
SOFTWARE REQUIREMENTS SPECIFICATION

Prison Management System

Prepared by
Amir k16-3641
Hassan k16-3617
Musawir k16-3621
Muteeb k16-3626
Nabeel k16-3627
Ubaid k16-3631

NUCES,FAST

April 8, 2019

Contents

1	Introduction	1
1.1	Purpose	1
1.2	Document Conventions	1
1.3	Intended Audience and Reading Suggestions	1
1.4	Project Scope	1
1.5	References	1
2	Overall Description	2
2.1	Product Perspective	2
2.2	Product Functions	2
2.3	User Classes and Characteristics	2
2.4	Operating Environment	2
2.5	Design and Implementation Constraints	2
2.6	User Documentation	3
2.7	Assumptions and Dependencies	3
3	External Interface Requirements	4
3.1	User Interfaces	4
3.2	Hardware Interfaces	4
3.3	Software Interfaces	4
3.4	Communications Interfaces	4
4	System Features	5
4.1	Detect Unwanted Persons	5
4.2	Prisoner Records	5
4.3	Face Recognition	5
5	Other Nonfunctional Requirements	6
5.1	Performance Requirements	6
5.2	Security Requirements	6
5.3	Software Quality Attributes	6
6	Architectural Design	7
7	ERD	8

8	Testing	9
8.1	Development Testing	9
8.1.1	Unit	9
8.1.2	Component	9
8.1.3	System	9

1 Introduction

1.1 Purpose

This software is a prison management tool. It will be used by the management of a prison to assist in the control and monitoring of the prisoners and increase the security

1.2 Document Conventions

This document uses Arial font at size 11pt. The sections are marked using headings and numbers.

1.3 Intended Audience and Reading Suggestions

The readers of this document include the government personnels and the prison management. This document will also help make the features certain and definite before the developing of the software begins.

1.4 Project Scope

This software aims to assist the management of a prison. It includes several important features bundled into a single software such as video surveillance and prisoner records.

1.5 References

<https://en.wikipedia.org/wiki/ONVIF>

https://pypi.org/project/face_recognition/

2 Overall Description

2.1 Product Perspective

This is a new product concept that will help the government body (prison management) to keep the prison record digital and secure. The project can be used by different prisons are required setup according to specific needs (like which areas are off limits etc)

2.2 Product Functions

The product offers 3 major functionalities which can further be divided into smaller functionalities.

- Detect unwanted persons
- Maintain prisoner record
- Face recognition based alert system

2.3 User Classes and Characteristics

The usage of the software can be divided into two major user classes.

- Prison security staff
- Prison supervisor
- Prison warden

2.4 Operating Environment

The software will be used by the prison management inside the prison itself. It will run on prison computers and will be connected to the surveillance network.

2.5 Design and Implementation Constraints

As this software will be used in a prison, security is highly important. The software must use encrypted login methods to allow only authorized users to access the video feed and the prisoner databases. Protection from other possibly malicious software is also a concern and must be taken into account.

2.6 User Documentation

The software will be bundled with a user manual detailing the use of the software features. As this product is intended for the prison management to use therefore the language in the manual will be used accordingly.

2.7 Assumptions and Dependencies

The software uses a third party face recognition system and is dependent on it. Similarly, it is also dependent on the camera hardware that will be used.

3 External Interface Requirements

3.1 User Interfaces

The user interface will be a desktop app developed in python. The interface will be easy to use and user friendly. There will be a user interface for adding and viewing the prisoner records. There will be another interface for live video monitoring.

3.2 Hardware Interfaces

Our software will interact with the CCTV surveillance system of the prison. The software will be connected to an existing CCTV monitoring system and will only use the footage of the surveillance system.

3.3 Software Interfaces

Database: MySQL / Mongo

Operating System: Windows / Linux

Tools: Python

Libraries: PyQT

3.4 Communications Interfaces

The software will use the ONVIF standard for communicating video surveillance footage as that footage can compromise the entire prison if it is stolen or misused. Internally the communication between various aspects of the system must be over secure channels.

4 System Features

4.1 Detect Unwanted Persons

The software will use live video surveillance to assist the guards. It will detect the movement in areas where there should be no movement for example the restricted areas or near the boundary walls of the prison. If movement is detected then it will notify and alert the guards.

4.2 Prisoner Records

Database and interface to maintain a record of the prisoners including their remaining sentence and the cell they have been allotted. These records will be used to assign security guards and other staff to the prison.

4.3 Face Recognition

Recognize if criminal or officer is accessing the area. The faces of the persons will be stored in the database before hand and then in critical areas there will be cameras installed that will use face recognition to identify if a certain person should be in that area or not.

5 Other Nonfunctional Requirements

5.1 Performance Requirements

Record maintenance in efficient way and maintain data integrity. The live face detection must be accurate. The video surveillance must not be delayed as this is a matter of security.

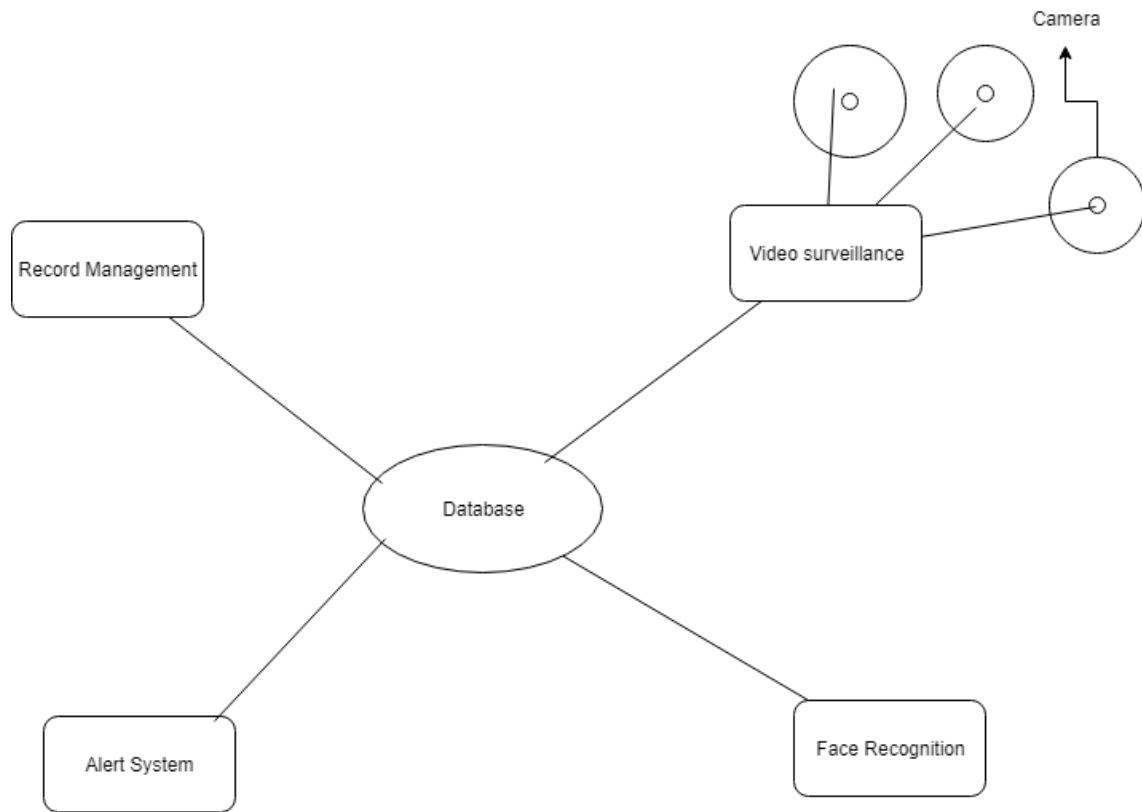
5.2 Security Requirements

The login mechanism being used for the system must use encrypted credentials and the transmission must be over secure channels. Similarly the video surveillance footage must also be over a secure and reliable channel

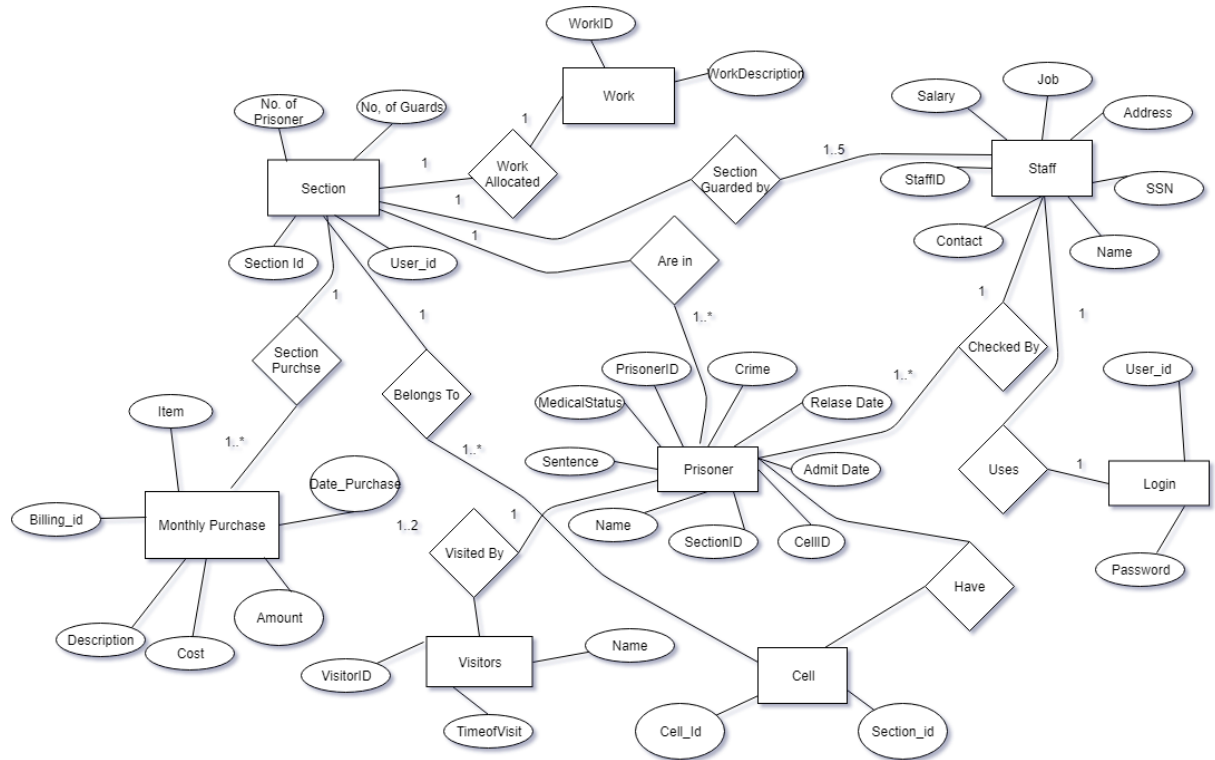
5.3 Software Quality Attributes

Our software can be used with any prison after the specifics have been defined. The software must also be ready for any expansions and changes the prison goes through such as increasing the number of cameras or expanding than number of prison cells.

6 Architectural Design



7 ERD



8 Testing

8.1 Development Testing

8.1.1 Unit

Unit testing will be done using each function output. We will rigorously test each unit.

8.1.2 Component

In component testing we will test the various components including the face recognition component, video surveillance component and record management component. Face recognition will be tested on pre-defined faces stored in the database. The video surveillance component will be stress testing using multiple cameras connected to the system. The record management component will be tested by entering and reading the data and also by inputting edge case entries.

8.1.3 System

In system testing we will test the system as a whole where all components are integrated together which will ensure integrity and consistency of the system.