

Code Review Strategy

Process:

One developer (the reviewer) will sit with the developer who wrote some code for this project and be guided line-by-line to understand what the code is doing and catch any issues.

Types of Issues:

Some types of issues to look out for, from highest to lowest priority/severity:

1. Functional Issues
 - a. Incorrect/Unexpected results
 - b. Syntax Errors
2. Logical Issues
3. Code Design/Redundant Code
4. Error/IO Handling (e.g. not sanitizing inputs/outputs)
5. Performance Issues (execution/loading takes too long)
6. Code-style and Documentation

Rules and Guidelines:

1. Every reviewer should review code that they have not written to provide a fresh look over the work
2. Review a maximum of 400 lines of code (LOC) in roughly an hour, do not work for longer than that in a given session
3. Take breaks between code review sessions
4. The reviewer must create a document to keep track of progress during code review
 - If an issue is found, the reviewer must first discuss the issue with the developer to understand why it is an issue and how it can be solved
 - Details including which file and specific section of the code causing the issue and must be recorded in the reviewers document along with what the issue is and a possible solution if one was discussed
 - A possible reviewer document template:

Section of Code	Issue	Possible Solution
Problem.java, getProblemID(), Lines 35-63,	Returns one less than the actual problem ID	Add one to return value

Code Review Delegation:

File	Reviewer
ApplicationDriver.java	Felix
AssignmentCreator.java	Olivia
Authenticator.java	Felix
ExtractData.java	Felix
Grading.java	Tito
Problem.java	Olivia
AssignmentCompletionGUI.java	Abhay
AssignmentCreationGUI.java	Olivia
AssignmentEditingGUI.java	Tito
DisplayMarksGUI.java	Olivia
InstructorListingGUI.java	Tito
InstructorRemarkGUI.java	Julian
LoginRegisterGUI.java	Felix
RegisterInstructorInfo.java	Abhay
RegisterStudentInfo.java	Abhay
RemarkRequestGUI.java	Felix
StudentListingGUI.java	Julian
StudentSubmissionDetailsGUI.java	Julian