Ticket Service: Coding Assessment

This ticket service facilitates the discovery, temporary hold, and final reservation of seats within a high-demand performance venue.

1. Assumptions

1. Ticket service is a single threaded application
2. SeatScore is dependent on the seat's proximity to the stage.
3. One can reserve a seat only if the seat is currently held.

2. Build Instructions:

From the command line terminal: >mvn test

3. Dependencies:
1. Java 8
2. JUnit
3. Maven3.2

4. Screenshot's of Test

```
-----------------------------------------------------------
 T E S T S
-----------------------------------------------------------
Running org.walmart.ticketrvice.ticket_service.RepositoryAvailableSeatsTest
Tests run: 3, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.157 sec
Running org.walmart.ticketrvice.ticket_service.RepositoryHeldSeatsTest
Tests run: 4, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 1.04 sec
Running org.walmart.ticketrvice.ticket_service.TicketServiceImplTest
Tests run: 3, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.033 sec

Results :

Tests run: 10, Failures: 0, Errors: 0, Skipped: 0
```

```
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time: 3.725 s
[INFO] Finished at: 2016-09-02T11:12:14-08:00
[INFO] Final Memory: 17M/243M
[INFO] ------------------------------------------------------------------------
```
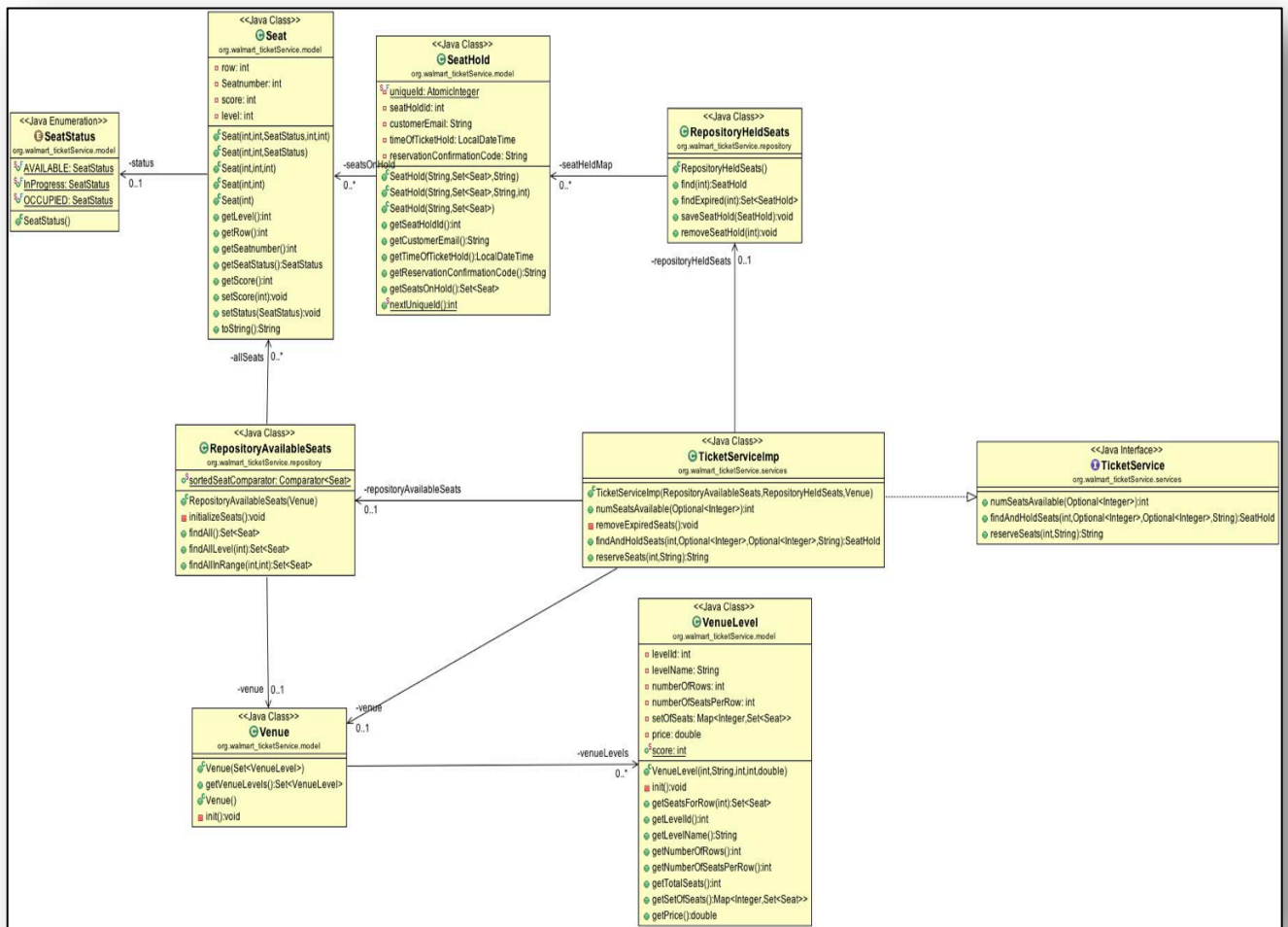
6. Class Diagram

7. Possible Enhancements (Can be implemented with some more time):

1. The In Memory repository can be replaced by a database.
2. The service can be modified to run in a multi-threaded environment
3. The code can be structured into MVC pattern