Task 1: Requirements elicitation

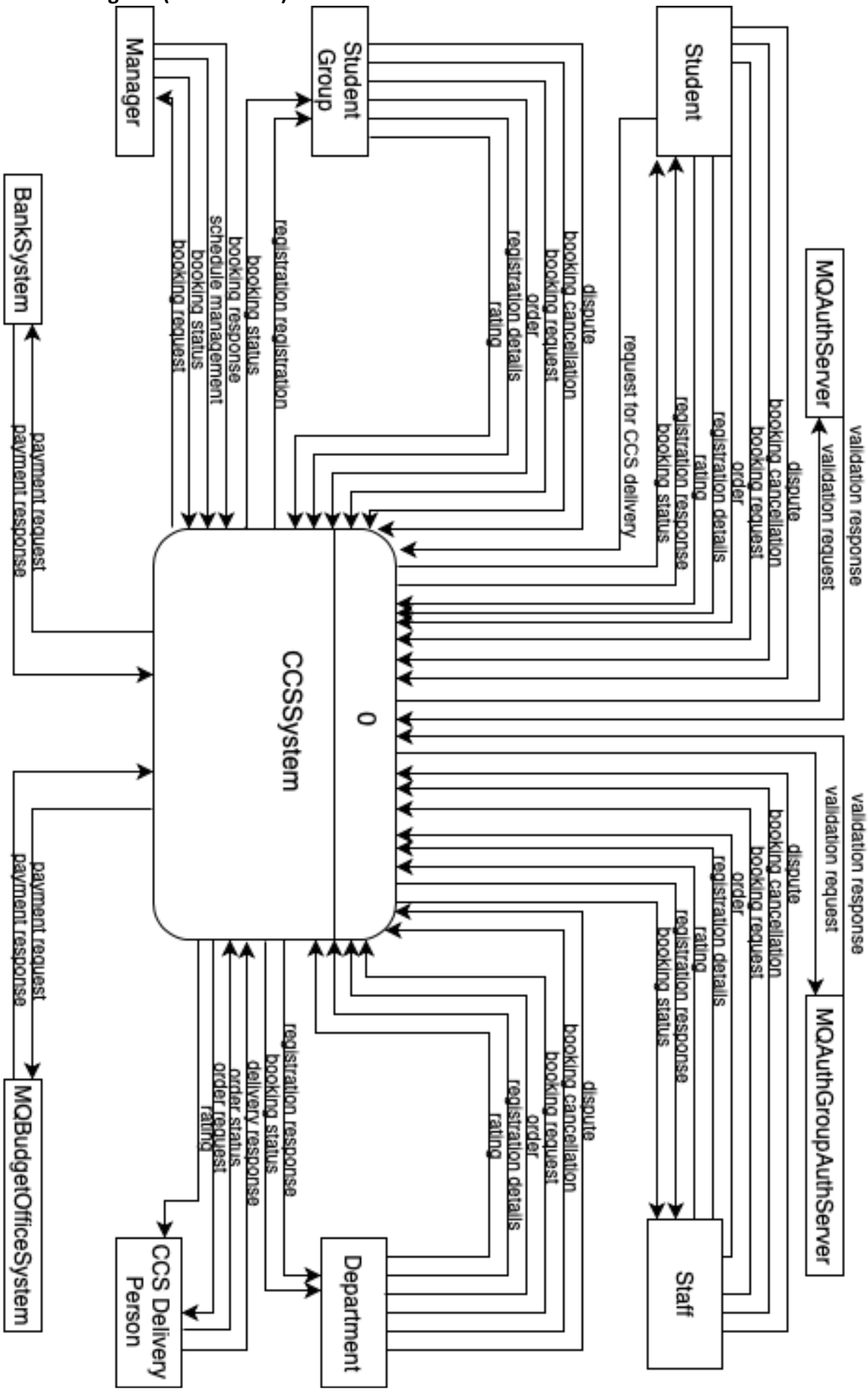
1. **Requirements gathering techniques**
   a. Interviewing: Since the Campus Common System will directly affect the students and staff of the university, it is necessary to seek the opinions of key users and their feeling about the current state of Campus Common, as opinions can be more telling than facts.
   b. Questionnaires: It is likely CCS will be used by a large number of people and groups. In addition to economically gathering information from a large number of people, questionnaires will also be useful in reflecting attitudes about the current system. Questionnaires will aid in quantifying a particular attitude that was revealed in the interviews and hence, can make it easier to prioritise problems based on their prevalence.
   c. Observation (sampling): A difficulty that may be faced in the requirements elicitation phase is ensuring the users (students, staff, student groups and departments) are widely represented to ensure the needs of everyone is being considered. Observing a sample of the population in their physical environment when done effectively, can reduce bias by providing valuable information about the population as a whole, rather than a particular subset (students). It is also less disruptive, controls costs, speeds the requirements elicitation process and adds to the value of the information gathered in the above methods.
   d. Investigating: The above techniques does not consider a wide range of hard data that can reveal information that other methods cannot, which is why investigating the Campus Common through quantitative documents is essential. Analysing the reports used for decision making as well as its performance reports, records and data capture forms can indicate the past and current states of the organisation and where its members deem it's heading towards.
   e. Prototyping: Is essential in this scenario as it provides details as to how users will use and respond to the Campus Common System and gives an indication of how well CCS suits and meets the needs of the users. This information can also be obtained through feedback sheets, interviews and questionnaires.
2. **Strategy**
   a. The first stage will involve observing a sample of the population, including all levels of the organisation (students, student groups, staff and departments). The observation will comprise of a reasonable sample size and will focus on how users interact with their current system.
   b. The second stage uses interactive methods to elicit information about current problems, needs and attitudes. Considering the information gathered in the first stage about how users interact with the current system, the second stage will concentrate on the issues and concerns surrounding that interaction. Interviews will be conducted first to gather certain sentiments about the current system. Then, considering the opinions and attitudes expressed in the interviews, questionnaires will be performed. The questionnaires will enquire about the concerns and needs raised in the interviews, so the users can express the degree to which they agree/disagree with the sentiment.
   c. The third stage is used to confirm or provide additional information to what has already been gathered in stages 1-2. Production, performance and summary reports will be analysed to provide background information about the organisation of the current system. Records are also investigated to check for any possible errors in amounts and totals, the design of the record form, the quantity and type of transactions and opportunities for certain tasks to be computerised. This provides an indication of the current usability.
   d. The fourth and final stage involves developing a prototype for the users and obtaining feedback. Considering the information elicited in stages 1-3, such as the interaction of users with their current system, their concerns, attitudes and opinions on it, and the usability of their current system, a prototype is developed and modified in successive iterations with an emphasis on the user interface to observe how those same users will interact with the new CCS System. Feedback sheets, interviewing and questionnaires are then conducted to provide information about concerns and difficulties and whether their needs are addressed.

Task 2: Requirements specification

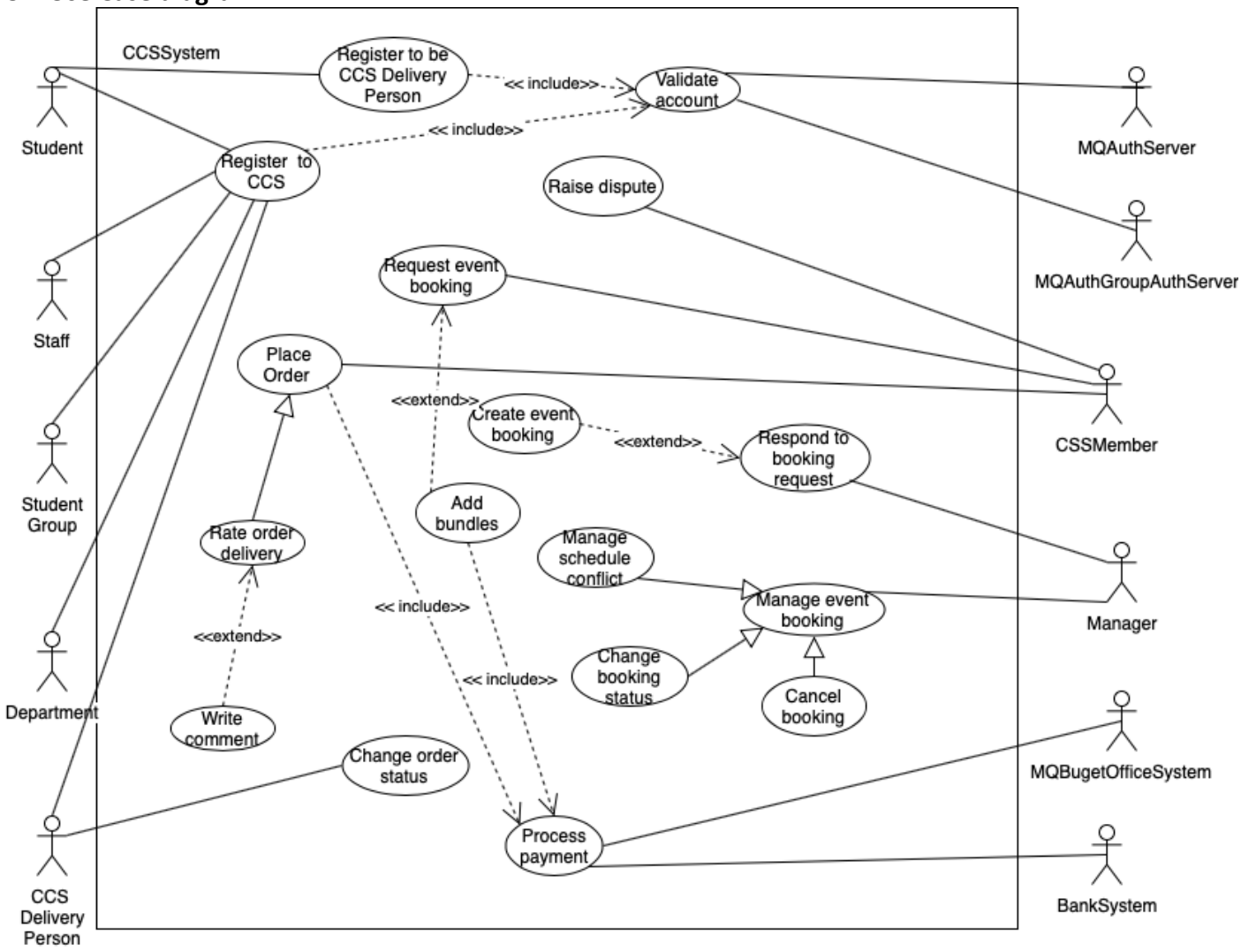
3. **User scenarios**
   a. CCS Delivery Person: On a typical day, I check for any order delivery requests in the CCS System. When I see a delivery request, I respond either accepting or denying the request. Once I collect the order from the restaurant and deliver it to the CCS Member, I access CCS to change the status of the order as delivered. I access CCS to see rating and comment about the delivery a CCS Member may have left.
   b. CCS Manager: On a typical day, I check for any event booking requests in the CCS System. When I see a booking request, I respond either accepting or denying the request. If the request is accepted, I access the create booking event screen, where CCS prompts me to enter the type, date, time and location of the booking. I manage schedule conflicts, ensuring events do not clash and are not booked at the same date, time and location as any other event. I change the status of the event accordingly, whether it is accepted, pending, denied or cancelled.
4. **User stories**
   a. As a MQAuthServer, I want to cross-check if the student or staff has valid identification so that they are able to validate their account to be a CCS Member or a CCS delivery person (for students).
   b. As a BankSystem, I want to process a payment transaction so that a CCS Member can place an order.
   c. As a Student Group, I want to book a space at Campus Common so that it can hold an event
   d. As a Staff, I want to register my details with CCS so that I can become a CCS Member
5. **Functional requirements for the proposed system**
   a. The system shall allow a Manager to create an event
   b. The system shall allow a Student to register their details for CCS membership
   c. The system shall allow a CCS Delivery Person to change the status of the order as delivered
6. **Non-functional requirements for the proposed system**
   a. The system shall respond to any booking request within 12 hours. If the time frame exceeds 12 hours, an estimate of time remaining shall be displayed to the CCS Member after the 12 hours. (measures responsiveness)
   b. The system shall support up to 100 membership requests in a 24-hour period. (measures performance)
   c. The system shall use encryption to deliver data to the Bank System and the MQ Budget Office System. (measures security).

Task 3: Diagrams for different systems perspectives

7. **Context diagram (Level 0 DFD)**



8. **Use Case diagram**

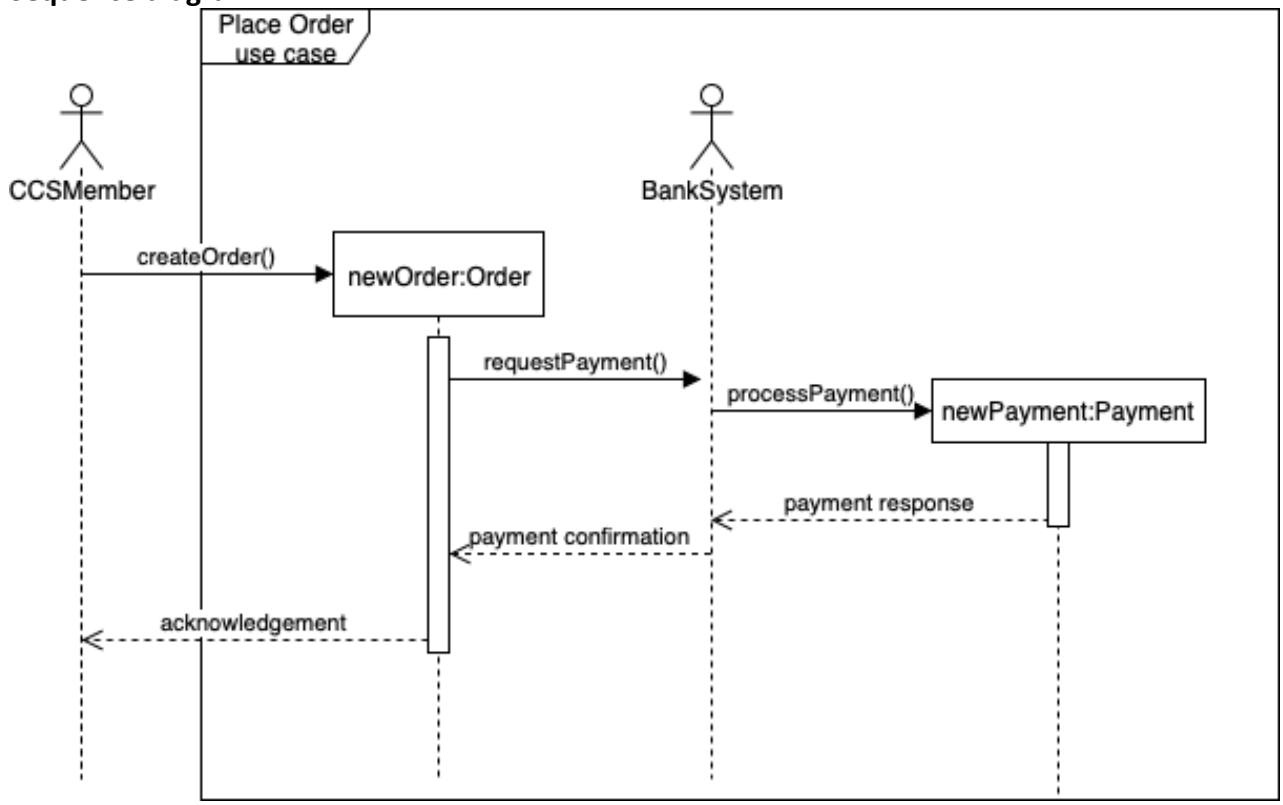


9. **Use Case description**

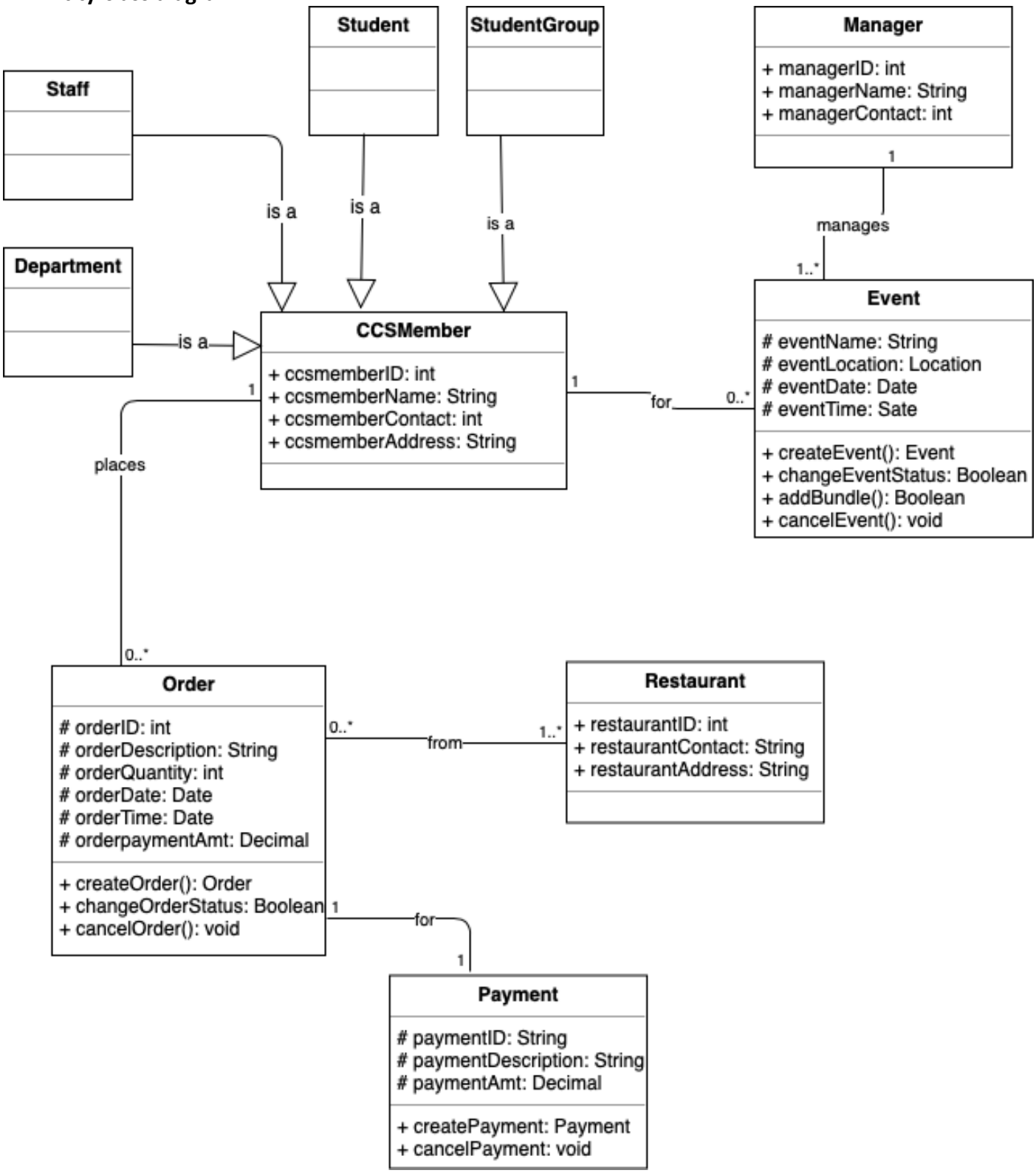
Use Case	Place Order	
Goal	To place an order for food and/or beverages	
Preconditions	CCS Member has all necessary details to enter into the system to place an order	
Success End Condition	Order has been placed successfully	
Failed End Condition	Order goes into 'pending' state and is not placed successfully	
Primary Actors; Secondary Actors	CCS Member MQ Budget Office System, Bank System	
Trigger	CCS Member seeks to order food and/or beverages	
Description / Main Success Scenario	**Step**	**Action**
	1	CCS prompts CCS Member to enter details about the order
	2	CCS member enters order description, order quantity and the date and time of the order
	3	CCS sends a request to Bank System or MQ Budget Office System to process the payment of the order
	4	Bank System or MQ Budget Office System processes payment and sends back a confirmation message

	5	CCS displays a message to show the payment has been successfully processed and order has been successfully placed
Alternative Flows	**Step**	**Branching Action**
	2.a	CCS Member enters a past date and time
	2.b	CCS displays an error message and prompts to re-enter a correct date and time
	4.a	Payment is not successfully processed
	4.b	Bank sends an error message to CCS that payment has not been successfully processed
	4.c	CCS displays an error message to CCS Member that payment has not been successfully processed
	5.a	Order is not placed and is put on a 'pending' state

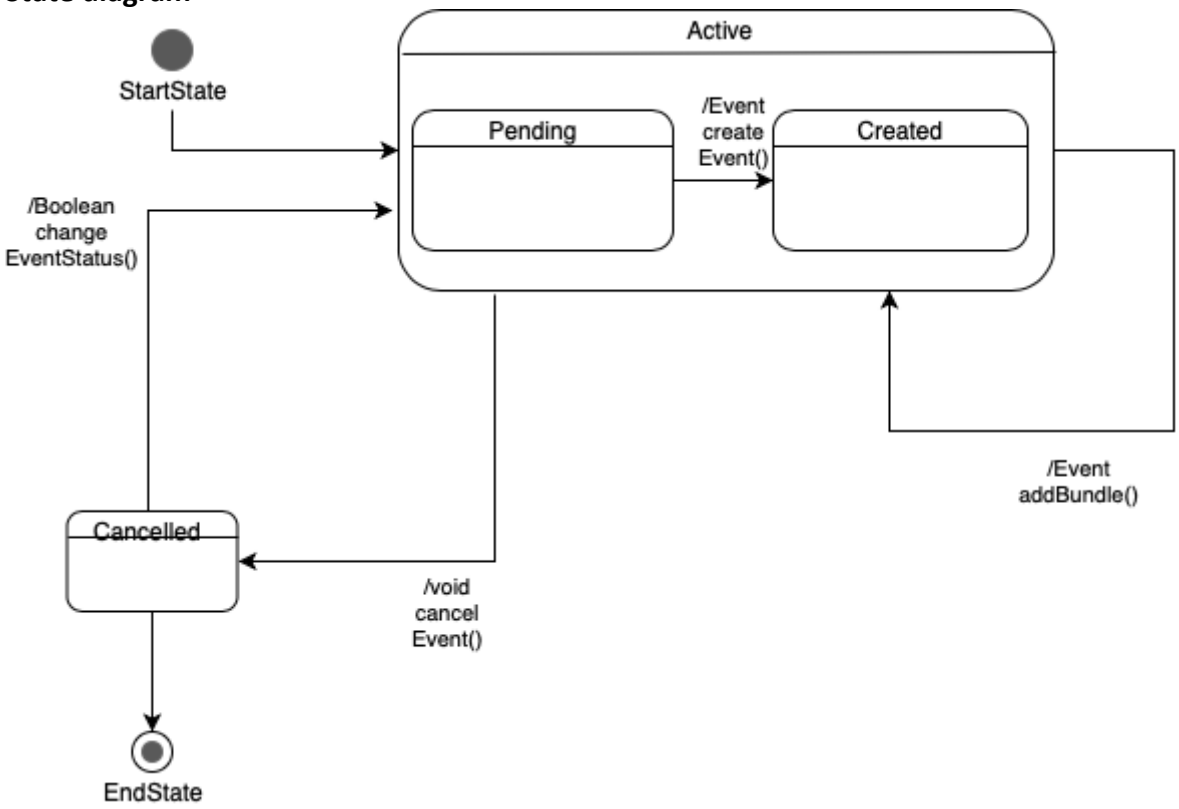
10. **Sequence diagram**



11. **Entity-class diagram**

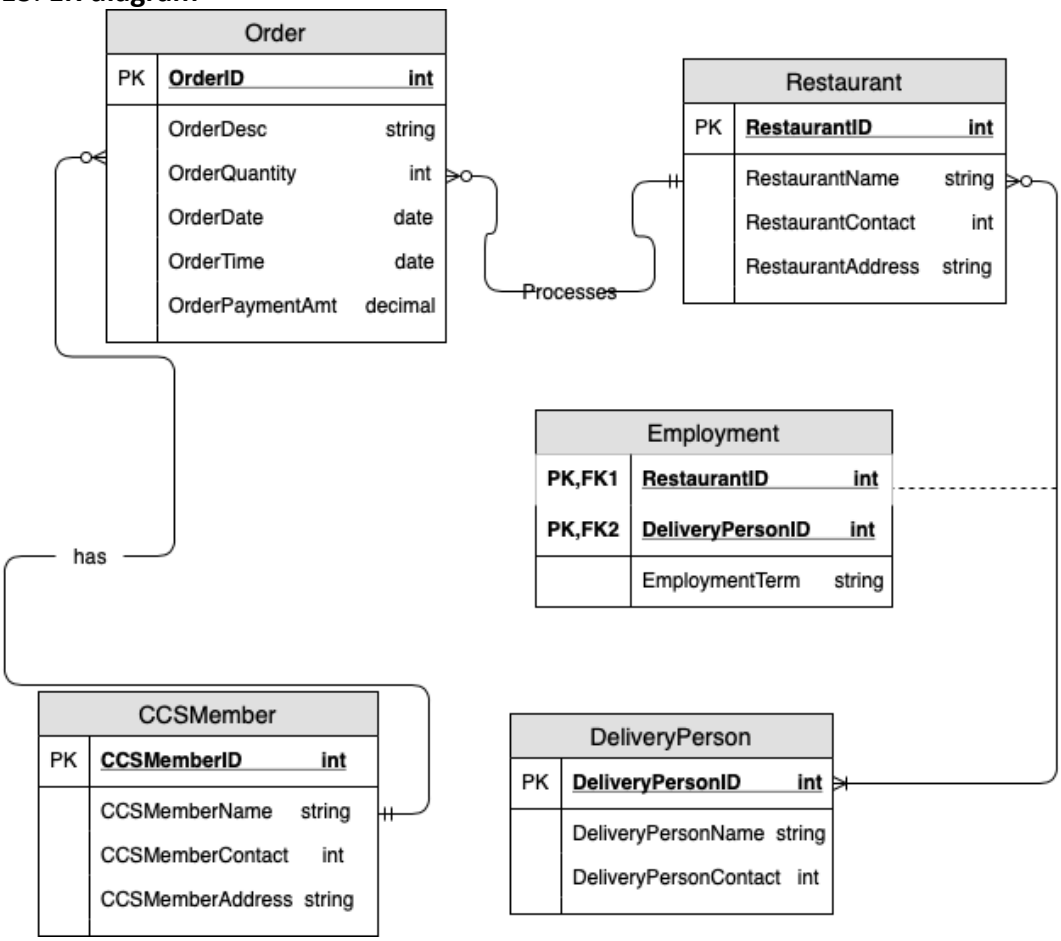


12. **State diagram**



Task 4: Data and storage considerations

13. **ER diagram**



Employment was added at an associative entity as it associates the entities Restaurant and DeliveryPerson in a many to many relationship.

14. **Tables**

CCSMember (CCSMemberID <pk>, CCSMemberName, CCSMemberContact, CCSMemberAddress)
Order (OrderID <pk>, OrderDesc, OrderQuantity, OrderDate, OrderTime, OrderPaymentAmt)
Restaurant (RestaurantID <pk>, RestaurantName, RestaurantContact, RestaurantAddress)
DeliveryPerson (DeliveryPersonID <pk>, DeliveryPersonName, DeliveryPersonContact)
Employment (RestaurantID <pk,fk>, DeliveryPersonID <pk,fk>, EmploymentTerm)