Project 1 Part 1 System Test Plan

Document Author: Rohit Kunta
Date: 9/28/2023

Introduction

These are 8 tests that ensure Project 1 is working properly. These tests cover all of Project 1's use cases at a system level, along with error checking.
To start the ProductBacklog application:
1. **Open eclipse on your computer and open the Project1 project file in the package explorer**
2. **Navigate into the src/ folder and right-click on the edu.ncsu.csc216.product_backlog.view.ui package**
3. **In the resulting menu, expand the "Run as" tab and click on Java Application**
4. **The Program will open**

Test Files

tasks1.txt

\# Shopping Cart Simulation
* 1,Backlog,Express Carts,F,jep,unowned,false
- [Backlog] Express carts always choose the shortest line. If there are multiple shortest lines, an express cart chooses the one with the smallest index.
* 2,Owned,Regular Carts,F,jep,sesmith5,false
- [Backlog] Regular carts always choose the shortest line excluding the express register line (at index 0). If there are multiple shortest lines, a regular cart chooses one with the smallest index.
- [Owned] Adding to sesmith5 backlog.
* 3,Processing,Java Swing,KA,sesmith5,sesmith5,false
- [Backlog] Learn more about Swing to debug GUI.
- [Owned] Adding to sesmith5 backlog.
- [Processing] Found Swing tutorials at http://docs.oracle.com/javase/tutorial/uiswing/start/.
If there are multiple shortest special register lines, a special cart

chooses one with the smallest index.

* 10,Rejected,Flatbed carts,F,jep,unowned,false
- [Backlog] Add flatbed carts to simulation.
- [Rejected] Rejected. Flatbed carts won't fit through physical register stations.
WolfScheduler
* 2,Rejected,Weekly Repeat,F,sesmith5,unowned,false
- [Backlog] Events should have a weekly repeat of every 1, 2, 3, or 4 weeks.
- [Rejected] Weekly repeat is unnecessary when creating ideal week.
* 6,Done,Add Course,F,sesmith5,jctetter,true
- [Backlog] Users can add courses to their schedule.
- [Owned] Assigning to jctetter.
- [Processing] Creating Course class.
- [Processing] Adding error checking on course name.
- [Processing] Adding tests for Course.
- [Verifying] Request peer review.
- [Done] Updates meet requirements and test pass.

exp_task_backlog.txt

Product
* 1,Backlog,title1,B,creator1,unowned,false
- [Backlog] note1
* 2,Backlog,title2,F,creator2,unowned,false
- [Backlog] note2
* 3,Backlog,title3,KA,creator3,unowned,false
- [Backlog] note3
A Product
* 1,Backlog,title01,TW,creator01,unowned,false
- [Backlog] note01
* 2,Backlog,title02,F,creator02,unowned,false
- [Backlog] note02

tasks16.txt

Shopping Cart Simulation
* 1,Backlog,Express Carts,F,jep,owner,false
- [Backlog] Express carts always choose the shortest line. If there are multiple shortest lines, an express cart chooses the one with the smallest index.

tasks17.txt

Shopping Cart Simulation
* 2,Owned,Regular Carts,F,jep,unowned,false
- [Backlog] Regular carts always choose the shortest line excluding the express register line (at index 0). If there are multiple shortest lines, a regular cart chooses one with the smallest index.
- [Owned] Adding to sesmith5 backlog.

tasks18.txt

Shopping Cart Simulation
* 3,Processing,Java Swing,KA,sesmith5,unowned,false
- [Backlog] Learn more about Swing to debug GUI.
- [Owned] Adding to sesmith5 backlog.
- [Processing] Found Swing tutorials at http://docs.oracle.com/javase/tutorial/uiswing/start/.

tasks19.txt

Shopping Cart Simulation
* 5,Verifying,Calculating Wait Time,B,jdyoung2,unowned,false
- [Backlog] Special carts are failing system tests associated with wait time.
- [Owned] Adding to sesmith5 backlog.
- [Processing] Replicated failure locally in unit test.
- [Verifying] Implementation complete. Requires peer inspection.

tasks20.txt
Shopping Cart Simulation

* 8,Done,Special Carts,F,jep,unowned,true
- [Backlog] Special carts always choose the shortest special register line.
If there are multiple shortest special register lines, a special cart
chooses one with the smallest index.
- [Owned] Adding to sesmith5 backlog.
- [Processing] Created hierarchy to prepare for other cart types.
- [Verifying] Implementation complete. Requires peer inspection.
- [Done] No problems found during inspection.

tasks21.txt

Shopping Cart Simulation
* 10,Rejected,Flatbed carts,F,jep,owner,false
- [Backlog] Add flatbed carts to simulation.
- [Owned] Rejected. Flatbed carts won't fit through physical register stations.

System Tests

Test ID	Description	Expected Results	Actual Results
Test 1: UC 0/1/3/7/8 Start ProductBacklog, Load Valid File, Clear System, Select Active Product, List Tasks	Preconditions: None 1. Run ProductBacklogGUI 2. *Check Results* 3. Click on "File" in top left of GUI, then click "Load" 4. In the **Load Products** FileChooser dialog select: `test-files/exp_task_back log.txt` 5. Select the drop down menu next to **Current Product** and choose "A Product". 6. *Check Results*	The GUI opens and displays the menu to load [UC1], save [UC2], and clear [UC3] system state and buttons to work with products (Add Product, Delete Product, and Edit Product)[UC4-7] and tasks (Add Task, Delete Task, and Edit Task)[UC9-11](see Figure 2), and Provides a way to select Current Product [UC7]. The system displays the following in the Task Table:	The GUI opens and displays the menu to load [UC1], save [UC2], and clear [UC3] system state and buttons to work with products (Add Product, Delete Product, and Edit Product)[UC4-7] and tasks (Add Task, Delete Task, and Edit Task)[UC9-11](see Figure 2), and Provides a way to select Current Product [UC7]. The system displays the following in the Task Table:

	7. Click "File" in the top left of the GUI, and then click "Clear" 8. *Check Results* 9. Close GUI	● 1, "Backlog", "Technical Work", "title01" ● 2, "Backlog", "Feature", "title02" The task table and product section is empty. Every button except **Add Product** is unclickable.	● 1, "Backlog", "Technical Work", "title01" ● 2, "Backlog", "Feature", "title02" The task table and product section is empty. Every button except **Add Product** is unclickable.
Test 2: UC 4/5/ Add Product, Edit Product	1. Run ProductBacklogGUI 2. Click **Add Product** 3. Enter Product name as: "Nerf" 4. *Check Results* 5. Click **OK** 6. *Check Results* 7. Click **Edit Product** 8. Enter Product name as: "NerfOrNothing" 9. *Check Results* 10. Click **OK** 11. *Check Results* 12. Select "NerfOrNothing" in the product menu if not already selected 13. *Check Results*	A dialog box with "Product Name?" has popped up and the field is filled with "Nerf" The Product is now "Nerf", task table is empty. A dialog box with "Update Product Name?" has popped up and the field is filled with "NerfOrNothing" The Product is now named: "NerfOrNothing" Task table is empty	A dialog box with "Product Name?" has popped up and the field is filled with "Nerf" The Product is now "Nerf", task table is empty. A dialog box with "Update Product Name?" has popped up and the field is filled with "NerfOrNothing" The Product is now named: "NerfOrNothing" Task table is empty
Test 3: UC 9/10 Add a Task, Delete a Task.	Preconditions: Test 1 & 2 1. Run ProductBacklogGUI 2. Click **Add Task** 3. *Check Results* 4. Enter Title: "Cookie", Task Type: Bug, Task Creator: "Rohit", Task Notes: "none" 5. Click **Add Task to Backlog** 6. *Check Results*	A Task input dialog comes up with Product, Title, Task Type, Task Creator, and Task Notes fields There is one task listed: Task ID: 1, Task State: Backlog, Task Feature: Bug, Task Title: Cookie Task Table is now empty	A Task input dialog comes up with Product, Title, Task Type, Task Creator, and Task Notes fields There is one task listed: Task ID: 1, Task State: Backlog, Task Feature: Bug, Task Title: Cookie Task Table is now empty

	7. Select Cookie task if not already selected 8. Click **Delete Task** 9. *Check Results*		
Test 4: UC 11/12/13/14 Edit a Task, Edit Task in Backlog State, Edit task in Owned State, Edit Task in processing state	Preconditions: None 1. Run ProductBacklogGUI 2. Click on "File" in top left of GUI, then click "Load" 3. In the **Load Course Catalog** FileChooser dialog select: `test-files/exp_task_back log.txt` 4. Click Task with ID: 1 5. Click **Edit Task** 6. *Check Results* 7. Enter "Rohit" into the Owner section. Note: Claimed By Rohit 8. Click **Claim Task** 9. *Check Results* 10. Click Task with ID: 2 11. Click **Edit Task** 12. *Check Results* 13. Add Note: "Nothing" 14. Click **Process Task** 15. *Check Results* 16. Click Task with ID: 3 17. Click **Edit Task** 18. *Check Results* 19. Add Note: - [Done] 20. Click **Complete Task** 21. *Check Results*	The User Interface for Editing a BackLog task is shown. ● Task Title: "Express Carts" ● Task Id: 1 ● Task State: Backlog ● Task creator: jep ● Task Type: unowned ● Task Type: Feature ● Notes: [Backlog] Express carts always choose the shortest line. If there are multiple shortest lines, an express cart chooses the one with the smallest index. The Task in the Task table should now say: 1, "Owned", "Feature", "Express Carts" The User Interface for Editing a Owned task is shown. ● Task Title: "Regular Carts" ● Task Id: 2 ● Creator: jep ● Owner: sesmith5 ● Task Type: Feature ● Notes: – [Backlog] Regular carts always	The User Interface for Editing a BackLog task is shown. ● Task Title: "Express Carts" ● Task Id: 1 ● Task State: Backlog ● Task creator: jep ● Task Type: unowned ● Task Type: Feature ● Notes: [Backlog] Express carts always choose the shortest line. If there are multiple shortest lines, an express cart chooses the one with the smallest index. The Task in the Task table should now say: 1, "Owned", "Feature", "Express Carts" The User Interface for Editing a Owned task is shown. ● Task Title: "Regular Carts" ● Task Id: 2 ● Creator: jep ● Owner: sesmith5 ● Task Type: Feature ● Notes: – [Backlog] Regular carts always

			`choose the shortest line excluding the express register line (at index 0). If there are multiple shortest lines, a regular cart chooses one with the smallest index.` ● `- [Owned] Adding to sesmith5 backlog.` The Task in the Task table should now say: 2, "Processing", "Feature", "Regular Carts" The User Interface for Editing a Owned task is shown. ● Task Title: "Java Swing" ● Task Id: 3 ● Task State: Processing ● Creator: sesmith5 ● Owner: sesmith5 ● Task Type: Knowledge Acquisition ● Notes: `- [Backlog] Learn more about Swing to debug GUI.` ● `- [Owned] Adding to sesmith5 backlog.` ● `- [Processing] Found Swing tutorials at` http://docs.oracle.com/javase/tutorial/uiswing/start/`.`	`choose the shortest line excluding the express register line (at index 0). If there are multiple shortest lines, a regular cart chooses one with the smallest index.` ● `- [Owned] Adding to sesmith5 backlog.` The Task in the Task table should now say: 2, "Processing", "Feature", "Regular Carts" The User Interface for Editing a Owned task is shown. ● Task Title: "Java Swing" ● Task Id: 3 ● Task State: Processing ● Creator: sesmith5 ● Owner: sesmith5 ● Task Type: Knowledge Acquisition ● Notes: `- [Backlog] Learn more about Swing to debug GUI.` ● `- [Owned] Adding to sesmith5 backlog.` ● `- [Processing] Found Swing tutorials at` http://docs.oracle.com/javase/tutorial/uiswing/start/`.`

		• If there are multiple shortest special register lines, a special cart • chooses one with the smallest index. The Task in the Task table should now say: 3, "Done", "Knowledge Acquisition", "Java Swing"	• If there are multiple shortest special register lines, a special cart • chooses one with the smallest index. The Task in the Task table should now say: 3, "Done", "Knowledge Acquisition", "Java Swing"
Test 5: UC 15/16/17 Edit Task in Verifying State, Edit task in Done State, Edit Task in Rejected state	PreConditions: Test 4 1. Click Task with ID: 2 2. Click **Edit Task** 3. Click **Verify Task** and add note: **"Verifying"** 4. Click Task with ID: 2 5. Click **Edit Task** 6. *Check Results* 7. Add Note: - [Done] 8. Click **Task Verified** 9. *Check Results* 10. Select dropdown menu for currentProduct and choose "**WolfScheduler**" 11. Click Task with ID: 6 12. Click **Edit Task** 13. *Check Results* 14. Add Note: Backlogged 15. Click **Backlog Task** 16. *Check Results* 17. Select dropdown menu for currentProduct and choose "**Shopping Cart Simulation**" 18. Click Task with ID: 10	The User Interface for Editing a Verifying task is shown. • Task Title: "Regular Carts" • Task Id: 2 • Task State: Verifying • Creator: jep • Owner: sesmith5 • Task Type: Feature • Notes: `- [Backlog] Regular carts always choose the shortest line excluding the express register line (at index 0). If there are multiple shortest lines, a regular cart chooses one with the smallest index.` • `- [Owned] Adding to sesmith5 backlog.` • `- [Processing] Nothing` • `- [Verifying] Verifying`	The User Interface for Editing a Verifying task is shown. • Task Title: "Regular Carts" • Task Id: 2 • Task State: Verifying • Creator: jep • Owner: sesmith5 • Task Type: Feature • Notes: `- [Backlog] Regular carts always choose the shortest line excluding the express register line (at index 0). If there are multiple shortest lines, a regular cart chooses one with the smallest index.` • `- [Owned] Adding to sesmith5 backlog.` • `- [Processing] Nothing` • `- [Verifying] Verifying`

	19. Click **Edit Task** 20. *Check Results* 21. Add Note: Backlogged 22. Click **Backlog Task** 23. *Check Results* 24.	The Task in the Task table should now say: 2, "Done", "Feature", "Regular Carts" The User Interface for Editing a Verifying task is shown. ● Task Title: "Add Course" ● Task Id: 6 ● Task State: Done ● Creator: sesmith5 ● Owner: jcetter ● Task Type: Feature ● Notes: − `[Backlog] Users can add courses to their schedule.` ● `− [Owned] Assigning to jctetter.` ● `− [Processing] Creating Course class.` ● `− [Processing] Adding error checking on course name.` ● `− [Processing] Adding tests for Course.` ● `− [Verifying] Request peer review.` ● `− [Done] Updates meet requirements and test pass.` The Task in the Task table should now say: 6, "Backlog", "Feature", "Add Course"	The Task in the Task table should now say: 2, "Done", "Feature", "Regular Carts" The User Interface for Editing a Verifying task is shown. ● Task Title: "Add Course" ● Task Id: 6 ● Task State: Done ● Creator: sesmith5 ● Owner: jcetter ● Task Type: Feature ● Notes: − `[Backlog] Users can add courses to their schedule.` ● `− [Owned] Assigning to jctetter.` ● `− [Processing] Creating Course class.` ● `− [Processing] Adding error checking on course name.` ● `− [Processing] Adding tests for Course.` ● `− [Verifying] Request peer review.` ● `− [Done] Updates meet requirements and test pass.` The Task in the Task table should now say: 6, "Backlog", "Feature", "Add Course"

		The User Interface for Editing a Verifying task is shown.	The User Interface for Editing a Verifying task is shown.
		- Task Title: "Flatbed Carts" - Task Id: 10 - Task State: Rejected - Creator: jep - Owner: unowned - Task Type: Feature - Notes: `– [Backlog] Add flatbed carts to simulation.` `– [Rejected] Rejected. Flatbed carts won't fit through physical register stations.` The Task in the Task table should now say: 10, "Backlog", "Feature", "Flatbed carts"	- Task Title: "Flatbed Carts" - Task Id: 10 - Task State: Rejected - Creator: jep - Owner: unowned - Task Type: Feature - Notes: `– [Backlog] Add flatbed carts to simulation.` `– [Rejected] Rejected. Flatbed carts won't fit through physical register stations.` The Task in the Task table should now say: 10, "Backlog", "Feature", "Flatbed carts"

Test 6: UC 2/18
 	
Save the Product Backlog, Quit Product Backlog
	Preconditions: Test 5
	
	1. **Check** *Task Table*
	2. Click "File" in the top left corner and Click **"Quit"**
	3. Click Save.
	4. *Check results*
	5. In the **Save Product Backlog** FileChooser dialog select:
	`test-files/actual_produc t_backlog.txt`
	6. Click **Save**
	7. *Check Results*
	8. **Close** GUI
		Task Table for Shopping Car Simulation contains:
		● 1, "Owned", "Feature", "Express Carts"
		● 2, "Done", "Feature", "Regular Carts"
		● 3, "Done", "Knowledge Acquisition", "Java Swing"
		● 10, "Backlog", "Feature", "Flatbed carts"
		
		The user is prompted to save their file by either choosing a file to save to [UC2] or declines to save the changes by selecting Cancel.
		
		The actual_product_backlog.txt contains:
		
		`# Shopping Cart Simulation`
		`* 1,Owned,Express Carts,F,jep,Rohit,false`
		`- [Backlog] Express carts always choose the shortest line. If there are multiple shortest lines, an express cart chooses the one with the smallest index.`
		`- [Owned] Claimed By Rohit`
		`* 2,Done,Regular Carts,F,jep,sesmith5,true`
		`- [Backlog] Regular carts always choose the shortest line excluding the express register line (at index 0). If there are multiple shortest lines, a regular`
			Task Table for Shopping Car Simulation contains:
			● 1, "Owned", "Feature", "Express Carts"
			● 2, "Done", "Feature", "Regular Carts"
			● 3, "Done", "Knowledge Acquisition", "Java Swing"
			● 10, "Backlog", "Feature", "Flatbed carts"
			
			The user is prompted to save their file by either choosing a file to save to [UC2] or declines to save the changes by selecting Cancel.
			
			The actual_product_backlog.txt contains:
			
			`# Shopping Cart Simulation`
			`* 1,Owned,Express Carts,F,jep,Rohit,false`
			`- [Backlog] Express carts always choose the shortest line. If there are multiple shortest lines, an express cart chooses the one with the smallest index.`
			`- [Owned] Claimed By Rohit`
			`* 2,Done,Regular Carts,F,jep,sesmith5,true`
			`- [Backlog] Regular carts always choose the shortest line excluding the express register line (at index 0). If there are multiple shortest lines, a regular`

| | | | cart chooses one with the smallest index.
- [Owned] Adding to sesmith5 backlog.
- [Processing] Nothing
- [Verifying] Verifying
- [Done] Verified
* 3,Done,Java Swing,KA,sesmith5,sesmith5,false
- [Backlog] Learn more about Swing to debug GUI.
- [Owned] Adding to sesmith5 backlog.
- [Processing] Found Swing tutorials at http://docs.oracle.com/javase/tutorial/uiswing/start/. If there are multiple shortest special register lines, a special cart chooses one with the smallest index.
- [Done] Done
* 10,Backlog,Flatbed carts,F,jep,unowned,false
- [Backlog] Add flatbed carts to simulation.
- [Rejected] Rejected. Flatbed carts won't fit through physical register stations.
- [Backlog] Backlogged
WolfScheduler
* 2,Rejected,Weekly Repeat,F,sesmith5,unowned,false
- [Backlog] Events should have a weekly repeat of every 1, 2, 3, or 4 weeks. | cart chooses one with the smallest index.
- [Owned] Adding to sesmith5 backlog.
- [Processing] Nothing
- [Verifying] Verifying
- [Done] Verified
* 3,Done,Java Swing,KA,sesmith5,sesmith5,false
- [Backlog] Learn more about Swing to debug GUI.
- [Owned] Adding to sesmith5 backlog.
- [Processing] Found Swing tutorials at http://docs.oracle.com/javase/tutorial/uiswing/start/. If there are multiple shortest special register lines, a special cart chooses one with the smallest index.
- [Done] Done
* 10,Backlog,Flatbed carts,F,jep,unowned,false
- [Backlog] Add flatbed carts to simulation.
- [Rejected] Rejected. Flatbed carts won't fit through physical register stations.
- [Backlog] Backlogged
WolfScheduler
* 2,Rejected,Weekly Repeat,F,sesmith5,unowned,false
- [Backlog] Events should have a weekly repeat of every 1, 2, 3, or 4 weeks. |

		- [Rejected] Weekly repeat is unnecessary when creating ideal week. * 6,Backlog,Add Course,F,sesmith5,unowned,false - [Backlog] Users can add courses to their schedule. - [Owned] Assigning to jctetter. - [Processing] Creating Course class. - [Processing] Adding error checking on course name. - [Processing] Adding tests for Course. - [Verifying] Request peer review. - [Done] Updates meet requirements and test pass. - [Backlog] Backlogged	- [Rejected] Weekly repeat is unnecessary when creating ideal week. * 6,Backlog,Add Course,F,sesmith5,unowned,false - [Backlog] Users can add courses to their schedule. - [Owned] Assigning to jctetter. - [Processing] Creating Course class. - [Processing] Adding error checking on course name. - [Processing] Adding tests for Course. - [Verifying] Request peer review. - [Done] Updates meet requirements and test pass. - [Backlog] Backlogged

| Test 7: Add Invalid Product, Add Invalid task | Preconditions: None

Start ProductBacklogGUI
  1. Click **Add Product**
  2. Leave the Product Name box **Empty.**
  3. Click **OK**
  4. Check Results
  5. Click **OK**
  6. Click **Add Product**
  7. Set name to: **Nerf**
  8. Click **Add Task**
  9. Leave the title box **blank**, set Task Type to **Feature**, set creator to **Rohit**, and set Task notes to "**Added Task**".
  10. Click **Add Task to Backlog**
  11. Check Results
  12. Set the title box to **Nerf3000**, set Task Type to **Feature**, leave creator **blank**, and set Task notes to "**Added Task**".
  13. Click **Add Task to Backlog**
  14. Check Results
  *15. Set the title box to **Nerf3000**, set Task Type to **Feature**, set creator to **Rohit**, and leave notes **blank**.*
  16. Click **Add Task to Backlog.**
  17. Check Results
  18. Close GUI | Pop-up box with the test "Invalid product name." comes up.

Pop-up box with the test "Invalid Task information" comes up.

Pop-up box with the test "Invalid Task information" comes up.

Pop-up box with the test "Invalid Task information" comes up. | Pop-up box with the test "Invalid product name." comes up.

Pop-up box with the test "Invalid Task information" comes up.

Pop-up box with the test "Invalid Task information" comes up.

Pop-up box with the test "Invalid Task information" comes up. | |

| Test 8: Load invalid files | Preconditions: none.

Start ProductBacklogGUI

1. Navigate to the File button in upper left of GUI and click **Load**
2. Select file: `test-files/tasks16.txt`
3. Click **Open**
4. Check Results
5. Navigate to the File button in upper left of GUI and click **Load**
6. Select file: `test-files/tasks17.txt`
7. Click **Open**
8. *Check Results*
9. Navigate to the File button in upper left of GUI and click **Load**
10. Select file: `test-files/tasks18.txt`
11. Click **Open**
12. *Check Results*
13. Navigate to the File button in upper left of GUI and click **Load**
14. Select file: `test-files/tasks19.txt`
15. Click **Open**
16. *Check Results*
17. Navigate to the File button in upper left of GUI and click **Load**
18. Select file: `test-files/tasks20.txt`
19. Click **Open**
20. *Check Results* | Nothing Should be added or Deleted in the Product Backlog GUI.

Nothing Should be added or Deleted in the Product Backlog GUI.

Nothing Should be added or Deleted in the Product Backlog GUI.

Nothing Should be added or Deleted in the Product Backlog GUI.

Nothing Should be added or Deleted in the Product Backlog GUI.

Nothing Should be added or Deleted in the Product Backlog GUI. | Nothing Should be added or Deleted in the Product Backlog GUI.

Nothing Should be added or Deleted in the Product Backlog GUI.

Nothing Should be added or Deleted in the Product Backlog GUI.

Nothing Should be added or Deleted in the Product Backlog GUI.

Nothing Should be added or Deleted in the Product Backlog GUI.

Nothing Should be added or Deleted in the Product Backlog GUI. |

	21. Navigate to the File button in upper left of GUI and click **Load** 22. Select file: `test-files/tasks21.txt` 23. Click **Open** **24.** *Check Results* 25. Close GUI		