Tshwane University of Technology

We empower people

Proposed By : Rakoma Tshegofatjo Peter
Student Number : 213159259
Subject Code : IDC30BT
Lecture : Masethe HD
Name of the Project: TUT Campus Incident Tracking System

IDC30BT Project

TUT Campus Incident Tracking System

Table of Contents

Phase 1 (Proposal) 3

Name of the project 4

Domain Analysis 4

- Explanation of the general field of business 4
- Terminology/glossary used 4
- The business environment 4
- Tasks and procedures currently performed 4
- Customers and users 5
- Competing software 5
- Similarities to other domains 5

Definition of the problem 5

- Opportunity 5

Scope definition 6

- Narrowed Scope 6
- Inputs 6
- Outputs 6
- Activities 6
- Outcomes 6
- Impact to the community 7

Vision and Objectives 7

- Vision 7
- Objectives 7

Users of the System 7

- Student 7
  - Roles 7
- Administrative Personnel 7
  - Roles 7
- Safety Officer 8
  - Roles 8

Mandatory Functions 8

- Register/Add 8
- Delete 8
- Update 8

Functional Requirements 8
Things that the system should do 8
Non-Functional requirements 10
Authentication 10
Availability 10
Use Case 11
Student Use Case Diagram 11
Safety Officer Use Case Diagram 12
Administrative Personnel Use Case Diagram 13
Tools and Technologies to be used 14
Phase 2 (Modelling with Classes) 15
Class Diagrams 16
Sequence Diagram 17
State Diagrams 20
Component Diagrams 24
Phase 3 (User Interface) 25
Design User Interfaces 25
Phase 4(Build Database and Demonstrate Integration) 33
Database Structure 33
Manage objects 34
Normalization Process 35
UNF 35
1NF 35
2NF 36
3NF 36
Manipulate your data 37
Manage transaction 39
Phase 5(Final Project Deliverable) 40
Documentation on Test cases and Test Plan 41
Reports 50
User manual 51

Phase 1 (Proposal)

Name of the project

- TUT Campus Incident Tracking System.

Domain Analysis

Explanation of the general field of business

- The field of business for which the proposed system is going to operate in is the field of Environmental Health and Safety at universities.

Terminology/glossary used

- Student- is the person who is registered TUT.
- Reporter- is a person who records an incident(s).
- Administrative personnel- encompass a variety of individuals who may have some responsibility for the activities that take place at the location.
- Health Hazard- is a disease, epidemic or infection or that has the potential of spreading to other individuals and cause harm to them or other health emergency.
- Safety officer- is the person responsible for reporting health hazards and incidents.
- Incident – may be a crime or Health hazard (E.g. If someone is injured or ill ; someone stealing or damaging something)
- Crime - is a criminal offence.
- Status- is the details about the progress of the reported incident.
- Criminal Offenses – may refer to Criminal Homicide, including: a) Murder and Non-negligent Manslaughter, b) Negligent Manslaughter; Sex Offenses including: a) Forcible b) Non-forcible; Robbery; Aggravated Assault; Burglary; Motor Vehicle Theft; and Arson.

The business environment

- Environmental Health and safety is responsible for dispatching communications and incident reporting which helps in campus public safety.

Tasks and procedures currently performed

- The safety officer/administrative personnel records the information relating to an incident at the scene and takes it for capturing, where it is recorded in a form.
- The Student who is a victim of the incident or witness goes to the Campus Security or Health office to report an incident, where he/she has to recall the information and report it.
- All incidents are tracked and reported using a file system.
- Reports about incidents are produced manually.
- Information about incidents is dispersed using fliers, loudspeakers or notice boards.
- Notices about incident progress/investigation is communicated to the Student via a phone call or letter (posted).
- Students also inquire at the Campus Security office about the status of the reported incident.

Customers and users

- Students.
- Administrative Personnel.
- Safety Officers.
- System Administrator.

Competing software

- Mobile Crowdsourcing System for Campus Safety.
- CROWDSAFE system.

Similarities to other domains

- Internet based crime reporting system.
- Allows people to report public safety issues.
- They are a powerful platform to support real-time searching and reporting public safety information.
- Enables both crowdsourcing of crime incidents and provides safe routing suggestions to avoid public safety issues.

Definition of the problem

- With the current system used at TUT there are significant costs the Student/Security Officer/Administrative personnel will consider, the time and effort to report an incident.
- Other potential costs include situations when the victim feels shame or embarrassment filing a sensitive safety report, or when the victim fears retaliation by the offender.
- Filling out handwritten reports can be a time-consuming endeavour that takes personnel away from their primary duty: patrolling the campus for incidents.
- Students/Security Officers/Administrative Personnel have a difficulty of recalling and retaining information relating to an incident which needs to be reported, thus resulting in inaccuracies in information about the incident being reported.
- The current system is also prone to fraudulent behaviour.
- The dissemination of information about incidents is not reliable and effective (not timely).
- Recent crimes occurring on college campus have significantly escalated due to the system currently used. Thus the need to focus our attention towards improving campus safety with an electronic, integrated incident reporting software system has been identified to address the above mentioned problem.

Opportunity

- The amount of time needed by Safety officers to process incident documents will be substantially decreased.
- Moreover, paperless reporting system will help to reduce operating expenses and support the “green” initiatives found on TUT campuses.

Scope definition

Narrowed Scope

- The system will have the ability to document and track reported incidents.
- The system will have the ability to track reported incidents.
- The system will be able to process reports about incidents.
- Keep track of incidents.
- Make more than one Safety Officer or Administrative Personnel responsible for maintenance of incidents.
- Include the nature of the crime, incident number, date/time reported, date/time occurred, general location and disposition of the reporter.

Inputs

- Student details.
- Safety Officer details.
- Administrative Personnel details
- Incident details.

Outputs

- Incident reports.
- Incident alerts.
- Notifications.
- Incident outcomes.

Activities

- Registration and Login.
- Students/Safety Officers/Administrative Personnel can report an incident.
- Students/Safety Officers/Administrative Personnel can view progress of reported incidents.
- System can generate reports when they are needed.
- Safety Officers/Administrative Personnel can send notification to Student about progress of reported incident.
- Safety Officers/Administrative Personnel can alert students about an incident.

Outcomes

- Reduced dependence on file system.
- Timely dissemination and receiving of information.
- Reduced costs and increased reporting of incidents.
- Reduced fraudulent behaviour.

Impact to the community

- It will improve the way in which Students/Safety Officers/Administrative personnel report incidents.
- It will reduce the time and effort to report an incident the Student/Security Officer/Administrative personnel.
- It will make the community feel much safer.
- There will be efficiency and convenience.

Vision and Objectives

Vision

- Is to develop an information technology system which will facilitate the process of reporting and tracking incidents at TUT, to support real-time searching and reporting public safety information.

Objectives

- To ensure that only authorised personnel can handle report incidents.
- To ensure that there a systematic way to track reported incidents and produce reports.
- To ensure that information about incidents is effectively disseminated.
- To meet information needs of Students, Safety Officers and Administrator Personnel by providing timely consistent information.
- To ensure that Student/Safety Officers/Administrative Personnel time and effort to report incidents is reduced.

Users of the System

Student

Roles

- Report an incident.
- View incident progress.
- View incident reports.

Administrative Personnel

Roles

- Register Safety Officers.
- Report incidents.
- Disseminate information about incidents.
- Insert, Update, delete, and incident details.
- Insert, Update, delete Student's and Safety Officer's details.

Safety Officer

Roles

- Report incidents.
- Disseminate incident information.
- Update incident details.

System Administrator

Roles

- Register Administrative personnel/Safety officer/Student.
- Delete Administrative personnel/Safety officer/Student.

Mandatory Functions

Register/Add

- The system should allow registration of Students and Safety Officers to the database.
- Add Incidents to the database.

Delete

- The system should allow deletion of Students, Incidents and Safety Officers data from the database.

Update

- The system should allow the update of Students, Incidents and Safety Officers data in the database.

Functional Requirements

Things that the system should do

- The system should show the students location (google maps) when the student navigates to the home page.
- The system should also show the location (google maps) when the user is not yet logged in, that is when the user navigates to the public home page.
- The Students should be able to register.
- Administrative personnel should be able to register Students.
- The system Administrator should be able to register Students.
- The Administrative personnel should be able to register Safety Officers.
- The system Administrator should be able to register Safety Officers.
- The system Administrator should be able to register Administrative.
- Registered Students, Administrative Personnel and Safety Officers should be able to report incidents.

- Registered Administrative Personnel and Safety Officers should be able to send notification about incidents to students.
- Registered Administrative Personnel should be able to search a reported incident and view it.
- Registered Administrative Personnel should be able to update an incident that was searched.
- Registered Administrative Personnel should be able to delete an incident that was searched.
- The system should allow registered Administrative Personnel to view all incident that were reported.
- Registered safety officers should be able to view incidents that were only reported to them.
- Registered students should be able to view only incidents that were reported by them.
- The system should allow registered student to search an incident that was reported by the respective student.
- Administrative Personnel should be able to update results (status) of reported incidents.
- When the Safety Officer/Administrative Personnel send incident notification to students, system should send e-mails to Students to alert them about the incident.
- Reports about incidents should be automatically generated and available for viewing.
- Information about how to categorise incidents when reporting them should be available on the home page.
- Information to students about how to safeguard themselves around the campus should be available on the home page.
- Registered Safety Officers and Administrative personnel should be able to confirm if an incident reported is closed or open.
- If the reported incident relates to a missing person, the system should provide the Student/Safety Officer to attach an image and provide details about the missing person.
- The should also inform automatically retrieve information about the missing persons contact persons details and send a notification to that person and disseminate information about the missing person until the incident status is changed to closed (displayed on the dashboard/home page.
- Information about how to respond to common incidents should be available on the home page.
- The system should automatically capture the geographic where the person is reporting incident.
- The system should create statistics about incidents and guide people about danger areas where incidents may occur.
- The system should have a panic button for students geographically in campus to report an incident if they are in danger for quick response, without having to provide detailed information (when they are logged in) for immediate response.
- The system should notify registered Safety Officers/Administrative about pending reported incidents.
- All users must be able to export reports in pdf format.
- Registered Students/Safety Officers/Administrative personnel should be able to send a notification (Contact Us Page) to communicate with System Administrator about issues relating the system.

- The system should display notification of queries reported by users on System Administrator's Dashboard (Home page).
- The System Administrator should be able to register Students/Safety Officers/Administrative personnel.

Non-Functional requirements

Authentication

- The system should allow only registered Students/Safety Officers to login to the system.
- Only authorised Administrative personnel and Safety Officers should be able to handle reported incidents.

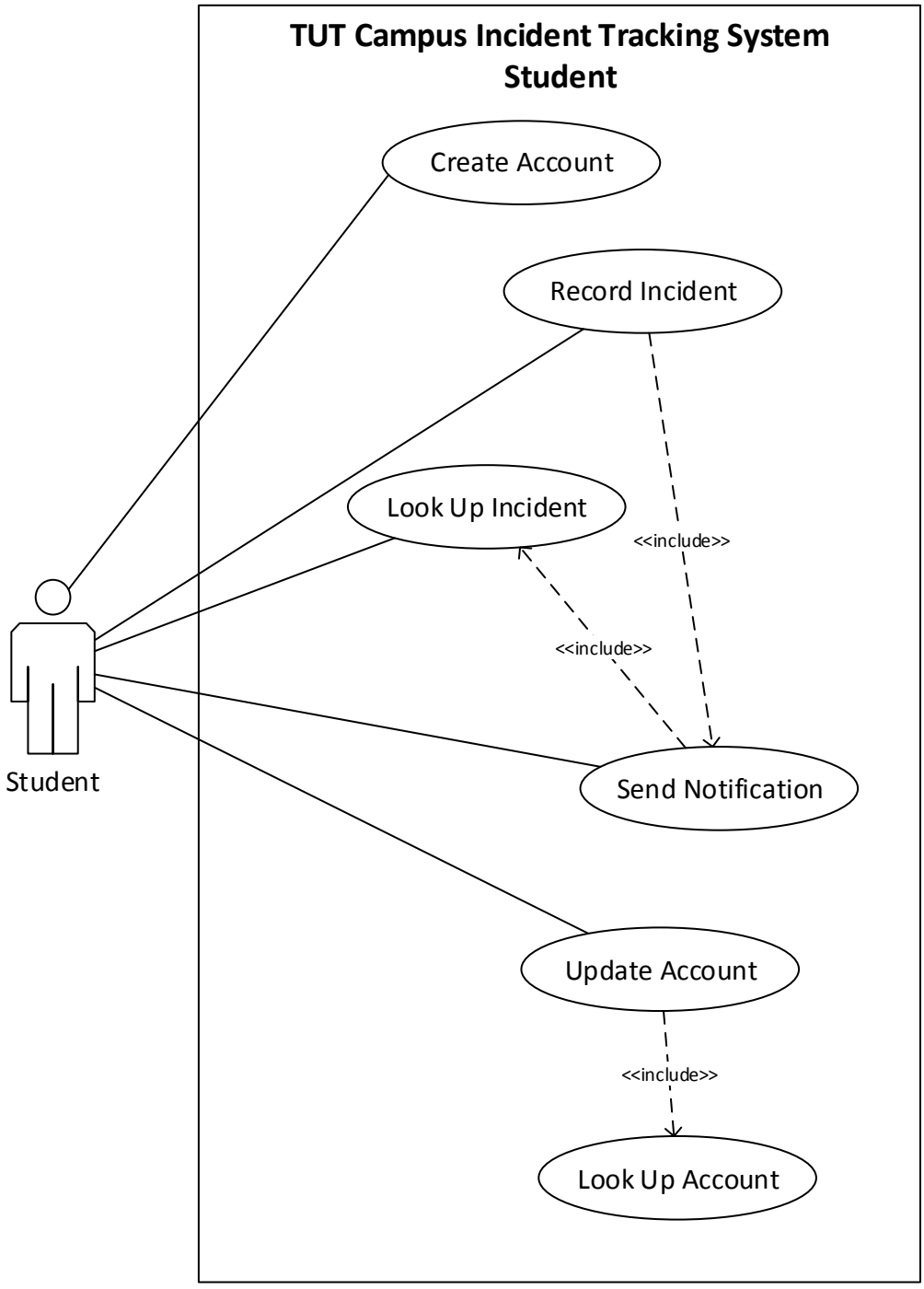
Availability

- The system should be available 24/7, 365 days.

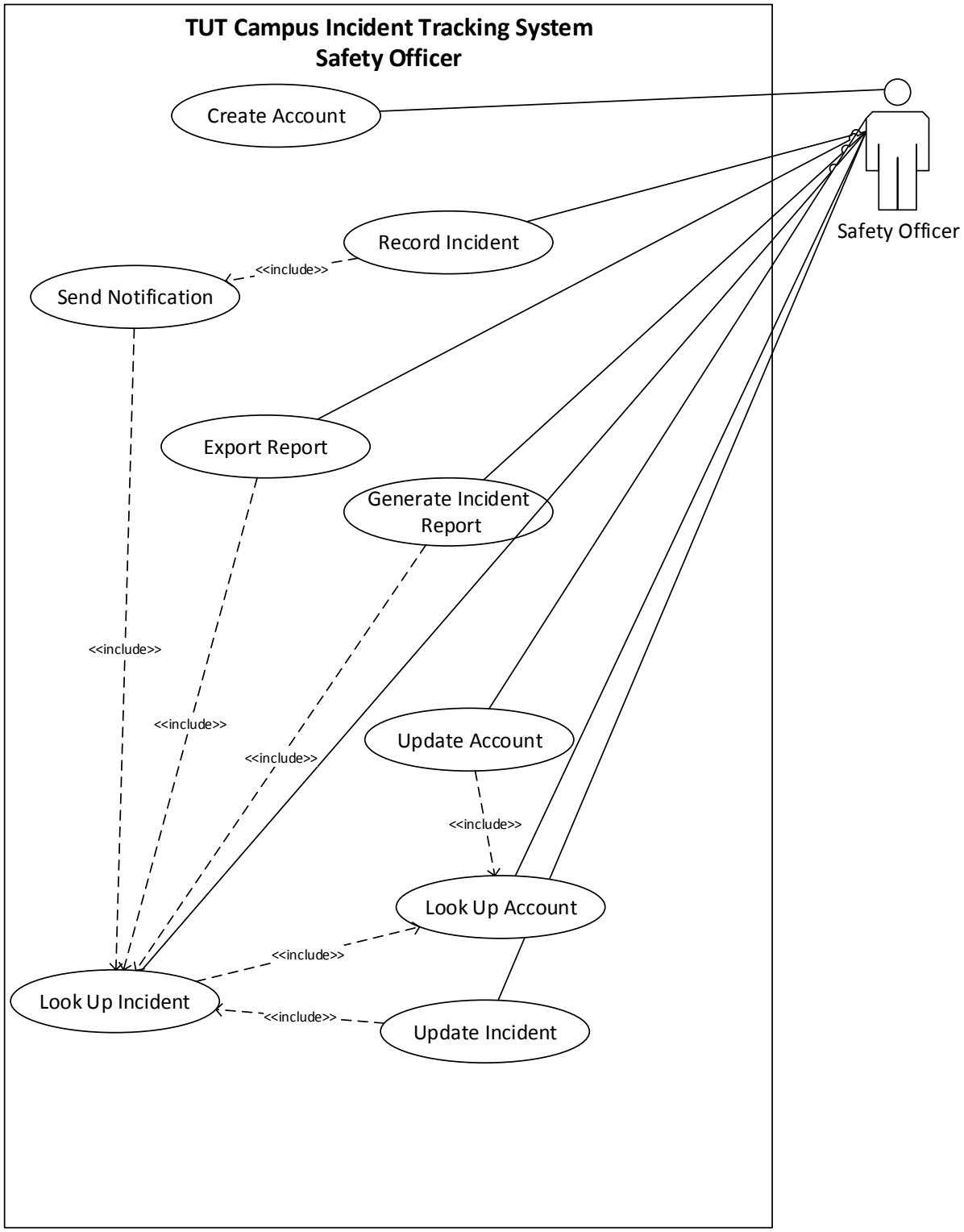
- The system should be compatible with all browsers.

Use Case

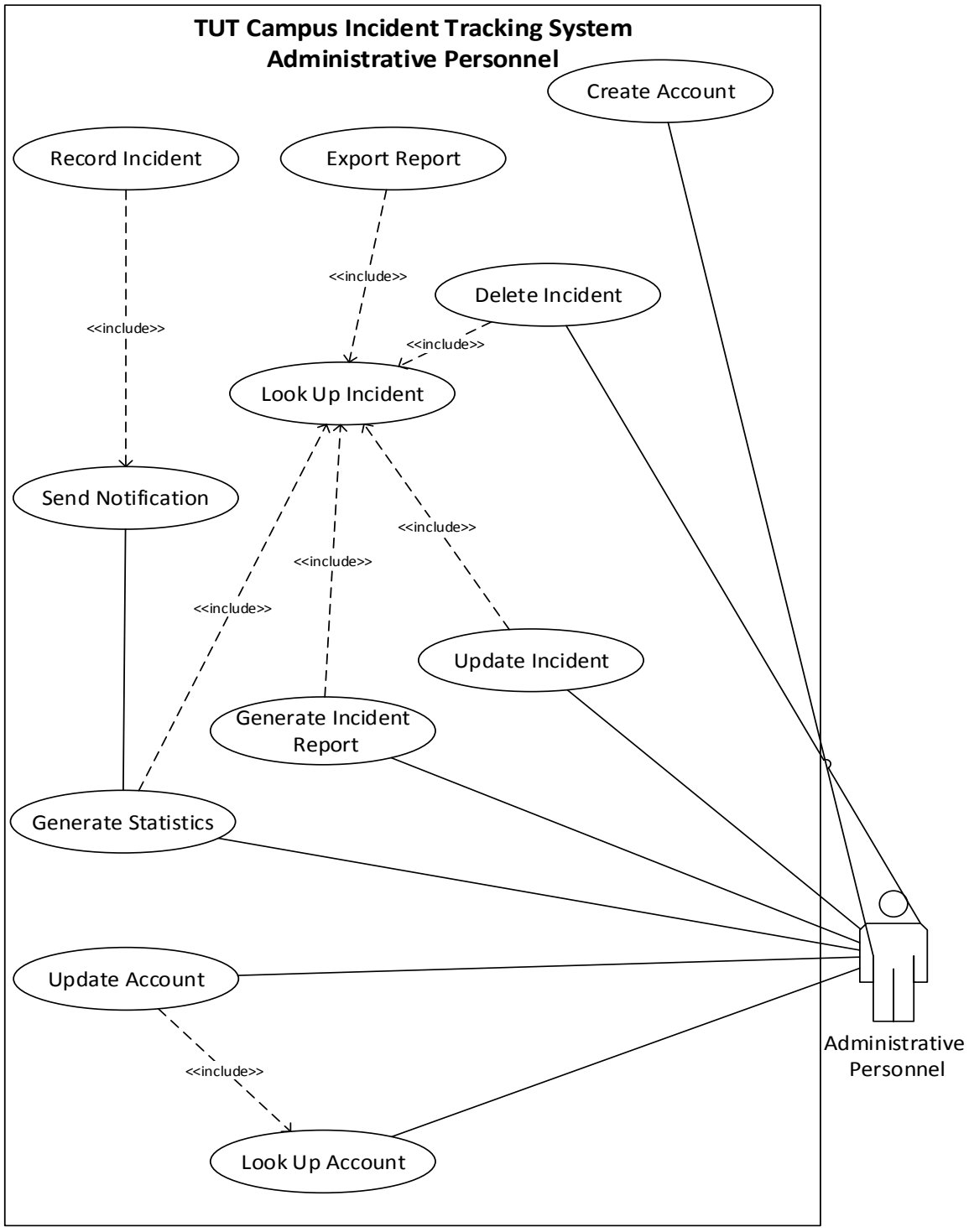
Student Use Case Diagram



Safety Officer Use Case Diagram



Administrative Personnel Use Case Diagram



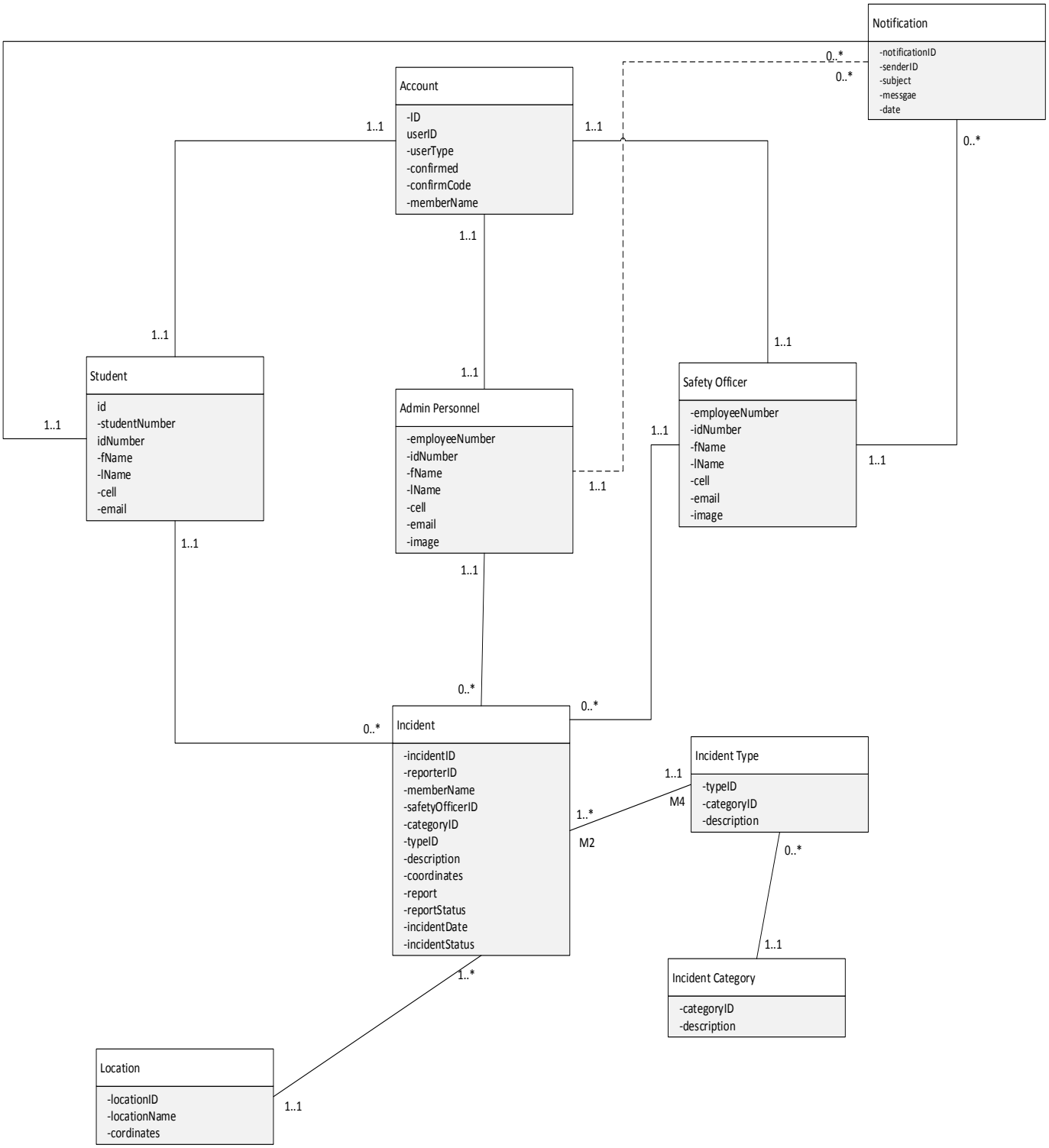
Tools and Technologies to be used

- HTML/5.
- CSS3 (Cascade Style Sheets).
- PHP
- XAMPP 1.8.3 (localhost webserver)
- phpMyAdmin (MySQL localhost server)
- FPDF (PDF Generator)

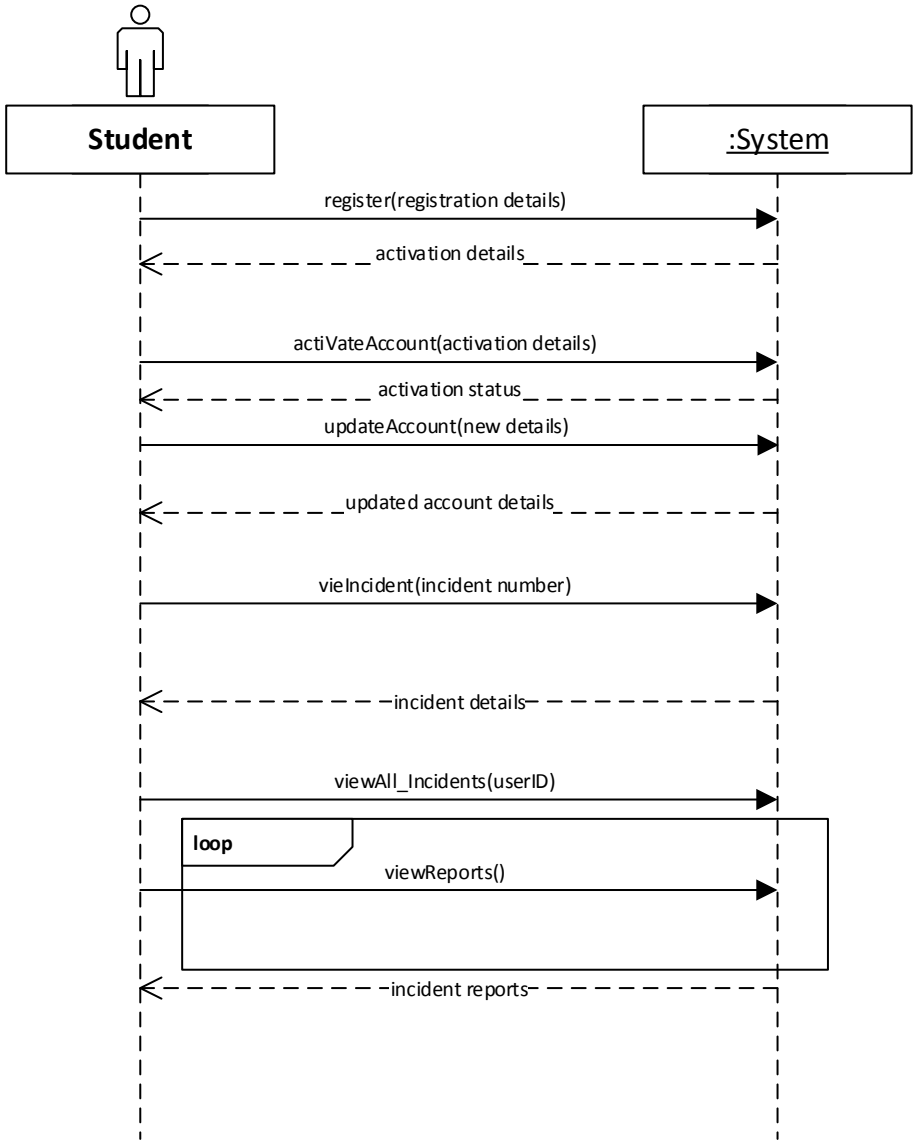
Phase 2 (Modelling with Classes)

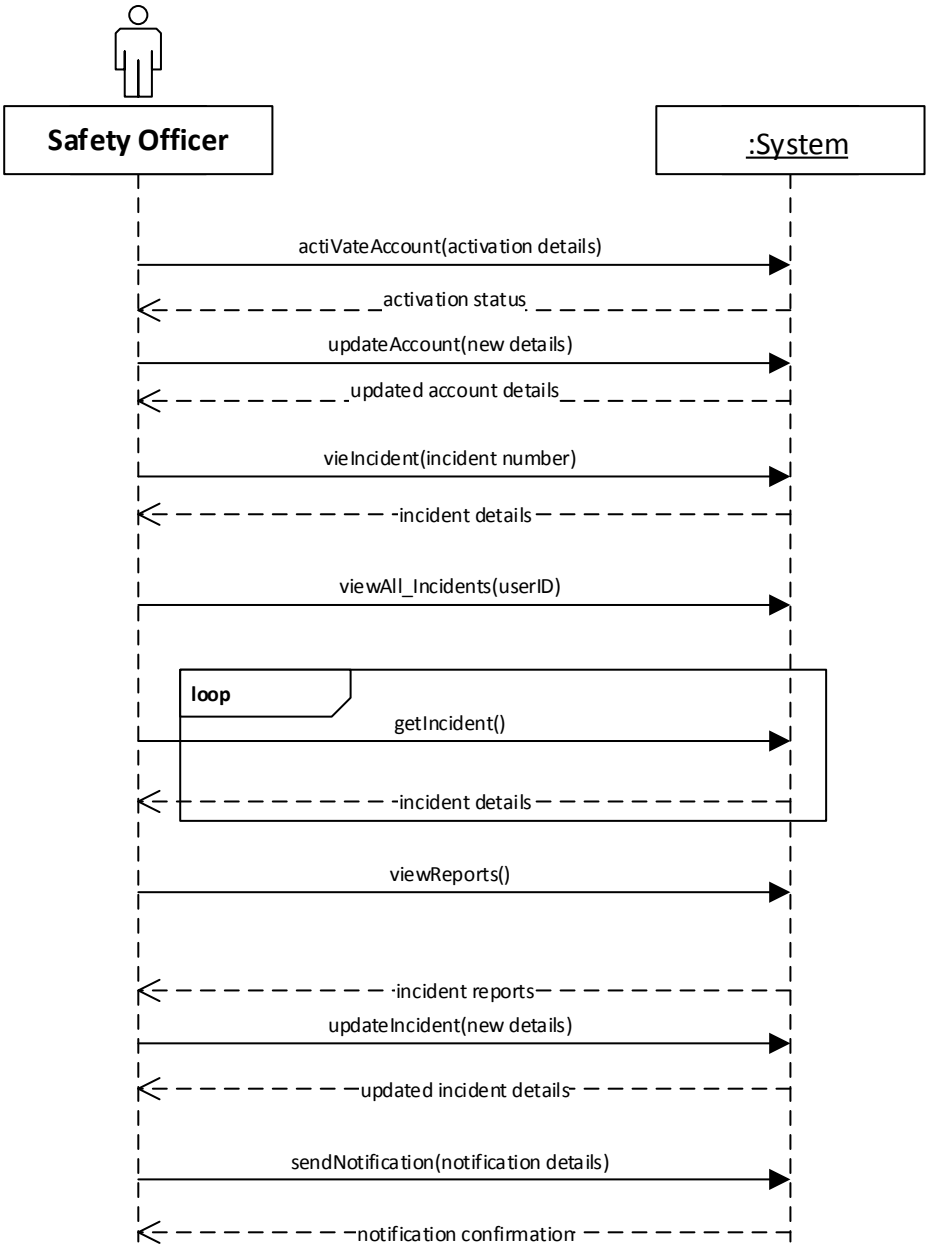
Class Diagrams

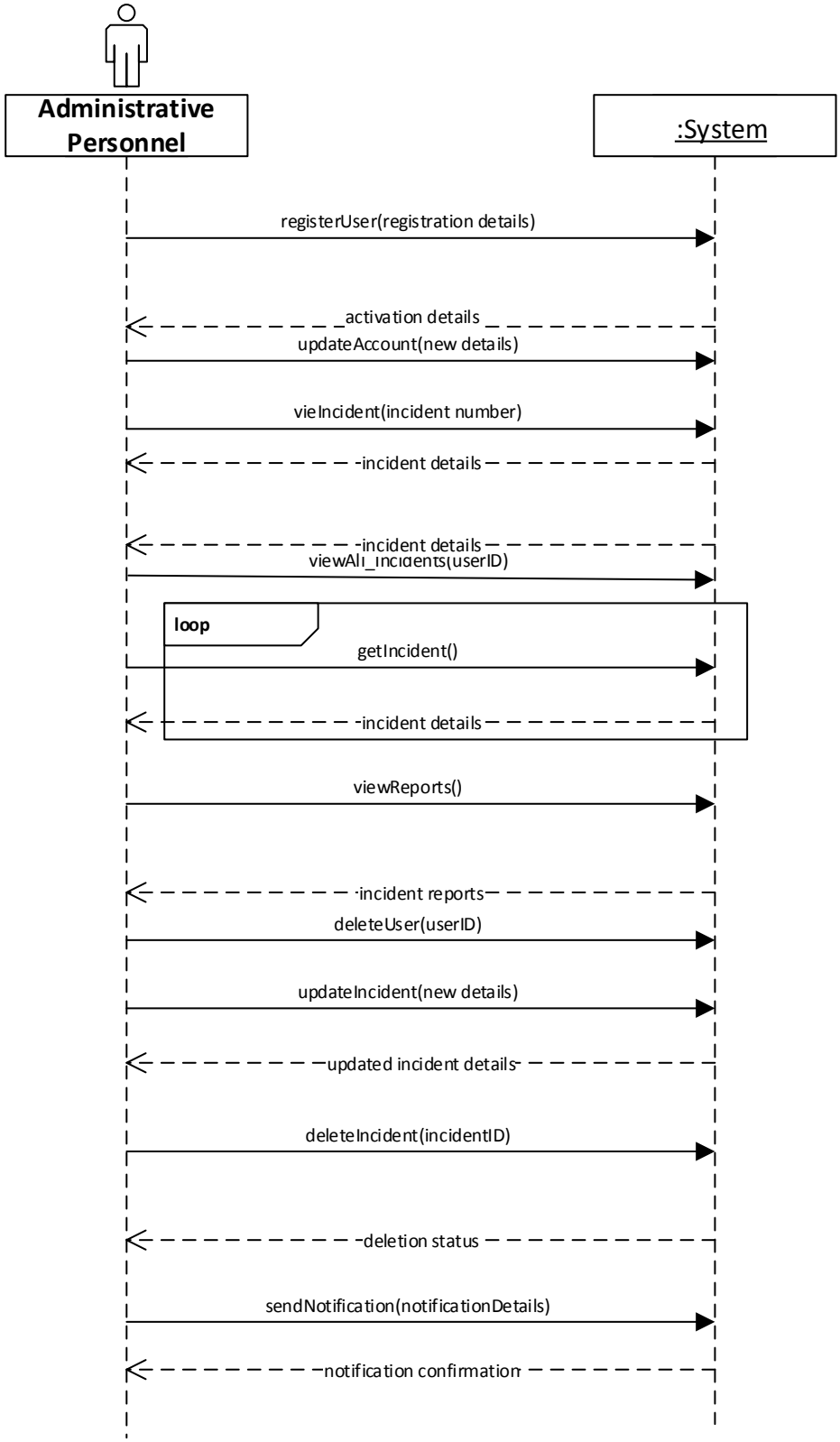
Domain Model Class Diagram of TUT CITS
(Campus Incident Tracking System)



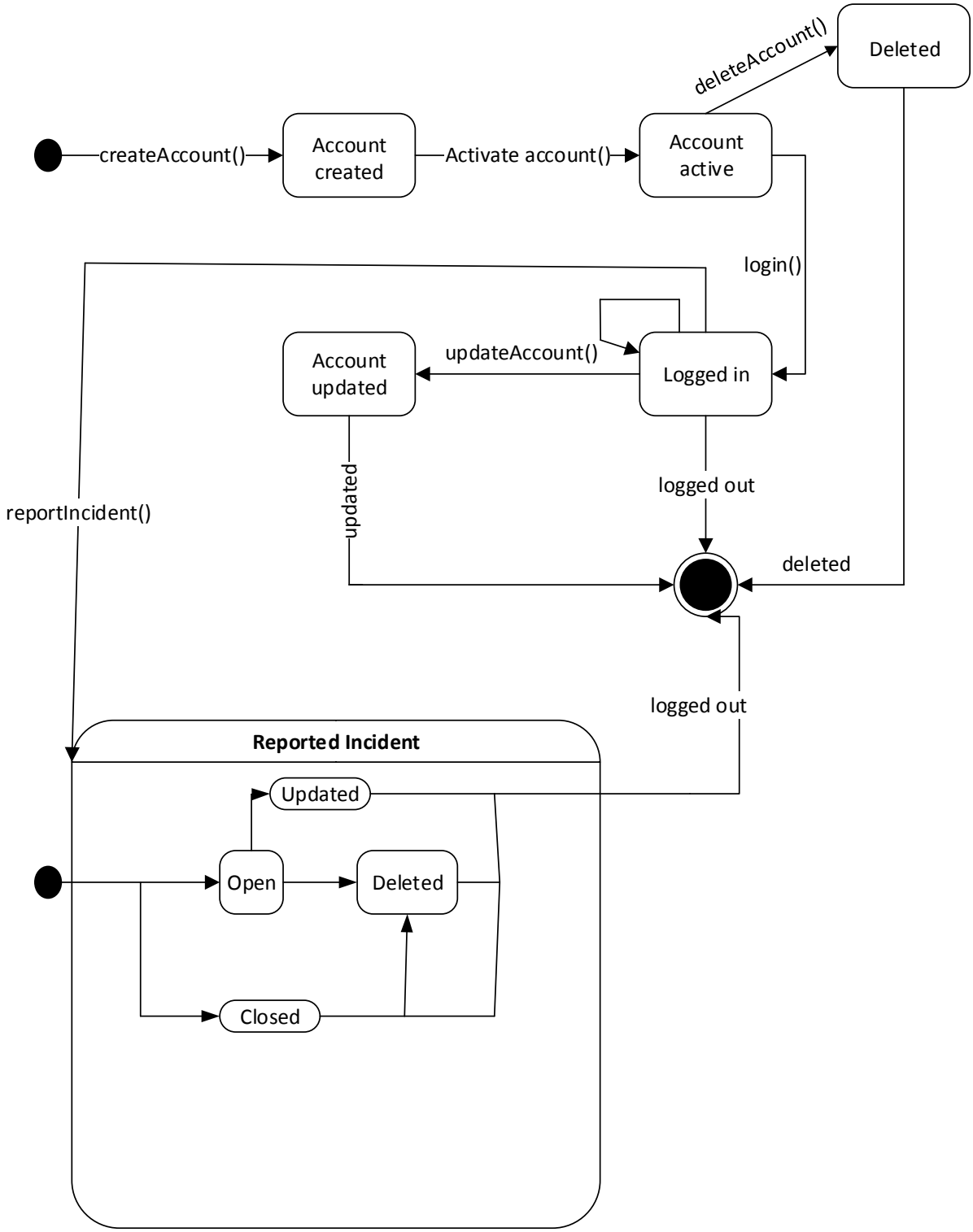
Sequence Diagram



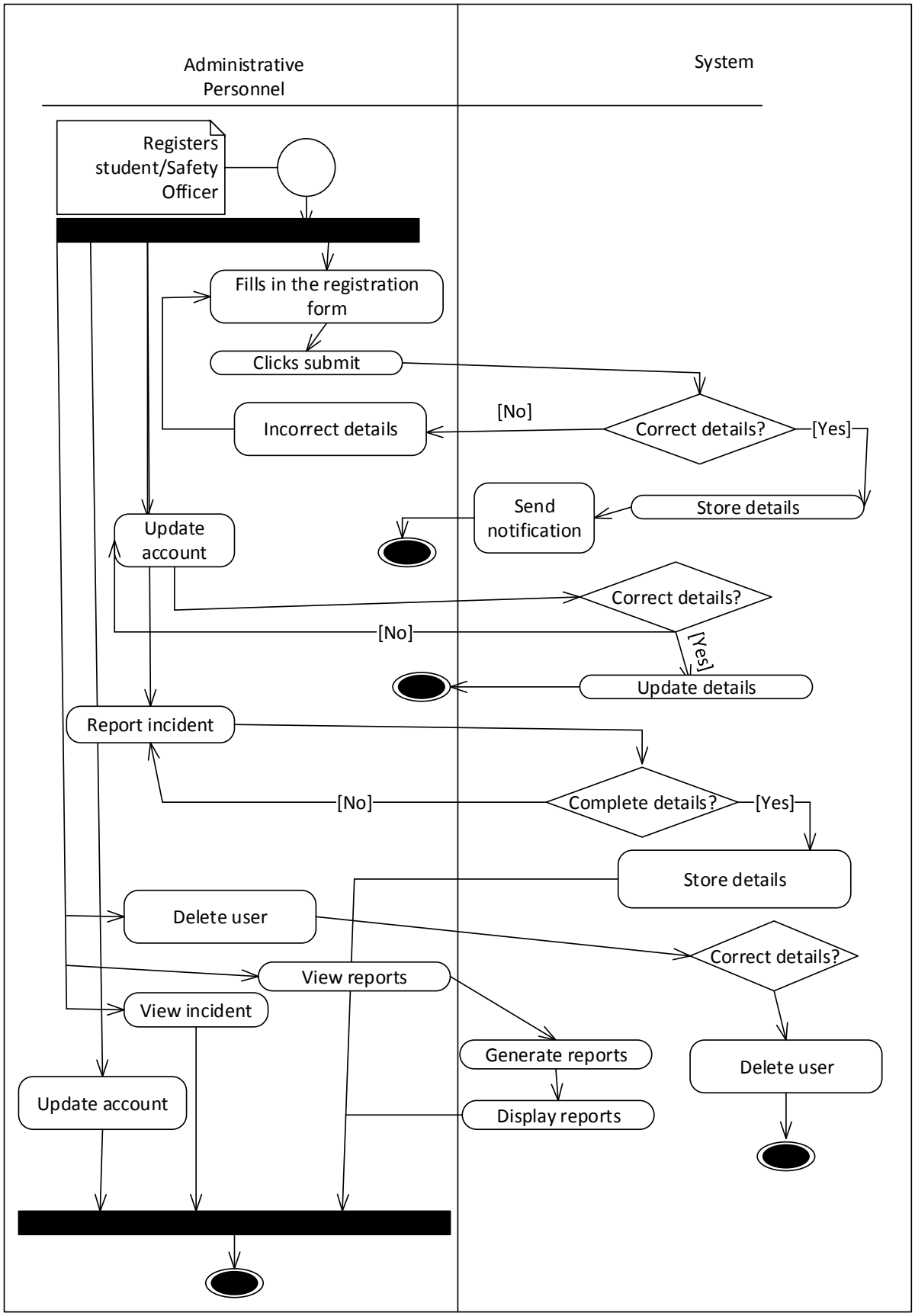

Safety Officer
:System
actiVateAccount(activation details)
activation status
updateAccount(new details)
updated account details
vieIncident(incident number)
incident details
viewAll_Incidents(userID)
loop
getIncident()
incident details
viewReports()
incident reports
updateIncident(new details)
updated incident details
sendNotification(notification details)
notification confirmation

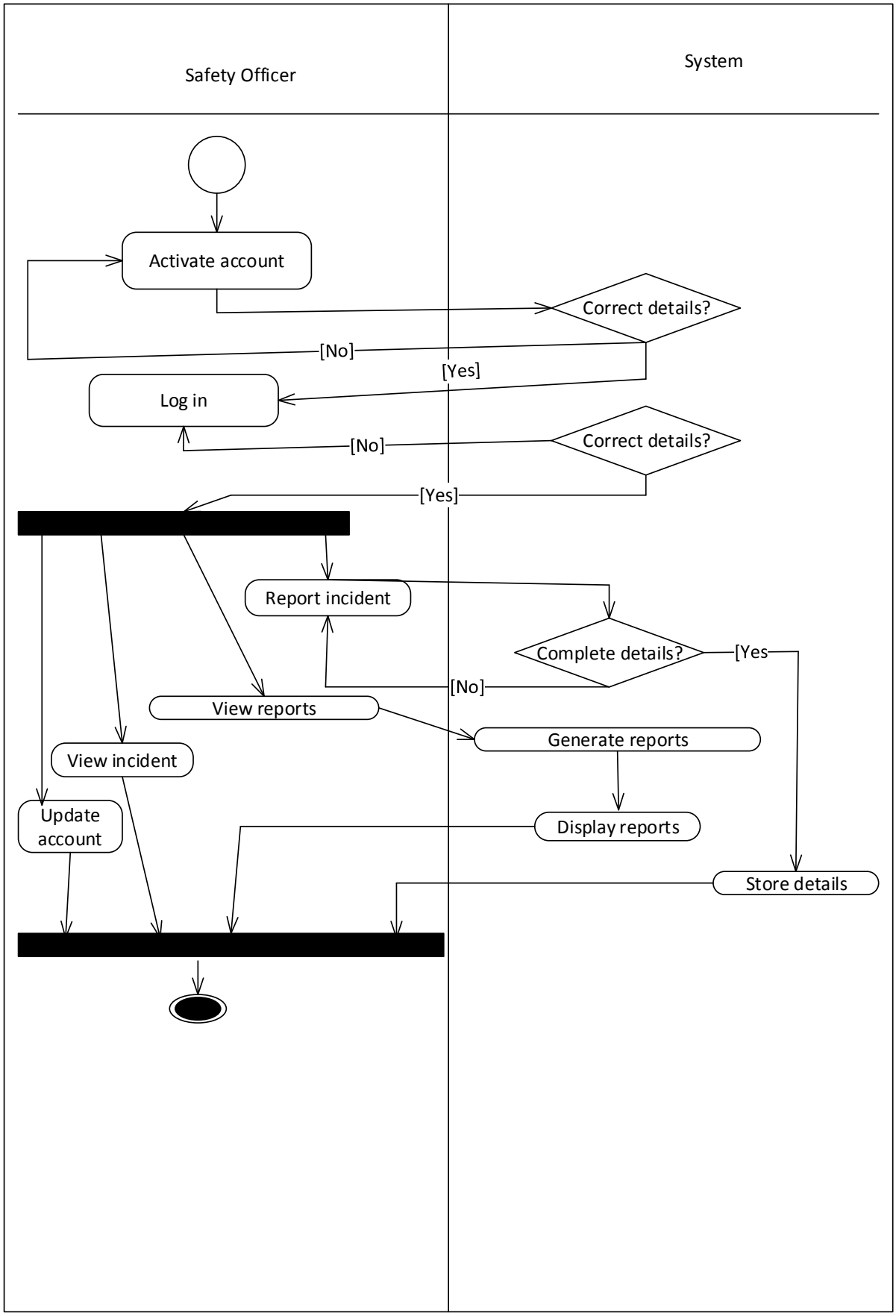

Administrative Personnel
:System
registerUser(registration details)
activation details
updateAccount(new details)
vieIncident(incident number)
incident details
incident details
viewAll_Incidents(userID)
loop
getIncident()
incident details
viewReports()
incident reports
deleteUser(userID)
updateIncident(new details)
updated incident details
deleteIncident(incidentID)
deletion status
sendNotification(notificationDetails)
notification confirmation

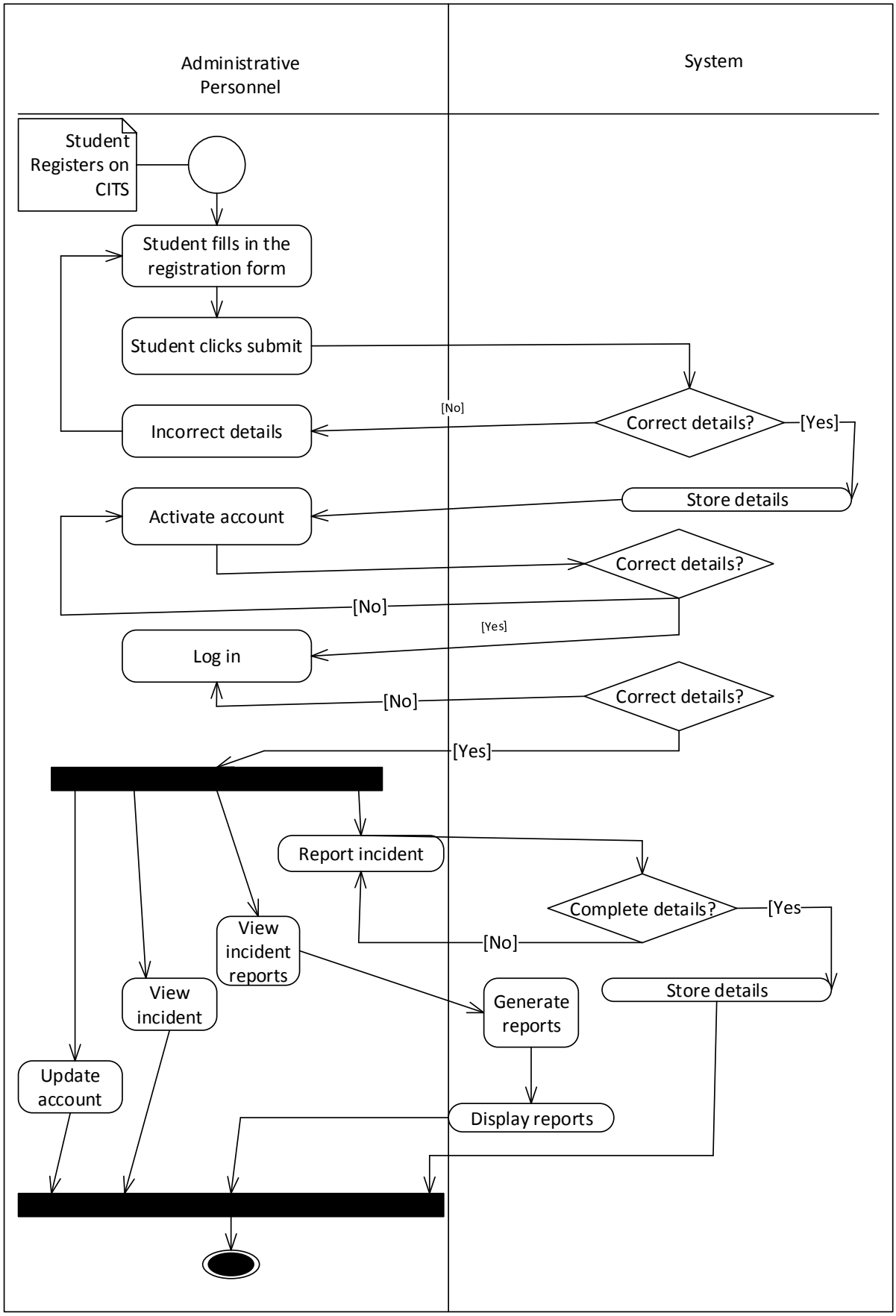
State Diagrams



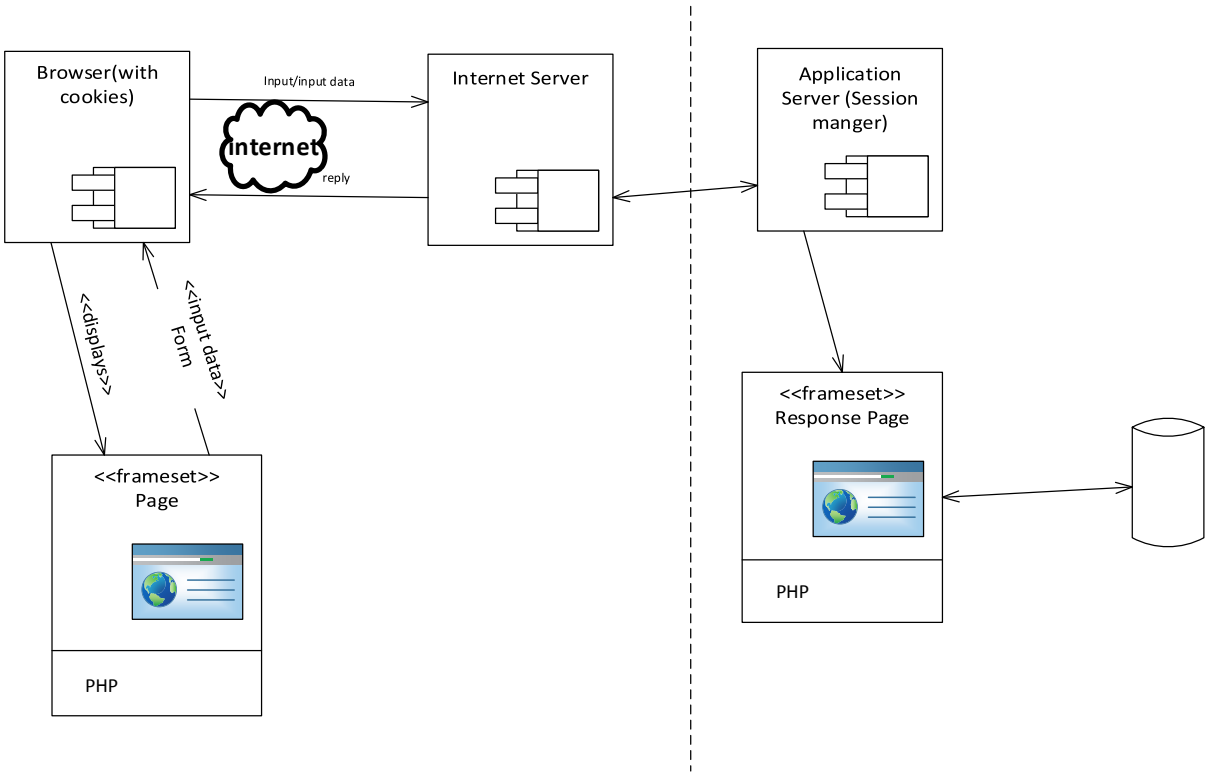
Activity Diagrams


Administrative Personnel
System
Registers student/Safety Officer
Fills in the registration form
Clicks submit
Incorrect details
[No]
Correct details?
[Yes]
Store details
Send notification
Update account
Correct details?
[No]
[Yes]
Update details
Report incident
[No]
Complete details?
[Yes]
Store details
Delete user
Correct details?
View reports
View incident
Generate reports
Delete user
Update account
Display reports

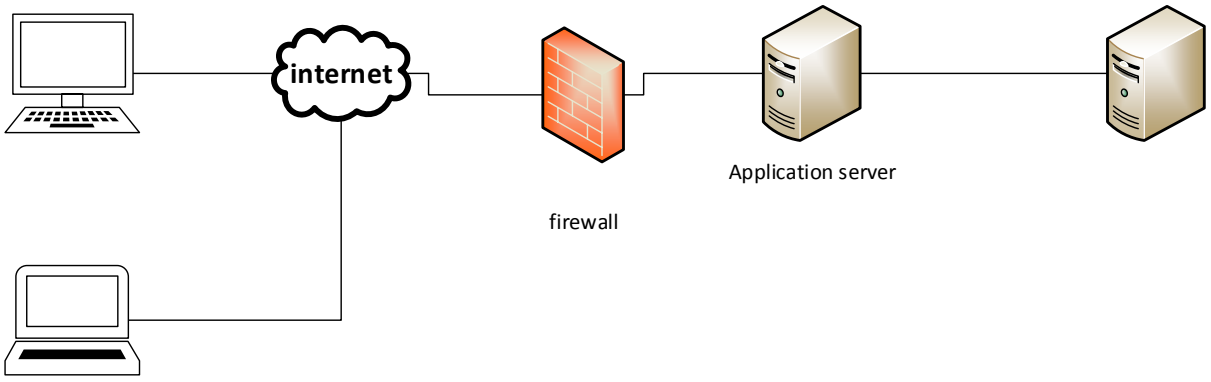

Safety Officer
System
Activate account
Correct details?
[No]
[Yes]
Log in
Correct details?
[No]
[Yes]
Report incident
Complete details?
[Yes
[No]
View reports
Generate reports
View incident
Update account
Display reports
Store details


Administrative Personnel
System
Student Registers on CITS
Student fills in the registration form
Student clicks submit
Correct details?
[No]
[Yes]
Incorrect details
Store details
Activate account
Correct details?
[No]
[Yes]
Log in
Correct details?
[No]
[Yes]
Report incident
Complete details?
[Yes
[No]
View incident reports
View incident
Generate reports
Store details
Update account
Display reports

Component Diagrams



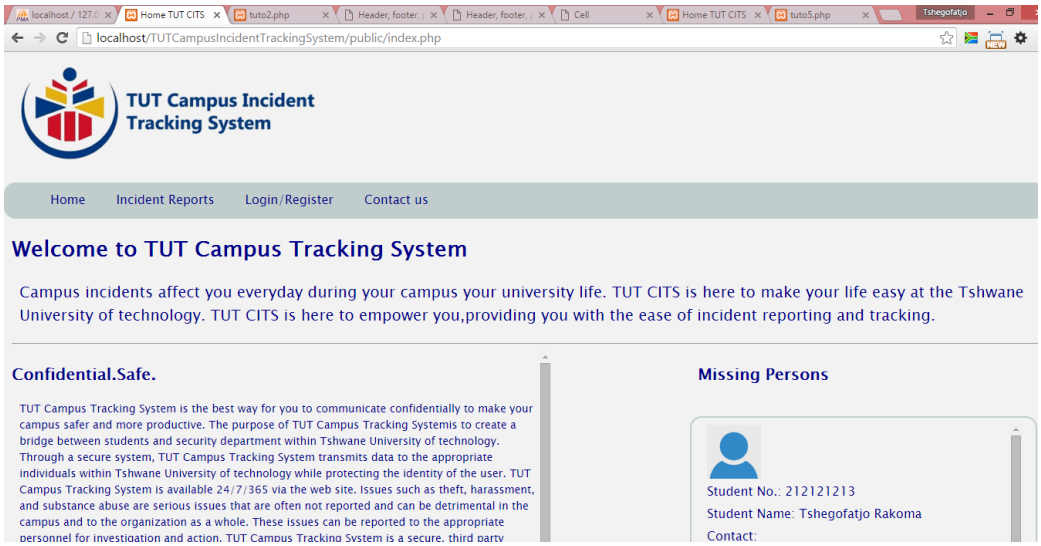
Deployment Diagram



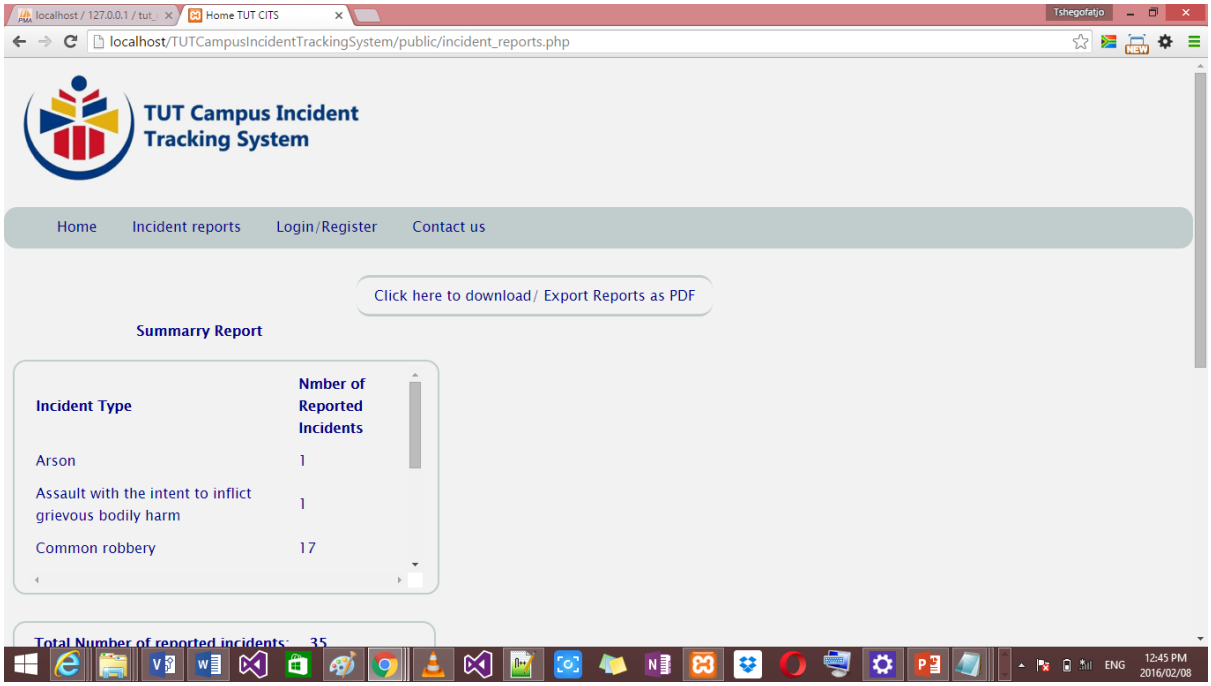
Phase 3 (User Interface)

Design User Interfaces

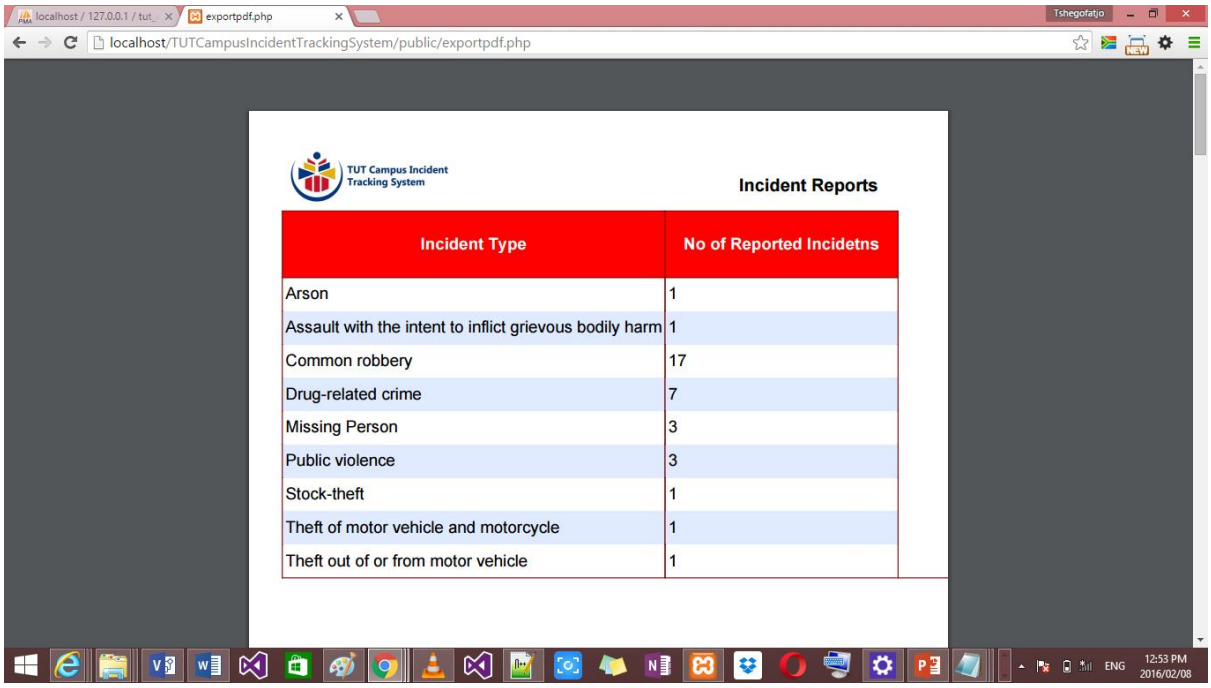
Home Page



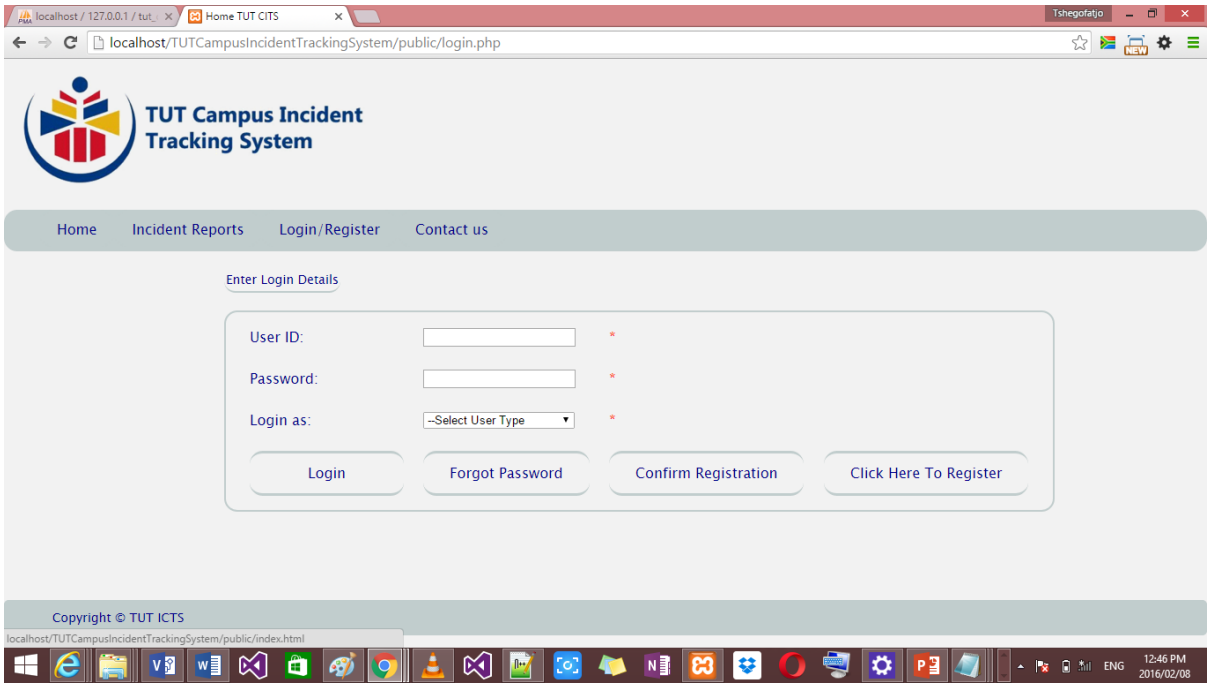
Incident Reports



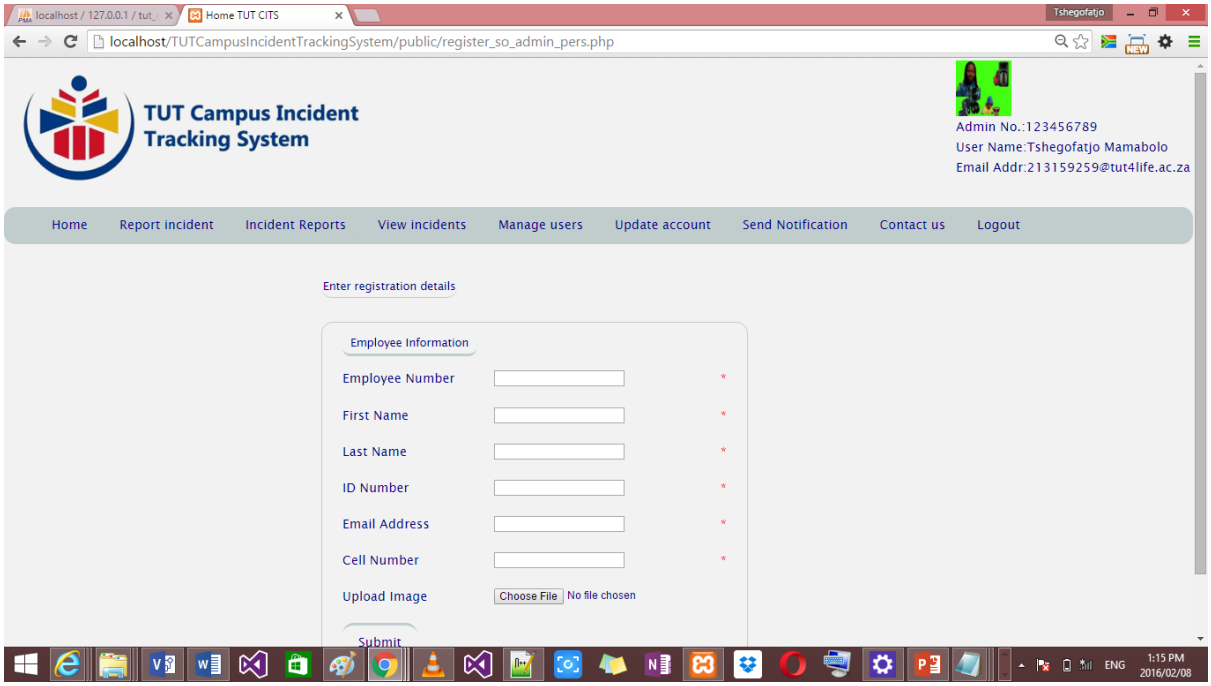
Export Reports



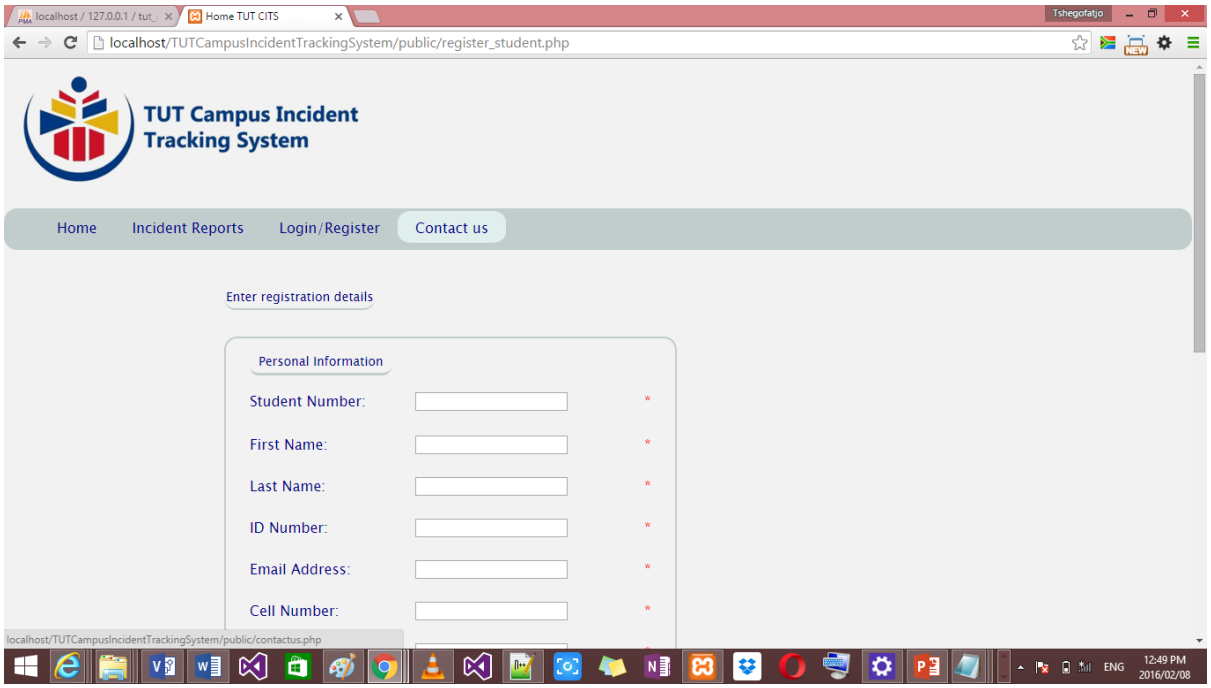
Login Page



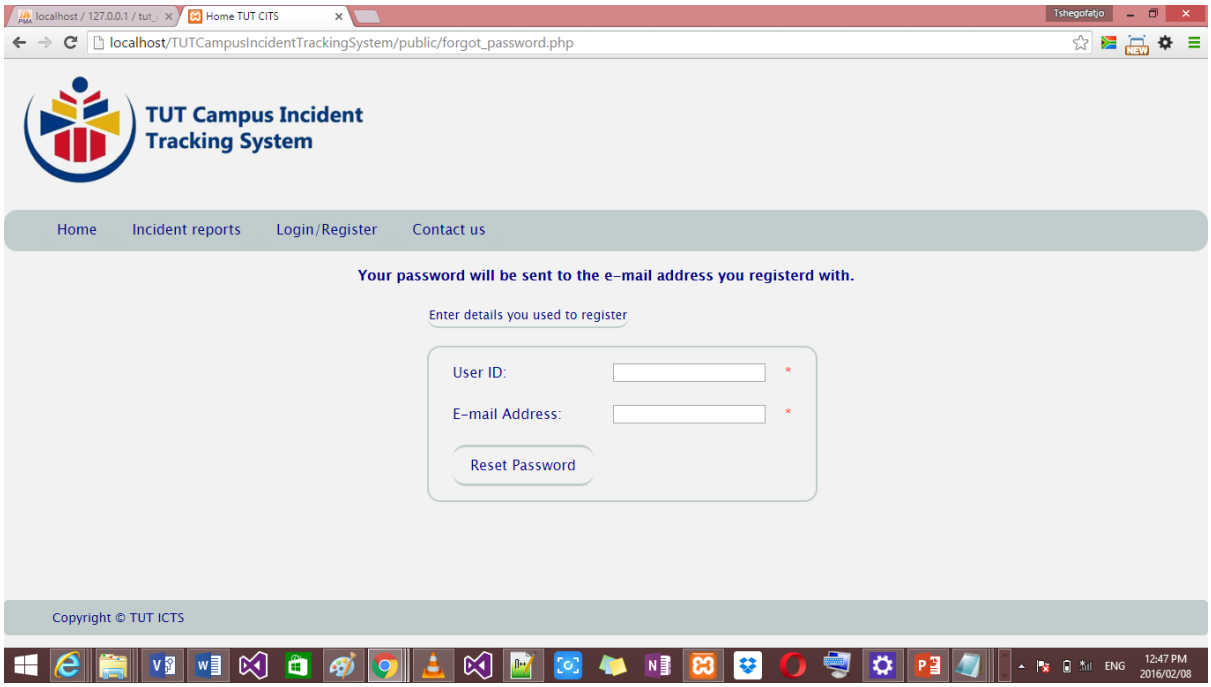
Student Registration Page



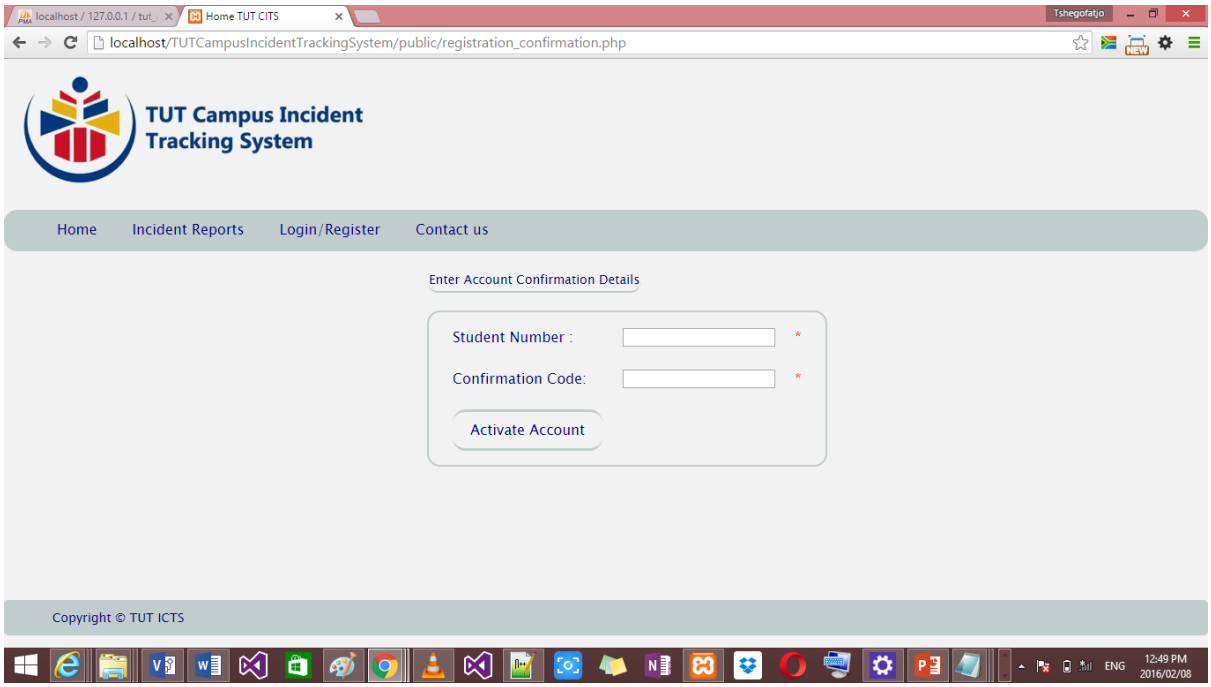
Safety officer/Administrative personnel Registration Page



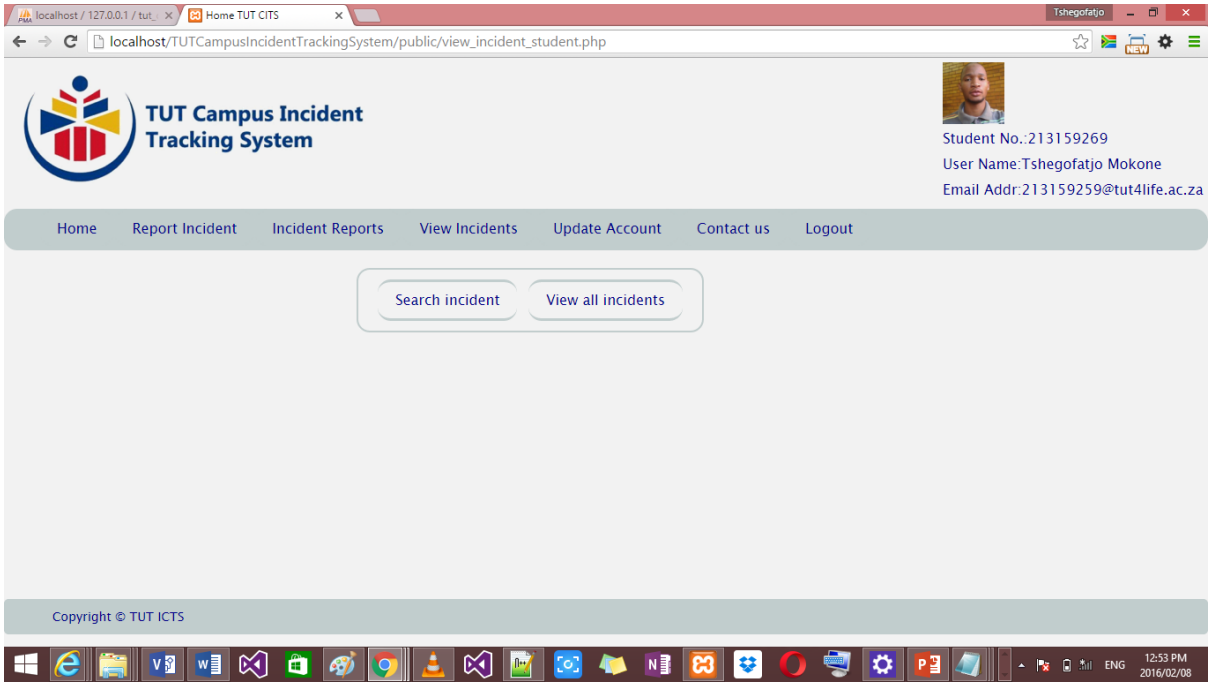
Activate Account Page



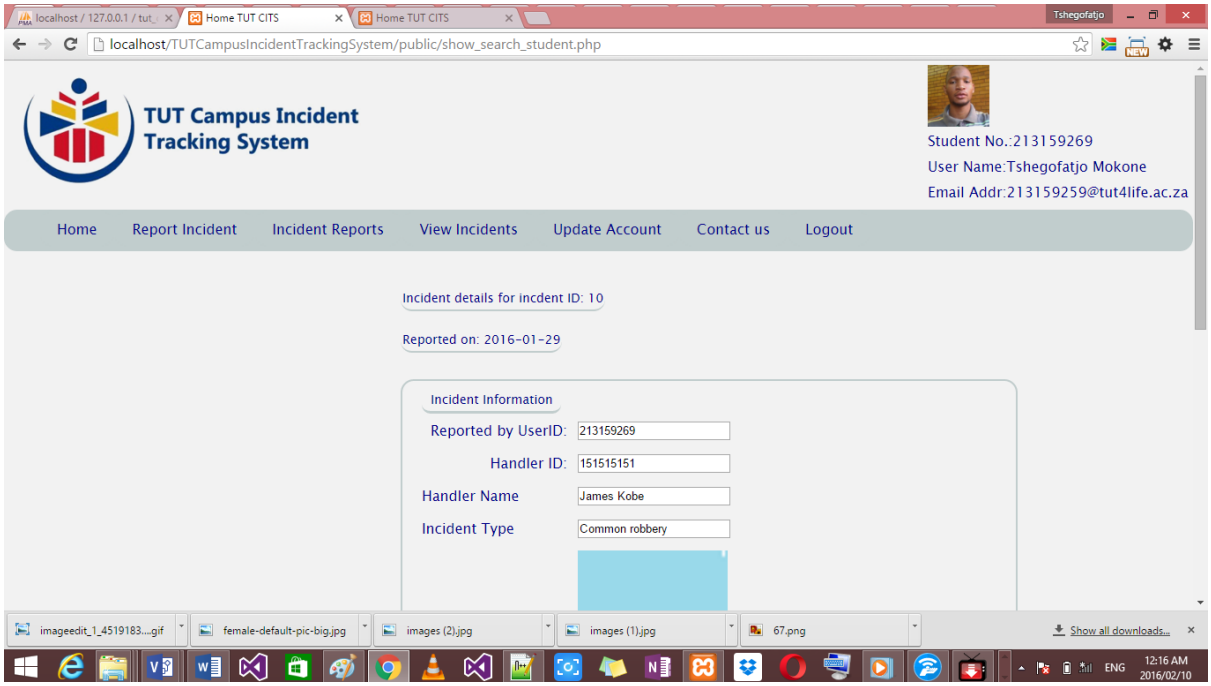
Forgo Password Page



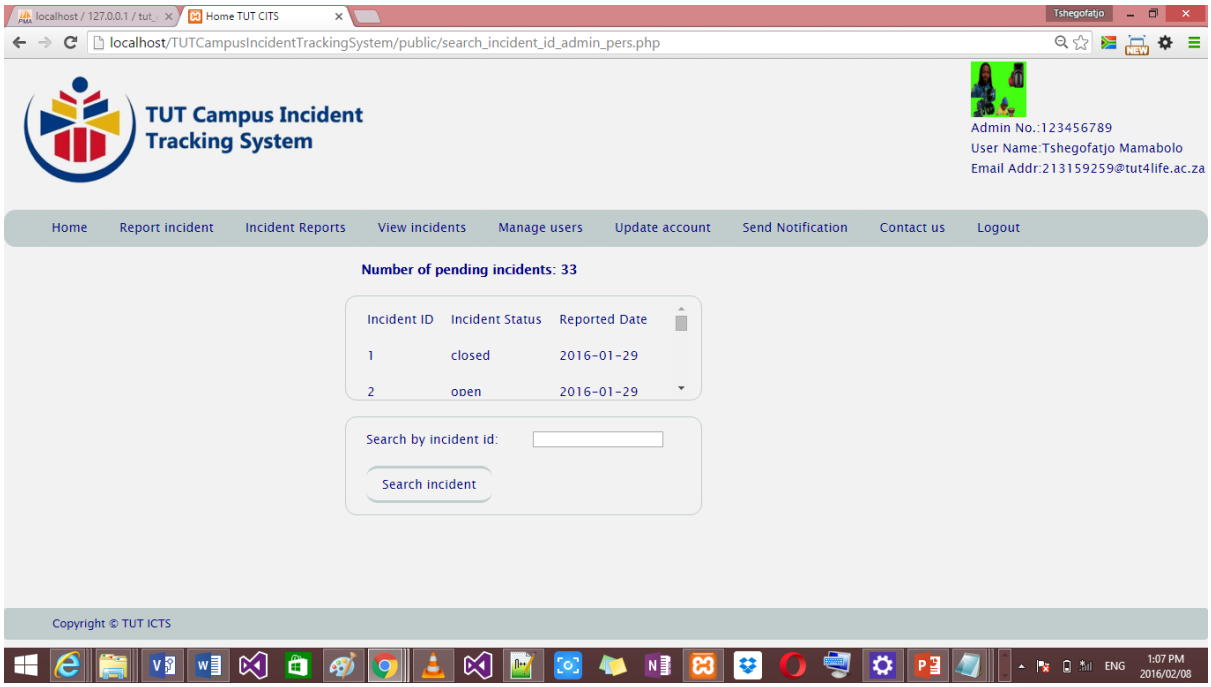
View incident Page



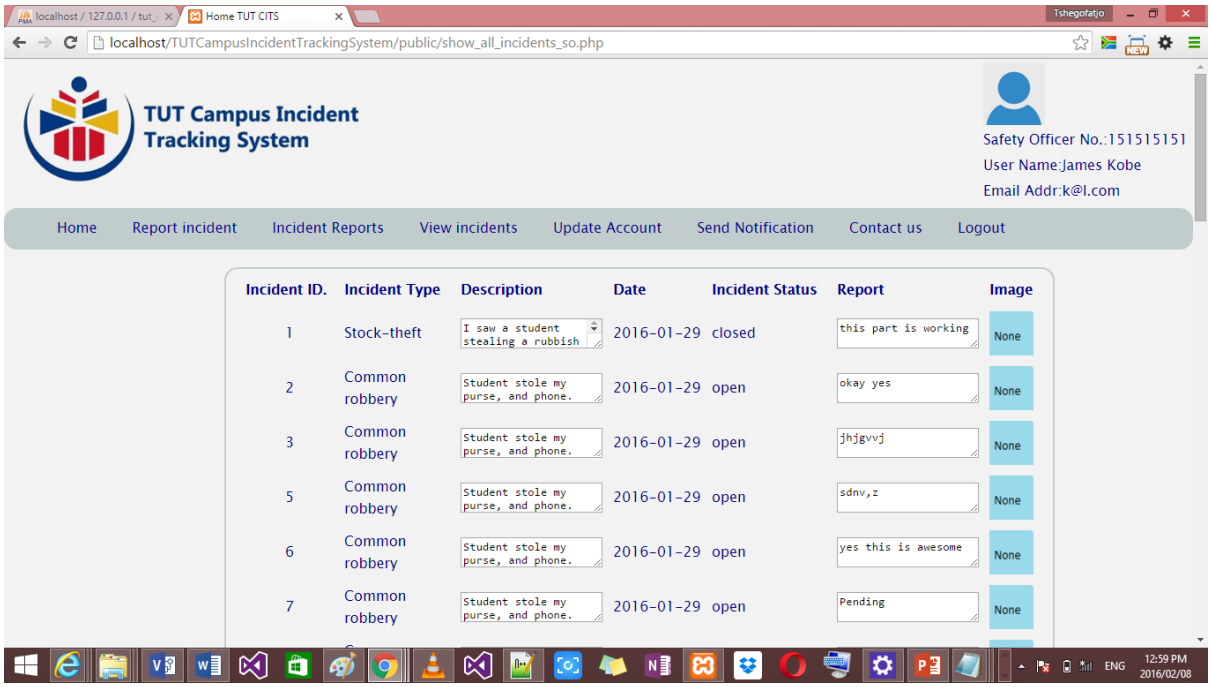
Searched incident Page



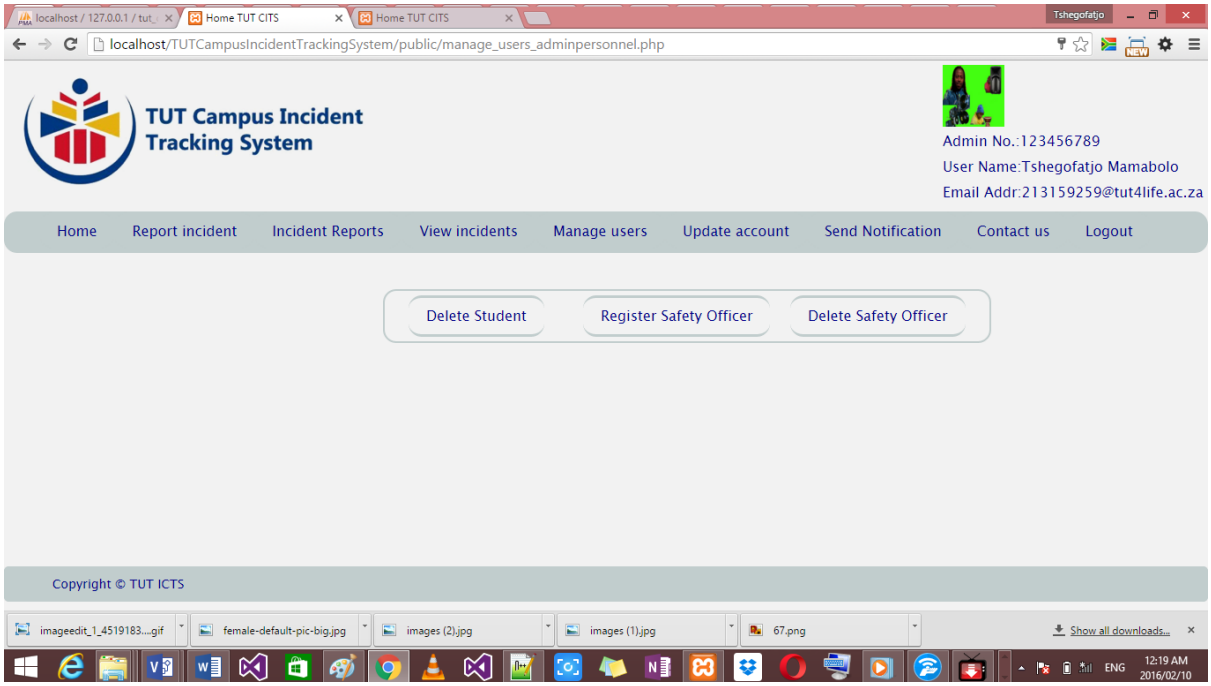
Search incident Page



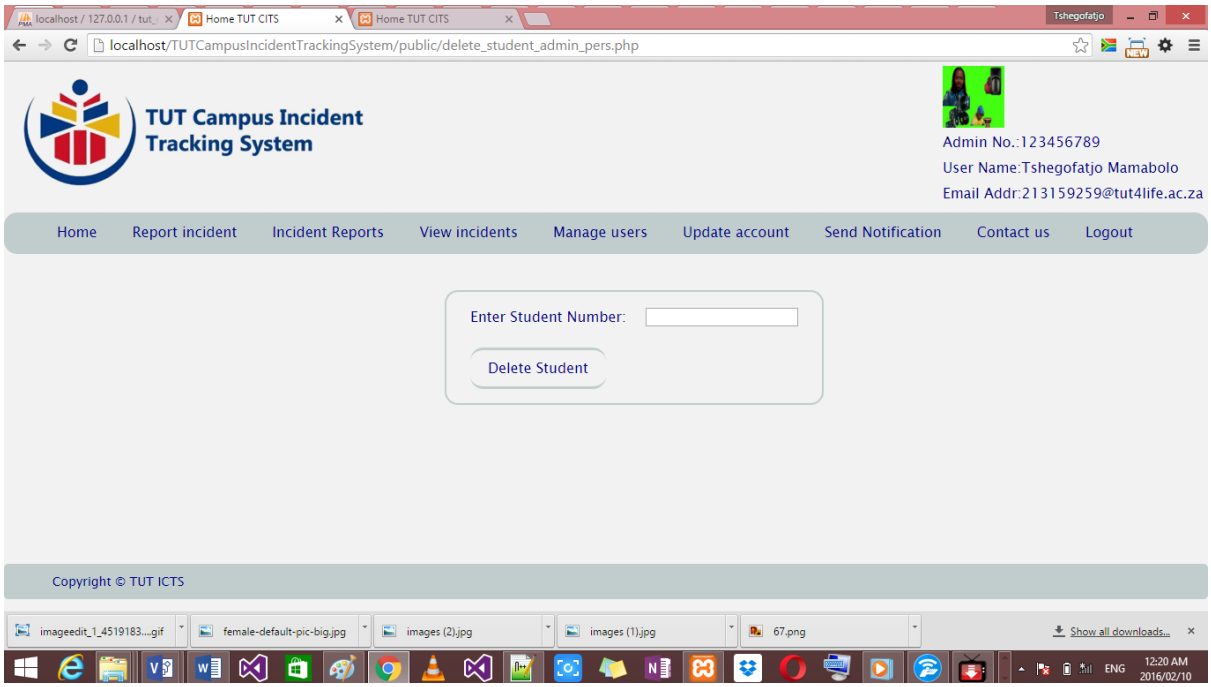
View all incidents Page



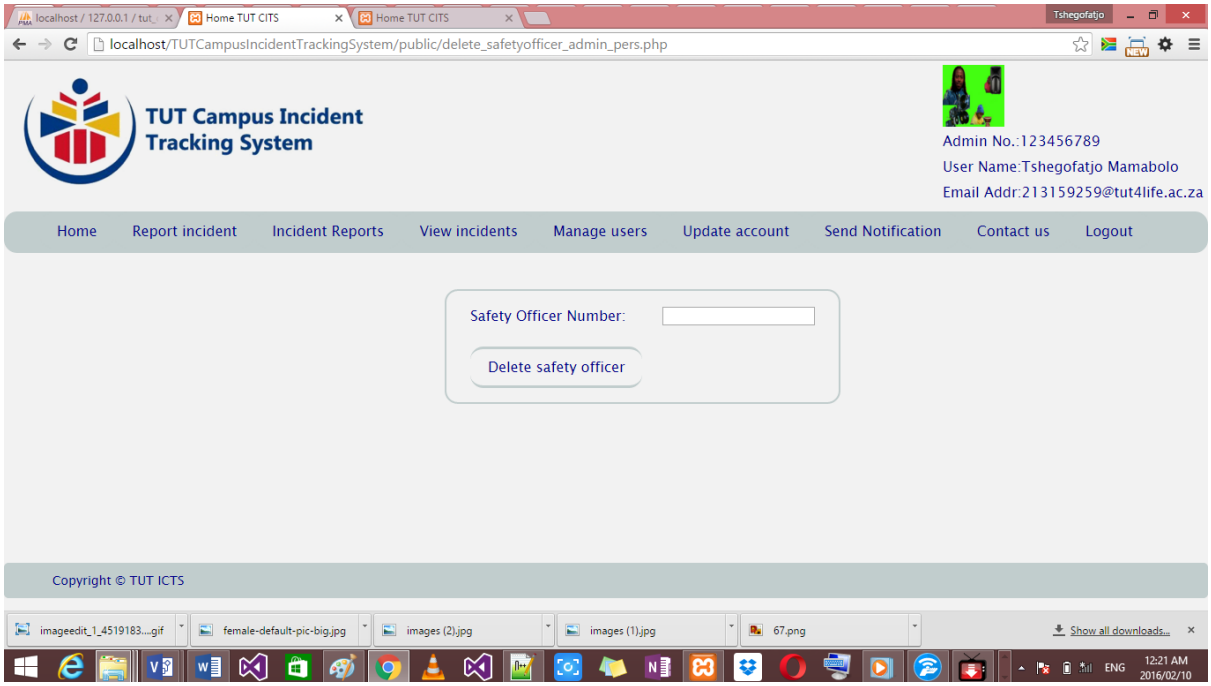
Manage Users Page



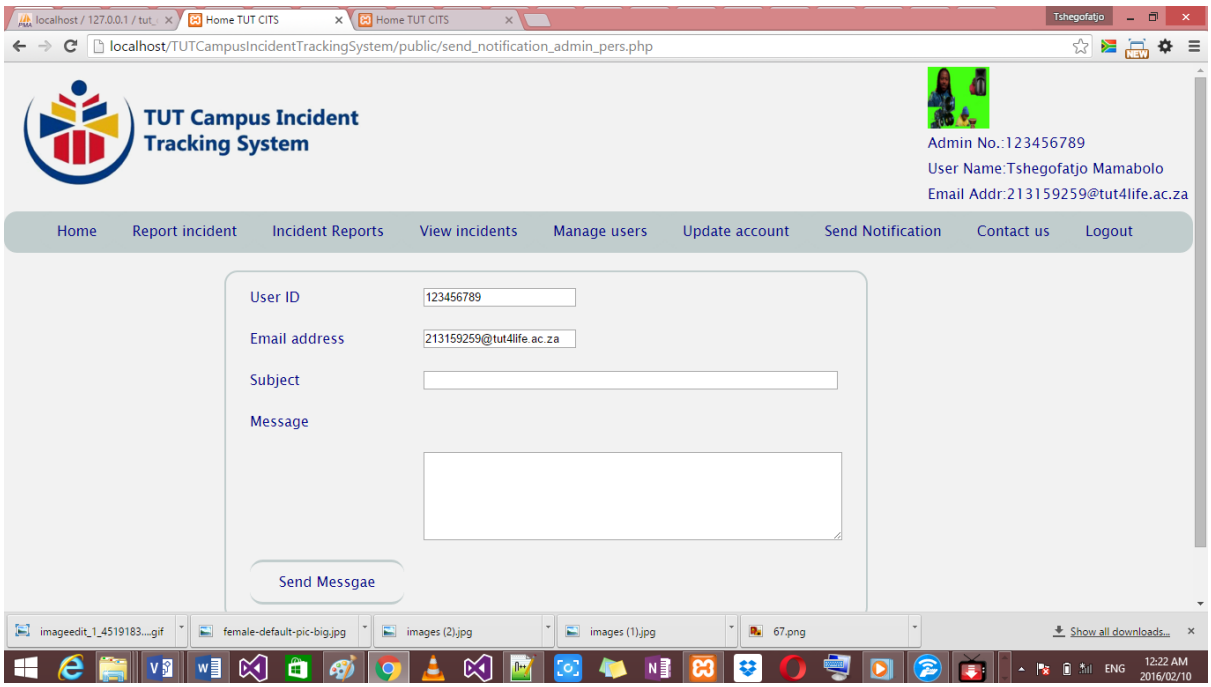
Delete Student Page



Delete Safety Officer Page

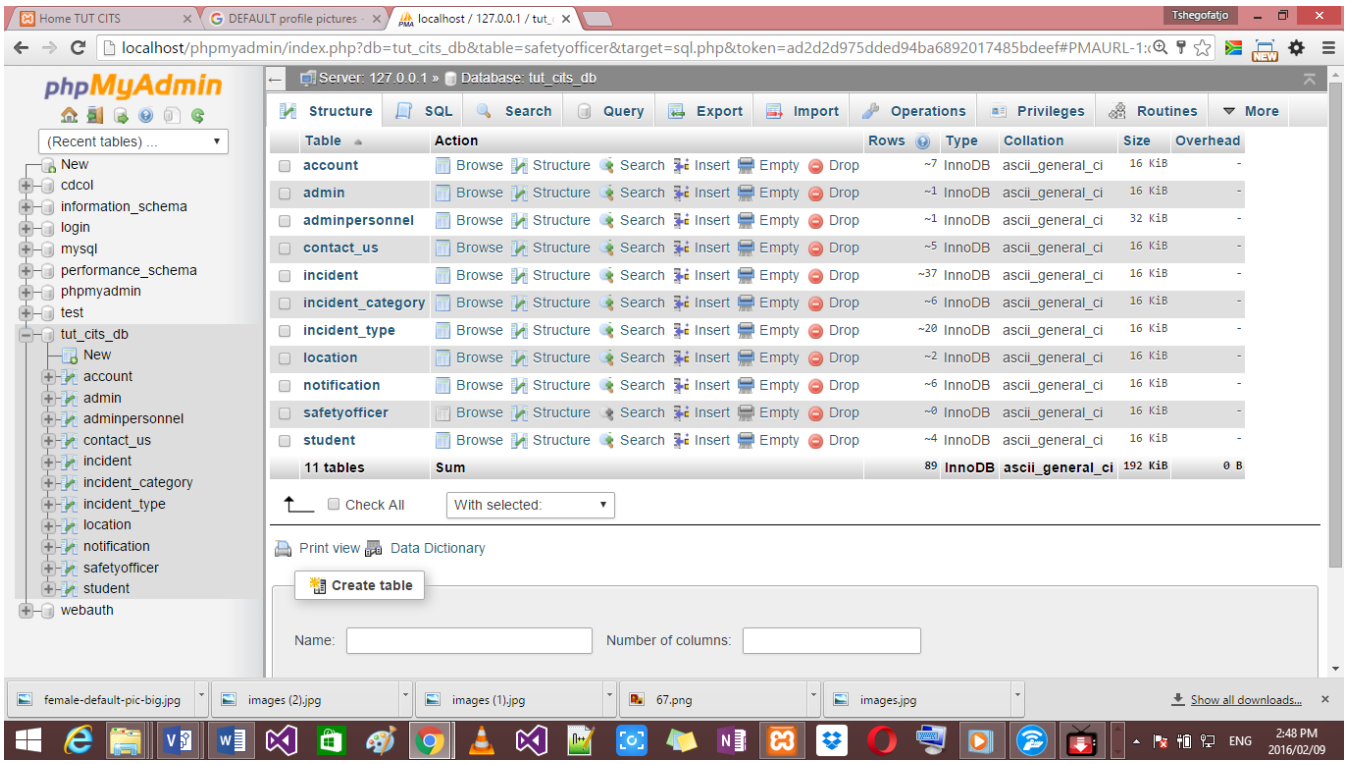


Send notification Page



Phase 4(Build Database and Demonstrate Integration)

Database Structure



Manage objects

CREATE TABLE IF NOT EXISTS `account` (`id` int(9) NOT NULL AUTO_INCREMENT, `userID` int(9) NOT NULL,`userType` varchar(10) NOT NULL, `password` varchar(30) NOT NULL,`confirmed` int(1) NOT NULL,`confirmCode` int(5) NOT NULL, PRIMARY KEY (`id`,`userID`)) ENGINE=InnoDB DEFAULT CHARSET=ascii COMMENT='Stores login information for a registered user' AUTO_INCREMENT=57 ;

CREATE TABLE IF NOT EXISTS `admin` (`adminid` varchar(30) NOT NULL, `name` varchar(30) NOT NULL, `password` varchar(30) NOT NULL, `image` varchar(50) NOT NULL) ENGINE=InnoDB DEFAULT CHARSET=ascii;

CREATE TABLE IF NOT EXISTS `adminpersonnel` (`id` int(11) NOT NULL AUTO_INCREMENT,`adminid` int(9) NOT NULL,`idnumber` varchar(13) NOT NULL,`fname` varchar(20) NOT NULL, `lname` varchar(20) NOT NULL, `email` varchar(30) NOT NULL, `cell` varchar(12) NOT NULL, `image` varchar(50) NOT NULL,PRIMARY KEY (`adminid`),UNIQUE KEY `id` (`id`)) ENGINE=InnoDB DEFAULT CHARSET=ascii AUTO_INCREMENT=2 ;

CREATE TABLE IF NOT EXISTS `contact_us` (`messageid` int(10) NOT NULL AUTO_INCREMENT,`userid` int(9) NOT NULL, `email` varchar(100) NOT NULL, `message` varchar(50000) NOT NULL, `date` varchar(100) NOT NULL, PRIMARY KEY (`messageid`)) ENGINE=InnoDB DEFAULT CHARSET=ascii AUTO_INCREMENT=18 ;

CREATE TABLE IF NOT EXISTS `incident` (`incidentID` int(10) NOT NULL AUTO_INCREMENT, `reporterID` int(9) NOT NULL, `handlerID` int(9) NOT NULL, `categoryID` int(9) NOT NULL, `typeID` int(9) NOT NULL,

`handlerName` varchar(35) NOT NULL,`description` varchar(5000) NOT NULL,`image` varchar(50) NOT NULL, `coordinates` varchar(35) NOT NULL, `incidentDate` date NOT NULL,`incidentStatus` varchar(20) NOT NULL, `report` varchar(5000) NOT NULL, `reportStatus` varchar(10) NOT NULL, `locationid` varchar(10) NOT NULL, `locationdesc` varchar(200) NOT NULL, `missingStudId` varchar(9) NOT NULL,PRIMARY KEY (`incidentID`)) ENGINE=InnoDB DEFAULT CHARSET=ascii AUTO_INCREMENT=42 ;

CREATE TABLE IF NOT EXISTS `incident_category` (`categoryID` int(10) NOT NULL AUTO_INCREMENT,`name` varchar(50) NOT NULL,`description` varchar(500) NOT NULL,PRIMARY KEY (`categoryID`)) ENGINE=InnoDB DEFAULT CHARSET=ascii AUTO_INCREMENT=8 ;

CREATE TABLE IF NOT EXISTS `incident_type` (`typeID` int(10) NOT NULL AUTO_INCREMENT, `categoryID` int(10) NOT NULL, `name` varchar(100) NOT NULL, `description` varchar(500) NOT NULL,PRIMARY KEY (`typeID`)) ENGINE=InnoDB DEFAULT CHARSET=ascii AUTO_INCREMENT=22 ;

CREATE TABLE IF NOT EXISTS `location` (`locationID` int(10) NOT NULL AUTO_INCREMENT, `locationName` varchar(100) NOT NULL, `coordinates` varchar(50) NOT NULL, PRIMARY KEY (`locationID`)) ENGINE=InnoDB DEFAULT CHARSET=ascii AUTO_INCREMENT=3 ;

CREATE TABLE IF NOT EXISTS `notification` (`notificationID` int(11) NOT NULL AUTO_INCREMENT,`senderID` varchar(9) NOT NULL, `subject` varchar(250) NOT NULL, `message` varchar(10000) NOT NULL,`date` varchar(100) NOT NULL,

PRIMARY KEY (`notificationID`)) ENGINE=InnoDB DEFAULT CHARSET=ascii AUTO_INCREMENT=7 ;

```
CREATE TABLE IF NOT EXISTS `safetyofficer` ( `id` int(10) NOT NULL AUTO_INCREMENT, `sonumber` int(9) NOT NULL,`fname` varchar(20) NOT NULL, `lname` varchar(20) NOT NULL,`idnumber` varchar(13) NOT NULL, `email` varchar(30) NOT NULL,

`cell` varchar(13) NOT NULL, `image` varchar(50) NOT NULL, PRIMARY KEY (`id`)) ENGINE=InnoDB DEFAULT CHARSET=ascii AUTO_INCREMENT=2 ;


CREATE TABLE IF NOT EXISTS `student` (

`studentNumber` int(9) NOT NULL,`idNumber` varchar(13) NOT NULL,`fName` varchar(20) NOT NULL,

`lName` varchar(20) NOT NULL, `cell` varchar(12) NOT NULL,`email` varchar(30) NOT NULL,

`image` varchar(50) NOT NULL, `gFName` varchar(20) NOT NULL, `gLName` varchar(20) NOT NULL, `gEmail` varchar(30) NOT NULL,

`gCell` varchar(12) NOT NULL, PRIMARY KEY (`studentNumber`)) ENGINE=InnoDB DEFAULT CHARSET=ascii;
```

Normalization Process

UNF

Account(userID,userType,password , confirmed, confirmCode, id,admin_id, admin_idnumber , admin_ fname , admin_lname , admin_email, admin_cell, admin_image,messageid,userid,email ,message,date ,incidentID, reporterID,handlerID,categoryID, typeID, handlerName, description,image,coordinates, incidentDate, incidentStatus,report, reportStatus,categoryID,name,description, typeID,categoryID,name,description,locationID,locationName,coordinates,notificationID,senderID, touserID, subject, message, date,id,so_number, so_fname,so_lname, so_idnumber, so_email, so_cell, so_ image, studentNumber, stud_idNumber, stud_fName, stud_lName, stud_cell, stud_email, stud_image, stud_gFName, stud_gLName, stud_gEmail, stud_gCell)

1NF

Account(userID,userType,password , confirmed, confirmCode, id,admin_id, admin_idnumber , admin_ fname , admin_lname , admin_email, admin_cell, admin_image,messageid,userid,email ,message,date ,incidentID, reporterID,handlerID,categoryID, typeID, handlerName, description,image,coordinates, incidentDate, incidentStatus,report, reportStatus,categoryID,name,description, typeID,categoryID,name,description,locationID,locationName,coordinates,notificationID,senderID, touserID, subject, message, date,id,so_number, so_fname,so_lname, so_idnumber, so_email, so_cell, so_ image, studentNumber, stud_idNumber, stud_fName, stud_lName, stud_cell, stud_email, stud_image, stud_gFName, stud_gLName, stud_gEmail, stud_gCell)

2NF

Account (userID,userType,password , confirmed, confirmCode)

Administrative_Personnel(admin_id, admin_idnumber , admin_fname , admin_lname , admin_email,cell, admin_image)

Safety_Officer(id,so_number ,so_,fname, so_lname, so_idnumber, so_email, so_cell, image)

Student(studentNumber, stud_idNumber, stud_fName, stud_lName, stud_cell, stud_email, stud_image, stud_gFName, stud_gLName, stud_gEmail, stud_gCell)

Incident (incidentID, typeID reporterID,handlerID,categoryID, handlerName, description,image,coordinates, incidentDate, incidentStatus,report, reportStatus,

Category (categoryID,name,description,typeID,categoryID,name,description)

Location (locationID,locationName,coordinates)

Notification (notificationID,senderID, touserID, subject, message, date, messageid,userid,email)

Contact_us (messageid,userid,email ,message,date)

3NF

Account (userID,userType,password , confirmed, confirmCode)

Adminpersonnel (id,admin_id, admin_idnumber , admin_fname , admin_lname , admin_email,cell, admin_image)

Contact_us (messageid,userid,email ,message,date)

Incident (incidentID, reporterID,handlerID,categoryID, typeID, handlerName, description,image,coordinates, incidentDate, incidentStatus,report, reportStatus)

Incident_category (categoryID,name,description)

Incident_type (typeID,categoryID,name,description)

Location (locationID,locationName,coordinates)

Notification (notificationID,senderID, touserID, subject, message, date)

Safetyofficer (id,so_number,fname, so_lname, so_idnumber, so_email, so_cell, image)

Student (studentNumber, stud_idNumber, stud_fName, stud_lName, stud_cell, stud_email, stud_image, stud_gFName, stud_gLName, stud_gEmail, stud_gCell)

Manipulate your data

INSERT INTO `account` (`id`, `userID`, `userType`, `password`, `confirmed`, `confirmCode`) VALUES(46, 123456789, 'ADMIN_PERS', '9212', 1, 4885),(48, 151515151, 'SO', '6197', 1, 4533),

(52, 212121212, 'STUD', '14', 1, 661),(53, 212121213, 'STUD', '15', 0, 8701),(54, 213159251, 'STUD', '12', 0, 2245),(55, 213159269, 'STUD', '248', 1, 29210),(56, 211082178, 'STUD', '1112K@mo', 1, 21770);

INSERT INTO `admin` (`adminid`, `name`, `password`, `image`) VALUES

('123456', 'peter', '15', '');

INSERT INTO `adminpersonnel` (`id`, `adminid`, `idnumber`, `fname`, `lname`, `email`, `cell`, `image`) VALUES

(1, 123456789, '2147483647', 'Tshegofatjo', 'Mamabolo', '213159259@tut4life.ac.za', '0790974928', '123456789.png');

INSERT INTO `contact_us` (`messageid`, `userid`, `email`, `message`, `date`) VALUES

(3, 213159269, '213159259@tut4life.ac.za', 'I am having trouble to login.', '2016-02-6'),

(6, 213159269, '213159259@tut4life.ac.za', 'I my issue was resolved', '2016-02-5'),

(16, 123456789, '213159259@tut4life.ac.za', 'Testing cookie', '2016-02-7'),

(17, 213159269, 'rakomatp94@gmail.com', 'test date', '2016-02-10');

INSERT INTO `incident` (`incidentID`, `reporterID`, `handlerID`, `categoryID`, `typeID`, `handlerName`, `description`, `image`, `coordinates`, `incidentDate`, `incidentStatus`, `report`, `reportStatus`, `locationid`, `locationdesc`, `missingStudId`) VALUES

(1, 213159269, 151515151, 6, 20, 'James Kobe', 'I saw a student stealing a rubbish bin.', '', '', '2016-01-29', 'closed', 'this part is working', 'draft', '2', 'Next to stairs', ''),

(2, 213159269, 151515151, 1, 6, 'James Kobe', 'Student stole my purse, and phone.', '', '', '2016-01-29', 'open', 'okay yes', 'draft', '1', 'Next to Lab 138', ''),

(3, 213159269, 151515151, 1, 6, 'James Kobe', 'Student stole my purse, and phone.', '', '', '2016-01-29', 'open', 'jhjgvvj', 'draft', '1', 'Next to Lab 138', '');

INSERT INTO `incident_category` (`categoryID`, `name`, `description`) VALUES

(1, 'CONTACT CRIMES (CRIMES AGAINST THE PERSON)', ''),(2, 'CONTACT-RELATED CRIMES', ''),(3, ' Health Hazard', ''),(4, 'Missing Person', ''),(5, 'OTHER CRIMES CATEGORIES', ''),(6, 'PROPERTY-RELATED CRIMES', ''),

(7, 'Emergency', 'It is an incident that is send by Student through the panic, Student does not provide incident description.');

INSERT INTO `incident_type` (`typeID`, `categoryID`, `name`, `description`) VALUES

(1, 1, 'Murder', ''),(2, 1, 'Sexual Offences', ''),(3, 1, 'Attempted murder', ''),(4, 1, 'Assault with the intent to inflict grievous bodily harm', ''),

INSERT INTO `location` (`locationID`, `locationName`, `coordinates`) VALUES

(1, 'Buliding 10', ''),(2, 'Building 20', '');

INSERT INTO `notification` (`notificationID`, `senderID`, `subject`, `message`, `date`) VALUES

(1, '123456789', 'Testing notification', 'Notification seems to be working', '2016-02-06'),

(2, '123456789', 'Testing notification', 'Notification works', '2016-02-4');

INSERT INTO `safetyofficer` (`id`, `sonumber`, `fname`, `lname`, `idnumber`, `email`, `cell`, `image`) VALUES

(1, 151515151, 'James', 'Kobe', '9408225818083', 'k@l.com', '0123456789', '');

INSERT INTO `student` (`studentNumber`, `idNumber`, `fName`, `lName`, `cell`, `email`, `image`, `gFName`, `gLName`, `gEmail`, `gCell`) VALUES

(212121212, '2147483647', 'Tshegofatjo', 'Rakoma', '0790974928',
'213159259@tut4life.ac.za', '212121212.png', 'po', 'sg', 'rakomatp@gmail.com',
'0790974928');

Manage transaction

mysql_query("INSERT INTO account (userid,usertype,password,confirmed,confirmcode) VALUES ('$studentNum','STUD','$pass1','0','$code')");

mysql_query("INSERT INTO student()
VALUES('$studentNum','$idNumber','$fName','$lName','$cell','$email','$image_db_name','$gFName','$gLName','$gEmail','$gCell')");

mysql_query("INSERT INTO account () VALUES ('','$studentNum','SO','$pass1','0','$code')");

mysql_query("INSERT INTO safetyofficer()
VALUES('','$studentNum','$fName','$lName','$idNumber','$email','$cell','$image')");

$query2 ="SELECT * FROM incident WHERE incidentstatus='open' ORDER BY incidentID ASC";

$query ="select it.name AS 'Incident Type',count(it.name) AS 'Nmber of Reported Incidents' from incident i,incident_type it where it.typeid=i.typeid group by it.typeid,i.typeid,it.name order by it.name ASC";

$query ="SELECT locationid,count(locationid) FROM `incident` group by locationid order by count(locationid) desc";

Phase 5(Final Project Deliverable)

Documentation on Test cases and Test Plan

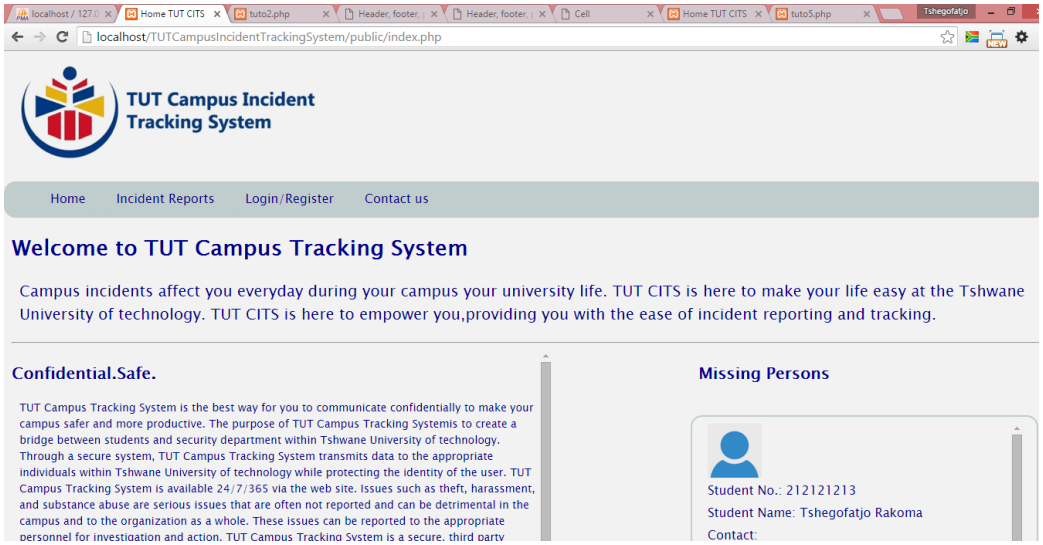
Reports

1. The system produces a report that shows a summary of incident by incident type category.
2. The system produces a report that shows the total number of incidents that were reported.
3. The system produces a report that shows the number of incidents reported at a location, this a statistic that shows top 3 dangerous area around campus where incidents might occur.
4. The system produces a detailed report that shows the number of incidents reported per category, number of open incidents, closed incidents, draft reports and finalised reports.

User manual

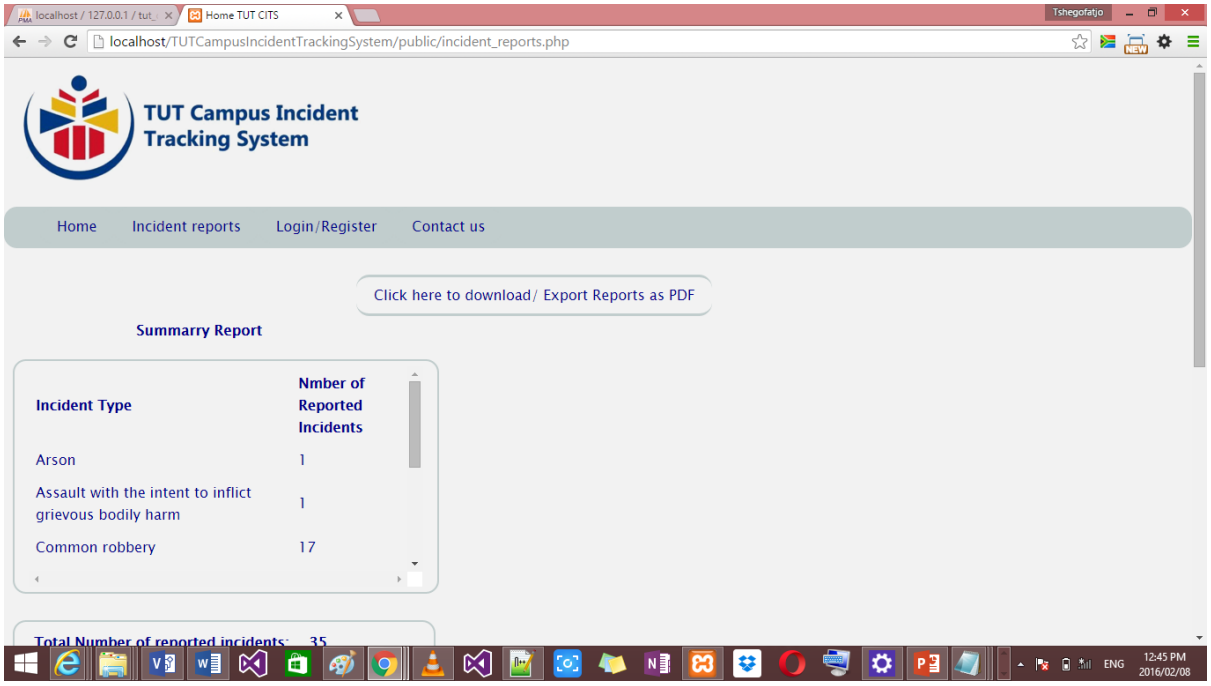
TUT Campus Incident Tracking System

Home Page



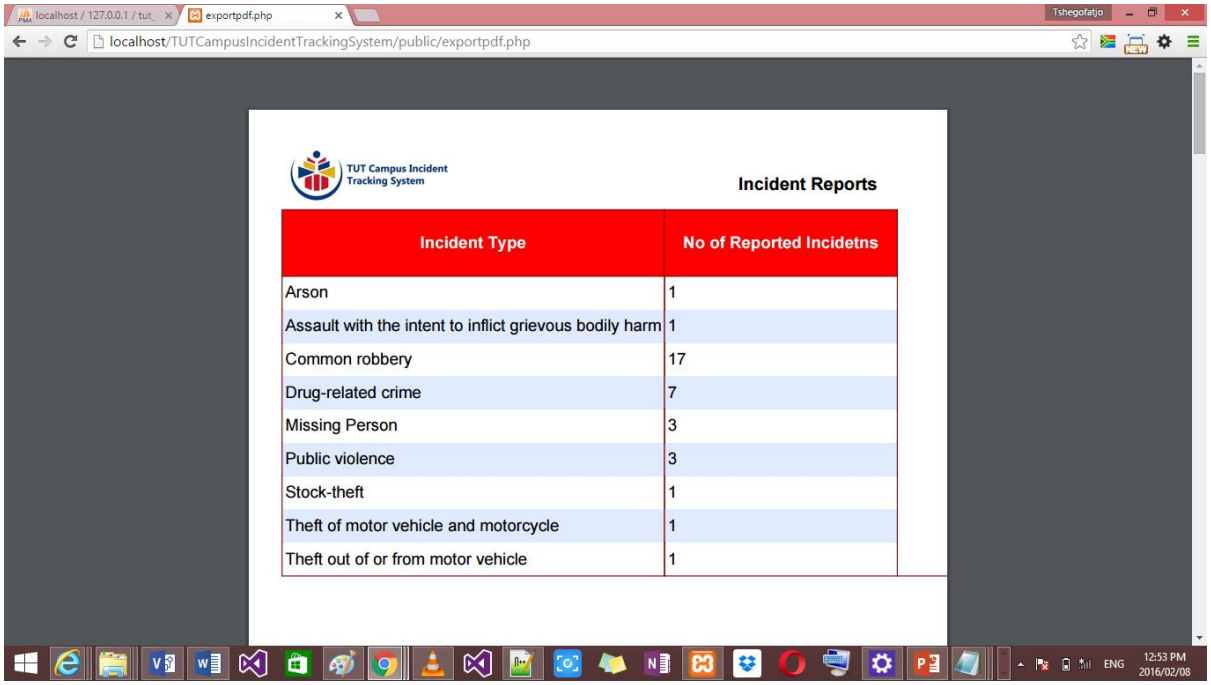
Incident Reports

1. Click on incident reports button to navigate to the incidents reports page.
2. Click on ‘Click here to download/Export reports as PDF Button’ to export reports.

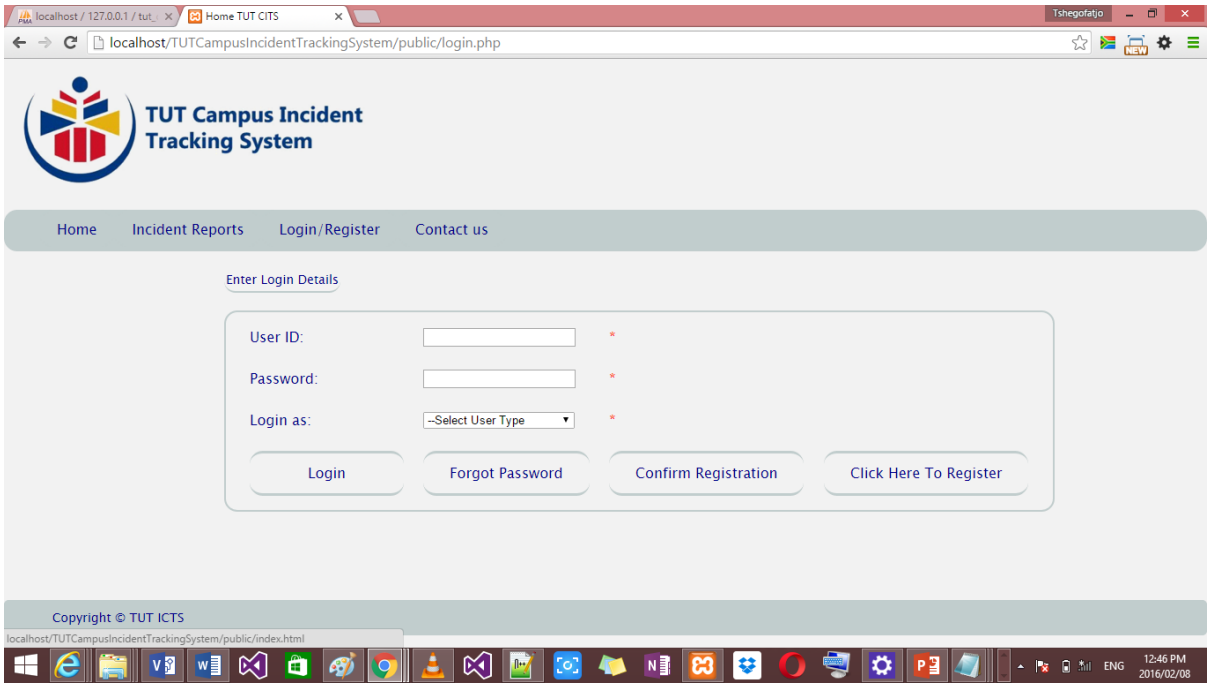


Export Reports

3. Reports are now ready to be saved as pdf.



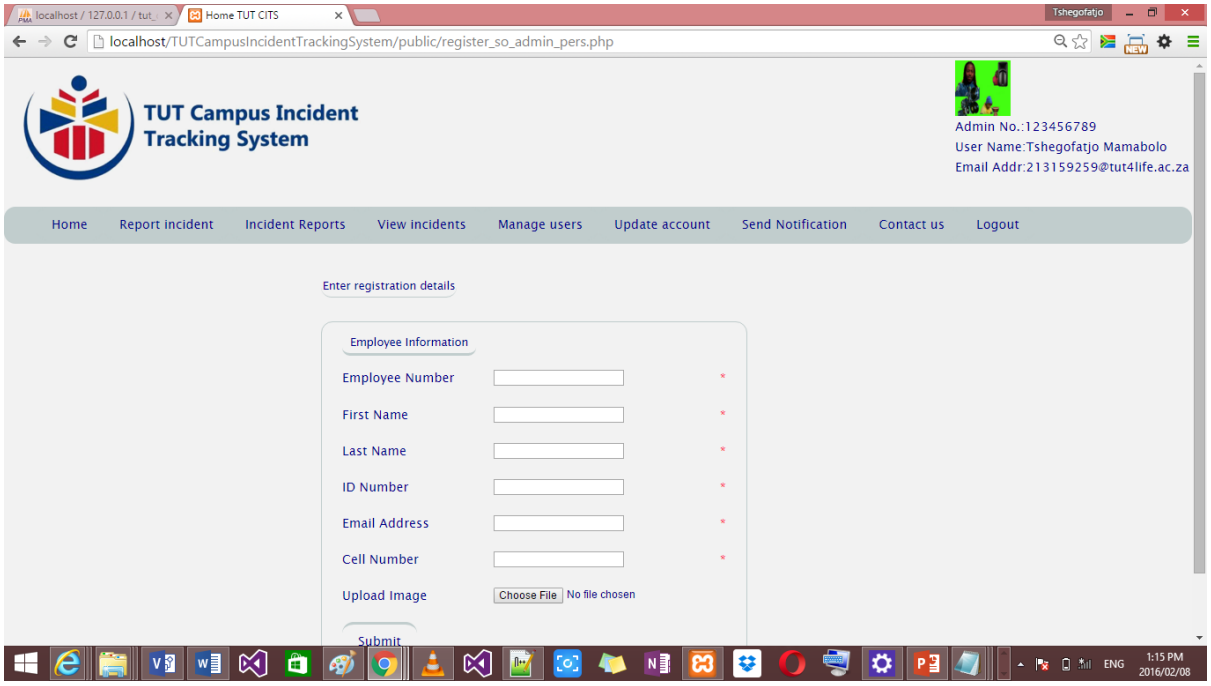
Login Page



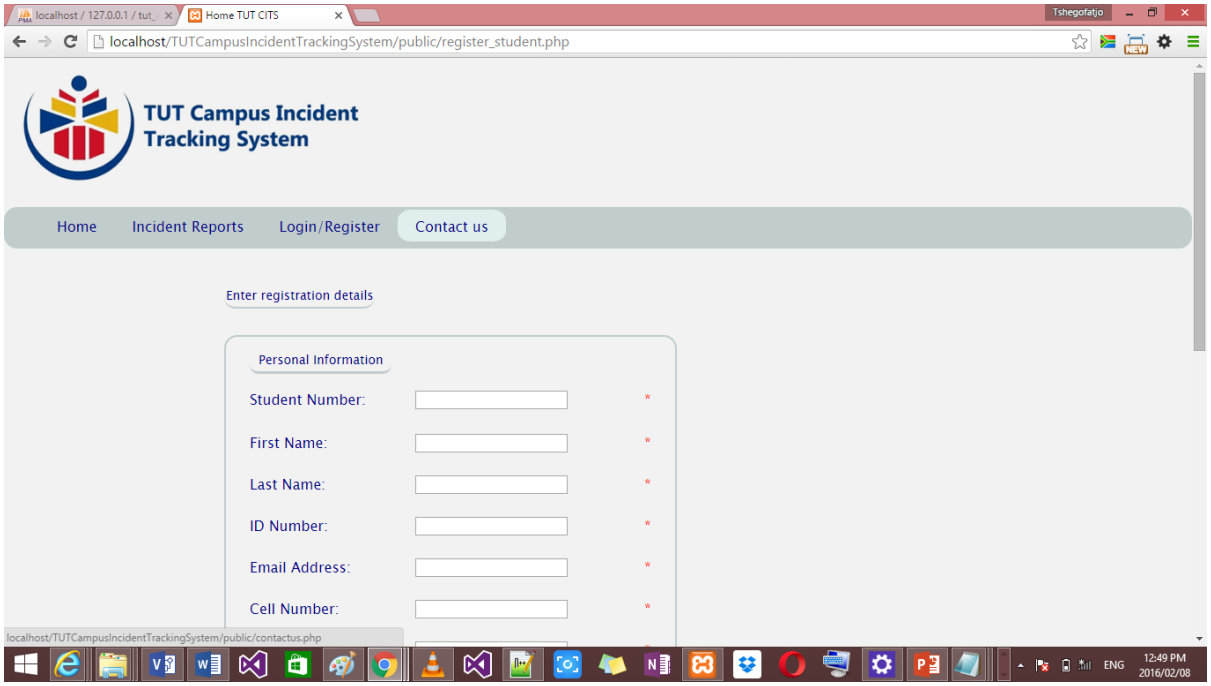
1. Enter valid user id (Student number/Safety officer Number/Admin ID).
2. Enter valid password.
3. Click on Login button.

Student Registration Page

1. Enter valid details in all fields.
2. Click on Submit button.
3. You will receive e-mail with account activation details and login details.

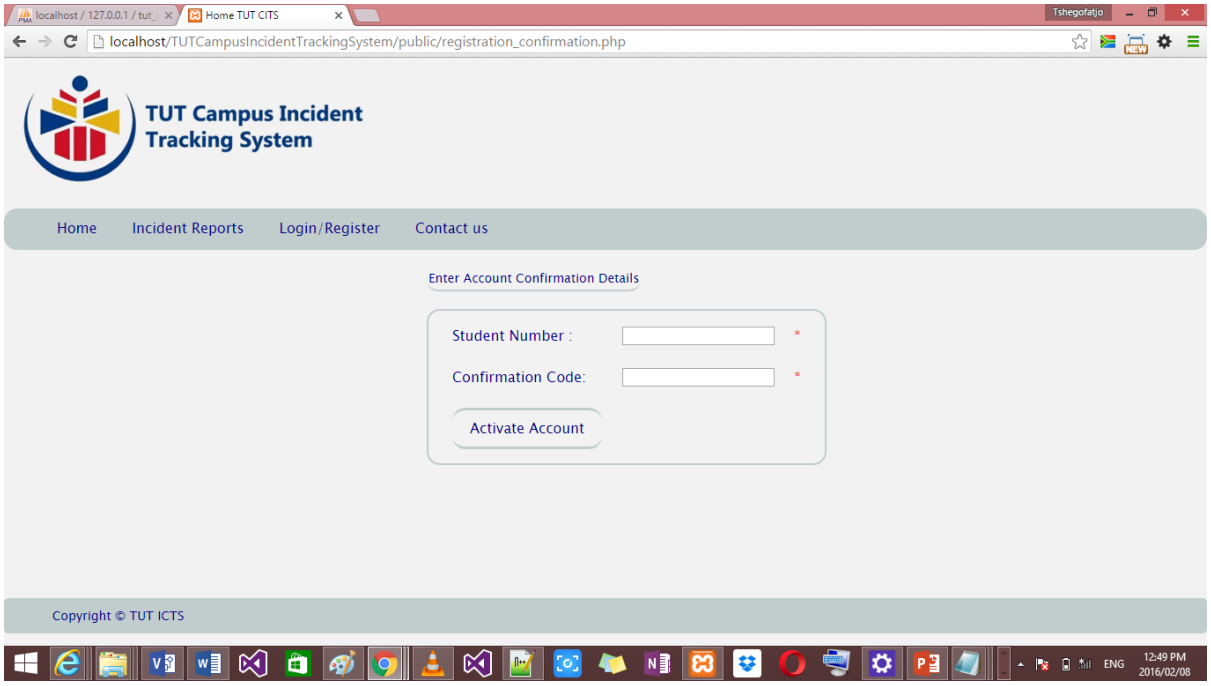


Safety officer/Administrative personnel Registration Page



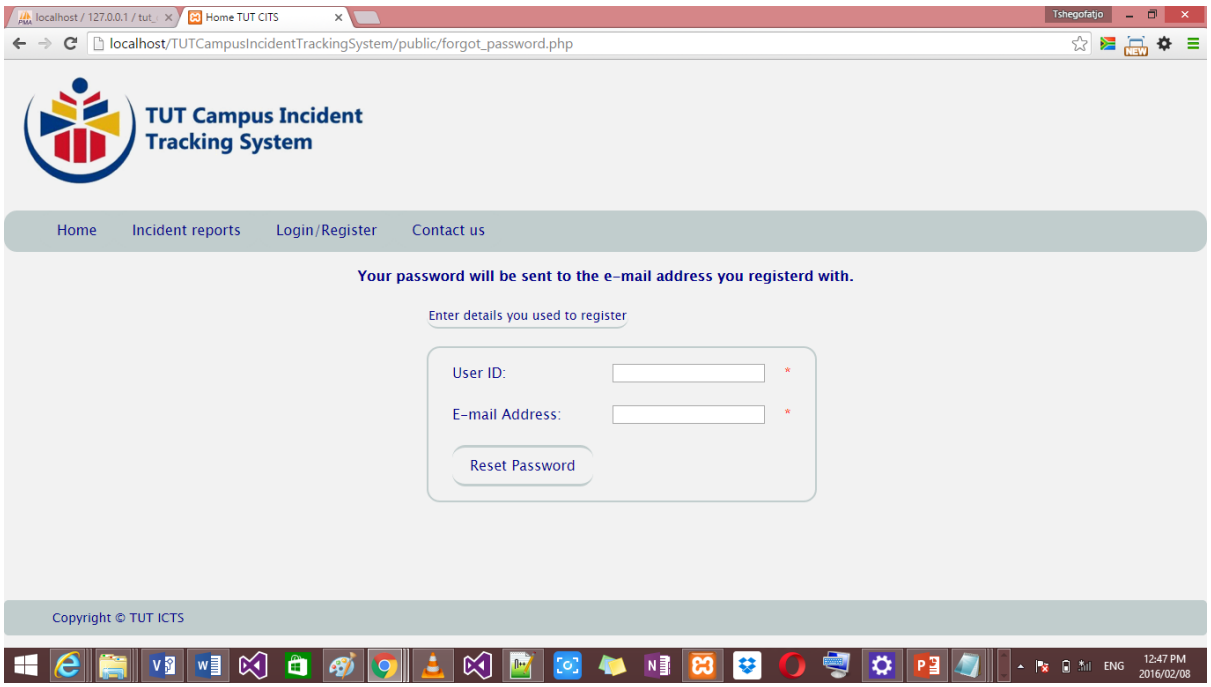
Activate Account Page

1. Enter user id (may be Student Number/ Safety officer number or Admin Number) and confirmation code.
2. Click on Activate Account. You are now ready to log in to the system.



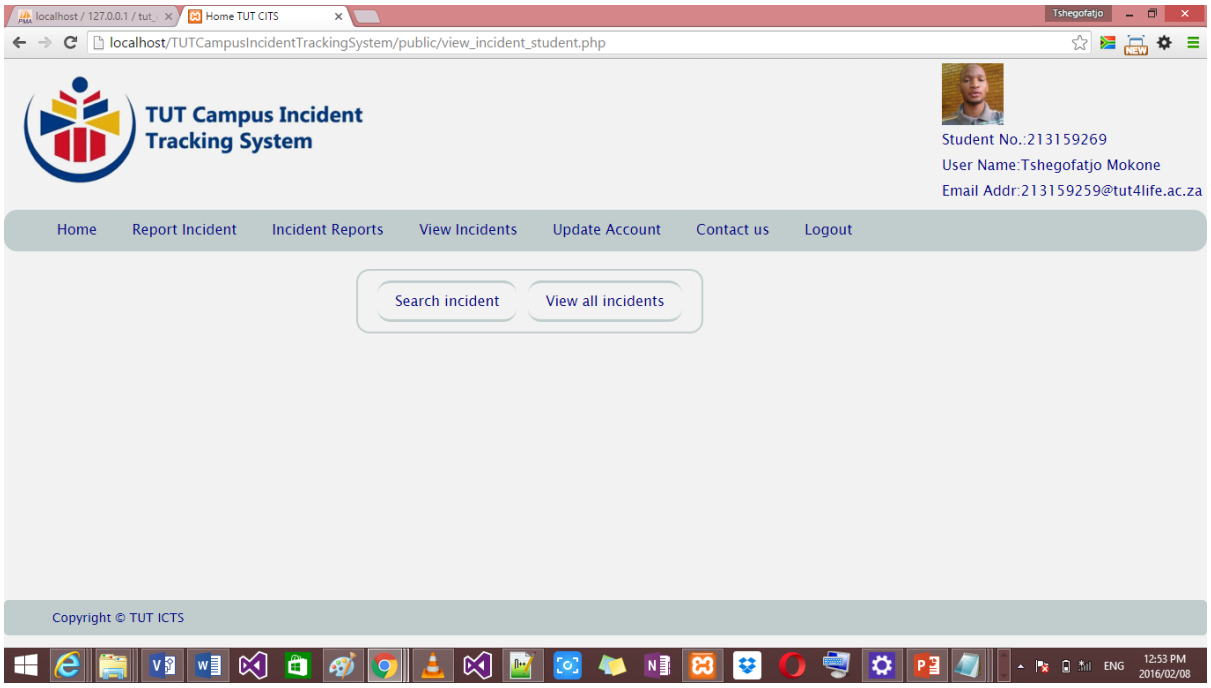
Forgot Password Page

1. To recover your password enter your user id and email address you used to register.
2. Click on reset/ recover password button. You will receive email with recovered or new password if not student.

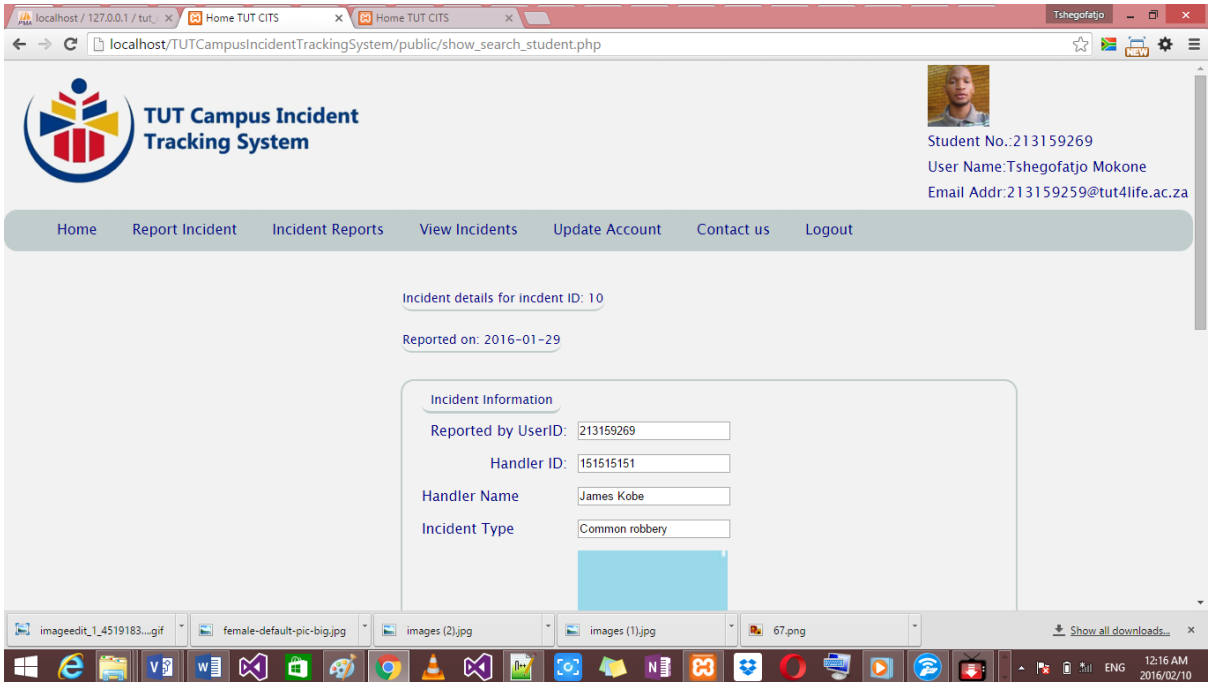


View incident Page

1. Click on Search incident to search incident by incident id.
2. Click on View all incidents to view all incidents reported to you/by you.

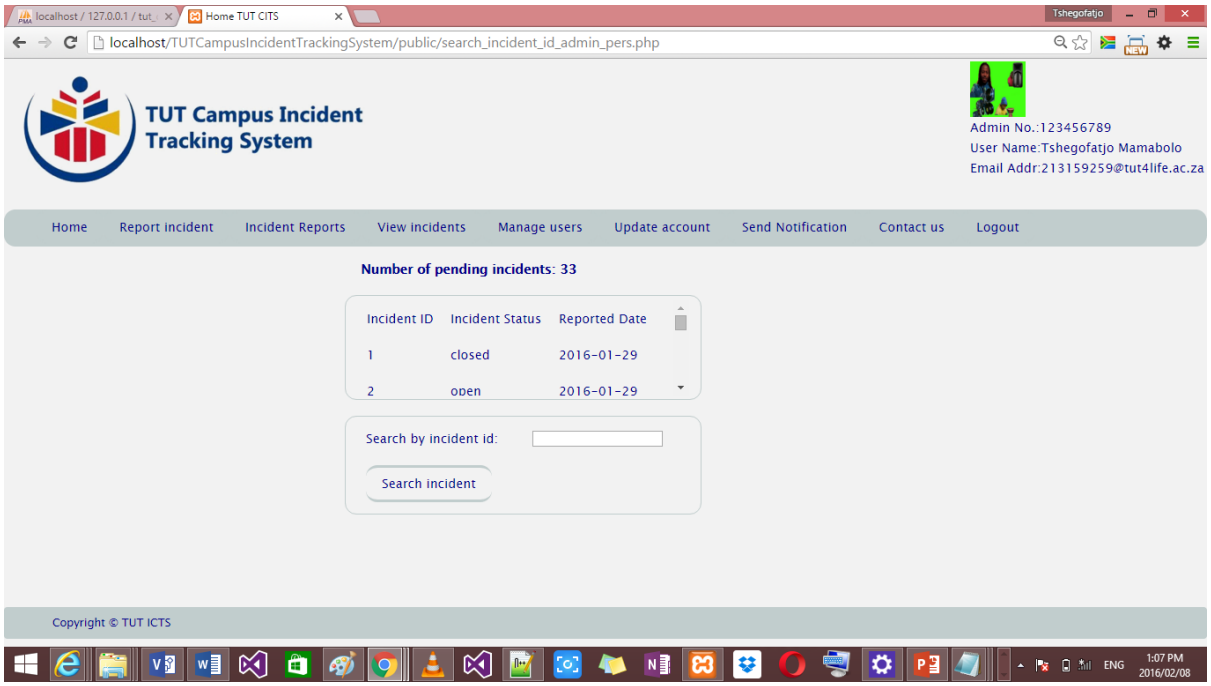


Searched incident Page



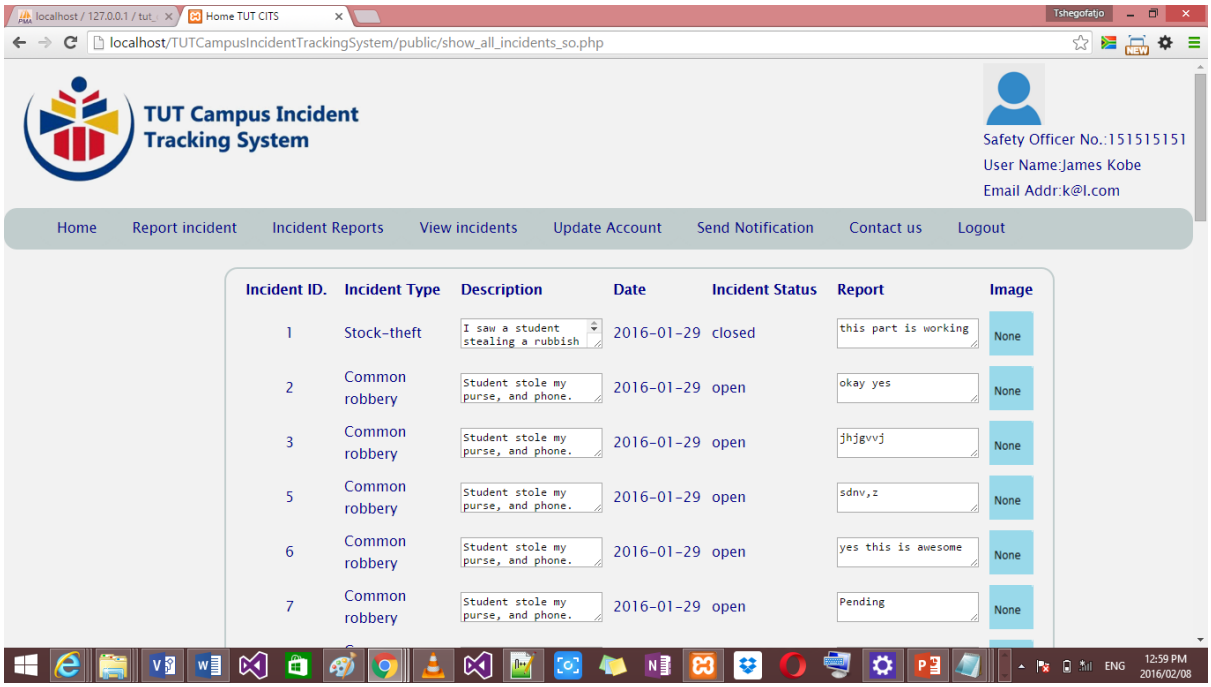
Search incident Page

1. You can view/update/delete incident depending on your access level.



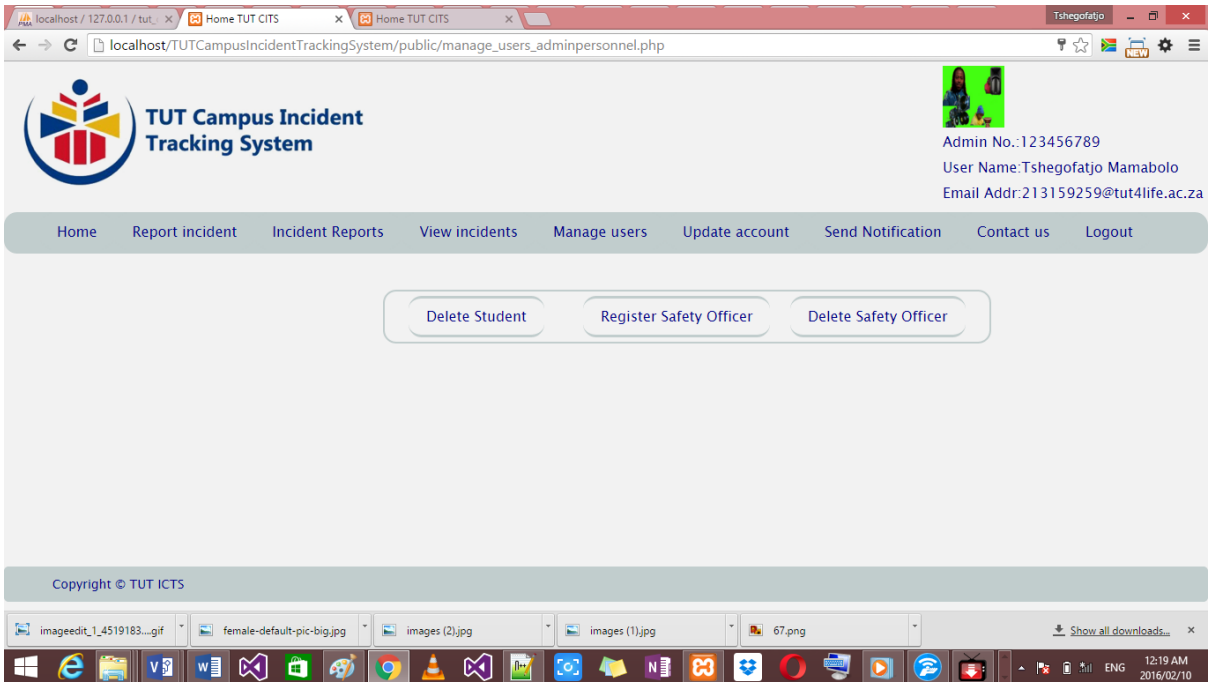
View all incidents Page

1. All incidents reported by you/to you or all are displayed depending on access level.



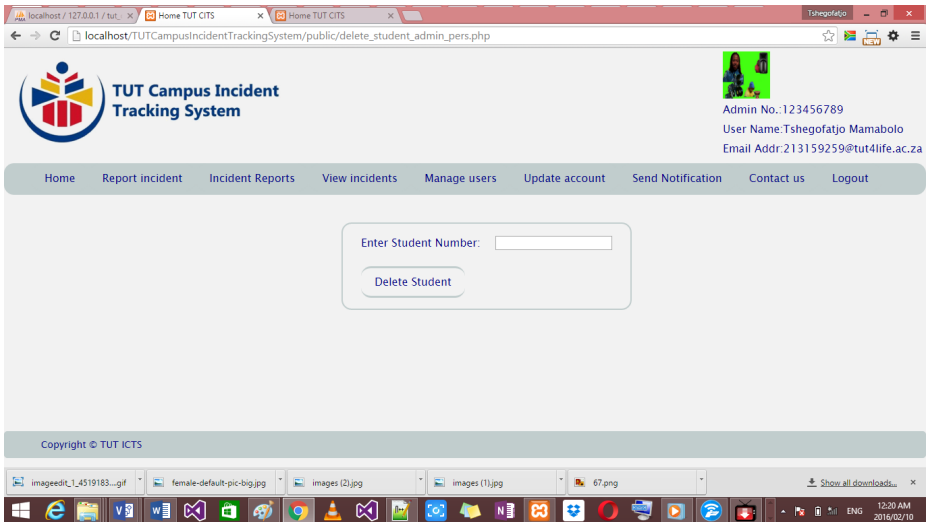
Manage Users Page

1. Click on Delete Student to delete student with given student number.
2. Click on register Safety officer
3. Click on Delete Safety officer to delete student with given Safety officer number.



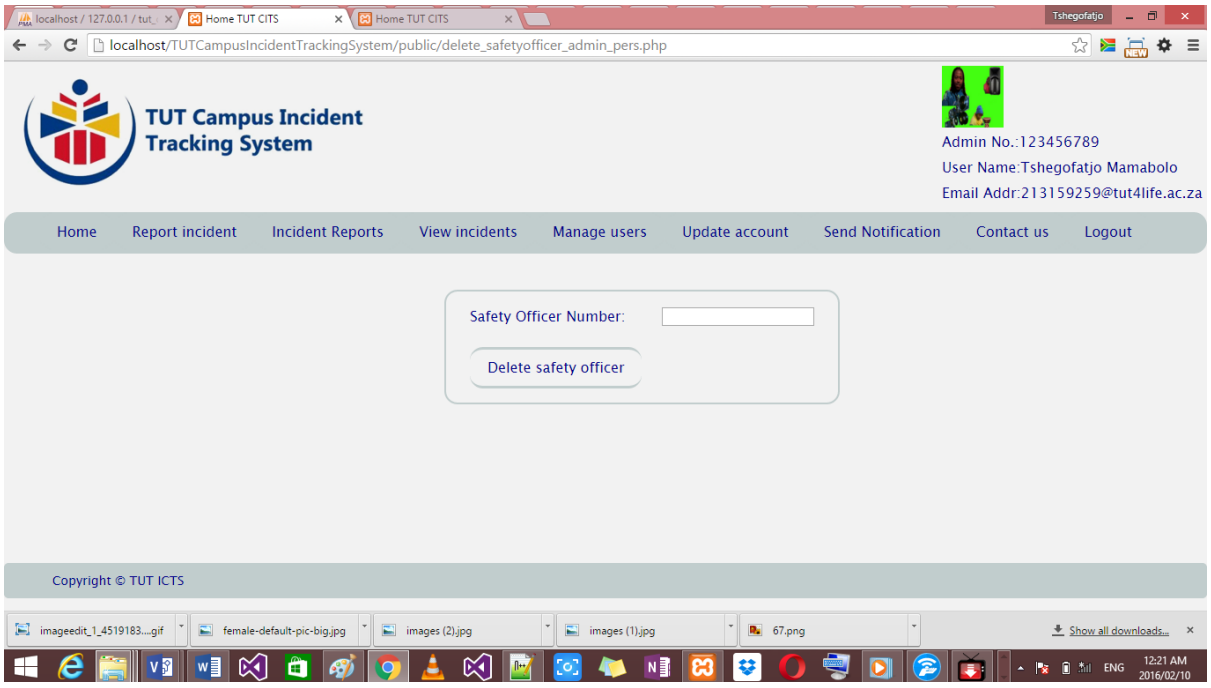
Delete Student Page

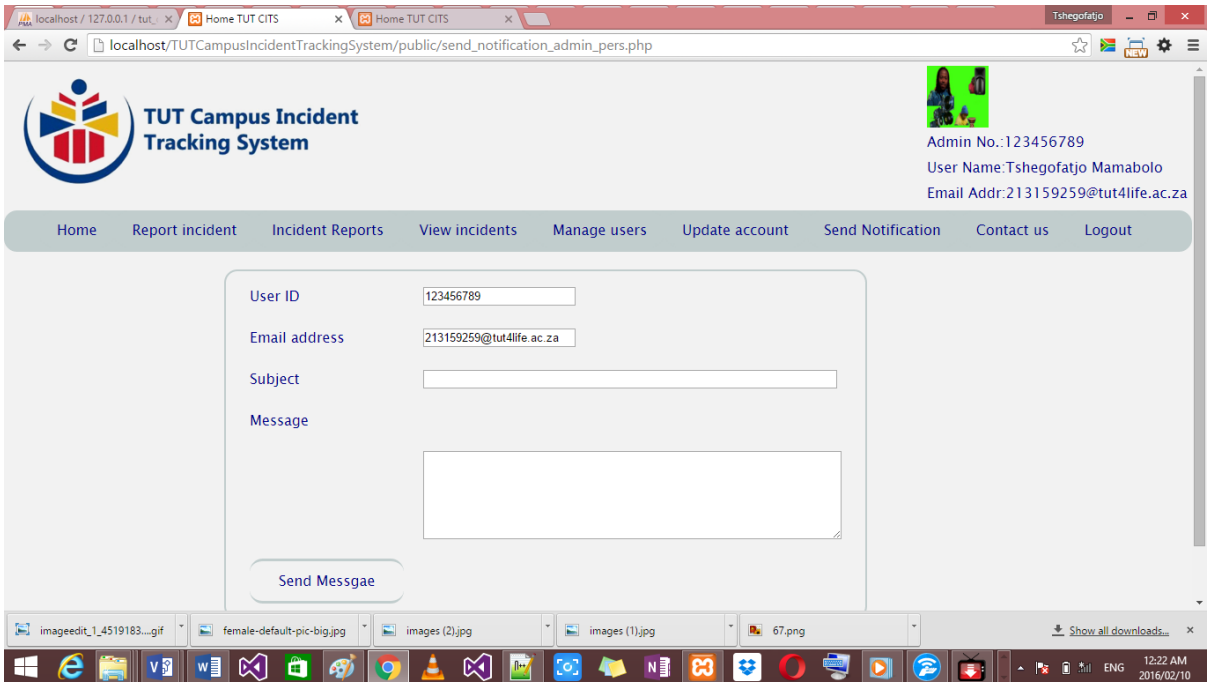
1. Enter student number of student you want to delete and click on delete student.



Delete Safety Officer Page

1. Enter Safety Officer number of Safety Officer you want to delete and click on Safety Officer.



Send notification Page

1. Enter subject and message and click on Send message button.