

Progetto Object Orientation

CdL in Informatica

A.A. 2020/2021

Caccavale Mariano – N86003303

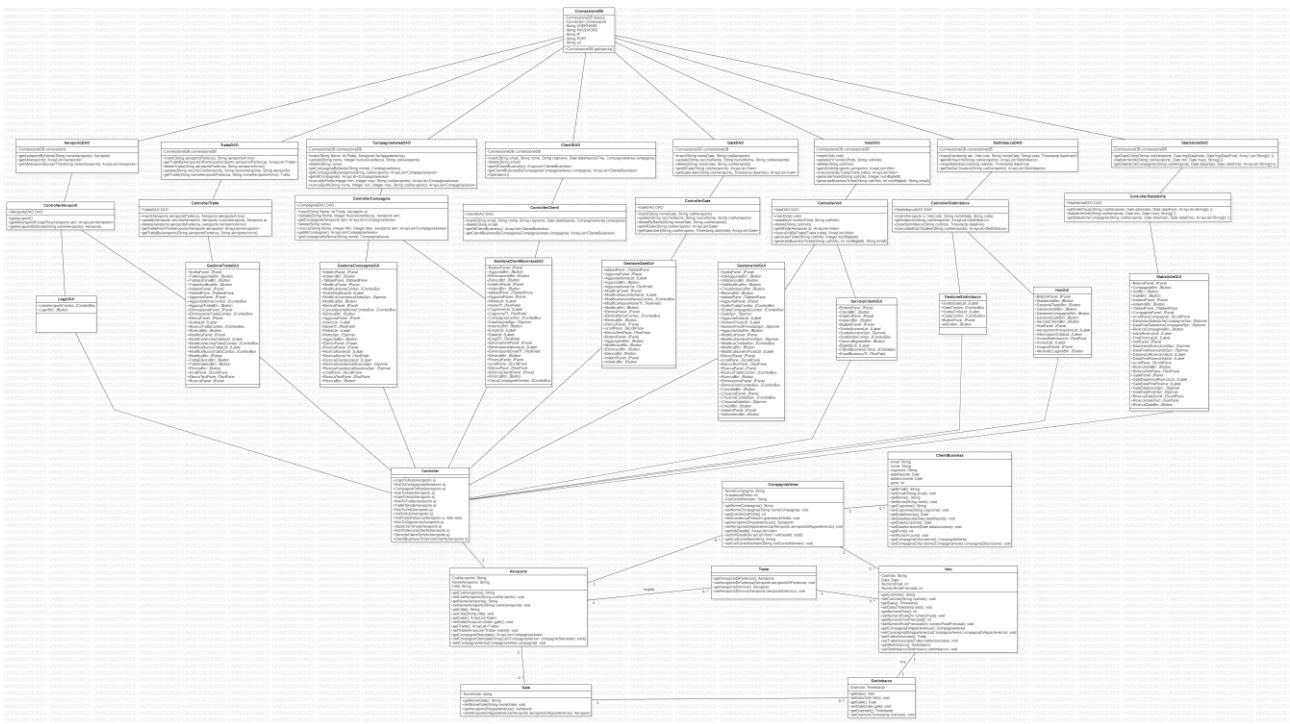
Ciccarelli Mariaelena - N86003212

ID Gruppo: 1

Analisi del problema

Il problema in questione è quello di creare un sistema di gestione di uno scalo aeroportuale che si interfacci con un database che immagazzini le informazioni più importanti riguardo le entità prese in considerazione. Come primo passo infatti, ci siamo chiesti quali sarebbero state le entità chiave del problema, e dopo averne fatto una lista più o meno esaustiva, abbiamo iniziato a pensare come queste entità fossero costituite, consci del fatto che sicuramente sarebbe servito raffinarle anche in fasi più avanzate della progettazione. Sin da subito abbiamo cercato di formalizzare il più possibile, in modo da non lasciare entità, relazioni, e/o attributi non descritti che ci sarebbero potuti servire man mano che il progetto prendeva forma; questo infatti ci avrebbe costretto a riconsiderare la progettazione, cosa che avrebbe sicuramente potuto causare problemi a catena, costringendoci a spendere altro tempo sulla progettazione. Da questa prima fase di progettazione e confronto è risultato il class diagram seguente, che ci ha accompagnato poi per tutta la programmazione fisica della basi di dati e del codice Java relativo all'applicativo.

Class Diagram

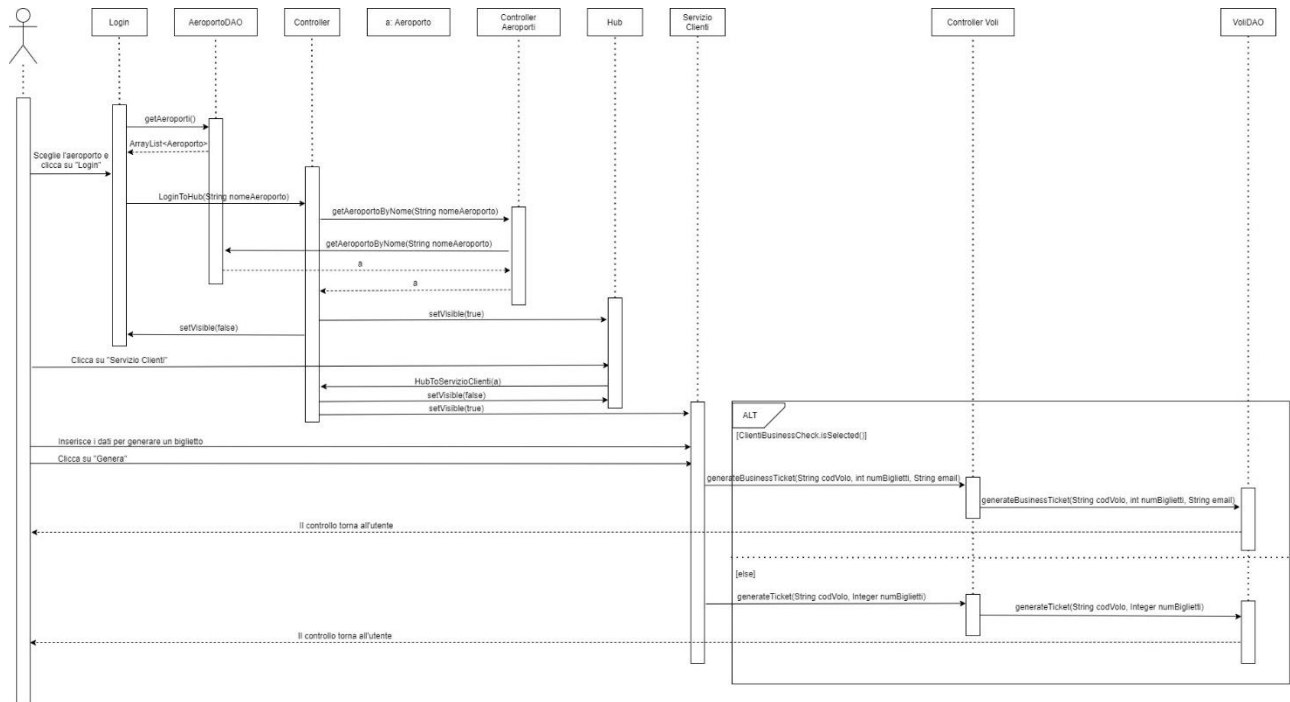


Il problema richiedeva la gestione di scali aeroportuali, quindi sicuramente l'entità iniziale doveva essere "Aeroporto". Era inoltre richiesto di poter tenere traccia di compagnia aeree, tratte e gate. Di conseguenza, la loro formalizzazione in classi quali "CompagniaAerea", "Tratta" e "Gate". Tutti gli attributi appartenenti a queste classi sono stati progettati tenendo soprattutto conto delle richieste di ricerca espresse nella traccia.

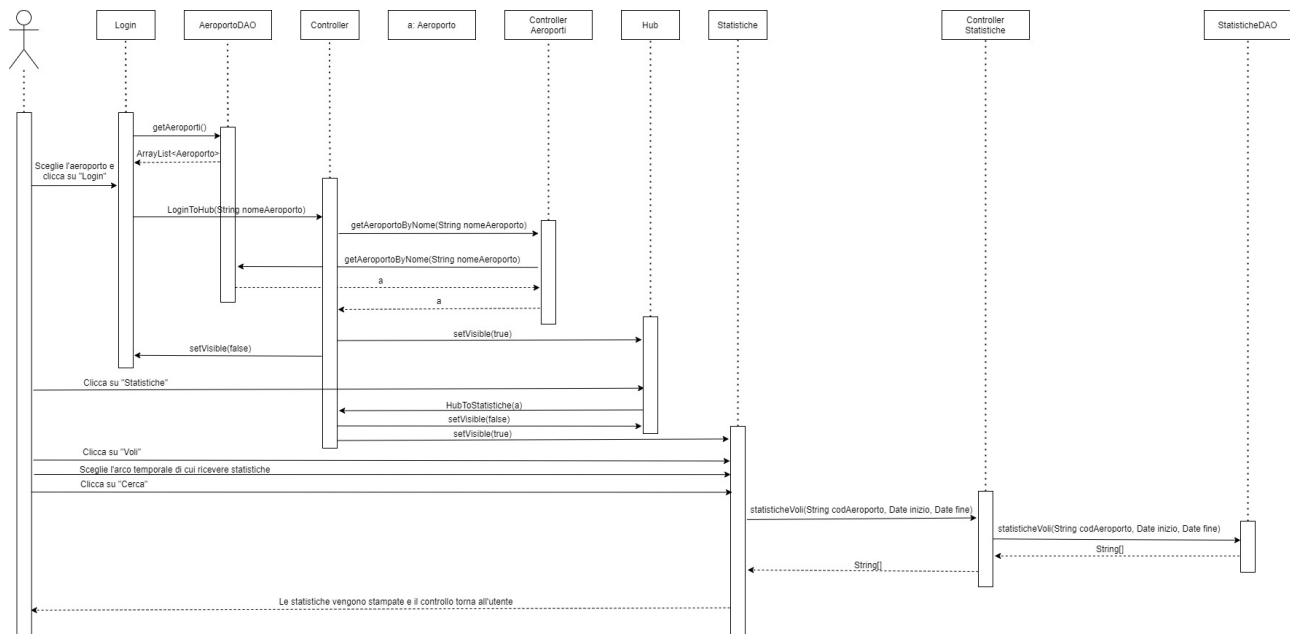
Per poter realizzare meglio alcune ricerche e agevolare la comprensione dell'intero elaborato abbiamo anche deciso di estrarre dall'entità "Tratta" quella che poi è diventata l'entità "Volo". In questo modo è più facile gestire gli stessi e anche metterli in relazione con altre entità. Inoltre a posteriori questa scelta si è rivelata vantaggiosa per l'implementazione del codice Java.

Sequence Diagram

Sequence diagram relativo alla generazione di un biglietto aereo



Sequence diagram relativo alle statistiche sui voli



CRC Card

Class Name: Login	
Superclass: JFrame	
Subclass:	
Responsability: componente grafica per il login iniziale	Collaborators: Controller, ControllerAeroporti

Class Name: Hub	
Superclass: JFrame	
Subclass:	
Responsability: componente grafica per la gestione dell'Hub	Collaborators: Controller, ControllerSlotImbarco

Class Name: GestioneTratte	
Superclass: JFrame	
Subclass:	
Responsability: componente grafica per la gestione delle tratte	Collaborators: Controller, ControllerTratte, ControllerAeroporti

Class Name: GestioneCompagnie	
Superclass: JFrame	
Subclass:	
Responsability: componente grafica per la gestione delle compagnie aeree	Collaborators: Controller, ControllerCompagnie

Class Name: GestioneGate	
Superclass: JFrame	
Subclass:	
Responsability: componente grafica per la gestione dei gate	Collaborators: Controller, ControllerGate

Class Name: GestioneSlotImbarco	
Superclass: JFrame	
Subclass:	
Responsability: componente grafica per la gestione degli SlotImbarco	Collaborators: Controller, ControllerSlotImbarco

Class Name: GestioneVoli	
Superclass: JFrame	
Subclass:	
Responsability: componente grafica per la gestione dei voli	Collaborators: Controller, ControllerVoli

Class Name: ServizioClienti	
Superclass: JFrame	
Subclass:	
Responsability: componente grafica per la gestione del servizio clienti	Collaborators: Controller, ControllerVoli

Class Name: Statistiche	
Superclass: JFrame	
Subclass:	
Responsability: componente grafica per la gestione delle statistiche	Collaborators: Controller, ControllerStatistiche

Class Name: GestioneClientiBusiness	
Superclass: JFrame	
Subclass:	
Responsability: componente grafica per la gestione dei clienti business	Collaborators: Controller, ControllerCompagnie, ControllerClienti

Class Name: AeroportoDAO	
Superclass:	
Subclass:	
Responsability: gestione delle query sulla tabella "aeroporto" del database	Collaborators: Connessione, ControllerAeroporti

Class Name: ClientiDAO	
Superclass:	
Subclass:	

Responsability: gestione delle query sulla tabella “clientibusiness” del database	Collaborators: Connessione, ControllerClienti
------------------------------------------------------------------------------------------	------------------------------------------------------

Class Name: CompagniaAereaDAO	
Superclass:	
Subclass:	
Responsability: gestione delle query sulla tabella “compagniaaerea” del database	Collaborators: Connessione, ControllerAeroporti

Class Name: GateDAO	
Superclass:	
Subclass:	
Responsability: gestione delle query sulla tabella “gate” del database	Collaborators: Connessione, ControllerGate

Class Name: SlotImbarcoDAO	
Superclass:	
Subclass:	
Responsability: gestione delle query sulla tabella “slotimbarco” del database	Collaborators: Connessione, ControllerSlotImbarco

Class Name: StatisticheDAO	
Superclass:	
Subclass:	
Responsability: gestione delle query sulla tabella “archiviovoli” del database	Collaborators: Connessione, ControllerStatistiche

Class Name: TrattaDao	
Superclass:	
Subclass:	
Responsability: gestione delle query sulla tabella “tratta” del database	Collaborators: Connessione, ControllerTratte

Class Name: VoloDAO	
----------------------------	--

Superclass:	
Subclass:	
Responsability: gestione delle query sulla tabella “volo” del database	Collaborators: Connessione, ControllerVoli

Class Name: Controller	
Superclass:	
Subclass:	
Responsability: Gestisce lo spostamento tra le diverse GUI.	Collaborators: Aeroporto

Class Name: ControllerAeroporto	
Superclass:	
Subclass:	
Responsability: Gestire la connessione tra l'interfaccia grafica e la DAO degli aeroporti	Collaborators: AeroportoDAO

Class Name: ControllerClienti	
Superclass:	
Subclass:	
Responsability: Gestisce la connessione tra l'interfaccia grafica e la DAO dei clienti	Collaborators: ClientiDAO

Class Name: ControllerCompagnie	
Superclass:	
Subclass:	
Responsability: Gestisce la connessione tra l'interfaccia grafica e la DAO delle compagnie aeree	Collaborators: CompagniaAereaDAO

Class Name: ControllerGate	
Superclass:	
Subclass:	
Responsability: Gestisce la connessione tra l'interfaccia grafica e la DAO dei gate	Collaborators: GateDAO

Class Name: ControllerSlotImbarco	
Superclass:	
Subclass:	
Responsability: Gestisce la connessione tra l'interfaccia grafica e la DAO degli slot imbarco	Collaborators: SlotImbarcoDAO

Class Name: ControllerStatistiche	
Superclass:	
Subclass:	
Responsability: Gestisce la connessione tra l'interfaccia e la DAO delle statistiche	Collaborators: StatisticheDAO

Class Name: ControllerTratte	
Superclass:	
Subclass:	
Responsability: Gestisce la connessione tra l'interfaccia grafica e la DAO delle tratte	Collaborators: TratteDAO

Class Name: ControllerVoli	
Superclass:	
Subclass:	
Responsability: Gestisce la connessione tra l'interfaccia grafica e	Collaborators: VoliDAO

la DAO dei voli	
-----------------	--

Class Name: ClientiException	
Superclass: Exception	
Subclass:	
Responsability: Gestire propriamente le eccezioni riguardanti i clienti	Collaborators:

Class Name: CompagniaException	
Superclass: Exception	
Subclass:	
Responsability: Gestire propriamente le eccezioni riguardanti le compagnie aeree	Collaborators:

Class Name: GateException	
Superclass: Exception	
Subclass:	
Responsability: Gestire propriamente le eccezioni riguardanti i gate	Collaborators:

Class Name: SlotImbarcoException	
Superclass: Exception	
Subclass:	
Responsability: Gestire propriamente le eccezioni riguardanti gli SlotImbarchi	Collaborators:

Class Name: StatisticheException	
Superclass: Excpetion	
Subclass:	
Responsability: Gestire	Collaborators:

propriamente le eccezioni riguardanti le statistiche	
------------------------------------------------------	--

Class Name: TrattaException	
Superclass: Exception	
Subclass:	
Responsability: Gestire propriamente le eccezioni riguardanti le tratte	Collaborators:

Class Name: VoloException	
Superclass: Excpetion	
Subclass:	
Responsability: Gestire propriamente le eccezioni riguardanti i voli	Collaborators:

Class Name: Aeroporto	
Superclass:	
Subclass:	
Responsability: Conservare le informazioni riguardanti un aeroporto	Collaborators:

Class Name: ClientiBusiness	
Superclass:	
Subclass:	
Responsability: Conservare le informazioni riguardanti un cliente business	Collaborators: CompagniaAerea

Class Name: CompagniaAerea	
Superclass:	
Subclass:	

Responsability: Conservare le informazioni riguardanti una compagnia aerea	Collaborators: Aeroporto
-----------------------------------------------------------------------------------	---------------------------------

Class Name: Gate	
Superclass:	
Subclass:	
Responsability: Conservare le informazioni riguardanti un gate	Collaborators: Aeroporto

Class Name: SlotImbarco	
Superclass:	
Subclass:	
Responsability: Conservare le informazioni riguardanti uno slot imbarco	Collaborators: Volo, Tratta, Gate

Class Name: Tratta	
Superclass:	
Subclass:	
Responsability: Conservare le informazioni riguardanti una tratta	Collaborators: Aeroporto

Class Name: Volo	
Superclass:	
Subclass:	
Responsability: Conservare le informazioni riguardanti un volo	Collaborators: CompagniaAerea, Tratta

Class Name: ConnessioneDB	
Superclass:	
Subclass:	
Responsability: Gestire la connessione e renderla un singleton.	Collaborators: Connection