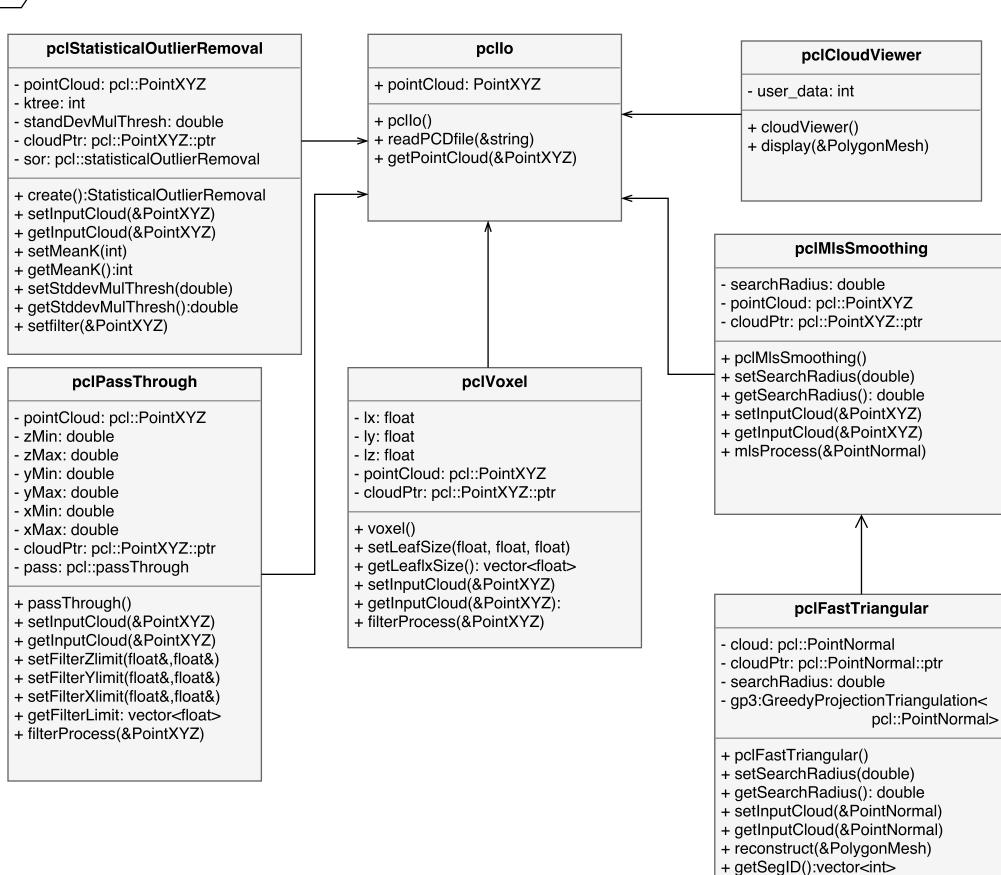
Midterm



findNearestPoint

- specificPoints:vector<double>
- · cloud: pcl::PointNormal
- +readtext(&String):vector<double>
- +setInputCloud(&PointNormal)
- +findNearestProcess():vector<int>
- +setPosition(vector<float>& pos): void

delaunay3

- pts: vector<Shx>
- · pt: Shx

Final

- cloud: pcl::PointNormal
- + delaunay3()
- + setInputCloud(&PointNormal)
- + putPointCloudIntoShx()
- + processDelaunay(vector<Triad>)
- + getShx(vector<Shx>)

dijkstraPQ

- V: int
- parent: int*
- adj: list<pair<int, double>>*
- triads: vector<Triad>
- triPartID: vector<int>
- cloud: pcl::PointNormal
- cloudPtr: pcl::PointNormal::ptr
- + dijkstraPQ()
- + addEdge(int,int,double)
- + shortestPath(int,int)
- + cal2Point(int ,int):double
- + distanceCalculation(std::vector<int>&)
- + setTri(vector<Triad>)
- + computeWeight()
- + returnDijkstraPath(int, int, vector<int>)
- + setInputCloud(&PointNormal)
- + returnDijkstraPath(int, int, vector<position>)

kukaControl

- + kukaControl()
- + initializeJoints(KDL::JntArray)
- + initializePoints(JointTrajectoryPoint,int, float)
- + nameJoints(JointTrajectory, int)
- + evalPoints(JointTrajectoryPoint, KDL::JntArray ,int)