

# Google Summer of Code 2019

## Submitty

Streamline instructor configuration of automated grading

---

### BASIC INFORMATION:

#### 1. Personal Information

- Name Drumil Patel
- Github [drumilpatel2000](#)
- Email [drumilpatel720@gmail.com](mailto:drumilpatel720@gmail.com)
- Phone No. +91-6351173421
- Location India 
- Time Zone India (UTC +5:30)
- Slack Drumil Patel
- Linkedin [Drumil Patel](#)
- Hackerrank [flipbyte](#)

#### 2. University Information

- University: Indian Institute of Technology, Roorkee
- Majors: Electrical Engineering
- Current: I Year (expect graduation in 2022)
- Degree: Bachelor of Technology (4 Year Program)

#### 3. Contact and Working hours

- Reachable anytime through **email, slack or contact number.**
- Typical working hours include:-
  - i. UTC 0430 - 0730 hrs (IST 1000 - 1300 hrs)
  - ii. UTC 0930 - 1230 hrs (IST 1500 - 1800 hrs)
  - iii. UTC 1530 - 2030 hrs (IST 2100 - 0200 hrs)

# CODING SKILLS:

## Programming Languages

- Sound Knowledge of OOPs and MVC Architecture
- Moderate knowledge in **Python, C++, and JAVA**
- Proficient in **HTML/CSS, Javascript, PHP**
- Databases - **MySQL, PostgreSQL**

## Hardware Specifications

- HP Pavilion 15 - cc129tx
- Intel® Core™ i5-8250U
- 8GB DDR4 RAM
- NVIDIA® GeForce® 940MX

## Development Environment

- Ubuntu 18.04 or Windows 10
- **PHPStorm** for development in PHP
- **PyCharm** for development in Python
- **DataGrip** for development in PostgreSQL
- **Git** for version control

# ABOUT ME:

I am an 18-year-old freshman, currently enrolled in electrical engineering at IIT Roorkee, I developed a passion for programming, web development in my first semester and from then most of my time goes into reading about new technologies and developing software. I have been contributing to open source regularly since about three-four months now.

I have an experience of working closely with a team as I am an active member of [Information Management Group](#) at IIT Roorkee, a bunch of passionate enthusiasts who manage the [institute main website](#), internet and intranet activities of the institute.

## My projects:

- A-farmer-s-help:-
  - An android application which helps a farmer to buy adequate quantities of fertilizers for a crop plantation and the required soil concentration.
  - It uses [firebase](#) to send data to fertilizer wholesaler

## Open Source Contributions:

- Submitty

### Pull request

- [#3538\(Open\)](#) [BUGFIX] Highlighting checkbox label on focus
- [#3529\(Open\)](#) Resubmit "[VPAT] html tables are commented and data tables are given th...
- [#3520\(Open\)](#) [Bugfix] Trimming unwanted spaces in blank input
- [#3498\(Open\)](#) [BUGFIX] Fix size of textarea when editing the post
- [#3495\(Open\)](#) [BUGFIX] Fix missing link in notifications when the post is deleted
- [#3493\(Open\)](#) Add shortcut(KeyD) for discussion panel in grading portal
- [#3491\(Open\)](#) Enhancing discussion gradeable portal
- [#3485\(Open\)](#) Request not submitted in regrade request
- [#3483\(Open\)](#) Fix CSS of button for regrade request
- [#3481\(Merged\)](#) Removing extra space in grading discussion portal
- [#3478\(Merged\)](#) Apply trim to frontend and backend in Authentication
- [#3463\(Open\)](#) [Bugfix]Aligning select color tag in 'Edit categories'
- [#3454\(Open\)](#) Enhancing new gradeable
- [#3444\(Open\)](#) [FEATURE]TA/Instructor can lock Thread
- [#3381\(Merged\)](#) Add stoplight legends to grade details page

- [#3358\(Merged\)](#) Error Handling on Login Page
- [#3336\(Merged\)](#) Customizable theme using layout tables
- [#3328\(Merged\)](#) Radio button grouping
- [#3322\(Merged\)](#) Contrast Error in Global footer view
- [#3321\(Merged\)](#) Remove redundant link in the sidebar
- [#3296\(Merged\)](#) Change contrast and remove inline CSS in the navigation button

## Issues

- [#3539](#) Tab navigation not working properly
- [#3519](#) No Validation for blank input
- [#3484](#) Wrong command in regrade request
- [#3482](#) Improper behavior when regrade request is resized
- [#3477](#) Authentication failing due to unwanted whitespaces
- [#3461](#) Misalignment on editing categories
- [#3410](#) Small textarea for editing in the Discussion forum
- [#3380](#) Sidebar links tooltip CSS overridden on new gradeable page
- [#3366](#) Wrong command for database
- [#3353](#) Redirection not handled properly on the login page
- [#3293](#) Improving form validation in New Gradeable

# PROJECT IDEAS:-

## Title:- Streamline instructor configuration of automated grading

### Introduction

Autograding is one of the greatest features which makes the instructor work more easier. It helps the instructor by grading score without reading the code. It basically compares the output file for a predefined input file with predefined output file given by the instructor.

Currently, instructors must write a configuration as a config.json (and any necessary additional files) and upload or store these files on the local file system. Automated grading can be improved by implementing the following features:-

- Graphical User Interface for creation config.json, input, etc
- Automated Creation of Expected Output Files
- Randomly Generate Test cases Input

We can reduce learning curve for new instructor by providing getting started tour feature.

- Getting started tour feature can be implemented using introjs

### Project Impact

#### Graphical User Interface for creation config.json, input, etc:-

Web GUI support for creating config.json, input, etc. This will result in streamlining the assignment configuration process for non-technical instructors, relevant for use in non-computer-science/non-programming courses.

#### Automated Creation of Expected Output Files:-

Presently instructor has to provide both inputs as well as output files if any exist. We can provide the instructor with the option to provide manually or auto-generate output files. This will auto-generate output files can be generated at assignment build time

## Randomly Generate Test Cases Input:-

As output files can be easily autogenerated. Auto Generated input files will make the instructor work easier. The instructor can set some set of rules in config.json which will provide random input

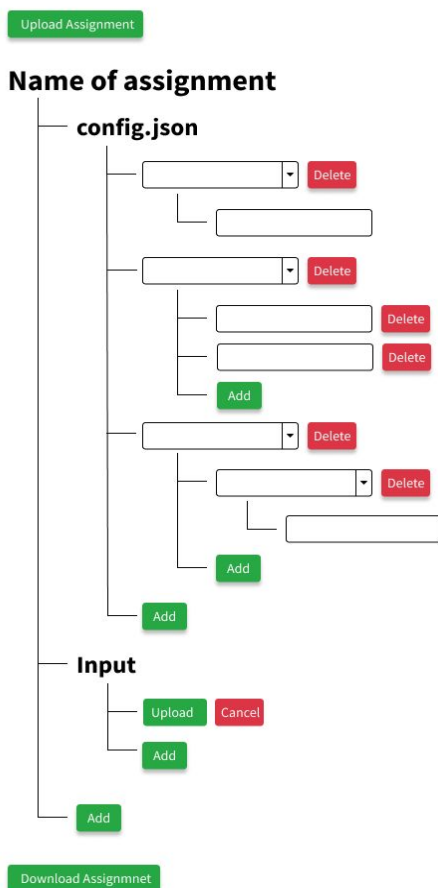
## Proposed Workflow

### Phase 1:- (GUI for configuration of automated grading)

#### 1.1 Add Basic GUI of a nested list of input tags

- Basic GUI for streamlining the process of creating config.json
- GUI will be a nested list of selects and input tags
- The particular section of list can be added or deleted using corresponding buttons

## Assignment Configuration



#### 1.2 Add a predefined set of input for the particular select tag

- For eg “test cases” should contain a list of the object so it will contain select tags
- “part\_names” should contain a list of words so it will contain a list of textarea

### 1.3 Add methods to direct upload and download assignment created in by GUI

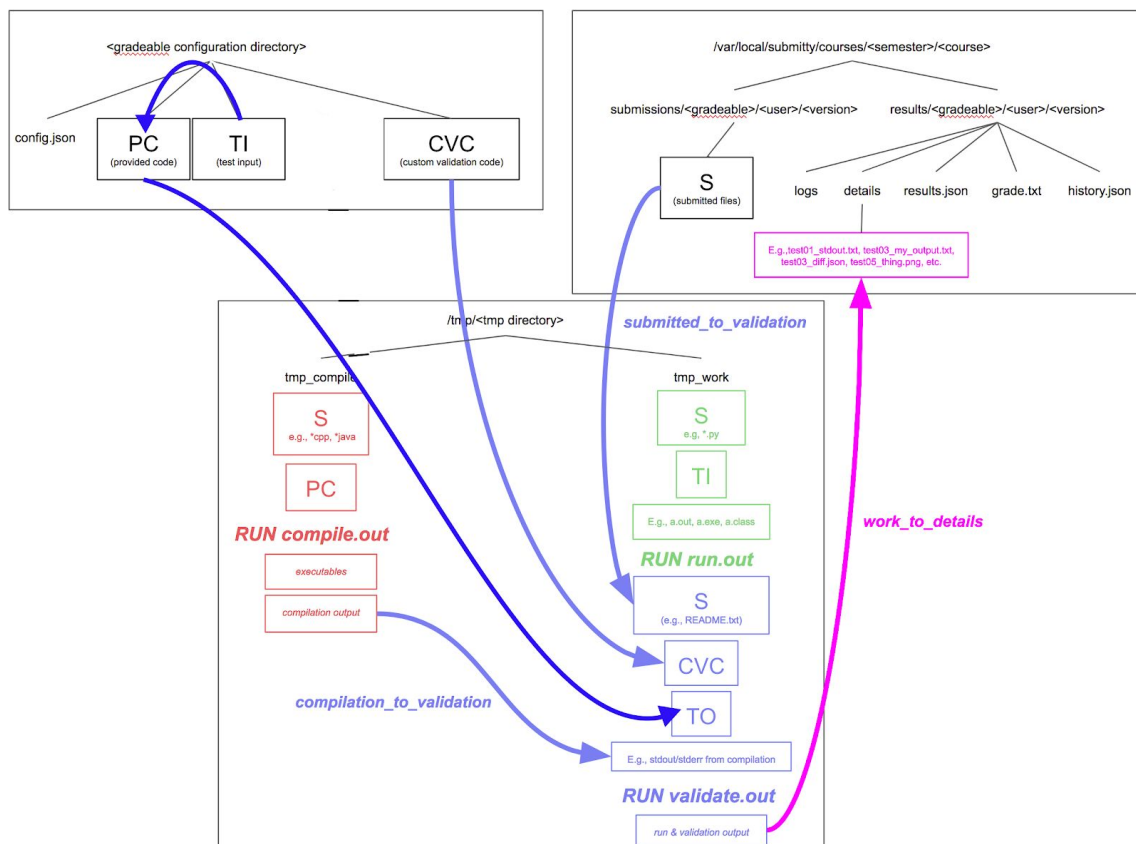
- Method to download config.json and other files directly into local computer
- Method to upload files directly to servers

### 1.4 Edit option for config.json and other files uploaded previously

## Phase 2:-

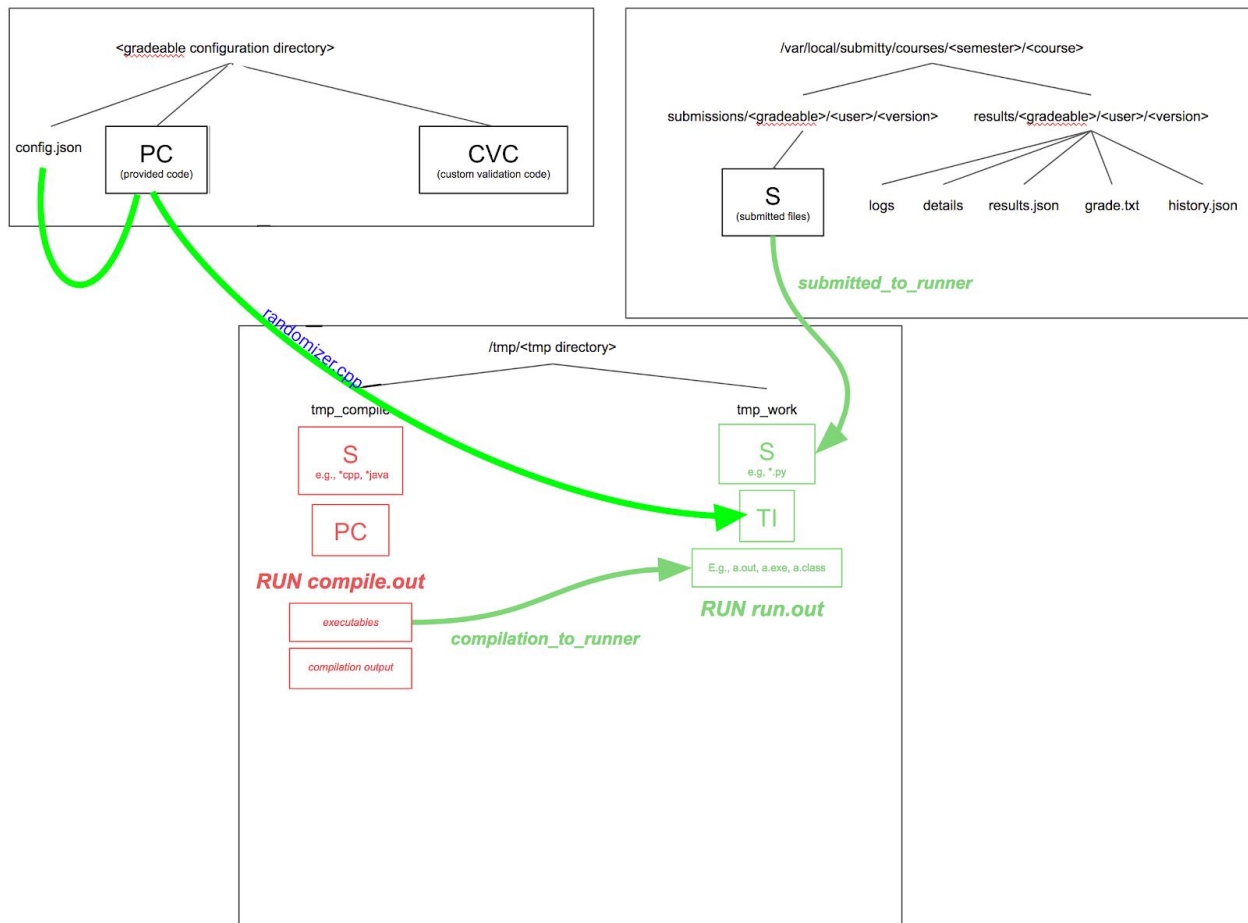
### 2.1 Add support for automated output

- “output from” will contain input file which would be used to auto-generate the output
- “output form” will be different for each test case
- Add solution.cpp for automated output
- “output form” will call solution.cpp which will render “output\_x.txt” where x is the respective number of input file



## 2.2 Add support for automated input for test cases

- “input rules” is a set of input rules which would help to generate random
- Randomizer.cpp will run when test case will contain “input rules”
- Randomizer.cpp will take input from input rules and generate random “input\_x.txt” where x is the number of test case
- Example of some of the test rules are as follows
  - ◆ number(1,10)
  - ◆ Array(number)
  - ◆ Array(Array(number))





## Phase 3:-

### 3.1 Implement tests for automated output and randomize input

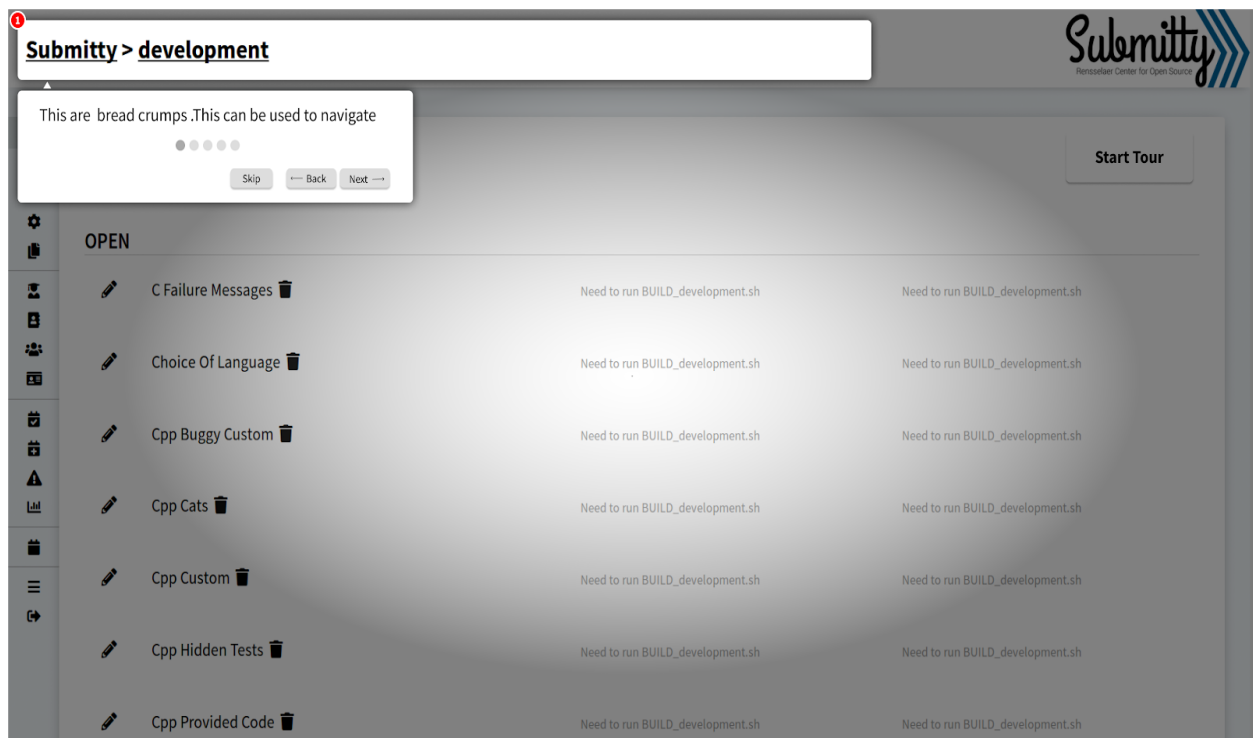
- Write unittest for solution.cpp
- Write unittest for randomizer.cpp to validate result of some of basic “input rules”

### 3.2 Write documentation for GUI, automated output and randomize input

### 3.3 Make bug fixes and code quality refactorings.

### 3.4 Add getting started tour feature in instructor UI

- Adding tour feature will decrease the learning curve for new instructor
- Use [introjs](#) for adding tour feature



## Extra Milestones:-

- Add getting started tour feature in student UI
- Add getting started tour feature in TA UI

# PROPOSED TIMELINE:

## Before GSOC

- Add Notification settings([#2797](#) and [#3160](#))
- Add Announcement only Forum([#3152](#))
- Add Scantron Feature([#3468](#))

## Community Bonding Period(May 6 - May 27)

Interact with the mentors of the project and set up feedback loops.  
Continue to refine the plans for the project in consultation with the mentors. Get more familiarize with **twig**, **PostgreSQL** and codebase  
Get involved with the community which is after all that this period is for.

## Week 1(May 27 - June 3)

- Add Basic GUI of the nested list of input tags

## Week 2(June 4 - June 10)

- Add a predefined set of input for the particular select tag

## Week 3(June 11 - June 17)

- Add methods to direct upload and download assignment created in by GUI

## Week 4(June 18 - June 24)

- Edit option for config.json and other files uploaded previously

## Phase 1 Evaluation

## Week 5 - 6.5(June 25 - July 4)

- Add support for automated output

## **Week 6.5 - 8(July 5 - July 22)**

- Add support for automated input for testcases

## **Phase 2 Evaluation**

## **Week 9(July 23 - July 29)**

- Implement tests for automated output and randomize input

## **Week 10(July 29 - August 6)**

- Write documentation for GUI, automated output and randomize input
- Make bug fixes and code quality refactorings.

## **Week 11 - 12(August 7 - August 19)**

- Add getting started tour feature in instructor UI

## **Final Evaluation**

# **MOTIVATION**

**Google Summer of Code** is a nice platform to get acquainted with the open source community and their skillful mentors. It gives you a professional work experience in your college years where one collaboratively builds a product for the welfare of the society. In this process, both the individual and the community improves and grows.

**Submitty** is an open source programming assignment submission system. Submitty provides a platform with which your development skill can be helpful to the huge community of students. Submitty is my first open source organization to which I actually contribute and I am glad about it. Submitty helped me to gain new skills and improve my present skills. The mentors here are knowledgeable and very responsive which makes the development of the project faster and encourage the number of contributors to get involved

## AVAILABILITY

My vacations start on 5<sup>th</sup> May 2019 and end on 11<sup>th</sup> July 2019. The official GSoC period is from 6<sup>th</sup> May 2019 to 26<sup>th</sup> August 2019. I can easily devote 50-55 hours per week until my college reopens and 40-45 hours per week after that. I am also free on weekends.

Other than this project, I have no other commitments or vacations planned for the summer. I shall keep my status posted to all the community members on a weekly basis and maintain transparency in this project.

## EXPECTATIONS

It is a great opportunity for me to apply my knowledge on a practical scale. I wish I can be a part of this program for gaining experience in software development and give back to the open-source community.

## AFTER GSOC

I have already learned a lot and picked up many skills by contributing to **Submitty** and I plan to continue to contribute to this organization by adding features or solving issues in the current ecosystem.

Till now, the highlight of my experience with **Submitty** has been the fast-paced development of the website and the active involvement of the mentors. With the community growing continuously, I feel responsible for the projects I contribute to. Having picked up a lot of development skills, my major focus after GSoC would be to enhance my mentorship skills so that I can give back to this community by helping other people navigate around and reviewing their contributions.

## RESOURCES

1. [Submitty Project Ideas](#)
2. [Submitty Documentation](#)
3. Github repositories of [Submitty](#), [Submitty.github.io](#)
4. Introjs:-
  - a. [Github](#)
  - b. [Docs](#)
  - c. [Demo](#)

\* \* \* \* \*