CO567
Object-Oriented Systems Development
Coursework 1

The BUCKS Centre for the Performing Arts
Design & Implementation

Alex Paulo da Costa

21611431

Jack Standen

21204638

James Harris

21606555

Qasim Maruf

21610356

1st December 2017

Computing & Web Development
Buckinghamshire New University

Summary

A Design **2**

1 Use Case Models . . . **3**
- A The Consumer . . . 3
  - A.1 Register . . . 3
  - A.2 Login . . . 3
  - A.3 Update User Profile . . . 3
  - A.4 View Upcoming Events . . . 3
  - A.5 View Shows by Date . . . 4
  - A.6 Purchase Tickets . . . 4
  - A.7 View Tickets . . . 4
  - A.8 Diagrams . . . 5
- B The Venue Manager . . . 5
  - B.1 Add Event . . . 5
  - B.2 Reschedule Event . . . 5
  - B.3 Cancel Event . . . 5
  - B.4 Add Show . . . 6
  - B.5 Reschedule Show . . . 6
  - B.6 Cancel Show . . . 6
  - B.7 Change Maximum-Seats-Per-Customer Value . . . 7
  - B.8 Add Promotion . . . 7
  - B.9 Assign Promotions . . . 7
  - B.10 Change Promotion . . . 7
  - B.11 Delete Promotion . . . 8
  - B.12 Create Agent's Contract . . . 8
  - B.13 Modify Agent's Contract . . . 8
  - B.14 Cancel Agent's Contract . . . 8
  - B.15 Renew Agent's Contract . . . 9
  - B.16 Diagrams . . . 9
- C The Agent . . . 10
- D Consolidated Model . . . 10

2 Identifying Classes . . . **11**

3 Data Dictionary . . . **12**

4 Report . . . **13**

B Implementation **14**

5 Requirements . . . **15**
- A Expected Functional . . . 15
- B Non Functional . . . 15

6 Design Class Diagram . . . **16**

7 Sequence Models . . . **17**

8 Detailed Design with Pseudo-code **18**

9 Implementation in Java **19**

C Appendix **20**

Section A

Design

Use Case Models

A The Consumer

A.1 Register

Pre-conditions: The consumer does not have an existing customer profile.

Post-conditions: A new customer profile exists in the system, with a name, email and password and the consumer is logged in.

Purpose: The consumer wants to book tickets for an upcoming show.

Description: The consumer fills in a registration form with their desired name, email and password. Once complete, they press the register button and they will be automatically logged in after their customer profile has been created on the system.

A.2 Login

Pre-conditions: The consumer has a customer profile.

Post-conditions: The consumer is logged in.

Purpose: The consumer has previously bought tickets for a show and wants to buy tickets for an upcoming show.

Description: The consumer enters their email and password. The details are verified on the system and if they are valid, the user is logged in. If the credentials are invalid, the consumer is prompted to re-enter their email and password.

A.3 Update User Profile

Pre-conditions: The consumer has a customer profile and is logged in.

Post-conditions: The stored customer profile information is modified.

Purpose: The consumer wants to change their email, shipping address or password.

Description: The consumer updates the fields they wish to change (e.g, their shipping address). Once they are happy with the changes, they click the save button and their customer profile will be modified on the system.

A.4 View Upcoming Events

Pre-conditions: Zero or more events exists.

Purpose: The consumer wants to see all the upcoming events that they can book tickets for.

Description: The consumer is presented with a list of events. Selecting an event will show the list of shows assigned to the event. If there are no upcoming events, then the list will be empty with a message indicating there are no events on.

A.5 View Shows by Date

Pre-conditions: Zero or more shows exist.

Purpose: The consumer wants to see all the shows on over a range of days.

Description: The consumer is presented with a list of shows that are taking place within their specified date range. Selecting a show will allow the consumer to book tickets for the show. If there are no shows taking place in this date range, the list will be empty with a message indicating there are no shows.

A.6 Purchase Tickets

Pre-conditions: The consumer is logged in and one or more shows exist.

Post-conditions: A new ticket exists in the system.

Purpose: The consumer wants to buy one or more tickets for a selected show.

Description: The consumer is asked how many tickets they wish to purchase. After selecting the amount, the system displays the best available seat(s) with their price. The consumer also has the option of manually picking their seats via a button. The manual option displays a seating chart with the available seats highlighted. The consumer can then select the needed amount of seats. With both methods, the currently selected seats are reserved and not available to other consumers that are booking tickets for the same show. The reservation is cancelled once the transaction is cancelled or has not been completed after 5 minutes.

Once they are happy with the selected seats, the consumer is shown the total cost of the tickets. Here they are able to add a valid promotion for the show. The consumer can then enter their credit card information and confirm the purchase.

Upon confirmation of the purchase, the ticket(s) are added to the system and displayed again to the user as a receipt.

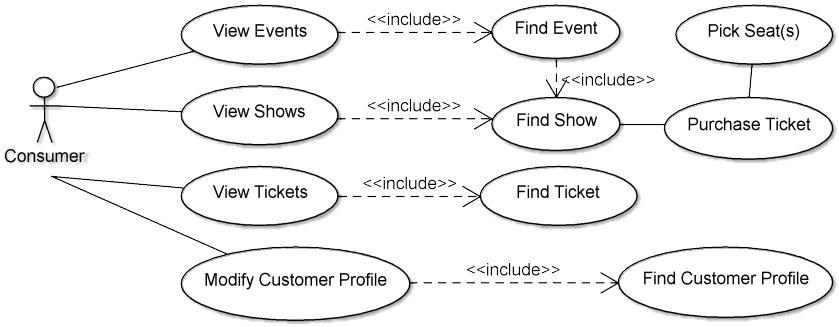
A.7 View Tickets

Pre-conditions: The consumer is logged in.

Purpose: The consumer wants to see their past purchases.

Description: The consumer is presented with a list of their tickets, showing the price, date and seat number.

A.8 Diagrams



B The Venue Manager

B.1 Add Event

Pre-conditions: The venue manager is logged in into the system.

Post-conditions: A new event exists in the system, with a start and end date and a name.

Purpose:The venue manager wants to add an event to the ticket selling system in order to allow customers to buy tickets for it and to promote it.

Description: The venue manager fills in the form for the creation of a new event with the name of the event and a description of it, then selects a start and end date from the ”calendar menus”. When they are satisfied with all the details, they press the save button and the changes are saved in the system.

B.2 Reschedule Event

Pre-conditions:The venue manager is logged in into the system and at least one event exists in the system that hasn’t been cancelled already

Post-conditions: An existing event has either a new start date, a new end date or both

Purpose:The venue manager wants to change the dates an event runs for in order to allow customers to have the accurate dates the event will be occurring and for them to be able to buy tickets for it

Description: The manager selects the event from a list of events, then a screen with the event’s information displayed in a form style appears, allowing the manager to make the changes he needs to make to the dates, then they press the save button and the changes are saved into the system.

B.3 Cancel Event

Pre-conditions: The venue manager is logged in into the system and at least one event exists in the system.

Post-conditions: A previously existing event will be displayed as cancelled in the system and no more tickets will be available for sale for that event (the event won't be deleted from the system though, in case customers have already seen the event in the system and/or bought tickets for it)

Purpose: The venue manager wants to cancel an event that was scheduled to occur for any reason in order to inform the customers that the event is no longer occurring.

Description: The venue manager selects the event they wish to cancel from a list and then the system prompts them with a message to confirm they wish to cancel that event. If the user selects yes, the changes are saved into the system and the event will then appear as cancelled to the users.

B.4 Add Show

Pre-conditions: The venue manager is logged in into the system and at least one event exists in the system.

Post-conditions: A new showing for an event exists in the system, with a date and start time and a maximum-seats-per-customer value.

Purpose: The venue manager wants to add a showing to an event that is planned to inform the customers and agents of the show occurrence and allow for tickets to be sold.

Description: The manager selects the event for which they want to add a show from a list, then enters the details of the show in a form (date the show is happening and start and end time as well as the maximum-seats-per-customer value) and then presses save which will save the changes in the system and inform the user of the changes.

B.5 Reschedule Show

Pre-conditions: The venue manager is logged in into the system and at least one event with a least one show exists in the system.

Post-conditions: Either the date or the time or both of a show will be modified in the system.

Purpose: The venue manager wants to reschedule a showing of an event for whatever reason, in order to inform customers and sell tickets for the show with accurate time and date information.

Description: The manager selects the event for which they want to reschedule the show from a list and then a list of the shows for that event appear, from which the user selects the show that they want to reschedule. After selecting the show, a form displaying the current date and times of the show is shown in the screen, and the user can then select the new date and/or times. When they are satisfied with the new details for the show, they press the save button. The system will then save the changes and notify the users that the changes have been saved.

B.6 Cancel Show

Pre-conditions:The venue manager is logged in the system and at least one event and one show that hasn't been cancelled for that event exist in the system.

Post-conditions: The show will be marked as cancelled in the system and no more tickets will be available for sale for that show (the show won't be deleted from the system though, in order to allow customers to see that the event has been cancelled).

Purpose: The venue manager wishes to cancel a show for whatever reason is order to inform the customers that the show is no longer occurring and block the sale of tickets for that showing of the event.

Description: The user selects the event that the showing relates to from a list, then a list appears with the possible shows. The user selects the show they wish to cancel. After that, the system displays a confirmation message to make sure that it is that show that the user wants to cancel. After the user confirms that it is that show that they want to cancel, the systems saves the changes and notifies the user that the changes have been saved.

B.7 Change Maximum-Seats-Per-Customer Value

Pre-conditions: The venue manager is logged into the system and at least one event and one show exist in the system.

Post-conditions: The show will have a different maximum-seats-per-customer value.

Purpose: The venue manager wants to change the amount of seats a customer can buy of a single show in order to adapt the number of tickets that can be sold to a single customer according to the current demand for that show.

Description: The user selects the event from a list and the system then displays a list with the shows for that event. After selecting the show the user wants to change, the system displays a form that user uses to change the maximum-seats-per-customer value. After changing the value, the user presses the save button. The system saves the changes in the system and then notifies the user that the changes have been saved.

B.8 Add Promotion

Pre-conditions: The venue manager is logged into the system

Post-conditions: A promotion (pricing structure) is created in the system, with a name, price structure for different types of tickets and applicable discounts (volume discounts and others)

Purpose: The venue manager wants to set a new promotion for an upcoming event, or for a particular time of the day, with specific prices and applicable discounts.

Description: The system displays a form with the information that needs to be filled in by the user to create the new promotion(name, prices for children, students, adults and seniors, as well as a section to add different types of discounts that is optional). The user fills in the form and presses the save button. The system saves the changes and notifies the user that the campaign has been added when all the changes have been saved.

B.9 Assign Promotions

Pre-conditions: The venue manager is logged into the system, and at least one event with one show and one promotion exist in the system.

Post-conditions: A promotion is assigned to some (if not all) seats of the selected show.

Purpose: The venue manager wishes to assign a promotion to the seats of a show so that the correct amount of money is charged to customers when they want to buy tickets for that show and in order to be able to sell tickets for the show.

Description: The user selects the event to which the show belongs from a list, then the system displays a list with the shows for that event. After the user selects the show, the system displays the seats of the room where the show is occurring. The user selects the seats to which the user wants to add a promotion, then the system displays the available promotions. The user selects the promotion to add to those seats. In that screen, the user can also select other ranges of seats and attribute a promotion to each of them. When satisfied with the changes, the user presses the save button. The system saves the changes and notifies that the promotions have been assigned when all the changes are stored.

B.10 Change Promotion

Pre-conditions: The venue manager is logged into the system, and at least one promotion exists in the system.

Post-conditions: The details of a promotion have been changed (either the name, or the pricing structure or available discounts for that promotion).

Purpose: The venue manager wants to change a promotion to update its prices, or modify discounts available for it.

Description: The user selects the promotion they want to change from a list. The system displays then a form containing the current information for the promotion and that also allows the user to change the information they want to change (except the name of the promotion). After they made the changes, they press the save button. The system saves the changes and notifies the user of that when all changes are stored.

B.11 Delete Promotion

Pre-conditions: The venue manager is logged into the system and at least one promotion exists on the system.

Post-conditions: The selected promotion is deleted from the system.

Purpose: The venue manager wishes to delete an irrelevant promotion from the system.

Description: The user selects the promotion they wish to delete from a list. The system displays a confirmation message to make sure that is the promotion that the user wishes to delete, and if the user confirms then the system saves the changes and notifies the user that the changes have been saved.

B.12 Create Agent's Contract

Pre-conditions: The venue manager is logged into the system.

Post-conditions: A new agent's contract is created.

Purpose: The venue manager wishes to create a new agent's contract to allow an agent to use the OTS and sell tickets with their own seats assigned and the right commission.

Description: The user fills in the form with all the details about the agent's contract (name, email, seats to be assigned, commission that they earn, start date of the contract and duration of the contract). The user then clicks on the save button and the system saves the changes. When all the changes are saved, the system will notify the users that the contract has been added successfully.

B.13 Modify Agent's Contract

Pre-conditions: The venue manager is logged into the system and there is at least one agent's contract in the system.

Post-conditions: The agent's contract has been changed and has now new values (assigned seats, commission, duration of the contract and/or start date if the contract hasn't started yet).

Purpose: The venue manager wishes to change an agent's contract for whatever reason so that the agent has access to the platform throughout the duration of their actual contract and no longer and to have a range of seats available for them to sell that is accurate to their sales, as well as to take the right commission.

Description: The user selects the name of the agent whose contract's details need to be changed. The system sends back a form with the details of the agent's contract. The user changes the details that need to be changed and, when finished, presses the save button. The system saves the changes and notifies the user when the changes have been applied.

B.14 Cancel Agent's Contract

Pre-conditions: The venue manager is logged in into the system and at least one agent contract is in the system.

Post-conditions: The agent's contract is terminated, the seats they had reserved are available to all non-agent customers, and the details of the contract are deleted from the system.

Purpose: The venue manager wishes to terminate the Contract the BCPA has with an agent for whatever reason in order to release the seats back to sale for general customers and to end their access to the system.

Description: The user selects a agent from a list, and a confirmation message appears to make sure that the user has chosen the right agent. When confirmed, the system removes the agent contract from the system and notifies the user when all the changes have been processed.

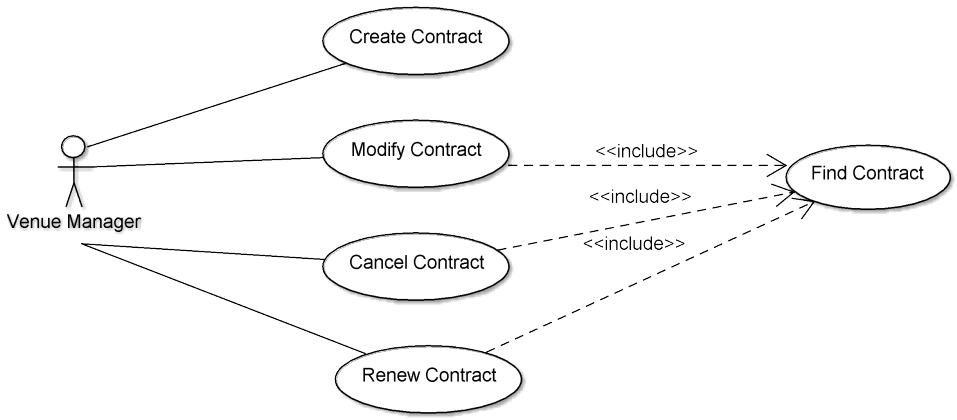
B.15 Renew Agent's Contract

Pre-conditions: The venue manager is logged into the system and there is at least one agent's contract in the system.

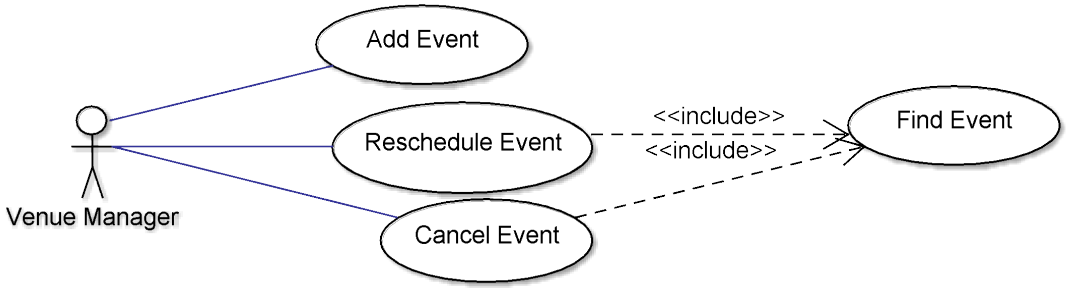
Post-conditions: The contract's end date is extended to the end of the current contract plus another duration of the contract.
Purpose: The venue manager wishes to renew a contract with an agent in order to allow them to sell tickets on the OTS for longer.
Description: The user selects an agent from a list, and a confirmation message appears to make sure that the user has chosen the right agent. When confirmed, the system will then change the end date of the contract in the system to that of the end of the current contract plus the duration of the contract and notify the user when the changes have been processed.

B.16 Diagrams

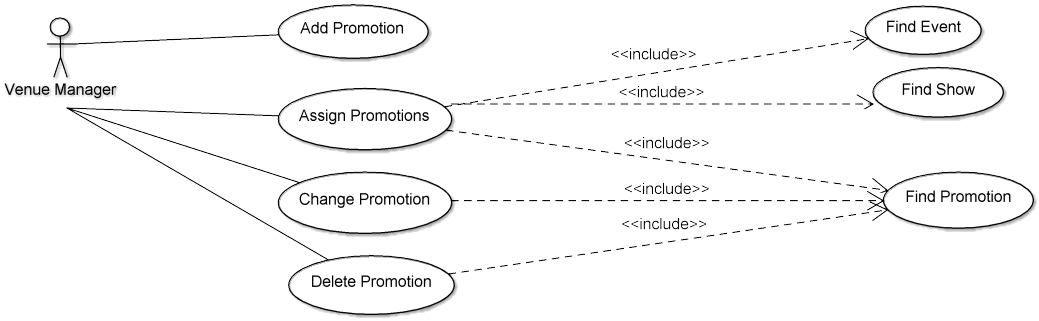
Contract-related use cases:



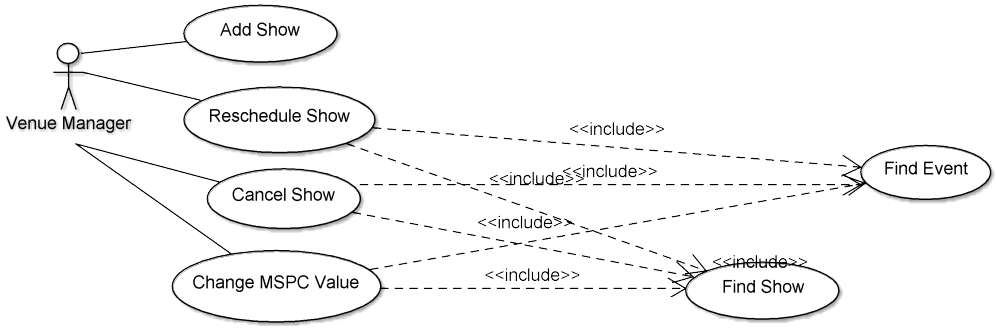
Event-related use cases:



Promotion-related use cases:



Show-related use cases:



C The Agent

D Consolidated Model

Identifying Classes

Data Dictionary

Left	Centre	Right
Text	Text	Text

Report

Section B

Implementation

Requirements

A Expected Functional

- Item

B Non Functional

- Item

Design Class Diagram

Sequence Models

Detailed Design with Pseudo-code

Implementation in Java

Section C

Appendix

#	Author	Date	Commit Message	Files	++	--
1	James Harris	2017-11-19	Initial commit	3	45	0
2	James Harris	2017-11-19	feat: add assignment brief	1	0	0
3	James Harris	2017-11-19	chore: rename brief	2	0	0
4	Alexandre Costa	2017-11-25	docs: add use case template	1	17	0
5	James Harris	2017-11-26	chore: convert brief to pdf	3	3	2
6	James Harris	2017-11-27	docs: add report skeleton	38	391	0
7	James Harris	2017-11-27	docs: add install/usage instructions	4	21	6
8	James Harris	2017-11-27	chore: dot fill chapter pages in toc	2	2	0
9	James Harris	2017-11-27	fix: change english babel commands to british babel	3	4	1
10	James Harris	2017-11-27	fix: deligaturise double dash in appendix table	3	3	2
11	James Harris	2017-11-27	chore: comment out java highlighting	2	4	2
12	Alexandre Costa	2017-11-27	docs: add text for venue manager use cases	2	305	18
13	James Harris	2017-11-28	chore: remove use case template	1	0	17
14	Alexandre Costa	2017-11-29	feat: add venue manager use case diagrams	4	0	0
15	Alexandre Costa	2017-11-29	chore: add common use cases argoUML file	2	4599	0
16	Alexandre Costa	2017-11-29	feat: add manager use cases	2	24	0
17	James Harris	2017-11-30	docs: add student IDs (#20)	2	5	4
18	Alex Costa	2017-11-30	chore: rename and move UML file	1	4599	0
19	James Harris	2017-11-30	chore: delete unneeded uml files	3	1	4599
20	Alex Costa	2017-11-30	chore: format manager text	1	0	7
21	Alex Costa	2017-11-30	chore: add image adding comment	3	2	4599
22	James Harris	2017-11-30	docs: add git to readme	1	22	0
23	wopian	2017-11-30	feat: add consumer usecases	33	6958	5059
24	wopian	2017-11-30	chore: update consumer usecase diagram	2	0	0