



Start

- 1-Instantiate, start,join(Hostess,Pilot,Passenger)
- 2-Instantiate locations
- 3-global repository

Departure

4-[WaitInQueue, ShowDocuments,] -> Passenger
5-[InformPlaneReadyForBoarding, WaitForAllInBoard, parkAtTransferGate] -> Pilot
6-[PrepareForPassBoarding, CheckDocuments, waitforNextPassenger, waitforNextFlight, InformPlaneReadyToTakeOff]->Hostess

Plane

```

7-[BoardThePlane, waitForEndOfFlight, LeaveThePlane] -> Passenger
8-[FlyToTheDestinationPoint, AnnounceArrival,FlyToDeparturePoint] -> Pilot
9-[waitBoarding]->Hostess

```

Destination

10-[Death] -> Passenger
11-[] -> Pilot
12-[...]->Hostess

General Repository

```

13 - synchronized setST_Hostess(Hostess.State) -> set state of hostess, during events. log_write(String type)
-> write state on file.
14 - synchronized setST_Pilot(Pilot.State)-> set state of pilot. during events. setFN(int id) -> set flight number.
board_start(String type) -> writing change of events. log_write(String type) -> write state on file
15 - synchronized setST_Passenger(int id, Passenger.State) -> set state of each passenger. log_write(String
type) -> write state of each passenger on file.
16 - synchronized setQ(ArrayList<Passenger>) -> set state of passenger in queue. departed(int n_capacity)
create a list of flights departed with a number of passengers. pass_check(String text) write string on the file of
passenger checked.
17 - synchronized setATL(ArrayList<Passenger>) -> set state number of passengers that arrived at destination.
18 - synchronized setIN_F(ArrayList<Passenger>) -> set state number of passengers that are in flight

```