

## CSIE5142/CSIE4302 Software Engineering

### Homework # 2 105598008 蘇浚賢

Due on 11/09/2016

1. (70%) You have been appointed to develop a Movie Ticket Booking System that allows customer to view movie information including description and show time, to s, to cancel tickets, and to collect tickets. It also allows staffs to add/update/delete/view movie information. For booking tickets, customers will first login as a registered customer. The registered customer then can select the movie and the show time. To book tickets, the system will retrieve and show the current available seats from the reservation database for the selected movie and show time. The customer then can choose his/her favorite seats and make a payment to reserve the tickets. The system will return a reservation number to the customer if the ticket booking is successful. The customer can collect the tickets by showing the reservation number to the staffs in the cinema. The customers may cancel the ticket booking 30 minutes before the show time of the movie and the system will refund the payment back to the credit card (or account) of the customer.

**(a) (10%) Write a user story for booking a movie ticket**

<b>Use Case UC#8</b>	#8 顧客送出訂購需求
<b>Primary Actor</b>	顧客、已註冊顧客、系統
<b>Preconditions</b>	想要訂購票的顧客
<b>Success Guarantees</b>	顧客透過保留號碼在電影院可以領取票根
<b>Main Success Scenario</b>	<ol style="list-style-type: none"><li>1. 顧客可以登入訂購系統，變成已登錄顧客</li><li>2. 已登錄顧客可以瀏覽、選擇電影及場次</li><li>3. 系統必須從資料庫中檢索並顯示出以已登錄顧客選擇的電影及場次條件的目前可取得的座位</li><li>4. 已登錄顧客可以選擇他喜愛的位置</li><li>5. 已登錄顧客可以選擇一種付款方式來保留座位</li><li>6. 如果付款成功，系統必須回傳保留號碼給已登錄顧客</li><li>7. 已註冊客戶可以憑保留號碼和電影院的員工領取票</li></ol>

<p style="text-align: center;"><b>Exceptions</b></p>	<p>*a. 在任何時間點，當系統失效時：為了支援系統復原能力並修正，確認交易中所有容易受影響的狀態與事件，不論是在情節的哪個步驟都能復原。</p> <ol style="list-style-type: none"> <li>1. 顧客輸入錯誤帳號、密碼組合，必須停留在第 1 步驟</li> <li>2. 已登錄顧客選取無效的電影或場次，必須停留在此步</li> <li>3. 若資料庫抓出資料為空集合，必須顯示資料給已登錄顧客並回第 2 步驟。</li> <li>4. 若該位置在同時間已被其它已登錄顧客購買並保留，必須顯示錯誤訊息並回到第 3 步驟。</li> <li>5. 同以上述續。</li> <li>6. 若付款失敗，必須保留座位 5 分鐘。</li> <li>6a. 若超過 5 分鐘，必須回到第 3 步驟。</li> <li>7. 若遺失保留號碼可藉由帳號、密碼領向員工取票。</li> <li>2-6a. 若無登入，回到第 1 步驟</li> <li>2-6b. 若要取消購票，必須在開演前 30 分鐘提交取消，並將款項退還於信用或者已登錄顧客餘額中。</li> </ol>
--	--

**(b) (10%) List both functional and non-functional user requirements for the system. Make proper assumptions if necessary.**

**Function:**

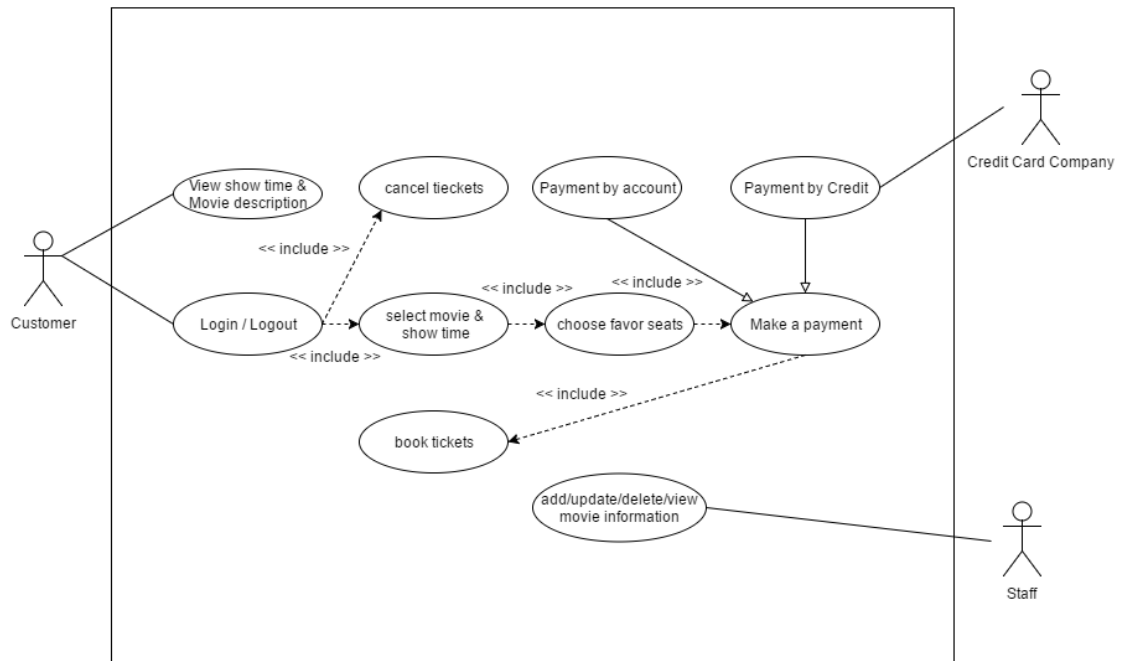
1. 顧客應該可以瀏覽電影資訊，包括電影描述及場次。
2. 顧客應該可以登錄訂票系統。
3. 顧客應該可以註冊後登錄而成為已登錄顧客
4. 已登錄顧客可以登出成為顧客
5. 已登錄顧客應該可以選擇欲觀看電影及場次。
6. 已登錄顧客應該可以在選擇欲觀看電影及場次後，觀看目前剩餘座位。
7. 已登錄顧客應該可以付款後保留票。
8. 已登錄顧客應該可以在付款後取得保留號碼。
9. 已登錄顧客應該可以藉由向電影院員工顯示保留號碼後取票。
10. 員工應該可以登錄員工帳號。
11. 員工應該可以新增/刪除/修改/查詢電影資訊。
12. 已登錄顧客應該可以在電影開播前 30 分鐘取消購票，並將金額退還於信用卡或已登錄顧客中。

**Non-function:**

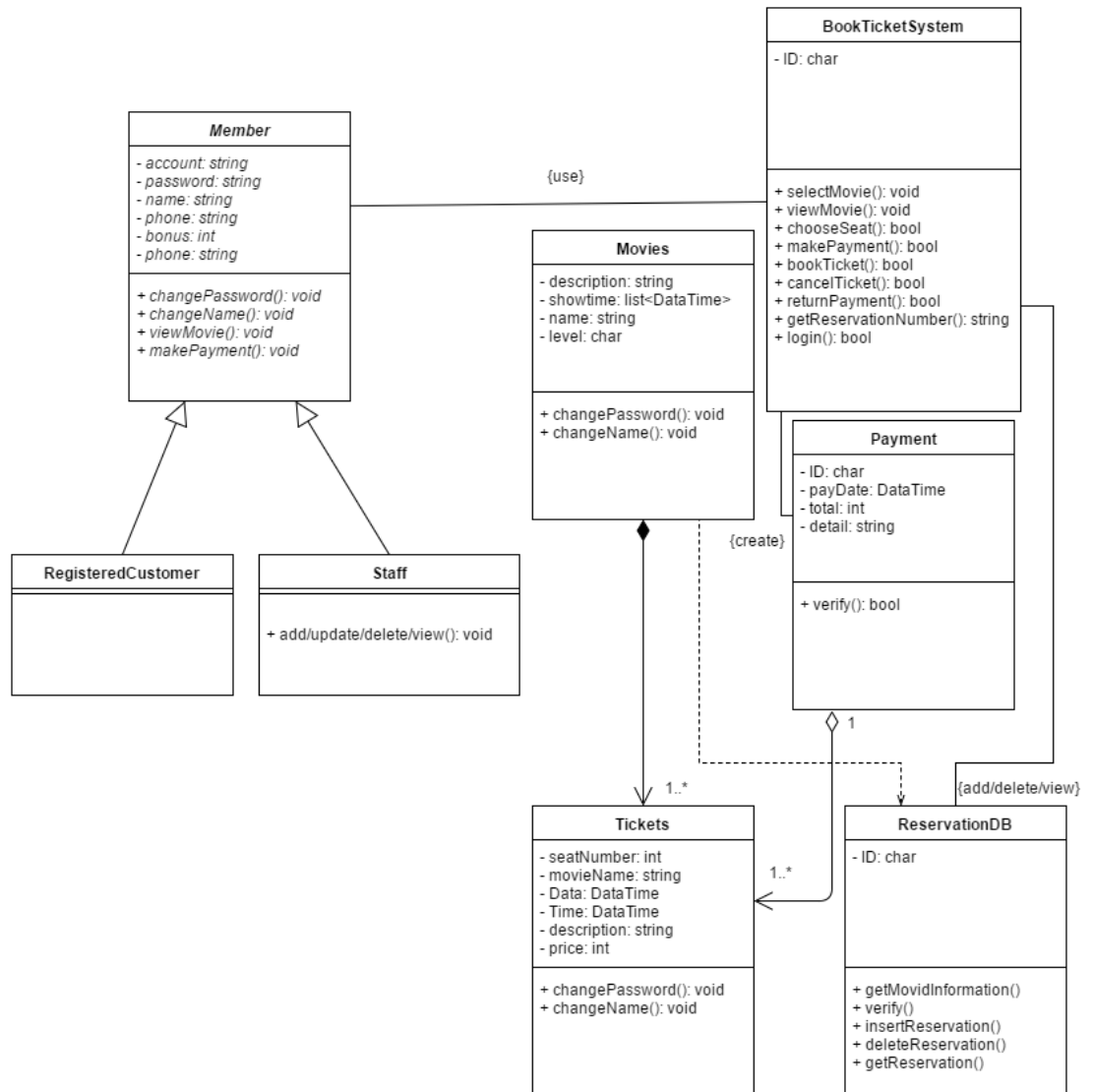
1. 系統反應時間必須小於在 0.04ms。
2. 系統必須可以復原成特定時間狀態。

3. 系統必須同時容納最大量 1000 人。
4. 系統必須符合軟體可測試性
5. 系統必須定時備份
6. 系統必須保證隱私權問題

(c) (10%) Draw the use case diagram for the system. The use case diagram must have at least two actors, such as customer and staff.

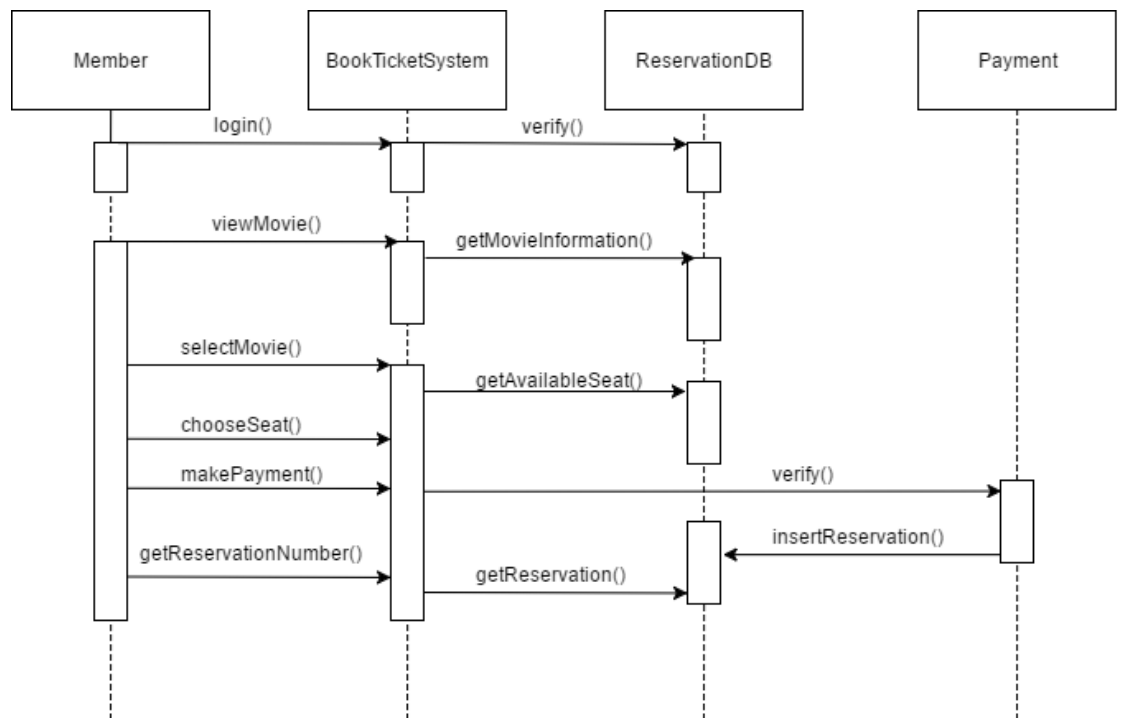


(d) (20%) Identified the possible classes such as **RegisteredCustomer**, **Movies**, **BookTicketSystem**, **Payment**, **ReservationDB**, and draw the UML class diagram for the system. Please give essential **attributes** and **methods** for each class and specify their visibility. Show appropriate relationships between the classes and specify the multiplicity for each relationship.

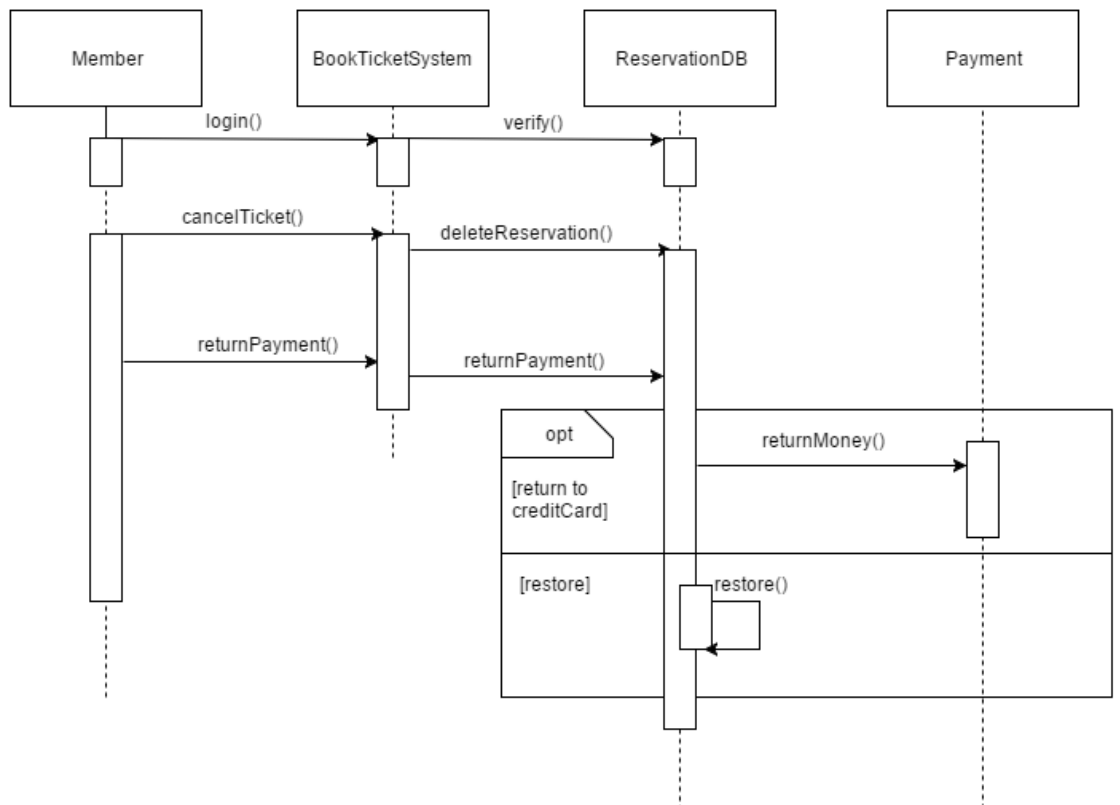


- (e) (20%) Draw the UML sequence diagram based on your class diagram for the scenarios 1) *a customer successfully books a movie ticket*; and 2) *a customer cancels the booked tickets*. Make proper assumptions if necessary.

**1) a customer successfully books a movie ticket:**

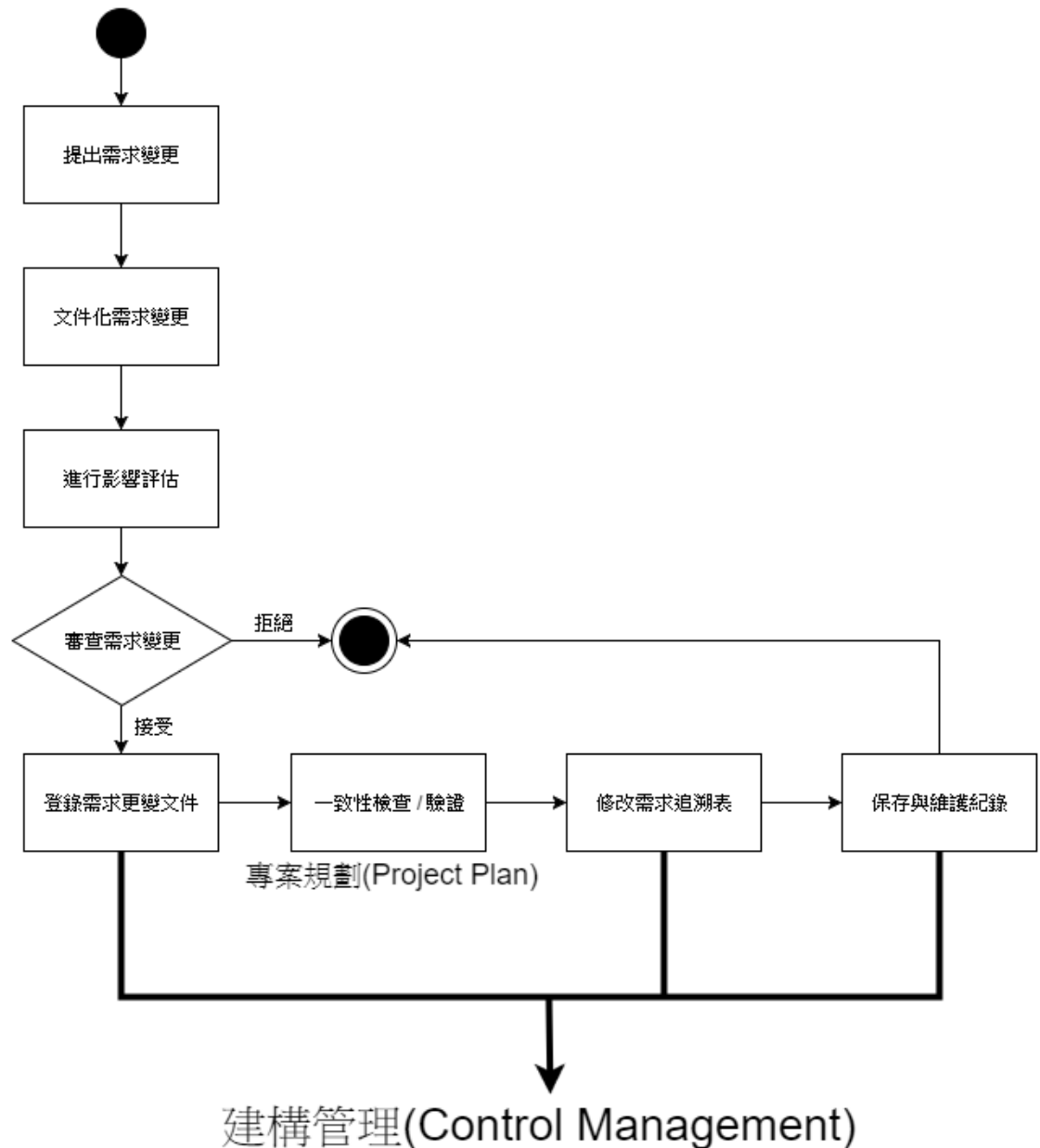


2) *a customer cancels the booked tickets.*



2. (30%) Consider the requirements change management.

(a) (20%) Draw an activity diagram to explain the process of requirements change management.



(b) (10%) What is the traceability matrix? How traceability matrix can be used in the requirements change management?

1.

可追溯矩陣(traceability matrix): 通常是以表格形式的一種文件，用以紀錄及檢查需求是否被滿足的文件。一般被追溯的屬性通常包括：描述、理由、負責人、優先順序、完成時間、狀態等。

2.

在完成「客戶需求列表」後，依產品特性分類，將相同屬性的歸到同一模組之中。因為需要在每個階段去驗證產出符合客戶需求，所以得

透過追溯矩陣關係告訴我們，哪個需求對應到哪個特性或模組，這個模組產出哪份文件，文件又產出哪些 code，透過一層一層追蹤下來，依照手上的工作產品去進行設計，就理論上不會錯誤。