought ...

```
`SELECT t.*, COALESCE(BIT_OR(g.rank), 0) as task_rank
    FROM tt_tasks t
    LEFT JOIN tt_tag_links tl ON t.id = tl.task_id AND tl.cid = $1
    LEFT JOIN tt_tags g ON tl.tag_id = g.id AND g.cid = $1
    WHERE t.cid = $1
    GROUP BY t.id
    ORDER BY task_rank DESC, t.id ASC`,[userId]
```