

Sugarizer Word Puzzle and Chart activities

Proposal : Gsoc 2023

Basic Details:

Name: Utkarsh Siddhpura

Email: utkarshsiddhpura09876@gmail.com

Github: [UtkarshSiddhpura](#) ↗

Languages: English, Hindi

Location: Gujarat, India

Timezone: India Standard Time (UTC +5:30)

Work Hours: 9-20(UTC) Flexible

I am committed to complete project during the summer. I am available to work on the project whenever needed and will make it my top priority to ensure that it is finished on time.

Open Source:

- My journey began with the Hacktoberfest event, where I was introduced to the concept of contributing to open source Projects. I completed it.
- I made several contributions to different projects, including this: [ztm](#)
- While these contributions were small in scope, they were valuable experience.

Sugarizer:

- I was researching organizations for the Gsoc program and came across SugarLabs, the organization behind Sugarizer, a free/libre learning platform. I was immediately intrigued by the idea of creating an educational platform that is accessible to children around the world, regardless of their socioeconomic background.
- I started contributing to Sugarizer by identifying\fixing bugs in the codebase. After spending some time exploring,I have taken the time to study the codebase of the Exerciser activity, including its interaction with Sugarizer.
- Pull Request: [Sugarizer Pull Requests](#)
- Issues: [Sugarizer issues](#)
- Exerciser Issues: [Exerciser issues](#) | [PR](#)
- [Sugarizer activity development tutorial.](#)

Project Details:

1.What am i making ?

Chart Activity: The Chart activity can help children to visualize and understand complex data sets in a more engaging/interactive way, making it easier for them to learn and explore new concepts.

Word Puzzle template: The new Word Puzzle template in the Exerciser activity allows teachers to quickly create custom word puzzles during lessons, giving learners an interactive way to practice vocabulary.

- I believe that I would be a **great fit** for this project, as I have a strong track record of developing successful projects using similar technologies. My experience of building Games using JS and Web apps using React and Redux has given me a deep understanding of the tools and frameworks which are also used in the Sugarizer & Exerciser activity.
- **List of some Projects uses technologies:** HTML, CSS, JS, React, react-router & redux.
 1. [E-commerce SPA](#) | [code](#) template anyone can use for their products.
 - Authentication, Payment, SPA, Persist data, Orders, Responsive, etc.
 2. [2048](#), [Tetris](#),
 3. and many more projects see on github.

Libs/Framework i know: Sugar-web/core, Activities(js/vue), Vue, react-ecosystem, react-intl, chart.js, etc.

2.How will the Project impact Sugar Labs ?

The addition Chart activity and Word Puzzle in Sugarizer can have a positive impact on Sugar Labs by providing learners and educators with more tools to engage with the platform. Chart activity can help learners visualize and analyze data in a creative and interactive way, while Word Puzzle can improve their vocabulary and spelling skills. This can lead to more widespread adoption of Sugarizer, and greater engagement and satisfaction among users.

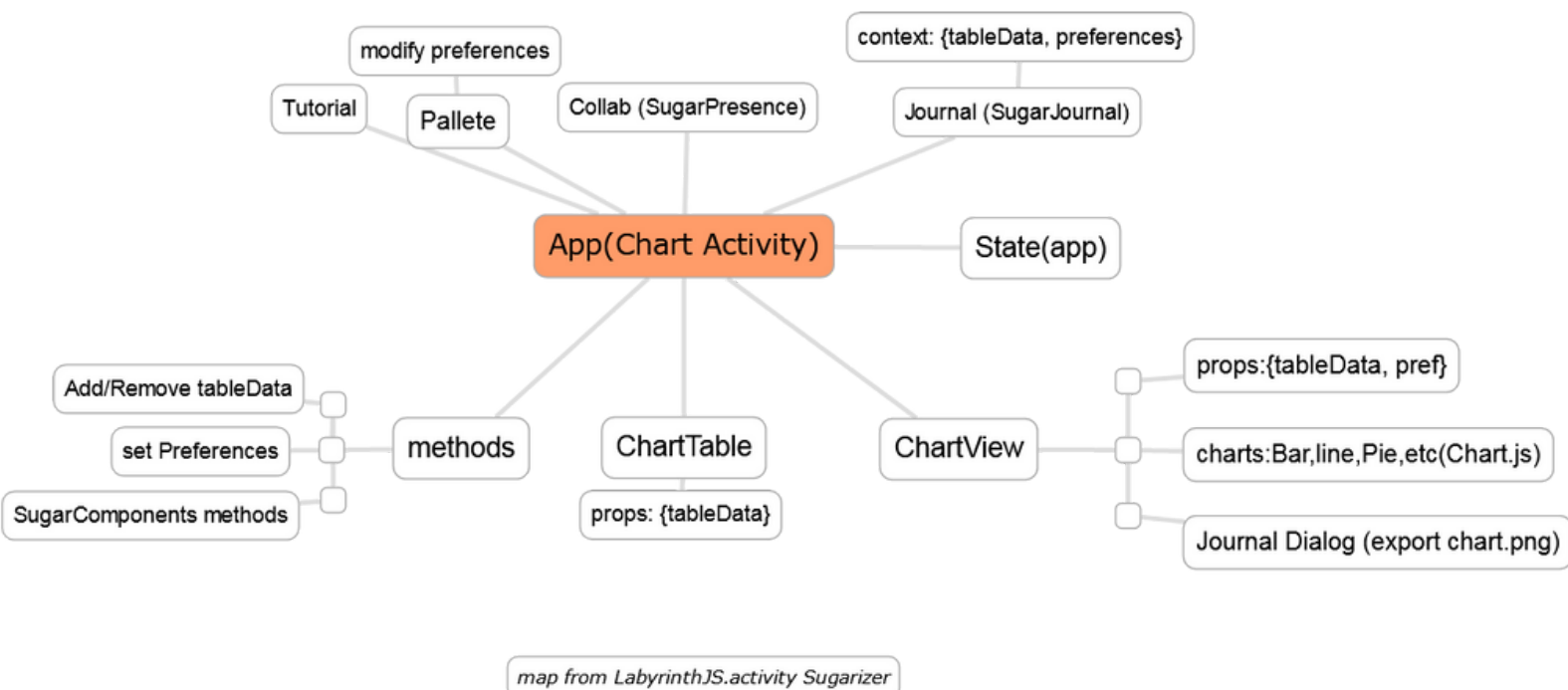
3.Technologies for Project

Chart Activity: HTML, CSS, Vue.js, Sugar-web(Vue components for Presence, Journal, Localization, Toolbar, Tutorial, etc), Chart.js, require.js.

Word Puzzle: HTML, CSS, React, redux, react-router, react-intl, intro.js & Sugar-web.

Implementation:

1. Chart Activity:





```
State(app): {
  tableData: [
    { label: x1, value: y1 },
    { label: x2, value: y2 },
    // and so on...
  ],
  preferences: {
    colors: {x:_,y:_},...
  }
}
```



```
State(app): {
  tableData: [
    [x1, y1],
    [x2, y2],
    [x3, y3],
    // ...
  ], ...
}
```

- Since, Most charting libraries allow you to pass data in the form of an array of objects, with each object representing a data point.

2. Word Puzzle Template:

- default activity obj: (to be Localized)

```
{
  "id": 7,
  "title": "WordPuzzle",
  "questions": [
    {
      "id": 1,
      "question": {
        "type": "text",
        "data": "....."
      },
      "hint": {
        "type": "text",
        "data": "....."
      },
      "answer": "word"
    }, {...}, .....
  ],
  .....
  "type": "WordPuzzle",
  "userLanguage": "en",
  "thumbnail": "puzzle.png"
}
```

- **Players/WordPuzzlePlayer.js :**

- **state:** {
 ...default_activity_obj,
 wordGrid,
 wordList,
 directions: ["|", "\\", "-", "/"], //vertical, diagonal left, horizontal, diag right
 positions: [0, 1, 2, 3,...],
 cell : [currentCell: cell, selectedCells: [...], endCell: cell] ,
 isSubmitted: false,
 currentTime: 0,

}

- **methods:** initializeGrid(onMount), createWordGrid, populateEmptyBoxes, etc...

- [createWordGrid from wordList \(Approaches\):](#)

1. Brute Force Approach (most common)

1. Select single word from wordList.
2. Select random direction & random position on WordGrid.
3. Check if word is placeable with given direction.
4. yes, Place the word, Go to step 1.
5. if not , Go to step 2.

2. !LearningApps Approach: a small Analysis [Here ↗](#)

- ✓ 3. Buckblog's Approach: See steps [Here ↗](#)


- uses Backtracking (appropriate method for this task, given its common use in various board games.)
- no retry and fails like in 2nd approach.
- places words randomly & also takes in account of already tried places unlike in 1st approach.
- Readable & Efficient.

- **Builders/WordPuzzleForm.js :** Just like other Form features

- to enter: Title, Questions & Hints of any media, answers,
- import from Journal Dialog, Test exercise & localized.

2023 Timeline (175 hr project)

days



May 4 - 28	Community Bonding Period: Finalize the features, Communicate with mentor and community members, gain a deeper understanding of code by solving issues.	25
May 29 - June 15	Coding officially begins: Implement Chart Activity, reproduces current features of sugar chart activity.	18
June 16 - July 9	Implement & test features discussed for chart activity, Base setup for word puzzle like router, defaults, state, scores, components, etc...	24
July 10 - 14	<u>Midterm</u> evaluations: Progress update report. (Finals week)	5
July 15 - 31	Word Puzzle template: implement word puzzle player with default activity obj localized & ability to share template.	17
August 1- 10	Implement word puzzle Form for teachers.	10
August 11 - 21	implement and test, convert to evaluation functionality.	11
August 22 - 28	<u>Final</u> Evaluations: Final work report.	

How many hours will you spend each week on your project ?

- min: 30 hrs/week
- max: 50 hrs/week (if required)

How will you report progress between evaluations ?

- I will maintain a markdown document on github for it.
- I will post updates of the progress, obstacles being faced, their solutions, and also regular pull requests to sugarizer & Exerciser repo.

Discuss your post GSoC plans :

- **Absolutely!** The experience and exposure that I gained from just contributing is invaluable, and I am grateful for the validation it has given me. Even though i will be ineligible for next year gsoc, I understand that the main goal is to stay committed to the organization even after the GSoC program ends & i'll do it because, I am excited to see where the Sugarlabs goes in the future.

Thank you for your consideration...