**World File – ‘new\_jersey.world’**

This file details that the SDF version of the world, which is named ‘default,’ is 1.4. The lighting of the world is named ‘sun’ which casts a directional light as well as cast shadows, also has a position and values for how much the light diffuses, is specular, and various attributes for its attenuation. The world uses physics type ‘ode’ which updates in real time. There are also properties for a model named ‘jersey\_barrier,’ which is static with collision on the model’s various sides (listed as ‘collision name’). Afterwards, the model ‘ground\_plane\_0’ is similarly detailed. The model state is known as ‘state world’ with values in simulation and real time. The file proceeds to list all the various other jersey barriers that are used in the map, along with a ‘mobile\_base,’ which has wheels. After listing the models, there is a property for the gui where the camera is positioned. The ‘spherical\_coordinates’ have properties that adjust coordinates in the world. The mobile\_base model has further properties listed after the coordinates that feature various links involved, such as ‘base\_footprint,’ which has inertial properties as well as various collision properties. These properties are notably for the base link, camera link, etc. Each of these collision links has a position in the world, a geometric attribute, as well as friction and bounce for its relations to physics. After the collision properties are visual properties for those collision links. The visual properties contain attributes for the position, mesh scale and the source URL of the mesh. The file then moves onto properties for the sensors, such as the bumper and camera, with update rates, position, and which collision, if any, it is associated to. It also includes the sensor’s velocity decay, and, for cases that include plugins such as the camera, there are additional attributes involved. Then, the file displays link and joint properties of the wheels as well as a kobuki plugin for its movement. The models of the barriers are listed again, up to barrer 55, but with additional properties for its mesh URL location and collision attributes. These include the collisions for ‘upright,’ ‘base,’ ‘base2,’ ‘base3,’ ‘left\_angle,’ and ‘right\_angle.’ Each collision has properties for position, size, friction and contact; the latter two properties reference the physics relation ‘ode.’

**Launch File – ‘robotics\_project\_world.launch’**

The launch file details several launch arguments (world\_file, base, battery, gui, stacks, 3d\_sensor) which serve as variables holding specific values, which control certain events. For example, the argument named ‘base’ can create a Roomba. The file then lists properties of which files to include. These included files also listed arguments that are passed into them. The next part is the node property, which has attributes for the package (which is ‘robot\_state\_publisher’), type, and name. The node, additionally, contains a parameter called ‘publish\_frequency’ that is stored as a double with a value of 30.0. The final part of the launch file details node packages for the fake laser. Both of these packages are of the ‘nodelet’ type, and the latter of the two packages, called ‘depthimage\_to\_laserscan’ has an argument loading a laserscan from a directory, and it also contains various parameters (scan\_height, output\_frame, range\_min, image, scan) with their assigned values.

**How to Launch and Run on CSN Linux Boxes in FH 300**