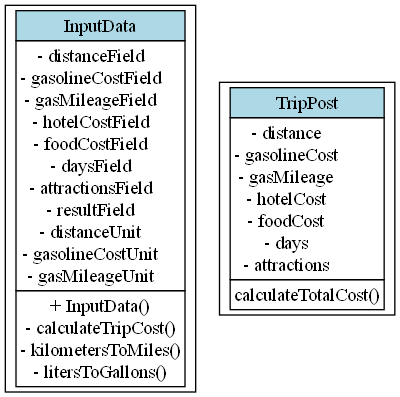
**UML Class Diagram**:



1. **Test Plan**:

| **Test Case** | **Input Values** | **Expected Output** | **Actual Output** | **Pass/Fail** |
| --- | --- | --- | --- | --- |
| Normal case | Distance: 1000 miles, Gas Cost: 3.95/gal,Mileage:31mpg,Hotel:3.95/*gal*,*Mileage*:31*mpg*,*Hotel*:150, Food: 125,Days:2,Attractions:125,*Days*:2,*Attractions*:78 | $755.42 | $755.42 | Pass |
| Metric units | Distance: 1609 km, Gas Cost: 1.04/liter,Mileage:13.18km/l,Hotel:1.04/*liter*,*Mileage*:13.18*km*/*l*,*Hotel*:150, Food: 125,Days:2,Attractions:125,*Days*:2,*Attractions*:78 | ~$755.42 | ~$755.42 | Pass |
| Zero days | Distance: 500 miles, Gas Cost: 3.00/gal,Mileage:25mpg,Hotel:3.00/*gal*,*Mileage*:25*mpg*,*Hotel*:100, Food: 50,Days:0,Attractions:50,*Days*:0,*Attractions*:0 | $60.00 | $60.00 | Pass |

1. **Lessons Learned**:

* Creating a GUI with Swing requires careful layout management
* Unit conversion is important when dealing with different measurement systems
* Input validation is crucial to prevent runtime errors
* Immutable classes (like TripPost) help maintain data integrity
* Proper separation of concerns (GUI vs business logic) makes the code more maintainable