Product Backlog

Team gitCoffee (L02_09)

Project Deliverable #6

CSCC01

Tony Hong
Saba Kiaei
Lors Kushtov
Stephen Luc
Frank Xu

November 27, 2017

Table of Contents

Personas

Version 0

Persona #1: Sohee Kang

35-year-old Statistics professor, female.

Works at UTSC and teaches multiple courses, with a large number of students in each class.

Very friendly, easygoing and co-operative.

She is extremely busy and her small office is full of students during her office hours, and she is not looking to make it busier.

One of her goals is to increase the engagement of the students for the assignments.

She has been looking for an electronic tool to use for a long time but has not found the perfect too, therefore she is frustrated a little.

Currently uses WeBWorK to post assignments for her Introductory Statistics course

Relatively good computer skills, and comfortable with technology. She is very comfortable with, and prefers LaTeX.

Does not know how to code, but has been able to edit the code and change the numebrs on WeBWork, so far. However, she prefers a simpler interface to change questions.

Persona #2: Alexander Chan

20-year-old Mathematics student, male.

Studies at UTSC.

Is not very organized, and procrastinates most of his work.

He plays video games in his free time, and like challenges.

Speaks Cantonese and English.
He is taking STAB22 because he is interested in statistics. The course is not a requirement for his program.

He likes to do extra credit activities, since he is interested in the subject.

Does most of his assignments during his commute to school.

Uses his phone a lot more than his computer.

Professional computer skills. Has used Microsoft Word, PowerPoint, Excel. He has also worked with Maple and can code.

He is very good at navigating through an application he has never used before.

Persona #3: Louise Monk

19-year old paramedicine student, female.

Is in a joint program with UTSC and Centennial College

A very organized, quiet, and studious person. Likes routines, and following a certain schedule.

Speaks English, and French.

Does most of her studying in her room.

Has to take STAB22 to fulfill a program requirement.

STAB22 is not a course she would be using in her career, so she does not want to spend a lot of time on this course.

Relatively good computer skills: uses applications such as Word, PowerPoint, Publisher, and Photoshop.

Prefers electronic notes rather than paper notes, and she believes technology and electronic aids ease the understanding of the material.

Persona #4: Jake Peralta

23-year-old Statistics major, and Teaching Assistant for STAB22, male.

Studies, and works at UTSC.

A very good TA who takes pride in how well his students do on assignments.

Does most of his work at the school's library.

Takes his job very seriously and is a very detailed marker.

Takes the time to look over the assignments students have done, and bring up common mistakes in Tutorials

Works very closely with the professor and helps her create the questions. Likes doing this and would gladly spend time on it.

Has used assignment tools like WebWorks before but does not like how tedious it is to switch between questions and grade assignments in general.
Prefers grading on paper but does not mind online platforms.

Has worked with LaTeX, Maple, SPSS, and Microsoft Excel.

User Stories

Version 0

1. As Sohee (a professor), I would like to be able to generate text question templates, where the details (mostly variable values) of the questions are randomized for each assignment instance.

2. As Sohee (a professor), I would like to be able to edit the question templates once I am finished by clicking an edit button and using my keyboard afterwards.

3. As Sohee (a professor), I would like to create a collection of question templates that are for a particular assignment, which will be the base for assignment instances.

4. As Sohee (a professor), I would like to be able to create as many accounts as there are students, where each account has a studentID, tutorial number, and a list of assignments.

5. As Louise (a student), I would like to login to my account, using my email, and a personalized password.

6. As Sohee (a professor), I would like to be able to set the start date of each assignment at which point the assignment is visible to the student under the list of their assignments.

7. As Sohee (a professor), I would like to be able to set the due date of the assignments at which point students will still see the assignment in the list of assignments, but are not able to edit it, post due date.

8. As Sohee (a professor), I want each student to be able to start an assignment which is randomly generated based on the assignment template I made.

9. As Alexander (a student), when the assignment is viewable (is under the list of my assignments) I want to be able to start the assignment with the click of a button.

10. As Louise (a student), I would like to submit my assignment by clicking a button.

11. As Alexander (a student), I would like to be able to see the grade of my assignment after I have submitted it.

12. As Sohee (a prof), I would like the students to be able to redo the entire assignment for a set amount of times, where each retry is a different set of questions from what they have already done.

13. As Sohee (a prof), I want the final assignment grade to be the maximum grade the student has achieved from all the previous tries.

14. As Louise (a student), I would like to see a table of every single assignment I have finished, and my final grade for each assignment.

15. As Sohee (a prof), I would like to see a table that contains all of the students and their marks for each assignment.

16. As Sohee (a prof), for each assignment, I would like to see the number of students who have finished the assignment, and the mean and standard deviation of the final grades, in a table.

17. As Sohee (a prof), I would like to create multiple TA accounts, where each TA account has a name, and a tutorial number.

18. As Jake (a TA), I would like to login to my account, using my email, and a personalized password.

19. As Jake (a TA), I would like to see a list of all the students enrolled in my tutorial number.

20. As Jake (a TA), I would like to see all the assignments a student has done by clicking on the student's name.

21. As Jake (a TA), I would like to access the questions and the student's answers in one view for each assignment, by clicking on the assignment number.

Version 1

1. As Sohee (a professor), I would like to be able to generate text question templates, where the details (mostly variable values) of the questions are randomized for each assignment instance.

2. As Sohee (a professor), I would like to have an account where I can save the question templates.

3. As Sohee (a professor), I would like to be able to edit the question templates once I am finished by clicking an edit button and using my keyboard afterwards.

4. As Sohee (a professor), I would like to create a collection of question templates that are for a particular assignment, which will be the base for assignment instances.

5. As Sohee (a professor), I would like to be able to create as many accounts as there are students, where each account has a studentID, tutorial number, and a list of assignments.

6. As Sohee (a professor), I would like to be able to set the start date of each assignment at which point the assignment is visible to the student under the list of their assignments.

7. As Sohee (a professor), I would like to be able to set the due date of the assignments at which point students will still see the assignment in the list of assignments, but are not able to edit it, post due date.

8. As Sohee (a professor), I want each student to be able to start an assignment which is randomly generated based on the assignment template I made.

9. As Alexander (a student), when the assignment is viewable (is under the list of my assignments) I want to be able to start the assignment with the click of a button.

10. As Louise (a student), I would like to submit my assignment by clicking a button.

11. As Alexander (a student), I would like to be able to see the grade of my assignment after I have submitted it.

12. As Sohee (a prof), I would like the students to be able to redo the entire assignment for a set amount of times, where each retry is a different set of questions from what they have already done.

13. As Sohee (a prof), I want the final assignment grade to be the maximum grade the student has achieved from all the previous tries.

14. As Louise (a student), I would like to see a table of every single assignment I have finished, and my final grade for each assignment.

15. As Sohee (a prof), I would like to see a table that contains all of the students and their marks for each assignment.

16. As Sohee (a prof), for each assignment, I would like to see the number of students who have finished the assignment, and the mean and standard deviation of the final grades, in a table.

17. As Sohee (a prof), I would like to create multiple TA accounts, where each TA account has a name, and a tutorial number.

18. As Jake (a TA), I would like to see a list of all the students enrolled in my tutorial number.

19. As Jake (a TA), I would like to see all the assignments a student has done by clicking on the student's name.

20. As Jake (a TA), I would like to access the questions and the student's answers in one view for each assignment, by clicking on the assignment number.

Changes From Previous Version

- Redundant User Stories Deleted.
- A very important user story about the professor account was missing. It was added.

Personas Version Used

- Version 0.

Version 2

1. As Sohee (a professor), I would like to be able to generate LaTeX question templates, where the details (mostly variable values) of the questions are randomized for each assignment instance.

2. As Sohee (a professor), I would like to have three types of question templates: Short answer, multiple choice, and True False by picking the type from a drop down list.

3. As Sohee (a professor), I would like to type in the answer of the questions as a LaTeX formula.

4. As Sohee (a professor), I would like to type in the range of acceptable answers by typing an error margin.

5. As Sohee (a professor), I would like to have an account where I can save the question templates.

6. As Sohee (a professor), I would like to be able to edit the question templates once I am finished by clicking an edit button and using my keyboard afterwards.

7. As Sohee (a professor), I would like to create a collection of question templates that are for a particular assignment, which will be the base for assignment instances.

8. As Sohee (a professor), I would like to be able to create as many accounts as there are students, where each account has a studentID, tutorial number, and a list of assignments.

9. As Sohee (a professor), I would like to be able to set the start date of each assignment at which point the assignment is visible to the student under the list of their assignments.

10. As Sohee (a professor), I would like to be able to set the due date of the assignments at which point students will still see the assignment in the list of assignments, but are not able to edit it, post due date.

11. As Sohee (a professor), I want each student to be able to start an assignment which is randomly generated based on the assignment template I made.

12. As Alexander (a student), when the assignment is viewable (is under the list of my assignments) I want to be able to start the assignment with the click of a button.

13. As Louise (a student), I would like to submit my assignment by clicking a button.

14. As Alexander (a student), I would like to be able to see the grade of my assignment after I have submitted it.

15. As Sohee (a professor), I would like the students to be able to redo the entire assignment for a set amount of times, where each retry is a different set of questions from what they have already done.

16. As Sohee (a professor), I want the final assignment grade to be the maximum grade the student has achieved from all the previous tries.

17. As Louise (a student), I would like to see a table of every single assignment I have finished, and my final grade for each assignment.

18. As Sohee (a professor), I would like to see a table that contains all of the students and their marks for each assignment.

19. As Sohee (a professor), for each assignment, I would like to see the number of students who have finished the assignment, and the mean and standard deviation of the final grades, in a table.

20. As Sohee (a professor), I would like to create multiple TA accounts, where each TA account has a name, and a tutorial number.

21. As Jake (a TA), I would like to see a list of all the students enrolled in my tutorial number.

22. As Jake (a TA), I would like to see all the assignments a student has done by clicking on the student's name.

23. As Jake (a TA), I would like to access the questions and the student's answers in one view for each assignment, by clicking on the assignment number.

Changes From Previous Version
-   New user stories added, introducing new features.

Personas Version Used
-   Version 0.

Tasks

Version 0

Personas: V0
User Stories: V0

User story	Task Number	Story Points	Dependencies	Description
U1	T1	4	-	Design a questionTemplate class with all the appropriate attributes and methods.
	T2	5	T1	Design a simple UI to allow making a question template.
U2	T3	1	T1	Add the appropriate methods to the questionTemplate class so that it becomes editable.
	T4	1	T2	Update the question template UI by adding a button.
U3	T5	4	-	Create an assignment class with appropriate attributes and methods.

	T6	3	T1	Design a method for the assignment class to create question templates and save it in this assignment.
	T7	5	T2	Design a simple UI to allow making an assignment.
U4	T8	1	-	Create a file to save Student data on.
	T9	4	-	Design a Student class with appropriate attributes and methods.
	T10	5	T9	Design a UI to add Students to the file.

Version 1

Personas: V0
User Stories: V1

User story	Task Number	Story Points	Dependencies	Description	Details
U1	T1	3	-	Design a questionTemplate class with all the appropriate attributes and the skeleton of the methods.	-Refer to the UML for the appropriate methods and attributes.
	T2	4	-	Design a simple UI to allow making a question template.	-Have a UI that contains a header, a text box for the question body, text box for the answer to the question, question types, and one button to save.
					-Two underscores for value holder
U2	T3	2	-	Create a prof class with the appropriate attributes and the skeleton of methods.	-Refer to the UML for the appropriate methods and attributes.
	T4	4	T2 & T1	-Sync the backend with the frontend to allow questionTemplate objects to be created.	-Request a post from the frontend to the backend to allow questionTemplate objects to be created
					-Use JS and AJAX

U3	T5	1	T1	Add the appropriate methods to the questionTemplate class so that it becomes editable	-Adding set methods.
	T6	1	T2	Update the question template UI by adding a button.	-Get questionTemplate object, which the user wants to edit, and display the object details.
					-Create a new HTML button and a new JS function.
U4	T7	3	-	Create an assignment class with appropriate attributes and methods.	-Refer to the UML for the appropriate methods and attributes.
	T8	1	T7	Design a method for the assignment class to create question templates and save it in this assignment.	
	T9	4	T2	Design a simple UI to allow making an assignment.	-Create a new HTML page with a button called "Create new Question template" and after clicking this button leads you to the question template page.
					-Display created questions.

User story	Task Number	Story Points	Dependencies	Description	Details
U5	T10	3	-	-Design a Student class with appropriate attributes and methods.	-Refer to the UML for the appropriate methods and attributes.
	T11	2	T3 & T10	-Add a method to professor class to read data from a file and create student objects	-Method would create student objects and save their object details in a JSON file.
	T12	3	T11	-Design a UI to import a file consisting of the student data.	-Allow user to upload a local file of student data.

Changes From The Previous Version

- Tasks were very general and needed to be broken down.
- Tasks did not have enough details. Details were added.

Version 2

Personas: V0
User Stories: V1

User story	Task Number	Story Points	Dependencies	Description	Details
U1	T1	3	-	Design a QuestionTemplate class with all the appropriate attributes and the skeleton of the methods.	-Refer to the UML for the appropriate methods and attributes.

	T2	10	-	Design a simple UI to allow making a question template.	-Have a UI that contains a header, a text box for the question body, text box for the answer to the question, question types, and one button to save.
					-Two underscores for value holder
U2	T3	2	-	Create a prof class with the appropriate attributes and the skeleton of methods.	-Refer to the UML for the appropriate methods and attributes.
	T4	4	T2 & T1	-Sync the backend with the frontend to allow QuestionTemplate objects to be created.	-Request a post from the frontend to the backend to allow QuestionTemplate objects to be created
					-Use JS and AJAX
U3	T5	1	T1	Add the appropriate methods to the QuestionTemplate class so that it becomes editable	-Adding set methods.
	T6	1	T2	Update the question template UI by adding a button.	-Get QuestionTemplate object, which the user wants to edit, and display the object details.
					-Create a new HTML button and a new JS function.

U4	T7	3	-	Create an assignment class with appropriate attributes and methods.	-Refer to the UML for the appropriate methods and attributes.
	T8	1	T7	Design a method for the assignment class to create question templates and save it in this assignment.	
	T9	4	T2	Design a simple UI to allow making an assignment.	-Create a new HTML page with a button called "Create new Question template" and after clicking this button leads you to the question template page.
					-Display created questions.
U5	T10	3	-	-Design a Student class with appropriate attributes and methods.	-Refer to the UML for the appropriate methods and attributes.
	T11	2	T10	-Student UI	
	T12	2	T3 & T10	-Add a method to professor class to read data from a file and create student objects	-Method would create student objects and save their object details in a JSON file.
	T13	3	T12	-Design a UI to import a file consisting of the student data.	-Allow user to upload a local file of student data.

U6	T14	1	T7	-Update the Assignment class to have a start date	
	T15	1	T9	-Create a UI to add a start date to the assignment	
	T16	1	T10	-Update the student class to show the assignment after that date.	
U7	T17	1	T2	-Updating the UI for making the assignment to allow inputting the due date.	
	T18	1	T7	-Update the assignment class.	-Make the assignment viewable but not editable after the due date.
U8	T19	7		-Set Up database for saving assignment templates to prof account	-Postgre SQL database
	T20	1	T1	Update the QuestionTemplate class to generate QuestionTemplates with random values.	
	T21	2	T7	-Update the assignment class to generate a random question for each of its QuestionTemplates and save it	-JSON File

				into a file.	
U9	T22	1	T10	- Add appropriate method to student class	- A method that will call the function in the Assignment Class to generate an instance of the assignment
	T23	1	T11	-Frontend: add a button to student UI to start assignment.	
U10	T24	1	T10	-Add appropriate methods to student class to submit an assignment.	
	T25	1	T11	-Frontend student adding button to submit assignment	
U11	T26	1	T7	-Add self marking method to assignment class	
	T27	1	T10	-Add GetAssignment Grade method to student class	
	T28	1	T9	-Create a UI for grade of assignment	
U12	T29	1	T11	-Add a redo button to the UI of Student for assignment.	
	T30	1	T10	-Add a redo function to the assignment class.	-Redo function creates a new instance of assignment.

User story	Task Number	Story Points	Dependencies	Description	Details
U13	T31	1	T26	-Update GetAssignment Grade method so it stores the max grade	

Changes From The Previous Version

- More user stories were broken down into tasks.

Version 3

Personas: V0
User Stories: V2

User story	Task Number	Story Points	Dependencies	Description	Details
U1	T1	3	-	Design a QuestionTemplate class with all the appropriate attributes and the skeleton of the methods.	-Refer to the UML for the appropriate methods and attributes.
	T2	3	T1	Update the QuestionTemplate to parse a LaTeX question.	-Extract variables
	T3	10	-	Design a simple UI to allow making a question template.	-Have a UI that contains a header, a text box for the question body, text box for the answer to the question, question types, and one button to save.

21

					-Two underscores for value holder
	TU1	1	T1 to T3	Test QuestionTemplate Class	-Unit tests.
U2	T4	3	T1	Create a python class for short answer question template.	-Using strategy design pattern.
	T5	1	T1	Create a python class for TF answer question template.	-Using strategy design pattern.
	T6	1	T1	Create a python class for multiple choice question template.	-Using strategy design pattern.
	T7	1	T3	Create a drop down list in UI to pick the question type.	
	T8	2	T6	Create appropriate fields for each question type in UI	
U3	T9	3	T2	Update the backend to parse a LaTeX formula	
U4	T10	1	T1	Update the backend to store the error margin.	

	T11	1	T3	Add a textbox to the UI to get the input for error margin.	
U5	T12	2	-	Create a prof class with the appropriate attributes and the skeleton of methods.	-Refer to the UML for the appropriate methods and attributes.
	T13	4	T3 & T1	-Sync the backend with the frontend to allow QuestionTemplate objects to be created.	-Request a post from the frontend to the backend to allow QuestionTemplate objects to be created
					-Use JS and AJAX
	TU5	1	T12 & T13	Test Professor class.	Unit tests
U6	T14	1	T1	Add the appropriate methods to the QuestionTemplate class so that it becomes editable	-Adding set methods.
	T15	1	T3	Update the question template UI by adding a button.	-Get QuestionTemplate object, which the user wants to edit, and display the object details.
					-Create a new HTML button and a new JS function.

U7	T16	3	-	Create an assignment class with appropriate attributes and methods.	-Refer to the UML for the appropriate methods and attributes.
	T17	1	T16	Design a method for the assignment class to create question templates and save it in this assignment.	
	T18	4	T3	Design a simple UI to allow making an assignment.	-Create a new HTML page with a button called "Create new Question template" and after clicking this button leads you to the question template page.
					-Display created questions.
	TU7	1	T16 to T18	Test Assignment class	-Unit test.
U8	T19	3	-	-Design a Student class with appropriate attributes and methods.	-Refer to the UML for the appropriate methods and attributes.
	T20	2	T19	-Student UI	
	T21	2	T12 & T19	-Add a method to professor class to read data from a file and create student objects	-Method would create student objects and save their object details in a JSON file.

	T22	3	T21	-Design a UI to import a file consisting of the student data.	-Allow user to upload a local file of student data.
	TU8	1	T19 to T22	Test Student Class	Unit test
U9	T23	1	T16	-Update the Assignment class to have a start date	
	T24	1	T18	-Create a UI to add a start date to the assignment	
	T25	1	T19	-Update the student class to show the assignment after that date.	
U10	T26	1	T3	-Updating the UI for making the assignment to allow inputting the due date.	
	T27	1	T16	-Update the assignment class.	-Make the assignment viewable but not editable after the due date.
U11	T28	7		-Set Up database for saving assignment templates to prof account	-Postgre SQL database
	T29	1	T1	Update the QuestionTemplate class to generate QuestionTemplates with random values.	

	T30	2	T15	-Update the assignment class to generate a random question for each of its QuestionTemplates and save it into a file.	-JSON File
U12	T31	1	T19	- Add appropriate method to student class	- A method that will call the function in the Assignment Class to generate an instance of the assignment
	T32	1	T21	-Frontend: add a button to student UI to start assignment.	
U13	T33	1	T19	-Add appropriate methods to student class to submit an assignment.	
	T34	1	T21	-Frontend student adding button to submit assignment	
U14	T35	1	T16	-Add self marking method to assignment class	
	T36	1	T19	-Add GetAssignment Grade method to student class	
	T37	1	T18	-Create a UI for grade of assignment	
U15	T38	1	T21	-Add a redo button to the UI of Student for assignment.	

	T39	1	T19	-Add a redo function to the assignment class.	-Redo function creates a new instance of assignment.
U16	T40	1	T35	-Update GetAssignment Grade method so it stores the max grade	
	TU16	1	T40	Test Student Grade	-Unit test.

Changes From The Previous Version

- The new user stories were broken down into tasks.