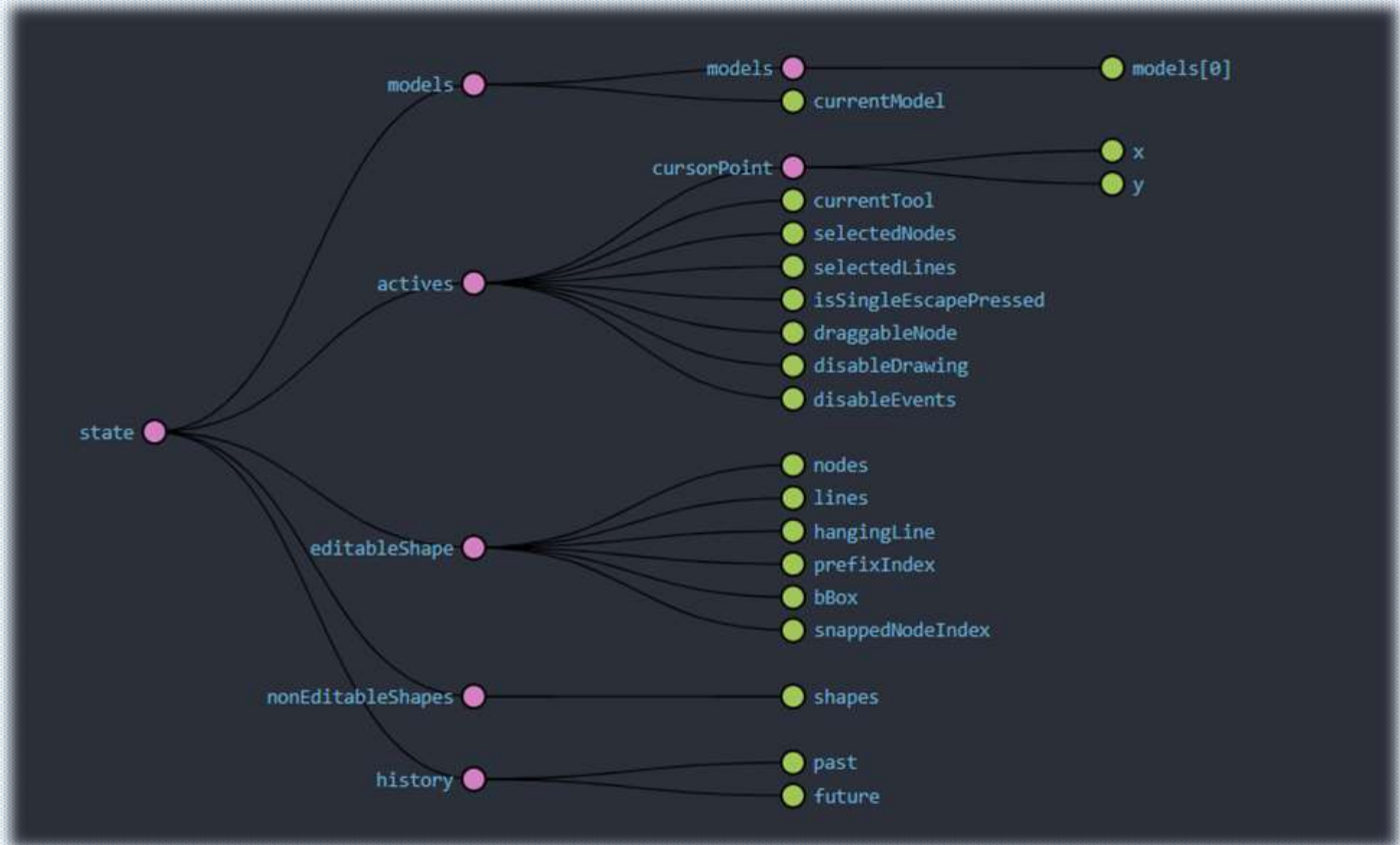
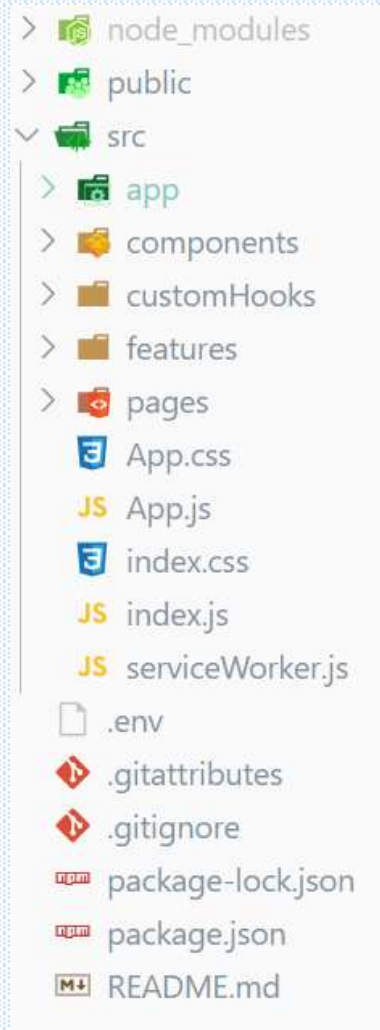


Store Structure:



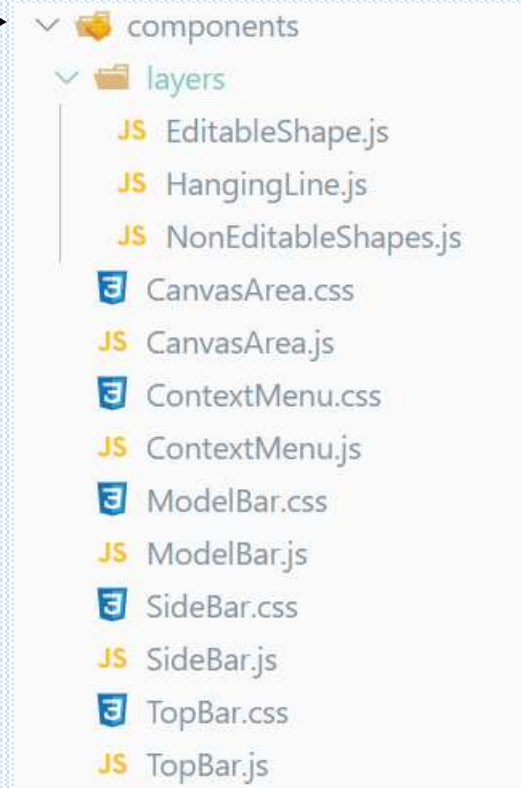
Folder Structure:



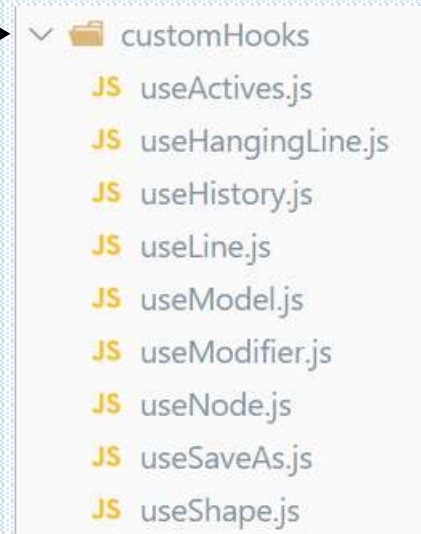
Store



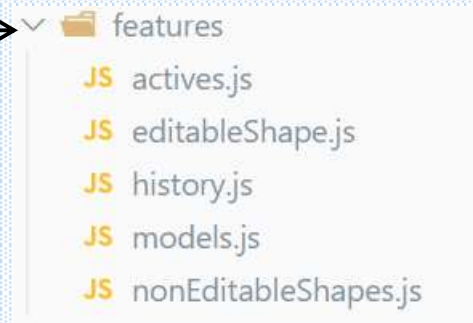
**Components &
Konva Layers**



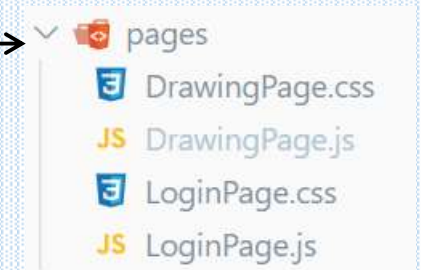
React Custom Hooks



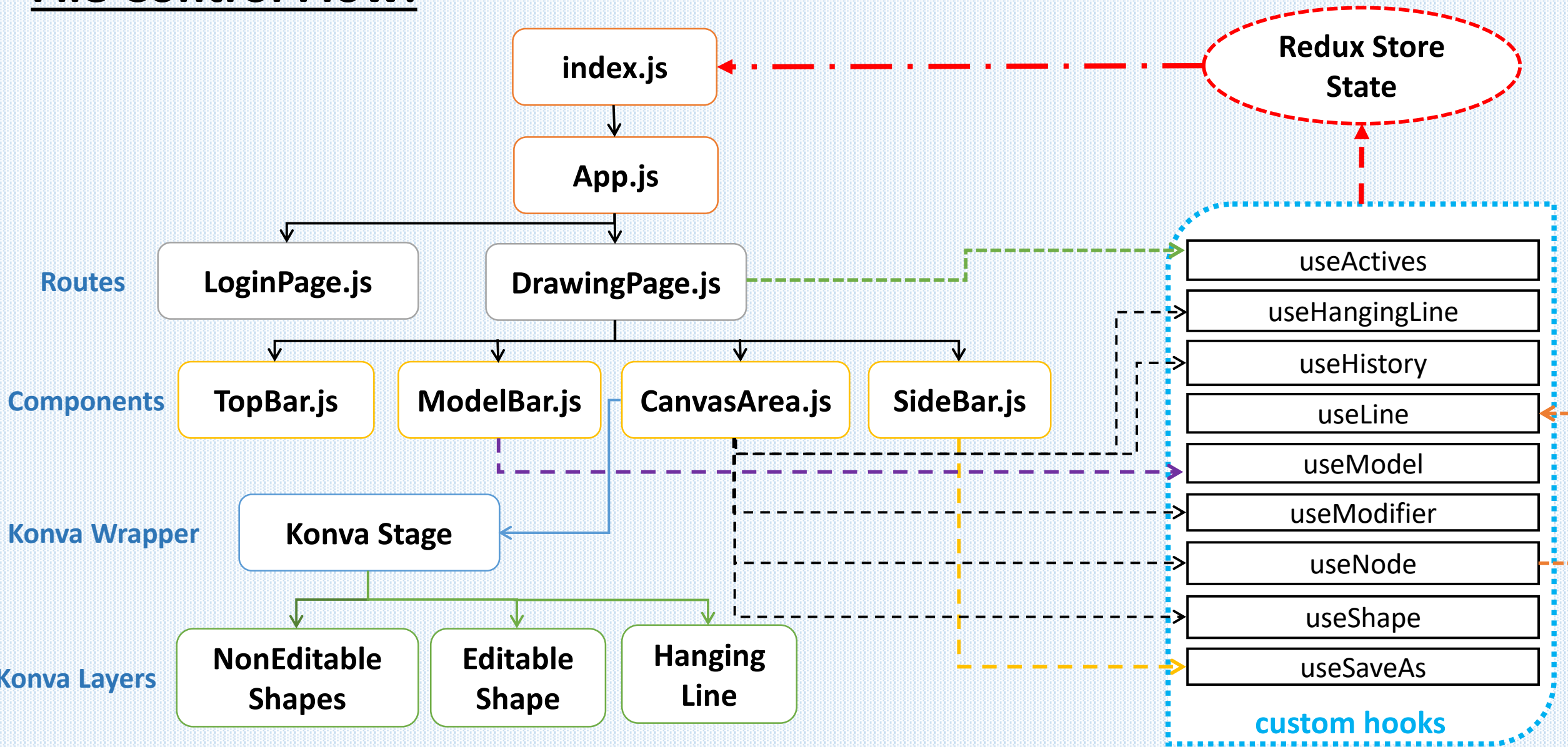
Redux-toolkit slices



Pages (Routes)



File Control Flow:



Files & their uses :

App.js --> Used for **Routing**.

LoginPage.js --> Used for **Logging** an user in.

DrawingPage.js --> Contains **TopBar**, **ModelBar**, **CanvasArea** and **Sidebar** as it's components. This file **also** listens to alphabets click on keyboard and changes tools based on **P** or **V** key is pressed. This file communicates with **useActives** custom hook to do these tasks.

TopBar.js --> This will contain name of our application and also a **Search bar**.

ModelBar.js --> This will have our model names and functionality to **add**, **remove** and **rename** our models. This will also **load model data** from the models slice (reducer) and populates it into other slices(reducers). This file communicates with **useModel** custom hook to do these tasks.

SideBar.js --> This file shows us on which mode are we in. This will also have buttons to **reset** everything in the current model, to **save** drawings present in the current model in **PDF** / **PNG** / **JPEG** Format. This file communicates with **useHistory**, **useActives** and **useSaveAs** custom hooks to do these tasks.

CanvasArea.js --> This file listens to combined events like **ctrl + z** / **ctrl + y**, to key presses like **Escape**, **Delete**, **Shift**, **Control** and **Mouse Clicks**. This file uses **useNode**, **useHangingLine**, **useHistory**, **useShape** and **useModifier** custom hooks to do the tasks like node, line, hangingLine creation, node dragging, node deletion and converting line to curve or curve to line.

Custom hooks & their uses :

useActives: This hook contains **changeTool** function. This is used to **change** our **tool**.

useHangingLine: This hook contains **createHangingLine** and **updateHangingLineData** functions. This is used to **create hanging lines** and **update its co-ordinates** during undo, redo actions.

useHistory: This hook contains **undo**, **redo**, **reset** and **updateHistory** functions. This is used to do **undo / redo** task, to **reset** the current model and to create a copy of old data when an event is called, this helps in undo / redo.

useLine: This hook contains **createLine** and **updateLine** functions. It is used to **create a line** and also to **update lines** data when an individual line is dragged.

useModel: This hook contains **updateModelData**, **updateModelDataOnAdd** and **fetchModelData** functions. These are used to **create** new **models**, add current canvas data to old model and fetch new data from selected model and update that onto current canvas.

useModifier: This hook contains **lineToCurve**, **curveToLine**, **updateControlNode** and **changeToEditableShape** functions. These are used to **convert a line into a curve or curve into a line**, **update curve** on dragging control nodes and **changing** a non editable shape back to editable shape when **double click** is made on the shape.

useNode: This hook contains **createNode**, **updateNode** and **deleteNode** functions. These are used to **create** a node, **delete** a node and **update** a node's co-ordinates when dragged.

useSaveAs: This hook contains **savePdf**, **savePng** and **saveJPG** functions. These are used to **save** drawing onto our system in PDF / PNG/ JPEG format.

useShape: This hook contains **createPath**, **createBBox** and **addToShapes** functions. These are used to **create a ghost path** on snapped shape, **create a bBox** on editable shape and to **transfer** snapped shape data from editableShape to nonEditableShapes.