**Problem Definition**

“Metropolitan Convention Center” would like to create a web site where participants can sign-up and buy tickets for events held in the center’s main hall and meeting rooms.

As a first step they are interested in seeing a prototype that allows users to:

* Create a login for the site
* Allow user to login to the site
* Look at a list of events
* Ability to Register for events

**Application Design Document**

* Customer and related fields *\*CLASS*
  + Name, Email, Password (masking)
* Events and related fields *\*CLASS*
  + Code, Title, and Description
* Event Registration *\*CLASS*
  + Event, Customer, Date, and Description
* Create a Login Java Class that encompasses process and/or additional Java classes to complete login and authentication to web application
  + Customer class
  + Authenticator class
  + Error handling, API status codes (404, 200, etc…)
* Authentication during customer creation, display auto-generated token when new user is created on front-end.
  + Securing authentication; prevent Token from being accessed with API GET requests
* Create API for Customer GET, POST, DELETE, PUT
* Create API for Events GET, POST, DELETE, PUT; get explicit list of all Events
* Create API for Event Registration GET, POST, DELETE, PUT
* Use a DB back-end to store the information (mySQL)
* Build a Jenkins pipeline that kicks off a build automatically on commit to master branch; utilize webhooks
* Build a Dockerfile to containerize the application
* Instantiate a Web Server (Tomcat, NGINX, etc…)
* Implement JUnit Tests for source code

**JIRA Backlog**

* Build a Jenkins pipeline that kicks off a build automatically on commit to master branch
  + Utilize webhooks
* Build a Dockerfile to containerize the application
* Create API for Customer GET, POST, DELETE, PUT and test with Postman
* Create API for Events GET, POST, DELETE, PUT; get explicit list of all Events and test with Postman
* Create API for Event Registration GET, POST, DELETE, PUT and test with Postman