**Maintenance Manual**



**Project name:**

Data Visualization

**Date**:11/18/2021

**TABLE OF CONTENTS**

[Introduction 5](#_Toc15749)

[Implementation tools and technologies 5](#_Toc15750)

[Runbook 5](#_Toc15751)

[1. List what to do if: 5](#_Toc15752)

[a. Server goes down 5](#_Toc15753)

[b. Database becomes corrupt 6](#_Toc15754)

[C. Third party services go down 6](#_Toc15755)

[2. How to configure a computer for development work 6](#_Toc15756)

[3. How to make a code change 8](#_Toc15757)

[4. How to get to the log files and examine them for errors 11](#_Toc15758)

[5. How to deploy releases 11](#_Toc15759)

[6. How to run automated tests 11](#_Toc15760)

[7. How to do “sanity” check to ensure code changes didn’t break anything 11](#_Toc15761)

[User interaction: 11](#_Toc15762)

[Logging In 11](#_Toc15763)

[Navigate Through User-view 12](#_Toc15764)

[Overview 13](#_Toc15765)

[Filling out the template 14](#_Toc15766)

[Going back in date 15](#_Toc15767)

[Export Report 17](#_Toc15768)

[Help Requests 20](#_Toc15769)

[Logging out 21](#_Toc15770)

[Navigate Through Admin-View 22](#_Toc15771)

[User Management 25](#_Toc15772)

[Add User 26](#_Toc15773)

[Add Manger 29](#_Toc15774)

[Edit User 30](#_Toc15775)

[Edit manager 31](#_Toc15776)

[Delete User 32](#_Toc15777)

[Delete Manager 33](#_Toc15778)

[Add/Edit 34](#_Toc15779)

[Navigating Through Manger-View 35](#_Toc15780)

[Manager Overview 36](#_Toc15781)

[Database 39](#_Toc15782)

# Introduction

Lgstx Services, Inc is a business Analyst, Material handling equipment supplier. The company is looking for data visualization products to replace their current use of excel sheets and convert them into a dashboard for each business trade they own. The product is a web-based platform. The dashboard is used to gather data from users and present it in a visually digestible manner for management to be able to review and understand.

# Implementation tools and technologies

Tools and technologies would be needed to maintain the software product.

Visual Studio Code

Express

Node.js

MongoDB

Languages: Java Script, html Install mongodb community:

https://www.youtube.com/watch?v=Z478ODY4ceQ

Install nodejs:

https://www.youtube.com/watch?v=AuCuHvgOeBY

I ran all these changes on my local end on fresh project.

/\*\*Updated needed packages\*\*/

Admin will be able to register himself and other users

# Runbook

## 1. List what to do if:

### a. Server goes down

i. How do they restart/recover?

* Create a backup of the data files.
* Create a backup copy of the data files in the --dbpath.
* download the latest version of MongoDB
* Start mongod with --repair.
* To repair the data files, start the mongod instance with the --repair option.
* Issue a command similar to the following for your standalone:

mongod --dbpath /data/db --repair

* Upon completion, the dbpath should contain the repaired data files and an empty mongod.lock file.

### b. Database becomes corrupt

ii. How to restore from backup

you can run the above repair command or use a MongoDB Backup

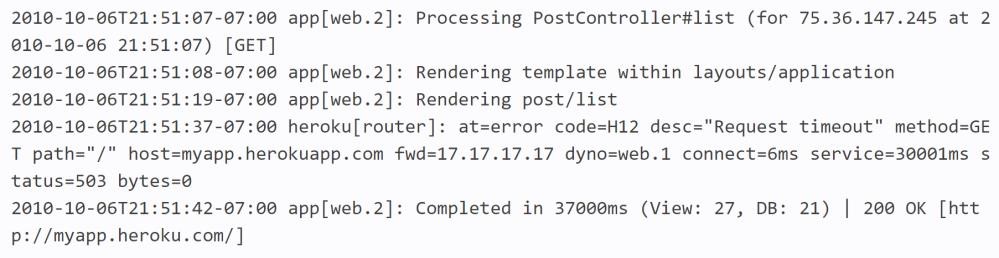
* Open System Settings.
* Go to Controller > Persistence.
* In the Data Folder field, browse to the folder for the backup.
* Select Next to continue through the System Settings windows and restart the service. When the service starts it will run off the restored folder.

### C. Third party services go down

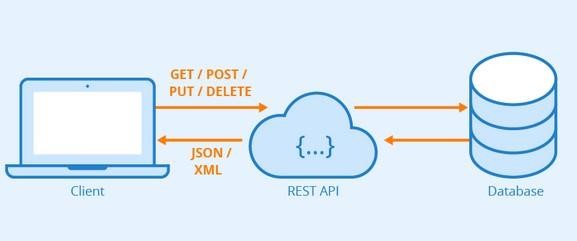
iii. There may be nothing they can do until the service is back up BUT:

1. How do they identify that this happened? if using Heroku as third party server, Heroku will return a standard error page with the HTTP status code 503. To help you debug the underlying error, however, the platform will also add custom error information to your logs. Each type of error gets its own error code, with all HTTP errors starting with the letter H and all runtime errors starting with R. Logging errors start with L.
2. What errors will they see?

Example of timeout error. There're more errors that could come up, please refer to https://devcenter.heroku.com/articles/error-codes



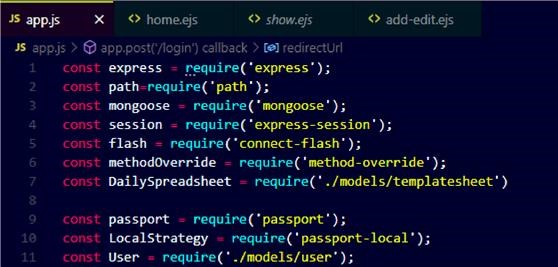
## 3. How to make a code change



We are using the rest api for our project, and this is an image of how it works. We are using Http requests such as Get, Post, Put and Delete, to do the backend/server side of our code.

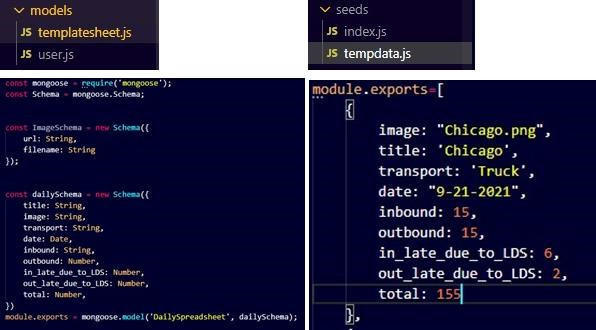
As an example user(clicks something on our client side or website) it will get it from the backend server side which in our case is app.js. The backend(app.js) would be able to get information from our database(MongoDB) for the client/website and then use one of http requests to access data from the database and display it on the client side.

All of the following are npm packages that I used for our serverside/app.js if you need to use any other packages you can install it and call it here (Think like importing java… library)



To get started, when I connected to mongodb the first time, I had created a template schema for our spreadsheet (add/edit fields for users). Located in /models/templatesheet.js (MongoDB uses JavaScript). You can use the example to create the schema needed for add/edit. I created the seeds/tempdata.js and put fixed values for each section.

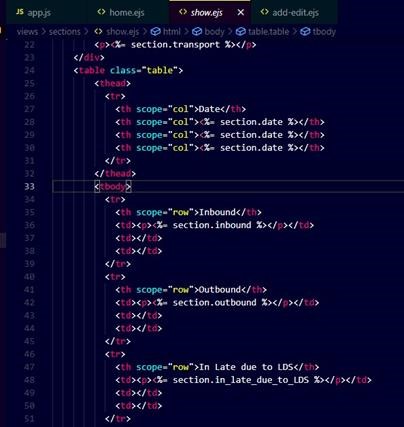
Printing and getting the pages to show on the website using the data from tempdata.js. First I get in backend from the database in app.js app.js



PP.js



In the sections folder I have show.ejs which are the sections and bootstrap tables I had printed along with the data. I used tags: <% [...] %> in the .ejs files to get them from backend/app.js show.ejs



I use the tags <% %> to get the partials/navbar.ejs(frontend page) and also username, printing each section using a for loop.(from the backend/app.js).

Index.ejs (is our dashboard/main page)



I followed the same procedure for the user management page.

## 4. How to get to the log files and examine them for errors

MongoDB logs can be found in the MongoDB log files at **/var/log/mongodb/mongodb.** **log**.

## 5. How to deploy releases

* **MongoDB ATLAS**

Create an Atlas account

Create Your Organization and Project

Deploy Your first cluster

Create a Database User for Your Cluster

Grant Authorized IP Addresses Access to Your Cluster

* **Heroku**

Log into the Heroku CLI

Clone My Demo App Create the Heroku App

https://devcenter.heroku.com/articles/preparing-a-codebase-for-heroku-deployment

## 6. How to run automated tests

The team decided to go with manual testing. Each component was tested individually to meet the predetermined input values.

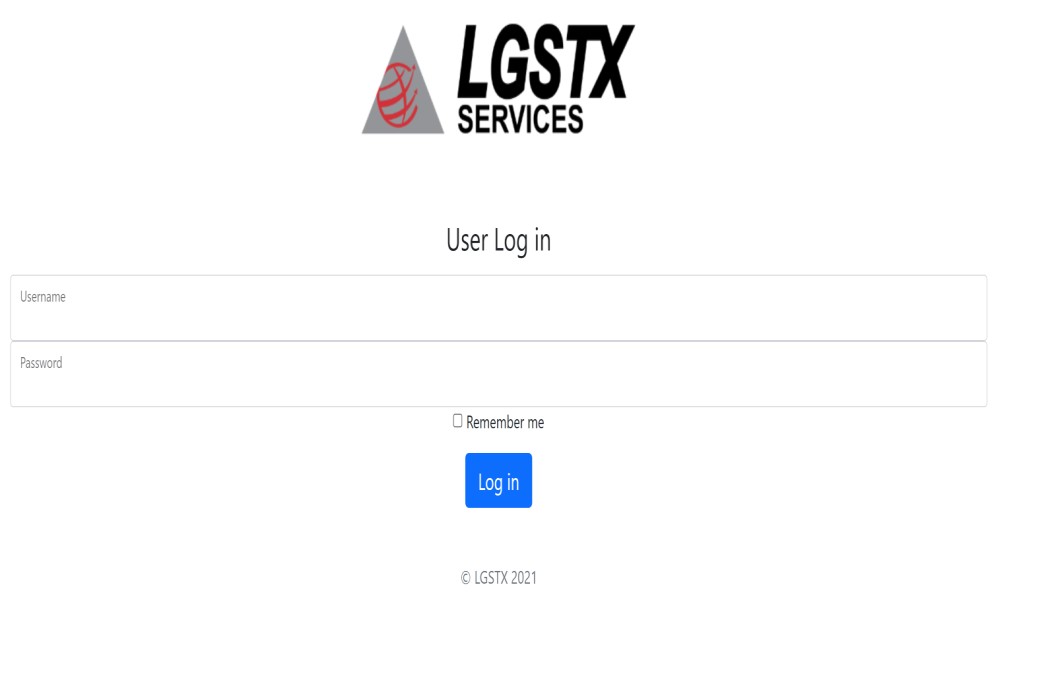
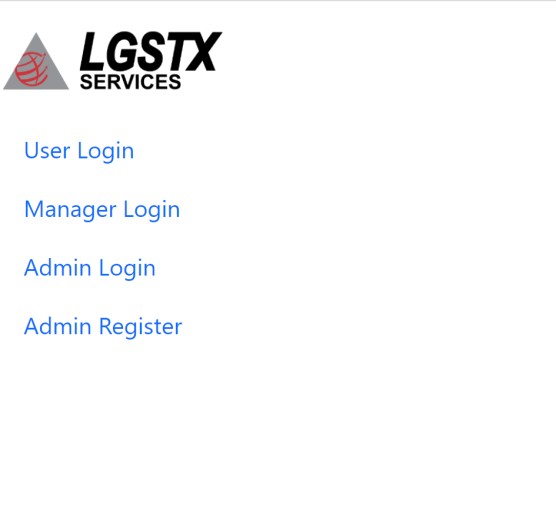
## 7. How to do “sanity” check to ensure code changes didn’t break anything

We made sure the input was validated.

# User interaction:

# Logging In

We have three main Logings: Admin, manger, and 10 business trades. Each business trade will have their own login and they can only see their section templates and data. Admin is responsible for creating a username and password for each one of them. Also, admin would create one login for all the managers to see all business trade in visualized manner

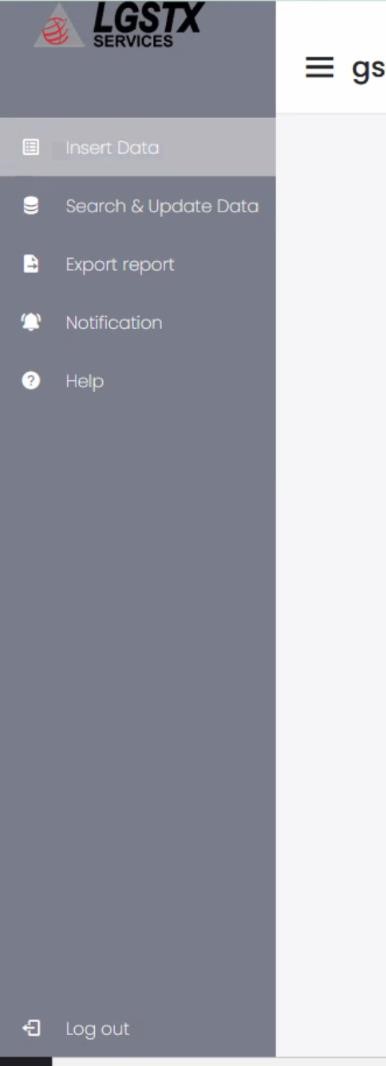


Login with given username and password. Contact admin Bashar Alnasrawi to provide you with access

# Navigate Through User-view

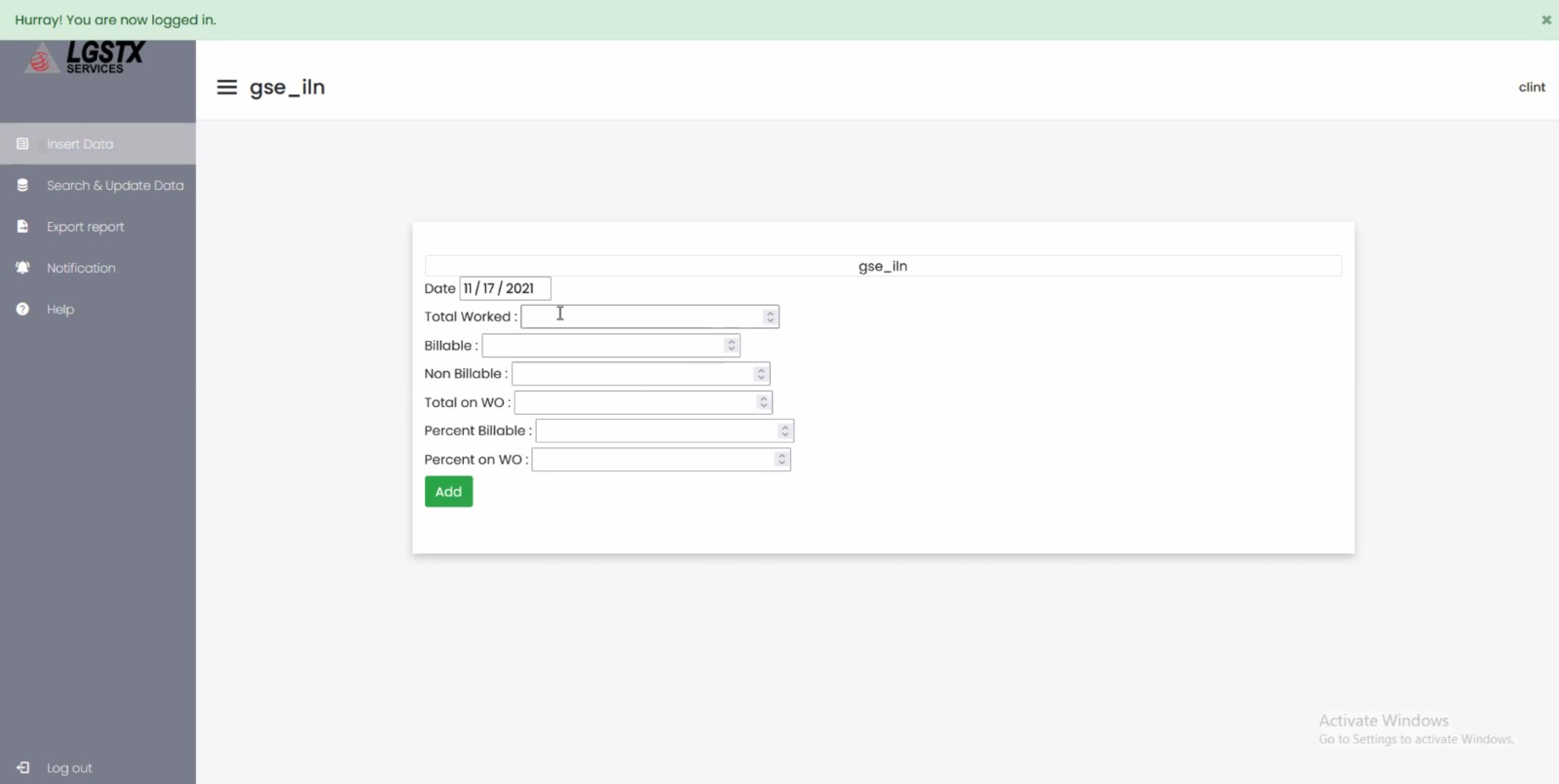
From the landing page, you can press any icon to get to its related area. Depending on your user level, you might not see all the options.

The top left three dashes menu button to access the page needed.



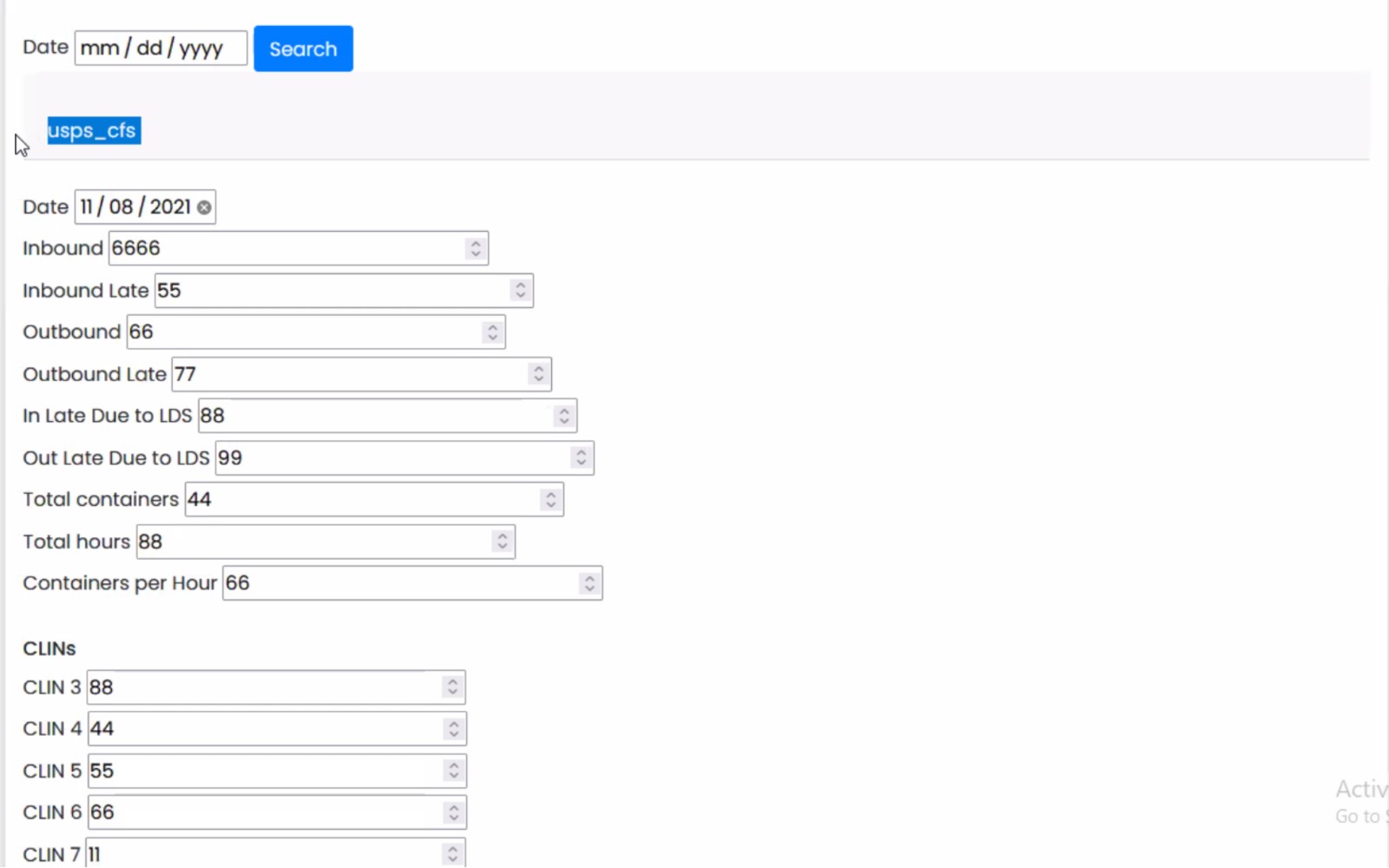
# Overview

Logging in as user should take you to page similar to the below depending on which section you’re in.



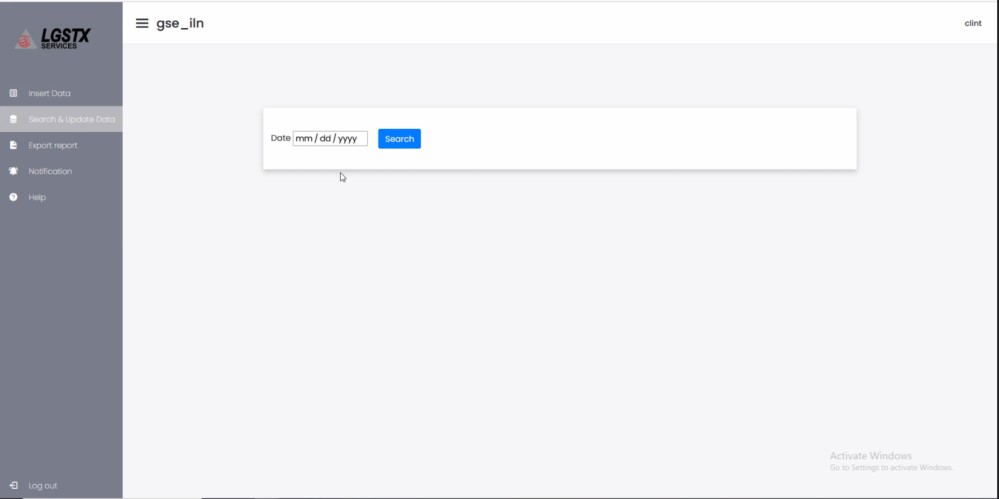
# Filling out the template

Each section will have to fill out this daily report by being on inset data tap and click the add button to report it to the admin and higher management. Only numbers should be entered.



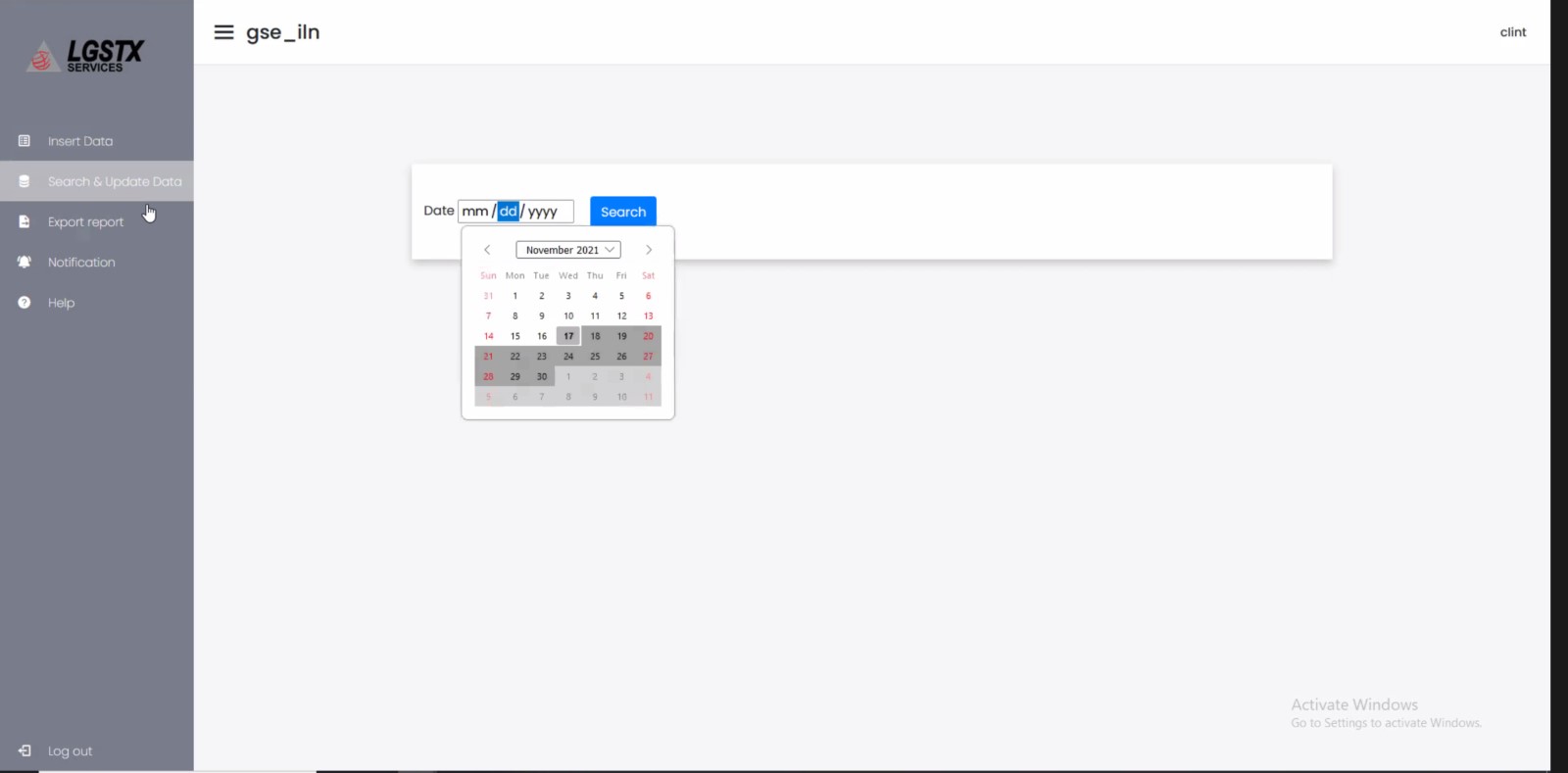
# Going back in date

Users should be able to go back in date and pick reports from the past to view or edit. Click on search & update data.

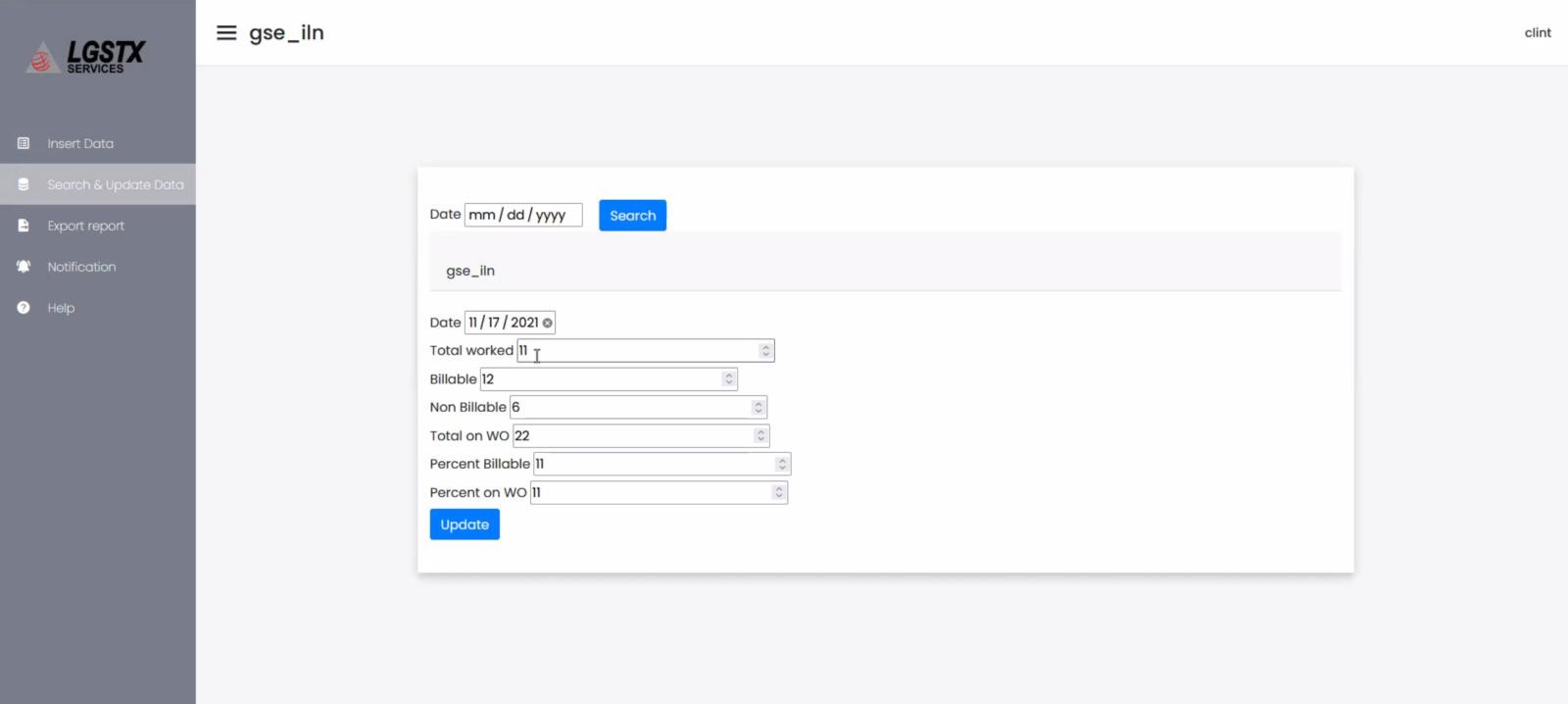


The benefit of this date button is to be able to edit previous reports if a mistake was made.

Pick data and click search.

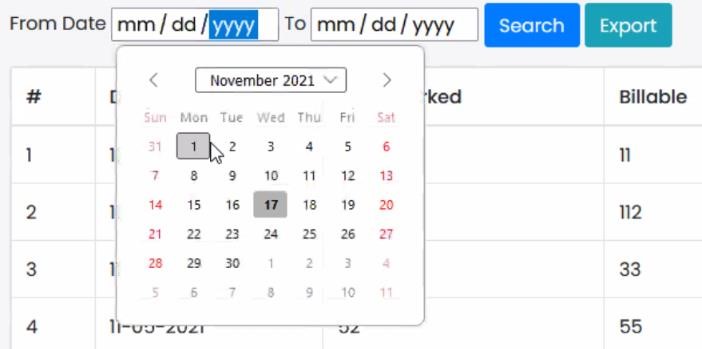


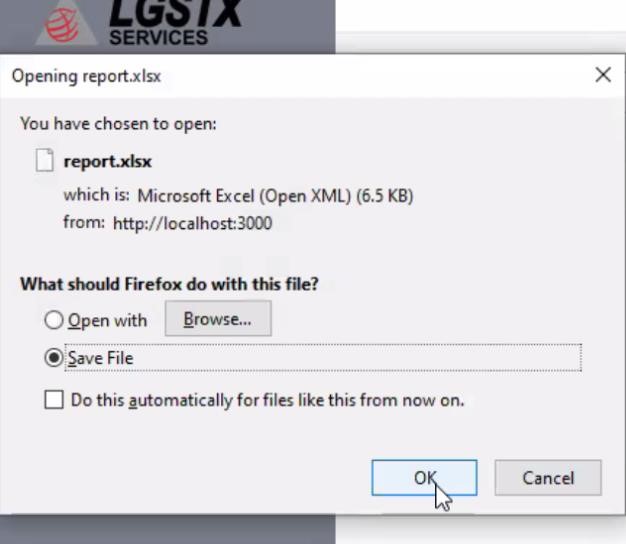
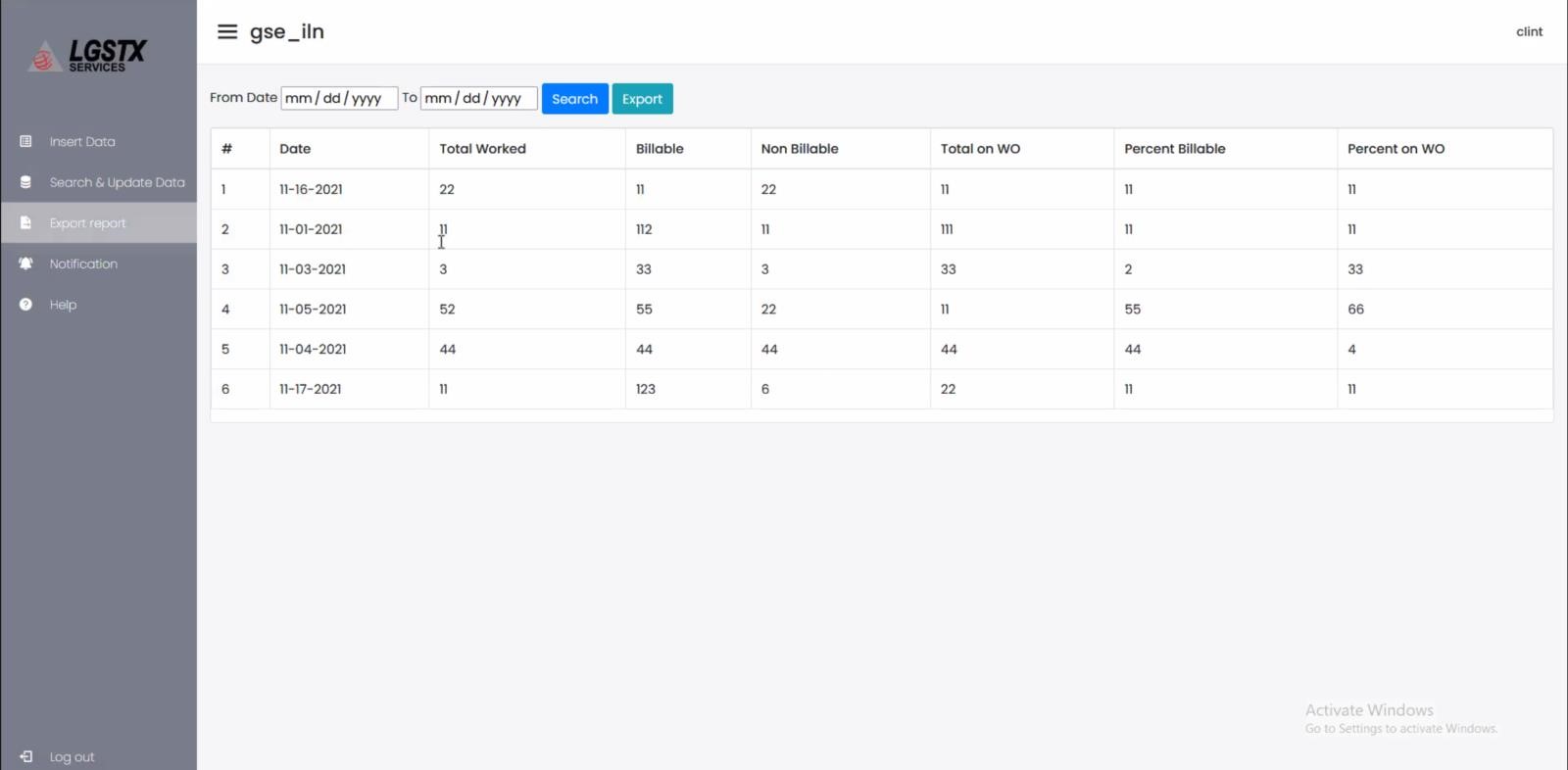
It will populate the report and you can click update once done updating the report

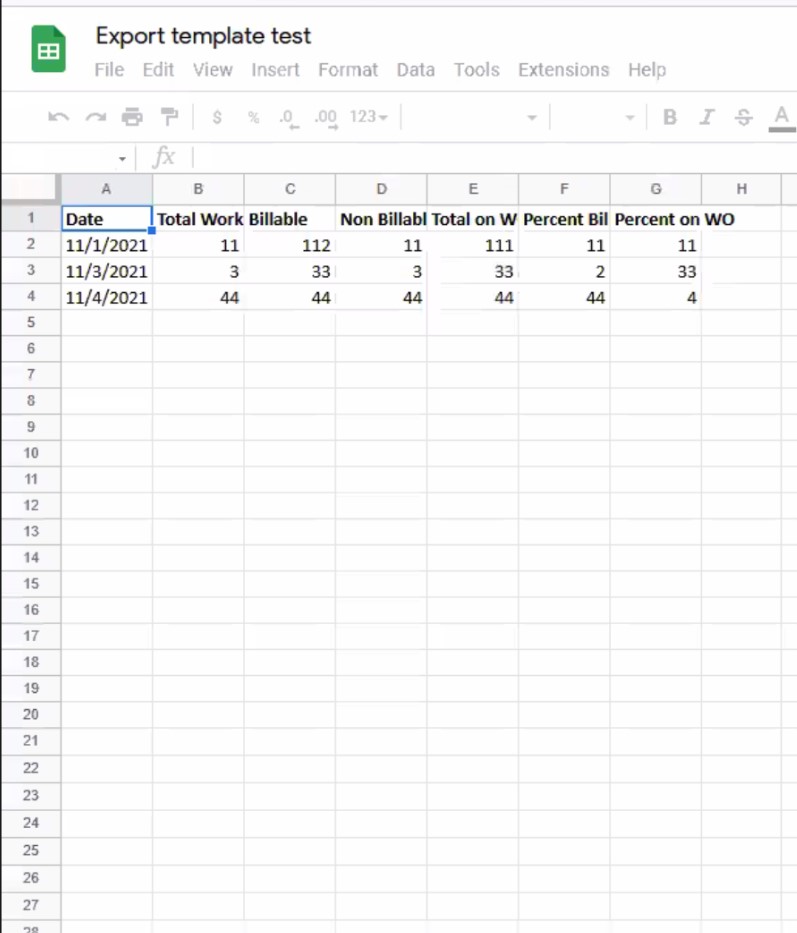


# Export Report

All users and admin should be able to export the reports into excel sheet so it could be downloaded and email easily. Search for the desired date range and click export

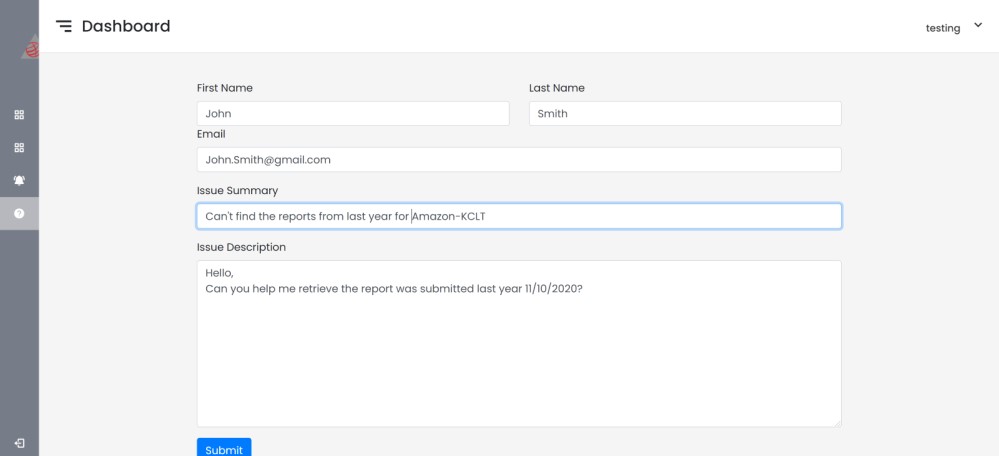






# Help Requests

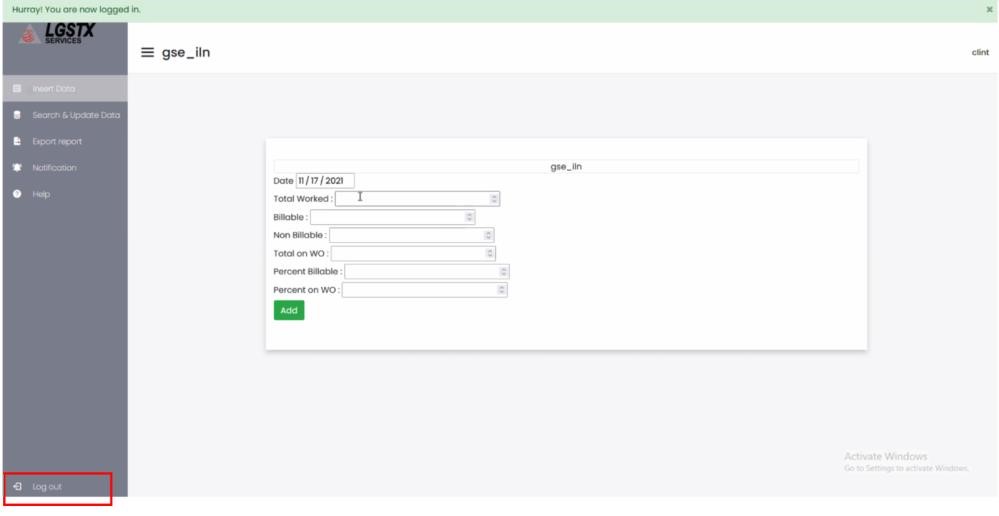
Regular users from each section and all mangers will have the option to submit a help form to the admin Bashar by clicking on the question mark icon and click submit when done.

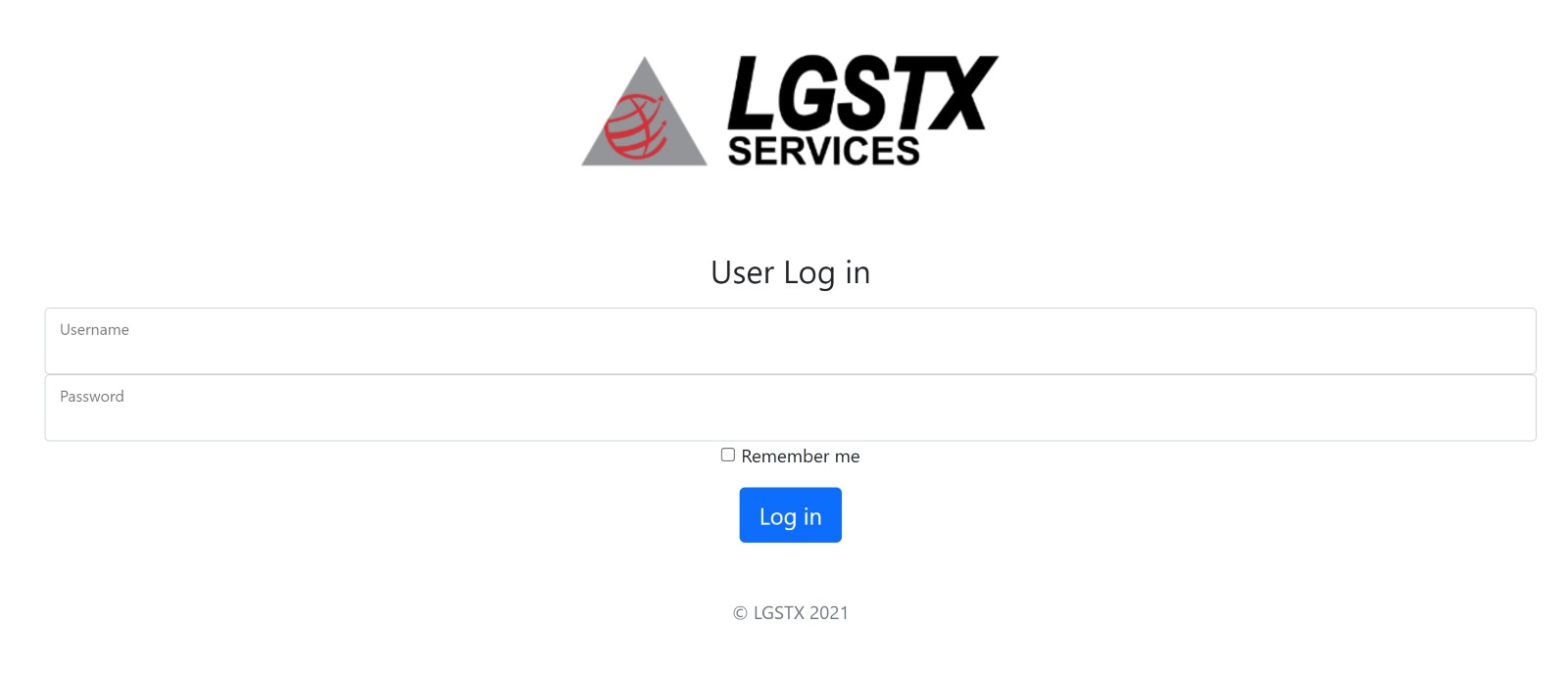


Once the submit button is clicked, the admin will get an email. He will be able to response directly to the person via their email

# Logging out

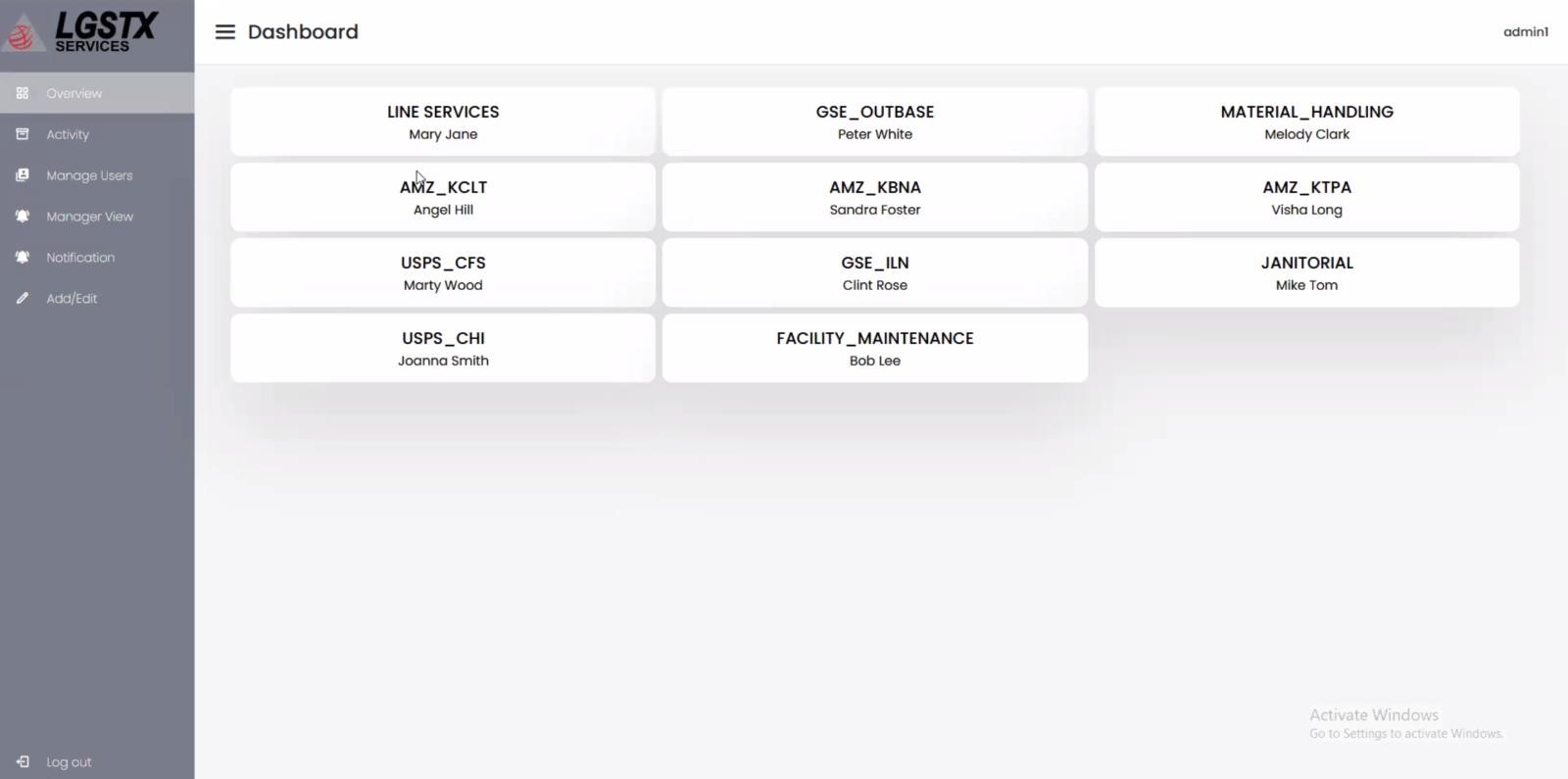
Clicking the logout button will simply take you to the login screen again.



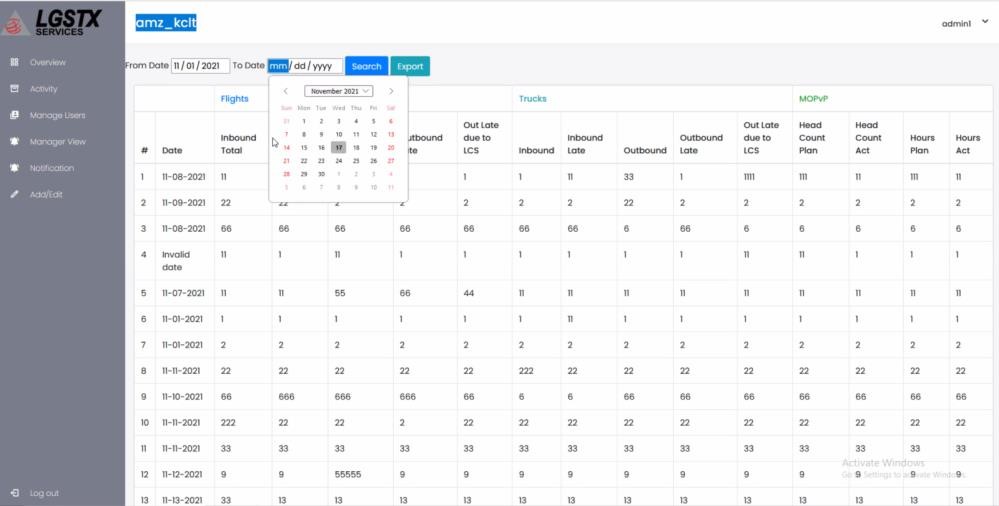


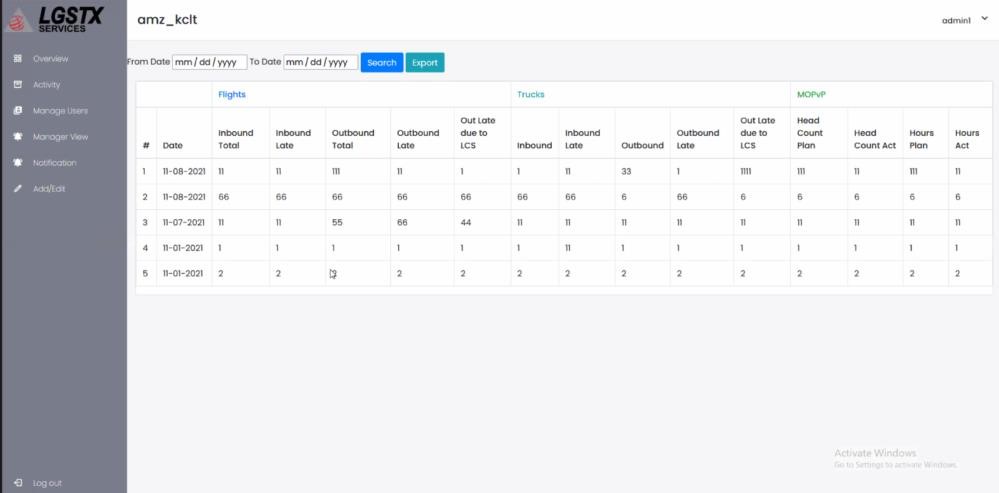
# Navigate Through Admin-View

Logging in as admin will have a dashboard view to access all business trades.



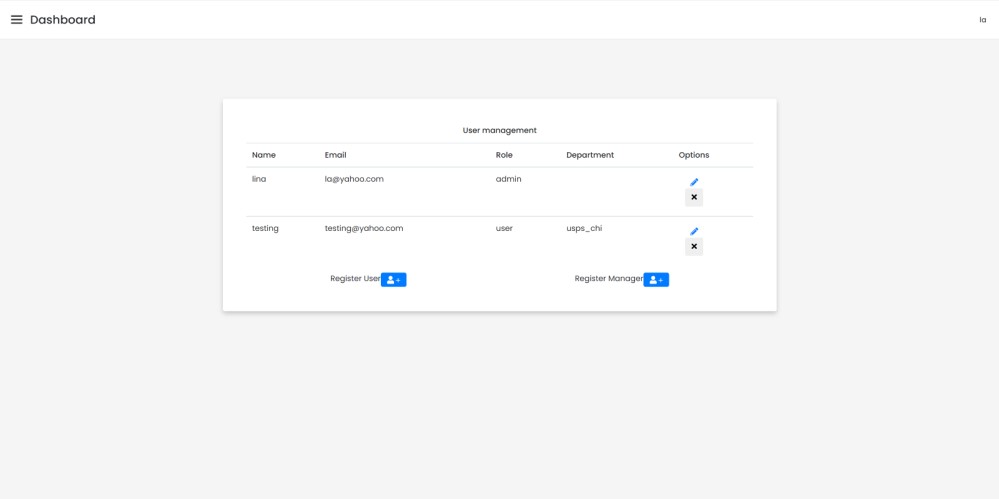
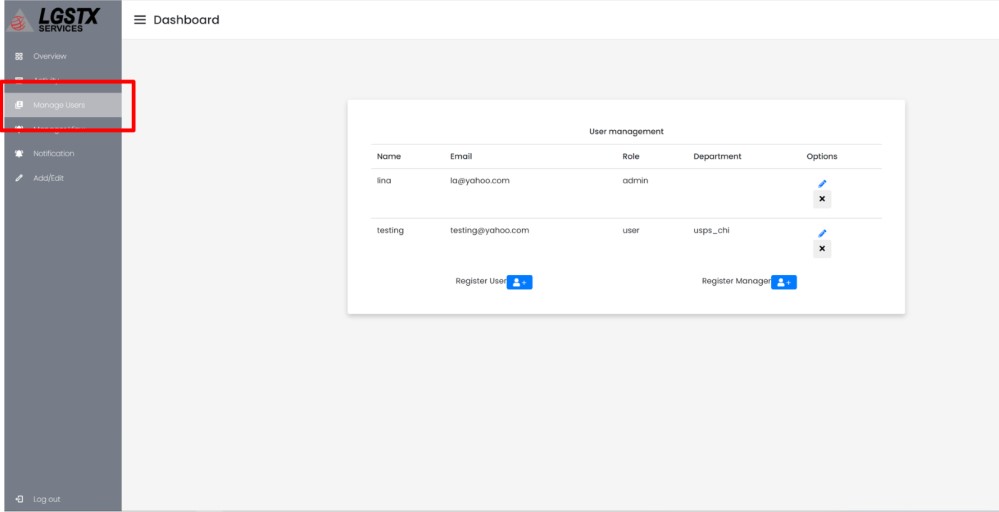
Clicking on any of the section should take you to their template. The admin should be able to pick a date to view the report





# User Management

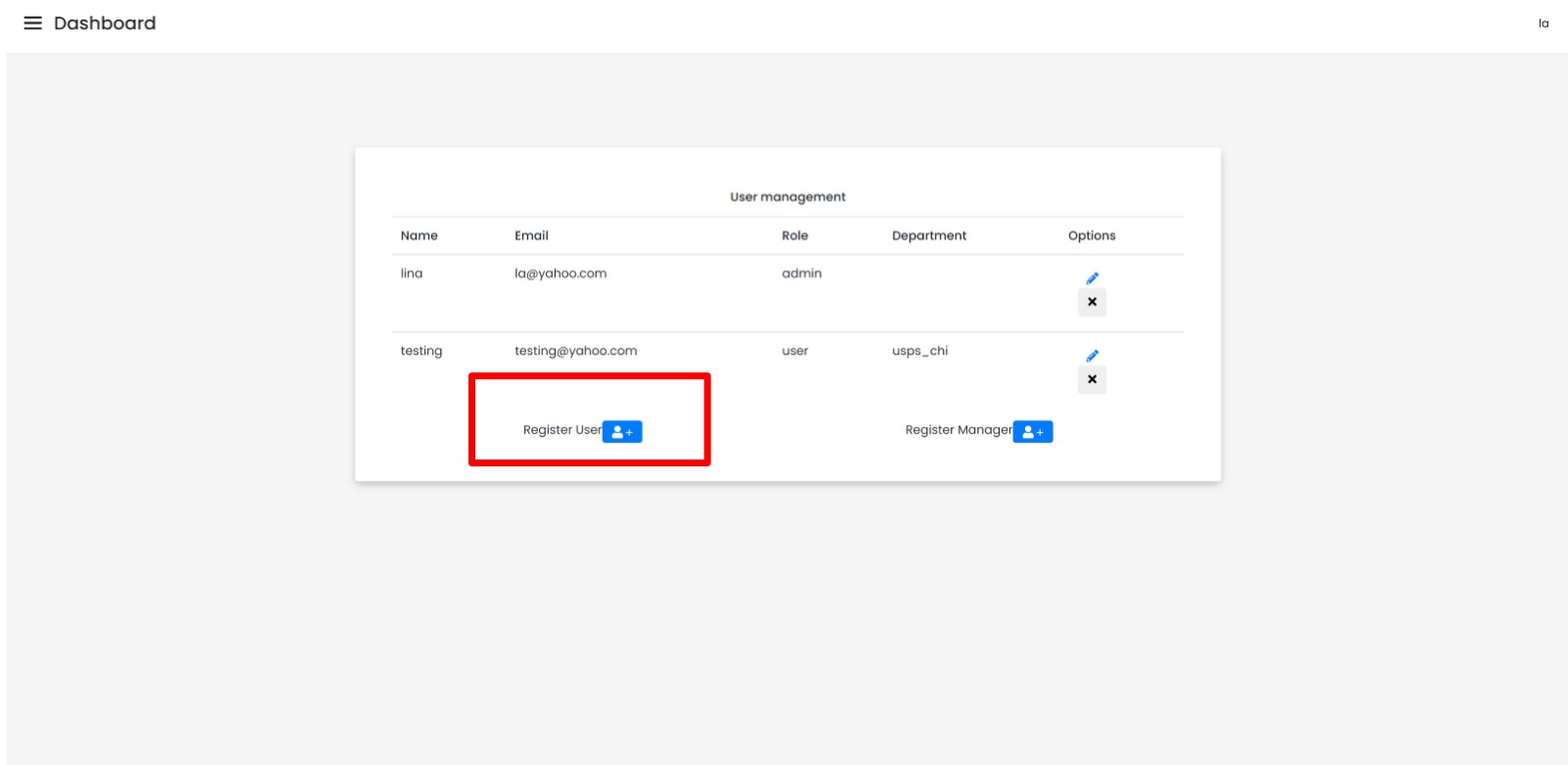
Clicking on manage user from the menu bar to register new users.



Admin only should be able to register new users and mangers.

# Add User

Click register user to sign up new user

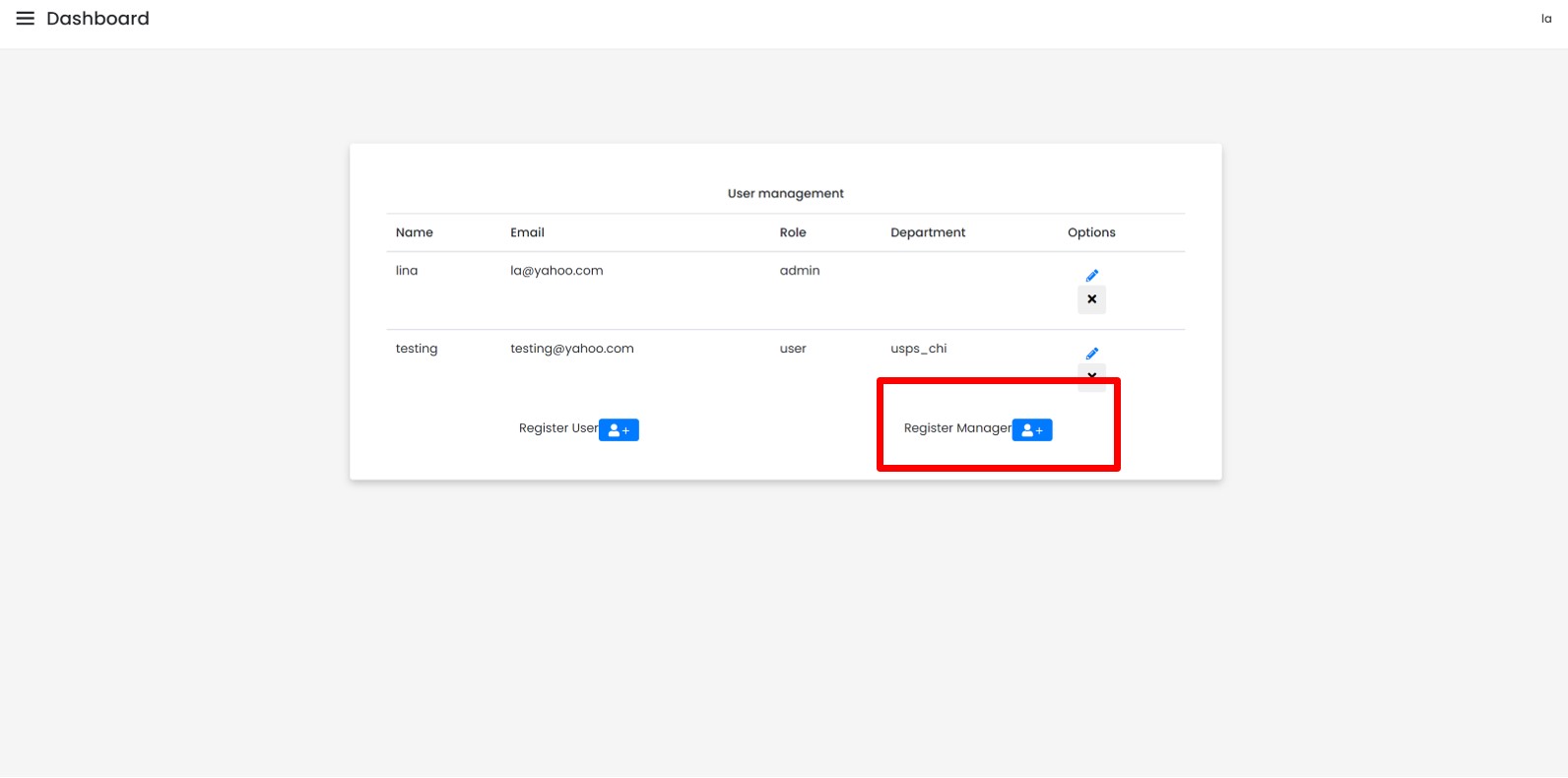




Fill out the required information and click the register. Do not forget to choose from the drop-down menu which section you are signing up for

# Add Manger

Click on register manger to sign up new manger

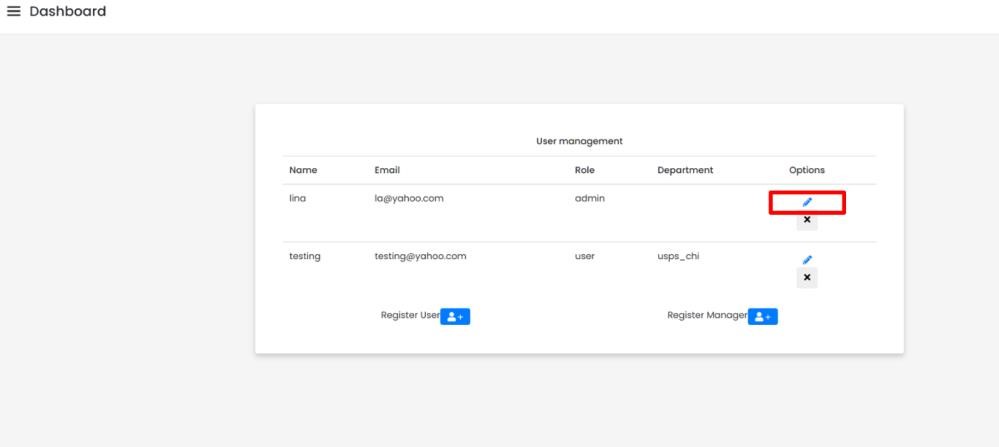




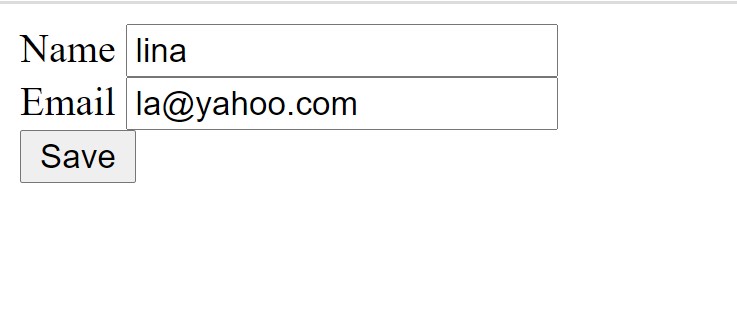
Fill out the required information and click the register button.

# Edit User

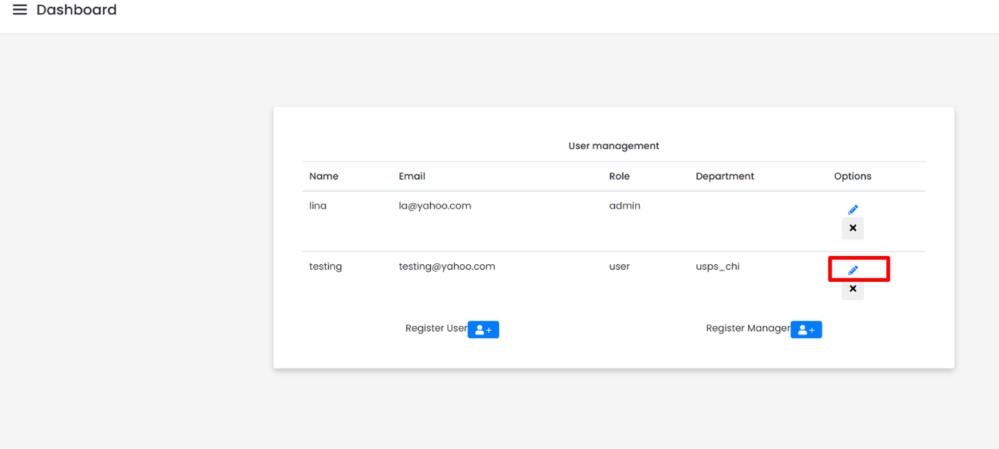
Admin can click on the pen to edit the user information.



It will take you to this page to edit. Click save when done.



# Edit manager

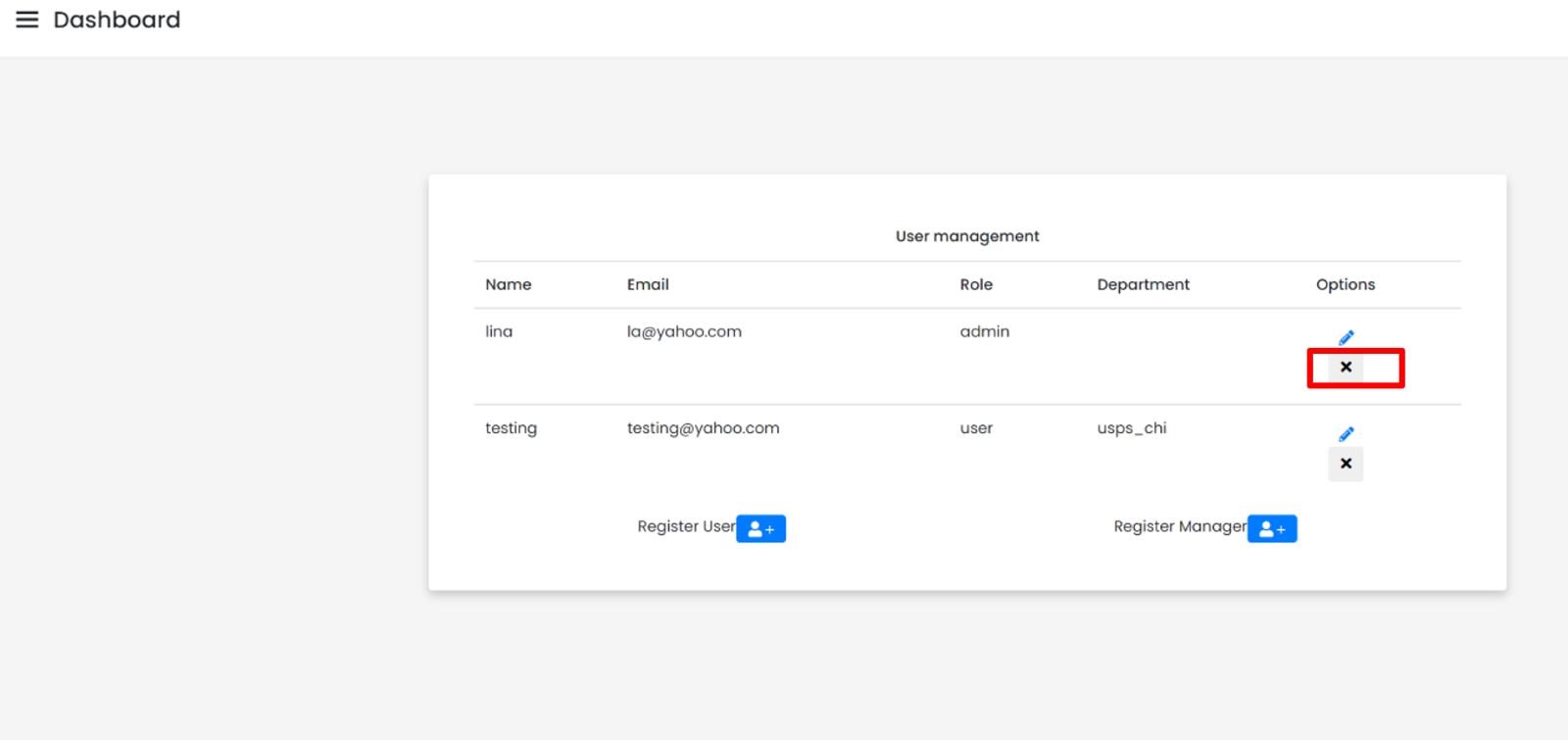


It will take you to this page to edit manger information. Click save when done editing.



# Delete User

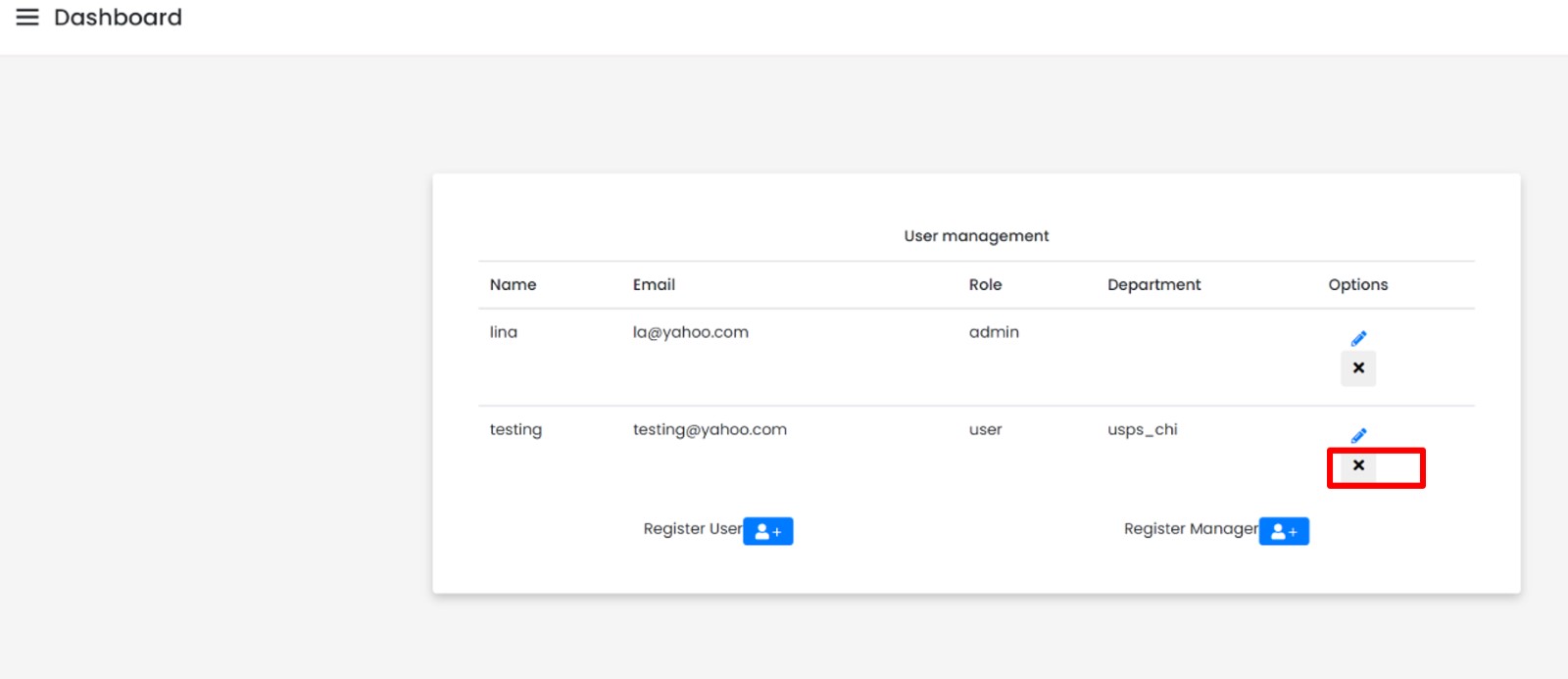
Admin can delete user by clicking the X button



This will delete them permanently

# Delete Manager

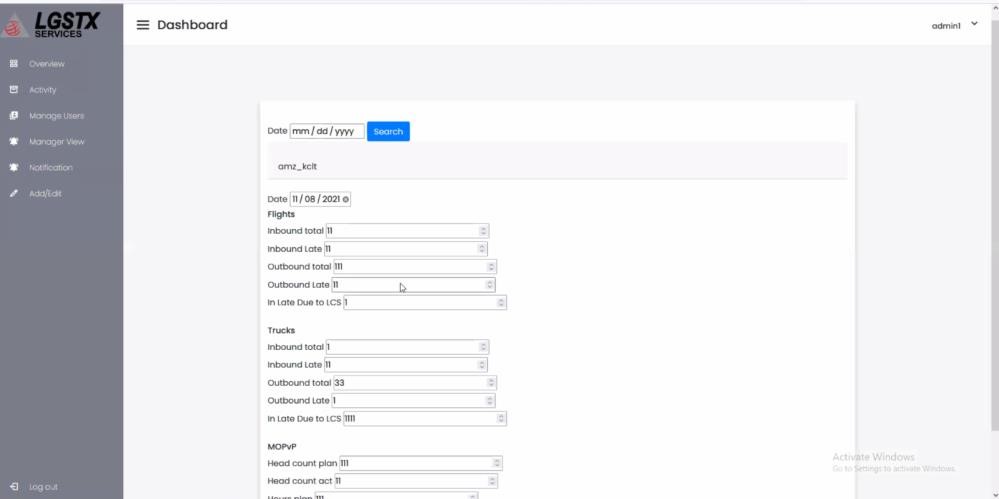
Admin can delete manger by click the X button



This will delete them permanently

# Add/Edit

Admin should be able to edit section if he/she chose to do so by clicking on add/edit tap

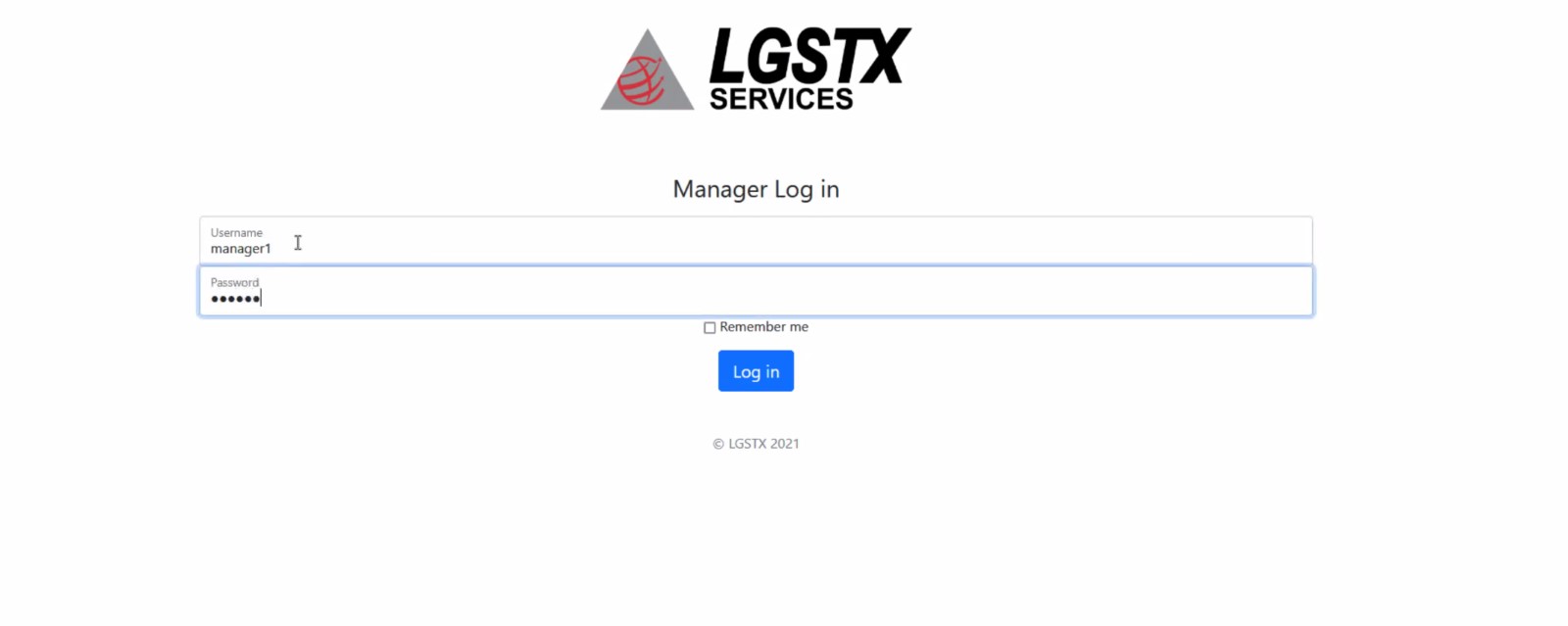


Note:

* The notification and activity tab on admin view was deleted per client request.
* Manger view should take the admin to manger view to look into data visualization

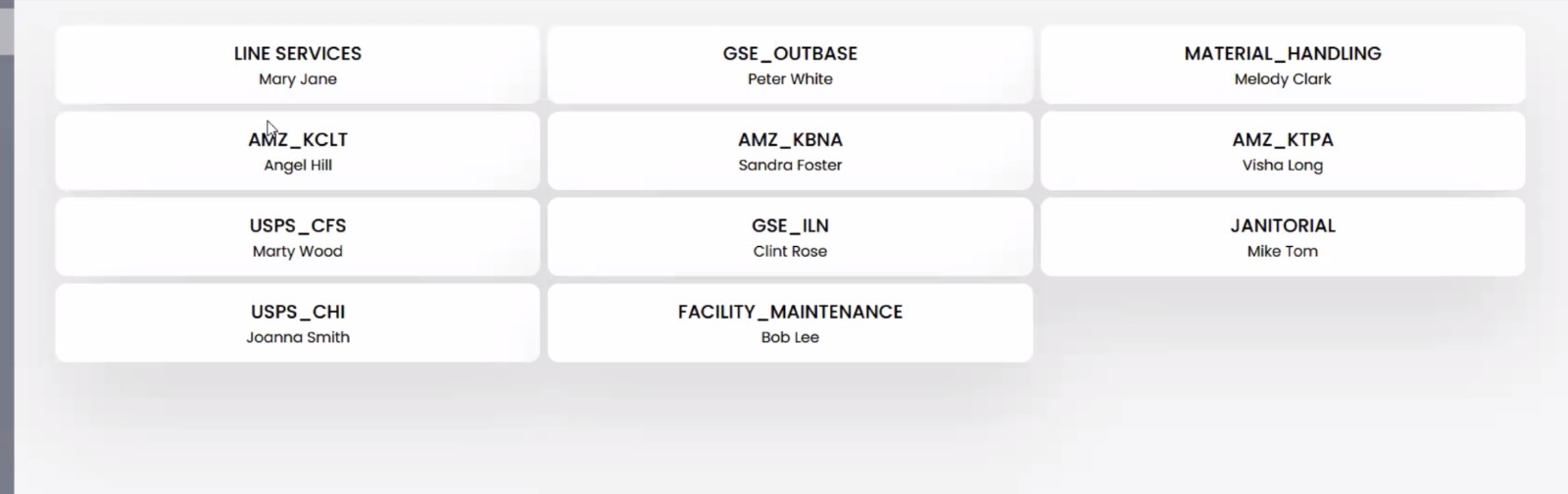
# Navigating Through Manger-View

All managers will have one login provided by admin. They should be able to see all sections data in visualized manner.

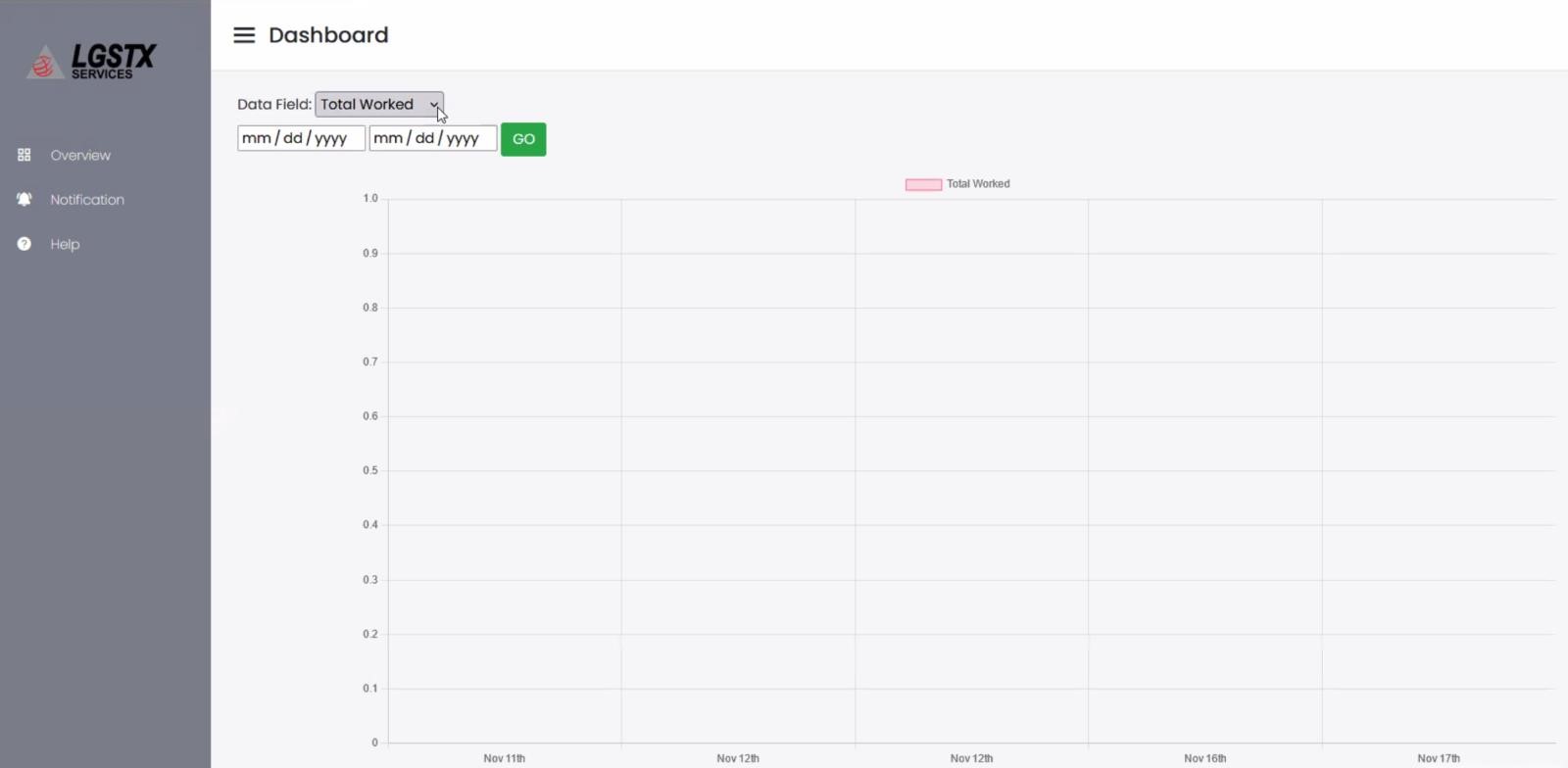


# Manager Overview

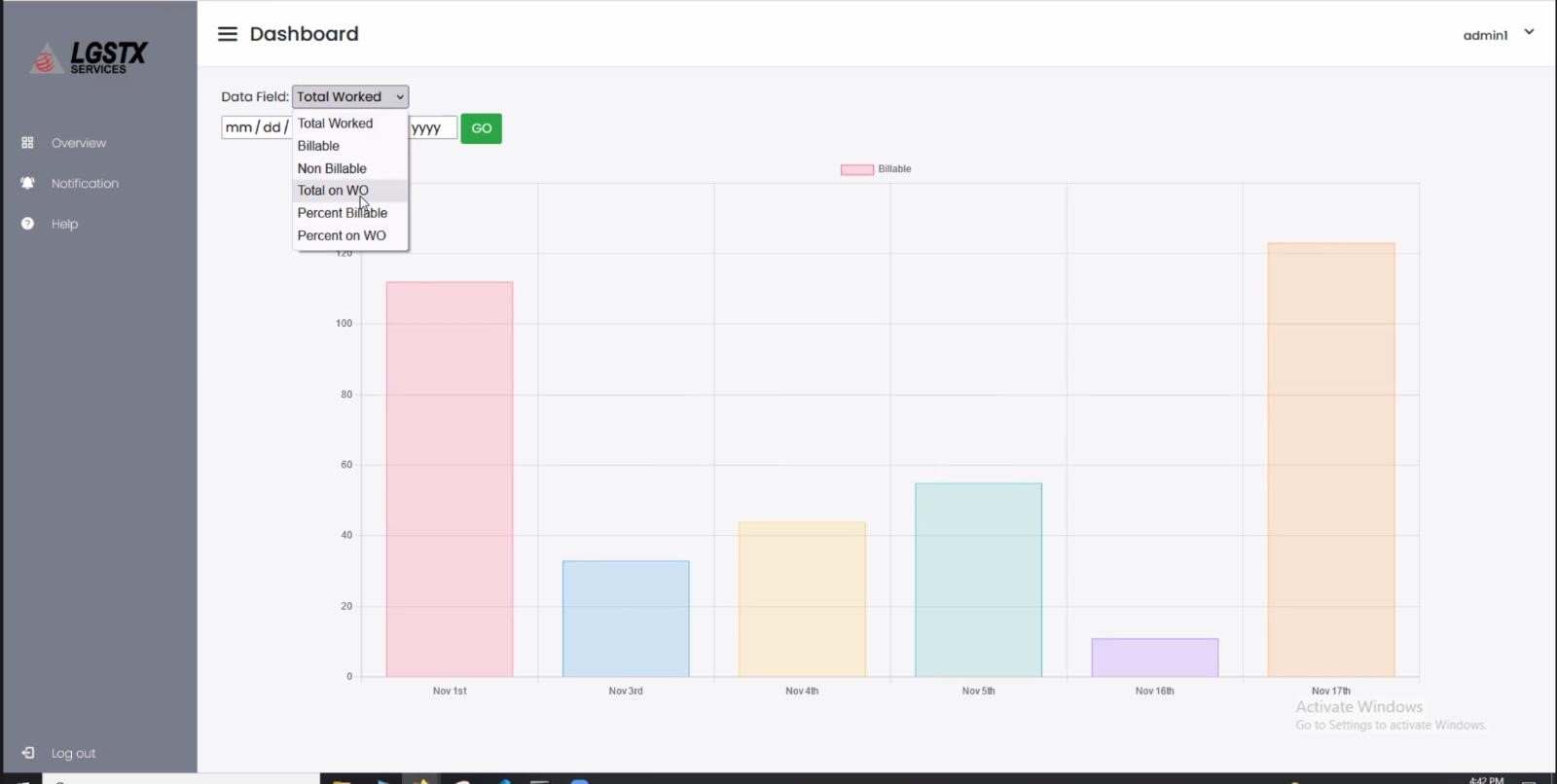
If the manager clicks on overview tab, they should be able to see all section.



Clicking on any section will take you to the page below



Pick data range to view how the data looking like. And pick the field requested well

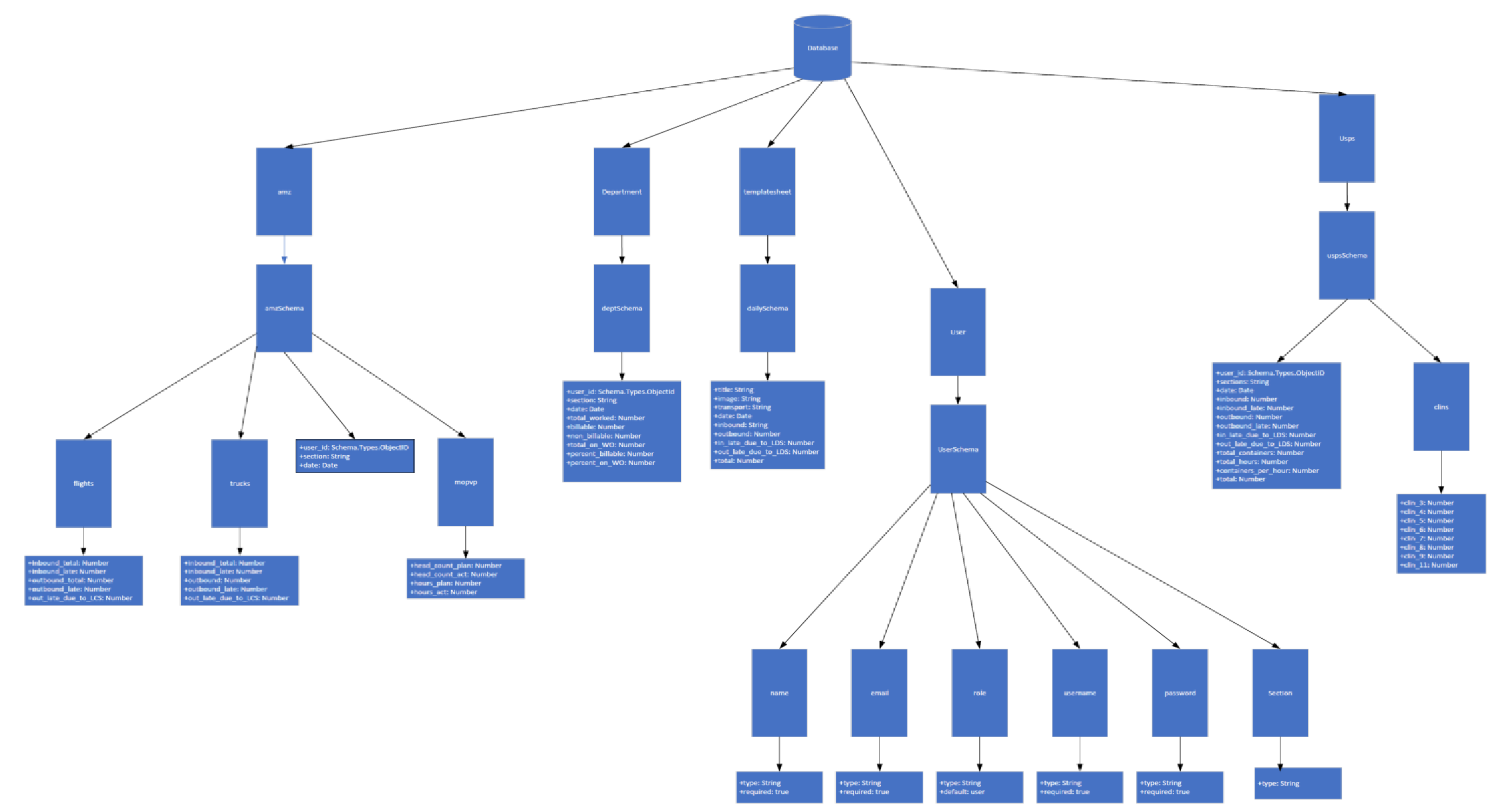


Note the y-axis was provided by the client. Number represents packages number.



Note: notification tab for manager view deleted per client request

# Database

List each table and its elements