

# Stakeholders

A task organization application, such as the one that we will develop, caters to the needs of a variety of user audiences. Project managers can use the application to effectively set and track project goals and milestones as well as share them with the other project members. Teachers can use the application to facilitate off-site, and on-site, learning activities for many students and classes. While the contexts of use, and their respective intended audiences, extend beyond the two aforementioned examples, we identified five extreme end-user groups that should be considered as stakeholders for this product:

**Generating users** – end-users that focus on generating, managing, and disseminating content. Such users would value features that will facilitate such functions.

**Viewing users** – end-users that focus on accessing and viewing already existing content on the application. Such users would value features that will allow them to easily navigate the various boards of the application, view and navigate tasks within the board itself, and perhaps share and record boards that they find of interest.

**Basic users** – end-users that will prefer to use the application with minimal set-up and customization. Such end-users would trade the flexibility of features in favor of their ease of use and rapidity.

**Advanced users** – end-users that will prefer to customize and cater the application to their own needs and preferences. Such end-users favor enhanced customization of the functionalities and features included in the application such as, but not limited to, binding keyboard shortcuts and changing color themes.

**Supervising users** – end-users that will prefer to use the application to manage and track the progress of other users. Such users will favor features that will allow them to create the outlines of boards as well as tools to share them with other users.

Aside from the user groups, administrators are another group of stakeholders that is responsible for maintaining and moderating the application. We have identified two specific roles that administrators can take that should be considered:

**Maintainers** – administrators that are responsible for the upkeep and maintenance of the application's server.

**Moderators** – administrators that are responsible for maintaining and moderating the data stored in the application. Such administrators would favor tools and functionalities that facilitate the viewing, update, deletion, and management of data.

# Terminology

**Card** – The purpose of a card is to represent a goal or task that needs to be accomplished. All cards must have a title that is not empty. All cards share the same upper limit constraint on the length of the title. Cards can only be contained inside a column. Additionally, cards can also contain a description with a universal constraint on their maximal length. Each card can also have its own background color that can be selected from a limited premade list.

**Column** – The purpose of a column is to organize cards within a board. A column is an ordered list of cards with a unique identifier and a name. Columns can only be contained inside a single board. All columns have an upper limit on the length of the title and the maximum number of cards the column can contain. Upon creation, by default, the column does not contain any cards. A column is considered to be empty if it contains no cards. In such a case, the column will display text in its background to prompt the user to add new cards to the column.

**Board** – The purpose of a board is to distinguish a unit of cards and their organization in columns. A board consists of a unique identifier and a name, with a constraint on character length. Moreover, each board also consists of a color theme that is chosen from a selection of prefabricated themes and that is also imposed on the columns of the board. All boards also share constraints on the maximal number of columns that can be displayed at once and that a board can contain. Upon creation of a new board, the default case is to have three columns titled 'To Do,' 'Doing,' and 'Done' respectively. By default, there will also be a sample card in the 'To Do' column. A board is considered empty if it does not contain any columns; in such case, the board will display a message in the background that will prompt the user to add new columns.

**Task** – The purpose of a task is to represent a sub-goal of a card that contributes to the completion of the overall goal described by the card. Each task contains a mandatory title that describes the sub-goal as well as a 'is complete' field. All tasks share a constraint on the maximal length of the title. Tasks can only be contained in a single card, with a further constraint on the total number of tasks a card may contain.

**Tag** – The purpose of a tag is to identify groups of cards on a board. Each board has its own list of tags, with a constraint on the maximum number of tags it can contain. When a board is created, its corresponding tag list is empty. Tags can be assigned to cards within the board, with an additional constraint on the maximal number of tags that may be attached to a single card. A tag contains a mandatory unique title that distinguishes it from others as well as a color that can be selected from a premade list of colors. All tags share a constraint on the maximal length of the title.

**Multi-Board** – Multi-board refers to a board that can be accessed by multiple users simultaneously.

**Password-protected Board** – Password-protected board refers to a board that has an associated password that grants a user permission to modify and edit the board.

**Main Overview** – First/initial view refers to the UI view that the user first encounters when opening the client application.

**Board Overview** – Board view refers to the UI view that displays the columns and cards of a selected board. In this view, the user may be able to view and modify all aspects of the board.

**Card Overview** – Card view refers to the UI view that displays all aspects of a specifically selected card. It allows the users to view and modify these aspects.

# Epics and User Stories

## 1. Epic – Basic features

*This epic concerns itself with the bare-bones features that are necessary in order for this application to function properly. The epic's scope extends to both front-end and back-end features, as well as the essential features as required by the client.*

1. As a user, I want to be able to access the application through my own device so that I can access all of the features and functionalities it provides.
2. As an administrator, I would like client applications to be able to send and retrieve data from the server so that users' data can persist and be accessed by them to ensure the proper functioning of the application
3. As a user, I want to have an interactive and intuitive graphical user interface so that I can easily use the application for my own purposes.

### Acceptance Criteria

- When the application is run, the user is shown the main overview (see fig. 1.)
  - There are dedicated overviews that display the contents of a specific board (see fig. 2.) and of a specific card (see fig 3.)
    - Users can access the separate overview for a specific card by clicking on the icon which appears on the card.
    - To simplify the board overview, users can view detailed information regarding the card in the card overview instead of the board overview.
  - When an action cannot be done or when the application encounters an error, the user is notified through the UI (see fig 4.a.; 4.b.)
  - The graphical UI interface clearly presents the available functionalities (see fig. 5.; 6.; 7.)
4. As a user, I want to be able to access the main overview directly when I open the client application so that I don't have to register or sign in
  5. As a user, I want to be able to access boards without needing to register or sign-in so that I don't have to undergo a registration process (see fig. 1).
  6. As a user, I want to be able to make modifications to a board's content without registering or signing-in so that I don't have to undergo a registration process.
  7. As a user, I want to be able to connect to and modify boards irrespective of the activity of other connected users so that I can work concurrently with other users.

### **Acceptance Criteria**

- Any user's activity is not impeded by the activity of another user.
8. As an administrator, I want to host multiple users on the same server so that multiple users can use it at the same time

### **Acceptance Criteria**

- The server application can handle many incoming user requests and respond to them with appropriate resources.
  - The server application can resolve conflicts that occur when two or more users attempt to modify a resource at the same time.
9. As a user, I want to have the application automatically display the changes others have made so that my current overview is up-to-date.

### **Acceptance Criteria**

- The overview of any local client applications attempts to reflect the current state as stored in the server's database.
  - The client application can determine whether its current overview is synchronized.
  - The application should periodically check whether it is synchronized.
  - If the client application's overview is not synchronized, then the application will automatically synchronize its overview.
  - The client application should notify the user of the most recent synchronization that has been made.
  - If a synchronization is not possible or an error is encountered during the process, the client application should alert the user that the current over is not up-to-date.
10. As a user, I want to automatically save changes, which I have made, in the server's database so that my local overview is synchronized with the server.

### **Acceptance Criteria**

- The local client application saves changes as they happen to the server.
  - If a synchronization is not possible or an error is encountered during the process, the client application should alert the user that the synchronization of the server with the local overview was unsuccessful.
  - Other users' overviews to display the changes, which I have made, so that their current overviews match my current overview.
11. As an admin, I want the server application to handle and resolve conflicting synchronization attempts made by multiple users so that the overview is consistent across applications.
12. As a user, I want to be able to drag and drop columns within the board view so that I can efficiently and rapidly organize/delete the board's columns.

#### **Acceptance Criteria**

- The columns of the board can be dragged to change the order in which they appear in the board overview (see fig. 2.)
- Any column can be dragged into the bin area (see fig.2., bottom-right) which not only deletes the column itself but also any of the cards it contains.

13. As a user, I want to be able to drag and drop cards within the board view so that I can efficiently and quickly organize the cards in a board.

#### **Acceptance Criteria**

- Cards can be dragged within a column to change the order in which they appear.
- Cards can be dragged across columns to change which column contains the card.
- Cards can be dragged into the bin area in order to delete them.

14. As a user, I want to be able to access, create, modify, and delete boards so that I can maintain boards that are suitable for my own needs and requirements.

#### **Acceptance Criteria**

- Boards can only be created if they satisfy all of their imposed constraints (see Terminology). When a modification is made, it can only be accepted if the constraints remain satisfied.
- When a board is created, it is populated with the columns and cards described in its default case (see Terminology).
- When a user attempts to delete a board, a confirmation dialog window must be displayed to the user in order to confirm this irreversible action (see fig. 4.d.).
- All changes made to a board should be persistent

15. As a user, I want to be able to access, create, modify, and delete cards within a specific board so that I can have my own goals and tasks in the board.

#### **Acceptance Criteria**

- Cards can only be created and modified if they satisfy all of their imposed constraints (see Terminology).
- When a user attempts to delete a card, a confirmation dialog window must be displayed to the user in order to confirm this irreversible action (see fig. 4.d.).
- Any changes made to a card should be persistent in the server and database.

16. As a user, I want to be able to access, create, modify, and delete columns within a specific board so that I can organize the board's cards more efficiently and in line with my needs and requirements.

#### **Acceptance Criteria**

- Columns can only be created and modified if they satisfy all of their imposed constraints (see Terminology).
- When a user attempts to delete a column, a confirmation dialog window must be displayed to the user in order to confirm this irreversible action (see fig. 4.d.).
- Any changes made to a column should be persistent in the server and in the database.

## **2. Epic – Multi-boards**

*As a distributed application, users should be able to collaborate their work across shared boards. This epic describes the necessary features to ensure the efficacy, effectiveness, and efficiency of such a functionality.*

1. As a user, I want to have a graphical user interface that will allow me to search for specific boards using their respective keys or names so that I can effectively and quickly access boards.

#### **Acceptance Criteria**

- The main overview contains a search bar that allows the user to search for a board given its key or its name and to view any matching boards in the boards list (see fig. 1.).
- When a user clicks on a board in the board list presented in the main overview (see fig. 1.), the user is redirected to the corresponding board overview (referring to fig.2.).

## **3. Epic – Cards can have details**

*Tasks and goals can oftentimes be more complicated than mere description. This epic describes the features that can extend the range of possible actions that can be done in a card.*

1. As a user, I want to be able to add a simple description to each card, so that I can further clarify and explain the goal or task represented by the card.

#### **Acceptance Criteria**

- By clicking on the card and entering text in the description box (see fig 3.), users can add text description to a card.
- The description should be editable and erasable as necessary.
- The description should be visible to all users in the card view.
- The description may be empty.
- The description should be displayed in a legible and readable font and formatting.
- The description may not exceed an upper limit on the number of characters.
- The description may only contain common and standard characters.

2. As a user, I want to be able to create and delete tasks for a specific card so that I can further divide the goal represented by the card into manageable sub-goals.

#### **Acceptance Criteria**

- In the card overview (see fig. 3.), the user should be able to view all existing tasks in a clear and simple list.
  - The user should be able to add new tasks if they comply with all of the imposed constraints (see Terminology).
  - The user should be able to delete or modify the contents of a specific task.
3. As a user, I want to be able to view and modify all currently existing tasks and to be able to mark tasks as complete or incomplete so that I can organize better

#### **Acceptance Criteria**

- The user should be able to mark a specific task as complete or incomplete by clicking on its corresponding checkbox.

## **4. Epic – Tag support**

*Features that aid users in quickly identifying and categorizing items are necessary in any application that facilitates organization. This epic describes the functionalities and features that may be implemented through a tag system.*

1. As a user, I want to be able to create tags to a specific card so I can group my cards with ease.

#### **Acceptance Criteria**

- There exists a maximum amount of characters for the tags and the user will be notified when the limit is reached
2. As a user, I want to be able to attach and detach already existing tags to a specific card.

#### **Acceptance Criteria**

- The UI must intuitively display the tags that are attached to a card.
- Users must be able to easily attach a tag to a card and select which tag to attach
- In the card view, users must be able to drag and drop an attached tag into the bin in order to detach the tag.
- The user must be notified and be denied from attaching cards if the maximum number of tags that a card can hold has been reached.
- Whenever a tag is attached or detached from a card in a client application, such a change persists in the server application.



## 5. Epic – Customization

*Some users may prefer to be able to customize their application as much as possible in order for it to be suitable to their taste. This epic details some of the possible customization features that can be implemented.*

1. As a user, I want to be able to customize boards so that they cater to my own preferences and needs.

### **Acceptance Criteria**

- Users should be able to associate a color theme, from a prepared selection of themes, with a board (see fig. 5.)
  - Customization must be applied immediately and persist in the server's database.
  - Text must be legible and readable irrespective of the theme chosen.
2. As a user, I want to customize the color of specific cards so that I can identify them quickly and effectively.

### **Acceptance Criteria**

- Users should be able to associate a color, from a prepared selection of colors, with a card (see fig. 7.).
  - Customization must be applied immediately and persist in the server's database.
  - The color of the card should not impact the legibility and readability of the cards.
3. As a user, I want to be able to have dedicated customization menus (see fig. 5.; 6.; 7.) so that I can view all of the possible customization options and avoid accidental and unwanted customization.

### **Acceptance Criteria**

- Customization menus have preset themes (see fig. 5.), which prevent the user from making accidental and unwanted changes to the appearance of the board. They also prevent the user from turning the visual elements unreadable.
  - Customization must be applied immediately and persist in the server's database. There will exist fields for the customization in the database.
  - The appearance of the visual elements should be expected to correspond to the currently selected theme.
4. As a user, I want to be able to associate a background color to each tag so that I can easily and flexibly identify them.

### **Acceptance Criteria:**

- Colors can only be selected from a pre-chosen list of colors.
- The choice of background color does not hinder the readability of the text
- When a tag is created, a tag is always assigned a background color.
- The background color of the tag can be modified.

- The associated background of the tag is used unconditionally when a tag is displayed on an attached card.
- The choice of the background color in a client application persists in the server application.

## 6. Epic – Keyboard shortcuts/usage

*Apart from interacting with the application using the graphical user interface, users may improve their productivity through keyboard shortcuts. This epic describes the features and functionalities necessary for such a feature.*

1. As a user, I want to be able to have keyboard shortcuts for frequently used features in the client application so that I can save time

### **Acceptance Criteria**

- The application should have a default list of keyboard shortcuts and their corresponding features.
  - Users can access and view their list of keyboard shortcuts and corresponding features
  - When all keys corresponding to a shortcut are pressed simultaneously, the corresponding feature is called and executed.
  - A keyboard shortcut may only be associated with a single feature.
2. As an advanced user, I want to be able to bind specific key combinations to features as keyboard shortcuts so that I can further customize the operation of the client application to my own needs and preferences.

### **Acceptance Criteria**

- Users can modify the existing keyboard shortcuts.
- Users do not need to re-configure the keyboard shortcuts because it is saved between sessions.
- A new keyboard shortcut must not be in conflict with an already existing shortcut.

## 7. Epic – Password-protected boards

*Although users may view any of the boards available in the application, some users might want to prevent others from modifying their own boards. This epic details the features necessary for the implementation of passwords and password-protected boards.*

1. As a user, I want to be able to protect the content of my boards using a password so that other users cannot modify my board.

### **Acceptance Criteria**

- Users can enable or disable password protection for boards that they can modify (see fig. 5., bottom).

- Users must enter a password when password protection is activated in order to modify the contents of the board; however, all users retain the ability to access and view all boards irrespective of password-protection.
  - If a user can modify a board, they may change the board's password to a new one (see fig. 5.).
  - If a user attempts to change a board's password, they must prove that they remember the password by entering it once more (see fig. 5.)
  - The password should be hashed so that the passwords are safely kept in the server's database.
2. As an administrator, I want the passwords to be stored safely (in hashed format) so that the app respects users' privacy.

# Mock User Interface

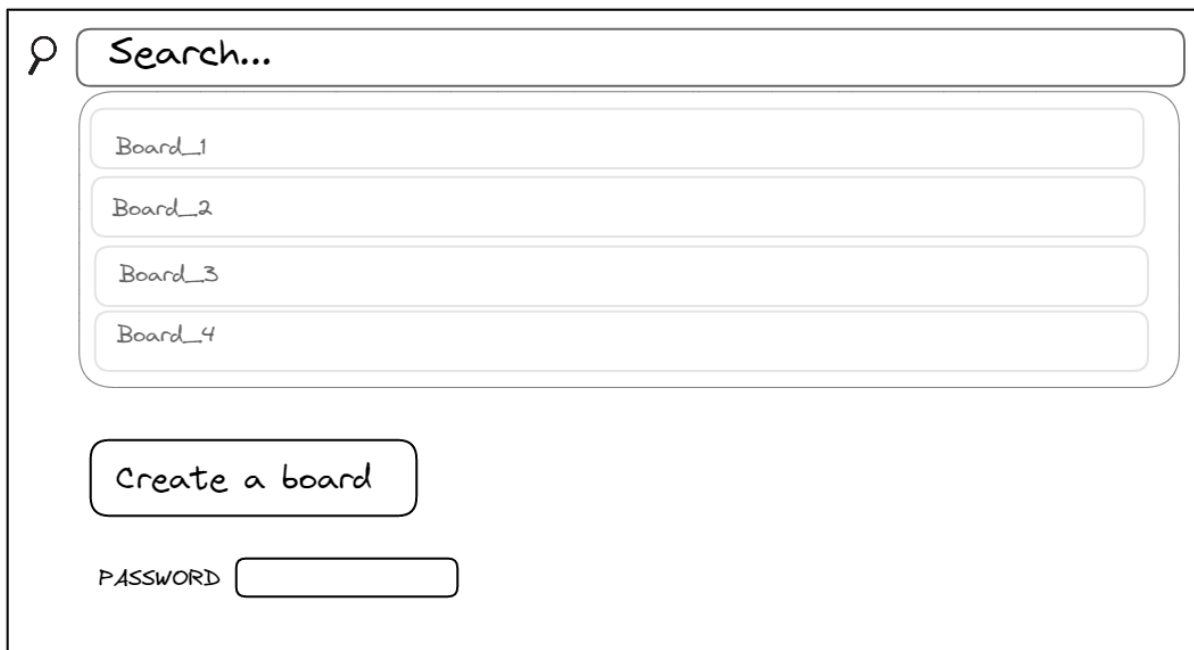


Figure 1.: Mock UI for the main overview

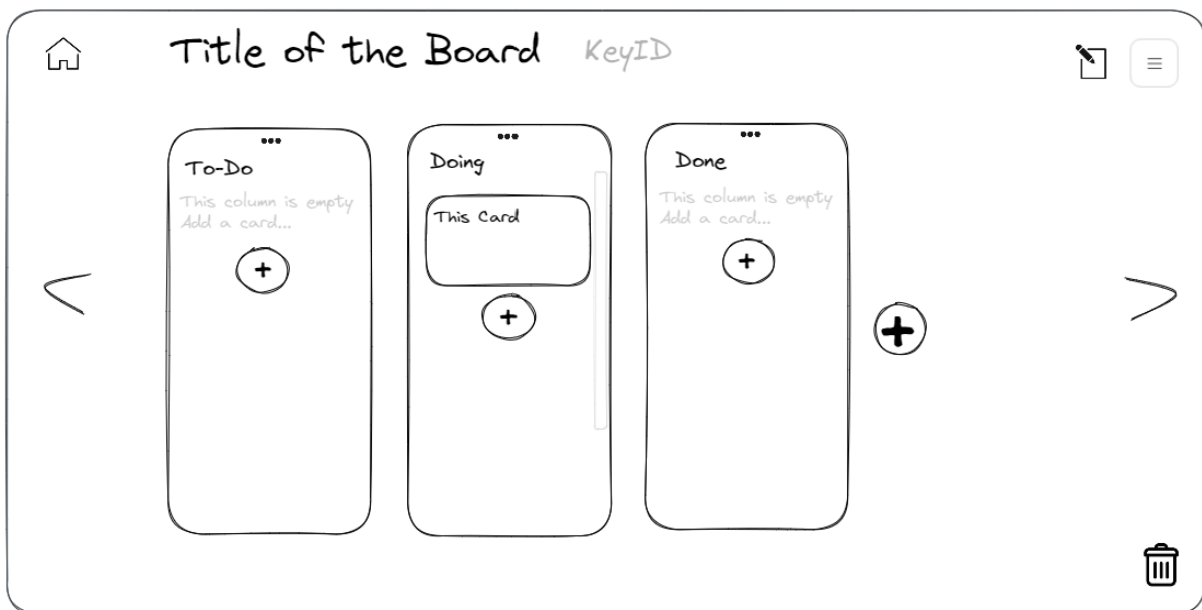
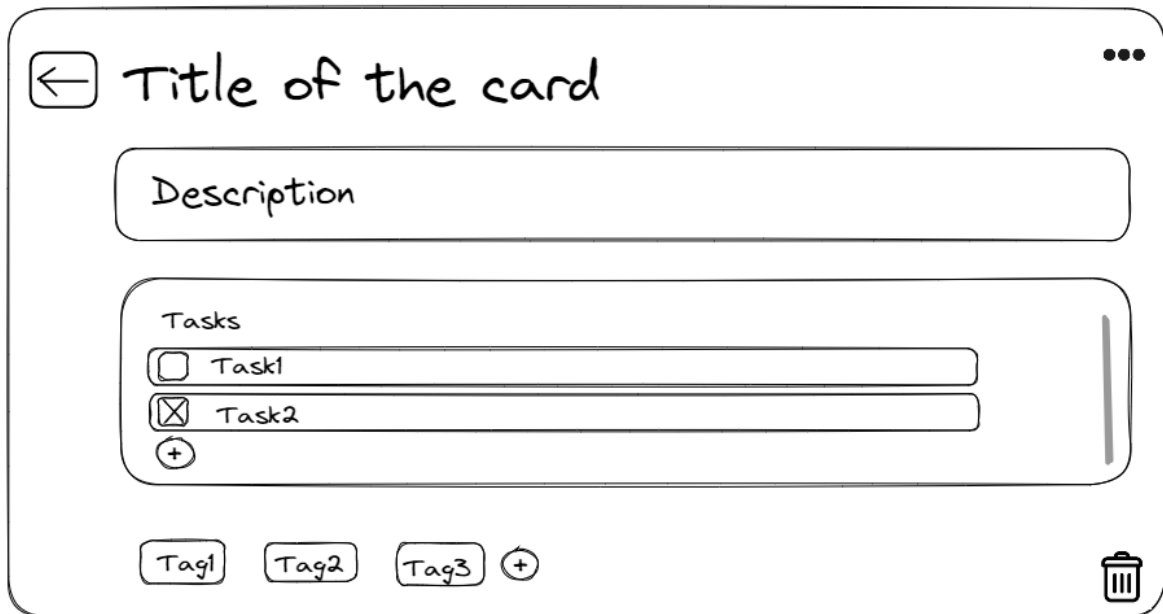
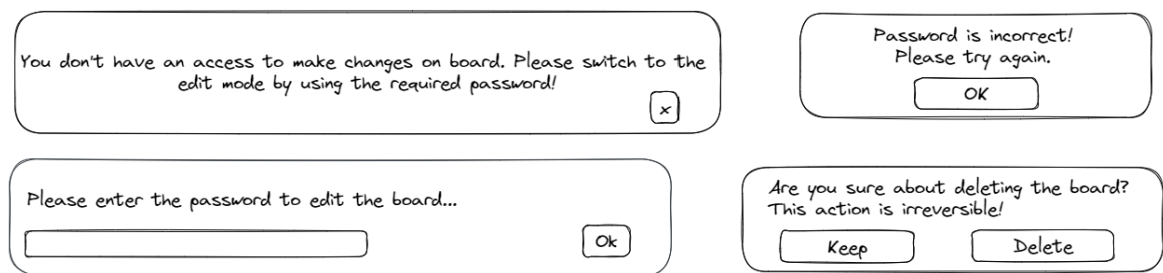


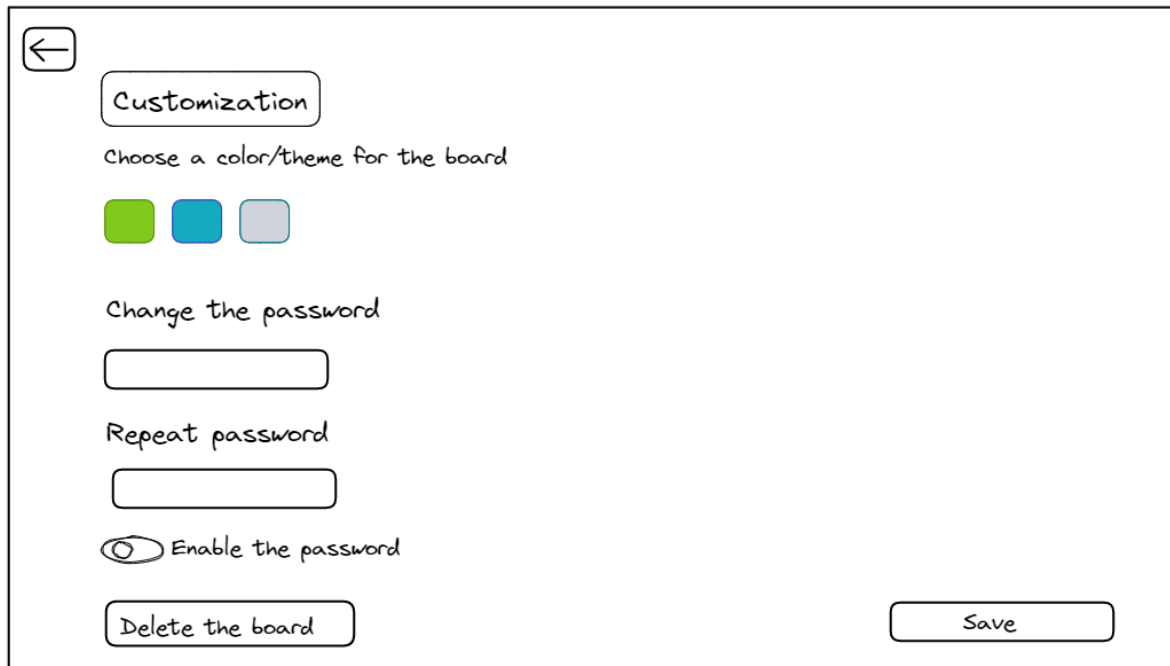
Figure 2.: Mock UI for the board overview



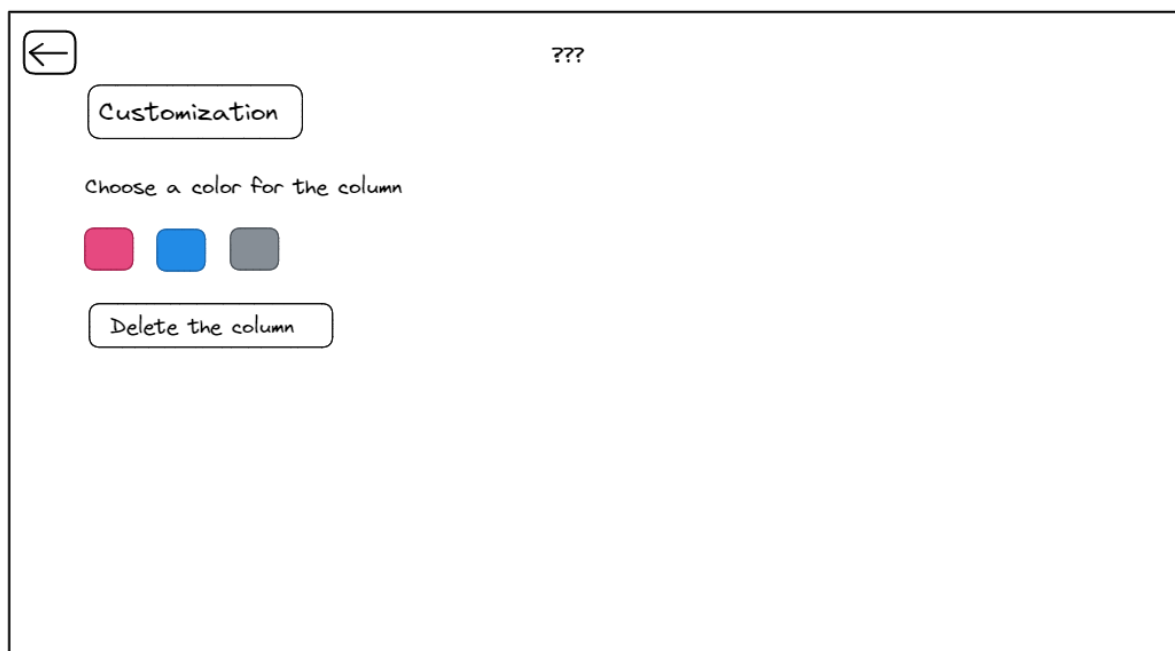
**Figure 3.:** Mock UI for the card overview



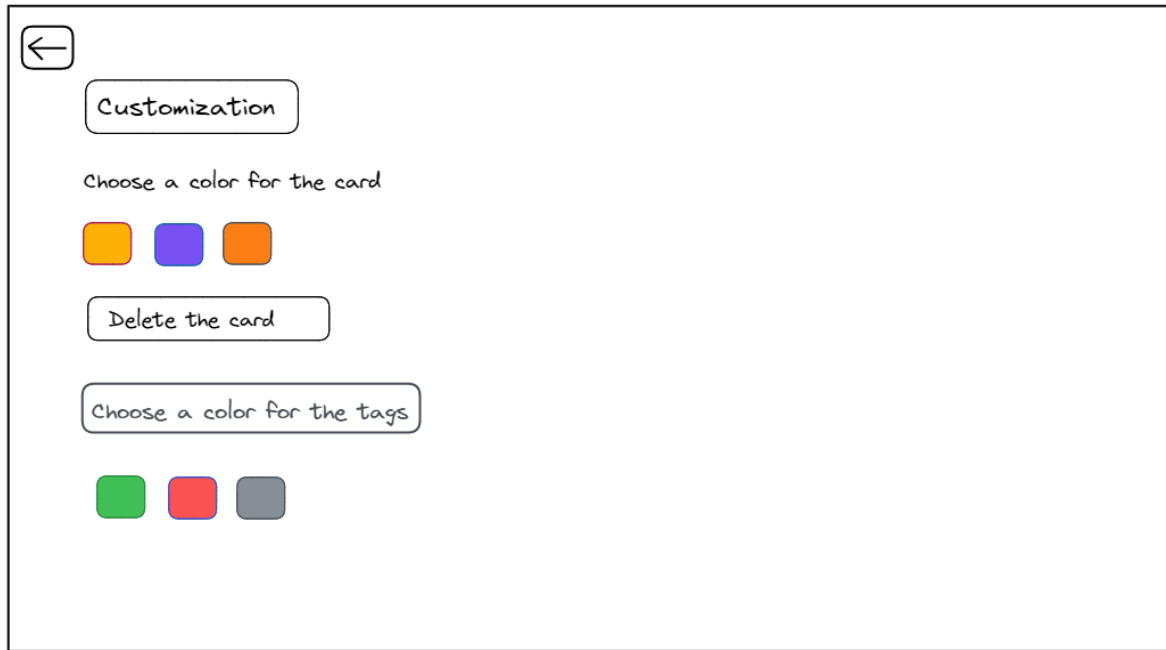
**Figure 4.:** Mock UI dialog windows for a) attempting to edit a password-protected board (top-left), b) inserting an incorrect password for a password-protected board (top-right), c) entering a password for a password protected board (bottom-left), and d) confirming the deletion of a board (bottom-right).



**Figure 5.:** Mock UI window for table customization.



**Figure 6.:** Mock UI window for column customization



**Figure 7.:** Mock UI window for card customization.