

Assigned: October 29, 2013

Deliverable Due Dates: **November 12, December 03, 06**

File Names: **TX** should be replaced with T1 or T2 for your team. **CY** should be replaced with C1/C2 for the number of the change request you are implementing. **P2** means phase 2. Dropbox submission is at <http://dropitto.me/verdicchio> and the password is fall13.

List of deliverables: Beware, if your change does not compile/run it will receive zero (0) points

No.	What	Where/When	Description	pts
1	Electronic copy of Concept Location & Impact Analysis report	Dropbox November 12 th	You are required to submit an electronic copy of the report describing how you used the incremental change process to implement the change request, in the format defined by <i>report_clia.doc</i> .	25
			File Name: 602-TX-PX-CY-CLIA-Lastname.doc	
2	Code Reviews	Dropbox December 3 rd	Each student will perform a code review for two other teammates based on an initial working copy of their change. Choose partners so everyone is reviewed only twice. It is your team's job to navigate the Thanksgiving Holiday. You will not be penalized if your teammate's code is not ready to review. You will be penalized if your code is not ready to review by December 1 st . Follow instructions in provided documents.	25
			File Name: 602-TX-PX-CODEREVIEW-YYY-for-ZZZ.docx	
3	Final Report	Dropbox December 6 th	You are required to submit an electronic copy of the report describing how you used the incremental change process to implement the change request, in the format defined by <i>report_format.doc</i> . The report should also contain the fragments of the source code files that were modified/added, with the code highlighted as specified in the <i>report-format.doc</i> file.	25
			File Name: 602-TX-PX-CY-REPORT-Lastname.doc	
4	Team Demo	In Class December 3 rd	2-3 group members will do a demonstration of the team project. (Each student must give one demonstration during the semester or face a 25 point deduction.) Since this date is before the due date for the change request, you may present a preliminary working version of the change.	0
5	SVN Commit	Your team's SVN server December 10 th	You are required to correctly commit your source code to your team's SVN repository. Committing to a wrong SVN repository or ruining the repository due to incorrect operations will result in a failing grade (0 points) for the phase. Make sure you resolve any conflicts prior to committing.	75
6	Source code files	Dropbox December 10 th	Submit one archive per team (.rar, .zip, .7z or .tar.gz) containing the entire folder of your local copy of JEdit, with all team changes implemented. This code should match the code in your team's SVN repository.	0
			602-TX-PX-SRC.rar	
7	Peer Assessment	Dropbox December 10 th	Submit Peer Assessment Form.xlsx to the dropbox. Penalty if not submitted.	0
			602-TX-PX-PEER-Lastname.xlsx	

More information on the next page...

Change Request

A list of change requests can be found on the next page. The team will decide which team members will be responsible for which change. If there are more change requests than team members, the team can decide which change(s) will not be implemented.

Collaboration

Each member of the group is responsible for their assigned change. You may ask questions of your fellow group members, but you must implement the change on your own. You may not collaborate with members of other groups, and you may not access other group's SVN repositories. If a team member helps you get started, credit that person in your report in the appropriate section.

Programming Environment

Eclipse and Subversion are required for the project to facilitate a uniform standard for teams and instructor support. Teams are required to set up an initial source commit to their subversion repository and host their version of JEdit there. This will involve one team member downloading the JEdit archive, creating the Eclipse project locally, and then making the initial commit to a subversion repository. The other team members should then check out the project from the team SVN repository through Eclipse and commit changes there. You have been added to a project on our repositoryhosting.com site.

Late policy

Because this phase runs to the end of the semester, **no late work will be accepted**. If a group member does not have his or her change completed in time, they must commit what they do have. The project must still compile and run even if certain changes were not completed, and any functional limitations due to this must be clearly documented in the code comments and in the final report. The team will not be punished for one member not completing a change.

Team Evaluation

At the conclusion of each phase of the project each team member will be asked to report feedback on the team experience. Team feedback will be factored into individual grades.

Thanksgiving

This phase includes the Thanksgiving holiday. As such, all the due dates are back-loaded. It is the job of the team to proactively plan for this and establish cooperation norms, especially for code reviews. Everyone has different plans, so work something out in advance. Do not expect someone to be available during the break, but if people are willing to work certain times, that is fine. Just make sure everyone is on the same page ahead of time so that all deliverables are on time. Do not expect to be able to slap everything together right after you return from the break.

Change requests are found on the next page...

Phase 2 Change Requests for JEdit

In case of any questions about the change request, consult the instructor as soon as possible. It is very important that you understand the task correctly.

You may also implement any change requests from Phase 1 that were not assigned to a team member. If you do this, assign the change request a number following the numbers in this list.

1. **“Signature”**. Allow the user to specify a signature to be used as the footer in all printed documents. An option should be available to enable/disable the signature. When the option is enabled, the signature will appear in the status bar.
2. **“Edit remote files”**. The user can indicate the URL of the file. JEdit retrieves the document to local machine for modification, and then puts it back to the location named by the URL to overwrite the original one. Either HTTP or FTP should be supported. This would be a good task for someone with access to a simple web server with FTP access (use get/put).
3. **“Simulate notepad appearance”**. Draw horizontal black lines, which separate continuous lines. The appearance is like paper in a notebook.
4. **“Record the typing speed of the user”**. Record how many characters and words the user types in this session and show how fast he/she is. Use words per minute to measure the speed. The information will be shown at the status bar.
5. **“vi-style input”**. VI is a text editor for the Unix command line. The procedure is: 1) double press key “ESC”; 2) input a number X; 3) type in something; 4) press “ESC” again, the sentence which you just typed will be inserted X times at the current position. Any key sequences not following the procedure exactly will not invoke this behavior.
6. **“Open read only”**. Allow the user to open the file in read-only mode. All the features except editing should be available.
7. (you may implement any unassigned change requests from the previous phase)