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

[Discuss the scope of the document and how it accomplishes its purpose.]

1.4   References

[List references and controlling documents, including: meeting summaries, white papers, other deliverables, etc.]

1.5   Assumptions and Constraints

[Provide a list of contractual or task level assumptions and/or constraints that are preconditions to preparation of the FRD. Assumptions are future situations beyond the control of the project, whose outcomes influence the success of a project.]

1.5.1   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.

1.5.2   Constraints

Constraints are boundary conditions on how the system must be designed and constructed. Examples include: legal requirements, technical standards, strategic decisions. Constraints exist because of real business conditions. For example, a delivery date is a constraint only if there are real business consequences that will happen as a result of not meeting the date. If failing to have the subject application operational by the specified date places the organization in legal default, the date is a constraint. Preferences are arbitrary. For example, a date chosen arbitrarily is a preference. Preferences, if included in the FRD, should be noted as such.]

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.]
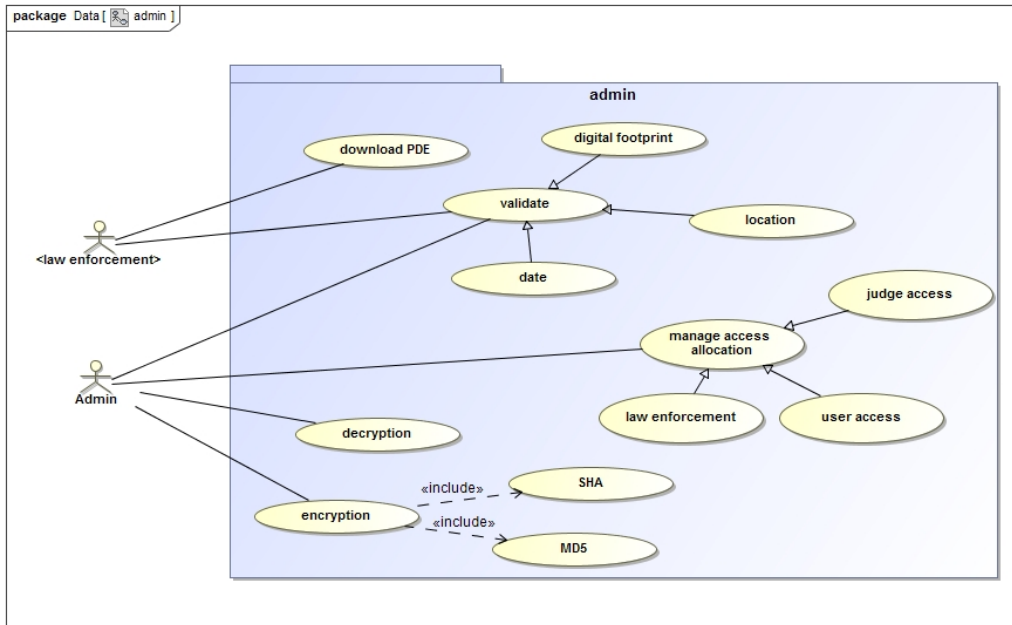
3.3 Use Case Diagrams

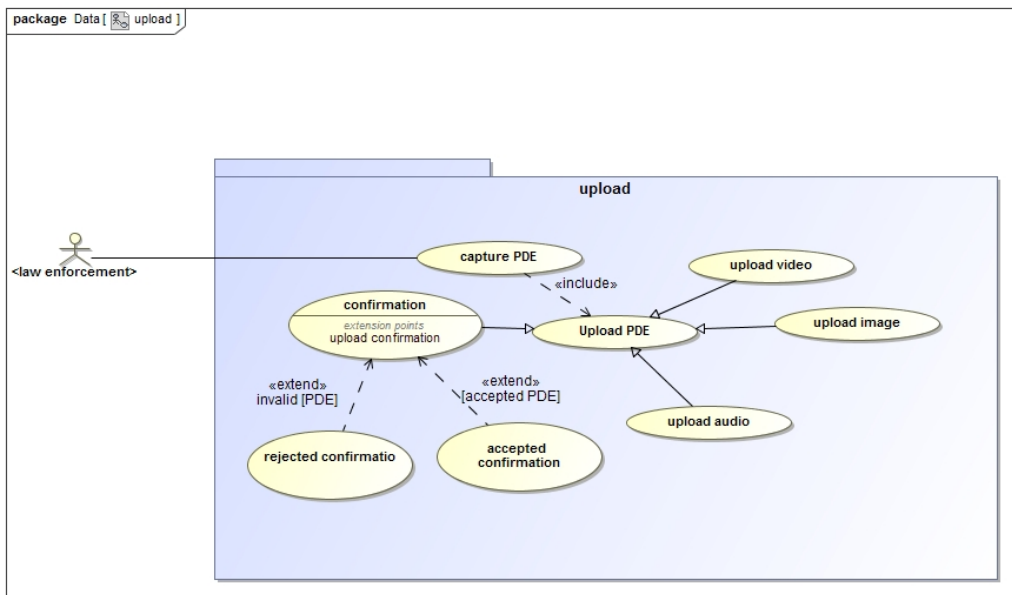Figure 1: Functional Requirements: Admin/Download Subsystem

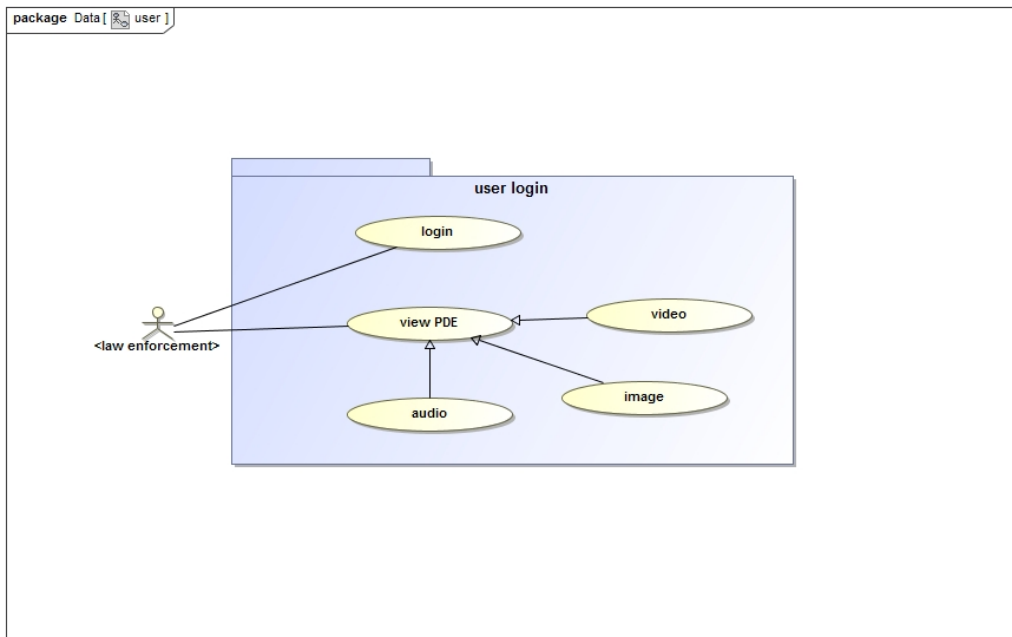Figure 2: Functional Requirements: Upload Subsystem

4

Figure 3: Functional Requirements: View Subsystem

3.4 Class And Sequence Diagrams

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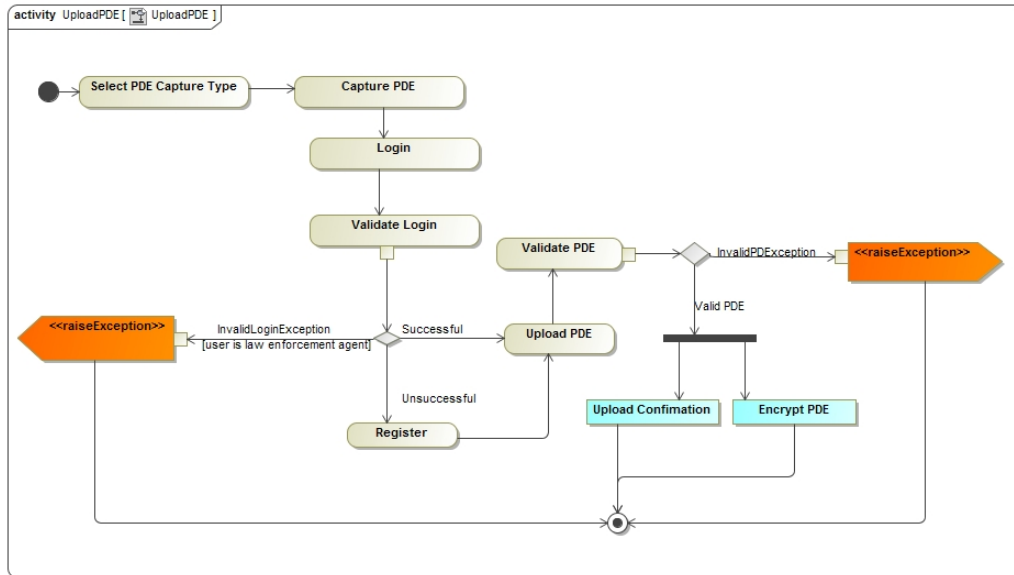
3.5 Activity Diagrams

Figure 4: Process Specification: Uploading Potential Digital Evidence
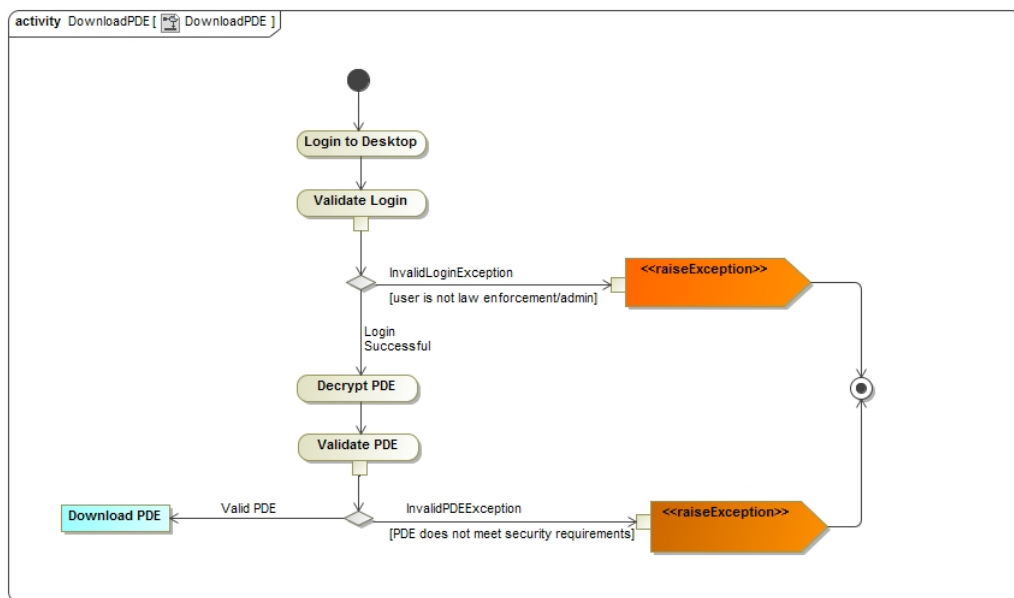

Figure 5: Process Specification: Downloading Potential Digital Evidence
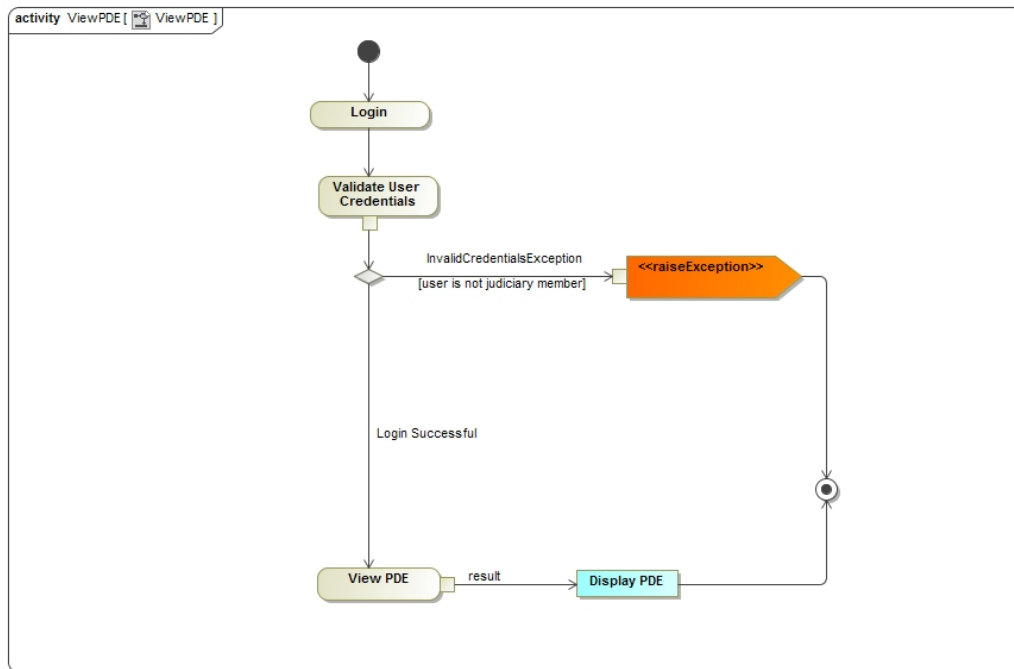
Figure 6: Process Specification: Viewing Potential Digital Evidence

3.6   Functional Requirements

[List the functional requirements of the system.]