Phase 5 - Junit Report

Test Case Failures: ActionAddCreditTest

#	Test Name	What is tested?	Failure Description	Behaviour	Action Taken
1	addCreditUserExists	Check if the user exits	N/A	N/A	N/A
2	addCreditUserDoesNotExists	Check if the credit was added	N/A	N/A	N/A
3	addCreditSuccess	Check if the user list was updated	N/A	N/A	N/A

Test Case Failures: ActionAdvertiseTest

#	Test Name	What is tested?	Failure Description	Behaviour	Action Taken
1	advertiseSuccess	Check if the item was added to the item list	N/A	N/A	N/A

Test Case Failures: ActionBidTest

#	Test Name	What is tested?	Failure Description	Behaviour	Action Taken
1	bidItemExisti	Check if the item exists	N/A	N/A	N/A
2	bitItemUpdated	Check if the item was updated to the list	N/A	N/A	N/A

Test Case Failures: ActionCreateTest

#	Test Name	What is tested?	Failure Description	Behaviour	Action Taken
1	userNotExists	Check if the user doesn’t exists	N/A	N/A	N/A
2	userSuccessAdded	Check if the new user was added to the users list	N/A	N/A	N/A

Test Case Failures: ActionDeleteTest

#	Test Name	What is tested?	Failure Description	Behaviour	Action Taken
1	userDoesntExit	Check if the user doesn't exists	N/A	N/A	N/A
2	userExists	Check if the user exists	N/A	N/A	N/A
3	userSuccessDelete	Check if the user was deleted	N/A	N/A	N/A

Test Case Failures: ActionRefundTest

#	Test Name	What is tested?	Failure Description	Behaviour	Action Taken
1	RefundNoNull	Check if the refund amount is not null	N/A	N/A	N/A
2	RefundInvalidTwoUsers	Check if two users enter the same amount	It was failing to when it was checking if one of the user missing	INVALID_USERS	Fixed the loop and if statement
3	RefundInvlidAmount	Check if the refund amount is not invalid	Returning a null when checking for invalid amount	INVALID_REFUND	Used a boolean variable to check if the amount was valid

Test Case Failures: ActionsTest

#	Test Name	What is tested?	Failure Description	Behaviour	Action Taken
1	create	Check if the create class exists	N/A	N/A	N/A
2	advertise	Check if the advertise class exists	N/A	N/A	N/A
3	bid	Check if the bid class exists	N/A	N/A	N/A
4	refund	Check if the refund class	N/A	N/A	N/A

		exists			
5	addCredit	Check if the addCredit class exists	N/A	N/A	N/A
6	delete	Check if the delete class exists	N/A	N/A	N/A
7	searchArray	Checks the threw a list and returns true if it exists in the loop	List was not getting accessed and list did not have a size	SEARCH_FAILED	Initializing the list and making the search boolean

Test Case Failures: FileControllerTest

#	Test Name	What is tested?	Failure Description	Behaviour	Action Taken
1	writeToFile Zero	Checks if it writes to the file zero lines	N/A	N/A	N/A
2	writeToFile One	Checks if it writes to file one line	N/A	N/A	N/A
3	writeToFile Manu	Checks if it writes to file many lines	N/A	N/A	N/A
4	readToFileZero	Checks if it reads the file zero lines	N/A	N/A	N/A
5	readToFile One	Checks if it reads the file one line	N/A	N/A	N/A
6	readToFile Many	Checks if it reads the file many lines	N/A	N/A	N/A

Test Case Failures: MainTest

#	Test Name	What is tested?	Failure Description	Behaviour	Action Taken
1	Refund	Check the file if refund is added	N/A	N/A	N/A
2	CreateUser	Check the file if the user is created	N/A	N/A	N/A

3	LoopCoverageOne	Checks if it was executed at least once	N/A	N/A	N/A
4	LoopCoverageTwo	Checks if it was executed at least twice	N/A	N/A	N/A
5	DeleteUser	Check the file if the user is deleted	N/A	N/A	N/A
6	LoopCoverageMany	Checks if it was executed many times	N/A	N/A	N/A
7	LoopCoverageZero	Checks if it was executed zero times	N/A	N/A	N/A
8	Bid	Check the file if the bid is recorded	N/A	N/A	N/A
9	EOF	Checks if EOF is reached	N/A	N/A	N/A
10	Advertise	Check the file if advertise is recorded	N/A	N/A	N/A
11	AddCredit	Check the file if AddCredit is recorded	N/A	N/A	N/A