Summary:

A Trello frontend clone built using React, styled with Bootstrap and has drag-and-drop features thanks to react-beautiful-dnd.

Deployment:

- The application is deployed on Vercel, URL - https://mytrelloapp.vercel.app/
- GitHub Repo - https://github.com/lookthisisaddy/Trello

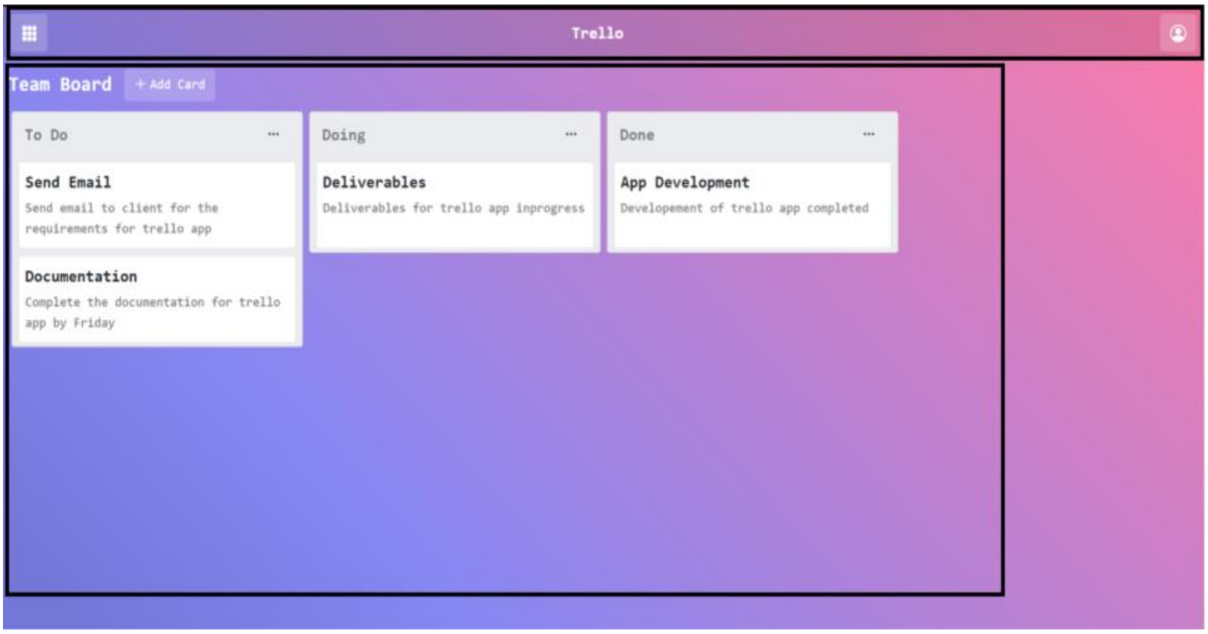
Features:

- The state should be maintained after refresh.
- Title & Description validated appropriately.
- Drag and drop functionality.
- Order of the cards maintained.

Pre-requisites knowledge:

- React
- Bootstrap 5
- react-beautiful-dnd
- HTML & CSS
- JavaScript
- Vercel

Design Approach:



Send Email
Send email to client for the requirements for trello app

Documentation
Complete the documentation for trello app by Friday

Deliverables
Deliverables for trello app inprogress

App Development
Developement of trello app completed

The App mainly comprises of three components: **Navbar, Main** and **Card**

- ***src/components/Navbar.jsx***
  This component returns the <nav> using the class bootstrap 'navbar'. CSS inline styling is used to make it translucent and sticky.

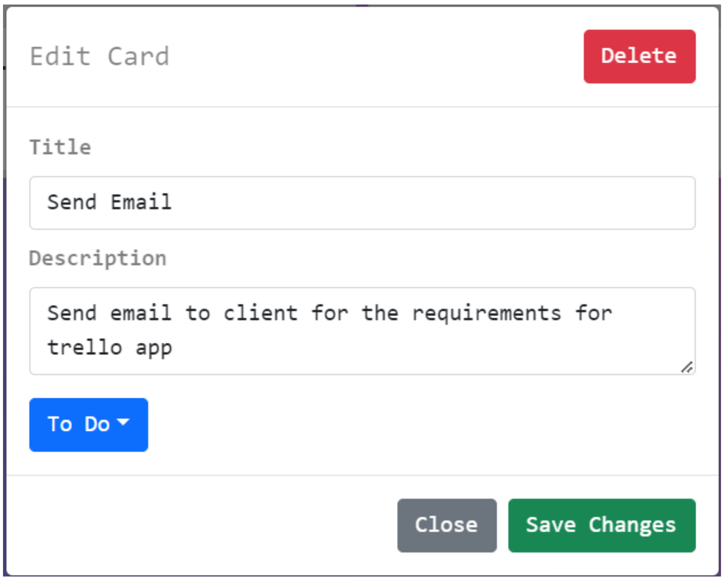
- ***src/components/Main.jsx***

This component divides into two sections: Header and Body, arranged in columns using flex properties. The Header includes a Title and an "Add Card" button, which opens a modal upon clicking.
The body consists of column lists: To-Do, Doing, and Done. Each column list displays cards representing tasks, featuring titles and descriptions. Bootstrap 5 classes, for columns and gutters, ensure alignment and responsiveness for an enhanced user experience.

- ***src/components/Card/Card.jsx***
  This component is a child to Main, which includes a Title and Description button, which opens a modal Add Card Modal upon clicking.

Other Components:

- **Modal**
  This Bootstrap 5 component triggers a modal popup when users click the "Add Card" button or any card item within the column lists. It's structured with a header, body, and footer.
  The header displays the title and a delete button (visible during editing card).
  The body contains a form with inputs for title, a text area for description, and a dropdown for selecting list options.
  The footer includes buttons for closing the modal and saving changes/adding a card.



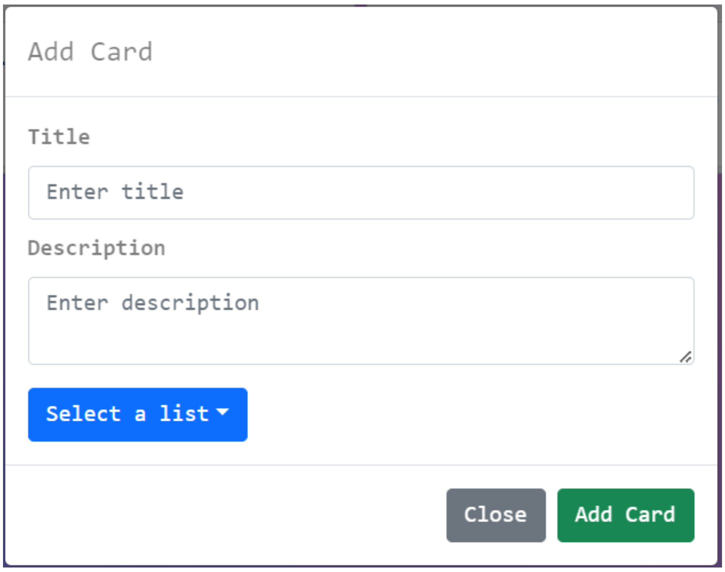
Edit Card

Title

Send Email

Description

Send email to client for the requirements for trello app



- **DragDropContext, Droppable & Draggable**
  These components are part of react-beautiful-dnd library to drag and drop for lists. Wrapping the column lists, the <DragDropContext/> component enables seamless drag-and-drop functionality, allowing users to move items effortlessly between lists. The <Droppable/> component encapsulates the card area within a list, where items can be dropped. Meanwhile, <Draggable/> surrounds each card item, allowing them to be dragged within and between lists.

React Hooks:

- **useState**
- **useContext**
- **useEffect**
- **useRef**

localStorage:

Used to maintain state after refresh.

Handlers:

- **handleModalTitleChange –** This function triggers when input is changed in the modal
- **handleModalDescChange -** This function triggers when textArea is changed in the modal
- **handleModalDropdownSelect -** This function triggers when an option is selected for the drop down in the modal
- **handleCardClick -** This function activates when a card is clicked within a column list, initiating the display of the modal with pre-filled data. Props such as title, description, selectedItem, and cardId are passed to the function.
- **handleModalClose –** This function tiggers when close button is clicked in the modal

- **handleSaveChanges –** This function executes when either the "**Save Changes**" or "**Add Card**" button is clicked within the modal. It incorporates validation logic for each control and handles the process of updating or adding a card to a list.
- **handleAddCard –** Triggers when Add Card button clicked. Resets the modal data and validation errors.
- **handleDeleteCard –** Deletes the card which is currently clicked.
- **onDragEnd –** Function to update lists in the allBoard state variable when a card is dragged from one list to another.

Constants:

- **src/data/Constants.jsx** – JSON data consisting of lists. It is passed to allBoard state variable during the first-time loading.