Code structure

This is about code structure in the project.

## dist

This folder is where the optimized and compiled production-ready code is placed after you build your application. It contains the minimized HTML, CSS, and JavaScript files that are ready for deployment to a web server. This folder is usually generated by build commands like npm run build.

## docs

This folder contains docs about the project

## e2e

Used for unit testing and end-to-end testing.

## Inputs

??

## node\_modules

This directory is automatically generated when you install project dependencies (using npm install). It contains all the libraries and packages your project relies on. You typically do not commit this folder to version control.

## public

This folder holds static assets that are directly copied to the build output directory without being processed by Webpack or Vite. Like images, fonts etc

## src

This folder contains source code of the application.

### assets

### auth

I think we should move this to components.

### components

These are the frontend components used in the application like jointjs, Babylon, vueflow, etc.

#### auth

### composables

functionality or themes which are applicable to complete application.

### design

Global layout, styles, themes, design token, branding assets.

### Layouts

Reusable layouts structures like Default layout of pages, headers, footers, navigation bar, sidebar etc.

### pages

pages of application or route specific components. Like homepage.vue etc

### router

route definition and all router level details, configuration etc.

### simulator

### stores

### utils

### views