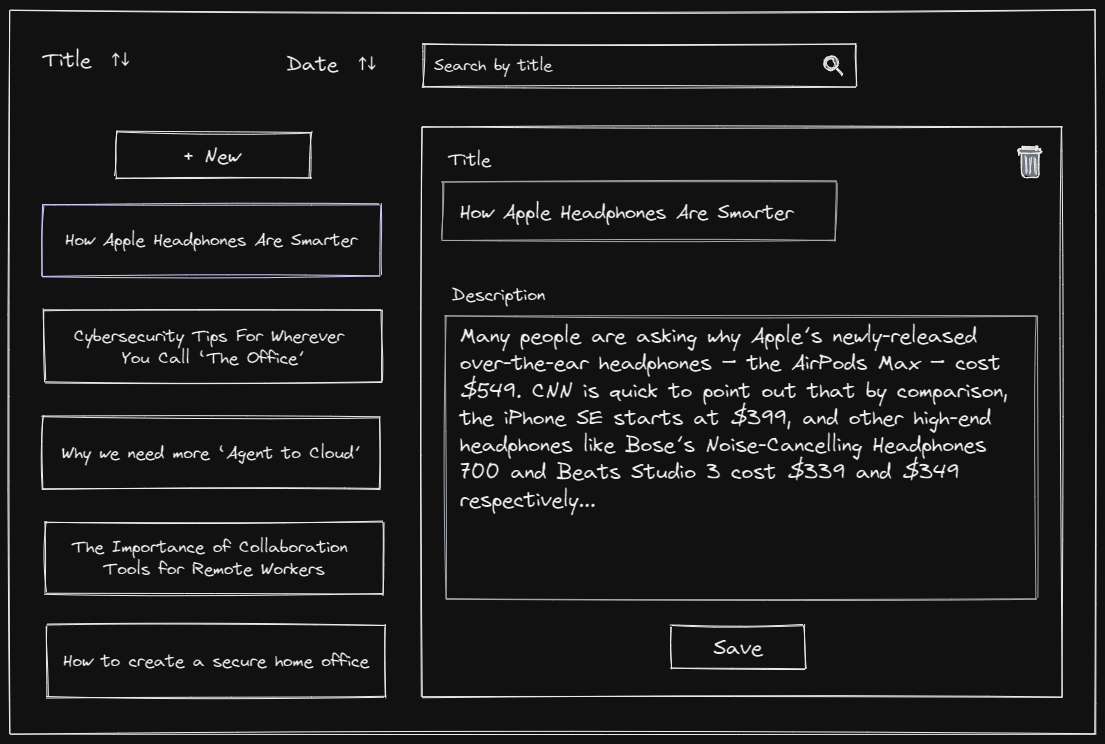
**Test task execution:**

1. Creating a GitHub repository;
2. Developing the test task;
3. Sharing your GitHub repository with reviewers;
4. Reviewing the code of the test task;
5. Receiving feedback.

**Test task:**

# **Note Editor App**



## **Project description:**

The main project purpose is the implementation of simple CRUD operations. In this project, the users expect that they can operate over notes entities. The "core features" give them basic functionality for managing notes and are essential for project completion. While the "advanced features" could extend functionality for more handy usage of the application, they aren't mandatory.

## **Feature List:**

* Core features (Required)
  + Notes list
  + Note Details
  + Add note
  + Edit note
  + Remove note
* Advanced features (Optional)
  + Local Storage
  + Quick Search
  + Sorting

### **General:**

* Consists of 2 sections:
  + Left: List of notes and “New” button
  + Right: Note details and “Delete” button
* Design should be implemented according to the wireframe
* Notes records only exist in memory. Data persistence is not mandatory.
* On the first enter user expects to see
  + Empty list of notes
  + “New” button

### **Left section:**

User should be able to see notes list and add a new one

* “New” button
  + on click, add a new empty note to top of the list with prefilled dummy title and select this note.
* Note List
  + consist of note items
* Note Item
  + display only title
  + should be highlighted on selection
  + on click - select this note and open the detailed note view

### **Right section:**

User should be able to Delete and Edit notes. The section displays info according to the selected note.

* Edit Note
  1. Fields.
     + Editable by default
     + Title
       - Text input
     + Description
       - Textarea
  2. Save
     + Will update note record
       - update note record at the note list
       - keep current note selected
* Delete note
  1. Delete the note from the note list
  2. Select the next available note in the list
     + No selection for an empty list

## **Advanced options:**

* Edit Note Validation
  + Title
    - Min/Max validation: 3 - 120 chars
  + Description
    - Min/Max validation: 5 - 500 chars
  + Any validation error should highlight invalid input
  + Save button
    - Disabled by default until form is valid
* Local Storage
  + Keep notes at the local storage. After page refresh, all records that were previously added, should remain.
* Quick search
  + Provide an ability to search across the notes. Users would search by the title field across the notes list.
* Sorting
  + Asc/Desc sorting by title or/and date created in
  + Provide an ability to sort the notes list by title and date created, both for asc and desc directions.
  + By selecting specific sorting, update the displaying of the sorting button according to the sorted direction.
    - In general, sorting should has three states
      * asc
      * desc
      * none
    - By default sorting should has state none

### **Tech requirements:**

Language: JavaScript or Typescript

The technology stack is up to the developer.

No restriction for frameworks, preprocessors, and third-party libraries.

## Wireframes

Link: <https://excalidraw.com/#json=5689304743411712,J8GLzylwMKL7x7T_hd0bYA>