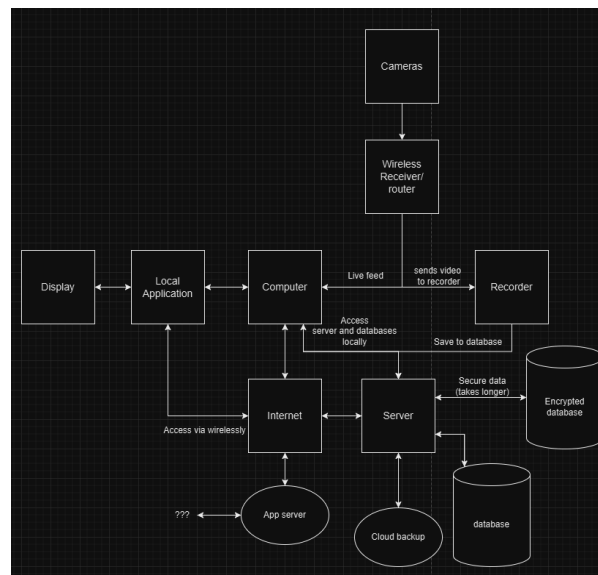
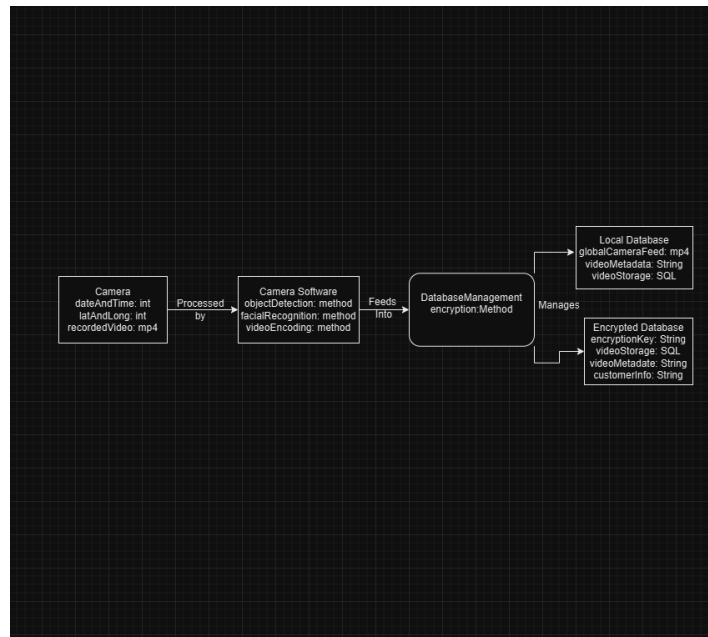


Design specification



Updated databases and pinpointers, added app server to support a mobile app.

Test set one

securityVideo()

Unit Test:

1. Test to ensure that the metadata gets properly assigned to the right video.
2. Steps:
3. Record 30 seconds of video from security camera.
4. Track metadata manually(File type, data, resolution)
5. Upload video to system
6. Ensure the metadata tracked is the same.

Unit Test Code

```
securityVideo test = new securityVideo();  
String fileType = mp4;  
Int dataSize = 300;  
Int resolution = 1080;
```

```
test.upload();
```

```
//TEST
```

```
if (test.getFiletype == fileType && test.getDataSize == dataSize && test.resolution == resolution)  
{  
    return true;  
}  
Else{  
    return false.  
}
```

Integration Test:

1. Upload video to be encrypted, then request it back from the database.
2. Steps:
3. Record 30 seconds of video from security camera
4. Upload to system, then through database.
5. Access recorded video from main server

```
securityCamera video = new securityCamera();  
securityCamera test = video.record(30);  
video.uploadEncrypted();
```

```
//TEST
```

```
if(video.videoFile == test.videoFile) {  
    return true;  
}  
Else{  
    return false;  
}
```

System level test:

At full capacity, the system should be able to automatically record video, track relevant video data, and then upload it to the database. After uploading, the video files and associated metadata should be accessible from the main server and any added admin accounts.

Test Description:

This tests the video storage system of the camera. We are testing to make sure that the video of the camera is correctly assigned the correct metadata so that when the user attempts to access the video's data from the database, the correct information is displayed. The unit test tests to make sure the video is assigned the correct metadata by first recording 30 seconds of video from the camera. We then manually and automatically track the file type, the data size, and the resolution of the video. If the data doesn't match, then the test fails. This covers the assignment part of the system by testing that the video is always assigned the correct information.

The integration test ensures that videos are properly stored in the database. It does this by first recording 30 seconds of video from the camera and then uploading it to the database using the `uploadEncrypted()` method. This method will encrypt the video to make sure the video's information is protected and then uploads the video to the database. The test checks if the video file is the same as the test video file and if it outputs false, then the video data is being mismanges. This covers the encryption and uploading part of the system by testing the full upload and download process.

The system level test will test the whole system to make sure that the video is assigned the correct information and then tests that the video's information is protected and uploaded properly to the database.

Test set two

securityCamera()

Unit test:

1. Have someone stand facing a security camera, and record a 30 second video.
2. Upload the facial scan from the test person
3. Compare the facial recognition's face with the manually recorded one.

Unit Test Code:

```
cameraSoftware test = new cameraSoftware();
cameraSoftware temp = new cameraSoftware();
temp.facialRec = system.in();
securityCamera testVid = new securityCamera();
testVid.record();
test.process(testVid);
```

```
//TEST
if(test.facialRec == temp.facialRec) {
    return true;
}
Else{
    return false;
}
```

Integration test:

1. Have someone stand facing a security camera, and record a 30 second video.
2. Repeat this process 3 times.
3. Search the database for the same face among recent videos.

Integration test code:

```
securityCamera testVid = new securityCamera();
securityCamera testVid2 = new securityCamera();
securityCamera testVid3 = new securityCamera();

testVid.record()
testVid2.record()
testVid3.record
cameraSoftware tester = new cameraSoftware();
tester.process(testVid);
tester.process(testVid2);
tester.process(testVid3);
```

```
//TEST
If
(tester.faceRec(tester.getVideo(testVid)) == tester.faceRec(tester.getVideo(testVid2)) ==
tester.faceRec(tester.getVideo(testVid3))) {
    return true;
}
Else{
    return false;
}
```

System level test:

At full capacity, the security camera should be able to document people recorded by the facial recognition system. When provided with a picture that has suitable quality, it should be able to determine and find all times that the given person was recorded.

Test Description:

In this test set, we test camera recording implementation and the facial recognition feature of the program. In unit test, we cross-examine a recording from a “camera” and then a sample to see if they are similar.

In integration test, we record multiple videos, process them so the facial recognition can read the video and test if it can recognize faces, doesn’t, and false recognize faces.

In system level test, we test the entire procedure of facial recognition, and all use cases it could have.

Test set three

camera()

Unit test:

Test the code with sample input code, then cross examination with one typed by hand of what it should look like

Unit Test Code:

```
Camera camSample = input;  
Camera camTest;
```

```
camTest.dateAndTime = "00112233";  
camTest.dateAndTime = "12345";  
camTest.dateAndTime = test.mp4;
```

```
// output as file with all information
```

```
if(camSample.equals(camTest))  
    Return true;  
Else  
    Return false;
```

Integration test:

Test if video recording can read from camera and try to video encode it.

Integration test code:

```
cameraSoftware camTest;
```

```
Camera camSoftSample = input;
```

```
cameraTest.videoEncoding() = camSoftSample;  
camTest.objectDetection() = objectDetectionDefault;  
camTest.facialRecognition() = facialDefault;
```

```
If (camtest.equal(softwareVideo))  
    Return true;  
Else  
    Return false;
```

System level test:

Camera actively outputs what it sees from its lens, gets recorded along with date and time of recording. It will also note which camera it is given by the latitude and longitude of the camera.

Test Description:

In this test, we test the camera, if it can receive input from the camera devices and if we can encode the input into a readable format/video.