

Travel Service Provider

Additional Features:

- Project is split into **3 packages**
 - o DTO – Data Transfer Objects
 - o Controllers – Where the actual Magic Happens
 - o Interfaces – Just the interfaces of few common methods
- Password is not stored as direct password; **it is stored as hash value of password as it cannot be reverse engineered.**
- **All the database is augmented by using HashMap**, where giving id as the key for HashMap and the actual DTO objects as the value for HashMap.
- **Getters and setters are very strictly used.**
- **Encapsulation is very strictly used.**
- **Exception handling is very strictly used.**
- ID's are used to uniquely identify customers, companies and journeys and they are generated as some computational **hash code which will be unique.**
- **Cannot book for a passed date.**
- **Cancelled Journeys are also stored** and are available to be viewed by admin and customer.
- Scanner object is created only once, and that object is imported using the static import in other classes.
- Coupons can be added by admin and by companies as well, when it is being added by admin he can make it available for all the companies.
- Booking journey follow 4 checks
 - o Available tickets
 - o Source
 - o Destination
 - o Id when booking
 - o Date
 - Past dates are not allowed
 - Invalid dates are not allowed

Assumptions:

- Admin/Customer/Company cannot change his details upon signup.
- Admin's password is set by default.
- There is a very little data that is pre fed to the system.
- The program has no coupons upon start.
- The customer wallet starts with 0.
- The money will be refunded to wallet upon cancellation.
- DTO's are used to carry objects during the traverse of data. The data basically traverses in the DTO.
- All the variables in DTO are private, so no way to access it from outside other than getters and setters.
- If today's ticket is cancelled, then no refund.

- Admin cannot remove customers or companies or journeys.
- The program never stops in a failure input.
- All the vehicles added by a company are available everyday.

Classes:

- Admin
- Bus
- Company
- Coupon
- Flight
- Journey
- Train

Interfaces:

- Planner
- UserOperations
- Vehicle

Relationships:

- Admin interacts with Coupon
- Admin interacts with Flight
- Admin interacts with Bus
- Admin interacts with Train
- Admin interacts with Company
- Company interacts with Vehicle
- Company interacts with Journey
- Company interacts with Customer
- Vehicle interacts with Customer
- Vehicle interacts with Admin
- Vehicle interacts with Journey
- Vehicle interacts with Company
- Customer interacts with Vehicle
- Customer interacts with Journey

Interactions:

- Customer can Login
- Customer can Signup
- Customer can Book a Bus
- Customer can Book a Train
- Customer can Book a Flight
- Customer can cancel a booking
- Customer can print his booking history
- Customer can print cancelled booking history
- Customer can check his profile
- Customer can Logout
- Company can Login
- Company can Logout
- Company can Signup

- Company can Add a new Bus/Flight/Train
- Company can Check all his vehicles
- Company can check his booking history
- Company can add a coupon
- Company can view all his coupons
- Company can check his profile
- Admin can login
- Admin can logout
- Admin can view all users
- Admin can view all companies
- Admin can view all buses
- Admin can view all trains
- Admin can view all flights
- Admin can view all journeys
- Admin can view all cancelled journeys
- Admin can view all coupons
- Admin can add coupon
- Admin can remove coupons