Crack-detector - Requirements analysis

1. Project details

Crack-detector is a web based application which detects the cracks from buildings, large pipes, roads etc. This detection is realized by user provided images via the web interface.

1. Actors
   1. The application user can choose a classifier, from a list of available algorithms. The results can then be compared with real life classified images. The provided results are in first phase a “Positive” or “Negative” response whether the image has or not cracks. In the second phase the user could see where the crack is located in the image.
   2. The administrator type user will be able to train the classifier by providing the image training set. This training set consists of several thousand positive and negative classified images.
   3. <----more user types---->
2. Use cases
   1. The administrator trains the classifier
      1. The admin user uploads the training file.
   2. The user uploads an image to be classified
      1. Using the web application the user uploads an image
      2. The image is shown on the web page
      3. User selects “Classification” type
      4. The user selects the classifies from the classifiers list
      5. The classifier computes the result
      6. The result will be shown as a “Positive” or “Negative” manner in the web page
   3. The user uploads an image for crack localization
      1. Using the web application the user uploads an image
      2. The image is shown on the web page
      3. User selects “Localization” type
      4. The user selects the classifies from the classifiers list
      5. The classifier computes the result
      6. If the result is:
         1. Positive, the a red square will be shown around the crack
         2. Negative, the image will contain a “Negative” green label
3. Components and structure of the application
   1. Web application that communicates with the server and provides a simple user interface
   2. Command line application used for the training of the classifiers, the command line app communicates with the server side
   3. The server application that stores the training results and sends responses to the web application and the command line application
4. Diagrams
   1. Web app: use case diagram, structure diagram, class diagram
   2. Console app: use case diagram, flow diagram, layers diagram, class diagram
   3. Server app: use case diagram, package diagram, class diagram