Software Requirements Specification (SRS) Document

Date: 15 Feb 2022

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

Brief problem statement

The virtual labs website is an experiment-based platform for interactive learning. It has many courses in different disciplines. But since the website was created a long time ago at that time the technologies that were used became obsolete and hence, we need to recreate and improve the simulations in every experiment of the course given to us using the latest web technologies and libraries like HTML, CSS, JS etc. and improve portability and usability. For our project we are given the course VLSI.

System requirements

The designed product should work on any latest and popular browsers like Chrome, Firefox, Edge, Opera, Safari etc.

The technologies required for the project are

1. HTML
2. CSS
3. JavaScript
4. JavaScript libraries like anime.js, JointJS, p5.js and Paper.js

Non-functional requirements

- The simulation should be responsive
- The page load time should be less than 1.5 seconds over a fast 3G network
- The maximum load time should be less than 3 seconds
- The size of the page should be less than 3MB

User's profile

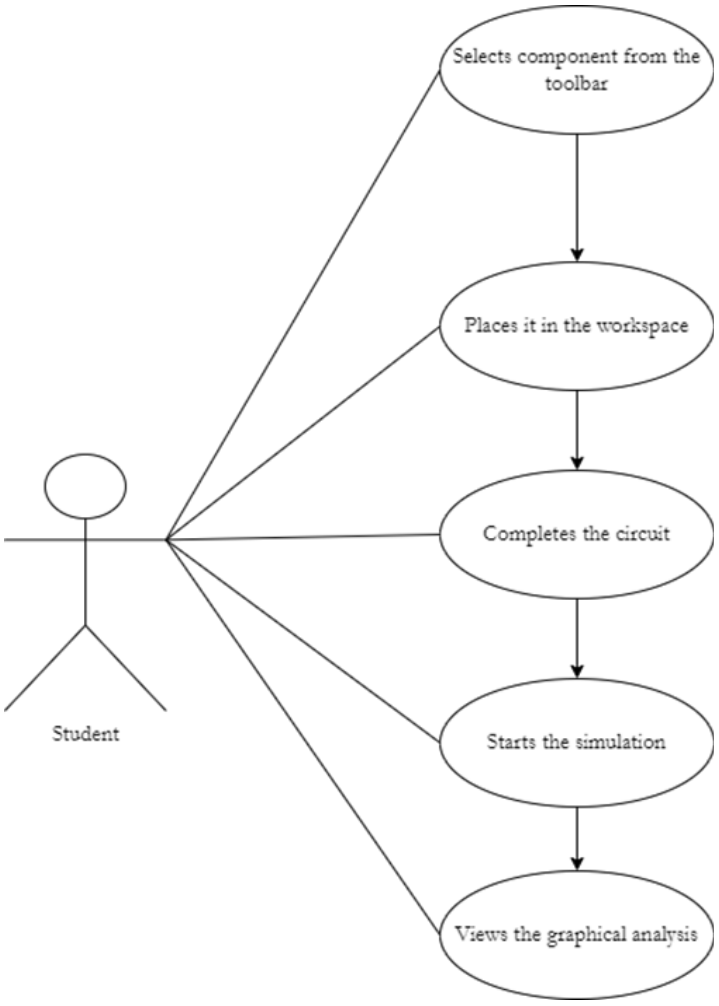
1. Students who want to learn VLSI. We assume that the students are familiar with basic functions and operations of a web browser and have a minimum computer knowledge which helps them to do the simulations and learn the concepts.

Feature requirements (described using use cases)

No.	User Case Name	Description	Release
1.	Toolbar	The simulation page contains a tool bar of components from which the user can drag and drop components to the workspace.	R1
2.	Error Detection	If the user clicks on a component which is not necessary for that experiment, we need to alert him/her	R1
3.	Workspace	It is in the workspace where the user does all the simulation parts and learns the concept	R1
4.	Simulate button	It is a button and on clicking it the simulation starts and the graph gets plotted.	R1
5.	Graphical analysis	It is the place where we show the simulation's analysis graphically.	R1
6.	Change in Experiment content	Change of experiment content according to simulation	R2
7.	Experiments - 1,2,3,4	User shall make NOT, AND, OR, XOR, XNOR, multiplexer using PMOS, NMOS and other components	R1

8.	Experiment - 5	User shall make a chain of inverters and delay is estimated	R1
9.	Experiment - 6	User shall make D Flip-flop, D-Latch using PMOS, NMOS and other components	R2
10.	Experiment - 7	User needs to use Spice code platform	R2
11.	Experiment - 8, 9	User shall design digital circuits, D-Flip Flop using verilog	R2

Use case diagram



Use case description

Use Case Number:	UC-01
Use Case Name:	Toolbar
Overview:	The simulation page contains a tool bar of components from which the user can drag and drop components to the workspace
Actors:	Students
Pre-condition:	None
Flow:	Main (success) Flow: User selects a component from toolbar Then the system makes that component ready to be used in the simulation workspace
Post Condition:	The next step is to select a component from the tool and check if there is an error.

Use Case Number:	UC-02
Use Case Name:	Error detection
Overview:	If the user selects a component which is not required for the experiment, we need to alert him/her.

Actors:	Students
Pre-condition:	Select a component from the ToolBar to place it in the workspace.
Flow:	Main (success) Flow: When the user does any task which is not necessary like selecting more components than required or using a component which is not required or the circuit is incomplete an error will be generated
Post Condition:	If there is no error, then place the selected component in the workspace

Use Case Number:	UC-03
Use Case Name:	Workspace
Overview:	It is in the workspace where the user does all the simulation parts and learns the concept.
Actors:	Students
Pre-condition:	Select a component from the ToolBar to place it in the workspace.
Flow:	Main (success) Flow: The user drags and drops all the components and connect them by a wire and complete the circuit

Post Condition:	The next step is to select all the remaining components, place and connect them properly to connect the circuit

Use Case Number:	UC-04
Use Case Name:	Simulate button
Overview:	It is a button and on clicking it the simulation starts and the graph gets plotted.
Actors:	Students
Pre-condition:	Complete the circuit in the workspace by connecting all the components
Flow:	Main (success) Flow: When user clicks the button and the circuit made is correct then graph will be displayed and animation takes place
Post Condition:	The next step is to view the generated graphical analysis.

Use Case Number:	UC-05
Use Case Name:	Graphical analysis
Overview:	It is the place where we show the simulation's analysis graphically.

Actors:	Students
Pre-condition:	Click on the simulate button to generate a graph of the simulation.
Flow:	Main (success) Flow: When user clicks the simulation button graph will be displayed
Post Condition:	Change the experiment condition based on simulation.

Use Case Number:	UC-06
Use Case Name:	Change in Experiment content
Overview:	Change of experiment content according to simulation
Actors:	Students
Pre-condition:	None
Flow:	Main (success) Flow: Change of experiment content according to simulation
Post Condition:	None

Use Case Number:	UC-07
Use Case Name:	Experiments - 1,2,3,4
Overview:	User shall make NOT, AND, OR, XOR, XNOR, multiplexer using PMOS, NMOS and other components
Actors:	Students
Pre-condition:	None
Flow:	Main (success) Flow: Users will take components from toolbar and then he will Drag them to simulation area and connect them with components and simulate the circuit
Post Condition:	User views the graphical analysis of the simulation

Use Case Number:	UC-08
Use Case Name:	Experiments - 5
Overview:	A chain of inverters will be made and delay is estimated
Actors:	Students
Pre-condition:	None

Flow:	Main (success) Flow: User first selects components from the toolbar, drags them to the simulation area and makes a series of NOT gates and after simulation the software prepares a graph.
Post Condition:	User views the graphical analysis of the simulation

Use Case Number:	UC-09
Use Case Name:	Experiments - 6
Overview:	User shall make D Flip-flop, D-Latch using PMOS, NMOS and other components
Actors:	Students
Pre-condition:	None
Flow:	Main (success) Flow: Users will take components from toolbar and then he will Drag them to simulation area and connect them with components and simulate the circuit
Post Condition:	User views the graphical analysis of the simulation

Use Case Number:	UC-10
Use Case Name:	Experiments - 7
Overview:	User needs to use Spice code platform
Actors:	Students
Pre-condition:	None
Flow:	Main (success) Flow: Users will take components from toolbar and then he will Drag them to simulation area and connect them with components and simulate the circuit.
Post Condition:	User views the graphical analysis of the simulation

Use Case Number:	UC-11
Use Case Name:	Experiments - 8, 9
Overview:	Design of digital circuits, D- Flip Flop using verilog
Actors:	Students
Pre-condition:	None

Flow:	Main (success) Flow: Users will take components from toolbar and then he will Drag them to simulation area and connect them with components and simulate the circuit using verilog
Post Condition:	User views the graphical analysis of the simulation