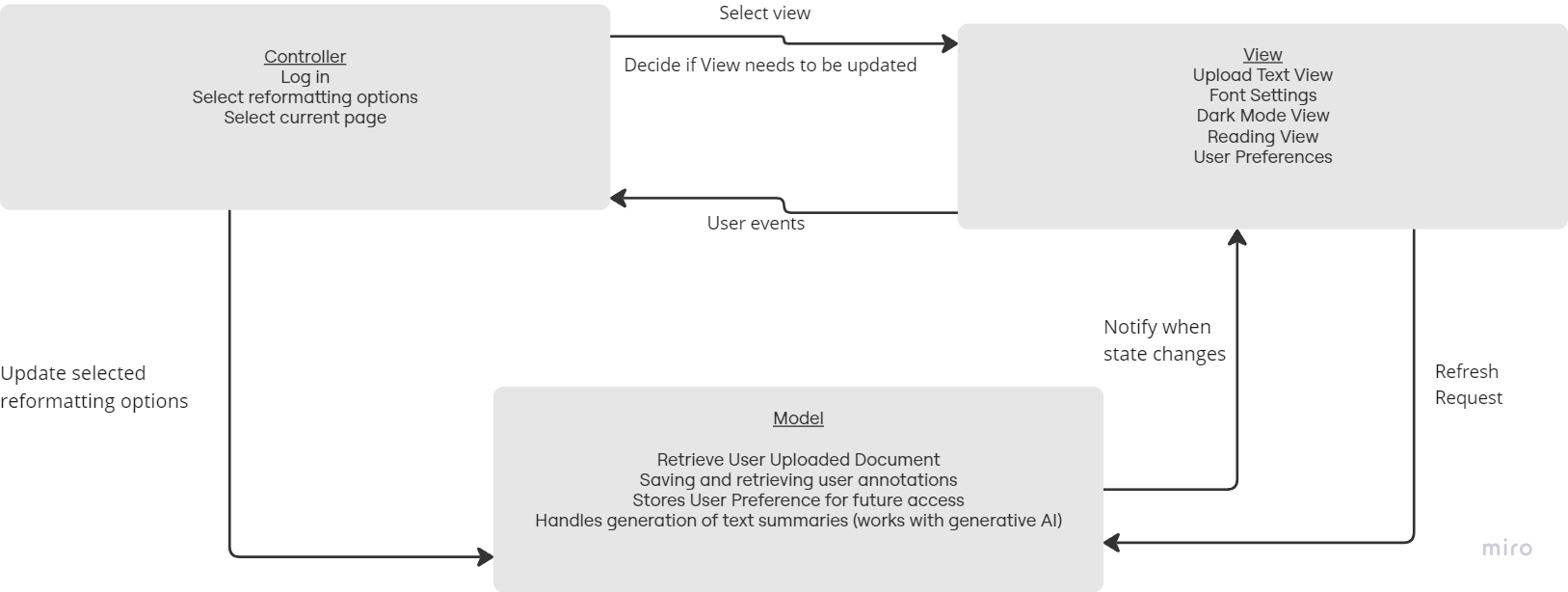
**Team: C**

**Architecture Design:**

For designing the architecture of ComfortRead we designed a Model-View-Controller architecture. This model was chosen because of the needs of scaling ComfortRead’s diverse features such as annotation, text customization, and reading features. It is important that the responsibilities are separated by the data management (Model), the user interface (View) and interactions (Controller), so that we can maintain independence of each component, making scalability and testing easier.

**Architecture Design Diagram**:



**Design Description:**

The architectural pattern consists of three components, the **view**, the **controller** and the **model**.

**View**

The View in ComfortRed is responsible for handling the user interface, which in return allows the user to interact with the applications text reformatting options. Views will change depending on the functional user choice such as “Text-to-Speech”, “Bionic Reading”, or “Focus Mode”. Whenever the interface is interacted with (ie. Document Upload) the View will then communicate it to the Controller. React.js is used for the front end development.

**Controller**

The Controller’s responsibility is that of managing user inputs and the interactions from the view. In our app ComfortRead, it is the controller’s duty to process actions such as generating text summaries to add to View or enabling “Dark Mode”. Once an input is received (ie. confirming option selection) the controller works with the model and view to make sure that the data or changes to the user interface are handled. We will use Firebase to implement login / logout functionality.

**Model**

The model for ComfortRead handles the data from the app, including documents, user preferences and annotations. This category makes sure that data is stored and retrieved efficiently when a user interacts with the app. When a user uploads a document the model will handle storing the document for future use such as format customization. The model communicates with the controller to update the view of any changes in data (ie.reformatting document). We will use Firebase to store users’ saved texts.