Last Updated By: Shannon Harris

Last Updated Date: 05/18/13

This document is now defunct and no longer represents actual class structure. Please see the actual python files or the class diagram for an up to date representation of class structure.

class Patient{

  private Horizontal;

  private Vertical;

  getFace();

  analyzeEyes(){

     results = strabismus();

     results.append(astigmatism());

     results.append(cataracts());

     return results;

  }

  strabismus(){

     // strabismus logic goes here

  }

  astigmatism(){

     //astigmatism logic goes here

  }

  cataracts(){

     // cataracts logic goes here

  }

}

class Eyes{

// include getters and setters

// eye setters must modify regions and crescents.

  private left.jpg

private leftRegion

private leftCrescent

  private right.jpg

private rightRegion

private rightCrescent;

  getEyes(photo){

     \\eyeDetection.py code goes here

// sets left and right

//returns regions

}

class Horizontal **extends**/implements? Eyes {

// include getters and setters

   Horizontal(newPhoto){

     getEyes(photo); // populate left and right from what was

returned from getEyes

  }

  private left.jpg;

  private right.jpg;

  private photo.jpg;

}

class Vertical **extends** Eyes {

// include getters and setters

  Vertical(newPhoto){

      rotate(newPhoto)

      getEyes(newPhoto);

  // populate left and right from what was returned from getEyes

  }

 private left.jpg;

 private right.jpg;

 private photo.jpg;

 rotate(photo){

    // rotate the photo

 }

}

controller(){

 thisPatient =  new Patient(Horizontal(photo from UI), Vertical(photo from UI));

// pass regions that horizontal and vertical’s regions to the UI

if user confirms

continue

else if user resets regions

reset the relevant regions using setter methods

// crescents and pupils will be detected when the new

// patient is made

// pass the keypoints of all the eyes to the UI

if the user confirms

continue

else if user resets keypoints

reset keypoints using setter methods

 print thisPatient.analyzeEyes();

}

TODO

\* place crescent finding code

\* place user reset keypoints code

IDEAS

Eyes has two eye vars left and right and each eye has a pupil each pupil has a crescent and each crescent has a region and keypoints.

RESOURCES

\* Python OOP: <http://net.tutsplus.com/tutorials/python-tutorials/python-from-scratch-object-oriented-programming/>

\*The PIL Image Library seems to be extremely useful. Attached is a possible example of how we can rotate the picture with a simple library call. check it out!

<http://www.daniweb.com/software-development/python/code/216426/rotating-an-image-python>