PARTY OF FOUR!

Due: 4/13/2015

Implementation and Tests: Part 4

Julie Quiroz

Travis Allan

Justin Durko

Use Case 3.1

Name: PlaceOrder

Participating actor: Customer

Entry Condition:

* Customer has been seated
* Customer is hungry

Exit Condition:

* Customer has sent order to kitchen

Event Flow:

1. Customer selects item of food desired to eat
2. The information of the item will be displayed
3. Add any special preparation direction (if any)
4. "Add to Cart" button will be pressed
5. In "Cart" press "Send to Kitchen"
6. The item will be added to the bill
7. A message will prompt saying  
   "\_\_\_ is now preparing your food!"

Outcome:

Events 1 through 6 are all achievable in this use case. The case fails when event 7 hits. Our program fails to create instances of orders to the table of orders in our database. We also do not working views for kitchen or wait staff.

Use Case 3.1.1

Name: CallAssistance

Participating actor: Customer

Entry Condition:

* Customer has been seated
* Customer needs help from human

Exit Condition:

* Customer has been helped

Event Flow:

1. Customer presses "Call for Assistance" button
2. The wait staff receives the notification
3. Wait staff is sent to table
4. Notification is removed from wait staff's display

Outcome:

This use case fails because there is no button functionality behind the area where the button that calls for wait staff assistance is located. Additionally there is no wait staff view for the wait staff to be notified.

Use Case 3.3.2

Name: ModifyOrder

Participating actor: Wait Staff

Entry Condition:

* Customer has placed order
* Customer wants to modify the order they have placed
* Wait staff has been informed of the modification the customer would like to make

Exit Condition:

* Order has been modified to reflect customer’s latest requests

Event Flow:

1. Customer gets the wait staff’s attention by using the “Call for Assistance” button
2. Wait staff responds to the call
3. Customer tells the wait staff what modification should be made to the order
4. Wait staff modifies the order

Outcome:

This use case fails because the customer cannot place an order and there is no ability for wait staff to receive notifications in.

Use Case 3.3.4

Name: SplitCheck

Participating actor: Customer

Entry Condition:

* Customer has no more orders to place with wait staff
* Customer has received bill from wait staff

Exit Condition:

* Customer has selected how many ways the bill should be split (max allowance is 4)

Event Flow:

1. Customer has been prompted and received the bill from wait staff
2. Customer confirms that the amount is correct
3. Customer selects to split bill 3 ways
4. Total amount on bill is divided by 3 and presented to customer

Outcome:

This use case fails because the customer cannot place an order. Additionally, since the wait staff does not have a screen to manage from, they cannot issue bills because they cannot see that the customer has submitted an order.

Use Case 3.2.1

Name: PlayGames

Participating actor: Customer

Entry Condition:

* The customer would like to play a game.

Exit Condition:

* The customer no longer wishes to play.

Event Flow:

1. The customer opens the game folder.
2. The customer chooses a game to play.
3. The customer ends the game.

Outcome:

This use case fails because there are no games for the customer to play.

Use Case 3.2.2

Name: WinDessert

Participating actor: Customer

Entry Condition:

* By chance, the customer wins a free dessert while playing a game.

Exit Condition:

* The message is sent to the wait staff and cooks.

Event Flow:

1. Customer is playing one of the games.
2. The customer wins a free dessert
3. The message is sent to the wait staff cooks
4. The customer receives a free dessert.

Outcome:

This use case fails because there are no games for the customer to play.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case No. | Requirement No. | Test Steps | Expected Output | Comments | Pass/Fail |
| 1 | 3.1 | 1. Press “Menu”  2. Press on “Mac-N-Cheese”  3. Run test case #8  4. Press “Add to Cart”  5. Go back to “Menu”  6. Press “Dr. Pepper”  7. Press “Add to Cart”  8. Press “Cart”  9. Press “Send to Kitchen” | The customer will successfully send items of food to be served in the restaurant. | Passes from test steps 1 - 8 | Pass\* |
| 2 | 3.1.1 | 1.Press “Call for Assistance” | The customer will successfully call the wait staff for assistance and the wait staff will be able to see a notification. | -- | Fail |
| 3 | 3.1.2 | 1.Run test case #1  2. Press “Drink Refill” | The customer will successfully request a refill of the desired individual in need of one.  (Ex. Seat 1: Coke) | -- | Fail |
| 4 | 3.1.3 | 1.Run test case #1  2. “Justin is now preparing your order!” will pop-up. | A message will display that will say “\_\_\_ is preparing your meal” | -- | -- |
| 5 | 3.1.4 | 1. Press “Pay”  2. View total balance  3. Press “Checkout”  4.Run test case #6  5. Payment transaction takes place  6. Press “Email Receipt?” or “Printed Receipt?”  6. Test case #7 takes place | During the payment process, the receipt will be either e-mailed to the customer or printed. | -- | Fail |
| 6 | 3.1.5 | 1.Press “15%” or “Enter Amount”  2. Total amount will update | The gratuity will be auto-calculated or allow entry of a desired amount and the total bill amount will be updated. | -- | Fail |
| 7 | 3.1.6 | 1.Run test case #5  2. “Would you like to participate in a survey?” message will appear  3. Press “Yes” or “No”  4. If “Yes”, survey will appear, then press “Submit”  5. If “No”, survey message will disappear | The customer will be prompted to take a survey about their dining experience, but can refuse the offer if desired. | -- | Fail |
| 8 | 3.1.7 | 1. In text box named: “Special preparation info?”  2. Type “No onions” | The customer will be able to successfully enter special preparation directions for the kitchen to follow. | -- | Fail |
| 9 | 3.2 | 1. Click Games tab to open games interface. | The interface will switch to the games tab, where the games will be displayed. | -- | Fail |
| 10 | 3.2.1 | 1. From the games interface, click on one of the two games. 2. The game will start. 3. Play the game. 4. Click the close button. | Two games will be displayed to choose from. The one you click will open and start. The game will be playable and fluid. The game will close when you click close. | -- | Fail |
| 11 | 3.2.2 | 1. Winning numbers are achievable within 20%. 2. The winning message is displayed properly. 3. A menu pops up to let you choose your dessert. 4. Your choice is added to your order properly. 5. A notification is properly sent through the system. | While playing the game, if your score matches up with one of the winning numbers, a message will display that you won a free dessert. A menu will then pop up to let you choose your dessert. Your choice will be added to your order, free of charge, and a notification of the change will be sent to the wait staff and cooks. | -- | Fail |
| 12 | 3.3 | 1. The information on the home menu is correct, and buttons are functional. 2. The login system is functioning correctly. 3. The individual interface displays correct information and all buttons work. 4. The customer order is displaying correct information and information can be entered manually, | From the home menu, the wait staff can see the status of all tables, access their individual interfaces, request help, change the settings, sign in, and see all of the current employees working. Upon clicking the tap corresponding to their name, and providing their password, or the master key, the wait staff member will be able to access their individual interface. From that interface they can interact with the customers order. They will be able to comp an item, choose payment, choose receipt type, and manually override the order if needed. | -- | Fail |
| 13 | 3.3.1 | 1. Clicking on the individual seats in the table box shows each customers order. 2. All of the information is correct as inputted. | From the wait staff members individual interface, the customers order can be viewed in full detail. | -- | Fail |
| 14 | 3.3.2 | 1. Clicking on an order item allows you to manually change it to another item. 2. Clicking the "+" button(not in the mockup) lets you add an item. 3. Clicking the "-" button(not in the mockup) lets you remove an item. | From the wait staff members individual interface, the customers order be modified to change an item, add an item, delete an item. | -- | Fail |
| 15 | 3.3.3 | 1. wait staff modifies item cost to "null" 2. item on bill displays cost of zero | Item will display on customers' bill with price equal to "VOID" | -- | Fail |
| 16 | 3.3.4 | 1. customer receives bill 2. customer selects option to split bill 3. bill is split according to customer's entry 4. option to pay each portion is presented to customers | Customer will be able to divide the total cost of the bill by values equal to "2", "3", or "4" | -- | Fail |
| 17 | 3.4 | Each item detailed in subsections below | Orders will be viewable to kitchen staff. The order that is being prepared will be assigned to the kitchen staff member who is preparing it. The kitchen staff may change the status of an order to "WIP", or "DONE" | -- | Fail |
| 18 | 3.4.1 | 1. kitchen staff member will be logged in to system 2. all kitchen staff members will be able to view all order that have been sent by customers and order modified by wait staff | All orders will be viewable to wait staff | -- | Fail |
| 19 | 3.4.2 | 1. from the list of orders that have been placed the kitchen staff member will assign themselves to the order 2. customer will be able to see the kitchen staff member's name who is working on order | The order that is being prepared will be assigned to the kitchen staff member who is preparing it. | -- | Fail |
| 20 | 3.4.3 | 1. kitchen staff member has completed cooking an item 2. item has been plated 3. kitchen staff changes status 4. wait staff delivers plated item to customer at table | The kitchen staff may change the status of an order to "WIP", or "DONE" | -- | Fail |