VIDEO SURVEILLANCE

SAYUJ A.P

Reg No: MES20MCA-2046

Product Owner: Prof. Hyderali K

TABLE OF CONTENTS

INTRODUCTION

This is an innovative approach to video surveillance software project. We normally find video cameras in museum and other organization that continuously record and save the recorded video footage for days or months. This utilizes a lot of battery life and storage capacity to store these large video footage. Well this video surveillance software is an enhanced version of organization security that continuously monitors but only records unusual changes in the organization. This project is implemented in a Museum, so we can improve the security of objects collected in it. As soon as the system catches any unusual activity it takes steps and informs the admin by: Sending an notification to the admin about an unusual activity and Sending an image of the activity to the admin so that he may check the problem seriousness and react accordingly.

The main advantage of the system is that it instantly alert about any suspicious activity at the place, and requires much less or no storage space as compared to the traditional surveillance system.

MODULES

- ADMIN
  - Login
  - Showcase Add
  - Items Add
  - Manage Security
  - View Alerts
  - View complaints
  - Post Reply

MODULES

- USER
  - Register
  - Login
  - View Items
  - View Alerts
  - Post Complaint
  - View Reply

MODULES

- SECURITY (Technical)
  - Start Camera
  - Capture Frame
  - Process Image
  - Detect Intruder
  - Generate Alerts

Methodology

The entire project is divided into two parts :

1. Web part
2. Technical part

In the web part it consist of a web application.

In the technical part its an Desktop application works on Open CV library.

Video Surveillance system targets to detect suspicious behavior in a specific area using different motion detection techniques. The system is based on the open source image-processing library (Open CV). Once suspicious event is detected, the Video surveillance system will instantaneously send user a notification alert indicates motion detection. Recording only of suspicious events will be an option in order to reduce the required storage capacity.

Open CV:**-** is a open source library of programming functions for image processing and performing computer vision tasks.

Background Subtraction Algorithm:

The usual assumption is that the images of the scene without the intruding objects exhibit some regular behavior that could be well described by a statistical model. If we have a statistical model of the scene, an intruding object can be detected by spotting the parts of the image that don't fit the model. This process is usually known as ***"background subtraction***".

The idea behind motion detection techniques is to first build a background model from a sequence of images in order to find the objects of interest from the difference between that background estimation and the current frame. Therefore, the accuracy of the segmentation process depends on how well the background is modeled. This simple approach will compare between the current frame and the background (static first frame as assumed), then we will compute the absolute difference; which is defined as **Delta**. The difference between the current frame and background frame(first frame); is computed as the following equation:

$$\mathbf{\textit{Delta}} = |\,\mathbf{\textit{Current frame - Background frame}}\,|$$

Finally, Applying threshold of difference (Delta) will identify whether there is motion detected or not.

Steps involving Background Subtraction:

1. Collect the images.

2. Pre-process the image (converted into binary code).

3. Overlays the images.

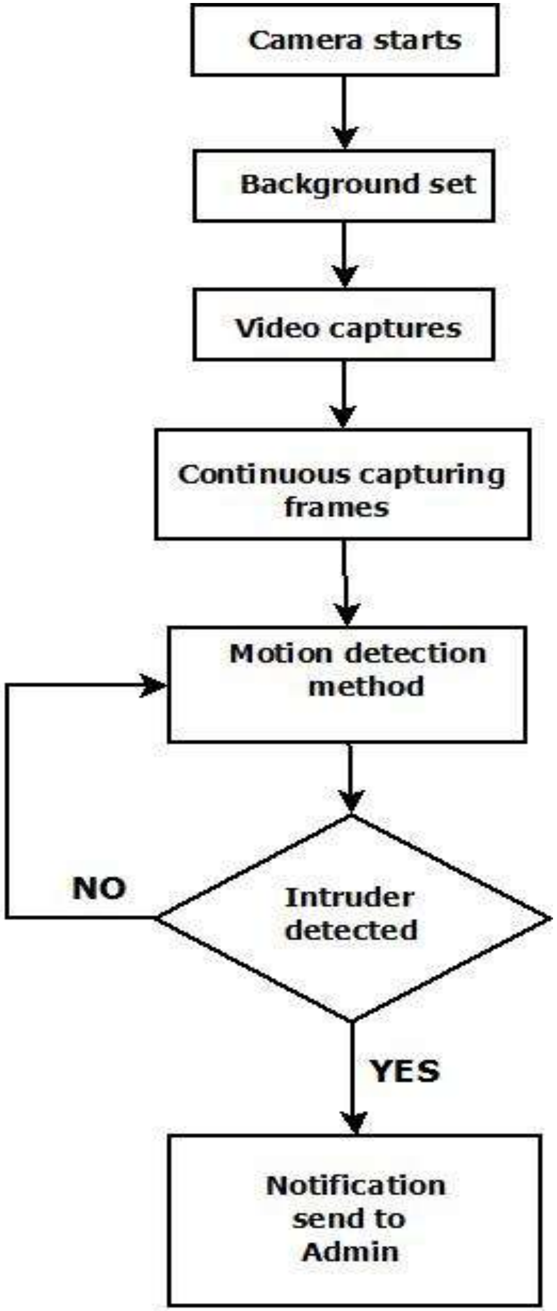
4. Subtract the pixel values.

5. Obtain **Delta** (Difference between Current frame and Background frame).

6. If Delta > Threshold.

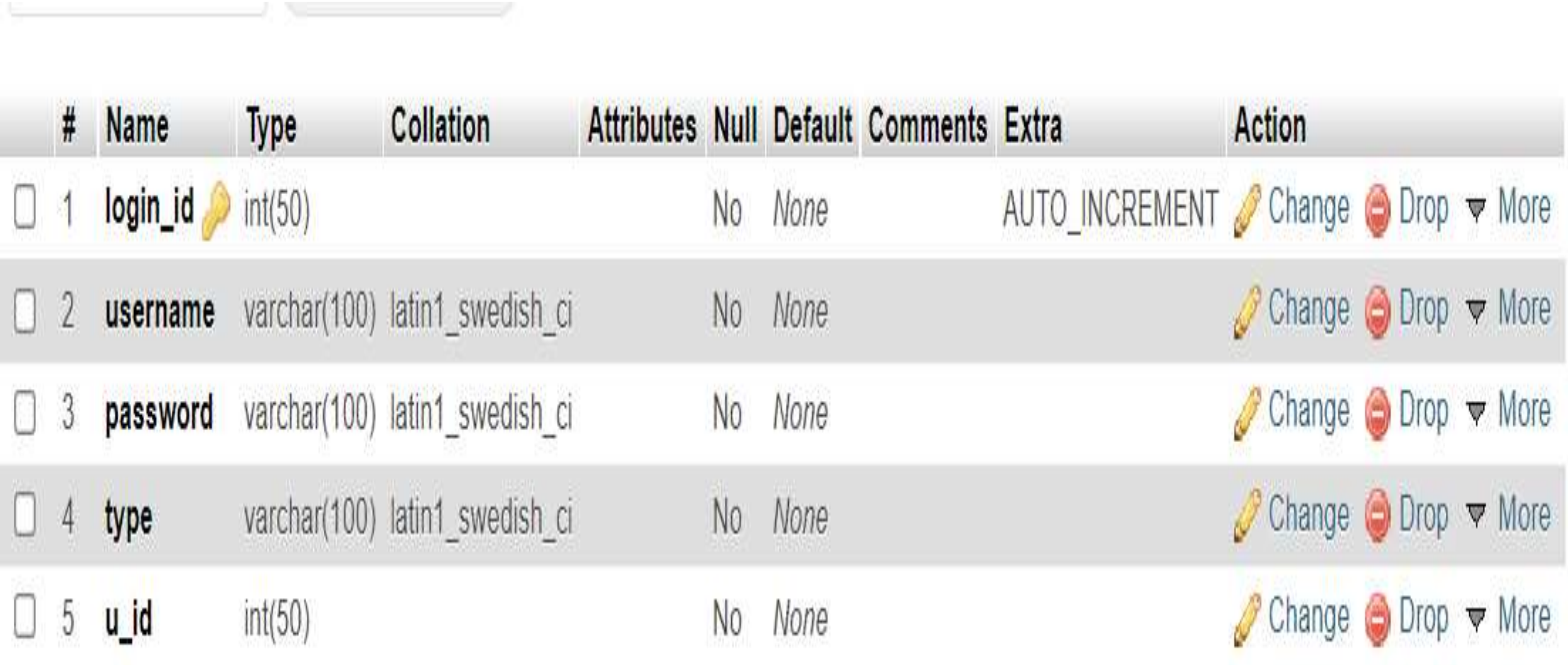
7. Generate alerts.

The core functions of this project is shown in the following flowchart :


Camera starts
Background set
Video captures
Continuous capturing frames
Motion detection method
Intruder detected
NO
YES
Notification send to Admin

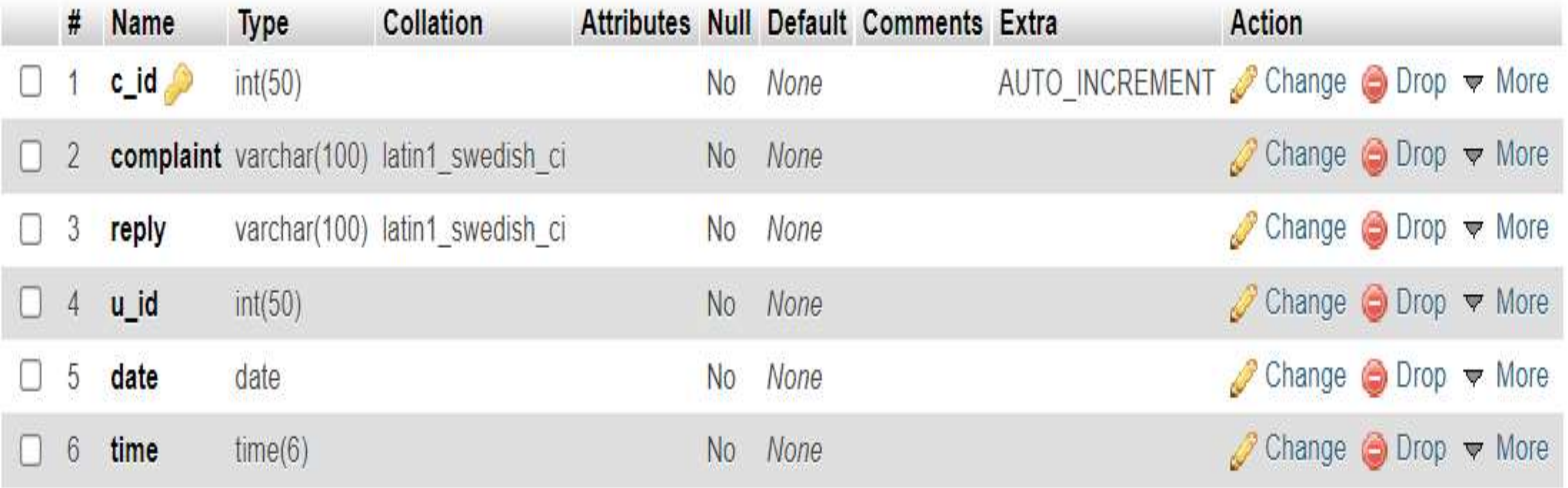
Tables

Login



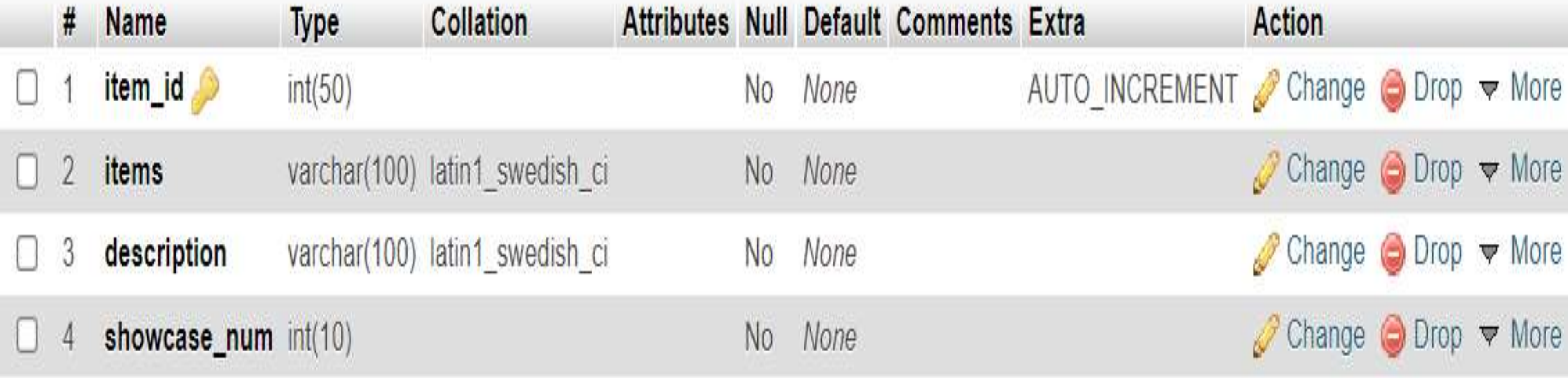
	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	login_id 🔑	int(50)			No	*None*		AUTO_INCREMENT	Change Drop ▼ More
□	2	username	varchar(100)	latin1_swedish_ci		No	*None*			Change Drop ▼ More
□	3	password	varchar(100)	latin1_swedish_ci		No	*None*			Change Drop ▼ More
□	4	type	varchar(100)	latin1_swedish_ci		No	*None*			Change Drop ▼ More
□	5	u_id	int(50)			No	*None*			Change Drop ▼ More

Complaint



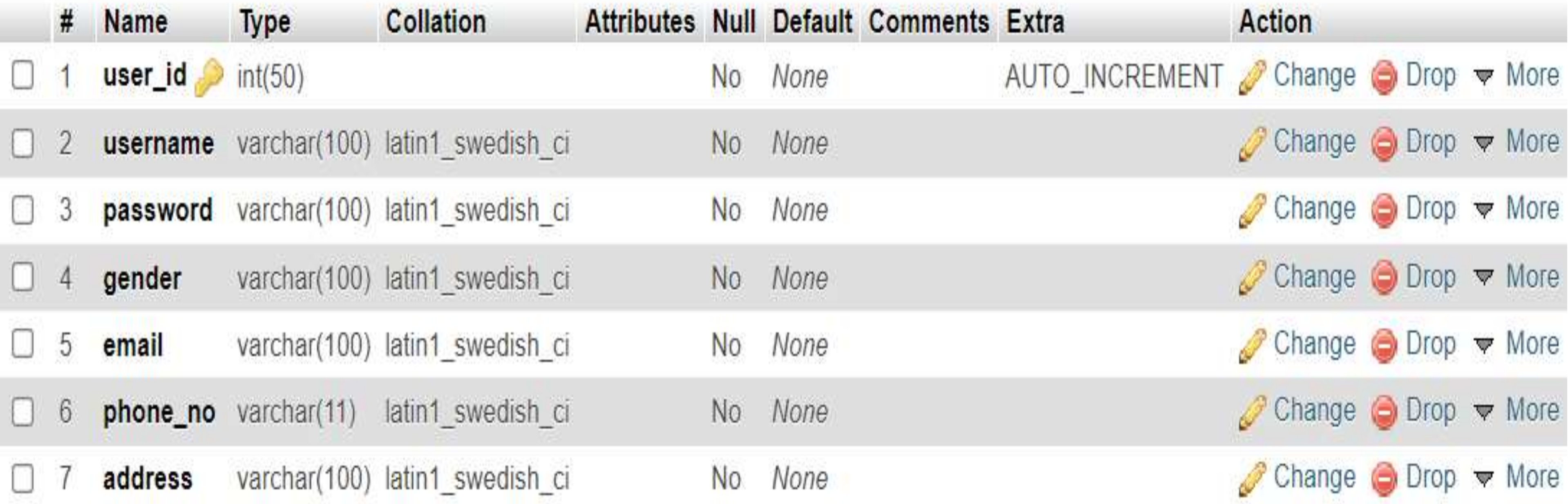
	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	**c_id** 🔑	int(50)			No	*None*		AUTO_INCREMENT	Change Drop ▼ More
□	2	**complaint**	varchar(100)	latin1_swedish_ci		No	*None*			Change Drop ▼ More
□	3	**reply**	varchar(100)	latin1_swedish_ci		No	*None*			Change Drop ▼ More
□	4	**u_id**	int(50)			No	*None*			Change Drop ▼ More
□	5	**date**	date			No	*None*			Change Drop ▼ More
□	6	**time**	time(6)			No	*None*			Change Drop ▼ More

Items



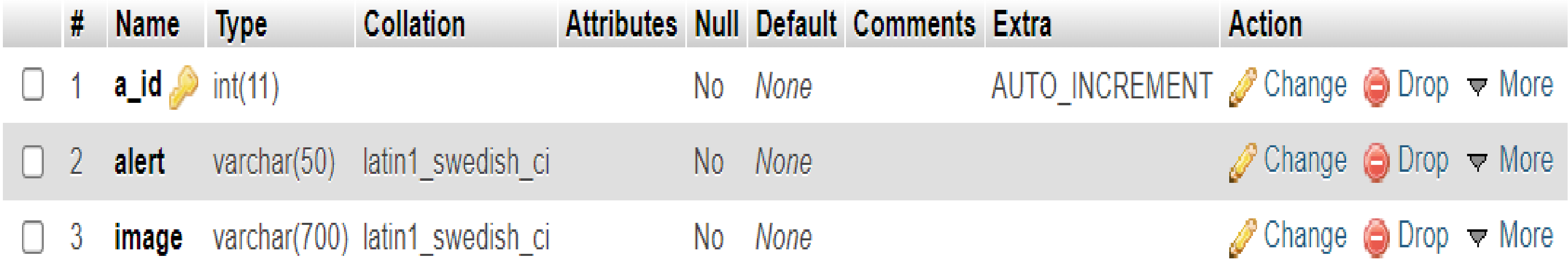
	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	**item_id** 🔑	int(50)			No	*None*		AUTO_INCREMENT	Change Drop ▼ More
□	2	**items**	varchar(100)	latin1_swedish_ci		No	*None*			Change Drop ▼ More
□	3	**description**	varchar(100)	latin1_swedish_ci		No	*None*			Change Drop ▼ More
□	4	**showcase_num**	int(10)			No	*None*			Change Drop ▼ More

User



	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	**user_id**	int(50)			No	*None*		AUTO_INCREMENT	Change Drop ▾ More
□	2	**username**	varchar(100)	latin1_swedish_ci		No	*None*			Change Drop ▾ More
□	3	**password**	varchar(100)	latin1_swedish_ci		No	*None*			Change Drop ▾ More
□	4	**gender**	varchar(100)	latin1_swedish_ci		No	*None*			Change Drop ▾ More
□	5	**email**	varchar(100)	latin1_swedish_ci		No	*None*			Change Drop ▾ More
□	6	**phone_no**	varchar(11)	latin1_swedish_ci		No	*None*			Change Drop ▾ More
□	7	**address**	varchar(100)	latin1_swedish_ci		No	*None*			Change Drop ▾ More

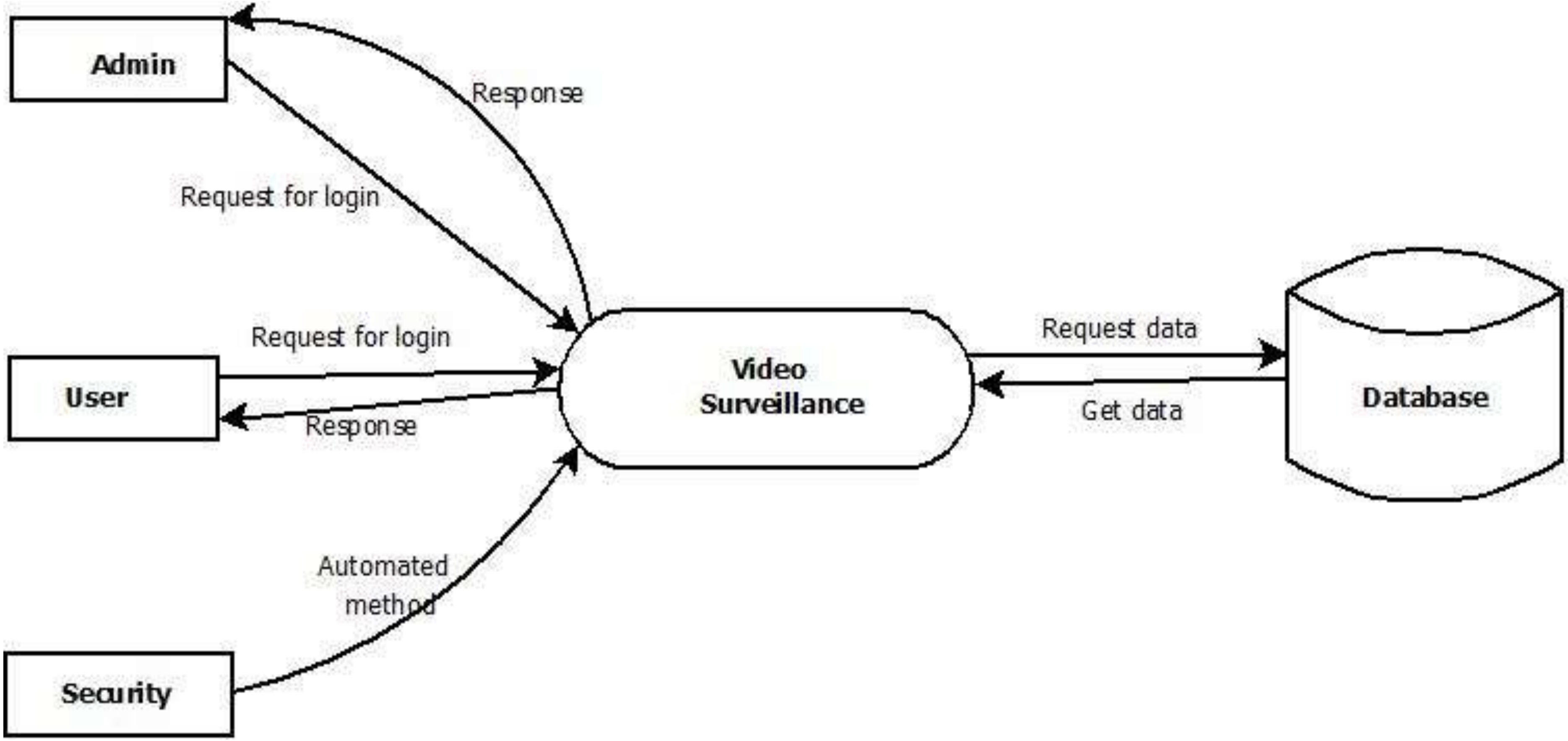
Alert



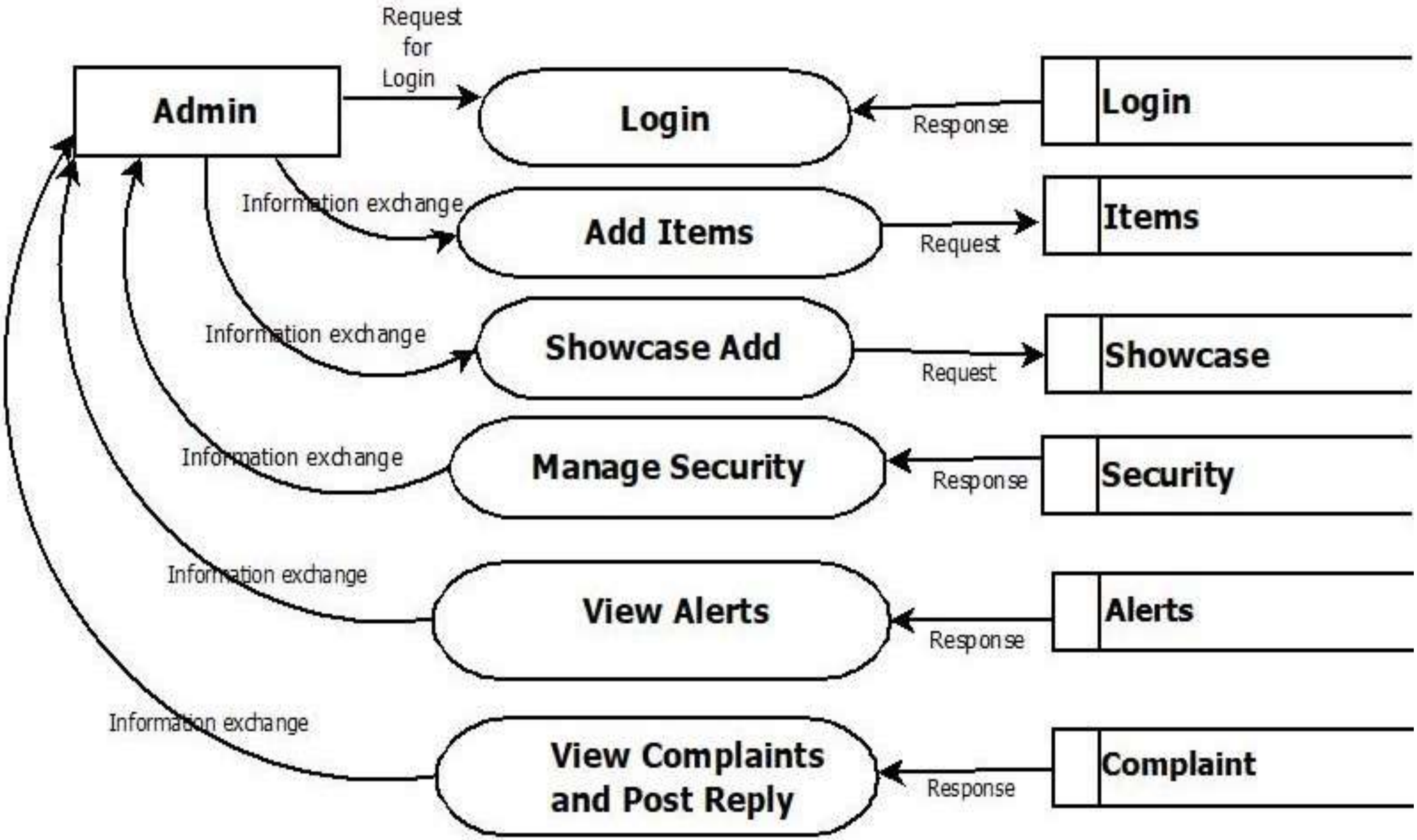
	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	**a_id**	int(11)			No	*None*		AUTO_INCREMENT	Change Drop ▾ More
□	2	**alert**	varchar(50)	latin1_swedish_ci		No	*None*			Change Drop ▾ More
□	3	**image**	varchar(700)	latin1_swedish_ci		No	*None*			Change Drop ▾ More

Data Flow Diagram

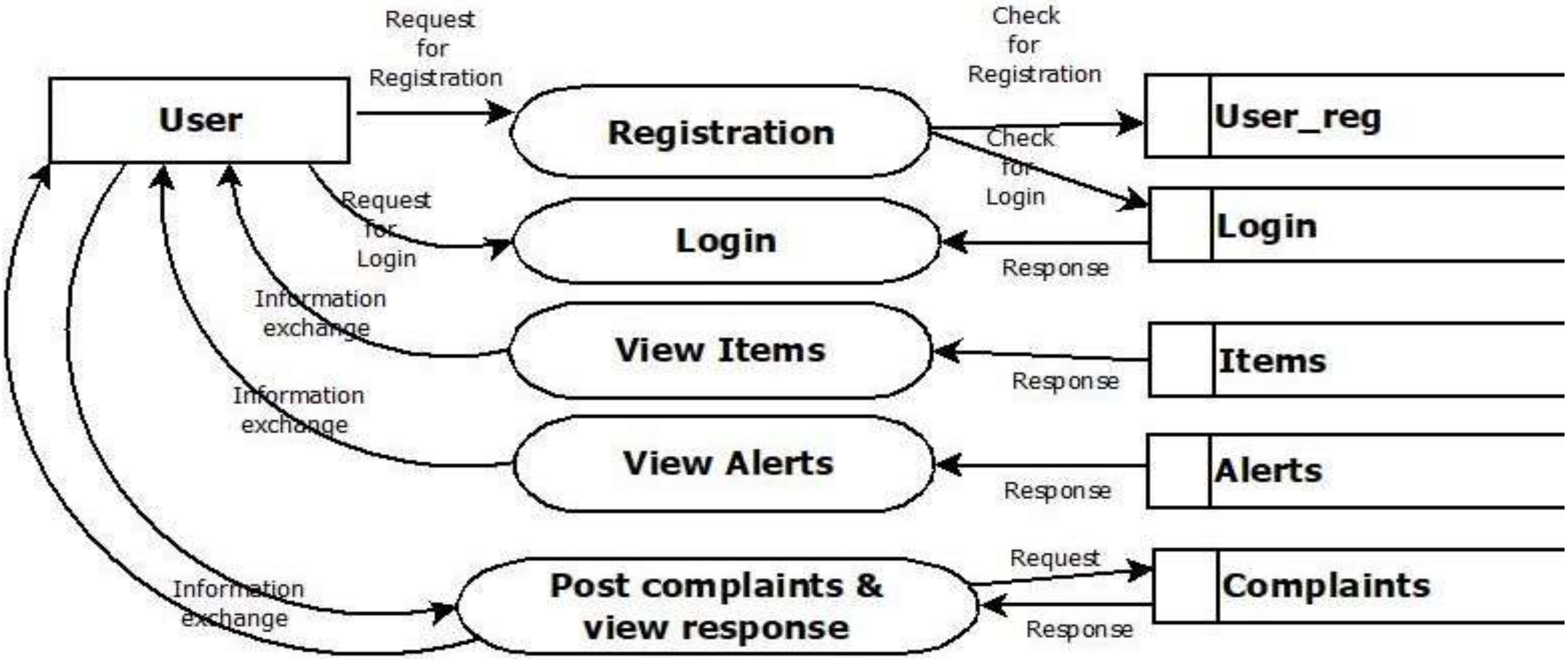
Level - 0



Level – 1 : Admin

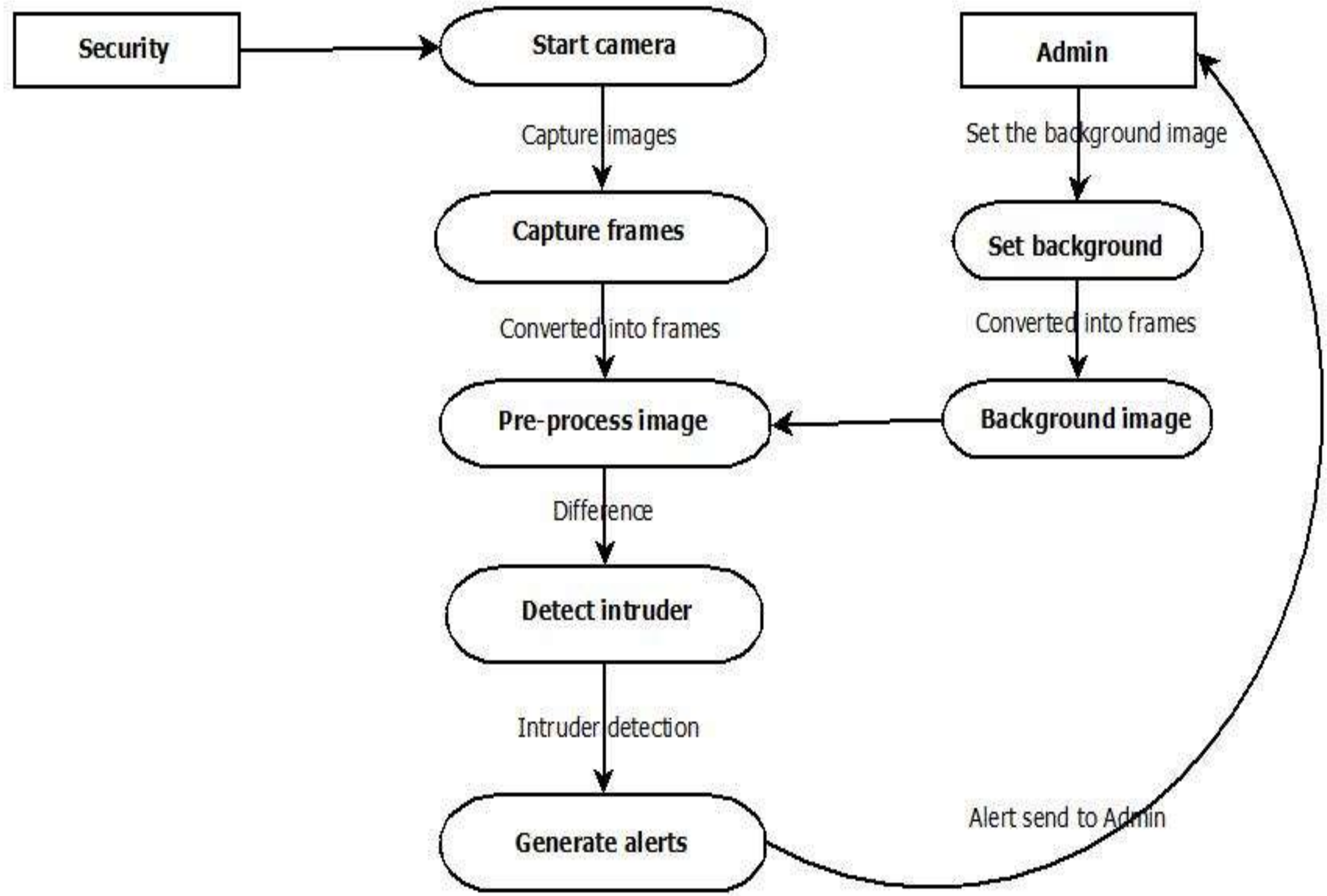


Level – 2 : User



User
Request for Registration
Registration
Check for Registration
User_reg
Check for Login
Login
Request for Login
Login
Response
Information exchange
View Items
Items
Response
Information exchange
View Alerts
Alerts
Response
Request
Complaints
Information exchange
Post complaints & view response
Response

Level – 3 : Security (Technical side)



DEVELOPING ENVIRONMENT

- Language: Python
- Front End: Html, CSS, Java script
- Back end: Python-django, Anaconda
- Database : MySQL
- IDE: PyCharm
- OS: Windows/Linux

PRODUCT BACKLOG

User story ID	Priority <High/Medium/Low>	Size (Hours)	Sprint <#>	Status <Planned/In progress/Completed>	Release Date	Release Goal
1	Medium	3	1	Completed	21-04-2022	Home page of admin
2	High	4		Completed	26-06-2022	Add showcase details
3	Medium	3	2	Completed	29-04-2022	Add items to the showcase
4	High	4		Completed	30-04-2022	User registration
5	Medium	2		Completed	03-05-2022	Login to the system
6	Medium	3		Completed	11-05-2022	View showcase
7	Medium	2		Completed	23-05-2022	View items
8	Medium	3	3	Completed	27-05-2022	View response
9	Medium	2		Completed	31-05-2022	Post Complaints
10	Medium	4		Completed	03-06-2022	View complaints and post response

User story ID	Priority <High/Medium/Low>	Size (Hours)	Sprint <#>	Status <Planned/In progress/Completed>	Release Date	Release Goal
11	Medium	3	4	Completed	17-06-2022	Provides security
12	Medium	4		Completed	21-06-2022	Capture the video
13	High	2		Completed	22-06-2022	Video converted into frames
14	High	3		Completed	27-06-2022	Processing images
15	High	3		Completed	06-07-2022	Detect any intruder
16	High	3		Completed	06-07-2022	Alerts generate

USER STORY

User story ID	As a <type of user>	I want to <perform some task>	So that I can <achieve some goal>
1	Admin	Access home page and login	Home page of admin access the system
2	Admin	Adding showcase details	Admin can list the showcase details
3	Admin	Adding items to the showcase	Admin can add items to the showcase
4	User	Registration	Register to the system and create profile
5	User	Login and access home page	Access the system
6	User	View the showcase details	View showcase
7	User	View the items in the showcase	View the items listed in the showcase
8	User	Post complaints	User post complaints
9	Admin	View complaints	View complaints and post response

User story ID	As a <type of user>	I want to <perform some task>	So that I can <achieve some goal>
10	User	View response	View response
11	Admin	Manage security	Admin provides security
12	Security	Capture the videos	Capture the videos
13	Security	Videos converted into frames	Videos converted into frames
14	Security	Processing images	Processing images
15	Security	Detect any intruder	Detect any intruder
16	Security	Generate alerts	Generate alerts

PROJECT PLAN

User Story ID	Task Name	Start Date	End Date	Days	Status
1	Sprint 1	20-04-2022	21-04-2022	4	Completed
2		25-04-2022	26-06-2022		Completed
3	Sprint 2	29-04-2022	29-04-2022	12	Completed
4		30-04-2022	30-04-2022		Completed
5		02-05-2022	03-05-2022		Completed
6		09-05-2022	11-05-2022		Completed
7		20-05-2022	23-05-2022		Completed
8	Sprint 3	26-05-2022	27-05-2022	7	Completed
9		30-05-2022	31-05-2022		Completed
10		02-06-2022	03-06-2022		Completed

User Story ID	Task Name	Start Date	End Date	Days	Status
11	Sprint 4	17-06-2022	17-06-2022	6	Planned
12		20-06-2022	21-06-2022		Planned
13		22-06-2022	22-06-2022		Planned
14		27-06-2022	27-06-2022		Planned
15		06-07-2022	06-07-2022		Planned
16		06-07-2022	06-07-2022		Planned

SPRINT BACKLOG ACTUAL

Backlog Item	Status and Completion date	Original Estimate in hours	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10	Day 11	Day 12	Day 13	Day 14	Completed <Y/N>
User story #1,2			Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	
UI Designing	21-04-22	2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	Y
Database Connectivity	21-04-22	3	0	0	0	0	2	0	0	1	0	0	0	0	0	0	Y
Coding	26-06-22	2	0	0	0	0	1	0	0	0	1	0	0	0	0	0	Y
Testing	26-06-22	3	0	0	1	1	1	0	0	0	0	0	0	0	0	0	Y
User story #3,4,5,6,7																	
UI Designing	29-04-22	3	2	1	0	0	0	0	0	0	0	0	0	0	0	0	Y
Database Connectivity	03-05-22	3	0	0	0	1	1	0	1	0	0	0	0	0	0	0	Y
Coding	11-05-22	3	0	0	0	1	0	0	0	1	0	0	0	1	0	0	Y
Testing	23-05-22	3	0	0	2	0	0	0	0	0	0	0	0	0	0	0	Y

Backlog Item	**Status and Completion date**	**Original Estimate in hours**	**Day 1**	**Day 2**	**Day 3**	**Day 4**	**Day 5**	**Day 6**	**Day 7**	**Day 8**	**Day 9**	**Day 10**	**Day 11**	**Day 12**	**Day 13**	**Day 14**	**Completed <Y/N>**
User story #8,9,10			Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	
UI Designing	27-05-22	4	1	1	1	0	1	0	0	0	0	0	0	0	0	0	Y
Database Connectivity	27-05-22	4	0	0	0	0	0	1	1	0	0	0	0	0	1	1	Y
Coding	03-06-22	5	1	0	0	1	0	1	0	0	0	1	0	0	0	1	Y
Testing	03-06-22	5	0	0	1	0	0	0	2	2	0	0	0	1	0	0	Y
User story #11,12,13,14,15,16																	
UI Designing	17-06-22	2	1	0	0	0	1	0	0	0	0	0	0	0	0	0	Y
Database Connectivity	22-06-22	3	1	0	0	1	0	0	1	0	0	0	0	0	0	0	Y
Coding	06-07-22	2	0	1	0	1	1	0	0	0	0	0	0	0	0	0	Y
Testing	06-07-22	3	0	0	1	0	0	1	1	0	0	0	0	0	0	0	Y
Total		50	7	4	6	6	8	3	6	4	1	1	0	2	1	2	

SPRINT BACKLOG PLAN

Backlog Item	Status and Completion date	Original Estimate in hours	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10	Day 11	Day 12	Day 13	Day 14
User story #1,2			Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours
UI Designing	21-04-22	2	1	1	0	0	0	0	0	0	0	0	0	0	0	0
Database Connectivity	21-04-22	3	0	0	0	0	2	0	0	1	0	0	0	0	0	0
Testing	26-06-22	2	0	0	0	0	1	0	0	0	1	0	0	0	0	0
Coding	26-06-22	3	0	0	1	1	1	0	0	0	0	0	0	0	0	0
User story #3,4,5,6,7																
UI Designing	29-04-22	3	2	1	0	0	0	0	0	0	0	0	0	0	0	0
Database Connectivity	03-05-22	3	0	0	0	1	1	0	1	0	0	0	0	0	0	0
Coding	11-05-22	3	0	0	0	1	0	0	0	1	0	0	0	1	0	0
Testing	23-05-22	3	0	0	2	0	0	0	0	0	0	0	0	0	0	0

Backlog Item	Status and Completion date	Original Estimate in hours	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10	Day 11	Day 12	Day 13	Day 14
User story #8,9,10			Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours	Hours
UI Designing	27-05-22	4	1	1	1	0	1	0	0	0	0	0	0	0	0	0
Database Connectivity	27-05-22	4	0	0	0	0	0	1	1	0	0	0	0	0	1	1
Coding	03-06-22	5	1	0	0	1	0	1	0	0	0	1	0	0	0	1
Testing	03-06-22	5	0	0	1	0	0	0	2	2	0	0	0	1	0	0
User story #11,12,13,14,15,16																
UI Designing	17-06-22	2	1	0	0	0	1	0	0	0	0	0	0	0	0	0
Database Connectivity	22-06-22	3	1	0	0	1	0	0	1	0	0	0	0	0	0	0
Coding	06-07-22	2	0	1	0	1	1	0	0	0	0	0	0	0	0	0
Testing	06-07-22	3	0	0	1	0	0	1	1	0	0	0	0	0	0	0
Total		50	7	4	6	6	8	3	6	4	1	1	0	2	1	2

THANK YOU