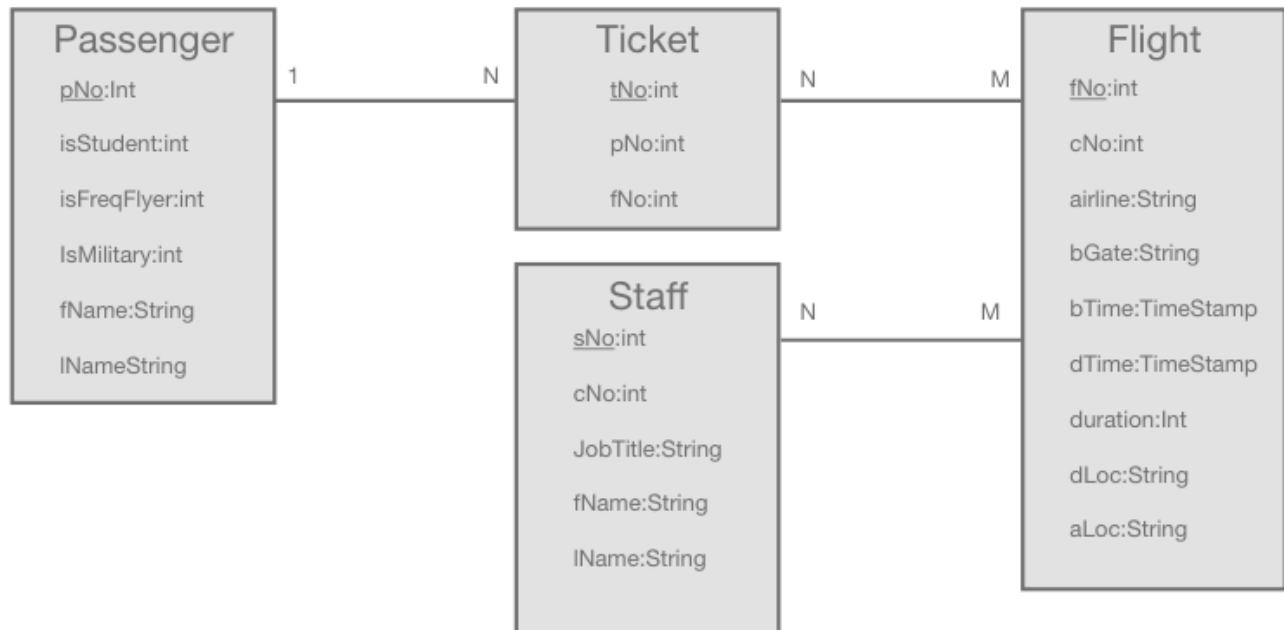# Documentation

Logical ER-Diagram

Conceptual Design

Schemas

Relation										
F	*fNo*	cNo	airline	gate	bTime	dTime	duration	dLoc	aLoc
P	*pNo*	isStudent	isFreqFlyer	isMilitary	fName	lName			
S	*sNo*	cNo	JobTitle	fName	lName				
T	*tNo*	pNo	fNo						

Notes

- Tno will have a trigger to instantly provide a unique ID to each ticket for normalization purposes
- Benefits will be calculated through three boolean fields stored on the passenger relation. These boolean fields can be updated by the user and will together determine the benefits the passenger receives.
- The Ticket relation allows us to have a relation that can hold all of flight history that passengers have. Passengers who have taken more than one flight will have more than

one ticket.
- The staff relation will have a unique value to identify the member, then a crew number that will be tied to a flight. Each crew will conceptually have 10 members, Pilot, Copilot, 4 flight-attendants, and 4 grounds crew.
- The Flight Relation will hold information regarding a flight, without any repetition, this is due to the fact that every staff that is part of the crew will have the same crew number. In addition, the ticket relation will reduce the amount of repetition needed in the flight relation.

Normalization Analysis

Flight(_fNo_, cNo, airline, bGate, bTime, dTime, duration, dLoc, aLoc)
$fNo \rightarrow cNo$
$fNo \rightarrow airline$
$fNo \rightarrow bGate$
$fNo \rightarrow bTime$
$fNo \rightarrow dTime$
$fNo \rightarrow duration$
$fNo \rightarrow dLoc$
$fNo \rightarrow aLoc$

Passenger(_pNo_, isStudent, isFreqFlyer, isMilitary, fName, lName)
$pNo \rightarrow isStudent$
$pNo \rightarrow isFreqFlyer$
$pNo \rightarrow isMilitary$
$pNo \rightarrow fName$
$pNo \rightarrow lName$

Staff(_sNo_, cNo, jobTitle, fName, lName)
$sNo \rightarrow cNo$
$sNo \rightarrow jobTitle$
$sNo \rightarrow fName$
$sNo \rightarrow lName$

Ticket(_tNo_, pNo, fNo)
$tNo \rightarrow fNo$
$tNo \rightarrow pNo$

$1NF$:
- All records are unique
- All record columns contain only a single value
$2NF$:

- DB is in 1NF

- All tables have a single column primary key

$3NF$:

- DB is in 2NF

- All tables are free from transitive functional depedencies