

Improve Sugarizer Server Dashboard

About me

What is your name?

My name is Anshuman Bhardwaj. I am a 2nd-year undergraduate student at Keshav Mahavidyalaya, University of Delhi, New Delhi, India.

My profile Link: <https://www.github.com/Anshuman71>

What is your email address?

My email address is anshu5074@gmail.com.

What is your first language? (We have mentors who speak multiple languages and can match you with one of them if you'd prefer.)

My first language of communication is English. I am also fluent in Hindi.

Where are you located, and what hours (UTC) do you tend to work? (We also try to match mentors by general time zone if possible.)

I am located in Gurugram, India which is Indian Standard Time (UTC +5:30). I am planning to work from 08:00 to 15:00 (UTC) but my timings are flexible. I'm very excited to work on this project and I can surely manage my time to attend a call or discuss ideas/problems with mentors at their convenience.

Share links, if any, of your previous work on open source projects?

- I have made my very first pull request to an open-source project ie **Sugarizer** for a new activity <https://github.com/llaske/sugarizer/pull/300> which is a unit converter. Currently, I am working on to make it more engaging and pedagogically beneficial for kids.
- Pull request for sugarizer-server to [enhance user experience](https://github.com/llaske/sugarizer-server/pull/125) making indication for a failed search <https://github.com/llaske/sugarizer-server/pull/125>
- Pull request for sugarizer-server to fix a [issue](https://github.com/llaske/sugarizer-server/pull/123) related to charts and misleading data <https://github.com/llaske/sugarizer-server/pull/123>
- Opened this issue <https://github.com/llaske/sugarizer-server/issues/126> for better organization of code and conceptual schema for the database.
- Opened this issue <https://github.com/llaske/ExerciserReact/issues/25> regarding warnings thrown by 'react-scripts' which should be resolved.

- I removed all the warnings one-by-one from the files and also used more performant practices to increase the efficiency of code then generated Pull request for the above issue and fixed all the warnings <https://github.com/llaske/ExerciserReact/pull/26>
- Opened Issue in ExerciserReact <https://github.com/llaske/ExerciserReact/issues/27>, Stateless react components should be made functions or functional component as per React Docs and specifications.
- Converted all the 'stateless class components' into functional components make the logic simpler, matching to the react specs and easy to maintain then generated Pull request for the above issue and changed all the 'redundant Class Components' to 'Functional components' <https://github.com/llaske/ExerciserReact/pull/28>
- Opened issue regarding better documentation about starting sugarizer using chrome <https://github.com/llaske/sugarizer/issues/305>
- Update the above issue by changing the Readme.md in this pull request <https://github.com/llaske/sugarizer/pull/306>

I have made a bunch of react-native components and one of them 'react-native-cool-snackbar' is [available on npm](#). I have made some **open source** projects myself which are available on GitHub under MIT license.

Some of my projects are listed below

- ➔ **CollegeBasket:** A mobile app for students to donate or pass on things that may be useful for others using ReactJs. <https://github.com/Anshuman71/CollegeBasket-RN> and <https://play.google.com/store/apps/details?id=com.goodtogo71.collegebasket> on google play store.
- ➔ **Site Audit:** A NodeJs, MongoDB and GraphQL backend using Apollo, for React Native mobile application. I use mongoose ORM and ES6. <https://github.com/Anshuman71/siteAuditProBackend>
- ➔ **The DuPlatform:** ReactJs mobile progressive-web app for Delhi University students for information about colleges and to share helpful digital media to peers. <https://github.com/Anshuman71/The-DuPlatform>
- ➔ **Submit:** An event management progressive web app for university teachers and students, with push notification updates. <https://awesomeattendance.firebaseio.com/>

- **Chatless:** A state-less chatting app for quick messaging built using ReactJs and Socket.io <https://github.com/Anshuman71/ChatLess> and live on <https://chat-less-79df8.firebaseio.com>
- **SnackBar:** Easy to use SnackBar component for react-native available on npm <https://github.com/Anshuman71/react-native-cool-snackbar>
- **EyeSaver:** A project on ElectronJs to save eyes from continuous screen time. <https://github.com/Anshuman71/EyeSaver--Electron-App>

Motivation

What is your motivation to take part in Google Summer of Code?

This summer I want to explore and dive into the open-source world to use the knowledge I have, to build tools for solving problems and also to keep the learning up. I have been fascinated by the open-source world ever since I came to know about Github and npm. I was amazed by the work people did to help the developer community, I understood how much they benefit us in kick-starting the project and now I want to bid hats off by contributing to open-source & be a part of this wonderful community.

Why did you choose Sugar Labs?

Sugar Labs work towards `Education for all` is inspiring and fascinating at the same time. I myself believe that to change the world the best we can do is to educate children and make study material accessible to all. I totally support their mission and I want to contribute my part starting this summer.

Why do you want to work on this particular project?

First of all, I meet all the prerequisites of this project and I do have experience working with backend technologies too. Making a dashboard is something I was looking forward to ever since I completed my react-native app College Basket. So, I decided to go for this one to learn how to implement it on a big scale.

What are your expectations from us during and after successful completion of the program?

I am just looking forward to getting things started in the open-source world. So I will be needing support during the program and after the successful completion, I would love to continue work as SugarLab member.

Project Details

What are you making ?

Sugarizer Server Dashboard is an admin tool for administrators and teachers of a Sugarizer deployment. The objective is to enhance the current dashboard in terms of user experience and responsiveness.

Currently, the process of adding Users and Classrooms is a time taking process. I am planning to add a **one-click** feature to add users and classrooms from **CSV and JSON** files. Allow admins to also export data from the dashboard in the form of csv and JSON files.

A tutorial is needed to help people set up and familiarize themselves with the server. I am planning to add a **step-by-step** tutorial featuring **gifs** and **screenshots** to make grasping it easier.

Mobile responsive web is the requirement of the hour, So we have to make sure the users are able to access and use all the feature from a mobile device.

Add API for an additional profile **'teacher'** giving the organization more flexibility assigning the roles for individuals. **Teacher**: Can access dashboard, but cannot access the general memory and storage stats A separate section on the UI end will be added then to consume the exposed API to handle operation related to the specific roles.

Add a **command-line** utility to enable admins to access the import/export feature exposed by the dashboard. I am planning to add the script in the first place and then later move on to 'yargs' for the command-line interface for a better experience and more actions.

The current implementation of database interface can be made easy to use and modifiable in the future by exposing **'models/schemas'** on the database using 'mongoose'. This will help the contributors to understand the conceptual schema as well as the design of the database and will let them add features easily.

Using the latest technology is better as it offers new features out-of-the-box also are more stable than the previous versions. So, I am planning to update all the dependencies which have better updates along with their old implementations in the codebase. EJS is the base of this project so **EJS** will be the one I am planning to start with.

The user interface needs some **renovation** work, I propose using less vibrant colors but still keep the material-ui. I am planning to adapt color schemes from **flatuicolors.com** and to follow up the **material-design guidelines** from Google as it ensures a rich experience to the users.

The tables in different sections like Users/Journal/Activities/Classrooms and Teacher(to-be-added) needs actions like sort and filter. I am planning to add sort by different columns and then add filters if required.

Bundling is a major part of any system. I propose adding webpack bundling to minify and pack all the assets and code to make it easier to ship and deploy on machines. Webpack is my personal choice but we can go with any good bundler like gulp as per the community specs.

Upgrading to the latest javascript standards is also a good ES6 offers some great features along with the sugar-syntax. It is brought up many times in the issues and I am personally looking forward to adding it to the project as it will great a sense to this big project. Babel requires a tiny configuration and it's totally worth it. I use it in all of my projects. ReactJs and other frameworks use it as well.

Beyond Project scope:

I think with increasing feature list and updates coming up, we should start thinking about GraphQL. I propose to change our backend from restful API to graph API.

It will come along with benefits like:

1. caching policy for faster results
2. real time data through polling
3. easy documentation and fast iteration
4. easy to maintain

How will it impact Sugar Labs?

The dashboard is used to control and manage the work of learners and manage and analyze all activities on a Sugarizer Server.

After the successful completion of the project, the dashboard user will be able to

- use it on a mobile device, which is great accessibility add on
- add classes and students from csv sources which reduces the time and efforts for managing the sugarizer classrooms and database.
- organize and see the information in a more digestible way with the help of sorting and charts
- make user profiles and they can give access at a more refined level
- make faster deployment and have optimized code to use fewer resources as well as giving better performance

The documentation and scripts will also take away most of the complex part away from the end user. The linting implementation will make it easier to maintain and contribute to the project later on which is the need of hour right now.

Overall as I think of it, this would be a good step towards the expansion for **Sugarizer-server** and hence **Sugarizer** in itself.

What technologies (programming languages, etc.) will you be using?

Javascript will be the core language used along with HTML5 and CSS(material-ui). MongoDB will be used as a database store. NodeJs as the javascript runtime with EJS framework. Gulp/Webpack as the bundler.

Timeline

Break down the entire projects into chunks and tell us what will you work on each week. As the summer goes on, you and your mentor will adjust your schedule, but it's good to have a plan at the beginning so you have an idea of where you're headed.

I have divided the project into smaller chunks of 8-12 days for better tracking of progress and goals.

Time Period	Milestone
May 6 - May 27	Community Bonding Period: Communicate with my mentors and other members to finalize the features also set up the final schedule for the project.
May 28 - June 10	Extend dashboard UI <ol style="list-style-type: none">1. Add sorting by column on all view2. Add size column on Journal views with the sort3. Add Classroom counter
June 11 - June 20	Extend dashboard UI <ol style="list-style-type: none">1. Add export/import to CSV2. Improve UI and UX3. Add tutorial
June 20 - June 24	Create a new teacher profile <ol style="list-style-type: none">1. Update API to allow a new type of profile named "teacher" with view on its classrooms only Take into account feedbacks for improvement before evaluation.
June 24 - June 28	Phase 1 Evaluation
June 29 - July 10	Create a new teacher profile <ol style="list-style-type: none">1. Add screens to handle teacher profile2. Update global view to handle teacher profile

July 11 - July 22	Build Improvement <ol style="list-style-type: none"> 1. Use gulp to bundle and minify 2. Upgrade any outdated dependency and EJS version Take in account feedback for improvement before evaluation
July 22 - July 26	Phase 2 Evaluation
July 27 - August 10	Add scripts <ol style="list-style-type: none"> 1. Create an initialization script to generate settings file and automate first admin creation 2. Provide import/export scripts for Users/Classrooms
August 10 - August 19	Buffer period and Documentation for the project. Take in account feedback for improvement before evaluation
August 19 - August 26	Final Evaluation

If you will be off-the-grid for a few days, then mention those in the timeline.

I don't have any other plans for the summer break. I just have my finals in May and I will be done with them well before the coding period begins so I am sure that I will be all set for the work throughout the period.

GSoC has three evaluations, once after every 30 days. Highlight the work you plan to complete before each evaluation.

1. Complete the **Extend dashboard UI** task as described in the timeline, before the first evaluation and submit the documentation for it. I am planning to keep things fluid and I will be taking feedback and suggestions from the mentors to make sure the quality is being maintained.
2. Complete the **Create a new teacher profile and Build Improvement** task as described in the timeline, before the second evaluation and submit respective documentation.
3. Complete the **Scripting** part of the specifications and finally complete and combine all work to see how it performs in harmony with each other. At last, take a final feedback from mentor to add any final touch or a new task to the project before submitting the final evaluation.

Also, discuss your plans after the GSoC period ends. Do you plan to continue working on the project after GSoC ends?

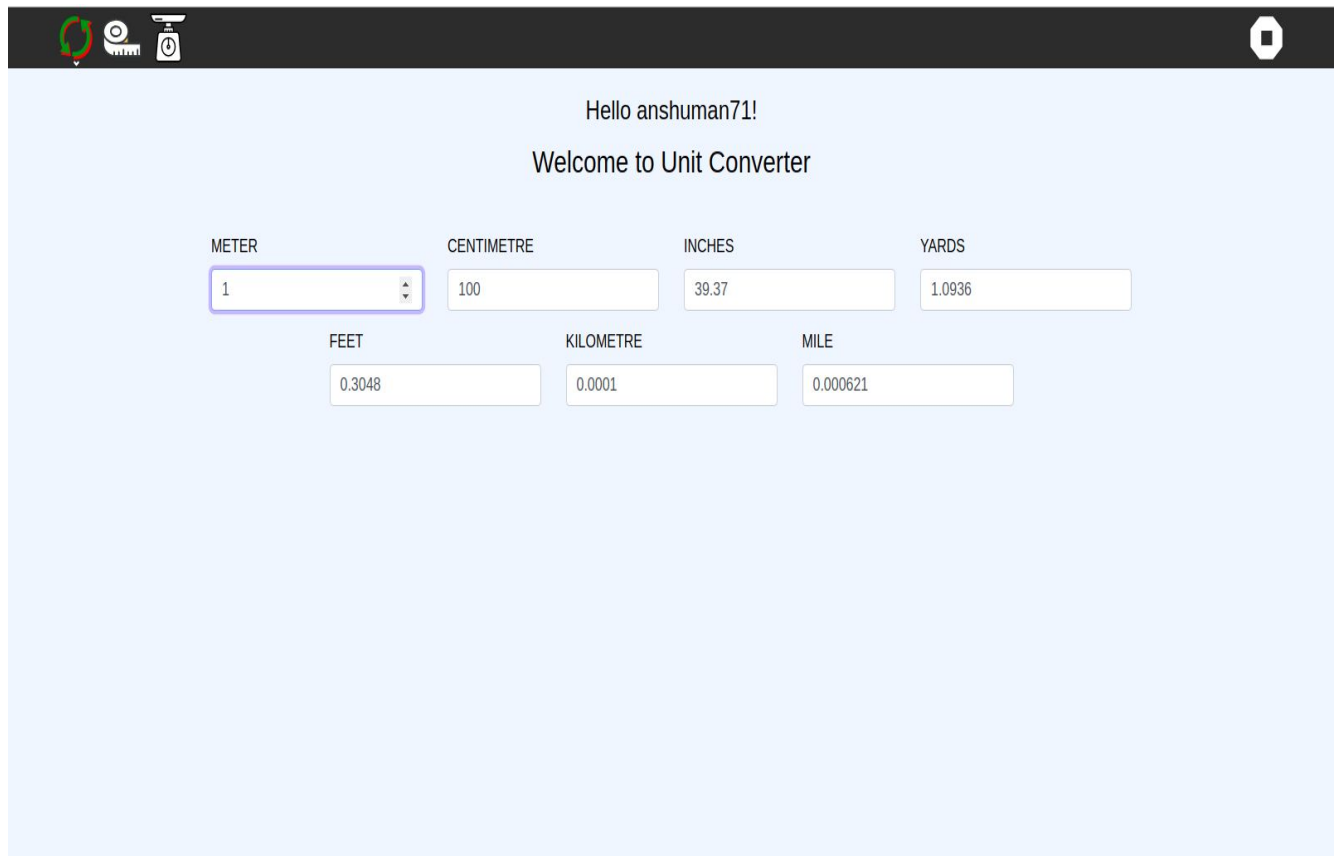
I am really excited about the project and I find it very interesting. I will be more than happy to be a contributor to such an amazing project. I am planning to add new features and new tech stack, for the better future of the ecosystem. I believe there is a lot to discover and to offer to the users, so I am looking forward to be a part of Sugar-Labs.

Mention how much time will you spend each week working on your project

I will be giving at least 35-45 hours per week. I am completely dedicated to GSoC this summer and coming months. I have scheduled my routine very well to match the time constraints.

Sugar Lab's Motivation

Convince us that you will be a good fit for this project, by sharing links to your contribution to Sugar Labs



After going through the pawn activity tutorial I alongside build this useful activity for kids. The pull request for the above activity <https://github.com/llaske/sugarizer/pull/300>.

A pull request for sugarizer-server to [enhance user experience](https://github.com/llaske/sugarizer-server/pull/125) <https://github.com/llaske/sugarizer-server/pull/125>

A pull request for sugarizer-server to fix a [issue](https://github.com/llaske/sugarizer-server/pull/123) related to charts and misleading data statistics <https://github.com/llaske/sugarizer-server/pull/123>

Opened this issue <https://github.com/llaske/sugarizer-server/issues/126> for better organization of code and conceptual schema and this one <https://github.com/llaske/ExerciserReact/issues/25> regarding warnings thrown by 'react-scripts' which should be avoided.

Generated Pull request for the above issue and fixed all the warnings <https://github.com/llaske/ExerciserReact/pull/26>

Opened Issue in ExerciserReact <https://github.com/llaske/ExerciserReact/issues/27>, Stateless react components should be made functions or functional component as per React Docs and specifications.

Converted all the 'stateless class components' into functional components make the logic simpler, matching to the react specs and easy to maintain then generated Pull request for the above issue and changed all the 'redundant Class Components' to 'Functional components' <https://github.com/llaske/ExerciserReact/pull/28>

Opened issue regarding better documentation about starting sugarizer using chrome <https://github.com/llaske/sugarizer/issues/305>

Update the above issue by changing the Readme.md in this pull request <https://github.com/llaske/sugarizer/pull/306>

I have contributed to different projects from sugar-labs. I am now getting along with the whole sugarizer-ecosystem and how sugarizer is contributing to different technologies and how they are helping the next generation.

I am well aware of the technology required for this project and I have been working on these for about 2 years now. I have worked as an intern last summer break and had a great experience there. I learned about teamwork and project management there. I worked on takkeh.com and its [mobile app](#) available on Google play.

DATE :
23 JULY 2018



Certificate of Excellence

IS PRESENTED TO

ANSHUMAN BHARDWAJ

OF

KESHAV MAHAVIDYALAYA, DELHI

for successfully continuing excellence in
FULL STACK WEB AND MOBILE DEVELOPER .
Awarded in recognition of outstanding support and
significant contribution to the
BODHOJAS Solutions Pvt. Ltd
(Brand Name - PenPencil).

A handwritten signature in black ink, appearing to read "Prateek Maheshwari".

Prateek Maheshwari
(Founder & CEO)

This Certificate of Excellence, issued for
maintaining consistently high standards
of workmanship and accomplishment
in increasing professional ability .

BODHOJAS Solutions Pvt. Ltd.
Email : penpencil.tech@gmail.com

Certificate of Excellence from the Internship

Moreover, I will be only working on this project so there will be no issue regarding falling short on time and not being able to achieve the target. I will have my full dedication and commitment to this project.