Product Design

Team: 17

Katha Rohan Reddy, Venneti Sri Satya Vinay, Shubhankar Kamthankar, Rajarshi Ray

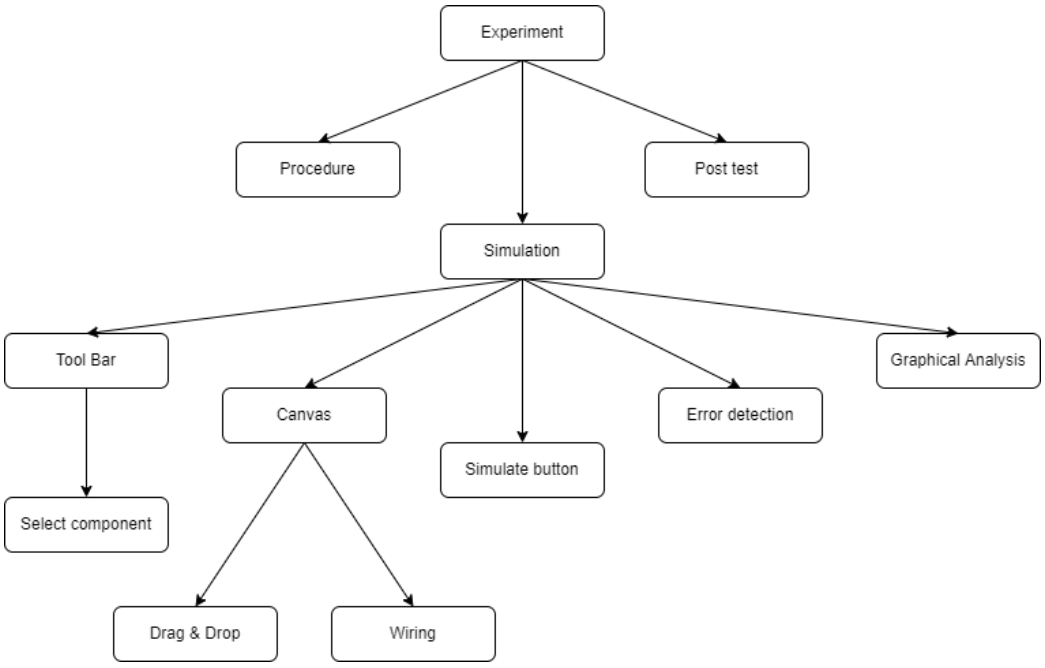
Design Overview

Architectural design

Our project is to recreate and improve the older simulations (made using Java which has now become obsolete) on the virtual lab's website using latest web technologies like HTML, CSS and JavaScript. The three modules for this project are

- Procedure
- Simulation
- Post test

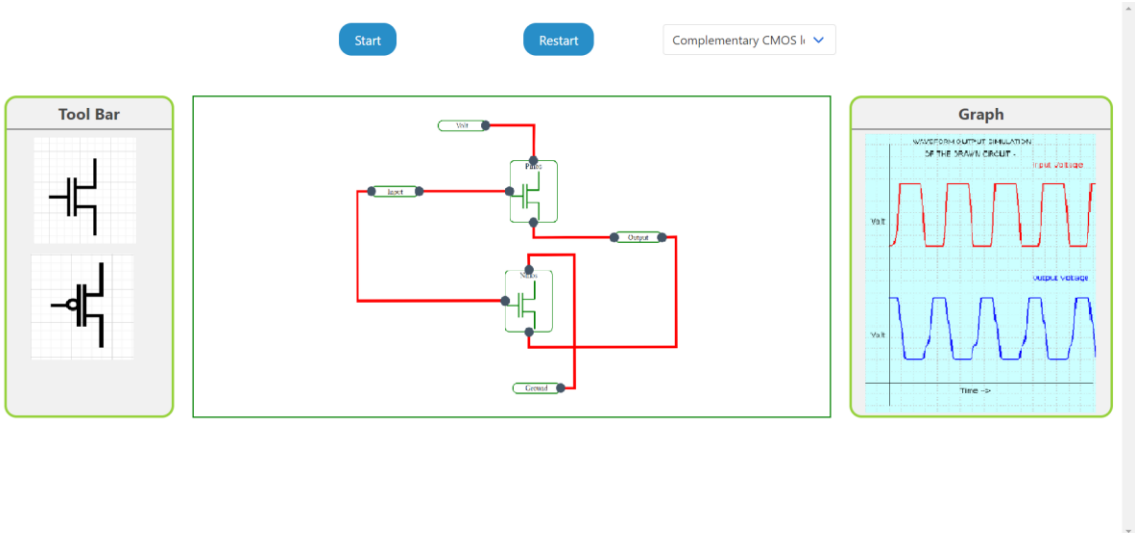
Based on the Simulation we make we need to change the procedure. The post test is a set of questions based on the experiment.



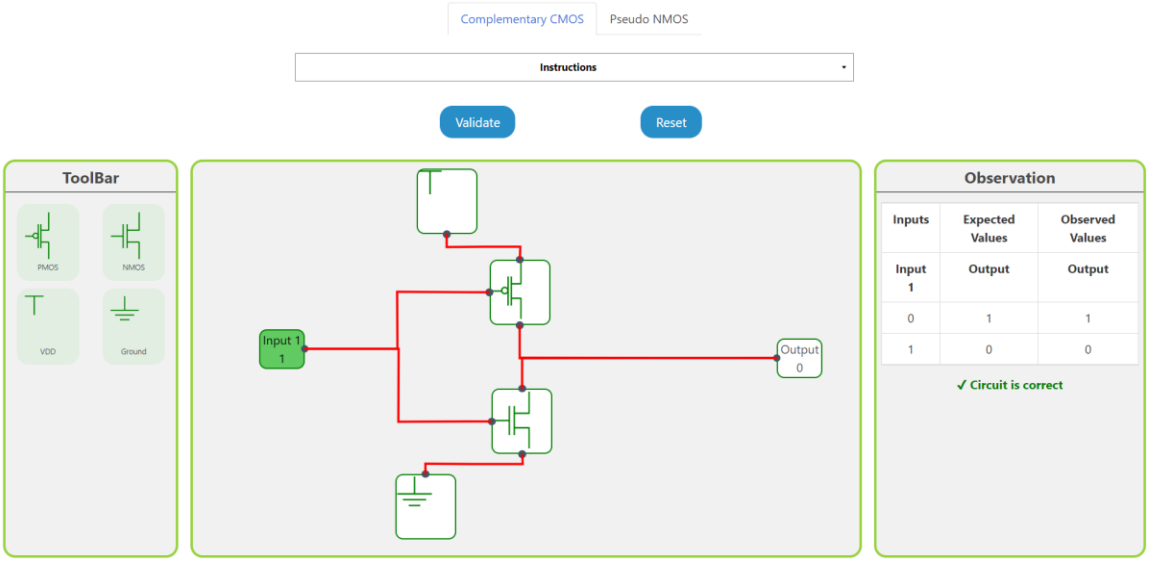
System interfaces

User Interface

In the simulation part the user can select components from the tool bar and place them in the simulation workspace. After dragging and dropping all the components and connecting to complete the circuit. The user clicks on simulate button and then sees the graph or truth table of the output. In the procedure section he just sees the procedure, in the post test area the user gives a test based on the concept he learned. In mobile view graph goes below tool bar and workspace. The old UI looked like this.



The updated UI looks like this.

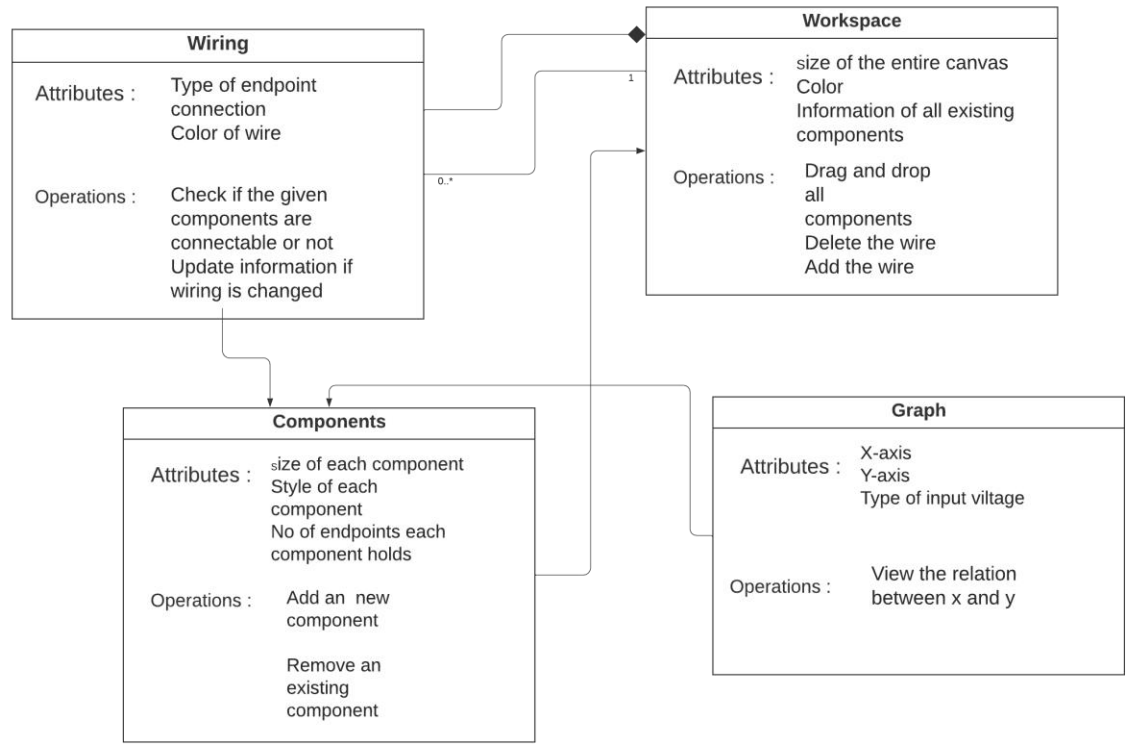


Model

jsPlumb library is used.

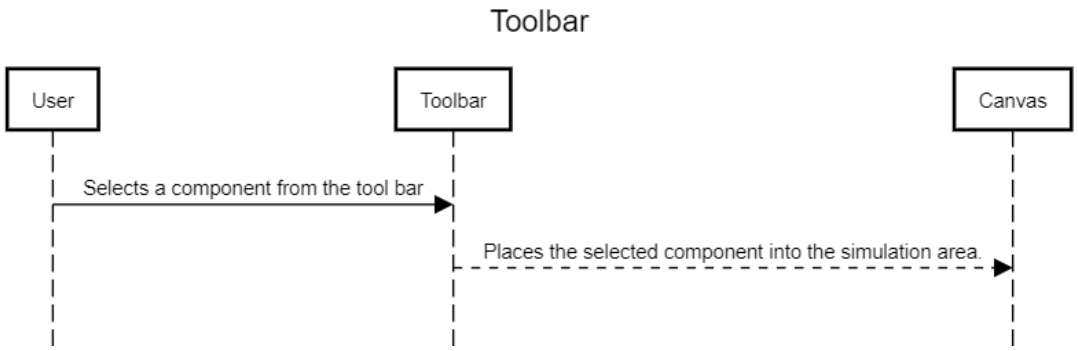
Components	Class state • Size of each component • Style of each component • No of endpoints each component holds Class behavior • Add new component • Remove the existing component.
Canvas	Class state • Size of the entire canvas • Color • Information of all existing components Class behavior • Drag and drop all components • Delete the wire • Add the wire
Wiring	Class state • Type of endpoint connection • Color of wire Class behavior • Check if the given components are connectable or not • Update information if wiring is changed

Graphical Analysis	Class state • X-axis • Y-axis • Input Voltage Class behavior • View the relation between X and Y and displays graph Update information if wiring is changed

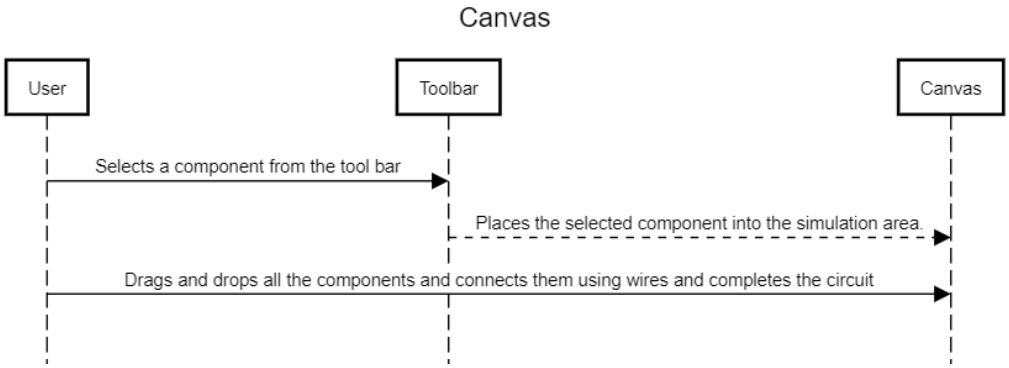


Sequence Diagram(s)

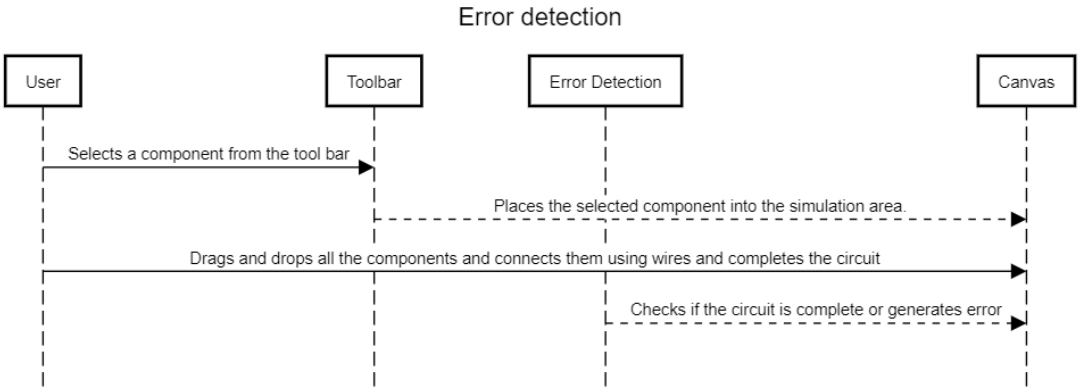
Tool Bar:



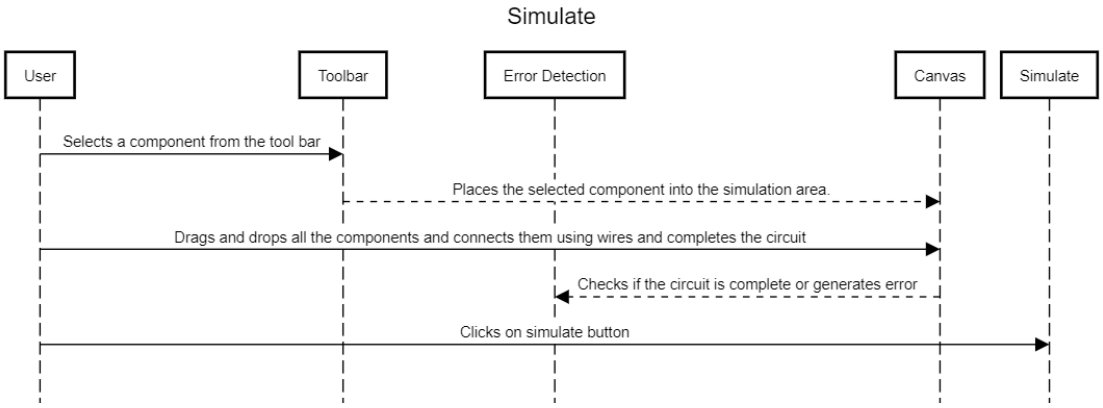
Canvas:



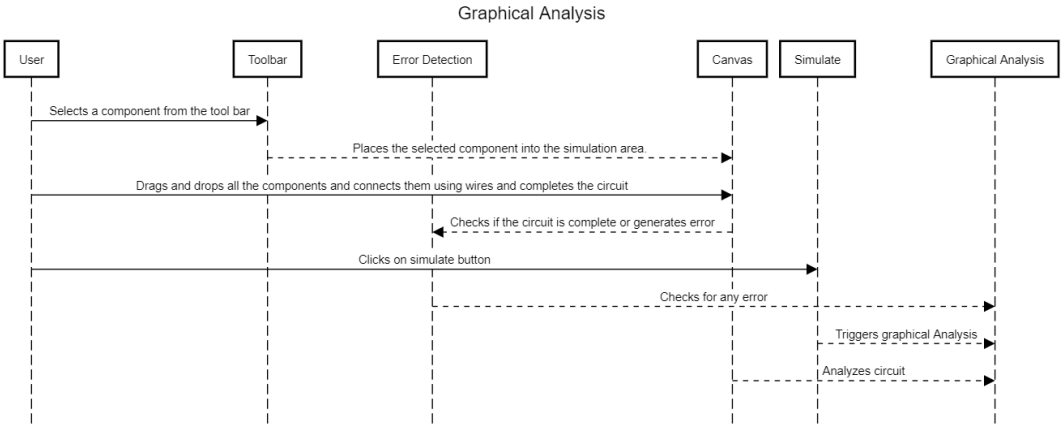
Error Detection:



Simulate:



Graphical Analysis:



Design Rationale

We here are mentioning two design rationale even though both are close enough they serve entirely different purpose this is for the entire output after the user makes circuit an clicks simulate button

In this case we had two different scenarios when the users circuit is not right

Design rationale 1

If User's simulation is not correct, we can show error message and not render the graph in the graph section and guide them to make the correct solution by giving hints about how to make the circuit properly and render

Design rationale 2

If the user simulation is not right even, then we render the graph although not correct one and show him what the right graph looks like and motivate him to do the right simulation.

Initially, we chose to go with design rationale 1. The following is the reason for it.

In Order to do the second design we have to know the number of inputs and outputs of the entire circuit at the start of the experiment which is not the purpose (as the experiment output is graph to draw it we need to know the parameters initially)

But after discussing with the client we decided to make experiments 1, 2, 3 using design rationale 2 but use design rationale 1 for all the remaining experiments.