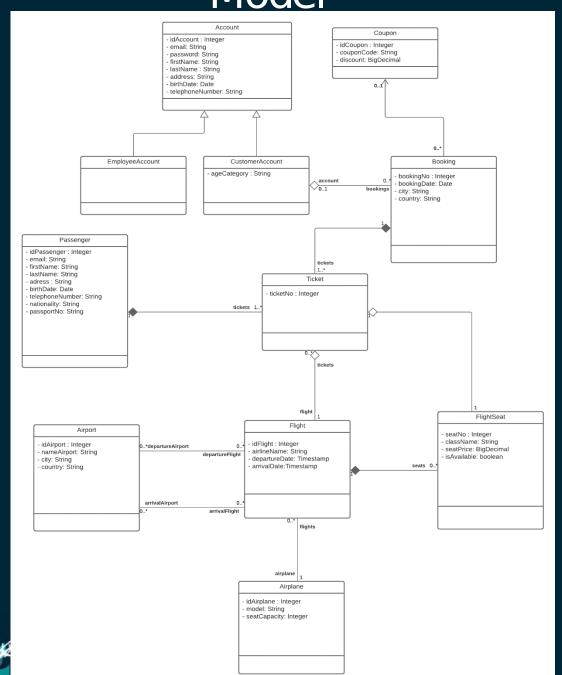
Model



View

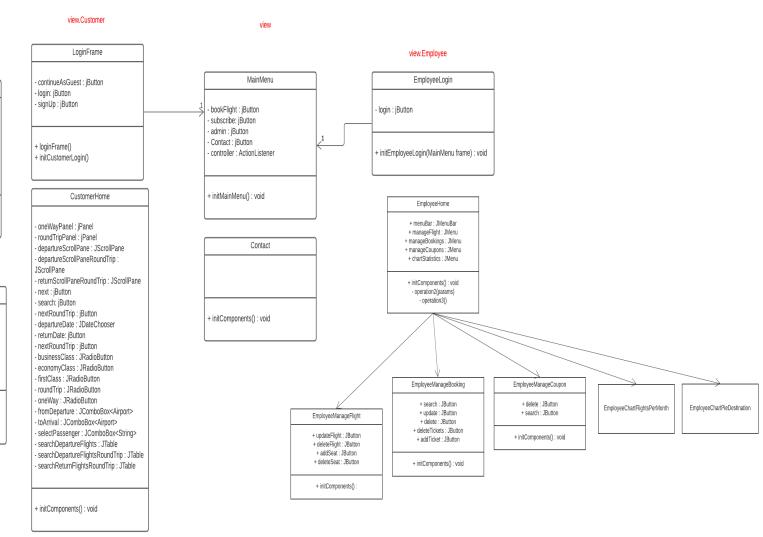
PassengerInfosFrame

- birthDates : jDateChooser
- DepartAvailableSeatChoice: ArrayList<JComboBox<FlightSeat>>
- ReturnAvailableSeatChoice : ArrayList<JComboBox<FlightSeat>>
- passengersScrollPane : JScrollPane
- passengersPanel : JPanel
- payment : JPanel
+ initComponents() : void

AccountFrame - booking : jButton - personalInfo: jButton - myAccountPanel : JPanel - myAccountInfos : JPanel - MyBookings : JPanel

+ publicMyAccountFrame(CustomerAccount loggedInCustomer): void

+ initComponents(): void



BookingController

- + addBooking(Date date,int idAccount): void
- + updateBooking(int bookingNo, Date bookingDate, int idAccount, int idCoupon, int newBookingNo): void
- + deleteBooking (int idBooking): void
- + checkBooking(String bookingNo): boolean
- + findAllBookings : ArrayList<Booking>

TicketController

- + addTicketToBooking(int bookingNo, Passenger passenger, int flightSeatNo, int idFlight): void
- + getTicketPriceNotLogged(int idFlight, int seatNo) : BigDecimal
- + getTicketPriceLogged(int idFlight, int seatNo, CustomerAccount customer): BigDecimal
- + deleteTicket (int ticketNo): void

CouponController

- + addCoupon(String code, float discount) : void
- +findAllCoupons(): ArrayList<Coupon>
- + deleteCoupon(String Code) : void
- + checkCoupon(String) : boolean

FlightSeatsController

- + addFlightSeat(FlightSeat seat, int idFlight) : void
- + deleteFlightSeat(FlightSeat seat): void

FlightController

- + addFlight(int idAirplane, String airlineName, String idDepartureAirport, String idArrivalAirport, Timestamp departureDate, Timestamp arrivalDate, int nbSeatsFirstClass, int nbSeatsBusinessClass, int nbSeatsEconomyClass, BigDecimal priceFirst, BigDecimal priceBusiness, BigDecimal priceEconomy): void
- + getAvailableSeats(int idFlight, String className) : FlightSeat[]
- + updateFlight(int idFlight, int idAirplane, String airlineName, String idDepartureAirport, String idArrivalAirport, Timestamp departureDate, Timestamp arrivalDate): void
- + CheckFlightInfo(Date dateDeparture ,Date dateArrival, String airlineField) : boolean
- + deleteFlight (int idFlight) : void

AirplaneController

+ getAllAirplanes() : Airplane[]

AirportController

- + getAllAirports() : Airport[]
- + findPercentageAllAirports : ArrayList<Double>
- + findAllAirportNames(): ArrayList<String>

CustomerFlightSearchController

- + oneWaySearchController(): void
- + roundTripSearchController(): void

BookingController

- + addBooking(Date date.int idAccount) : void
- updateBooking(int bookingNo, Date bookingDate, int idAccount, int idCoupon, int newBookingNo) : void
- deleteBooking (int idBooking) : void
- checkBooking(String bookingNo) : boolear + findAllBookings : ArrayList<Booking>

TicketController

Controller

- addTicketToBooking(int bookingNo, Passenger passenger, int flightSeatNo, int idFlight) : void

- getTicketPriceNotLogged(int idFlight, int seatNo) : BigDecimal
- getTicketPriceLogged(int idFlight, int seatNo, CustomerAccount customer): BigDecimal
- deleteTicket (int ticketNo): void

+findAllCoupons() : ArravList<Coupon>

+ deleteCoupon(String Code): void

+ checkCoupon(String) : boolean

CouponController

+ addCoupon(String code, float discount) : void

- addFlightSeat(FlightSeat seat, int idFlight) : void
- deleteFlightSeat(FlightSeat seat) : void

FlightSeatsController

- getAvailableSeats(int idFlight, String className) : FlightSeat[]

- + addFlightfint idAirplane, String airlineName, String idDepartureAirport, String idArrivalAirport, Timestamp departureDate, Timestamp arrivalDate, int nbSeatsFirstClass, int nbSeatsBusinessClass, int nbSeatsEconomyClass, BigDecimal priceFirst, BigDecimal priceBusiness, BigDecimal priceEconomy) ; void
- getAvailableSeats(int idFlight, String className) : FlightSeat[]
- r updateFlight(int idFlight, int idAirplane, String airlineName, String idDepartureAirport, String idArrivalAirport, Timestamp departureDate, Timestamp
- + CheckFlightInfo(Date dateDeparture ,Date dateArrival, String airlineField) : boolean
- + deleteFlight (int idFlight) : void

AirplaneController

getAllAirplanes() : Airplane[]

AirportController

- getAllAirports(): Airport[] findPercentageAllAirports : ArrayList<Double>
- findAllAirportNames(): ArrayList<String>

CustomerFlightSearchController

- oneWaySearchController(): void
- + roundTripSearchController() : void

CheckFlightSeatInfo

- CheckFlightSeatInfo(String seatNo,String price) : boolean

FieldsController

- + checkSignUp(String email, String password, String lastName, String firstName, String telephoneNumber) : boolean
- + checkPassengerInfo(String email, String password, String lastName, String firstName, String telephoneNumber, String nationality, String passportNo): boolean
- + checkSeatNumbers(ArrayList<JComboBox<FlightSeat>> DepartAvailableSeatChoice, ArrayList<JComboBox<FlightSeat>> ReturnAvailableSeatChoice, int numberofpassengers) : boolean getMD5Hash(String s) : String

EmployeeLoginController

- + EmployeeLoginConnection(String mail, String password) : boolean
- getEmployeeAccount(String mail, String password) : EmployeeAccount

PassengerController

getLastNamePassengerByTicketNo(int TicketNo) : String getFirstNamePassengerByTicketNo(int TicketNo) : String

CustomerController

- + CustomerLoginConnection(String mail, String password) : boolear
- +SignUpCustomerAccount(String email, String password, String lastName, String firstName, String telephoneNumber, Date birthDate, String address): boolean
- + update(CustomerAccount account) : void
- + checkBooking(String bookingNo) : boolean + findAllBookings · Arrayl ist<Booking>

DAO

