Using the SEAS Code Repository For Courses

Overview

This page discusses how to apply the SEAS Code Repository at

- [http://code.seas.harvard.edu](http://code.seas.harvard.edu)

Getting Started with code.seas

The SEAS Code Repository is a central service that provides large chunks of the typical workflow used in courses, including

- “handing out” of assignments
- providing updates to homework contents
- self-service tools for students to manage code and development work
- basic mechanisms for submitting work for evaluation

However, the structure of the coursework will strongly influence how the SEAS Code Repository can be used in a course, and how well it fits the needs of any particular courses.

The following are some thoughts on how it could be used.

Use Cases for Courses

Distribution of Course Material

In its simplest form, the SEAS Code Repository can be used to publish assignments, in particular assignments that include initial stub code, to students.

This would entail something like the following steps:

1. Create a project for the course (i.e. “CS1234”)
2. Create a team for the instructors, and change the project ownership to that team
3. Create repositories for each assignment, and populate each with the files for the assignment
4. Create wiki pages (if desired) for each assignment
5. Point the students to the main page of the project, for getting access to the assignments.

Source Code Management for Individual Students

Another approach to use in courses is to encourage students to use the tool for managing their assignments, likely in conjunction with publishing the assignments.

Students would

1. Create a repository on their own or Clone a published assignment,
2. Set the repository to be private
3. Develop against the codebase
4. Submit completed assignment using the preferred method of the instructor.

Used in this way, the SEAS Code Repository acts as a development productivity tool, and encourages code software engineering practices.

Source Code Management for Student Collaborations

In this scenario, students will use the system to collaborate on assignments, again either created from bare repositories themselves or by cloning published assignment repositories.

The steps involved would be similar to the above, except that the students would manage user collaborators individually. Once students would “own” a repository, and then invite other students in a “section” or study group to have commit access. The student owner would likely be responsible for final submission of the assignment on behalf of the group.
Automated Submission of Assignments

More involved use of the SEAS Code Repository to mimic real course assignment workflow would include using the service to "hand in" assignments at a given time. This approach assumes the following:

- Assignments are published as repositories by instructors
- Students all have accounts
- Students work on copies of the assignments that are private, but hosted on the server.

In this model, the instructors use the fact that all clones of a given repository are tracked, so that for a given assignment, a list of student work repositories can be generated.

At the deadline, all student clones are "handed-in" automatically by checking out each, and then integrating that student's work as a branch of the main assignment. The resulting repository, with one branch for each student or section, represents the "completed homework bundle" for that assignment, ready for grading.

This functionality is not yet available in the GUI, and would require a small amount of backend work by Academic Computing staff to implement for a given course (mail us at achelp@seas.harvard.edu if you are interested in this option).

Open Software Engineering

This approach most closely resembles real-world software engineering. Students openly work in groups on different aspects of the same project, negating the need for careful tracking of permissions. Each repository (or set of repos) developed against is some component or aspect of the topic of the course, and the assignments may build up over the term to a initial "version 1.0." release.

The repository service serves as the central code management tool for the coursework, and the final project may actually involve merging all repositories into a working system.