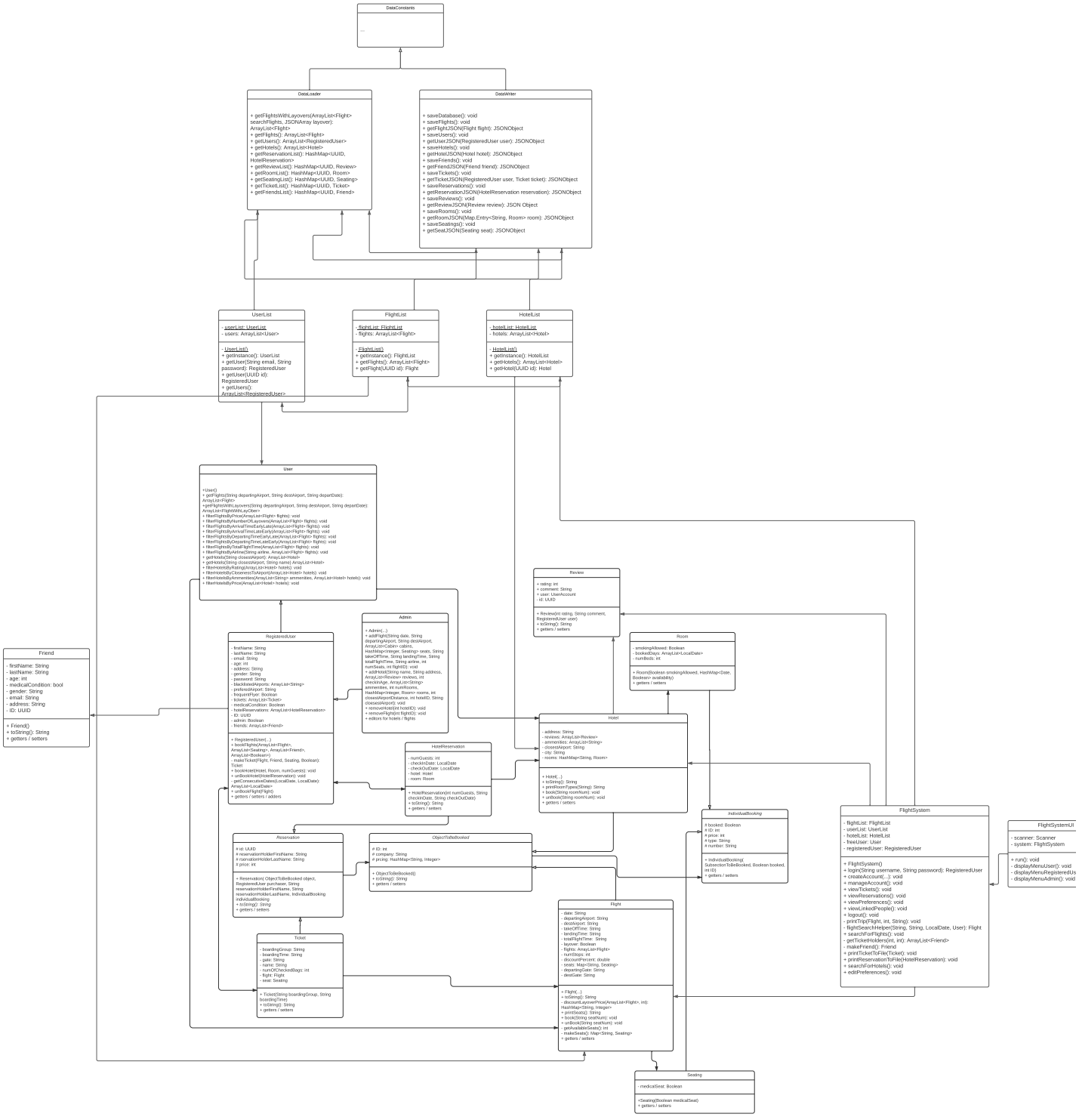
DataConstants
...
DataLoader
+ getFlightsWithLayovers(ArrayList<Flight> searchFlights, JSONArray layover): ArrayList<Flight>
+ getFlights(): ArrayList<Flight>
+ getUsers(): ArrayList<RegisteredUser>
+ getHotels(): ArrayList<Hotel>
+ getReservationList(): HashMap<UUID, HotelReservation>
+ getReviewList(): HashMap<UUID, Review>
+ getRoomList(): HashMap<UUID, Room>
+ getSeatingList(): HashMap<UUID, Seating>
+ getTicketList(): HashMap<UUID, Ticket>
+ getFriendsList(): HashMap<UUID, Friend>
DataWriter
+ saveDatabase(): void
+ saveFlights(): void
+ getFlightJSON(Flight flight): JSONObject
+ saveUsers(): void
+ getUserJSON(RegisteredUser user): JSONObject
+ saveHotels(): void
+ getHotelJSON(Hotel hotel): JSONObject
+ saveFriends(): void
+ getFriendJSON(Friend friend): JSONObject
+ saveTickets(): void
+ getTicketJSON(RegisteredUser user, Ticket ticket): JSONObject
+ saveReservations(): void
+ getReservationJSON(HotelReservation reservation): JSONObject
+ saveReviews(): void
+ getReviewJSON(Review review): JSON Object
+ saveRooms(): void
+ getRoomJSON(Map.Entry<String, Room> room): JSONObject
+ saveSeatings(): void
+ getSeatJSON(Seating seat): JSONObject
UserList
- userList: UserList
- users: ArrayList<User>
- UserList()
+ getInstance(): UserList
+ getUser(String email, String password): RegisteredUser
+ getUser(UUID id): RegisteredUser
+ getUsers(): ArrayList<RegisteredUser>
FlightList
- flightList: Flight.ist
- flights: ArrayList<Flight>
- FlightList()
+ getInstance(): FlightList
+ getFlights(): ArrayList<Flight>
+ getFlight(UUID id): Flight
HotelList
- hotelList: HotelList
- hotels: ArrayList<Hotel>
- HotelList()
+ getInstance(): HotelList
+ getHotels(): ArrayList<Hotel>
+ getHotel(UUID id): Hotel
User
+User()
+ getFlights(String departingAirport, String destAirport, String departDate): ArrayList<Flight>
+getFlightsWithLayovers(String departingAirport, String destAirport, String departDate): ArrayList<FlightWithLayOver>
+ filterFlightsByPrice(ArrayList<Flight> flights): void
+ filterFlightsByNumberOfLayovers(ArrayList<Flight> flights): void
+ filterFlightsByArrivalTimeEarlyLate(ArrayList<Flight> flights): void
+ filterFlightsByArrivalTimeLateEarly(ArrayList<Flight> flights): void
+ filterFlightsByDepartingTimeEarlyLate(ArrayList<Flight> flights): void
+ filterFlightsByDepartingTimeLateEarly(ArrayList<Flight> flights): void
+ filterFlightsByTotalFlightTime(ArrayList<Flight> flights): void
+ filterFlightsByAirline(String airline, ArrayList<Flight> flights): void
+ getHotelsz(String closestAirport): ArrayList<Hotel>
+ getHotels(String closestAirport, String name) ArrayList<Hotel>
+ filterHotelsByRating(ArrayList<Hotel> hotels): void
+ filterHotelsByClosenessToAirport(ArrayList<Hotel> hotels): void
+ filterHotelsByAmmenities(ArrayList<String> ammenities, ArrayList<Hotel> hotels): void
+ filterHotelsByPrice(ArrayList<Hotel> hotels): void
Friend
- firstName: String
- lastName: String
- age: int
- medicalCondition: bool
- gender: String
- email: String
- address: String
- ID: UUID
+ Friend()
+ toString(): String
+ getters / setters
RegisteredUser
- firstName: String
- lastName: String
- email: String
- age: int
- address: String
- gender: String
- password: String
- blacklistedAirports: ArrayList<String>
- preferedAirport: String
- frequentFlyer: Boolean
- tickets: ArrayList<Ticket>
- medicalCondition: Boolean
- hotelReservations: ArrayList<HotelReservation>
- ID: UUID
- admin: Boolean
- friends: ArrayList<Friend>
+ RegisteredUser(...)
+ bookFlight(ArrayList<Flight>, ArrayList<Seating>, ArrayList<Friend>, ArrayList<Boolean>)
- makeTicket(Flight, Friend, Seating, Boolean): Ticket
+ bookHotel(Hotel, Room, numGuests): void
+ unBookHotel(HotelReservation): void
- getConsecutiveDates(LocalDate, LocalDate): ArrayList<LocalDate>
+ unBookFlight(Flight)
+ getters / setters / adders
Admin
+ Admin(...)
+ addFlight(String date, String departingAirport, String destAirport, ArrayList<Cabin> cabins, HashMap<Integer, Seating> seats, String takeOffTime, String landingTime, String totalFlightTime, String airline, int numSeats, int flightID): void
+ addHotel(String name, String address, ArrayList<Review> reviews, int checkInAge, ArrayList<String> ammenities, int numRooms, HashMap<Integer, Room> rooms, int closestAirportDistance, int hotelID, String closestAirport): void
+ removeHotel(int hotelID): void
+ removeFlight(int flightID): void
+ editors for hotels / flights
Review
+ rating: int
+ comment: String
+ user: UserAccount
- id: UUID
+ Review(int rating, String comment, RegisteredUser user)
+ toString(): String
+ getters / setters
Room
- smokingAllowed: Boolean
- bookedDays: ArrayList<LocalDate>
- numBeds: int
+ Room(Boolean smokingAllowed, HashMap<Date, Boolean> availability)
+ getters / setters
Hotel
- address: String
- reviews: ArrayList<Review>
- ammenities: ArrayList<String>
- closestAirport: String
- city: String
- rooms: HashMap<String, Room>
+ Hotel(...)
+ toString(): String
+ printRoomTypes(String): String
+ book(String roomNum): void
+ unBook(String roomNum): void
+ getters / setters
HotelReservation
- numGuests: int
- checkInDate: LocalDate
- checkOutDate: LocalDate
- hotel: Hotel
- room: Room
+ HotelReservation(int numGuests, String checkInDate, String checkOutDate)
+ toString(): String
+ getters / setters
Reservation
id: UUID
reservationHolderFirstName: String
reservationHolderLastName: String
price: int
+ Reservation(ObjectToBeBooked object, RegisteredUser purchaser, String reservationHolderFirstName, String reservationHolderLastName, IndividualBooking individualBooking
+ toString(): String
+ getters / setters
ObjectToBeBooked
ID: int
company: String
pricing: HashMap<String, Integer>
+ ObjectToBeBooked()
+ toString(): String
+ getters / setters
IndividualBooking
booked: Boolean
ID: int
price: int
type: String
number: String
+ IndividualBooking(SubsectionToBeBooked, Boolean booked, int ID)
+ getters / setters
FlightSystem
- flightList: FlightList
- userList: UserList
- hotelList: HotelList
- freeUser: User
- registeredUser: RegisteredUser
+ FlightSystem()
+ login(String username, String password): RegisteredUser
+ createAccount(...): void
+ manageAccount(): void
+ viewTickets(): void
+ viewReservations(): void
+ viewPreferences(): void
+ viewLinkedPeople(): void
+ logout(): void
- printTrip(Flight, int, String): void
- flightSearchHelper(String, String, LocalDate, User): Flight
+ searchForFlights(): void
- getTicketHolders(int, int): ArrayList<Friend>
- makeFriend(): Friend
+ printTicketToFile(Ticket): void
+ printReservationToFile(HotelReservation): void
+ searchForHotels(): void
+ editPreferences(): void
FlightSystemUI
- scanner: Scanner
- system: FlightSystem
+ run(): void
- displayMenuUser(): void
- displayMenuRegisteredUs
- displayMenuAdmin(): void
Ticket
- boardingGroup: String
- boardingTime: String
- gate: String
- name: String
- numOfCheckedBags: int
- flight: Flight
- seat: Seating
+ Ticket(String boardingGroup, String boardingTime)
+ toString(): String
+ getters / setters
Flight
- date: String
- departingAirport: String
- destAirport: String
- takeOffTime: String
- landingTime: String
- totalFlightTime: String
- layover: Boolean
- flights: ArrayList<Flight>
- numSlops: int
- discountPercent: double
- seats: Map<String, Seating>
- departingGate: String
- destGate: String
+ Flight(...)
+ toString(): String
- discountLayoverPrice(ArrayList<Flight>, int): HashMap<String, Integer>
+ printSeats(): String
+ book(String seatNum): void
+ unBook(String seatNum): void
- getAvailableSeats() int
- makeSeats() Map<String, Seating>
+ getters / setters
Seating
- medicalSeat: Boolean
+Seating(Boolean medicalSeat)
+ getters / setters