GSoC 2025 Proposal - CircuitVerse

Project Name: Assignment Suite Enhancement
Duration: 175 hours
Difficulty: Easy
Technologies: Ruby on Rails, JavaScript,TypeScript,Canvas API, VueJS
Mentors: Yashika Jotwani, Aman Asrani, Siddharth Asthana, Philip Abbey(self-mentor)

The project aims to enhance CircuitVerse's educational tools by improving the classroom and assignment management features, making it a more effective platform for both teachers and students. It involves developing a multi-level classroom structure based on topics, lessons, projects and achievements, allowing students to form collaborative subgroups for collaborative projects, creating a dashboard for the techers to manage the classroom and different subgroups, and introducing a flexible assignment system for both individual and group submissions. The classroom structure will take inspiration from google classroom with additional features integrating CircuitVerse. Additionally, new features such as pre-built circuit submissions with integrated test cases and auto-verification from practice sessions will be added to streamline the learning process. Integration with Canvas LMS will be improved to strengthen CircuitVerse's utility in academic settings, supporting educators and students with user-friendly tools, ensuring seamless workflow between CircuitVerse and existing educational infrastructures. UI/UX of classroom and assignment section will be improved. These enhancements will make CircuitVerse a powerful, user-friendly tool for digital circuit design to be used by students and teachers.

Deliverables:

1. Multi-level classroom structure like project groups, homework teams and test groups.
2. Subgroup feature for combined projects and tests
3. Pre-Built Circuit Submissions with Integrated Test Cases using Testbench
4. Canvas LMS Integration Enhancements
5. Dashboard for teachers to manage classrooms and subgroups
6. Role-based classroom and subgroups like creators, evaluators, members, etc

Personal Details:

Name: Vivek Kumar
Email: vk092kumar@gmail.com
GitHub: https://github.com/092vk
LinkedIn: https://www.linkedin.com/in/vk092/
Phone: 9596133638
Country: India

Portfolio Website: Portfolio Website
Resume/CV: Resume

About Yourself:

1. Please describe yourself, including your development background and specific expertise.

My name is Vivek Kumar Ray, I am a 3rd-year B.Tech student in the branch of Electronics and Communication. I have extensive knowledge of HDLs like Verilog, experience with Machine Learning, Web development, software development, and electronics. I have worked on several EDA tools similar to CircuitVerse and have contributed to open-source EDA tools and software in the past. I have contributed to eSim which is an EDA desktop-based analog circuit simulator. I have done research work infields of Sensor Fusion, Battery life prediction, Software-defined radio, and other fields that combine the knowledge of both software and hardware. I have also worked with IOT, Cloud, and Desktop applications using PYQT and done software packaging for various LINUX distributions and versions. I have knowledge of a range of tools and technologies which include JavaScript, TypeScript, Node, Ruby on Rails, Vue, React, CSS, Bootstrap, Tailwind, HTML, Python, C, C++, AWS, SQL, MongoDB, Postgres, Redis, other software development languages, libraries, framework, and tools.

2. Why are you interested in the CircuitVerse project(s) you stated above?

I have been using CircuitVerse for the past 3 years for digital circuit design and have learned digital design using it, which inspired me to choose CircuitVerse as my GSOC project. Since being an electronics student who also has interests in Software development, CircuitVerse is the go to choice for me since it combines the best of both worlds. Together with my experience in **Digital circuit design**, Circuitverse becomes a perfect choice for me that aligns with my skill sets and passion, hence the choice. I have contributed to CircuitVerse extensively, which also makes it an obvious choice.

3. Have you participated in an open-source project before?

Yes, I have worked on open-source tools before namely, I have worked on eSim which is an EDA tool similar to CircuitVerse but eSim focuses on analog circuits while CV deals with digital circuits. My work at eSim involved:

1. Removing bugs from the eSim 2.4 Windows installer.
2. Updating the eSim version.
3. Building a toolchain for eSim to manage its resources.
4. Enabling dual plotting using Matplotlib and NgSpice
5. Repackaging of eSim for LINUX using FlatPack.

I have also participated in Open-Source hackathons like Hacktoberfest and Google Developers Club and contributed extensively to them.

Commitment

1. No, I am not planning any vacations during the GSoC 2025 period (June 2 - September 8, 2025, or June 2 - November 17, 2025, for extended projects).
2. I will be taking one remote class of ML during the GSOC period.
3. No, I don't have any employment during the GSOC period.
4. I am applying for a small (175 hours) project size.
5. I am planning to work 16-20 hours per week on the project. I am flexible with my work hours but I prefer to work from 9:00 am to 2:00 pm.
6. Currently, I don't think I need an extended timeline.

Contributions So Far

- PRs merged
    - Removed the conf files for GitPod and its refrence
    - Fixed the Assignment UI
    - Fixed the Testbench
    - Fixed security vulnerability
    - Fixed the timming diagram
    - Removed third-party services duplicate information
    - Added testbench description in feature section
    - Added Docs for Verilog
- PRs unmerged
    - Improved Teachers section and added a new UI teachers_component
    - Added warning for special condition in FlipFlops
    - Changed the SETUP.md description to setup Yosys serer in local
    - Fixed the Language Filter
    - Replaced offset-based pagination with cursor based pagination
- Issues and bugs found
    - Verilog Testbench
    - Verilog module undefined for JK FF, SR FF, T-latch etc
    - Flip Flop invalid conditions not handled
    - Testbench object not detected issue
    - Timing Diagram overlapping
    - Mobile view for testbench
    - Inconsistent style for testbench window
- Documentation contributions
    - Added Docs for Verilog

- Other contributions to CircuitVerse
    - Updated Wiki notes regarding GitPod and removed its reference
    - Updated Wiki notes for Verilog
- Prototypes Built
    - Real-time communication and shared editing
    - Finite State Machine Simulator
    - Comparator Circuit Element

Proposal

Overview: Proposed Solutions

Multi-Level Classroom Structure

- Allow the creation of classrooms with sections and subgroups.
- This facilitates team-based learning by enabling student collaboration in groups and compete with one another..

Subgroups for Collaborative Projects

- Students can form project groups within classrooms.
- Multiple students of the same group can work on same project and do combined submission..

Flexible Assignment Management System

- Supports individual and group submissions.
- Teachers can set deadlines(already implemented), track progress, and review assignments.

Pre-built circuit Submissions with Integrated Test Cases

- Allows submission of circuits with integrated test cases using the testbench.
- Automated test by teachers for quick evaluation.

Improved Assignment Submission Workflow

- Simplifies submission and grading process.
- Enables bulk submissions, grading, and feedback.

Canvas LMS enhancements

- Synchronizes assignments and grades with the Canvas Learning Management System.
- Allow the teachers to use LMS to grade students and import data about the classroom.

Detailed Description

Why ?

CircuitVerse is a powerful platform for learning digital circuit design, but to maximize its impact in academic settings, it needs enhanced classroom and assignment management capabilities. Currently, instructors face challenges in organizing students, assigning collaborative work, and automating assessments, which limits the platform’s effectiveness in structured courses.

By introducing a multi-level classroom structure and collaborative subgroups, students can engage in team-based learning, similar to real-world engineering projects. Subgroups will increase competitiveness among students and will allow them to work on groups against other groups on assignments. A flexible assignment system with pre-built circuits and test cases will help educators create structured assessments, reducing manual effort in grading. Auto-verification from practice sessions ensures students receive instant feedback, improving learning efficiency.

Additionally, integrating with Canvas LMS will help teachers with assignments and grades, making CircuitVerse more accessible in university and academic settings. These improvements will transform CircuitVerse into a user-friendly educational tool, bridging the gap between theory and hands-on circuit design practice. Helping both students and teachers in learning and teaching digital circuit design easily.

Implementation plan:

1. Multi-level Classroom structure

Currently, circuitverse allows its users to form groups, but doesn't provide a multi-level class room structure. A multi level classroom structure can help students compete with one-another for posts. These hierarchies will be based on performance, group roles, departments, clubs and intrests in the same classroom.

1. Schema changes will be done in Databse for realisation of multi-level classroom structure.
2. Relationship will be defined between new tables and queries and routes defined
2. UI/UX changes will be done to navigate the classroom structure and use the feature this will be done in both new and old code base **(src/components , app/views folders)**
3. Several roles will be defined based on hierarchy with the creator of classroom having the privilege to assign the roles
4. Badges will be made to assign to top performers.
5. Models and Controllers will be changed in legacy-codebase to reflect the classroom feature.
6. Current group based structure will be replaced by classroom based structure with groups forming parts of classroom.
7.Other things like assignments will be similar to current group section.

2.Subgroups for Collaborative Projects

1. Inside classroom subgroups will be created like current groups section.
2. Roles structure in the classroom will be kept same from the current group structure.
3. Roles such as creator, teachers , members will be enhanced.
4. UI/UX will be improved for both classroom and subgroups by using bootstrap classes.

5. Backend logic for subgroup creation and management will be formed.
6. New routes will be defined.
7. New src/components will be defined like class cards , subgroups cards etc.
8. Classroom schema and model will be created.

3. Flexible Assignment Management System

Support for Individual & Group Submissions:

- Assignments can now be assigned to individuals or groups.
- Group submissions will allow multiple students to collaborate on a circuit.

Assignment Review & Resubmission:

- Students receive feedback and can resubmit circuits before the final deadline.
- Teachers can annotate circuit submissions to suggest corrections.

4. Circuit Submissions with Integrated Test Cases

Automated test cases have been a long-awaited feature missing from the CircuitVerse. This will be done by using the testbench.

Step 1: Do Validations before tests :

- Only input and output elements will be allowed
- Validations of number of input and output will be done
- (optional) presence of certain circuit elements can be validated based on requirements like in sequential circuits we can verify the presence of Flip Flops
- Labels will be assigned in problem statement for input and output , their presence will be validated.

Currently, all this is implemented inthe testbench but it needs little improvisation

Step 2: Input and Output labels

- Automatically, once the user clicks on testbench, input and output will be assigned according to the labels present.
- A matrix will be attached with each problem statement, which contains input and output for the given circuit to be tested.
- Testbench will run automatically once the user clicks run tests.
- Depending on how many cases are passed, results will be displayed.

5. Canvas LMS enhancements

CircuitVerse has limited integration with Canvas LMS, requiring manual intervention to sync assignments and grades.

Seamless Assignment Syncing:

- Assignments created in Canvas will **automatically** appear in CircuitVerse.
- Teachers won't need to duplicate effort in both platforms.

Auto-Grade Syncing:

- Grades from auto-verified assignments will be **pushed to Canvas**, eliminating manual grade entry.

Embedded Circuit Simulations in Canvas:

- CircuitVerse simulations can be **directly embedded** into Canvas modules, allowing students to **test circuits** without leaving Canvas.

Project Plan

Project Size: Small 175 hours
Project Timeline (12 weeks)

May 8 - June 1 - Community Bonding Period

Week	Dates	Tasks to be Completed
Week 1	June 2 - June 8, 2025	Schemas and Controllers will be written for multi-level classroom structure
Week 2	June 9 - June 15, 2025	Frontend for the multi-level classroom structure will be made.
Week 3	June 16 - June 22, 2025	Testing and spec files will be written for the multi-level classroom structure.
Week 4	June 23 - June 29, 2025	Subgroups and Assignment management system will be implemented.
Week 5	June 30 - July 6, 2025	Frontend for the assignment management system and sub-groups will be implemented.
Week 6	July 7 - July 13, 2025	Spec files will be written and testing will be done for the features added.
Midterm Evaluation	**July 14 - July 18, 2025**	**Midterm Evaluation Milestone**
Week 7	July 14 - July 20, 2025	Backend for Circuit submission with integrated testcases will be implemented

Week 8	July 21 - July 27, 2025	The integrated testbench will be tested and frontend code written.
Week 9	July 28 - August 3, 2025	LMS enhancements will be implemented.
Week 10	August 4 - August 10, 2025	LMS feature spec files and testing will be done.
Week 11	August 11 - August 17, 2025	All the feature testing will be done along with spec files will be coded.
Week 12	August 18 - August 24, 2025	Feedback from mentors will be implemented.
Final Week	August 25 - September 1, 2025	Final testing, documentation, and code cleanup
Final Submission	**September 1, 2025**	**Project submission deadline**

Major Milestones:

1. June 9: Multi-level and subgroups structure will be finalised along with their schemas and controllers.
2. July 7: Multi-level structure and subgroups will be implemented along with testing.
3. July 21: Assignment management feature will be implemented.
4. July 28: Integrated testbench submission feature will be implemented.
5. August 4: Integrated testbench will be tested along with spec files will be written.
6. August 18: LMS enhancements
7. September 1: Final report submission.

Additional Information

Why me?

This project requires expertise in EDA tools, electronic devices, Verilog, web development, and desktop application development. My proficiency in these areas, combined with my experienceof working on the CircuitVerse codebase, gives me a strong advantage in implementing these features within the given timeframe. In the past, I have contributed to CircuitVerse by enhancing the timing diagram, implementing Verilog and testbench features, improving simulator logic, fixing bugs, and introducing new functionalities. This hands-on experience gives me with the necessary skills to successfully complete this project on a tight schedule. Additionally, my prior experience in developing the eSim desktop application and my research projects have provided me with the knowledge to integrate serial communication features effectively, thoroughly test them, and create comprehensive documentation to ensure a seamless and user-friendly experience for CircuitVerse users.

Yes, I am applying to other projects in Circuitverse, namely:

Project 6: Open Hardware Component Library and
Project 4: Assignment Suite Enhancement

Project Size and Timeline Selection

Since I have already implemented one of the circuit elements namely comparator, building other components won't take much time, as all the circuit elements share the same class and method design. I have selected my project size to be 90 hours which I think will be more than enough to compete my project. However, dueto the complexity of the serial communication feature and the desktop application of Circuitverse which is still in the development phase, I might have to increase the project size to 175 hours. This condition can arise if the serial communication library is not compatible with circuitverse codebase and we have to introduce our own changes to it to make it compatible or the circuitverse desktop application itself has many bugs that hinder the integration of serial communication protocols.

Ending Note

By working on the CircuitVerse code base and this proposal I have learned a lot of things. I have learned a lot about how big projects are managed, how open-source projects are changing the world for the better, and how much I love to code and build stuff.

References:

All the contents written in this proposal or the images used belong to me.