Flow Chart Algorithm:

1. First user selects their DGU and the major that they are interested in looking at.
2. From there a user would have to select a track. Once the user has selected a track a flowchart will be created.
   1. The chart will be organized in groups based on the following selection criteria. Each entity is a one to many relationship.
      1. DGU to Majors
      2. Majors to Tracks
      3. Tracks to Groups
      4. Groups to Sets
      5. Sets to Courses
   2. The chart will be organized by groups:
      1. Pre-requisites
      2. Core Requirements
      3. Electives
   3. Within each set there will exist a list of courses.
3. The flowchart visualization can be displayed using two different algorithms:
   1. If a flowchart for that particular track, group, or set already exists in the database the information will be retrieved from this source.
      1. Track view:
         1. Using the track ID and find all groups inside that particular track.
         2. Get each group ID and find all sets inside that group.
         3. Get each set ID inside each group and find the courses inside the sets.
         4. Now place each course information inside an array based on the group’s position, the sets positions, and finally the courses position.
      2. Group view:
         1. Using the group ID find all sets inside that group.
         2. Get each set ID inside each group and find the courses inside the sets.
         3. Now place each course information inside an array based on the sets positions, and finally the courses position.
      3. Set view:
         1. Using the set ID find the courses inside the set.
         2. Now place each course information inside an array based on the courses position.
   2. If a flowchart does not exists we create a standard or default view for the track, group, or set. The information is pulled using the curriculum database.
      1. Track view:
         1. Using the track ID and find all groups inside that particular track.
         2. Get each group ID and find all sets inside that group.
         3. Get each set ID inside each group and find the courses inside the sets.
         4. Apply the requisites algorithm on the set of courses to order the course set in ascending order based on the number of pre-requisites each course has.
            1. Iterate through the list of courses while comparing each course to the number of pre-requisites it has by retrieving information from the his\_requisites table.
         5. Now place each course information inside an array based on the group id, set id, and the course order. The group id and set id are placed in ascending order.
      2. Group view:
         1. Using the group ID find all sets inside that group.
         2. Get each set ID inside each group and find the courses inside the sets.
         3. Apply the requisites algorithm on the set of courses to order the course set in ascending order based on the number of pre-requisites each course has.
            1. Iterate through the list of courses while comparing each course to the number of pre-requisites it has by retrieving information from the his\_requisites table.
         4. Now place each course information inside an array based on the set id and the course order. The set id are placed in ascending order.
      3. Set view:
         1. Using the set ID find the courses inside the sets.
         2. Apply the requisites algorithm on the set of courses to order the course set in ascending order based on the number of pre-requisites each course has.
            1. Iterate through the list of courses while comparing each course to the number of pre-requisites it has by retrieving information from the his\_requisites table.
         3. Now place each course information inside an array based on the course order.
4. Users can modify the structure of these groups by moving these courses dynamically. When moving courses their positon value will be updated.
5. If the user clicks submit when finished modifying the flowchart, the changes will be replicated to the database so when that flowchart is reloaded the order is preserved.
   1. Only admins and advisers have access to this functionality. Once the submit button is clicked the respective controller for Track, Group, or Set is called.
      1. The flowchart information is then passed to the function via post data. From there the function iterates through the track, group, or set and parses important information such as id, position, and flowchart id.
      2. Once the information has been parsed the values are submitted to the database.