Acceptance Tests:

Given	When	Then
A player enters the correct answer	On the question screen	The correct number of points will be added
A player has the most points	At the end of the game	The correct winner will be displayed
A question has already been answered	On the gameboard	The same question can not be selected
A player enters an incorrect answer	On the question screen	The correct number of points will not be added
The use enters more than 1 player	On the number of players selection screen	The game will iterate through each player’s turn

Final Report:

Basic description

- Trivia game with pass and play components. There will be multiple categories of multiple types of questions, with several questions per category. The questions will get harder as the point values increase, and will consist of multiple choice and true false questions. The questions themselves, as well as their answers will be pulled from multiple text files. The categories are yet to be determined, and will be chosen by each group member individually.

Majors changes / alterations / things dropped from final release

- There are no major alterations or changes from the original project plan.

User stories, Estimation of difficulty, Estimation of time

Estimation of difficulty, Estimation of time, Pair worked on/plan to work on. (Equal difficulty for all)

1. Create main menu
   a. Start menu: start game, settings, saved files
   b. Game over menu: who won, play again, etc
2. Multiplayer
   a. Make the game turn based/sequential by submitting answers or having a time limit to answer the question, then letting another player repeat the process.
3. Question Selection
   a. Buttons to select questions in each category, as well as which level of difficulty for the questions.
4. Categories (Low difficulty)
   a. Make different categories for questions.
5. File Reading
   a. Read from a file to display player information, specifically reading the questions and answers from a text file.
6. Final question/endgame
   a. Final question that all players answer, potentially multiplying the score of or giving a chance at a comeback to losing players. Will be more difficult and appear after all other questions are exhausted.
7. Scoring
   a. Values will be assigned to questions based on difficulty, and the player will either gain or lose the appropriate amount of points based on whether that get the right or wrong answer.
8. Question types
   a. Will support true/false and multiple choice questions, will read both from differing text files

Accomplishments of the Project

- In this project, a trivia game was made using software engineering tools such as unit testing, acceptance testing, and refactoring. Trivia questions were written in a text file and read into “Trivia.py”.

Three examples of Code Refactoring

- Changed player display to initialize from “Player 0” to “Player 1” | lines 332, 334 in Trivia.py
- Original Trivia.py code was refactored to work with question_reader.py, as well as to add question answering and player scoring functionalities.
- Added width, line skip, and x- and y-skip parameters to drawTextCentered() function to facilitate more precise placement of text as well as properly format multi-line question strings.

Pair programming chart

		Names								Expected Time (hours)	Actual Time (hours)
		Audrey	Alyssa	Daniel	Garrett	Grey	Nick	Terry	Tyson		
Tasks Ordered by Priority	Menus	---	---	---						5	4
	Multiplayer						—-		—-	5	3
	Question selection				---	---	—-			5	3
	Categories			---			—-	---		5	5
	File reading				---	---				5	3
	Final question/endgame	---	---	---					—-	5	6
	Scoring					---		---		5	2
	Question types				---			---		5	1