**Analysis of Time and Space Complexity:**

*Space Complexity:*

|  |  |
| --- | --- |
| Class | Space-Complexity Analysis |
| Ride | The Ride class contains five attributes (tripId, driverName, driverPhoneNumber, cabNumber, inProgress), each of which occupies a constant amount of space. Therefore, the space complexity of the Ride class is O(1). |
| User | The space complexity of the User interface and its implementations (Traveler, TravelerCompanion, Admin) is also O(1) because they mainly consist of method declarations. |
| Admin | The **Admin** class contains a Map to store rides, where the space required depends on the number of rides stored. Let's denote the number of rides as n. Therefore, the space complexity of the Admin class is O(n). |

From a complete system perspective, the space complexity of the system can be considered O(n), where n is the number of rides stored by the Admin.

*Time Complexity:*

|  |  |
| --- | --- |
| Class | Space-Complexity Analysis |
| Ride | The Ride class methods mainly involve accessing or modifying the attributes of the ride object, which takes constant time. Therefore, the time complexity of operations on the Ride class is O(1). |
| User | The time complexity of the methods in the User interface and its implementations (Traveler, TravelerCompanion, Admin) depends on the specific operations performed within each method. However, since these methods are primarily performing I/O operations and simple computations, their time complexity can be considered O(1). |
| Admin | addRide(): Adding a ride to the Map takes O(1) time on average if the map is unordered else O(logn).  accessAllRides(): Iterating through all rides in the Map takes O(nlogn) time, where n is the number of rides stored.  accessExperienceFeedback(): This method doesn't involve any loops or complex operations, so its time complexity is O(1). |

From a complete system perspective, the time complexity of the system depends on the specific operations performed but can be considered to be varying between O(n) and O(logn) in the worst case, where n is the number of rides stored by the Admin.