THIS IS OLD/OUTDATED

Big picture: create 3D models of golf courses

Specific tasks:

1. Adrian: augment the output of your clickable app
2. Adrian+Davis: define a set of different types of trees
   1. I'm envisioning that a tree has a **point** (location), a tree **type** (from a list of "bushes", "pine tree", "deciduous tree"?), a **height** and a **radius**. Maybe a "**trunk height**" for how far the trunk goes up before the branching begins (with some reasonable default based on the height or 0 for bushes)?
   2. it would probably be helpful to have a "group of trees" type so if there is a large area of similar trees, Adrian's app can allow us to trace the entire area and fill it with multiple copies of the typical tree in the area (either by manually clicking individual tree locations or by automatically filling in the area)
   3. Adrian: make your app allow for trees.
   4. Davis: make your blender code allow for trees.
3. Adrian+Davis+Prof. Sanft: define a set of attributes for bunkers (similar to attributes for trees)
   1. this might require looking at the shotlink data. I'm thinking attributes include **depth** (how deep is the bunker--google's elevation data probably won't capture this accurately) in feet (or perhaps categories like: shallow, average, deep?) and face **steepness** (a deep bunker with a steep face is harder to get out of than a deep bunker with a gradual incline of the face) perhaps in 3 levels: steep, average, flat. Defaults could be average and average.

Model building strategy:

* crude elevation data from google elevation API
  + Prof. Sanft has funding for this but it goes away after this semester!
  + How bad is it?
    - compare it to a course with ShotLink data
* Adrian's clickable outline app to define boundaries of fairways, greens, bunkers, trees, etc.
  + How does Adrian's code handle overwritten things?
    - a dictionary (keys are "f", "s", "g", "w") of lists of lists
  + Can Adrian's code output the data in addition to the png?
    - output:
      * latitude/longitude coordinates of the bounding box
      * the dictionary of the click data
      * the pixel dimensions (e.g. 1280, 720)
      * .png file
  + it would probably be helpful to have a "group of trees" type so if there is a large area of similar trees, we can trace the entire area and fill it with multiple copies of the typical tree in the area (either by manually clicking individual tree locations or by automatically filling in the area)
  + ~~perhaps augment with computer vision? future enhancement~~
* an additional "attributes editor" to edit bunkers, trees
  + Can we use ShotLink to find the range of bunker depths, then set up a "depth" attribute of, say, shallow, medium, deep, perhaps also a face height attribute or something (face height+steepness?)?
    - we can look at shotlink data to get a sense of distributions of attributes
* Use strackaline data or inclinometer data to construct detailed green maps

Ultimate goal: get sufficiently accurate models to simulate realistic shots. That is, if we have an accurate shot simulator, we can determine where a shot strikes the ground and the landing conditions (velocity vector, spin rate, spin axis), then, based on the 3D course model slopes (and other variables like ground firmness and length of grass), we can predict the distribution of where the ball ends up. We can then use this for professional or UNCA golf team strategy work.