Mkhanyisi Gamedze

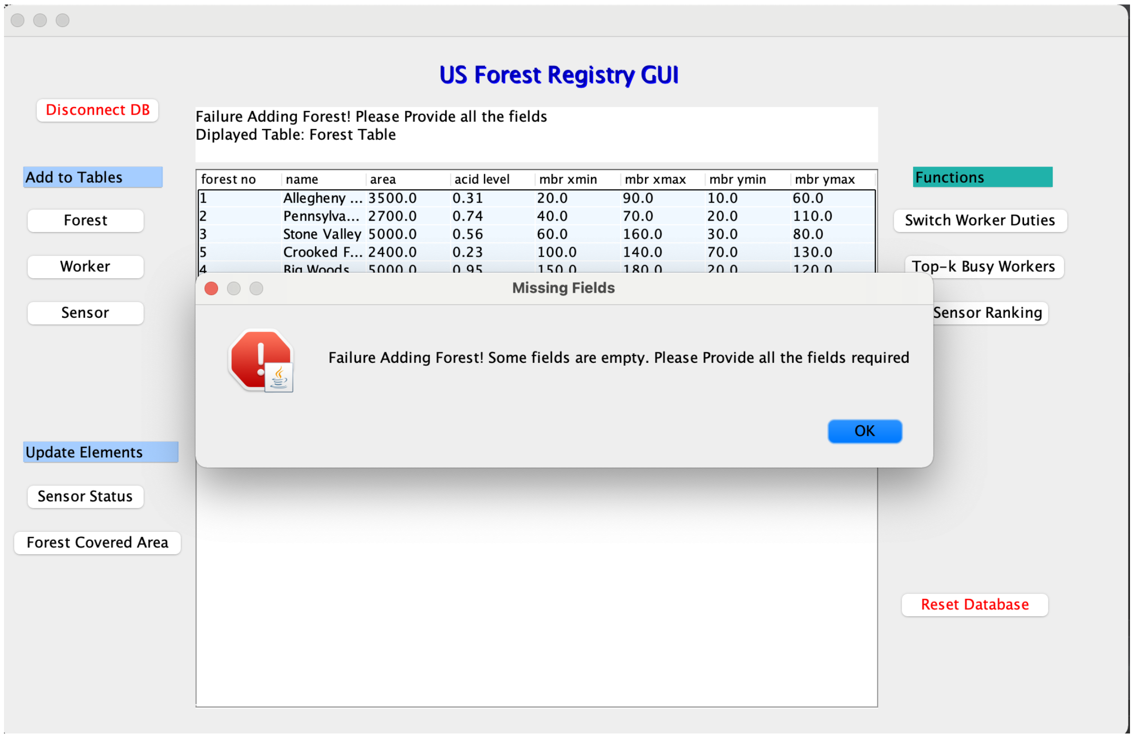
COSI 127B Database Management Systems
PA1

My US Forest Registry GUI was implemented using JDBC, Postgresql and Java Swing JFrame library tools. The GUI needs to be connected to the Database for almost all of the functions to execute on the database. I also utilized the net.proteanit.sql.DbUtils library to make my GUI more interactive and display the initial plus updated tables for each of the core operations within my GUI. This is important for the right side operations for displaying sensor ranking and top busy workers, as the GUI makes it more interactive. Pop up error prompts or success messages also guide the user on whether operations are successful or not.

A run through video documentation handling all of the tasks are highlighted within my demo videos and quick fixes I noted during testing. Almost all of the functionality was implemented smoothly, but the were some missed development areas with error checking, especially for variable types, as sensors for example need datetime values and this format needs to match database. A user can easily get this wrong and that can be an issue. All other query errors are handled by pop ups or terminal print statements.

Summary of Functions

1. Add Forest - Requires all fields to be entered by user and cannot have duplicate “forest no” or “name” which already exist in table

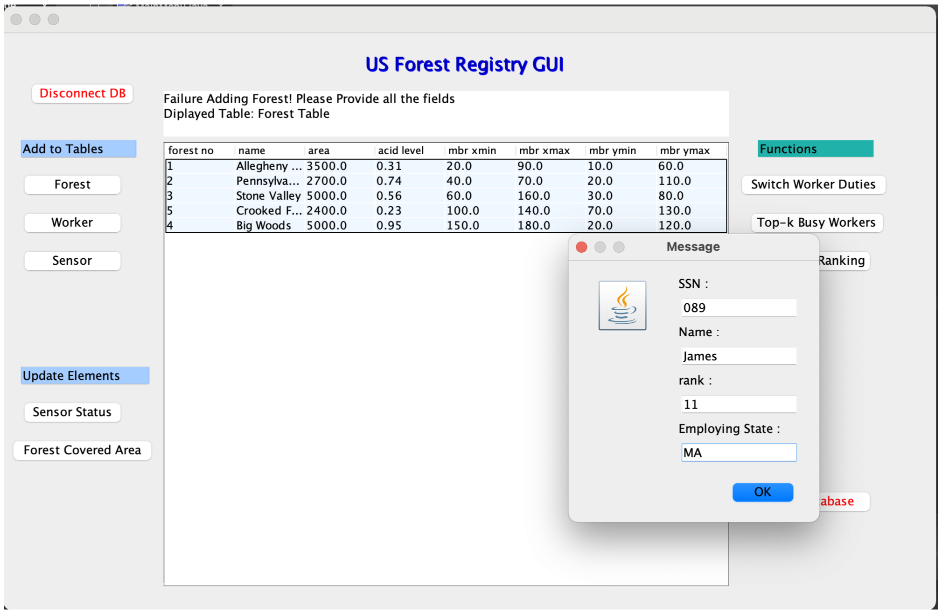


Failure Adding Forest! Please Provide all the fields
Diplayed Table: Forest Table

forest no	name	area	acid level	mbr xmin	mbr xmax	mbr ymin	mbr ymax
1	Allegheny ...	3500.0	0.31	20.0	90.0	10.0	60.0
2	Pennsylva...	2700.0	0.74	40.0	70.0	20.0	110.0
3	Stone Valley	5000.0	0.56	60.0	160.0	30.0	80.0
5	Crooked F...	2400.0	0.23	100.0	140.0	70.0	130.0
4	Big Woods	5000.0	0.95	150.0	180.0	20.0	120.0

Failure Adding Forest! Some fields are empty. Please Provide all the fields required

2. Add Worker - Prompts user to enter details and only allows states within the select forests, as the database rejects queries with states not in the state table. Unique SSN primary key required, and all fields for user to be added to table

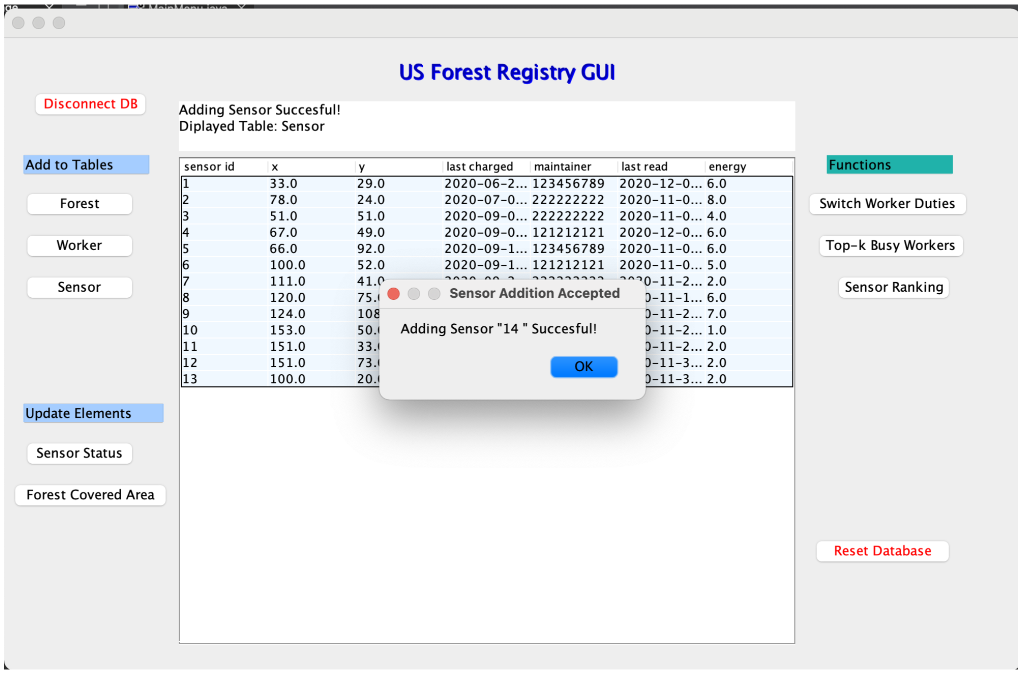


Failure Adding Forest! Please Provide all the fields
Diplayed Table: Forest Table

forest no	name	area	acid level	mbr xmin	mbr xmax	mbr ymin	mbr ymax
1	Allegheny ...	3500.0	0.31	20.0	90.0	10.0	60.0
2	Pennsylva...	2700.0	0.74	40.0	70.0	20.0	110.0
3	Stone Valley	5000.0	0.56	60.0	160.0	30.0	80.0
5	Crooked F...	2400.0	0.23	100.0	140.0	70.0	130.0
4	Big Woods	5000.0	0.95	150.0	180.0	20.0	120.0

SSN : 089
Name : James
rank : 11
Employing State : MA

3. Add Sensor - Enter all fields required and date needs to be formatted properly for addition to be accepted. Produces an error message if a sensor with the same coordinates or sensor ID already exists.

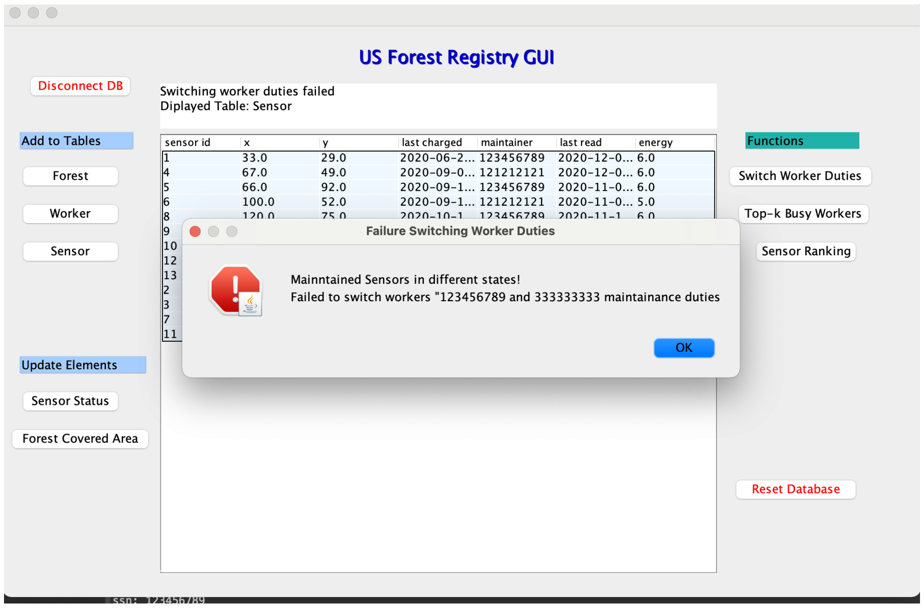


Adding Sensor Succesful!
Diplayed Table: Sensor

sensor id	x	y	last charged	maintainer	last read	energy
1	33.0	29.0	2020-06-2...	123456789	2020-12-0...	6.0
2	78.0	24.0	2020-07-0...	222222222	2020-11-0...	8.0
3	51.0	51.0	2020-09-0...	222222222	2020-11-0...	4.0
4	67.0	49.0	2020-09-0...	121212121	2020-12-0...	6.0
5	66.0	92.0	2020-09-1...	123456789	2020-11-0...	6.0
6	100.0	52.0	2020-09-1...	121212121	2020-11-0...	5.0
7	111.0	41.0	[illegible]	[illegible]	2020-11-2...	2.0
8	120.0	75.	[illegible]	[illegible]	0-11-1...	6.0
9	124.0	108	[illegible]	[illegible]	0-11-2...	7.0
10	153.0	50.	[illegible]	[illegible]	0-11-2...	1.0
11	151.0	33.	[illegible]	[illegible]	0-11-2...	2.0
12	151.0	73.	[illegible]	[illegible]	0-11-3...	2.0
13	100.0	20.	[illegible]	[illegible]	0-11-3...	2.0

Adding Sensor "14 " Succesful!

4. Switch Worker Duties - provides a dropdown to change the sensors for two workers, and only successfully switches if both workers are in the same employing state.



US Forest Registry GUI

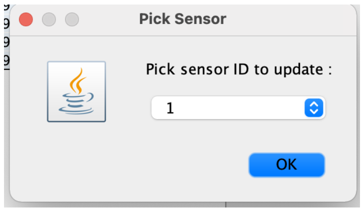
Switching worker duties failed
Diplayed Table: Sensor

sensor id	x	y	last charged	maintainer	last read	energy
1	33.0	29.0	2020-06-2...	123456789	2020-12-0...	6.0
4	67.0	49.0	2020-09-0...	121212121	2020-12-0...	6.0
5	66.0	92.0	2020-09-1...	123456789	2020-11-0...	6.0
6	100.0	52.0	2020-09-1...	121212121	2020-11-0...	5.0
8	120.0	75.0	2020-10-1	123456789	2020-11-1	6.0
9						
10						
12						
13						
2						
3						
7						
11						

Failure Switching Worker Duties

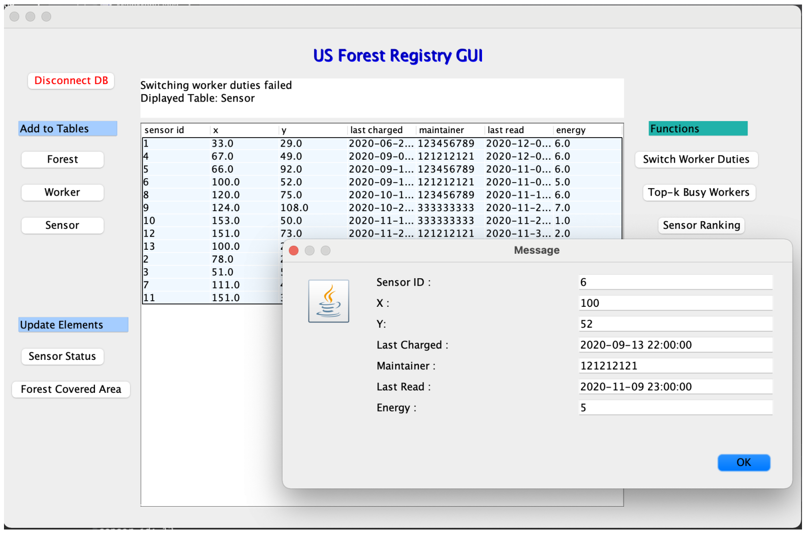
Mainntained Sensors in different states!
Failed to switch workers "123456789 and 333333333 maintainance duties

5. Update Sensor Status - Allows users to pick sensor and then update the value of sensor. After the user enters update values, quality check logic within the database is similar to adding a new sensor and checks for duplicate ID plus location (x,y) and no delete values. Successfully updates value if all is clear.



Pick Sensor

Pick sensor ID to update : 1



US Forest Registry GUI

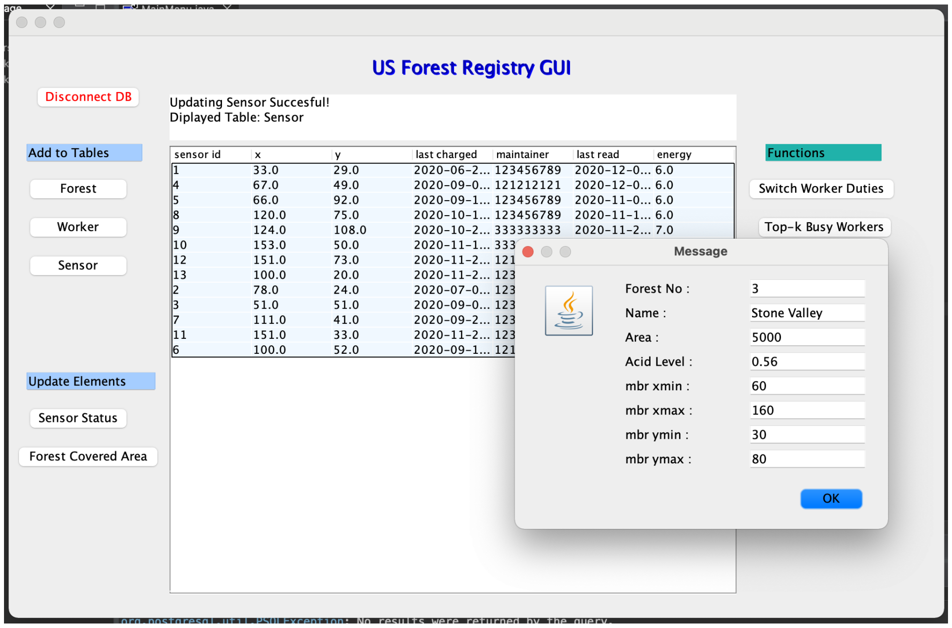
Switching worker duties failed
Diplayed Table: Sensor

sensor id	x	y	last charged	maintainer	last read	energy
1	33.0	29.0	2020-06-2...	123456789	2020-12-0...	6.0
4	67.0	49.0	2020-09-0...	121212121	2020-12-0...	6.0
5	66.0	92.0	2020-09-1...	123456789	2020-11-0...	6.0
6	100.0	52.0	2020-09-1...	121212121	2020-11-0...	5.0
8	120.0	75.0	2020-10-1...	123456789	2020-11-1...	6.0
9	124.0	108.0	2020-10-2...	333333333	2020-11-2...	7.0
10	153.0	50.0	2020-11-1...	333333333	2020-11-2...	1.0
12	151.0	73.0	2020-11-2...	121212121	2020-11-3...	2.0
13	100.0					
2	78.0					
3	51.0					
7	111.0					
11	151.0					

Message

Sensor ID :	6
X :	100
Y:	52
Last Charged :	2020-09-13 22:00:00
Maintainer :	121212121
Last Read :	2020-11-09 23:00:00
Energy :	5

6. Update Forest Covered Area - Prompts user with dropdown to select forest and updates saved values, checking for no forest name or number duplicates before running query.

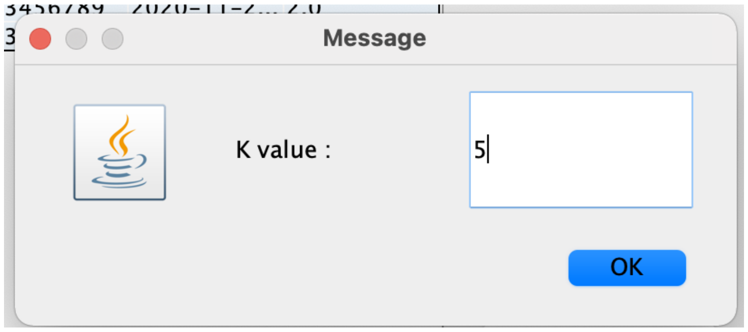


Updating Sensor Succesful!
Diplayed Table: Sensor

sensor id	x	y	last charged	maintainer	last read	energy
1	33.0	29.0	2020-06-2...	123456789	2020-12-0...	6.0
4	67.0	49.0	2020-09-0...	121212121	2020-12-0...	6.0
5	66.0	92.0	2020-09-1...	123456789	2020-11-0...	6.0
8	120.0	75.0	2020-10-1...	123456789	2020-11-1...	6.0
9	124.0	108.0	2020-10-2...	333333333	2020-11-2...	7.0
10	153.0	50.0	2020-11-1...	333		
12	151.0	73.0	2020-11-2...	121		
13	100.0	20.0	2020-11-2...	123		
2	78.0	24.0	2020-07-0...	123		
3	51.0	51.0	2020-09-0...	123		
7	111.0	41.0	2020-09-2...	123		
11	151.0	33.0	2020-11-2...	123		
6	100.0	52.0	2020-09-1...	121		

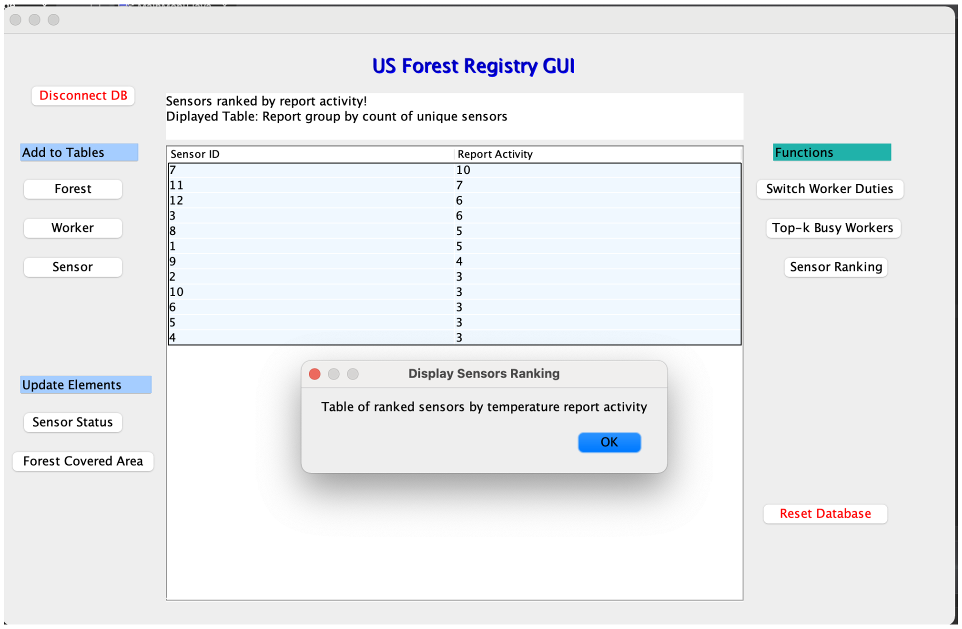
Forest No :	3
Name :	Stone Valley
Area :	5000
Acid Level :	0.56
mbr xmin :	60
mbr xmax :	160
mbr ymin :	30
mbr ymax :	80

7. Asks user to enter k-value for Top-k busy workers, then runs SQL query plus limits result to k values. Parses an Int



K value : 5

8. Sensor Ranking, from the reports table, this groups sensors by the total number of reports in table and then sorts the counts in descending order.



Sensors ranked by report activity!
Diplayed Table: Report group by count of unique sensors

Sensor ID	Report Activity
7	10
11	7
12	6
3	6
8	5
1	5
9	4
2	3
10	3
6	3
5	3
4	3

Table of ranked sensors by temperature report activity

Further Development areas:

Error handling and pop up messages for some user inputs that are successful with program but do not get added to database. Displaying a query failure window instead when that happens. I did not do this, as even for successful queries, this still returns a no result exception and thus could not differentiate this and so left it unhandled for cases where query may fail, but there are many error check conditions before that. Very few test cases were not handled.

Variable type checking and dropdowns to avoid user input that can lead to wrong values.