Tasks

Tasks	Stories	Related Documents/Notes
☒1 Design the format of grade files	S1, S2, S3, S4	The specifications for `.grd` files
☒2 Identify the signature for a method for reading a single category of grades	S1, S2, S3, S4	The specification for the `readCategory()` method in the UML class diagram
☒3 Design a class that can contain all of the relevant information about a category	S1, S2, S3, S4	The UML class diagram and specifications for the `Category` class
☐4 Create unit tests for the class in task 3	S1, S2, S3, S4	
☐5 Test and debug the class in task 3	S1, S2, S3, S4	
☒6 Identify the signature for a method for reading a all of the categories in a course	S1, S2, S3, S4	The specification for the `readCourse()` method in the UML class diagram
☒7 Design a class that can contain all of the relevant information about the categories in a course	S1, S2, S3, S4	The UML class diagram and specifications for the `Course` class
☐8 Create unit tests for the class in task 7	S1, S2, S3, S4	
☐9 Test and debug the class in task 8	S1, S2, S3, S4	
☒10 Organize the methods in tasks 2 and 6 into a class	S1, S2, S3, S4	The specifications for the `CourseReader` class
☒11 Implement the code that uses command line arguments	S5	The `Gradient` class
☒12 Implement the code that calculates and displays the course grade		The `Gradient` class
☒13 Create test file for course with five categories, one of which drops the lowest and one of which drops the highest and lowest	S1	`five-categories_01.grd`

☒14 Create test file for course with six categories with no dropped grades	S1	`six-categories_01.grd`

☒14 Create test files for system tests for previous version	S2	`original_01.grd`
		`original_02.grd`
		`original_03.grd`
		`missing_all.grd`
		`missing_one-one-one.grd`
		`missing_one-in-each.grd`
		`missing_multiple-in-each.grd`

☐15 Create Eclipse "Run Configurations" for each test	S5	
☐16 Test the system	S1-S5	
☐17 Debug the system (if necessary)	S1-S5	