Deliverable 4
Progress Report
Team 10: <**Rojan**, **Jenny**, **Ted**, **Ethan**>
Repository: https://github.com/Taguah/SoftwareEnggProject/tree/Production

Member	User Name	Contributions
Rojan Omidvar	RojanOmidvar	Implemented GUI (which contains getting input file, set them and call other classes in order to get output). Generating text files for master course list. Wrote Junit tests in order to test some of the functionality of the system.
Jenny Wang	ShuaijinWang	Created classes and methods used to read and store information from a folder of transcripts, and a grade schema (all txt files). Joined the dataset from normal courses and retaken courses. Set up packaging for the project to bundle all class files into a jar and include the required libraries on the file path. Debugged and altered the GUI and ExcelParser classes.
Edward Camp	Ecamp	Added new methods to detect retaken courses and methods to implement this data in the distribution classes. Periodically cleaned and rewrote methods as they were replaced or became obsolete. Wrote test cases to demonstrate full functionality of TranscriptList and Distribution for use in the excel writer. Debugged extensively.
Ethan Heskett eheskett@unb.ca	Taguah	Split the ExcelParser class into ExcelWriter and ExcelReader to conform to the single responsibility principle, altering pieces where necessary Provided error handling for the GUI and Excel

		classes

Progress Report:

The major change to the functionality of the non-excel and non-GUI classes was adjusting for retaken courses. This was done by using a group of maps keyed to each course name in the transcript class to track courses passed and failed as well as a list of retaken courses. This required rewriting some methods that had previously been expecting lists rather than maps. The new system, while it looks a little more complicated, tracks retaken courses much effectively and the output methods in Distribution and TranscriptList didn't need to be changed to adapt to them. Since the GUI and reader/writer use those classes exclusively, they also didn't need to be changed - a good example of open/closed design.