

# **Road Repair and Tracking Software Test Suite Design**

Satyesh Mundra – 14CS10058  
Prabal Singh – 14CS30023

## 1.Objective:

The primary objective of testing this application is to assure that the system meets the full requirements, including quality requirements (AKA : Non-Functional Requirements) and fits metrics for each quality requirement and satisfies the use case scenarios and maintain the quality of the product.

The secondary objective of testing this application is to identify and expose all issues and associated risks, communicate all known issues and ensures that all issues are addressed in an appropriate matter before release.

## 2.Testing Methodology:

This document incorporates the UNIT TESTING method i.e. both the BLACK-BOX and the WHITEBOX approach towards Test Suite Design of the software 'RRTS'. In essence, we must therefore test each class in isolation, and each functionality in isolation using white box and black box techniques. This document shows the use cases, their description, test inputs and the Black-Box and the White-Box outputs. If there does not exist both type of outputs, only the appropriate output is displayed.

Regression Testing was done during coding the software by re-running existing tests against the modified code to determine whether the changes break anything that worked prior to the change and by writing new tests where necessary. Adequate coverage without wasting time was the primary consideration during conducting regression tests.

## 3. Features to be tested:

### Clerk:

- User Sign In
- Enter Complaints
- Print List
- View Complaints

### Supervisor:

- User Sign In
- Print Complaints List
- Receive Work
- Enter Complaint after Supervision
- Update Phase of the work

### **Administrator:**

- User Sign In
- Print List
- Update materials, machines and manpower.
- See Possible Work Allotment
- Reschedule Work
- Allocate Work

### **Mayor:**

- User Sign In
- Print Past Assignments
- Print Undergoing Assignments
- Print Future Assignments
- View Complaints Information
- View Materials Statistics
- View Repair Type Statistics

## **4. Features not to be tested:**

The Print Complaint List feature cannot be tested as printer is not available.

## **5. Information Stored Initially:**

### **Clerk:**

- Username: clerk
- Password: clerk123

### **Supervisor:**

- Username: supervisor
- Password: supervisor123

### **Administrator:**

- Username: admin
- Password: admin123

### **Mayor:**

- Username: mayor
- Password: mayor123

## Database:

- Username: root
- Password: saty

A database – named 'rrts' is created and hosted on a central server in which various tables are created for storage of data. The database has also been provided as a sql file with the software. So the user doesn't have to create the database himself. The various tables are all initially empty and get filled with time, as the application gets used by different users.

The Software has been tested previously many times hence there are various entries already present in the database table. Therefore before this phase of testing, the various tables are:

- Complaints – Stores the complaints of the users
- Work – Contains the data of requirements and status of various complaints
- Resources – Contains the data of various resources and their availability.

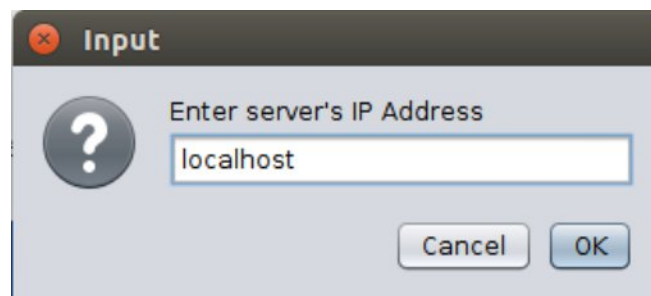
## 6. Test Cases:

### TEST #1:

**Description:** Enter IP Address

**Input:** User enters the IP address of the central server.

**Output:** Login page opens



## TEST #2:

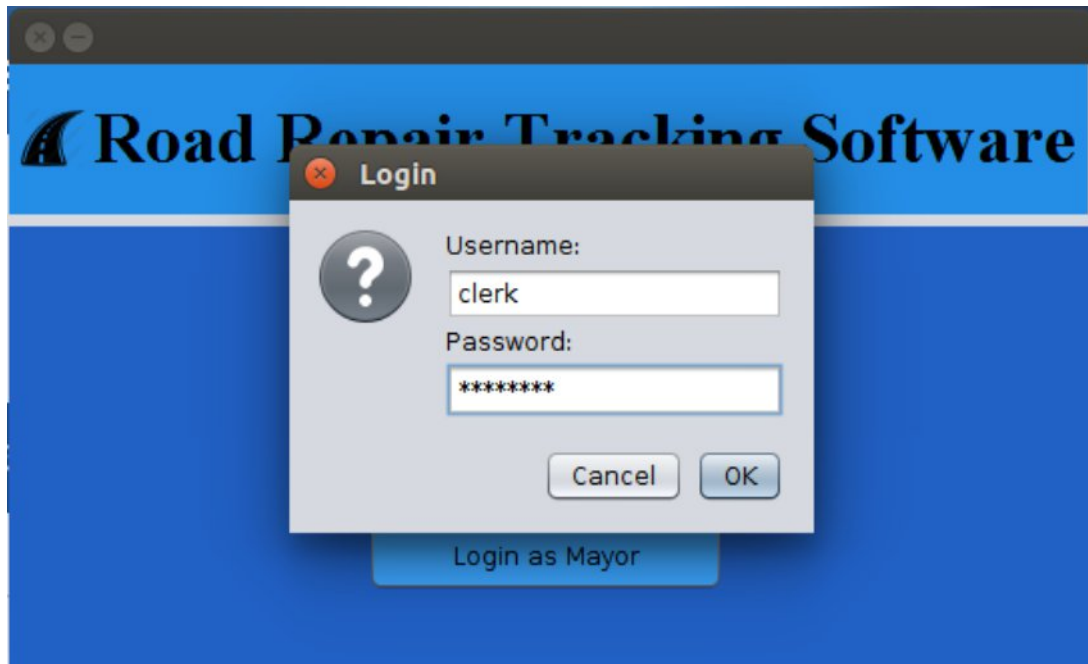
**Description:** User Sign In

**Input:** The user type, user name and password of the person using the software.

User type: clerk/supervisor/admin/mayor

Password: clerk123/supervisor123/administrator123/mayor123

**Output:** Home page of the user opens.



### TEST #3:

**Description:** User Sign In

**Input:** The user type, user name and password of the person using the software

User type: Clerk/Supervisor/Administrator/Mayor

Password: anything other than the correct password for the selected user types.

**Output:** Home page of the user fails to open and a message saying "Incorrect Password" is displayed.

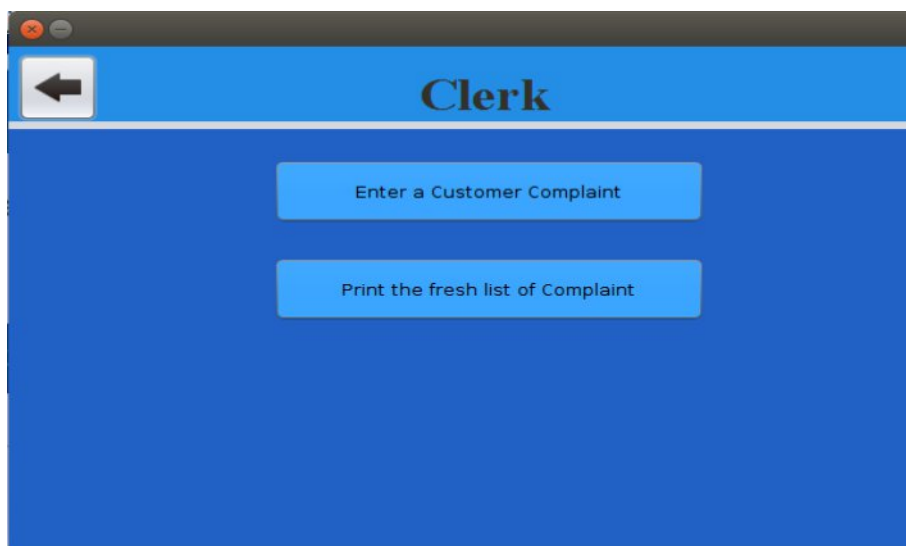


### Test #4:

**Description:** Display Clerk Menu.

**Input:** The clerk logs in to do his work.

**Output:** The Menu of the clerk opens.

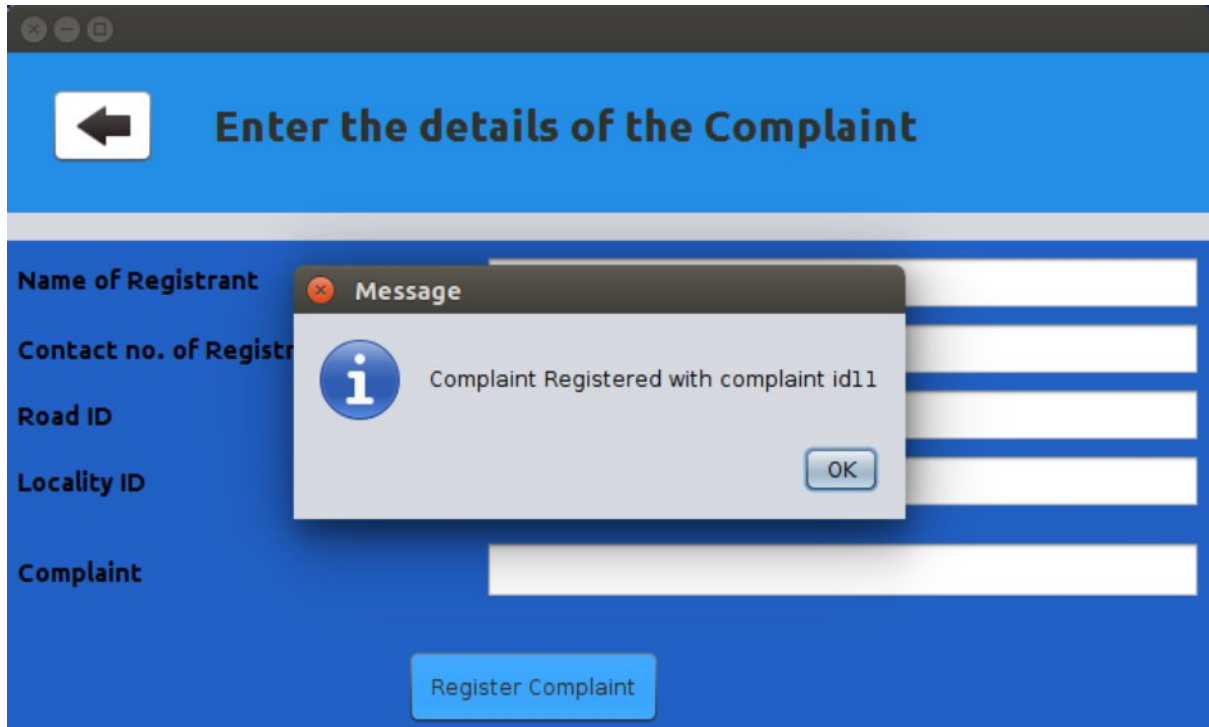


### Test #5:

**Description:** Enter Complaints

**Input:** Name, address, location id, phone, email, description (of the complaint) of the person who files the complaint.

**Output:** A confirmatory message saying "complaint registered with id \_". The data entered is fed into the database of the clerk.



The screenshot displays a web application interface for registering complaints. The main heading is "Enter the details of the Complaint" with a back arrow icon. The form includes fields for "Name of Registrant", "Contact no. of Registrant", "Road ID", "Locality ID", and "Complaint". A "Register Complaint" button is at the bottom. A modal message box is overlaid, stating "Complaint Registered with complaint id11" with an "OK" button.

← Enter the details of the Complaint

Name of Registrant

Contact no. of Registrant

Road ID

Locality ID

Complaint

Register Complaint

**Message**

**i** Complaint Registered with complaint id11

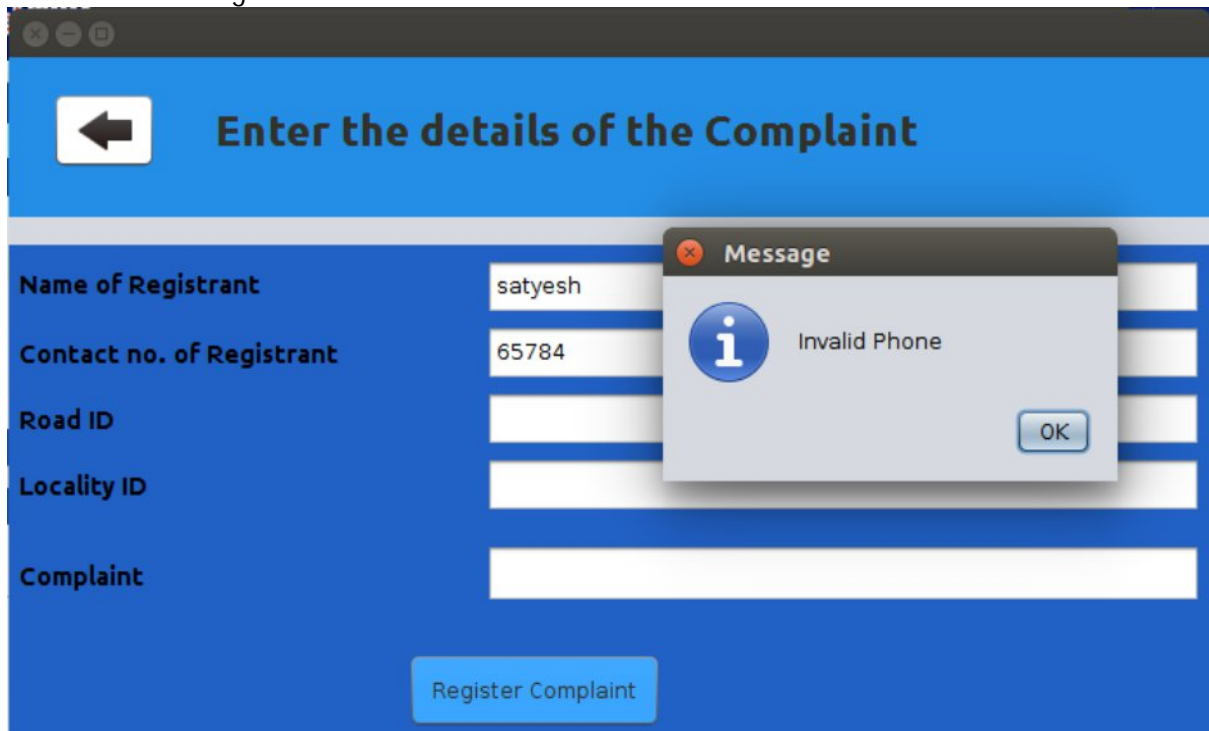
OK

## Test #6:

**Description:** Enter Complaints

**Input:** Name, address, location id, phone, email, description (of the complaint) of the person who files the complaint with some field wrong.

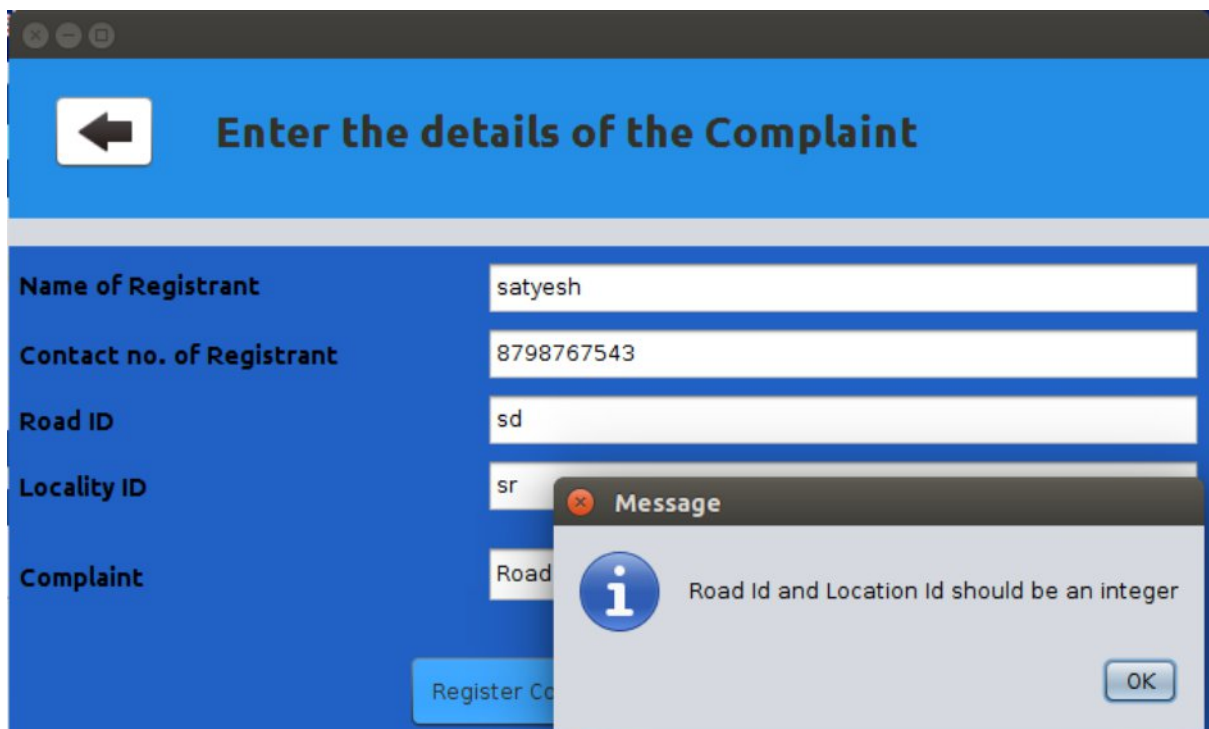
**Output:** An error message showing "Invalid Phone" if mobile number is wrong is shown or if road id and locality id is not an integer then an error message is shown "Road and locality id should be an integer".



The screenshot shows a web application window titled "Enter the details of the Complaint". The form contains the following fields:

- Name of Registrant: satyesh
- Contact no. of Registrant: 65784
- Road ID: (empty)
- Locality ID: (empty)
- Complaint: (empty)

A blue "Register Complaint" button is at the bottom. A modal message box is displayed over the form, titled "Message", with an information icon and the text "Invalid Phone". An "OK" button is in the bottom right of the message box.



The screenshot shows the same web application window. The form fields are now filled with:

- Name of Registrant: satyesh
- Contact no. of Registrant: 8798767543
- Road ID: sd
- Locality ID: sr
- Complaint: Road

The "Register Complaint" button is partially visible. A modal message box is displayed, titled "Message", with an information icon and the text "Road Id and Location Id should be an integer". An "OK" button is in the bottom right of the message box.



## TEST #7:

**Description:** Get Complaint List.

**Input:** User Action (Button Click)

**Output:** Pop up asking the locality id of supervisor is shown and after entering correct locality id the complaints of that locality are shown in the table to supervisor.

The screenshot displays the SUPERVISOR interface. At the top, there is a blue header with a back arrow and the title 'SUPERVISOR'. Below the header are two buttons: 'GET COMPLAINT LIST' and 'UPDATE COMPLAINT STATUS'. A table shows the complaint list with columns: Complaint ID, Road ID, Status, and Complaint. The table contains two rows: (1, 1, Completed, saty) and (2, 2, Completed, bridge). Below the table, there is a 'Locality' pop-up dialog with a question mark icon and a text input field containing '4'. The dialog has 'Cancel' and 'OK' buttons. The background interface includes sections for 'Complaint ID', 'Type of work', 'Priority', 'Machines', 'Raw Materials', 'Days required to complete the work', and 'Workers Required', each with input fields. At the bottom, there is a button labeled 'Enter and schedule work'.

Complaint ID	Road ID	Status	Complaint
1	1	Completed	saty
2	2	Completed	bridge

**Locality**

Locality ID: 4

Cancel OK

**Complaint ID**  **Type of work**  **Priority**

**Machines** Road Rollers  Asphalt Drum Mix Plant   
Road Marking Machines  Chips Spreader

**Raw Materials** Sand bags  Asphalt  Cement bags   
Brick Trucks  Concrete  Limestone

**Days required to complete the work**  **Workers Required**

Enter and schedule work

## TEST #8:

**Description:** Enter complaint requirements

**Input:** User fills the details of the material required for complaint work.

**Output:** Selecting a component from table initialises the fields below and now the user can enter the details of raw materials and machines required for work and then enter that into database and the complaint status is changed to reviewed.

The screenshot displays the SUPERVISOR web application interface. At the top, there is a navigation bar with a back arrow and the title "SUPERVISOR". Below this, there are two buttons: "GET COMPLAINT LIST" and "UPDATE COMPLAINT STATUS".

A message dialog box is open in the center, titled "Message", with an information icon and the text "Resources Updated". It has an "OK" button.

Below the message dialog, there is a table showing complaint details:

Complaint ID	Road ID	Status
9	5	Complete
10	6	Complete
11	3	Pending

Below the table, there is a form for entering complaint details. The form includes the following fields:

- Complaint ID:** 11
- Type of work:** Bridge Repair (dropdown menu)
- Priority:** Low (dropdown menu)
- Machines:**
  - Road Rollers: 0
  - Concrete Road Pavers: 0
  - Asphalt Drum Mix Plant: 0
  - Road Marking Machines: 2
  - Chips Spreader: 02
  - JCB: 01
- Raw Materials:**
  - Sand bags: 02
  - Asphalt: 01
  - Cement bags: 020
  - Brick Trucks: 01
  - Concrete: 01
  - Limestone: 0
- Days required to complete the work:** 02
- Workers Required:** 020

At the bottom of the form, there is a button labeled "Enter and schedule work".

### TEST #9:

**Description:** Enter complaint requirements

**Input:** If user left any input blank, User Action (Button Click)

**Output:** An error message is shown saying "All fields must be filled".

The screenshot shows the 'SUPERVISOR' interface. At the top, there are two buttons: 'GET COMPLAINT LIST' and 'UPDATE COMPLAINT STATUS'. Below these is a table with the following data:

Complaint ID	Road ID	Status	Complaint
9	5	Completed	boo
10	6	Completed	road
11	3	Pending	Road Block

Below the table is a form for entering complaint requirements. The form includes fields for 'Complaint ID' (11), 'Type of work' (Br), 'Priority' (Low), 'Machines' (Road Rollers, Road Marking Machines, Chips Spreader, Asphalt Drum Mix Plant, JCB), 'Raw Materials' (Sand bags, Brick Trucks, Asphalt, Concrete, Cement bags, Limestone), 'Days required to complete the work' (02), and 'Workers Required' (020). A message box is displayed over the form, stating 'All fields must be filled'.

### TEST #10:

**Description:** Update Phase of the Work

**Input:** User Action (Button Click)

**Output:** A table is shown containing all complaints of the supervisor's locality id.

The screenshot shows the 'Update Complaint Status' interface. At the top, there is a button with a left arrow. Below it is a table with the following data:

Complaint ID	Road ID	Current Status
9	5	Completed
10	6	Completed
11	3	Reviewed

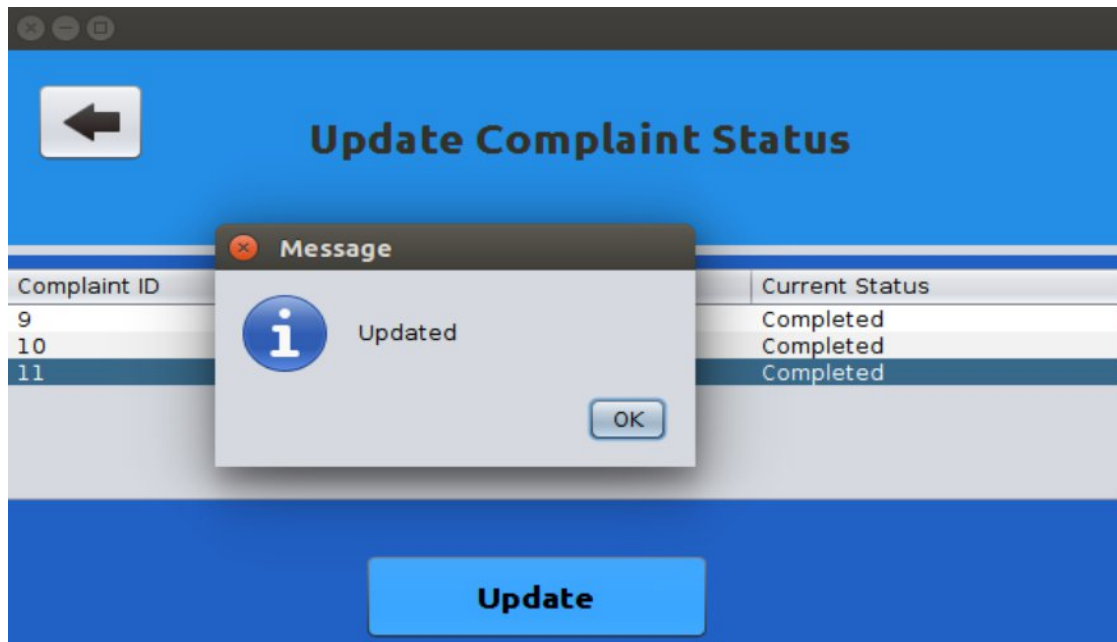
Below the table is a large blue button labeled 'Update'.

### TEST #11:

**Description:** Update Phase of the Work

**Input:** Select an entry from table, user action (Button Click)

**Output:** The status of that Allocated complaint is updated to complete.

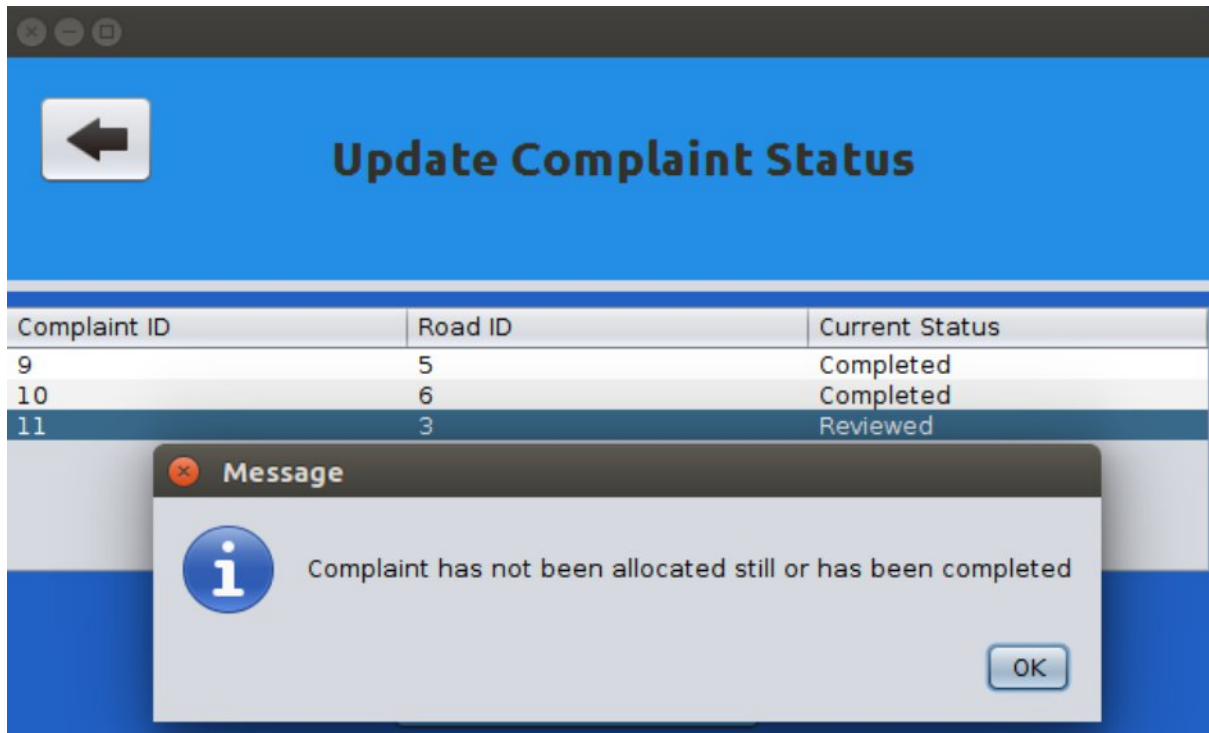


## TEST #12:

**Description:** Update Phase of the Work

**Input:** User Action (Button Click)

**Output:** If the selected complaint is not allocated yet or is completed then an error message is shown saying "Complaint has not been allocated yet or has been completed"

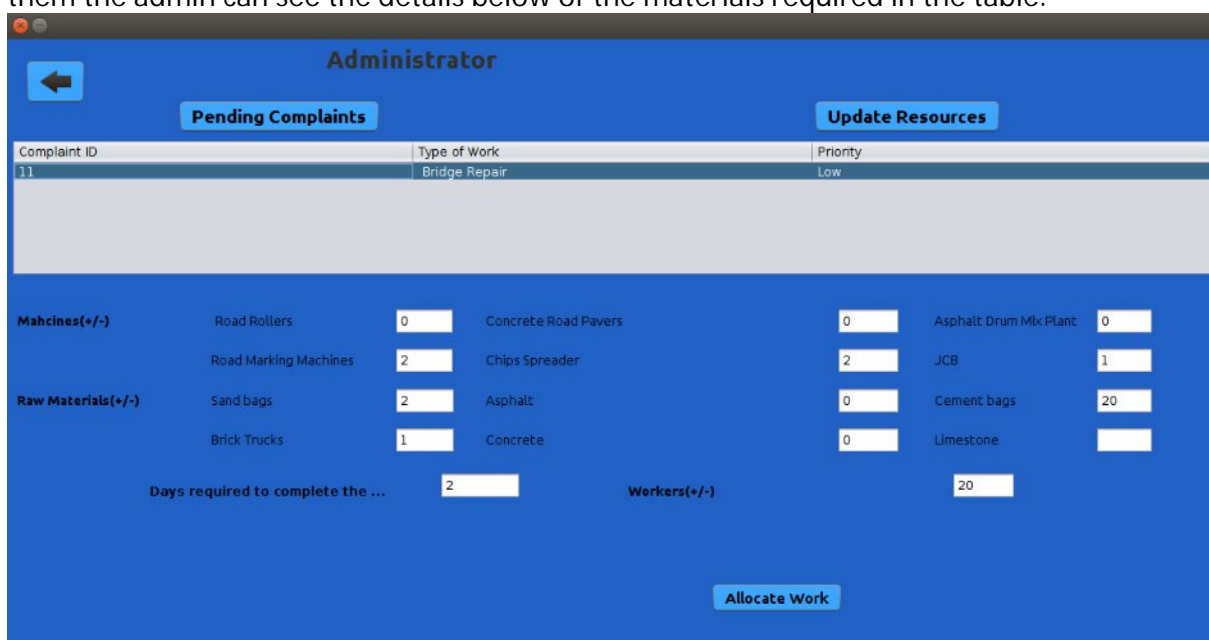


## TEST #13:

**Description:** Admin gets pending complaints.

**Input:** User Action (Button Click).

**Output:** Table gets filled with complaints which have to be allocated work and by clicking them the admin can see the details below of the materials required in the table.



## TEST #14:

**Description:** Allocate Work

**Input:** User Action (Button Click)

**Output:** All complaints are allocated work if resources are available in order of priority.

The screenshot shows the 'Administrator' interface with a 'Pending Complaints' table and resource allocation controls. A message box is displayed in the center.

Resource type	Total	Employed	Unemployed
11	Bridge Repair	Low	

**Machines(+/-)**

Road Rollers	0
Road Marking Machines	2

**Raw Materials(+/-)**

Sand bags	2
Brick Trucks	1

**Concrete**

0
---

**Workers(+/-)**

20
----

**Update Resources**

0	Asphalt Drum Mix Plant	0
2	JCB	1
0	Cement bags	20
0	Limestone	0

**Message**

Work Has been allocated to complaint ids:-,11

OK

**Allocate Work**

## TEST #15:

**Description:** Allocate Work

**Input:** User Action (Button Click)

**Output:** If resources are not available then an error message is shown saying "No work has been allocated".

The screenshot shows the 'Administrator' interface with a 'Pending Complaints' table and resource allocation controls. A message box is displayed in the center.

Complaint ID	Type of Work	Priority
11	Bridge Repair	Low

**Machines(+/-)**

Road Rollers	0
Road Marking Machines	2

**Raw Materials(+/-)**

Sand bags	2
Brick Trucks	1

**Concrete**

0
---

**Workers(+/-)**

20
----

**Update Resources**

0	Asphalt Drum Mix Plant	0
2	JCB	1
0	Cement bags	20
0	Limestone	

**Message**

No work has been allocated

OK

**Allocate Work**

Days required to complete the ... 2

## TEST #16:

**Description:** Update Resources.

**Input:** User Action (Button Click)

**Output:** The table is filled with the all the details of the materials details.

The screenshot shows the 'Administrator' interface with a blue header. Below the header, there are two tabs: 'Pending Complaints' and 'Update Resources'. The 'Update Resources' tab is active. Below the tabs, there is a table with the following data:

Resource type	Total	Employed	Unemployed
R1	10	0	10
R2	7	0	7
R3	8	0	8
R4	6	0	6
R5	10	0	10
R6	8	0	8
W	70	0	70

Below the table, there are several input fields for updating resources:

- Machines(+/-)**: Road Rollers (0), Concrete Road Pavers (0), Asphalt Drum Mix Plant (0), Road Marking Machines (0), Chips Spreader (0), JCB (0).
- Raw Materials(+/-)**: Sand bags (0), Asphalt (0), Cement bags (0), Brick Trucks (0), Concrete (0), Limestone (0).
- Workers(+/-)**: 0

At the bottom right, there is an 'Update Resources' button.

## TEST #17:

**Description:** Update Resources.

**Input:** User fills up the new numbers of resources added or removed (+/-)

**Output:** The resource table is updated with the new numbers and message saying "New work may be allocated please click on allocate work".

The screenshot shows the 'Administrator' interface with a blue header. Below the header, there are two tabs: 'Pending Complaints' and 'Update Resources'. The 'Update Resources' tab is active. Below the tabs, there is a table with the following data:

Resource type	Total	Employed	Unemployed
R1	10	0	10
R2	7	0	7
R3	8	0	8
R4	6	0	6
R5	10	0	10
R6	8	0	8
W	70	0	70

Below the table, there are several input fields for updating resources:

- Machines(+/-)**: Road Rollers (0), Concrete Road Pavers (0), Asphalt Drum Mix Plant (0), Road Marking Machines (0), Chips Spreader (0), JCB (0).
- Raw Materials(+/-)**: Sand bags (0), Asphalt (0), Cement bags (15), Brick Trucks (0), Concrete (0), Limestone (0).
- Workers(+/-)**: 0

A message dialog box is displayed in the center of the screen with the text: "New work may be allocated. Please Click on Allocate Work". The dialog box has an 'OK' button.

At the bottom right, there is an 'Update Resources' button.

### TEST #18:

**Description:** Complaints De-allocated.

**Input:** User enters a negative value.

**Output:** If resources are decreased more than unemployed value the complaints are deallocated from lowest to highest priority.

The screenshot shows the 'Administrator' interface with a blue header and a white table. A modal message box is displayed in the center, stating 'Works deallocated are,11'. The table has four columns: 'Resource type', 'Total', 'Employed', and 'Unemployed'. Below the table, there are input fields for 'Machines(\*/-)' and 'Raw Materials(\*/-)' with values of 0. A 'Workers(\*/-)' field at the bottom has a value of 0. An 'Update Resources' button is located at the bottom right.

Resource type	Total	Employed	Unemployed
M1	14	0	14
M2	10	2	8
M3	12	0	12
M4	15	2	13
M5	12	0	12
M6	12	1	11
R1	10	2	8

### TEST #19:

**Description:** Display Mayor Menu

**Input:** User action (Button Click)

**Output:** Mayor Menu is shown.

The screenshot shows the 'Mayor' menu interface with a blue header and a white table. The menu contains six buttons: 'Repair Work in Progress', 'Material Statistics', 'Repair Work Completed', 'Complaints Info', 'Repair Works Pending', and 'Repair Type Statistics'. A 'Back' button is located at the top left.

Menu Item
Repair Work in Progress
Material Statistics
Repair Work Completed
Complaints Info
Repair Works Pending
Repair Type Statistics

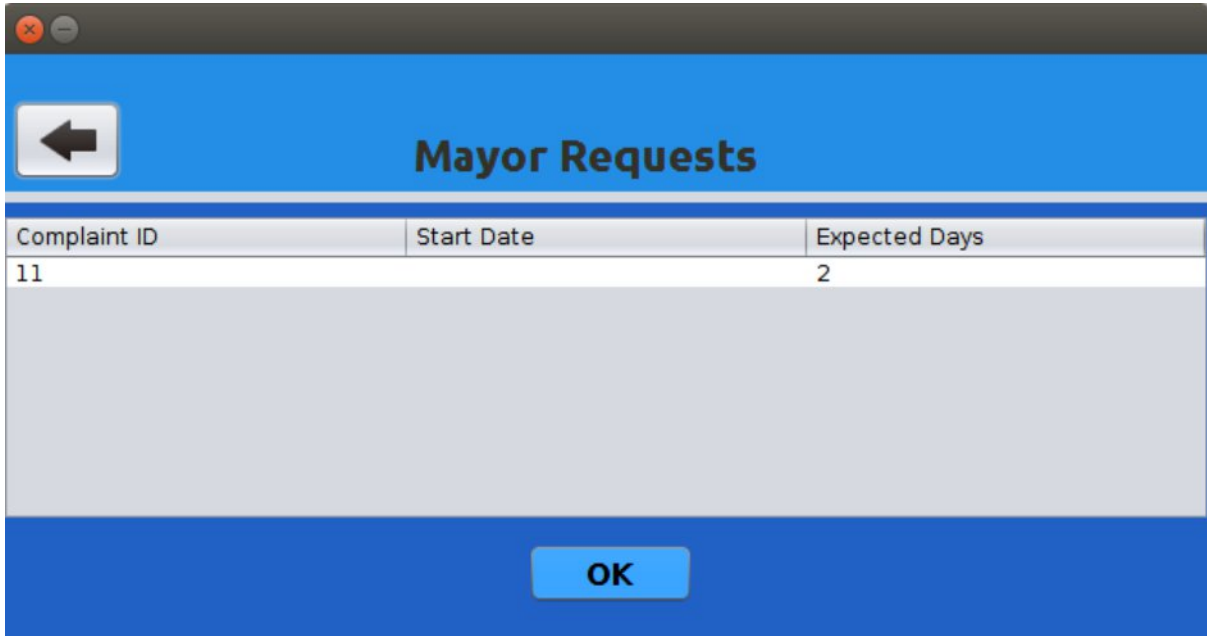


### TEST #20:

**Description:** Repair Work in progress.

**Input:** User action (Button Click)

**Output:** Table is shown containing the details of repair works going on.



The screenshot shows a window titled "Mayor Requests" with a blue header and footer. A back arrow button is in the top left. The table has three columns: "Complaint ID", "Start Date", and "Expected Days". It contains one row with the value "11" under "Complaint ID" and "2" under "Expected Days". The "Start Date" column is empty. An "OK" button is at the bottom center.

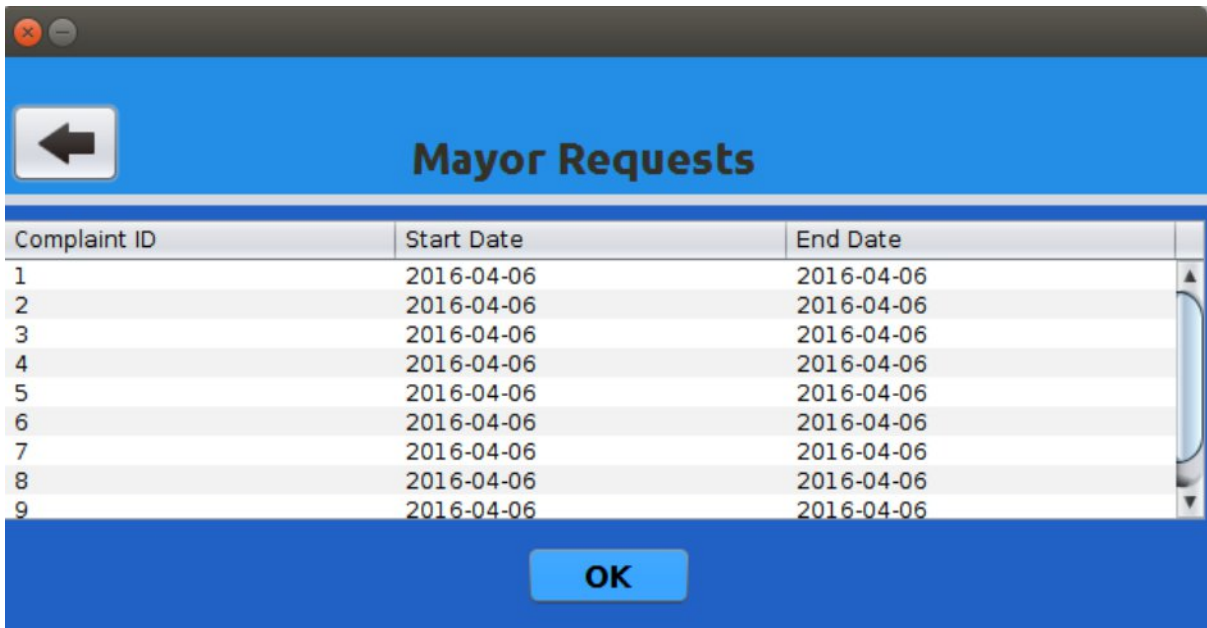
Complaint ID	Start Date	Expected Days
11		2

### TEST #21:

**Description:** Repair Work in completed.

**Input:** User action (Button Click)

**Output:** Table is shown containing the details of repair works completed.



The screenshot shows the same "Mayor Requests" window, but the table now has four columns: "Complaint ID", "Start Date", "End Date", and an empty fourth column. It contains nine rows of data, all with the date "2016-04-06" in both the "Start Date" and "End Date" columns. An "OK" button is at the bottom center.

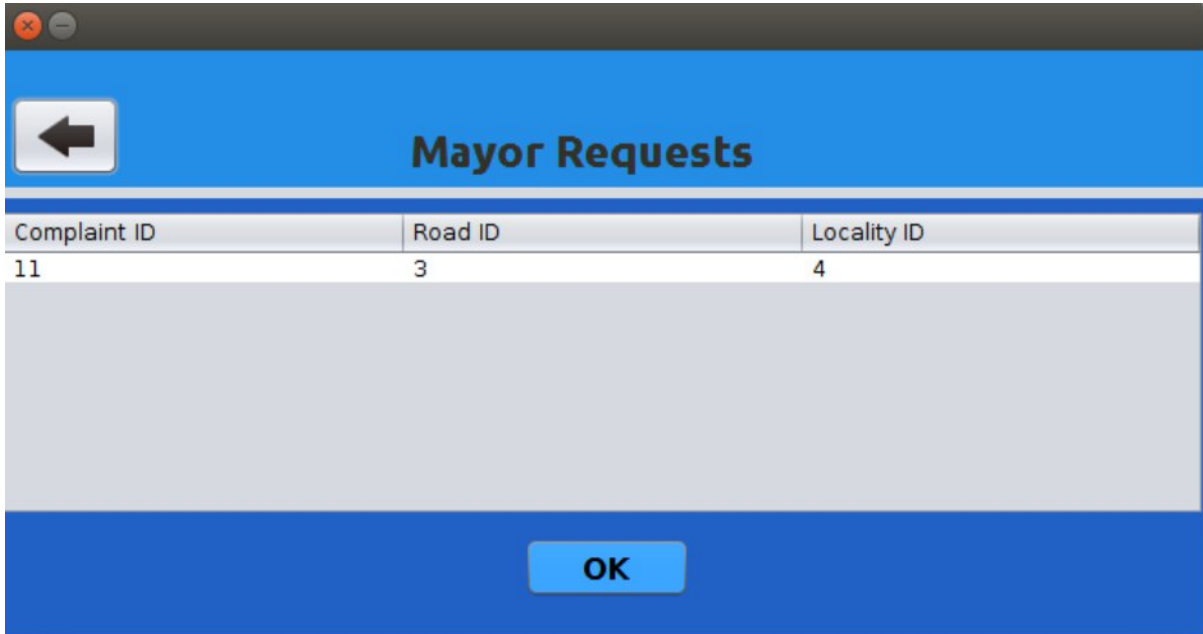
Complaint ID	Start Date	End Date	
1	2016-04-06	2016-04-06	
2	2016-04-06	2016-04-06	
3	2016-04-06	2016-04-06	
4	2016-04-06	2016-04-06	
5	2016-04-06	2016-04-06	
6	2016-04-06	2016-04-06	
7	2016-04-06	2016-04-06	
8	2016-04-06	2016-04-06	
9	2016-04-06	2016-04-06	

## TEST #22:

**Description:** Repair Works pending.

**Input:** User action (Button Click)

**Output:** Table is shown containing the details of pending repair works.

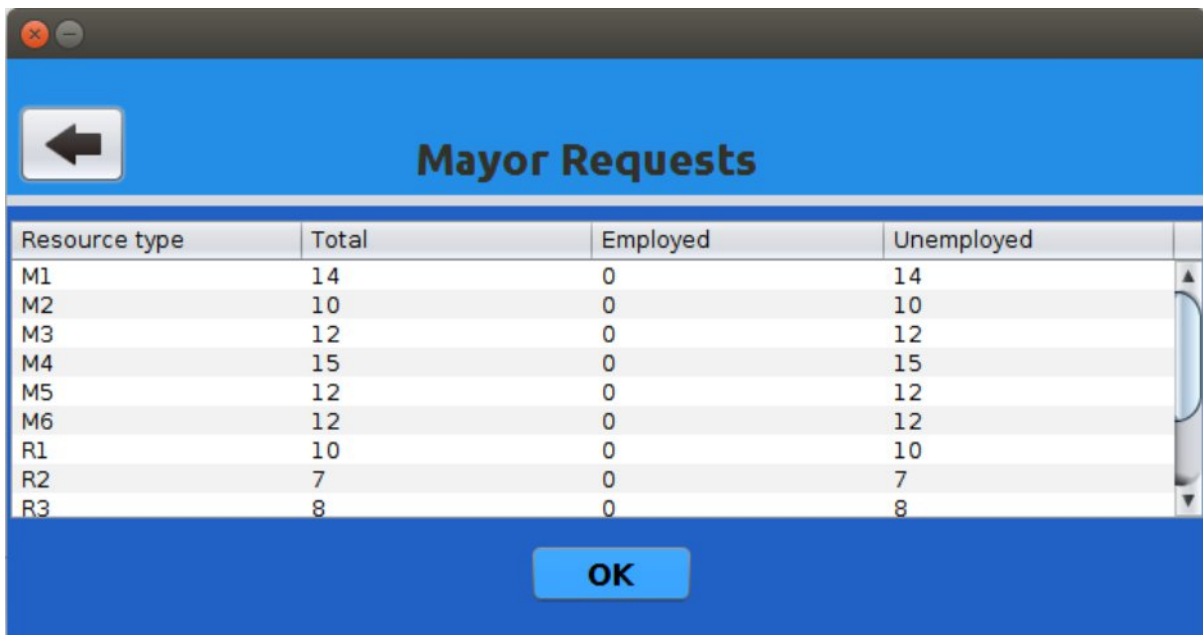


## TEST #23:

**Description:** Material Statistics.

**Input:** User action (Button Click)

**Output:** Table is shown containing the details of all the statistics of materials.

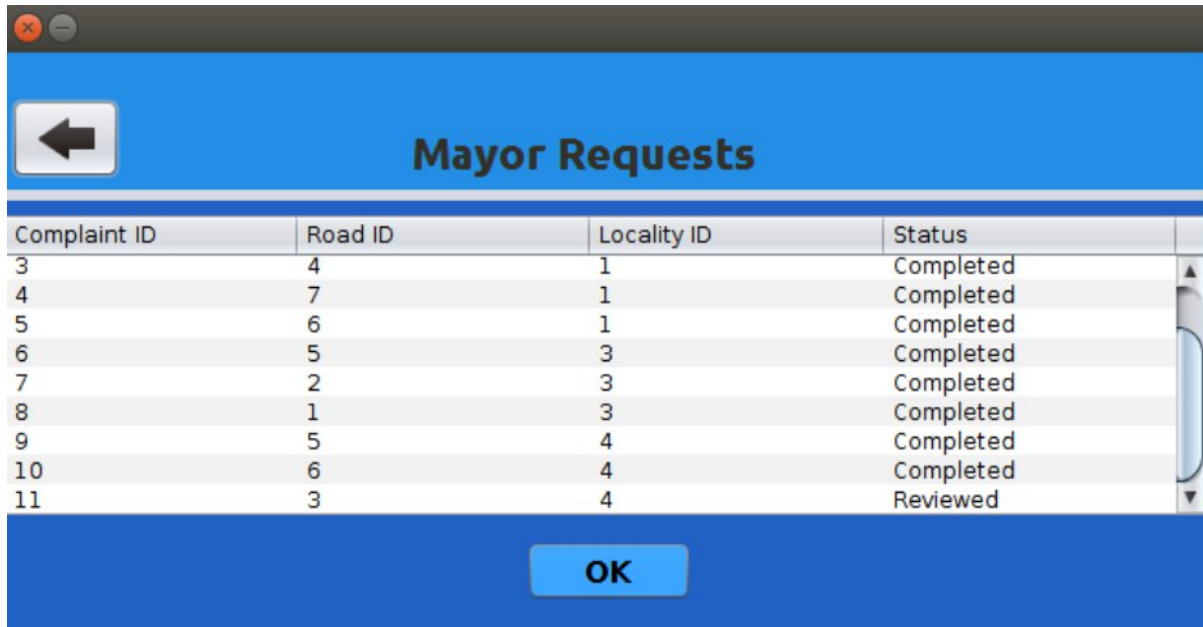


### TEST #24:

**Description:** Complaints info.

**Input:** User action (Button Click)

**Output:** Table is shown containing the details of all the complaints.



A screenshot of a software window titled "Mayor Requests". The window has a blue header bar with a back arrow button on the left. Below the header is a table with four columns: "Complaint ID", "Road ID", "Locality ID", and "Status". The table contains 11 rows of data. At the bottom of the window is a blue bar with an "OK" button.

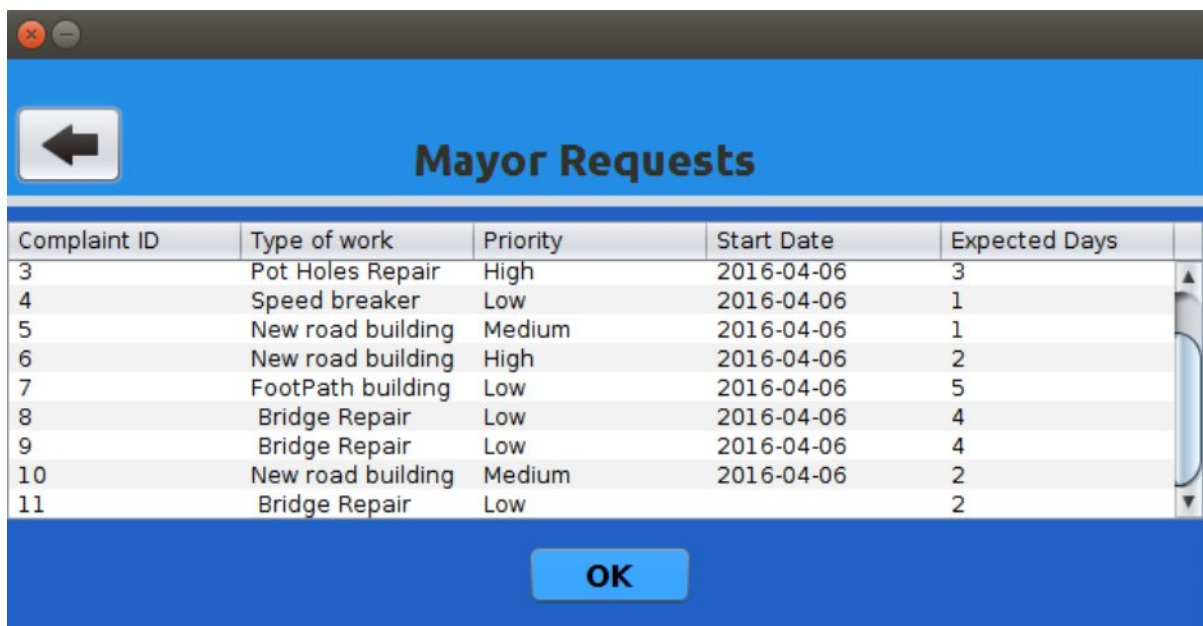
Complaint ID	Road ID	Locality ID	Status
3	4	1	Completed
4	7	1	Completed
5	6	1	Completed
6	5	3	Completed
7	2	3	Completed
8	1	3	Completed
9	5	4	Completed
10	6	4	Completed
11	3	4	Reviewed

### TEST #25:

**Description:** Repair type statistics.

**Input:** User action (Button Click).

**Output:** Table is shown containing the details of the type of work done.



A screenshot of a software window titled "Mayor Requests". The window has a blue header bar with a back arrow button on the left. Below the header is a table with five columns: "Complaint ID", "Type of work", "Priority", "Start Date", and "Expected Days". The table contains 11 rows of data. At the bottom of the window is a blue bar with an "OK" button.

Complaint ID	Type of work	Priority	Start Date	Expected Days
3	Pot Holes Repair	High	2016-04-06	3
4	Speed breaker	Low	2016-04-06	1
5	New road building	Medium	2016-04-06	1
6	New road building	High	2016-04-06	2
7	FootPath building	Low	2016-04-06	5
8	Bridge Repair	Low	2016-04-06	4
9	Bridge Repair	Low	2016-04-06	4
10	New road building	Medium	2016-04-06	2
11	Bridge Repair	Low	2016-04-06	2