

## TODO Sheet for Ticket System

A lot of code has been provided for you in week10/project. There may be some minor bugs - but you should be able to work around these and deliver a good result.

The JSPs which have been provided are visually terrible but give an idea of basic functionality. You could improve these using your web development skills but remember that in this unit functionality is much more important than look and feel so don't spend too much time trying to make the pretty. If you do want to play - look at the early JSP examples - week2-3 where I used bootstrap.

If you have problems with your application, and particularly if you change the data type objects in the model, you may need to delete the old database in tomcat and start again. You will find the location under tomcatserver >properties > Catalina home . Delete the temp directory and possibly your application under work\Catalina\localhost

TASKS TO DO	Detail
you need use cases	your use cases should define what operations need to be supported in the application. Where functionality is already provided, you can reverse engineer the use cases. For new functionality you need matching use cases.
you need a test plan	test plan must contain tests which illustrate each of the use cases Each usecase may have more than one test; what happens when the user does correct actions and what happens when there is incorrect input. Don't over do this - I am more interested in the correct response use cases than covering all of the incorrect responses. Use cases should at a minimum have test number, title, description, preconditions, actions to perform, post conditions expected after the action. Lots of example formats online or invent your own
you will need robustness diagrams	look at the smartMeter example for inspiration of how to represent the client poller. Each JSP, Façade and DAO will be a boundary object. The MeterConfiguration is an example of a data object and the controller objects basically represent the actions on a jsp or the methods on an interface.
<b>client side</b>	
gate JSP's	
<ul style="list-style-type: none"><li>validate ticket code</li></ul>	You need to check the ticket has been issued by a valid authority (using PKI) use TicketEncoder and TicketEncoderImpl See also TicketEncoderImplTest in project-model for examples
<ul style="list-style-type: none"><li>check time, rate and departure zone against arrival zone before opening gate</li></ul>	You need the check that the customer has paid to travel to the zone where the gate is.

TASKS TO DO	Detail
<ul style="list-style-type: none"> <li>allow user to enter arrival station</li> </ul>	This is so that you can test the issues ticket in different scenarios
<ul style="list-style-type: none"> <li>allow user to enter arrival time</li> </ul>	
ticket machine JSP's	
<ul style="list-style-type: none"> <li>allow user to set uuid</li> </ul>	complete in changeConfig.jsp
<ul style="list-style-type: none"> <li>poll controller service for configuration</li> </ul>	complete and uses ConfigurationPoller in project-client-rest
<ul style="list-style-type: none"> <li>allow customer to select from destination station list</li> </ul>	there is an example but incomplete clientstationlist.jsp
<ul style="list-style-type: none"> <li>issue peak or off peak ticket according to time</li> </ul>	
<ul style="list-style-type: none"> <li>allow user to enter departure time</li> </ul>	
<b>server side</b>	
<ul style="list-style-type: none"> <li>complete controller rest interface</li> </ul>	The TicketMachineRestService in project-web responds the ReST call. This uses the stationDAO to give a list of stations but has a hard coded chargingDetails which you will need to access using the PriceCalculatorDAO
<ul style="list-style-type: none"> <li>allow controller to create / delete update stations (CRUD)</li> </ul>	the stationList.jsp and station.jsp allow you to update stations but you cannot yet add ticketMachines
<ul style="list-style-type: none"> <li>allow controller to create delete update ticket machines</li> </ul>	
<ul style="list-style-type: none"> <li>allow controller to assign ticket machines to stations</li> </ul>	