SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
(Deemed to be **University** *u/s 3 of UGC Act, 1956)*

Department of Computer Science Engineering SRMIST,

Kattankulathur – 603 203

Sub Code & Name: 18CSC202J- Object Oriented Design and Programming

Assignment Number	01
Name	Anubhav Mishra
Register Number	RA1911029010023
Teammate Name	Taha Baba
Teammate Register Number	RA1911029010029
Domain Name	Airport Check in and Security Screening

Abstract:

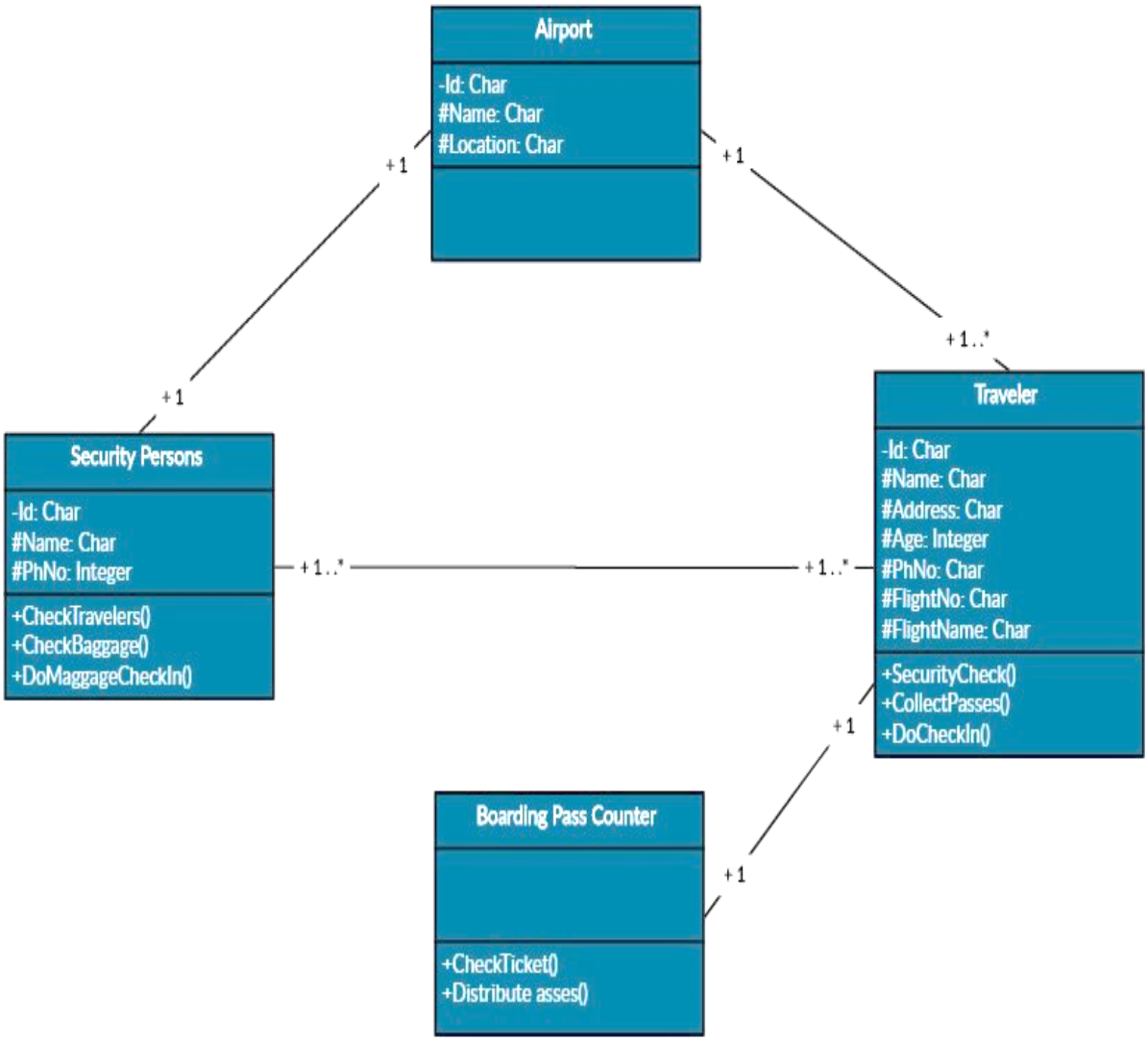
Airport security measures serve to protect the traveling public, crew and aircraft. According to the TSA over 600 million passengers travel on commercial airlines and more than 700 million pieces of luggage are screened each year. With such a large number of people traveling, airports and aircraft have become natural targets for terrorists.

Airport security measures can be grouped into two types; standardized screening techniques, which all passengers must undergo and elevated-risk screening for which only a sub-set of passengers are selected. In the current study, an undergraduate sample was surveyed regarding the professionalism of security screening staff, as well as perceived safety, threat to dignity, and enplanement intentions, following standard and elevated-risk screening measures. Professionalism and safety were positively correlated with enplanement intentions, and dignity threat was negatively associated with perceived safety. As the perceived safety from the use of a security measure decreased, enplanement intentions also decreased. Notably, when a screening measure is perceived as having negative consequences (e.g., threatening one's sense of dignity) the safety of the measure is personally invalidated. Airport security screening processes are essential to ensure the safety of passengers and the aviation industry.

The level of airport security is continuously improving with the help of advanced technology and trained security officers. However, airports have witnessed a significant increase in the number of passengers, which makes optimal security screening processes costly to implement for airport operations, and time-consuming for passengers and airlines. Different methods have been proposed to optimise the queueing process, reduce processing time, and strike a balance between security and time delay. This paper reviews the existing methods used to optimise the security process at airports, the technology being used, the importance of experienced security officers, and the impact of the screening process on passengers and the economy.

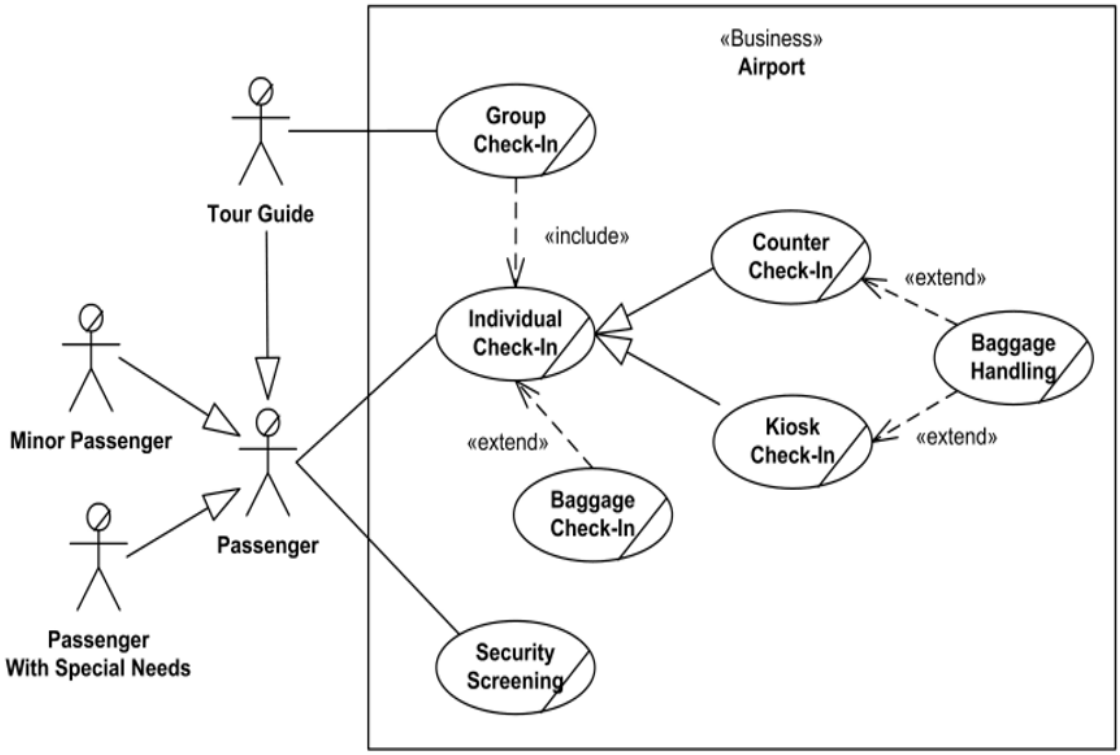
The check-in area of airport terminal is one of the busiest sections at airports at certain periods. The passengers are subjected to queues and delays during the check-in process. These delays and queues are due to constraints in the capacity of service facilities. It is mandatory to carry a government issued photo identification (ID) proof along with the E-Ticket copy. The same is verified by the airport security as well as the airline at the check in counter. On presenting your photo ID proof and E-Ticket copy, your boarding pass will be issued.

UML class diagram



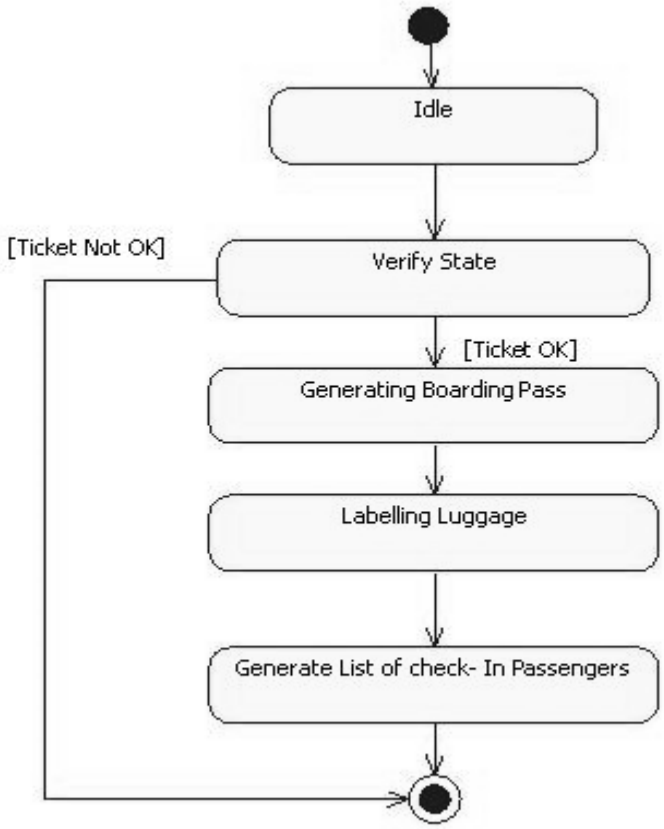
UML Use Case Diagrams:

Four actors using Airport interfaces are Tour Guide, passenger, Minor passenger, Passenger with special needs (e.g. with disabilities), all playing external roles in relation to airport business. Passenger uses a subset of functions available to the Tour Guide. All top-level use cases shown are abstract as each represents some group or "package" of administrative functionality. Top level use case diagram for the Airport Management system:



STATE CHART DIAGRAM FOR CHECK-IN EMPLOYEE [AS OBJECT]

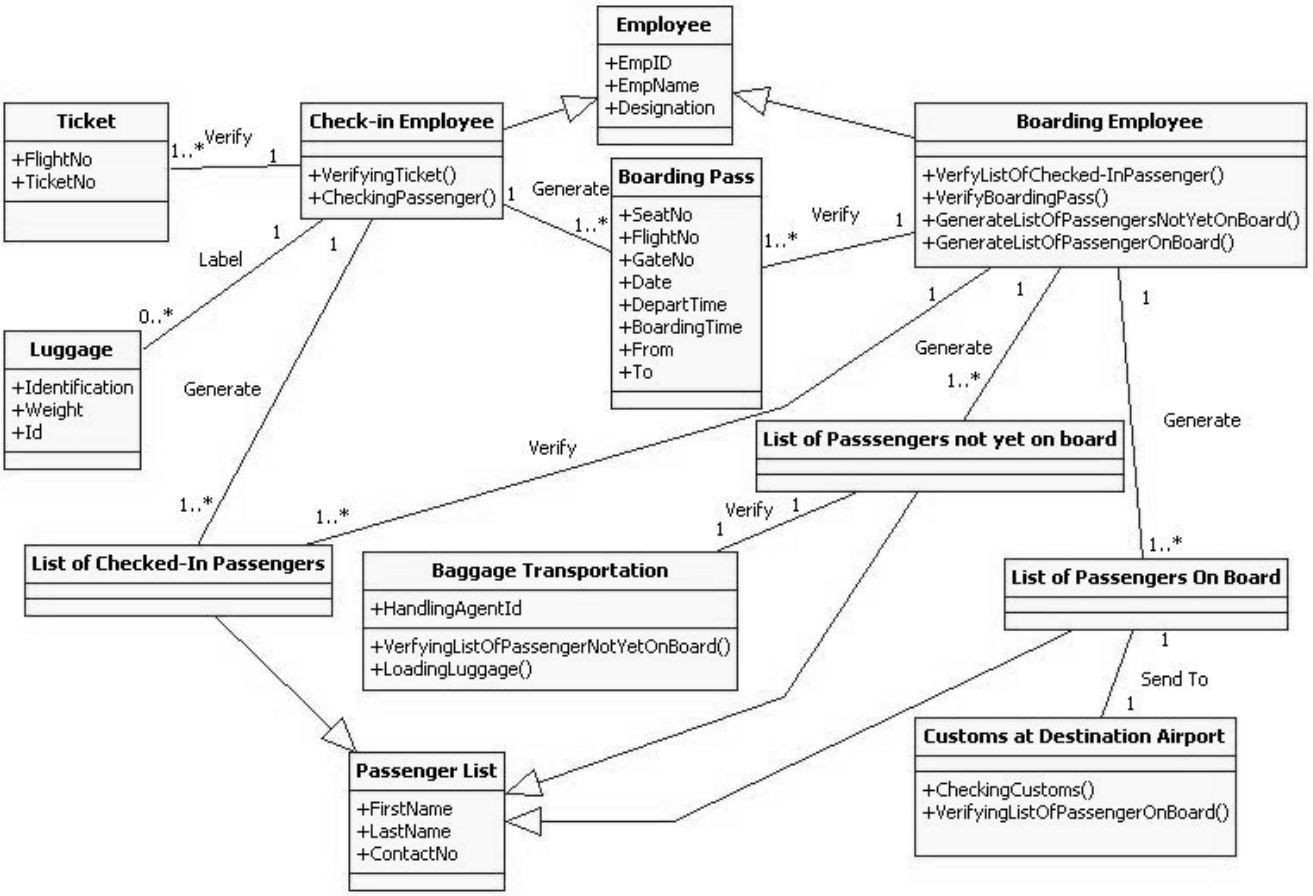
The initial state in this system is idle, which is then transited into verify state. The verify state is followed by a Guard condition which implies the ticket has been verified. If the guard condition is true then only the next transition takes place otherwise it will follow the 'Ticket not OK' guard condition and transition will end by reaching directly to its final state. Here verifies state is the operational state when it is not followed the idle(object) will be directed to the fault state. Generating Boarding Pass, Labelling Luggage are two more states after the verification state is completed. Generate list of check in passengers is the last state after which it reaches to its final state and the transition is complete.



CLASS DIAGRAM FOR AIRPORT CHECK-IN AND BOARDING OF PASSENGERS

Actors are Passenger, Tour Guide, Minor (Child), Passenger with Special Needs (e.g. with disabilities), all playing external roles in relation to airport business.

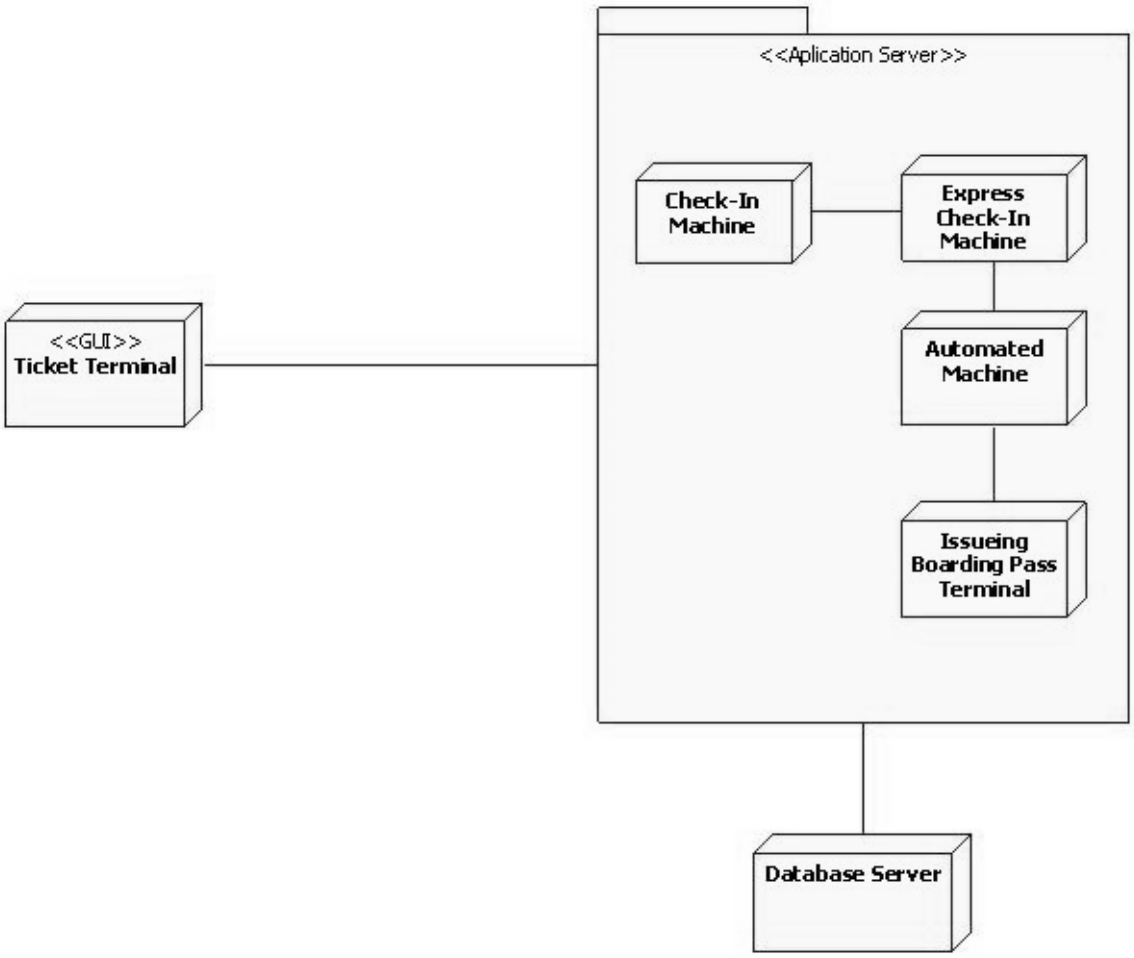
Use cases Baggage Check-in and Baggage Handling extend Check-In use cases, because passenger might have no luggage, so baggage check-in and handling are optional.



DEPLOYMENT DIAGRAM FOR AIRPORT CHECK-IN AND BOARDING OF PASSENGERS

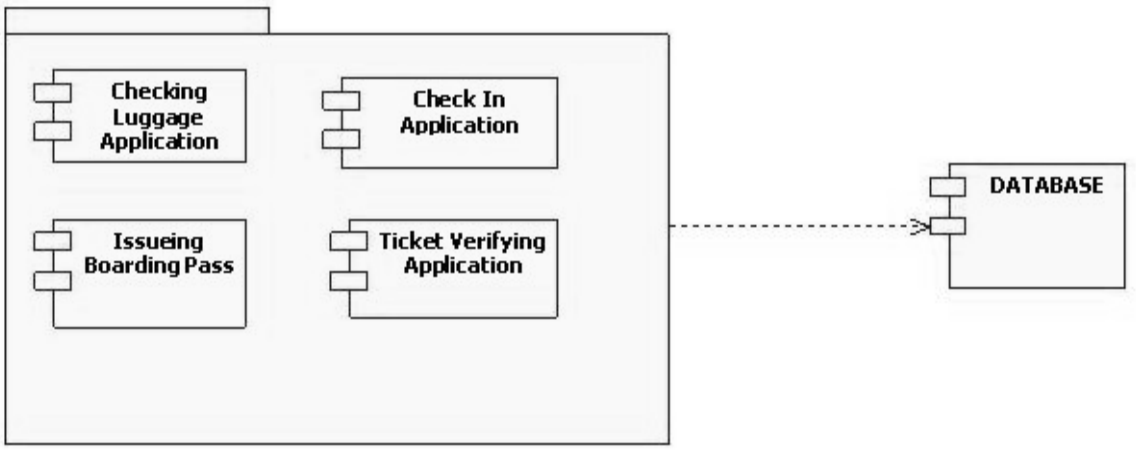
The deployment diagram is used to give a physical view of system. This deployment diagram is a client-server system. Here Ticket terminal acts as a client from which input is received.

There are four nodes in the server with their respective association. The Check in Machine, Express check in Machine, Automated machine and Issuing boarding pass Terminal are four nodes of the server of the respective airport. The application server contains a cache server for a temporary functionality, apart from it there lies a data base server in which all the information of traveller is stored to get a boarding pass.



COMPONENT DIAGRAM FOR AIRPORT CHECK-IN AND BOARDING OF PASSENGERS

The component diagram represents the components of the system and how one software works with the other. The component either recieves or gives information. Here the component contains interfaces which are checking luggage application, check in application, Issueing Boarding pass and Ticket verification application. All these components are dependent on the database for the information and have to wait for the information from database to act on.



Department of Computer Science Engineering

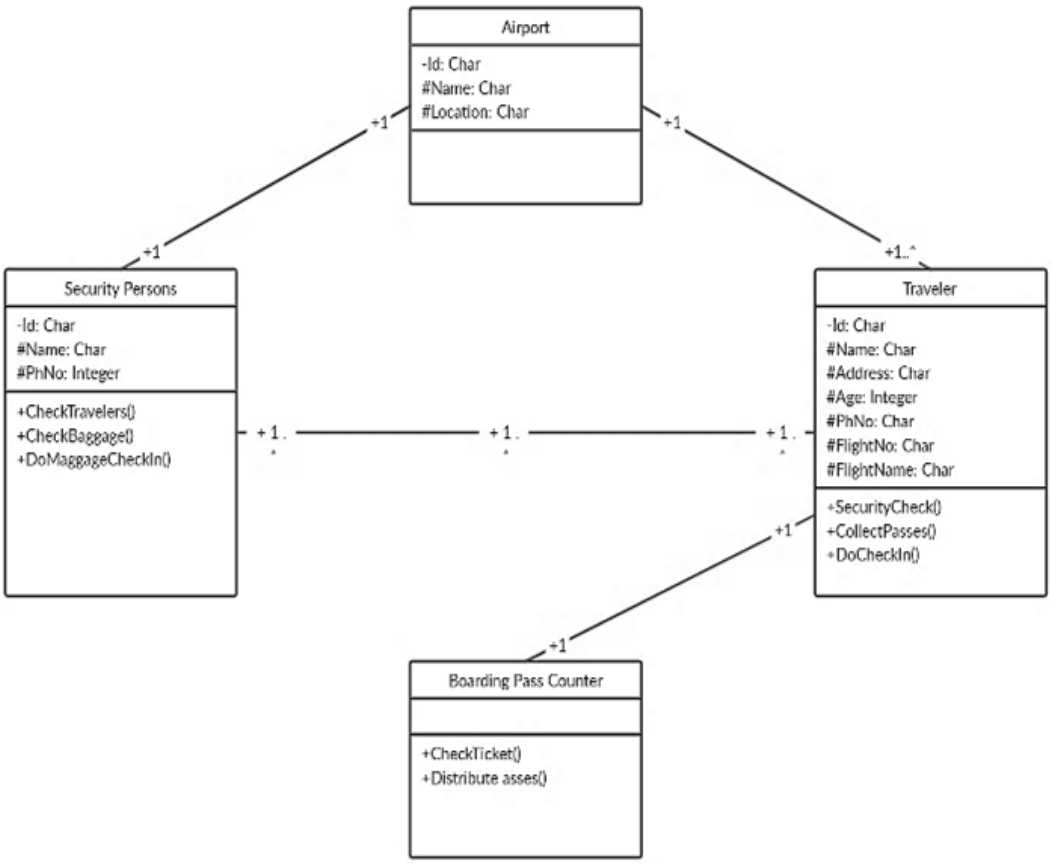
SRMIST, Kattankulathur – 603 203

Sub Code & Name: 18CSC202J- Object Oriented Design and Programming

Assignment Number	02
Name	Anubhav Mishra
Register Number	RA1911029010023
Teammate Name	Taha Baba
Teammate Register Number	RA1911029010029
Domain Name	Airport Check in and Security Screening

1. Class diagram

Actors are Passenger, Tour Guide, Minor (Child), Passenger with Special Needs (e.g. with disabilities), all playing external roles in relation to airport business.
Use cases Baggage Check-in and Baggage Handling extend Check-In use cases, because passenger might have no luggage, so baggage check-in and handling are optional.
Security checks the baggage and the personnel to make sure both are safe for travel

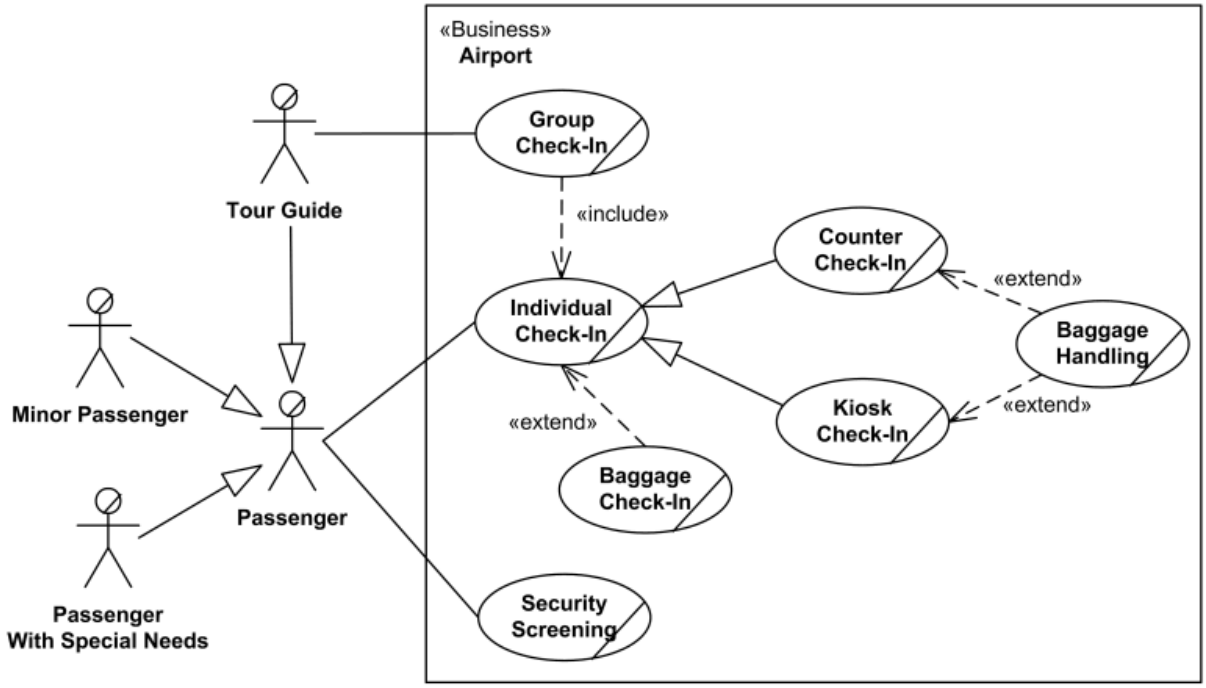


2. Use case diagram

Actors are Passenger, Tour Guide, Minor (Child), Passenger with Special Needs (e.g. with disabilities), all playing external roles in relation to airport business.

use cases are Individual Check-In, Group Check-In (for groups of tourists), Security Screening, etc. - representing business functions or processes taking place in airport and serving the needs of passengers.

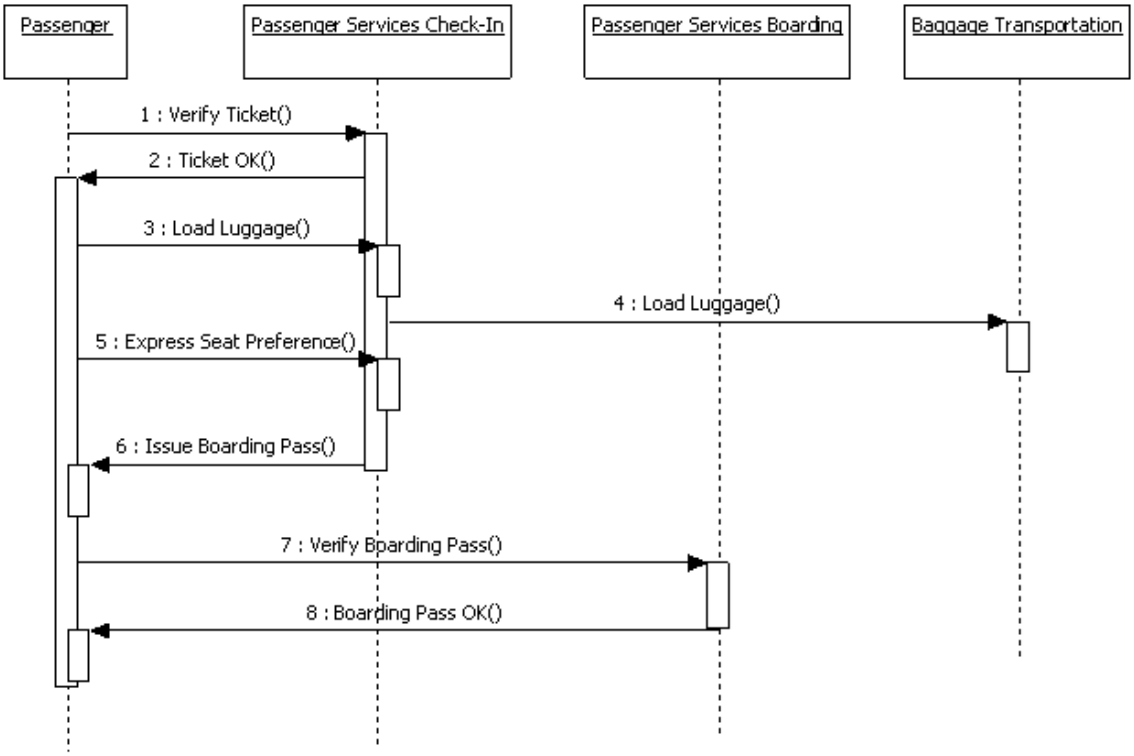
use cases Baggage Check-in and Baggage Handling extend Check-In use cases, because passenger might have no luggage, so baggage check-in and handling are optional.



3. Sequence diagram

A sequence diagram is a type of interaction diagram because it describes how—and in what order—a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process. Sequence diagrams are sometimes known as event diagrams or event scenarios.

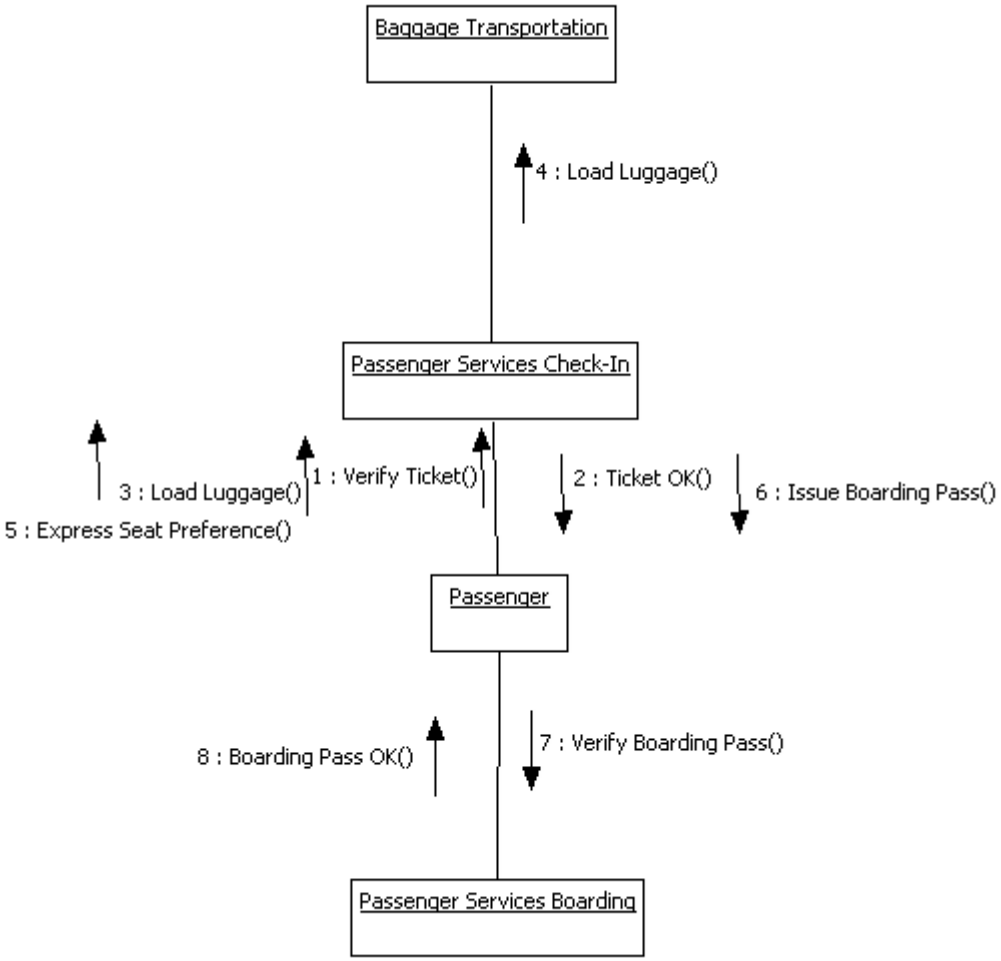
There are two types of sequence diagrams: UML diagrams and code-based diagrams. The latter is sourced from programming code



4. Collaboration Diagram

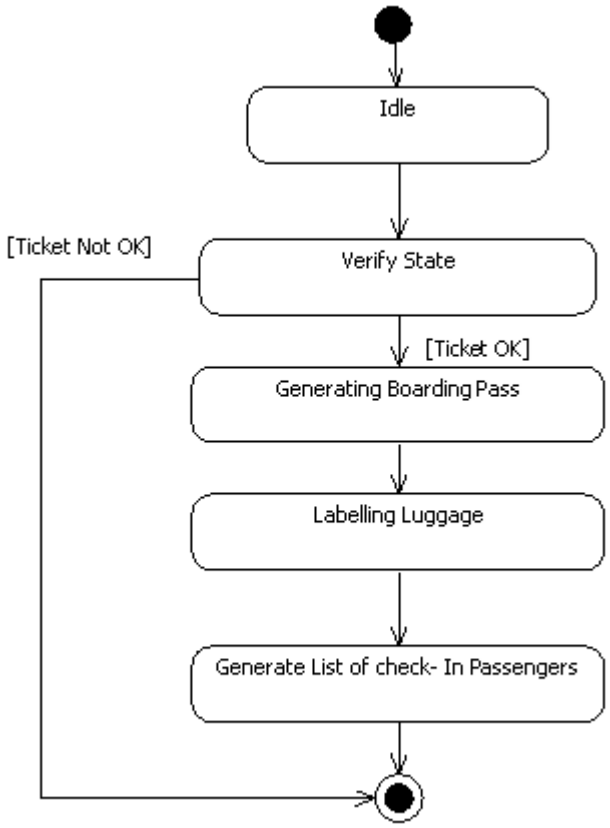
A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modelling Language (UML). These diagrams can be used to portray the dynamic behaviour of a particular use case and define the role of each object.

Collaboration diagrams are created by first identifying the structural elements required to carry out the functionality of an interaction. A model is then built using the relationships between those elements. Several vendors offer software for creating and editing collaboration diagrams.



5. State Chart Diagram

The initial state in this system is idle, which is then transited into verify state. The verify state is followed by a Guard condition which implies the ticket has been verified. If the guard condition is true then only the next transition takes place otherwise it will follow the 'Ticket not OK' guard condition and transition will end by reaching directly to its final state. Here verifies state is the operational state when it is not followed the idle(object) will be directed to the fault state. Generating Boarding Pass, Labelling Luggage are two more states after the verification state is completed. Generate list of check in passengers is the last state after which it reaches to its final state and the transition is complete.

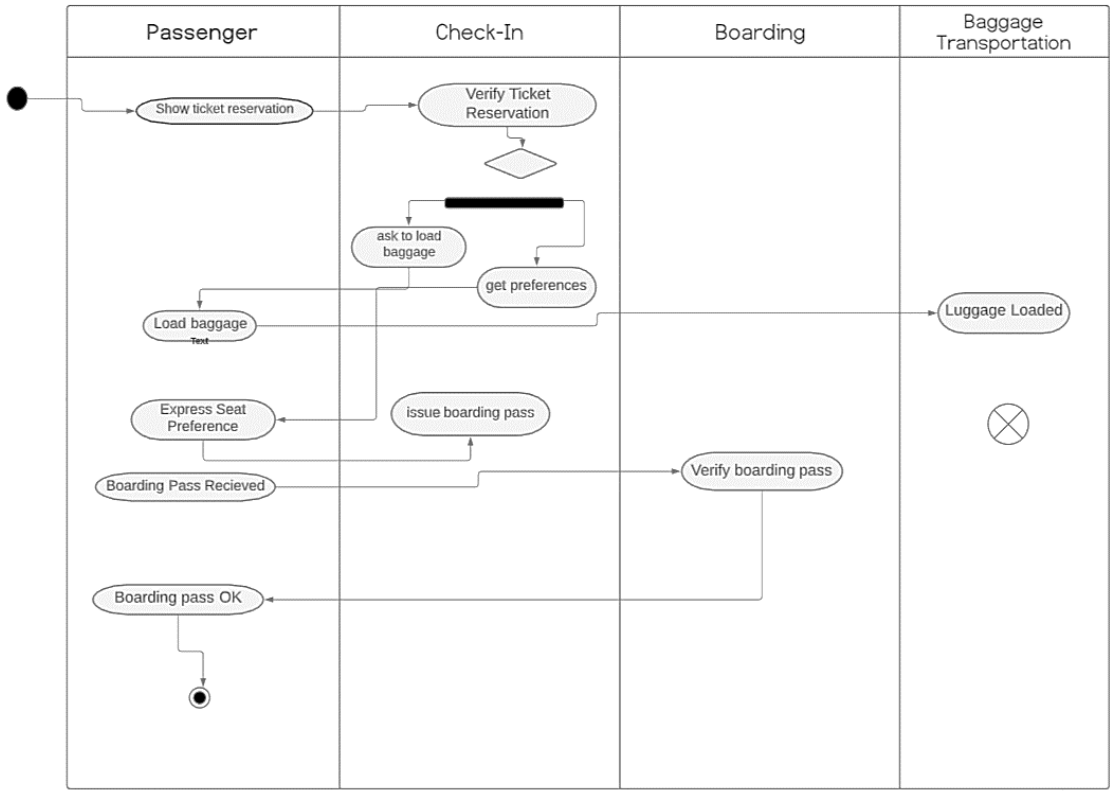


6. Activity Diagram

An activity diagram is a behavioural diagram i.e. it depicts the behaviour of a system.

An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed. We can depict both sequential processing and concurrent processing of activities using an activity diagram. They are used in business and process modelling where their primary use is to depict the dynamic aspects of a system.

An activity diagram is very similar to a flowchart.



7. Component Diagram

The component diagram represents the components of the system and how one software works with the other. The component either receives or gives information. Here the component contains interfaces which are checking luggage application, check in application, Issuing Boarding pass and Ticket verification application. All these components are dependent on the database for the information and have to wait for the information from database to act on.