

SWE, Group-7: Documentation

PlagoD



**Anti-plagiarism assignment submission system using
Google Classroom API**

Client: Dr. Vineeth Balasubramanian

Chinmay Chandak, Nagendra Sainath, Vamshi Ratna Reddy, Satya Vasanth Reddy

1. Motivation

Plagiarism is rampant in today's world. Students 'take help' from each other, and submit assignments, and hence circumvent their incumbent work. Also, Google Classroom has developed an awesome platform where all the classwork, projects, and quizzes/assignments can be managed very easily by the Professors, and can be easily accessed by the students. However, obviously, plagiarism still pervades on this platform. Our goal is to ensure that the degree of 'infringement of copyright' is reduced, and so, we propose to add an extra layer of security to implement the same.

2. Concept

What if a follow-up assignment, based on the concepts tested in the main assignment, is given to be solved (posted on the Classroom) as soon as the deadline of the main assignment passes? Also, this assignment (more like a test) would have a short time-limit to solve, something which the faculty can set easily. This would ensure that all the students who have submitted the assignment by working out the problems themselves, and understood the intricacies behind them, would also probably score well on the follow-up assignment; as opposed to this, people who've just blatantly copied would get lower grades on the test. The combination of these marks could be aligned towards giving the final grade of the assignment.

Some issues that have to be tackled by the faculty are:

- 1) Would the set of questions asked in the follow-up test be the same for all students?
- 2) The time-limit should be set considering the type of questions being asked.
- 3) How would the combo-grading policy be?

The honest students would obviously have to spend extra time 'proving' their integrity, but there seems to be no other way out.

3. Software Technologies used

Now that we know what we have to implement to achieve our goal, we mention what all we have used to develop this project.

a) Google Classroom API:

Google provides all the Classroom features which were integrated in our software through its API. We replicated the Google Classroom environment, using some of the important features, like creating and posting an assignment, scheduling deadlines, etc.

b) Google Form API:

This API was used to provide the faculty with a link to a Google Form which he/she can schedule to be posted as the follow-up quiz once the main assignment's deadline passes. This form, like the traditional Google Form would have all the varied options for the faculty to create a marked assignment test.

c) Spring MVC (Model-View-Controller) framework:

All the code has been written using this Framework.

- The Model is the lowest level of the pattern which is responsible for maintaining data. It has the overview of what our software would look like.
- The View covers the different aspects of what kinds of features and operations a user can access; for example, the student features would be exactly same as the normal Classroom software, but the faculty would have an extra feature of creating and scheduling the extra test.
- And finally, the Controller is responsible for controlling the flow of the application execution. When you make a request (means request a page) to the MVC application, a controller is responsible for returning the response to that request.

Thus, overall, the MVC framework provides a modularised, easy-to-integrate milieu for UI/UX, development code, and other API services.

d) GitHub & Visual Studio Team Services:

All the code our team collated from each other, and then the final creation of a versioned, all-integrated software, was managed on GitHub, and Visual Studio.

e) Jetty & Maven:

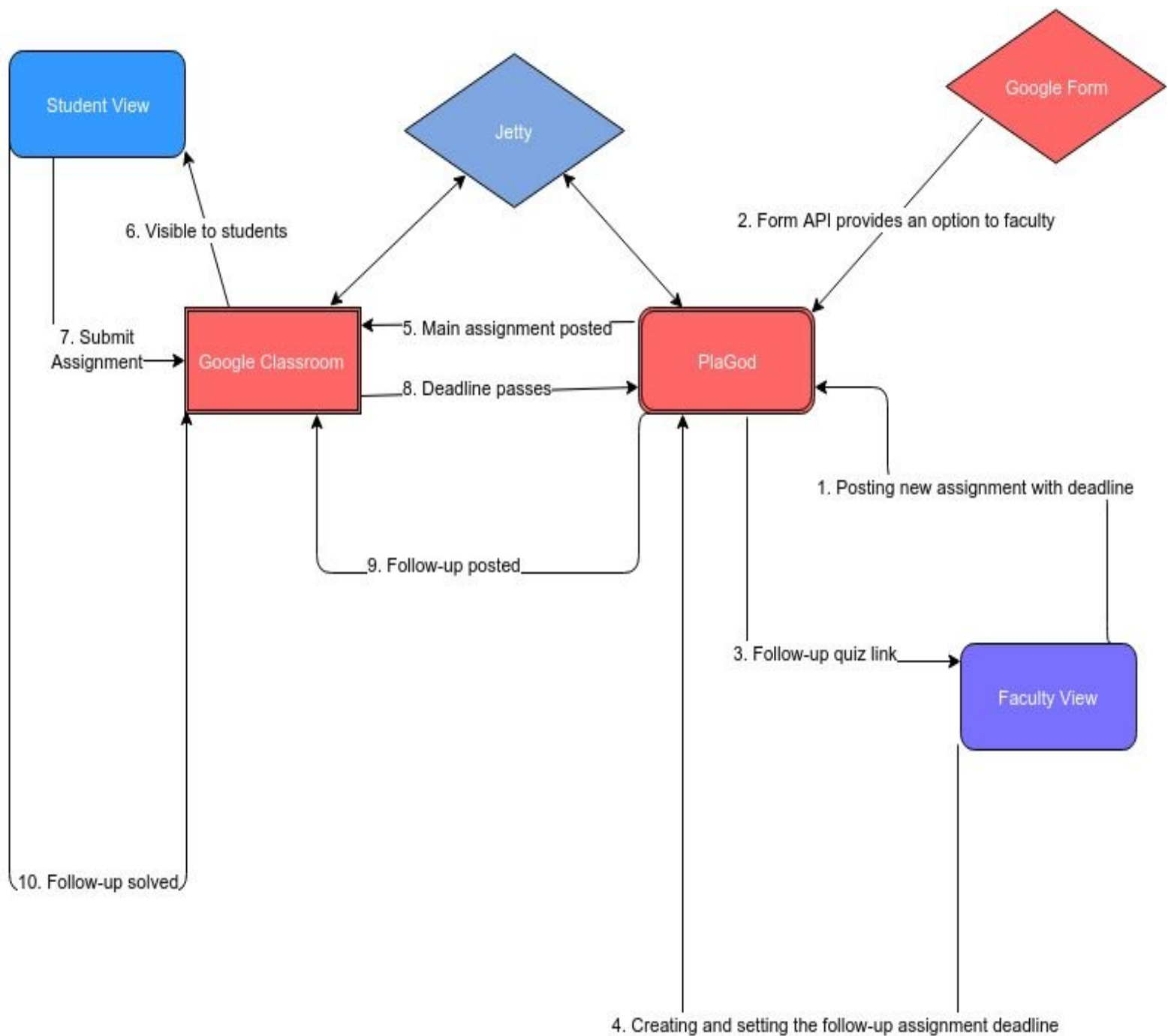
Eclipse Jetty is a Java HTTP (Web) server and Java Servlet container which we used to host our software. Maven is the repository which has the necessary packages to manage the network and machine-to-machine communication for our software.

4. Authentication process

Another issue which we faced was the authentication of a Google user through our software, because it is overriding Google Classroom, and hence it takes a toll on Google's API security. So, we had to store the cookies of the previous sessions, and then perform all the further actions like posting an assignment, etc. using the authentication saved in the cookie.

5. Possible extension

- Google Classroom does not allow selective group assignments, i.e., separate follow-up assignments could be assigned to different groups of students, just like different sets of exam papers.
- There is no option in the Classroom API for uploading an assignment yet. All the links to relevant PDFs, and other material have to be put in the description.



**FLOWCHART DEPICTING THE CONTROL FLOW OF OUR
OVERRIDING SOFTWARE ARCHITECTURE**

6. Overview of the control flow:

- A. The faculty or course instructor creates a main assignment with a deadline.
- B. Our software, PlaGod, hosted over a Jetty and Maven supported server, asks the instructor whether or not he would like to hold a plagiarism-check follow-up quiz.
- C. In case, he/she does not opt for it, the further procedure would be same as the emulation on Classroom.
- D. If he does opt for a follow-up quiz, he would be provided a link to upload the follow-up assignment, along with the deadline (duration of this assignment).
- E. When the deadline of the main assignment passes, the follow-up assignment is posted automatically, and all the students must solve it within the stipulated time.
- F. The final grade would be a combination of these two assignments, and therefore a little stringent anti-plagiarism policy is being implemented through our software.
- G. As mentioned before, there are certain issues which, if resolved, would further improve the efficiency of our software.
 - a. Multiple sets of follow-up assignment questions.
 - b. A probable bio-metric scanning of the student's eyes throughout the duration of the follow-up assignment.

7. UML diagram showing the file hierarchy

