Functional Requirements(uWatch digital forensic tool)

Group Name: MPHETamines

Taariq Ghoord	10132806
Martha Mohlala	10353403
Phethile Mkhabela	12097561
Sboniso Masilela	
Harrison Maphuti Setati	12310043

Git repository link:
`https://github.com/MPHETamines/MPHETamines/`

Date: 20 May 2015

Contents

1 Introduction

This document sets out the Software Requirements Specification and Technology Neutral Process Design for the COS 301 group project entitled *Online Neighbourhood Watch(ONW).* The aim for this project is to follow a combination of waterfall and agile software development approach within which the application functionality is developed iteratively. The information provided in this document is presented in such a way as to provide precise and testable requirements. The emphasis is on performing an upfront software architecture engineering for iterative stimulation of the detailed requirements for a use case so that each use case can be built, tested and deployed before the detailed requirements for the next case are added.

1.1 Background

Crime is a prominent issue in South Africa as it is all over the world, Many criminal activities go unresolved or even attended to due to the lack of evidence or concrete witnesses. Mobile applications have become increasingly popular all over the world and are used in our everyday and work life for common things such as checking the weather; maps for directions and news feed updates. Digital forensic science hopes to utilise this increasing growth in the use of mobile applications to address the lack of evidence to crime cases in South Africa. The application is referred to as online neighbourhood watch(ONW) accessible via mobile devices and computers over the internet. The two main users of the ONW model are the uploader (user of the mobile device) and the forensic investigator or law enforcement agent. This tool is to be used by the citizens of South Africa to capture, collect and store potential evidence which can later be viewed and analysed by the Police department and used in the prosecution and detention of criminals.

1.2 Purpose

The online neighbourhood watch is aimed to provide a tool that can assist the South African police services(SAPS) reduce crime by enabling the members of the community to be part of the judicial system. The ONW application captures and stores potential digital evidence of criminal activities which will then be accessed by law enforcement agents and digital forensic investigators. The goal is to enhance the successful rate of trials and secure a higher number of convictions. The application can be used in various scenarios, basically it should be used in any setting where a community member feels like a crime has been committed, it is then up to the ONW model to decide if the uploaded data is a potential crime scene. The application should enable a user to capture digital evidence such as digital photographs, audio and video of a potential crime scene in the domain of the ONW to maintain integrity of the data, the data is then stored into the ONW repository.

1.3 Scope

The Scope of the ONW is too broad and with the time allocated to the team for this project we decided to focus on only one element of the application, users will only upload pictures as evidence for now and disregard the video and audio part of the ONW model. There are three aspects of the system the mobile application side, utilised by a community member; the desktop side where a law enforcement agent logs in and the algorithm which sits in between the mobile and desktop. The algorithm will be used to determine if the photo uploaded is a crime photo. we will focus on human and object detection for now and maybe at a later stage if time permits other media could be uploaded as well.

1.4 References

S.Omeleze,H.S.Venter, Toward a model for acquiring digital evidence using mobile devices. Information and Computer Security Architecture (ICSA) Research Group, University of Pretoria.

1.4.1 Assumptions

The ONW will be tested around Pretoria Hatfield with the local SAPS precinct. If it passes the test then it will be progressed to other cities in Gauteng, eventually to the whole South Africa.

2 Methodology

[Describe the overall approach used in the determination of the FRD contents. Describe the modeling method(s) so non-technical readers can understand what they are conveying.]

3 Functional Requirements

3.1 Context

[Provide a context diagram of the system, with explanations as applicable. The context of a system refers to the connections and relationships between the system and its environment.]

3.2 User Requirements

[Provide requirements of the system, user or business, taking into account all major classes/categories of users. Provide the type of security or other distinguishing characteristics of each set of users. List the functional requirements that compose each user requirement. As the functional requirements are decomposed, the highest level functional requirements are traced to the user requirements. Inclusion of lower level functional requirements is not mandatory in the traceability to user requirements if the parent requirements are already traced to them. User requirement information can be in text or process flow format for each major user class that shows what inputs will initiate the system functions, system interactions, and what outputs are expected to be generated by the system. The scenarios should be comprehensive, to the extent that all user types and all major functions are covered. Give each user requirement a unique number. Typically, user requirements have a numbering system that is separate from the functional requirements. Requirements may be labeled with a leading U or other label indicating user requirements.]

3.3 Use Case Diagrams

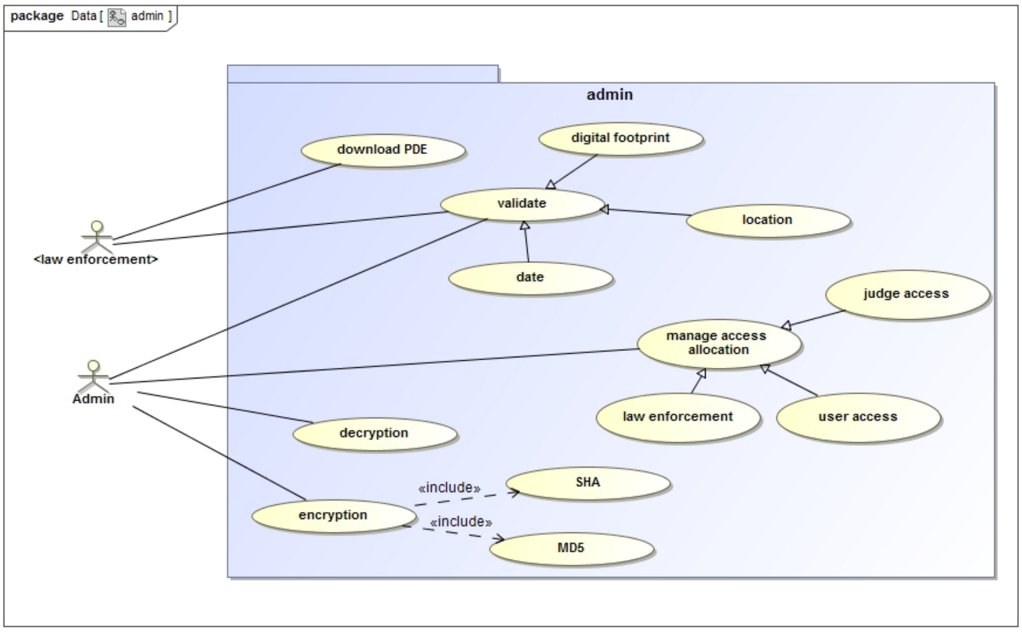

Figure 1: Functional Requirements: Admin/Download Subsystem

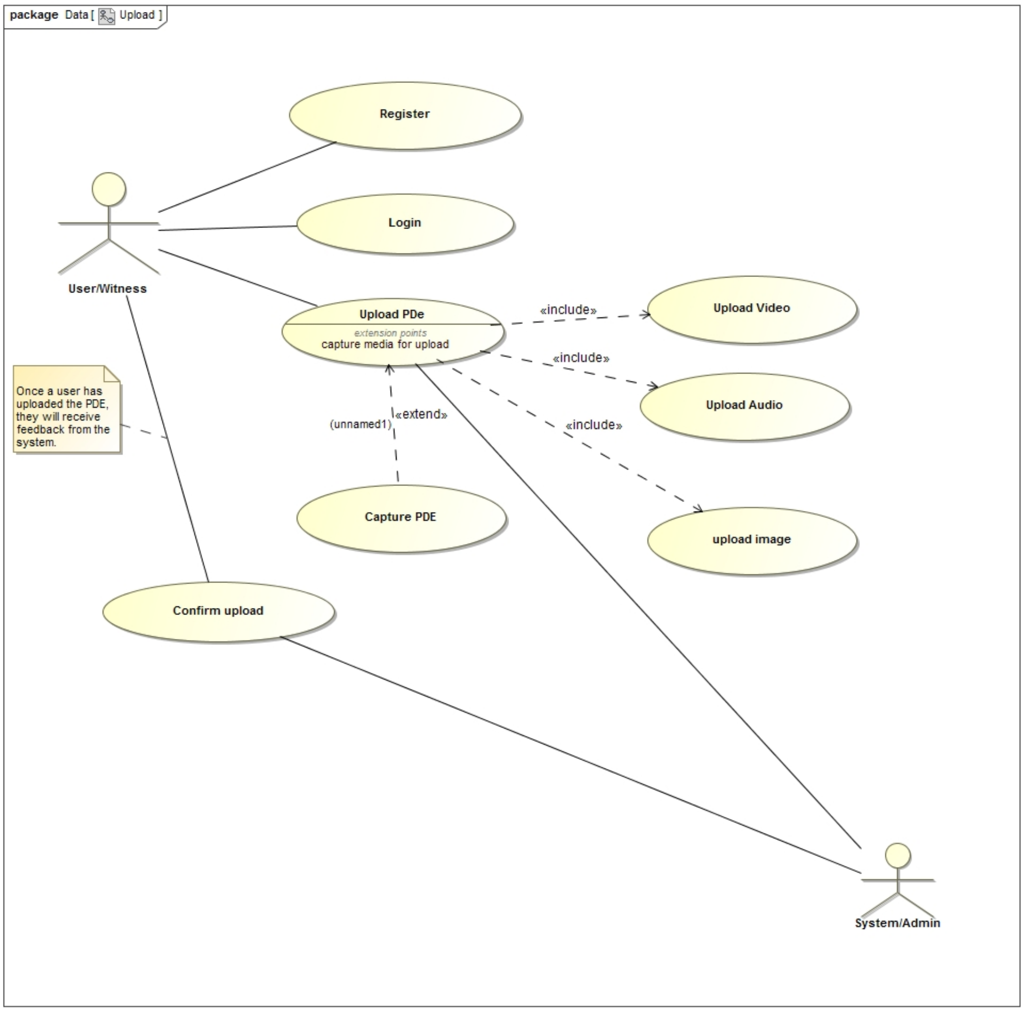

Figure 2: Functional Requirements: Upload Subsystem

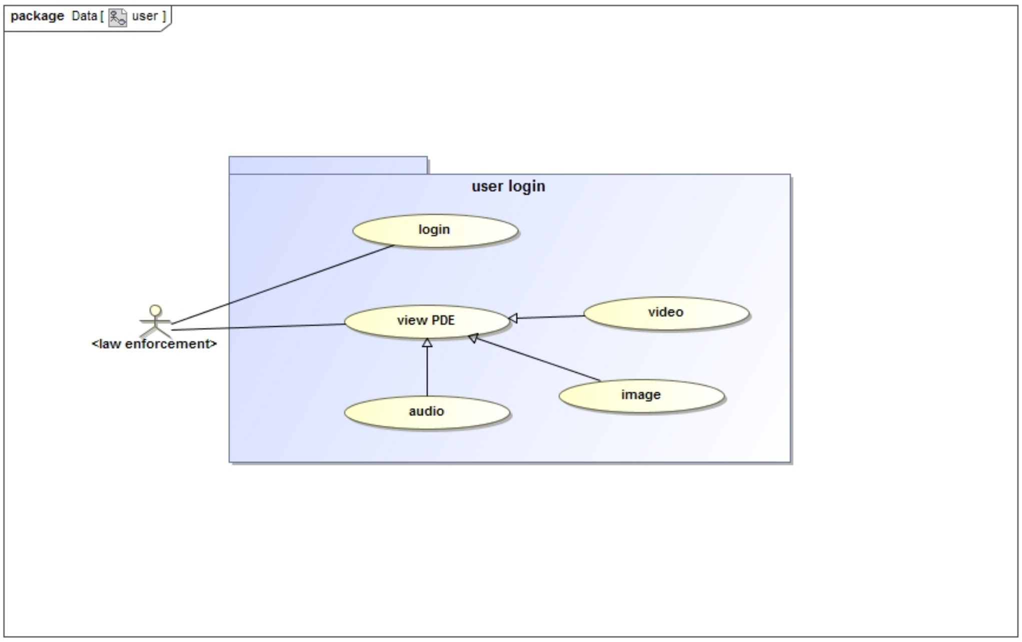

Figure 3: Functional Requirements: View Subsystem

3.4 Class And Sequence Diagrams

[service contracts and their corresponding sequence diagrams to be shown here]

3.5 Activity Diagrams

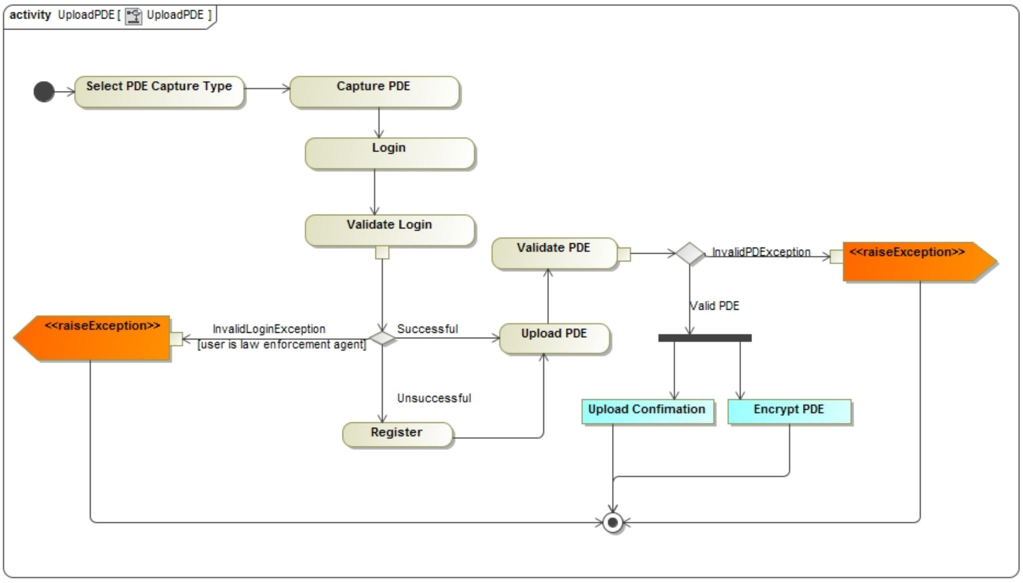

Figure 4: Process Specification: Uploading Potential Digital Evidence

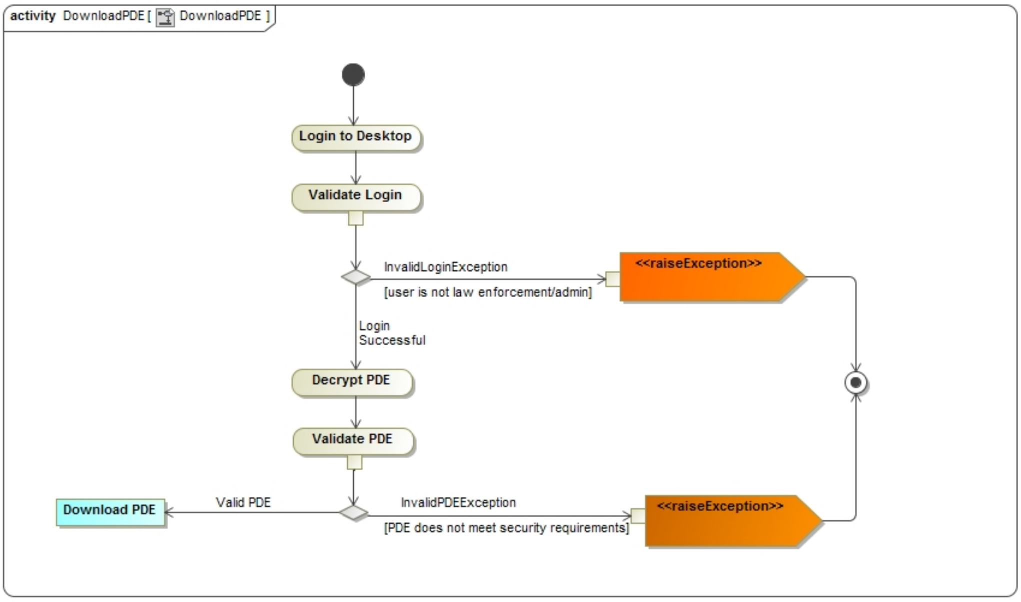

Figure 5: Process Specification: Downloading Potential Digital Evidence

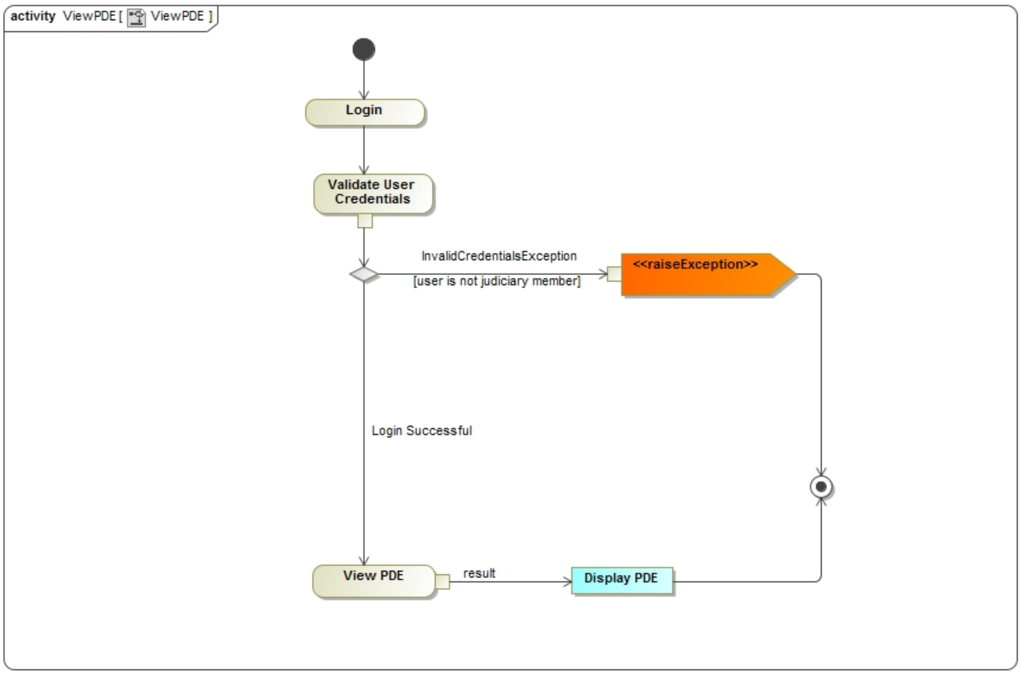
activity ViewPDE [ViewPDE]
Login
Validate User Credentials
InvalidCredentialsException
[user is not judiciary member]
<<raiseException>>
Login Successful
View PDE
result
Display PDE

Figure 6: Process Specification: Viewing Potential Digital Evidence

3.6 Functional Requirements

3.6.1 The Administration Module

The functionality provided by the Administration module includes the following:

- It manages access to the systems desktop version of the model
- It provides means to download the potential evidence
- It provides functionality to validate the evidence, whether by location, date or digital footprint
- It provides encryption and decrypt functionality

3.6.1.1 Use Cases
The Administration module provides services to download, validate, encrypt, decrypt and manage access to the system
DownloadPDE-Priority:high
3.6.1.1 Service Contract
pre-condition: For any media to be downloaded, the media must be in the

database.
post-condition: The PDE must be persistent
post-condition: The PDE is received by the

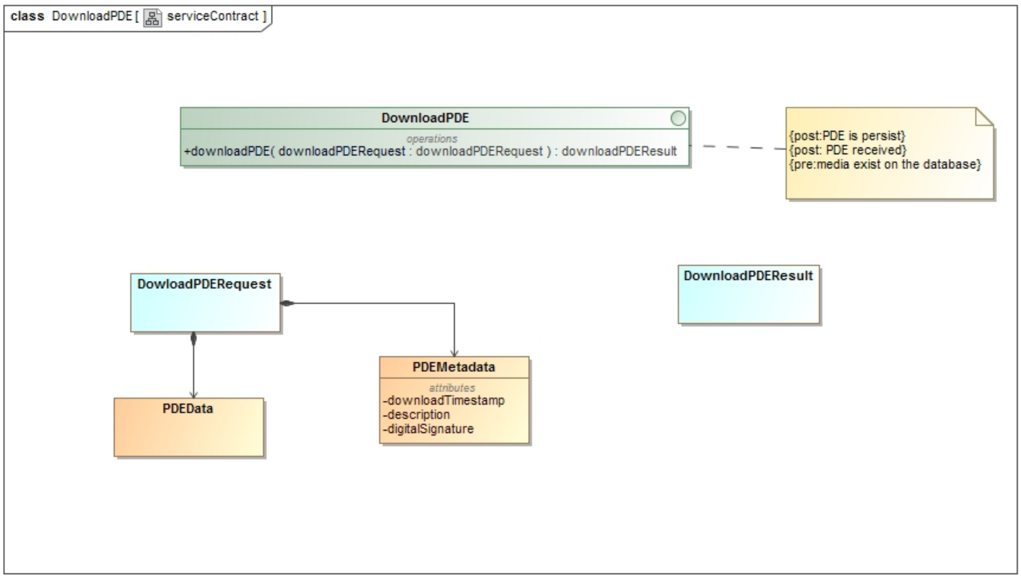

Figure 7: Process Specification: Downloading Potential Digital Evidence

3.6.1.2 Functional Requirement
The law enforcement agent needs to download the PDE to use it in the court of law.

3.6.1.2 ValidatePDE-Priority:high
3.6.1.2.1 Service Contract
post-condition: The PDE should be checked if it conforms to the standard set for valid evidence.
pre-condition: There has to be data to be validated
3.6.1.2.2 Functional Requirement
The system needs to validate that the data to ensure it's integrity.

3.6.1.3 EncryptPDE-Priority:high
3.6.1.3.1 Service Contract
post-condition: The PDE should be encrypted.
3.6.1.3.2 Functional Requirement
The system encrypts the PDE for it to be stored in the database.
3.6.1.4 DecryptPDE-Priority:high

3.6.1.4.1 Service Contract
post-condition: The PDE should be decrypted before it is used in the court of law.
3.6.1.4.2 Functional Requirement
The system decrypts the PDE for it to be viewed in the court of law.

3.6.1.5 ManageAccessAllocatio-Priority:high
3.6.1.5.1 Service Contract
post-condition: No unauthorised user can log in to the system.
3.6.1.5.2 Functional Requirement
The system is suppose to manage who has access to the system, this is a form of security measure.

3.6.2 Login and Administrative user

The system will authenticate against the South African Home Affairs Database for community members who are to upload potential digital evidence.

3.6.1.2 Use Cases
The Login module provides services to login, view the potential digital evidence to be uploaded to the ONW system.
3.6.1.2.1 Login-Priority:high
3.6.1.2.1.1 Service Contract
pre-condition: Only law enforcement agent with provided credentials can login.
pre-condition: Could connect to the SAPS database.
post-condition: UserID on results is populated by user's ID.

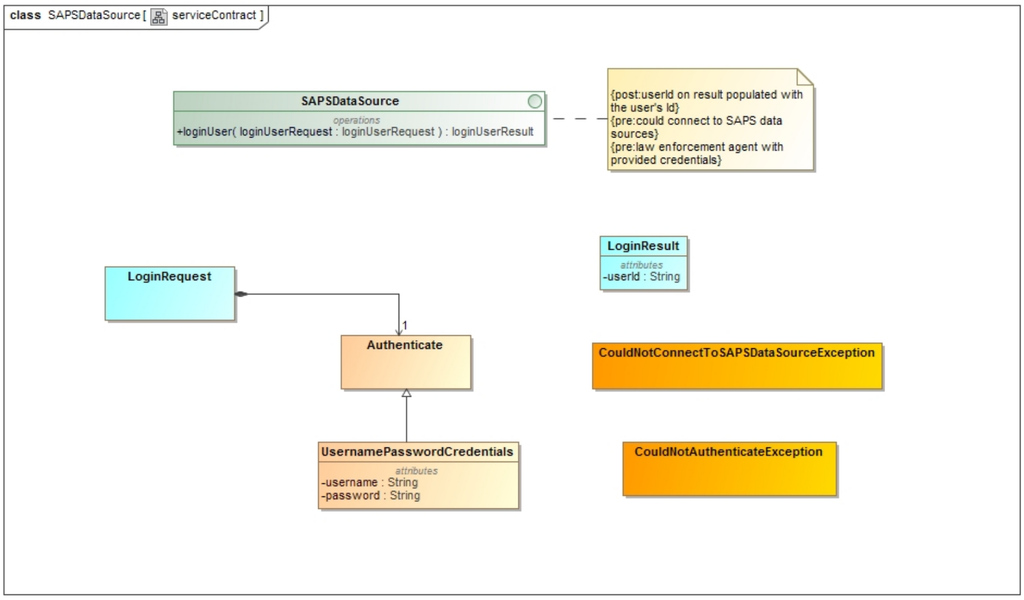
class SAPSDataSource [serviceContract]
SAPSDataSource
operations
+loginUser(loginUserRequest : loginUserRequest) : loginUserResult
{post:userId on result populated with the user's Id}
{pre:could connect to SAPS data sources}
{pre:law enforcement agent with provided credentials}
LoginResult
attributes
-userId : String
LoginRequest
1
Authenticate
CouldNotConnectToSAPSDataSourceException
UsernamePasswordCredentials
attributes
-username : String
-password : String
CouldNotAuthenticateException

Figure 8: Process Specification: Downloading Potential Digital Evidence

3.6.1.2.1.2 Functional Requirement
The law enforcement agent should be able to log in to the system to view the PDE.

3.6.1.2.2 ViewPDE-Priority:high
3.6.1.2.2.1 Service Contract

pre-condition: The PDE is in the database to be viewed.
post-condition: The law enforcement agent can view the PDE.
3.6.1.2.2.2 Functional Requirement
The law enforcement agents should be able to view what the witness has uploaded.

3.6.3 ONW Application Module

The Application module enables a user to upload potential evidence to the system
3.6.1.3 Use Cases
The Application module will provide services to capture PDE, confirm upload and send back a confirmation status of the upload to the uploader.

3.6.1.3.1 CapturePDE-Priority:high

3.6.1.3.1.1 Service Contract

post-condition: The PDE is captured and added to the database.

3.6.1.3.1.2 Functional Requirement

A user should be able to capture media in any situation they feel that a crime has been committed.

3.6.1.3.2 UploadPDE-Priority:high

3.6.1.3.2.1 Service Contract

pre-condition: There is enough space to add the new data.
pre-condition: Could connect to the SAPS database.
post-condition: The PDE is added to the database.

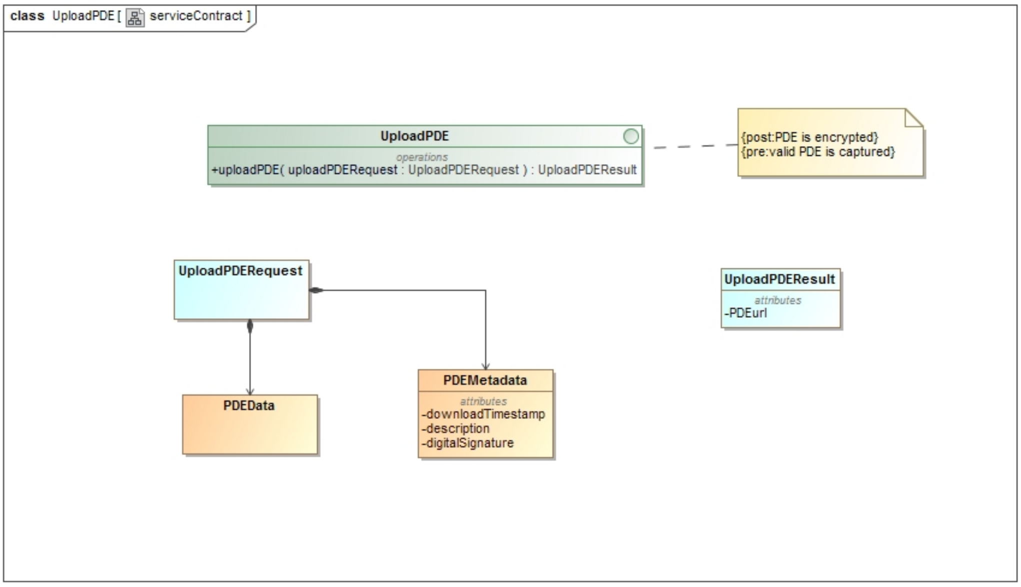

Figure 9: Process Specification: Downloading Potential Digital Evidence

3.6.1.3.2.2 Functional Requirement

A user needs to be able to upload a picture, video or audio.

3.6.1.3.3 ConfirmPDE-Priority:high

3.6.1.3.3.1 Service Contract

pre-condition: The user must have uploaded something to be confirmed.
post-condition: The user gets a confirmation from the system.

3.6.1.3.3.2 Functional Requirement

The user waits to receive a confirmation from the system telling them if their

PDE was accepted or rejected.