

# Walmart Ticket Service Application

Akarsh Cholaveti, [cholaveti.akarsh@gmail.com](mailto:cholaveti.akarsh@gmail.com)

## Application Details:

The Ticket service application will have 3 main functionalities

1. Get the available seats for the user
2. Find and Hold the seats for the user
3. Reserve the held seats for the user.

The application can be run in four different modes. Each mode has its own advantages and disadvantages of the ticketing system. The developer can use his/her mode of choice.

- 1. Ticket Service Atomic:** This mode will run the application in the atomic way by holding all the resources for the user until he/she completes the operation. This mode will use the accurate values for the user by removing timed out objects before the start of every operation. Thus by giving the best seats possible.  
**Advantages:** Accurate data is used for the operations; user will be guaranteed to get all the requested seats (if shown available before the hold)  
**Disadvantages:** Removing the holds before every operation for every user will affect the performance of the system, holding all the resources for every user will make other users to wait for more time to get access.
- 2. Ticket Service Atomic with periodic hold removal:** This mode will run in the atomic way by holding all the resources for the user until he/she completes the operation. This mode will remove the holds periodically (for every 2 minutes) but not for every request. This mode may not use very accurate information for performing the operation.  
**Advantages:** Performance of the application is better than the previous mode, user will be guaranteed to get all the requested seats (if shown available before the hold)  
**Disadvantages:** Accurate data might not be used for the operations for user, holding all the resources for every user will make other users to wait for more time to get access.
- 3. Ticket Service Parallel:** This mode will run in the parallel way by giving all resources to all the active users and synchronization is used only at seat level. This mode will use the accurate values for the user by removing timed out objects before the start of every operation. Thus by giving the best seats possible.  
**Advantages:** Wait time is less for the users, accurate data is used for every operation of every user  
**Disadvantages:** The user may not get all the requested seats even though they are available at the beginning of the operation. Removing the holds before every operation for every user will affect the performance of the system
- 4. Ticket Service Parallel with Periodic Hold Removal:** This mode will run in the parallel way by giving all resources to all the active users and synchronization is used only at seat level. This mode

will remove the holds periodically (for every 2 minutes) but not for every request. This mode may not use very accurate information for performing the operation.

**Advantages:** Wait time is less for the users and performance of the application is better than the previous mode.

**Disadvantages:** The user may not get all the requested seats even though they are available at the beginning of the operation. Accurate data might not be used for the operations for user

### **Running The Application:**

To run the application, please download and extract the project. Go to the target directory and use the command “java -jar Walmart-Ticket-Service-Application-1-jar-with-dependencies.jar ‘mode-choice’” or build the project using the following commands in the project directory.

1. mvn clean
2. mvn package

The newly built jar file can be found in the target folder of the project directory.

As discussed previously, the project can be run on four different modes and choice of mode should be given as command line argument while running the jar. The choices are listed below.

1. TicketServiceAtomic
2. TicketServiceAtomicWithPeriodicCache
3. TicketServiceParallel
4. TicketServiceParallelWithPeriodicCache

### **Input and Output:**

The program has no interactive input. The program will start with the default values and execute the threads. The details of the operation are printed in the command window

### **Details and Assumptions:**

1. The four levels and the seat layout for each level will be created and loaded into memory after the main function is started.
2. The seathold waiting time is set as default value of 2 minutes.
3. To demonstrate the timeout functionality, the ‘walmart’ thread is made to wait for 2.5 minutes before the reservation. Therefore, the thread will cross the bound time and it will fail.
4. The periodic removal of holds for mode 2 and mode 4 will run in separate in thread. For testing purposes, the periodic thread will be killed if there are no active threads in the application.
5. A seat can be reserved only after it is held.

### **Unit Testing**

The unit test cases of the application are executed using the Junit Suite of 3.8.1 version. If the application is built using the maven instructions as described above. The unit test cases will get executed before the Jar is built.

**Dependencies:**

Following language, tools and dependencies are used for the project.

1. Java 1.8
2. Junit 3.8.1
3. Maven build tool
4. Guava 19 (Application Cache)