**Visily Assessment**

**Khang Tran**

* **Task 1**: Building an application similar to Visily, I want to use these technologies:
  + **React.js**: React.js is a widely-used JavaScript library for building user interfaces. It's known for its component-based architecture, which facilitates the creation of interactive and modular UIs. React's virtual DOM also helps improve performance by efficiently updating the user interface.
  + **Redux**: Redux is a state management library that works seamlessly with React. It helps manage the application's state in a predictable way, making it easier to debug and maintain as the application grows. For a complex application like Visily, a robust state management solution is crucial.
  + **Material-UI**: Material-UI is a popular React UI framework that provides pre-designed components based on Google's Material Design. It can help ensure a consistent and visually appealing user interface, saving development time and effort.
  + **Node.js**: Node.js is a server-side JavaScript runtime that allows developers to use JavaScript for back-end development. It has a non-blocking, event-driven architecture, making it well-suited for handling concurrent connections and building scalable applications.
  + **Express.js**: Express.js is a minimal and flexible Node.js web application framework that simplifies the process of building robust and scalable APIs. It is lightweight and allows for easy configuration, making it an excellent choice for the back-end of a web application.
  + **MongoDB**: MongoDB is a NoSQL database that stores data in a flexible, JSON-like format. Given the dynamic nature of wireframes and prototypes, a document-oriented database like MongoDB is suitable. It allows for quick iterations and schema flexibility, which can be beneficial during the development phase.
  + **Mongoose**: Mongoose is an Object Data Modeling (ODM) library for MongoDB and Node.js. It provides a straightforward way to interact with MongoDB by defining schemas and models, making it easier to work with data in a structured manner.
  + **WebSocket (e.g., Socket.io)**: For real-time collaboration features, consider using WebSocket technology. Socket.io is a popular library that enables real-time bidirectional communication between clients and the server, making it suitable for collaborative functionalities in wireframing applications.  
      
    These technology choices provide a well-rounded and scalable stack for building a web application similar to Visily. It's important to adapt these recommendations based on the specific requirements and constraints of the project.
* **Task 2**: Visily has done an excellent job with their UI products, and it's challenging for candidates to redesign it better on your own. However, if I had to choose, I would choose to redesign the Product Card component as below:  
  A screenshot of a product page

  Description automatically generated
  + **Reasons:**
    - The layout is not reasonable, the location of each item is not clear
    - Missing necessary item (Ex: cart button, price,…)
    - Not enough to attract viewers
    - Lacks minor effects but provides good visual effects (Ex: shadow, 3D effect,…)
  + **Improvements:**
    - Align information clearly (Ex: left margin, right margin)
    - Align the content size just right, not too big or too small
    - Add all necessary related item
    - Add some small effects but create a good look (shadow, 3D effect,…)  
      A screenshot of a product

      Description automatically generated
* **Task 3**:
  + I have attached all my code in the compressed file (zip)
  + Please open **build/index.html** to see my implementation.