***EXERCISE 10***🡪 <https://ceur-ws.org/Vol-2503/paper2_2.pdf>

**PROMPT:** The following should be textually analyzed and a use case diagram created containing several use cases. Identify the actors, use cases and associations. Please, use the user goal level approach. Please give me the PlantUML code for the use case diagram corresponding to the following text:

***Software has to be developed that supports a company in organizing business trips. The story could be characterized as follows: As an employee, I want to get the permission of a business trip, so that train tickets and hotel are booked. The corresponding use case can be called organizing a trip. The primary actor is Employee, and there are two secondary actors called Manager and Agent. A manager has to give the permission of a business trip, and an agent supports the booking of hotel and train tickets. The main success scenario could be: ask for permission, wait for an answer, ask for booking, wait for documents, travel.***

ChatGPT

**ChatGPT**

**User Goal Level Approach:**

**Actors:**

1. Employee
2. Manager
3. Agent

**User Goals:**

1. **Get Permission for Business Trip:**
   * *Actor:* Employee
   * *Description:* Employees want to obtain permission for a business trip.
2. **Give Permission for Business Trip:**
   * *Actor:* Manager
   * *Description:* Managers want to provide permission for employees to go on a business trip.
3. **Support Booking of Hotel and Train Ticket:**
   * *Actor:* Agent
   * *Description:* Agents want to assist in the booking process for hotels and train tickets for business trips.
4. **Organize a Trip:**
   * *Actor:* Employee
   * *Description:* Employees want to organize a business trip, including obtaining permission, booking hotels, and train tickets.

**Associations:**

1. Employee asks for permission.
2. Manager gives permission.
3. Employee asks for booking assistance.
4. Agent waits for documents.
5. Employee travels.

Now, let's create the PlantUML code for the use case diagram:

@startuml

left to right direction

actor "Employee" as Employee

actor "Manager" as Manager

actor "Agent" as Agent

rectangle "Business Trip Organization" {

usecase "Get Permission for Business Trip" as UC1

usecase "Give Permission for Business Trip" as UC2

usecase "Support Booking of Hotel and Train Ticket" as UC3

usecase "Organize a Trip" as UC4

}

Employee --> UC1

Manager --> UC2

Agent --> UC3

Employee --> UC4

Agent --> UC4

@enduml

**PROMPT:** First, I think that the employee is the only main actor,, and manager and agent are supporting actors, or secondary actors. Then, I think that just one use case for "organizing a trip" could be shared by both employee and manager, avoiding the inclusion of the use cases of getting permission and giving permission, since the main goal here is to organize a trip. Moreover, I think that there could be two "inclusion" associations between the use case "organize a trip" and the use cases "book a hotel" and "Book a train". This are two separate goals of the agent.

ChatGPT

**ChatGPT**

Thank you for the clarification. Based on your feedback, let's modify the analysis and PlantUML code accordingly:

**User Goal Level Approach:**

**Actors:**

1. Employee (Main Actor)
2. Manager (Supporting Actor)
3. Agent (Supporting Actor)

**User Goals:**

1. **Organize a Trip:**
   * *Actor:* Employee
   * *Description:* Employees want to organize a business trip, including obtaining permission, booking hotels, and train tickets.
2. **Book a Hotel:**
   * *Actor:* Agent
   * *Description:* Agents want to assist in booking hotels for business trips.
3. **Book a Train:**
   * *Actor:* Agent
   * *Description:* Agents want to assist in booking train tickets for business trips.

**Associations:**

1. Employee organizes a trip.
2. Agent includes booking a hotel in the trip organization.
3. Agent includes booking a train in the trip organization.

Now, let's create the updated PlantUML code for the use case diagram:

@startuml

left to right direction

actor "Employee" as Employee

actor "Manager" as Manager

actor "Agent" as Agent

rectangle "Business Trip Organization" {

usecase "Organize a Trip" as UC1

usecase "Book a Hotel" as UC2

usecase "Book a Train" as UC3

}

Employee --> UC1

Manager --> UC1

Agent --> UC1

Agent --> UC2

Agent --> UC3

@enduml

**PROMPT:** You didn’t put the extend arrows that I asked you before

ChatGPT

**ChatGPT**

Apologies for the oversight. I've added the "extend" arrows to represent the extension relationships between the "Organize a Trip" use case and the "Book a Hotel" and "Book a Train" use cases. Here's the updated PlantUML code:

@startuml

left to right direction

actor "Employee" as Employee

actor "Manager" as Manager

actor "Agent" as Agent

rectangle "Business Trip Organization" {

usecase "Organize a Trip" as UC1

usecase "Book a Hotel" as UC2

usecase "Book a Train" as UC3

}

Employee --> UC1

Manager --> UC1

Agent --> UC1

Agent --> UC2

Agent --> UC3

UC1 --> UC2 : <<extend>>

UC1 --> UC3 : <<extend>>

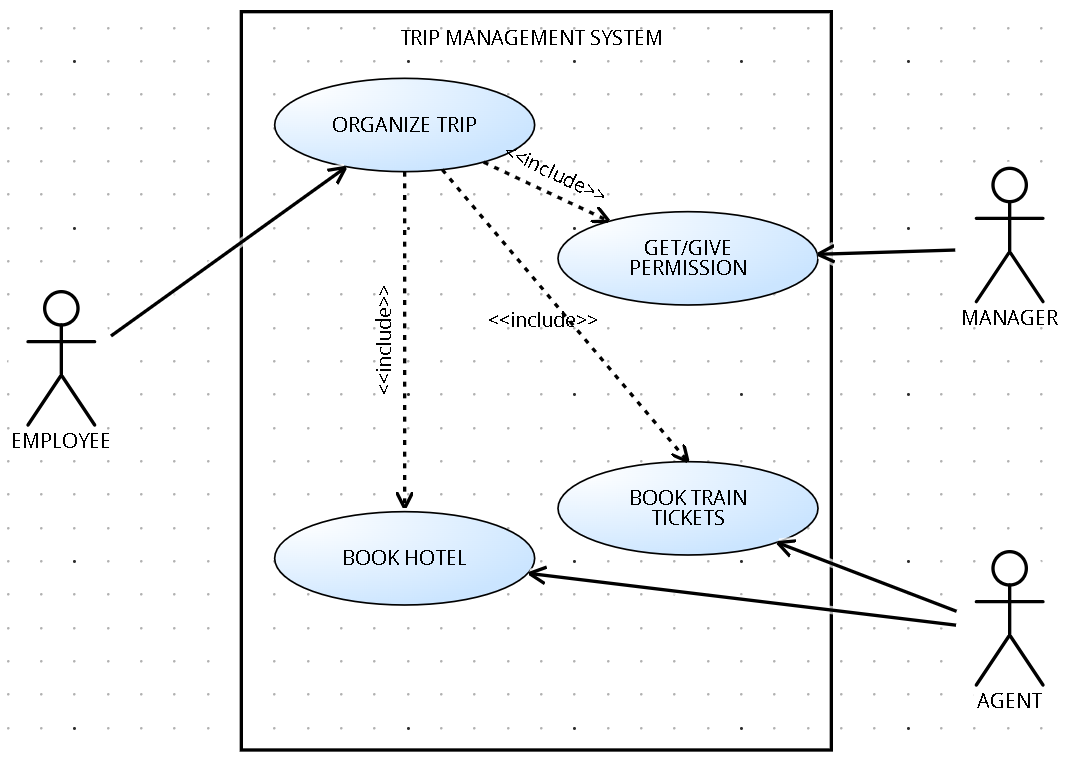
@enduml

**Analysis:**

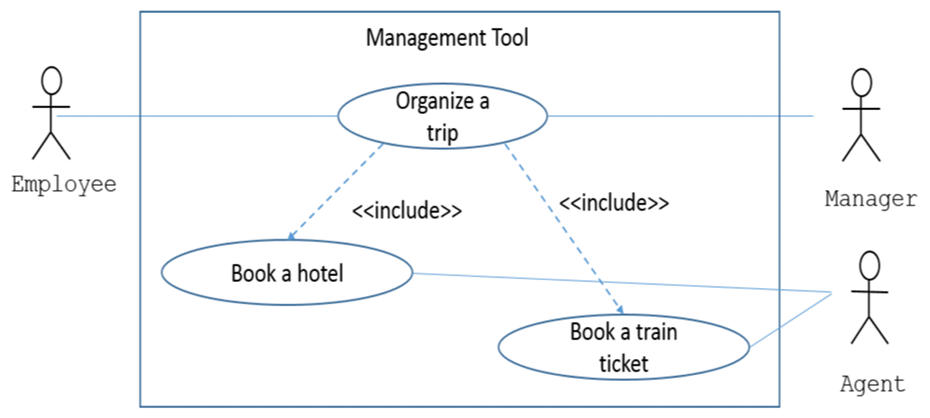
This exercise, while simple in terms of the quantity of actors and use cases, posed a challenge for ChatGPT to achieve the optimal solution. The generated code included unnecessary use cases for the actions of getting permission (for employee) and granting permission (for manager). These actions are secondary and can be encompassed in the overarching goal use case "organizing a trip." Additionally, ChatGPT combined the actions of booking a hotel and booking a train into a single use case. While not entirely incorrect, a more detailed approach would consider them as two separate use cases, especially from the perspective of user goal level. Alternatively, a summary-level approach could have grouped them together into one use case.

Furthermore, ChatGPT demonstrated a limitation in assessing when there should be an "include" or "extend" association between use cases. In the context of organizing a trip, it is necessary to have both booked the hotel and arranged transportation. Therefore, these use cases should be included within the "organize a trip" use case for a higher-quality Use Case Diagram (UCD). This optimal UCD was only achieved after three prompts, highlighting the need for iterative refinement and an understanding of the solution's deficiencies.

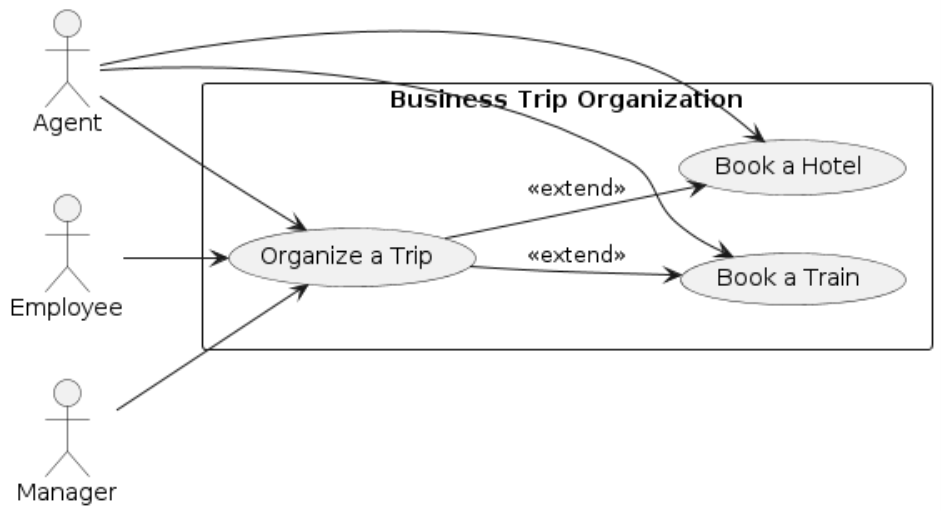
**MY SOLUTION:**

****

**REFERENCE SOLUTION:**

****

**LLM SOLUTION:**

****