

sugarizer

B. Vishnu Vardhan Reddy

Google Summer of Code'24 Proposal

Sugarizer Volume Activity

Basic Info

Full Name: Bheem Reddy Vishnu Vardhan Reddy

First Name: Vishnu

GitHub: [VishnuVardhanBR](#)

Email: mmsvishnu23@gmail.com

First Language: English

Time zone: Indian Standard Time (IST / UTC +05:30)

Introduction:

I am a final year student pursuing a bachelor's in computer science and engineering with a passion for education and teaching. Sugar Labs' commitment to providing educational tools for children resonates with my personal experience as a mentor at my college, where I have taught my juniors about Data Structures and Algorithms, among several programming topics. I have actively contributed to Sugar Labs, focusing on migrating activities and enhancing functionality. My skills include full-stack development, Large language models, and databases. I have built real-time applications and deployed them successfully. Additionally, I have also worked with Three.js, for a hobby project.

Contribution to Sugar Labs:

I have made significant contributions to Sugar Labs, with 25 pull requests merged, with multiple issues raised and closed. Notable contributions include,

Activities migration from webL10n to i18next:

As a part of Sugarizer's transition from WebL10n to i18next, I have contributed in migrating **16** activities.

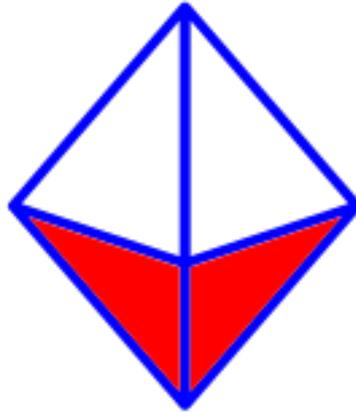
Stopwatch	Curriculum
Moon	LabyrinthJS
Flip	Chart
Calculate	Implode
XOEditor	Abacus
XmasLights	Vote
Tangram	Story
FractionBounce	DollarStreet

Enhancements/ Fixes to existing activities:

1	Added pause functionality to Blockrain activity	Merged
2	Add pause button in Gears Activity	Merged
3	Change hover effect on popup items for v2	Merged
4	Aligns chat button properly in Speak Activity	Merged
5	Fixed tooltip localization in Reflection activity	Merged
6	Fix misalignment of close button	Merged

Apart from this, I have also been active in the community, answering queries and communicating effectively.

Project Details:



Sugarizer 3D Volume Activity (350 hours)

What are you making?

I propose to develop a new Sugarizer activity called "Sugarizer 3D Volume" that aims to facilitate learning about volume using 3D dice simulations. This activity will contain several selected volumes of varying sizes, which double as dices, and can be shaken around interactively. More details are mentioned later below.

How will it impact Sugar Labs?

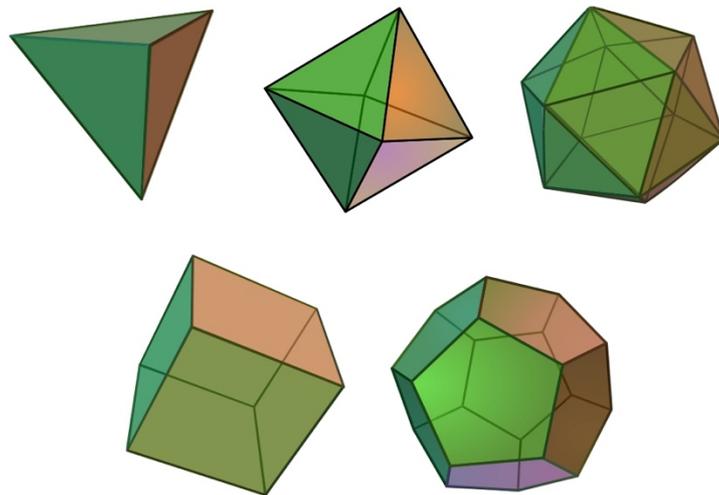
This project will enhance the educational resources available within the Sugar Labs ecosystem by providing a visually engaging tool for exploring volume concepts. It will offer interactive features such as volume manipulation, color customization, and collaborative sharing, fostering active learning and collaboration among students.

What technologies will you be using?

For this project, I will utilize Vanilla.JS mainly. I will leverage a lightweight and open-source 3D JavaScript library called [Three.js](#) for the 3D object rendering, considering its user community, this library is a good choice and is future proof. Along with it I will make use of [Cannon.js](#) for the physics, chosen for its simplicity and performance. The activity will be lightweight and interactive.

Implementation Overview:

The implementation of the Volume activity will follow Sugarizer's design practices, ensuring an optimal learning experience for children. It will feature an interactive board where students can explore volume concepts using 3D dice simulations. The activity will allow users to manipulate volumes, customize their appearance with different colors and transparency levels, and even simulate dice rolls to add a fun math perspective. The board will support intuitive interactions, such as zooming in/out and rotating, making it easy for students to engage with the content. Additionally, the activity will leverage neighborhood support, enabling collaborative learning experiences among users. With features like customizable playmats and shake board functionality, the Volume activity aims to make learning about volume both educational and enjoyable for young learners.



(Left to Right) Proposed volumes: Tetrahedron, Octahedron, Decahedron, Cube, Icosahedron.

Current Progress:

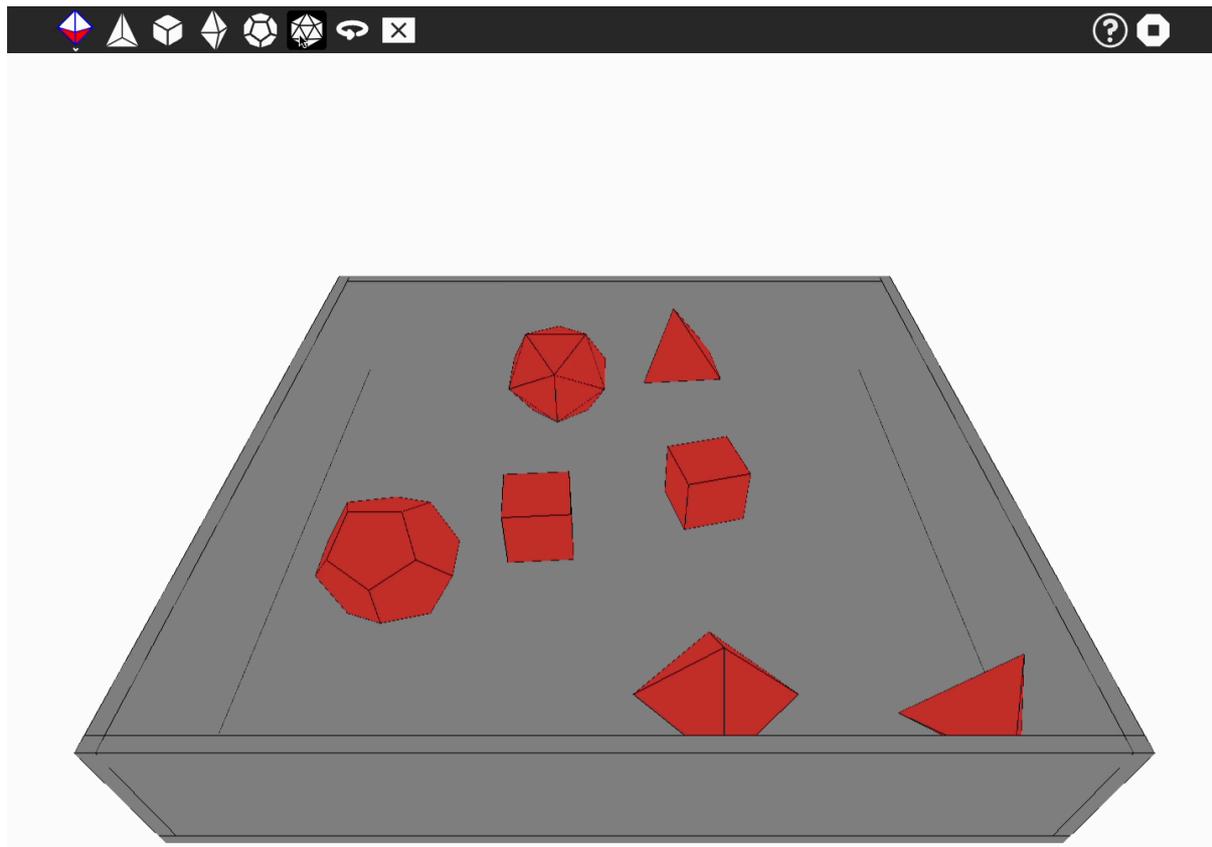


Fig: Mock-up of the activity

A mockup/ demo version of the activity has been created, where basic functionality and an early look is showcased. The user can insert various volumes in the play area, which have physics applied to them and fall to the ground and roll on collision. Volumes can also be deleted from the board. Rotation of the board is also allowed by using the rotate button. A basic shake function is implemented, but needs improvement. A shake button has been added to add random forces to the volumes, for showcase. Ability to look around using the mouse was tried but proved to be unnecessary. This prototype serves as a base for understanding the requirements properly, and a good way to communicate my vision of the activity.

Please note the above is just the base and will be improved throughout the program if selected.

A video demo of the activity is shared below, along with a live deployed version of it.

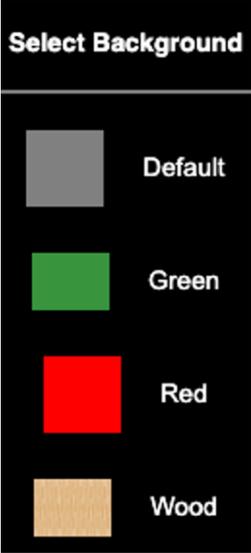
Video: <https://youtu.be/4Io7s9a761Q>

Live Website: <https://gsoc-volume-demo.vercel.app/>

To-do/ Implementation Details:

The Volumes	<ul style="list-style-type: none">• The volumes in the demo currently are basic, dodecahedron and icosahedron are to be improved with proper physics bodies matching their shapes.• Shadows for the volumes to make it realistic, also acts as reference when board is rotated.• Walls are used to represent the limits of the play area.
Insertion	<ul style="list-style-type: none">• In the demo, volumes spawn randomly on top of the board when selected. This will be changed to allow users to select a volume and click on the board to “place” the board.• Behaviour will be tested to see whether additional changes in the implementation is required.
Deletion	<ul style="list-style-type: none">• When the clear mode is selected in the toolbar, the volumes are deleted when touched/ clicked.• Will be implemented by using raycasting to check the first intersected volume.• To delete volumes behind other volumes, the user can rotate the board.
Zoom in/out	<ul style="list-style-type: none">• Pinch in/out on mobile for zoom in/ out.• A 3D magnifying icon will be fixed to the top left of the board, rotating with the board when its rotated. Varying levels of zoom will be available. <div data-bbox="762 1438 1070 1738" data-label="Image">A 3D magnifying icon consisting of a magnifying glass with a plus sign inside the lens, positioned above a red rectangular volume on a grey board. The board is shown in perspective, with the red volume placed on its surface.</div> <p data-bbox="815 1771 1018 1798"><i>Icon subject to change</i></p>

<p>Rotate board</p>	<ul style="list-style-type: none"> • Two finger gestures for mobile users to rotate the board, along with a button for desktop users. • A 3D rotate button will be fixed to the top right of the board, rotating with the board when its rotated. A 45 degree rotation per click will be available. • Additionally something like a compass could be included to show the how much the board has been currently rotated. (Optional) <div data-bbox="790 622 1141 952" data-label="Image"> </div> <p data-bbox="863 981 1066 1010"><i>Icon subject to change</i></p>
<p>Colors</p>	<ul style="list-style-type: none"> • The dices can be customized to have different colors, the default will be the buddy colors. It will be ensured that the colors are chosen keeping in mind the color blind, ensuring the numbers are readable. • Custom Colors can be selected from the toolbar, shown in a palette.
<p>Volume size</p>	<ul style="list-style-type: none"> • The size of volumes can be selected from the toolbar, selecting a new size will apply for the volumes inserted after. • Range of sizes are available, implemented simply by changing the size in Three.js and Cannon.js.
<p>Volume type</p>	<ul style="list-style-type: none"> • One of three different volume types can be chosen from the toolbar in shown a palette. • Semi-Transparent, without number, with number. <div data-bbox="539 1771 1123 1962" data-label="Image"> </div> <p data-bbox="544 1984 895 2018"><i>Volumes subject to change</i></p>

<p>Board background</p>	<ul style="list-style-type: none"> • Board background can be similarly changed, from the tool bar. • The friction of the board depends on the background selected, which will affect the way the volumes move around the board when shaken. Can be implemented using Cannon.js by using the friction parameter. • I propose different walls for each board too, making it more fun! For example, it can be a green/red net, and a wooden fence based on the background. 
<p>Shake board</p>	<ul style="list-style-type: none"> • The volumes on the board get moved around by using impulses from cannon.js. This is done when the user shakes the device using the accelerometer or presses the shake button when on desktop. • Forces are applied to the volume on x, y, z axes based on the data from the device's accelerometer, if available.
<p>Activity Sharing</p>	<ul style="list-style-type: none"> • The volumes, volume position, board background, volume size, volume type, volume colours are stored to effectively use the Sugarizer's presence framework. • Connected users can use the regular features of the activity, however, only the activity owner's accelerometer is taken into consideration. • Any subsequent complications should be taken into consideration, appropriate testing will be done.

Display total	<ul style="list-style-type: none"> • When the volumes are settled after all the forces are applied while shaking, the volumes with numbers on them are taken into consideration and the top face is checked to get a sum of all the numbers on them. • A acceptance factor is applied to disregard volumes that do not maintain a significantly upright position.
----------------------	---

Additional enhancements:

High quality ground: 3D Grass for the green playmat, Red carpet for the red playmat, and metal for the default playmat with unique see through walls.

Sound effects: Sound effects for volumes, varying with the size selected.

Project Timeline:

<p>Community Bonding: May 1-26</p>	<ul style="list-style-type: none"> • Get to know the mentor, clarify any sugarizer development-related doubts. • Make changes to the proposed timeline or proposed features and plan further actions.
<p>First Week: 27th May - 2nd June</p>	<ul style="list-style-type: none"> • Work on the existing demo and improve quality of unfinished volumes and their physics. • Add shadows for the volumes.
<p>Second Week: 3rd June - 9th June</p>	<ul style="list-style-type: none"> • Design and add the three different volume types • (Semi-Transparent, without number, with number.) • Implement functionality to change the volume sizes
<p>Third Week: 10th June - 16th June</p>	<ul style="list-style-type: none"> • Add board customization • Implement different friction for each board type • Implement calculation of sum of numbered volumes.
<p>Fourth Week: 17th June - 23rd June</p>	<ul style="list-style-type: none"> • Design the 3D zoom icon and add to the board with functionality. • Design the 3D rotate icon and add to the board with functionality.

Fifth Week: 24th June - 30th July	<ul style="list-style-type: none"> • Implement zoom and touch gestures for mobile users. • Add functionality to place the volumes directly on the board via click/ touch.
Sixth Week: 1st July - 7th July	<ul style="list-style-type: none"> • Implement the shake button for desktop users, applying random forces on click.
Seventh Week 8th July - 12th July	<ul style="list-style-type: none"> • Implement shake support for accelerometer enabled devices. • Complete the core functionality of the activity. • Test and clean the activity before the first evaluation.
Mid Term evaluation	
Eighth Week: 13th July - 21st July	<ul style="list-style-type: none"> • Add collaboration support using the presence framework.
Ninth Week: 22nd July - 28th July	<ul style="list-style-type: none"> • Localize the activity in multiple languages. • Add a tutorial to the activity including localization.
Tenth Week: 29th July - 4th August	<ul style="list-style-type: none"> • Perform through testing, debugging, and optimization of the activity, primarily collaboration and shake features.
Eleventh Week: 5th August - 11th August	<ul style="list-style-type: none"> • Implement additional enhancements. • Buffer time.
Twelfth Week: 12th August - 18th August	<ul style="list-style-type: none"> • Prepare documentation on final work and final mentor evaluation by August 19, 18:00 UTC.
Final Evaluation 19th August - 26th August	<ul style="list-style-type: none"> • Submit the final evaluation for GSoC'24.

Availability:

Sugarizer is the only organisation I will be applying to. I can dedicate 15-18 hours a week for this program. I do not have any days off planned as of now. I will be in contact with my mentor through the preferred communication channel. A weekly follow up will be maintained to check up on progress, and discuss any difficulties while developing the activity.

Post GSoC, I will be willing to contribute to SugarLabs whenever I have the time to do so. Their mission really resonates with me. This allows me to utilize my time contributing to some cause, while constantly learning new things.

As required by the mentor, here is the video of the Pawn activity from the tutorial.

Video: <https://youtu.be/i7REB4tk9XA>