

Sprint 5 Plan

Week of Nov. 13, 2017

User Stories	Tasks	Story Points	Dependencies
U10	T18	2	-
U11	T19	2	-
U11	T20	3	T19
U12	T21	2	-
U13	T22	1	-
U13	T23	1	-
U14	T24	2	-
U14	T25	1	T24
U15	T26	1	-
U15	T27	3	-
U16	T28	2	-
U17	T29	2	T28

Provisional Burndown Chart

	Task Points	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
T18	2	T:1	T:1					
T19	2	F:2						
T20	3				J:3			
T21	2	A:2						
T22	1	O:1						
T23	1			T:1				
T24	2		O:2					
T25	1			O:1				
T26	1		J:1					
T27	3			J:3				
T28	2				A:2			
T29	2					O:2		

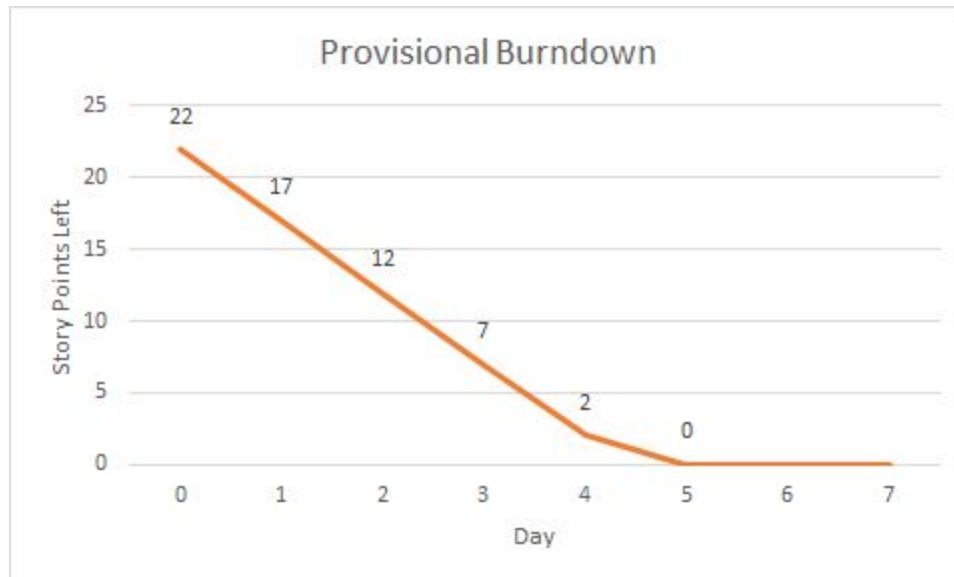
J - Julian [story points per day = 3]

A - Abhay [story points per day = 3]

F - Felix [story points per day = 3]

T - Tito [story points per day = 2/3]

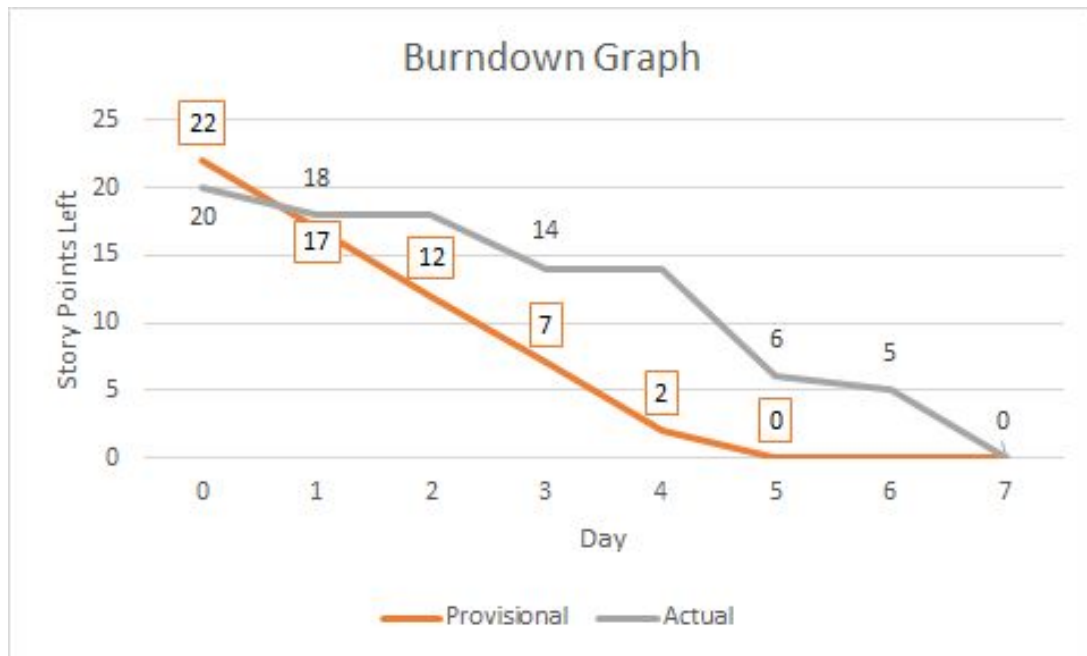
O - Olivia [story points per day = 2]



Final Burndown Chart

	Task Points	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
T18	2					T:1		
T19	2	F:2						
T20	3							J:3
T21	2					A:2		
T22	1			O:1				
T23	1						T:1	
T24	2			O:2				
T25	1			O:1				
T26	1					J:1		
T27	3					J:3		
T28	2					A:2		
T29	2							O:2

J - Julian [story points per day = 4]
A - Abhay [story points per day = 3]
F - Felix [story points per day = 3]
T - Tito [story points per day = 2/3]
O - Olivia [story points per day = 4]



Backlog

U10: As Dana (a student), I would like to see which questions on an assignment I got right and wrong after a submission so that I can retry the assignment if the deadline has not passed.

T18: Create a “submission summary” interface that is shown after a student hits the “submit” button which displays all the questions and a check mark or cross denoting whether it was answered correctly or not.

U11: As Karen/Ben (a statistics professor), I want to be able to view a student's grades for all closed assignments that they have submitted.

T19: Create a basic command-line interface for an instructor to enter a student's id and print all assignments with their corresponding grades by parsing through each of the submissions files (assignment#_submissions.csv) and locating the student's entry.

T20: Create a GUI for displaying the student's assignment marks.

U12: As Dana (a student), I would like to receive detailed feedback about my assignment submission after the deadline has passed regarding how much time I spent on the assignment, how many tries I took, and the average mark of all my completed attempts.

T21: Create button on the interface that lists all completed/closed assignments beside each listing to return the logged-in student's time spent, tries attempted and average mark of all attempts from the (assignment#_submissions.csv).

U13: As Ben (a statistics professor), I would like to see the grade distribution of an assignment over the student body after the deadline has passed.

T22: Create function to compute average of an assignment by parsing through the submissions file (assignment#_submissions.csv) and using the final mark column to compute the mean.

T23: Add the text UI element to display the course average beside the assignment listing.

U14: As Karen (a statistics professor), I would like the application to generate a spreadsheet that I can download of all student marks for a specific assignment.

T24: Implement UI that lists all assignments that are closed by parsing through all assignment files in the directory.

T25: Create button beside a closed instructor's assignment listing to export the student's marks to a location the instructor desires (essentially a 'save as' menu that saves the (assignment#_submissions.csv for that particular assignment to the location the instructor chooses).

U15: As Dana (a student), I would like to be able to submit a remark request with an explanation after the assignment deadline has passed so that my grade can be reevaluated.

T26: Lists student assignments based on whether they are closed or open.

T27: Create a "Remark" button on the assignment listing, and enable it after the deadline for the assignment has passed. When pressed, a message box is prompted to the screen and the message is sent to the Instructor's email.

U17: As Jenny (a TA), I would like to be able to manually adjust a student's assignment submission mark after they have submitted a remark request.

T28: Create an interface with text fields for a student number, assignment #, and new grade with a save button that saves to the assignment#_submissions.csv file for that specific student. [Also create button on instructor listing screen to take the instructor to this new interface]

U17: As Dana (student), I would like to be notified via email when my remark request has been processed with information regarding the instructor's decision.

T29: When the "Remark" button is pressed on the remark interface, send an email with the assignment # and final grade to the student.

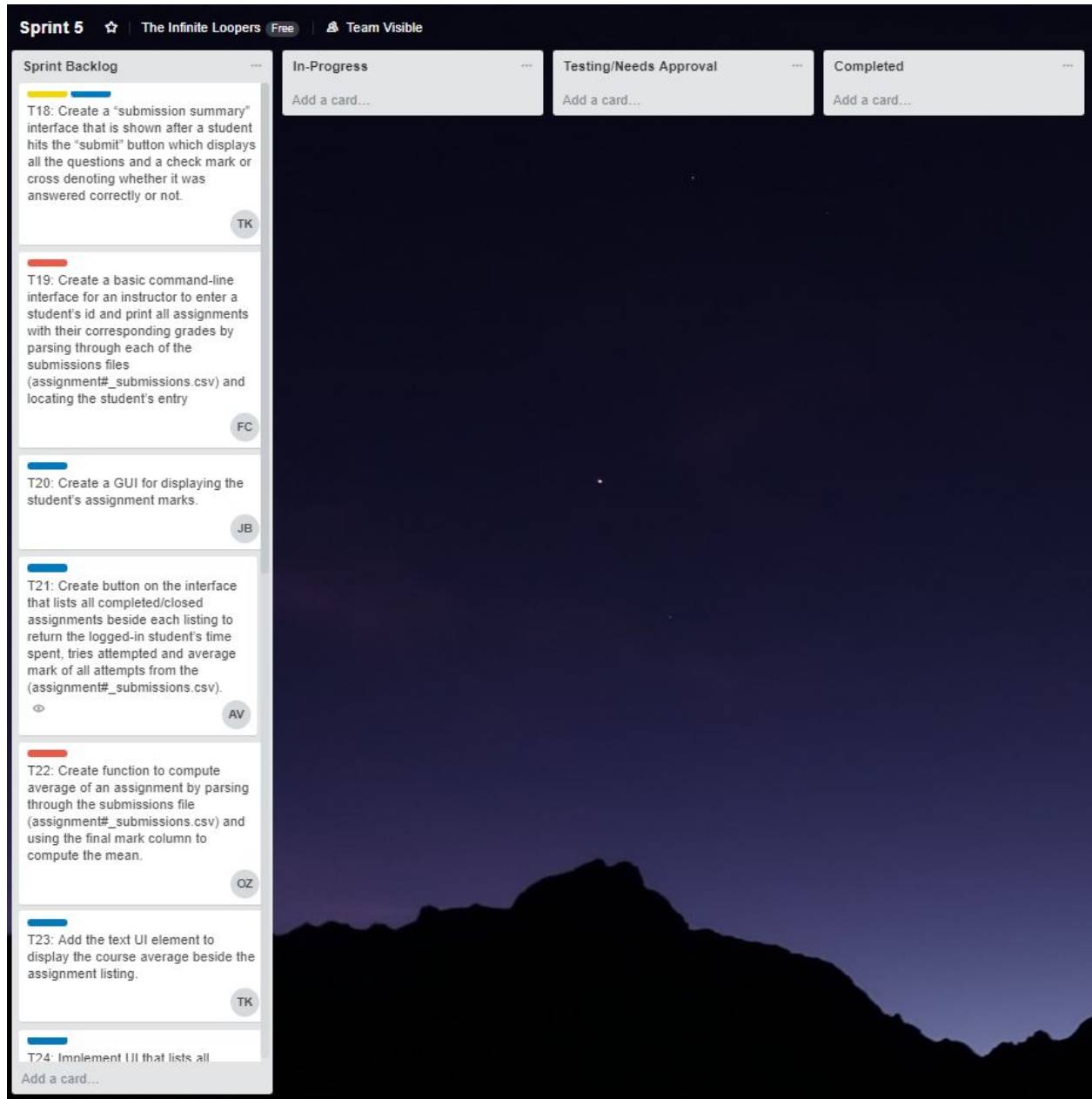
Report

- Julian made the InstructorListingGUI refresh after a an instructor user edits an assignment.
- When a pull request was made to merge AssignmentEditGUI, Tito and Abhay suggested that Julian use a WindowAdapter instead of WindowListener to prevent having dead code, and to notify the instructor user that an assignment is set to "Unreleased" after editing. Thanks to this feedback, Julian successfully incorporated the AssignmentEditingGUI class into InstructorListingGUI class.
- Julian added a warning icon next to the warning message that an assignment is set to "Unreleased" after editing.
- Julian added a scroll bar to InstructorListingGUI.
- A minor functionality issue was found in AssignmentEditGUI and Julian applied the fix which was: users are no longer allowed to rename the assignment the same name of an existing assignment.
- Julian added implementation to StudentListingGUI class to list closed assignments.
- Julian created class RemarkRequestGUI; when the submit button is pressed, an email is sent to every registered instructor containing a reason for remark message and the assignment details (questions with associated submitted answers). RemarkRequestGUI.java was merged to master upon completion with approval of team members.
- Julian wrote the DisplayMarksGUI class; a student number is entered in the text field, and when the search button is pressed, all the marks of that particular student is listed below.
- Julian modified Grading.java to return a HashMap<Integer, String> to associate the keys as the assignment number and the values as the student's mark for that assignment.
- The display marks feature was merged to master upon completion with approval of team members.

- Julian detected and resolved a bug in InstructorListingGUI.java and StudentListingGUI.java; a calendar object was being used and two arguments to the set() method were not the appropriate inputs. As a consequence, the assignments were not being listed as “closed”.
- Julian and Abhay set the average of assignments in InstructorListingGUI to be 0.0% instead of “NaN” if the Assignment#.csv file does not have any submissions from students.
- Julian wrote the ExtractDataTest test case file and implemented every test.
- Olivia fixed a bug in updateCsvFile() method from previous sprint
- Olivia Made an open button for each released assignment in student assignment listing
- Olivia took off the autosave feature when students and change the way of saving student answers to clicking the save button
- Olivia linked student listing to login panel
- Olivia added a feature to take in students and instructors personal information when they are registering their account
- Olivia wrote a function to get the average mark of an assignment by parsing through each student’s final mark in the submission file
- Olivia implemented UI that lists all assignments that are closed by parsing through all assignment files in the directory.
- Olivia created button beside a closed instructor’s assignment listing to export the student’s marks to a location the instructor desires (essentially a ‘save as’ menu that saves the (assignment#_submissions.csv for that particular assignment to the location the instructor chooses)
- Olivia added a feature that when the “Adjust mark” button is pressed on the remark interface, the program will send an email with the assignment # and final grade to the student.
- Olivia implemented unit test cases to test the method isInstructor() and authenticate() methods in the Authenticator class
- Tito added the ability for a professor to view all closed assignments’ average mark
- Tito continued to work on submit summary locally and encountered indexing issues (currently still unfinished)
- Tito wrote part of AutheticatorTest debugging branch titoAutheticatorTest to try and figure out the issue that makes users.csv and usersOriginal.csv to not be handled properly
 - Seem to have figured out issue lies somewhere in autheticate() but has yet to confirm cause of bug

- Abhay created an interface that displays the final mark, average mark, number of tries, and time spent on a given closed assignment
 - The button to view these statistics only shows up on the assignments that the student attempted
- Abhay created an interface for an instructor to change the final mark of a student's assignment submission given a student number, assignment number, and new final mark. (Also added a button to access this interface to the instructor dashboard/listing interface)
- Abhay wrote the AssignmentCreatorTest.java testing class to test assignment creation
 - Noticed that a file stream object had not been closed in AssignmentCreator.java and fixed the issue
- Felix created a Grading.java to get all of a given student's grades for all their assignments
- Some miscellaneous fixes, tweaks, and enhancements that were completed this sprint:
 - Made button in student assignment listing to be able to complete an assignment
 - Added some form of handling (scroll bar) for long text in assignment creation
 - Don't autosave when clicking radio buttons for assignment completion
 - Auto-fill assignment answers when a student opens an assignment they've worked on before
 - Show student dashboard when student logs in
 - Added scrollbar to assignment listing for instructor and student
 - Added extra registration GUI for gathering more details and better authentication
 - Linked dashboard with user's name

Taskboard Snapshots



Before

Sprint 5

☆

The Infinite Loopers

Free

Team Visible

Sprint Backlog

Add a card...

In-Progress

T18: Create a "submission summary" interface that is shown after a student hits the "submit" button which displays all the questions and a check mark or cross denoting whether it was answered correctly or not.

TK

Add a card...

Testing/Needs Approval

Add a card...

Completed

T19: Create a basic command-line interface for an instructor to enter a student's id and print all assignments with their corresponding grades by parsing through each of the submissions files (assignment#_submissions.csv) and locating the student's entry.

FC

T20: Create a GUI for displaying the student's assignment marks.

JB

T21: Create button on the interface that lists all completed/closed assignments beside each listing to return the logged-in student's time spent, tries attempted and average mark of all attempts from the (assignment#_submissions.csv).

AV

T22: Create function to compute average of an assignment by parsing through the submissions file (assignment#_submissions.csv) and using the final mark column to compute the mean.

OZ

T23: Add the text UI element to display the course average beside the assignment listing.

TK

T24: Implement UI that lists all assignments that are closed by parsing through all assignment files in the directory.

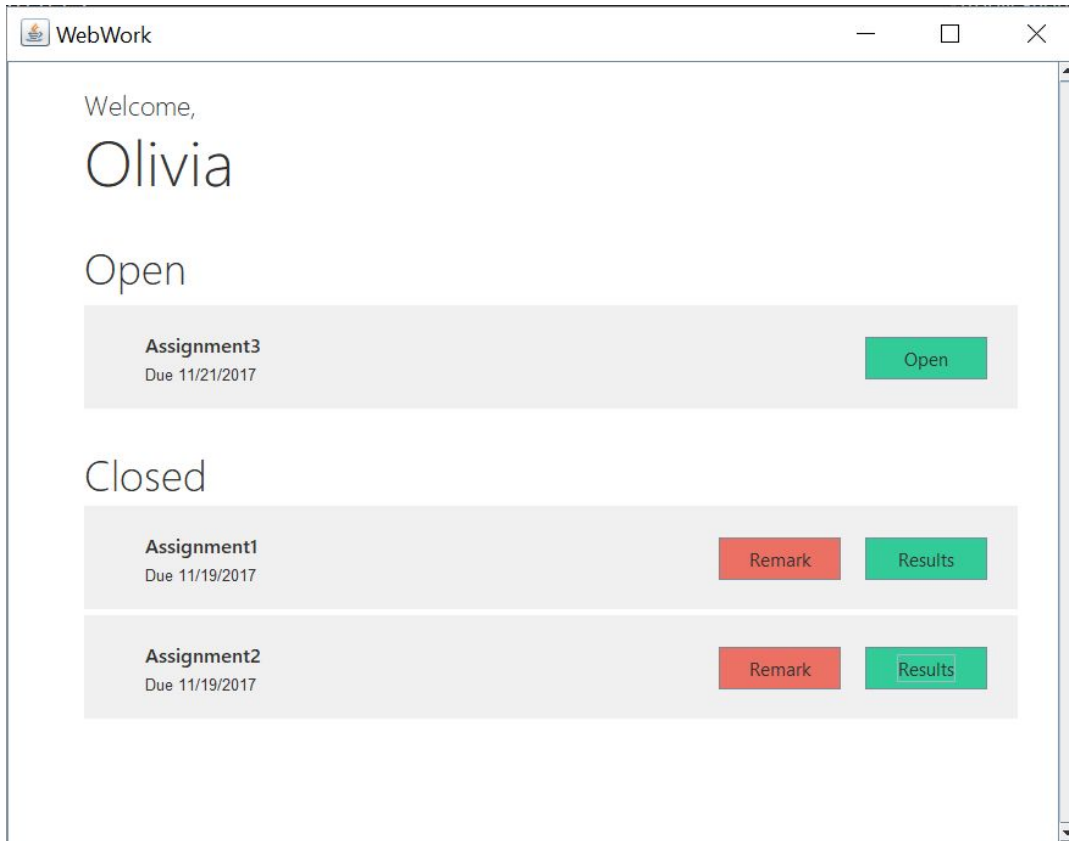
OZ

T25: Create button beside a closed instructor's assignment listing to export the student's marks to a

Add a card...

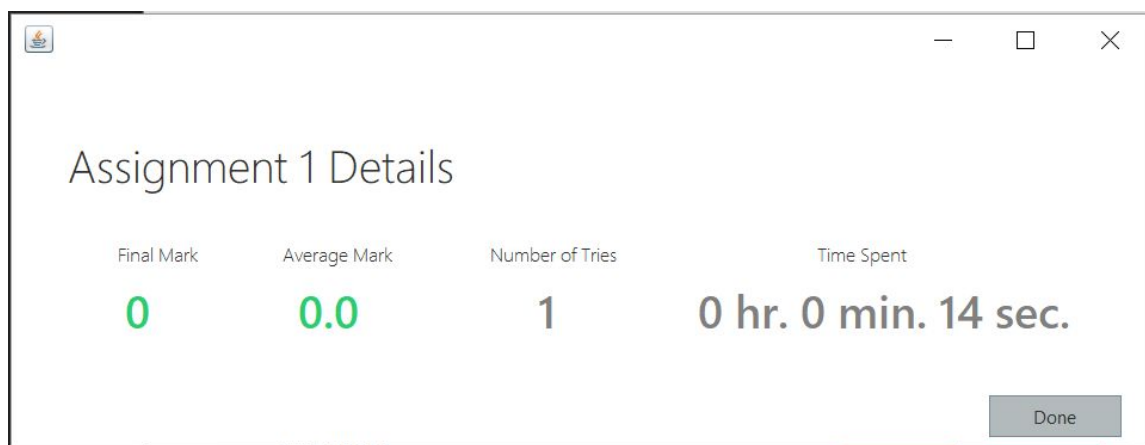
After

Screenshots

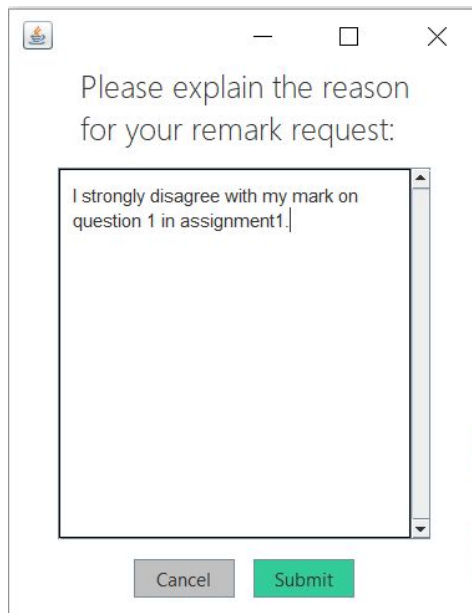


Lists closed assignments for students.

For assignments that are passed the due date, students have the option to submit a remark request and to view their results.



These are the results a student may view on a completed assignment.

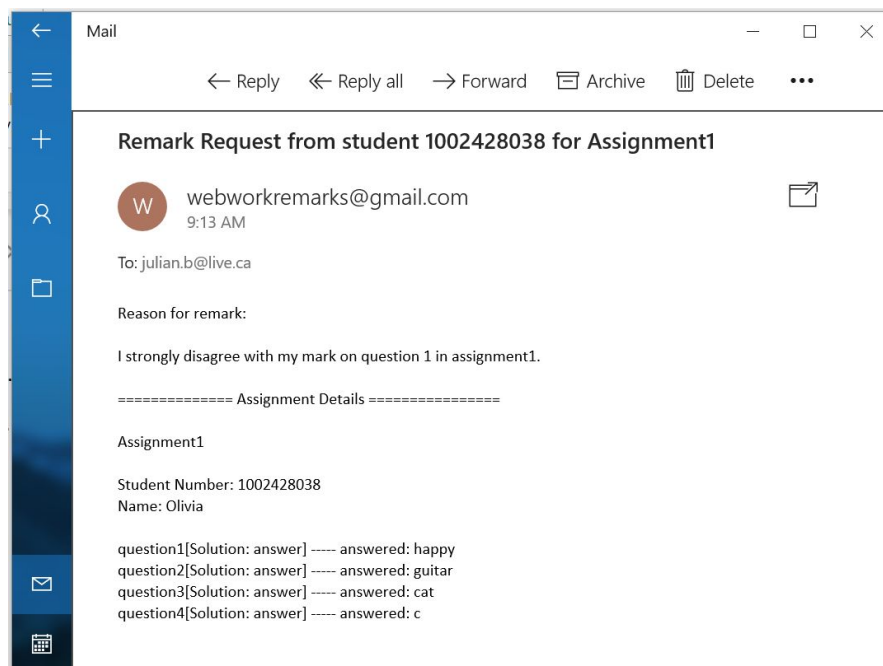


Please explain the reason for your remark request:

I strongly disagree with my mark on question 1 in assignment1.

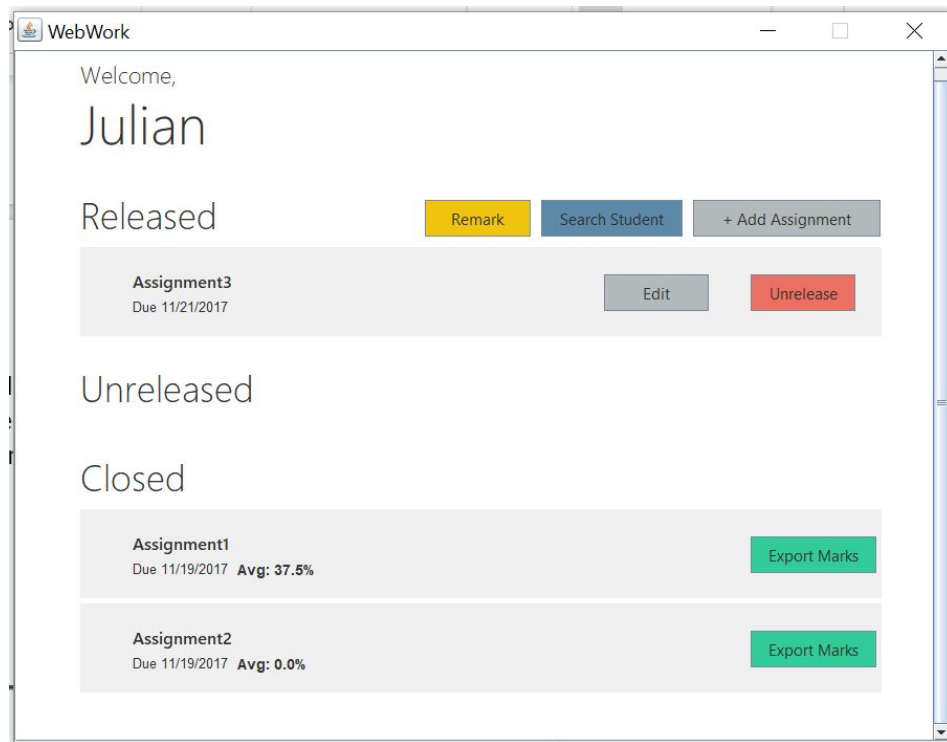
Cancel Submit

Remark submission window.



Once a student submits a remark request, it is sent directly to every instructor's email (professor and TAs).

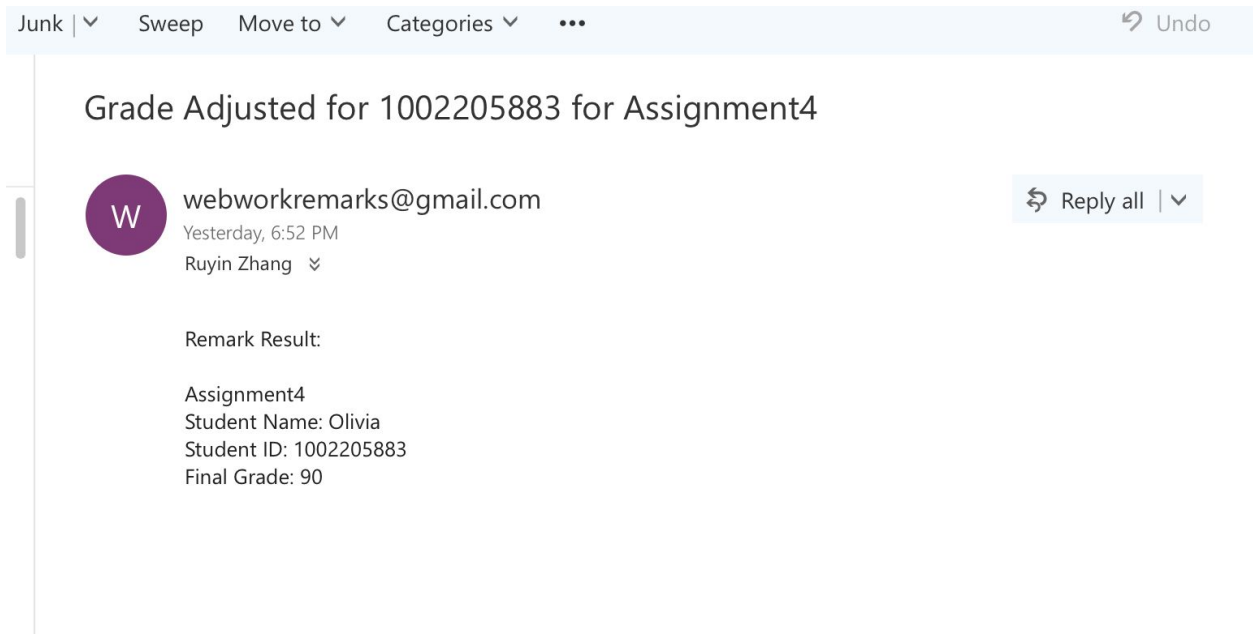
The email includes the student reason for their remark request, and the assignment details such as, which assignment is being submitted for a remark, the student's submitted answers and the actual solution to the questions.



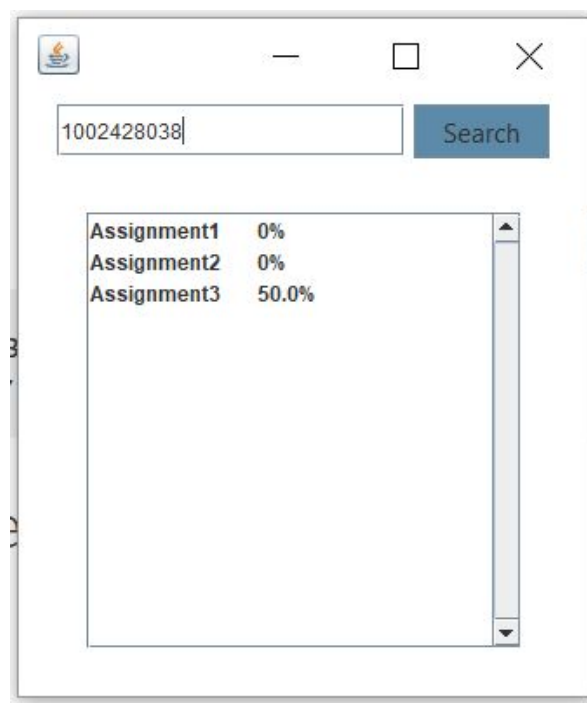
The instructors may view the averages of complete assignments, remark assignments and search a student and retrieve their marks.

The screenshot shows a form titled "Remark Assignment". It contains three input fields: "Student ID" with the value "1002428038", "Assignment #" with the value "1", and "Final Mark" with the value "95". At the bottom of the form, there are two buttons: "Cancel" and "Adjust Mark".

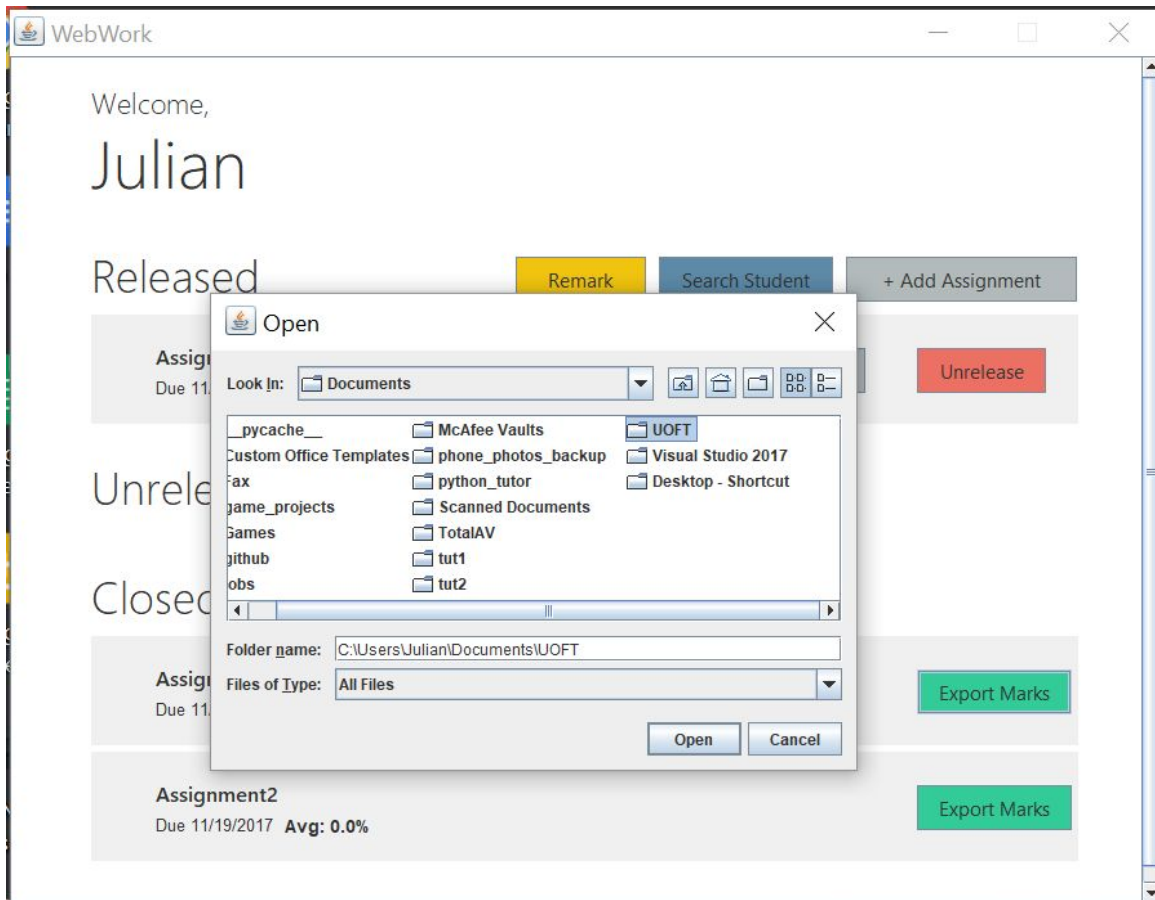
Instructors may alter a student's mark for any existing assignment.



Students get email notification from the application once their marks got adjusted



The student search feature enables any instructor to view a particular student's marks.



The export marks feature enables instructors to download a spreadsheet of the student marks for any closed assignment.

The screenshot shows an Excel spreadsheet titled 'Assignment1Submission - Excel'. The spreadsheet has columns for 'StudentID', 'Question', 'Question', 'Question', 'Question', 'Average N', 'Number o', and 'Time (in s)'. The data row shows '1E+09', 'happy', 'guitar', 'cat', 'c', '0', '1', and '14'. The spreadsheet is opened by 'Julian Barker'.

StudentID	Question	Question	Question	Question	Average N	Number o	Time (in s)	Final Mark
1E+09	happy	guitar	cat	c	0	1	14	0

An assignment submissions spreadsheet after exporting the marks.
(Note: only one student submitted marks for this assignment.)