

**CMA CGM Shared Service Centre (India) Pvt.Ltd.**

**CIRA F2F SQ CREATION**

**Version 1.1**

**Technical Specification Document**

**Version History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Description** |
| 1.0 | 26-10-2016 | Sunil Wananje | Initial Baseline |
| 1.1 | 12-12-2016 | Sunil Wananje | Enhancement Update |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

***Note:***

*Standardized version numbering convention:*

*Drafts – (Before approval signature)*

*0.#*

*0.#+1*

*0.#+2*

*Published – (Once there is an approval on the document)*

*1.0*

*1.#*

*1.#+1*

**Approval List**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Organization** | **Approval Status** |
| BABU Hemanth Kumar | Deputy Director – IT Development | CMA CGM SSC Chennai | Approved |

**Contents**

[1 Introduction 5](#_Toc421534230)

[1.1 Purpose 5](#_Toc421534231)

[1.2 Scope 5](#_Toc421534232)

[1.3 Definitions & Acronyms 5](#_Toc421534233)

[1.4 References 5](#_Toc421534234)

[2 Design Alternate Solutions 6](#_Toc421534235)

[2.1 Design Alternatives 6](#_Toc421534236)

[2.2 Selection Criteria 6](#_Toc421534237)

[3 Technical Overview 6](#_Toc421534238)

[3.1 Approach 6](#_Toc421534239)

[3.1.1 Process Block : F2F SQ Creation 7](#_Toc421534240)

[3.2 Assumptions 8](#_Toc421534241)

[4 Detailed Design 8](#_Toc421534242)

[4.1 Prerequisites 7](#_Toc421534243)

[4.2 Login 9](#_Toc421534243)

[4.3 Adminstaration 9](#_Toc421534244)

[4.4 Dashboard 9](#_Toc421534244)

[4.5 Indexing 10](#_Toc421534244)

[4.6 Publishing Queue 11](#_Toc421534244)

[4.7 User Process Form 12](#_Toc421534244)

[4.8 All Data/Live Data 12](#_Toc421534244)

[4.9 Auditing Queue 14](#_Toc421534244)

[4.10 Audit Process Form 15](#_Toc421534244)

[4.11 RFI Queue 16](#_Toc421534244)

[4.12 Completed Queue 18](#_Toc421534244)

[4.13 Follow Up Queue 19](#_Toc421534244)

[4.14 Pricer Database 21](#_Toc421534244)

[4.15 Partner Code Database 22](#_Toc421534244)

[4.16 Reports 21](#_Toc421534246)

[4.16.1 Daily Report 23](#_Toc421534247)

[4.16.2 Weekly Report 24](#_Toc421534248)

[4.16.2 Monthly Report 25](#_Toc421534248)

[4.16.2 RFI Log Report 26](#_Toc421534248)

[4.16.2 CAPA Report 27](#_Toc421534248)

[4.16.2 Error Report 27](#_Toc421534248)

[4.16.2 Productivity Report 28](#_Toc421534248)

[4.17 Entity Relationship Diagram 29](#_Toc421534246)

[4.17.1 Stored Procedure 29](#_Toc421534248)

[5 Interfaces 29](#_Toc421534249)

[6 Error Handling 29](#_Toc421534250)

[7 Installation Requirements 30](#_Toc421534251)

# Introduction

This document is designed to be a reference for any person wishing to implement or any person interested in the architecture of the **CIRA F2F SQ Creation** application or the **CIRA\_ F2F** database. This document describes application’s architecture and sub-architecture their associated interfaces, database schemas, and the motivations behind the chosen design.

## Purpose

This document provides a comprehensive overview of the design approach for the development of **CIRA F2F SQ Creation** application. It is intended to capture and convey the significant design decisions that have been made on the system during the design analysis to proceed with the development as expected by the business team.

## Scope

This Document specifies main components and services of the **CIRA F2F SQ Creation**, such as Dashboard, Adminstration, Indexing, Process, Audit, RFI, Follow Up and Pricer. The architecture presented here affects and influences all the work packages of the **CIRA F2F SQ Creation** project. It is also shaped by the requirements of the **CIRA F2F SQ Creation**.

## Definitions & Acronyms

|  |  |
| --- | --- |
| **Acronym** | **Definition** |
| ISR | Inside Sales Representative |
| UPF | User Process Form |
| APF | Audit Process Form |
| TAT | Turn-Around-Time |
| OOT | Out Of TAT |
| CAPA | Corrective Action & Preventive Action |
| MQC | Monthly Quantity Contribution |
| NVOCC | Non Vessel Operators Common Carriers |
| FAK | Freight of All Kind |
| MWR | Mid-West Region |
| NER | North East Region |
| SAR | South Atlantic Region |
| WCR | West Coast Region |
| F2F | Foreign 2 Foreign |
| POL | Port Of Loading |
| POD | Port Of Discharge |
| SQ | Special Quotation |

## References

| **Name** | **Business Group** | **Key Activities** |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Design Alternate Solutions

## Design Alternatives

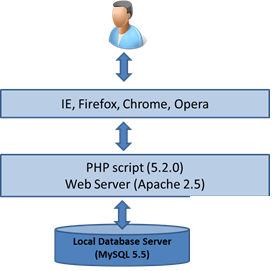
*N/A*

## Selection Criteria

*N/A*

# Technical Overview

## Approach

**

***Presentation Layer***

*• Decoding URLs: protocol/host/file (*[*http://10.13.90.26/cira\_*](http://10.13.90.26/cira_)*f2f)*

*• Host name converted to IP address (10.13.90.26)*

*• Issue request to remote server using HTTP protocol (usually HTTP)*

*• accept the returned HTML file*

*• Issue requests for any embedded links*

*• allow plug-ins to handle new file types*

*• execute client-side scripts in JavaScript*

*• support interaction between client-side scripts and the web page (DHTML)*

*• Accept user input via a variety of controls on a form.*

***Application Layer***

*•* ***Server*** *(Apache)*

*– Identifying appropriate action to take – fetch a file, pass request to an interpreter*

*– Sending output back to caller in MIME package*

*– Support for:*

*• Thousands of concurrent users*

*• Multi-threading [allow multiple processes to run concurrently]*

*• caching [holding results in a temporary store to reduce re-calculation]*

*•* ***Server Script*** *(PHP)*

*– Interacting with the server (accessing input and generating output)*

*– interpreting the requests according to business rules and past transactions from this client*

*– requesting the appropriate data from the Persistence layer*

*– computing derived data*

*– creating the HTML (or GIF, MIDI...) for the page*

***Persistence Layer***

*• Interaction with the database using standard languages e.g. SQL queries using database-specific protocol over TCP/IP*

*• define and modify the data structures (e.g. tables) themselves (the Database Schema)*

*• insert, update and delete data*

*• maintain data persistently, with backup and recovery*

*• Handle transactions to support concurrent access to the database via locking etc.*

*Optimise access by compilation of queries, indexing, replication of tables etc.*

### 

### Process Block : F2F SQ Creation

|  |  |
| --- | --- |
| **Folder Type** | **Action Items** |
| Inbox | All the request would be received in inbox.  Indexing will happen & items would be assigned to respective folder as mentioned in detailfor action |

|  |  |
| --- | --- |
| Request | All the request will moved in request folder by indexer |

|  |  |
| --- | --- |
| Query | If the user have any query, he will rais the query and move the related mail in query folder |

|  |  |
| --- | --- |
| Audit | Once request is done by user, he will move the mail tom audit |

|  |  |
| --- | --- |
| Follow Up | All the request which are sent to pricer status comes in follow up queue for sending reminder mail |

|  |  |
| --- | --- |
| Complete | Once the auditor, audits the request and sent to sent to customer then it will move to completed queue |

## Assumptions

*Some assumptions or mandatory things we are using in our application. They are as follows:-*

1. *There are various types of status category but 1st must be user process form and 2nd should be Audit Process form that should have 1 and 2 id in mst\_status*
2. *Status should be IN PROCESS, PENDING IN, PENDING OUT, SEND TO AUDIT, DONE, DISREGARD in msd\_status table.*

# Detailed Design

## Prerequisites

Administration Module is the Place where all the basic settings needed for other modules are carried out. That is the dynamic settings are done here. User

Please find the Block Diagram for Administration Module below,

## Login

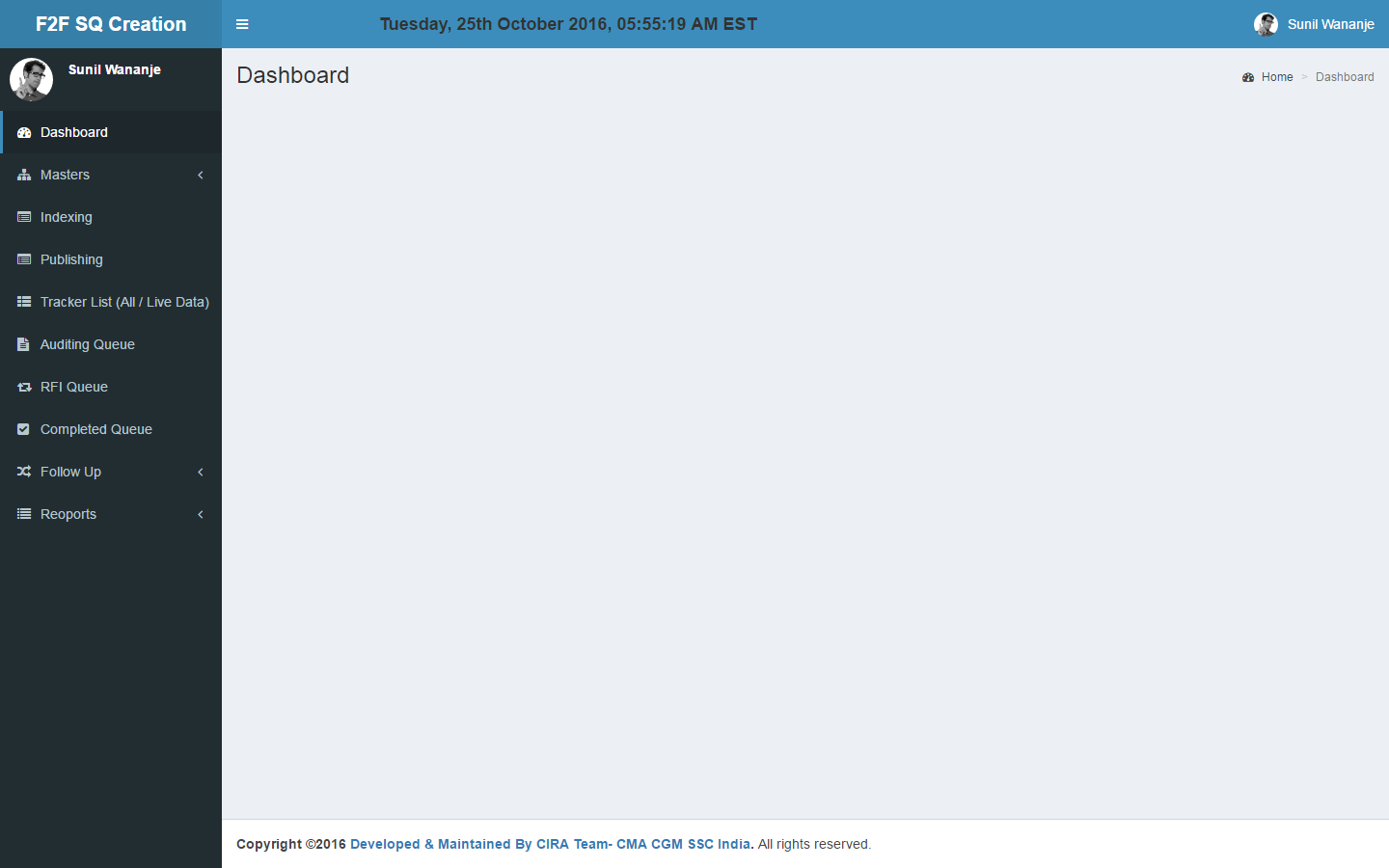


## Administrations

Administration Menu is the one where all the Basic settings needed in CIRA is Carried Out. Ex: User Access, Region, Holiday List for Specific Countries, Status, Modes, Pricing Area, TAT etc.

## Dashboard

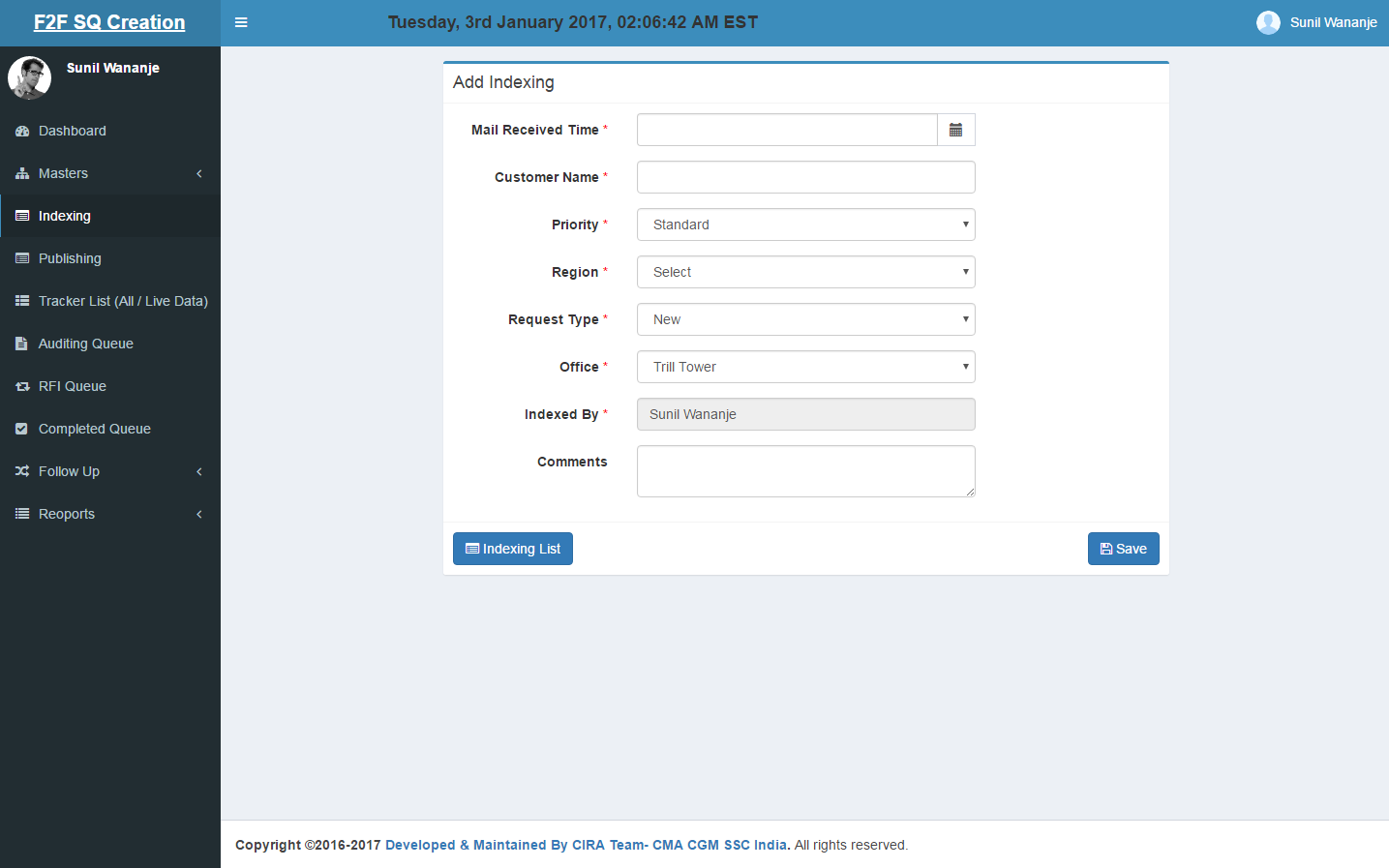
When Admin login successfully then user see dashboard page following screen



## Indexing

Any user which has user or admin access can create indexing. The screen is come under indexing section. Following points are added in enhancement.

* Copy paste option is enabled for mail received date time field.
* Name of the customer field able to move cursor right and left.
* Default "TRILL TOWER” value is selected for office field.

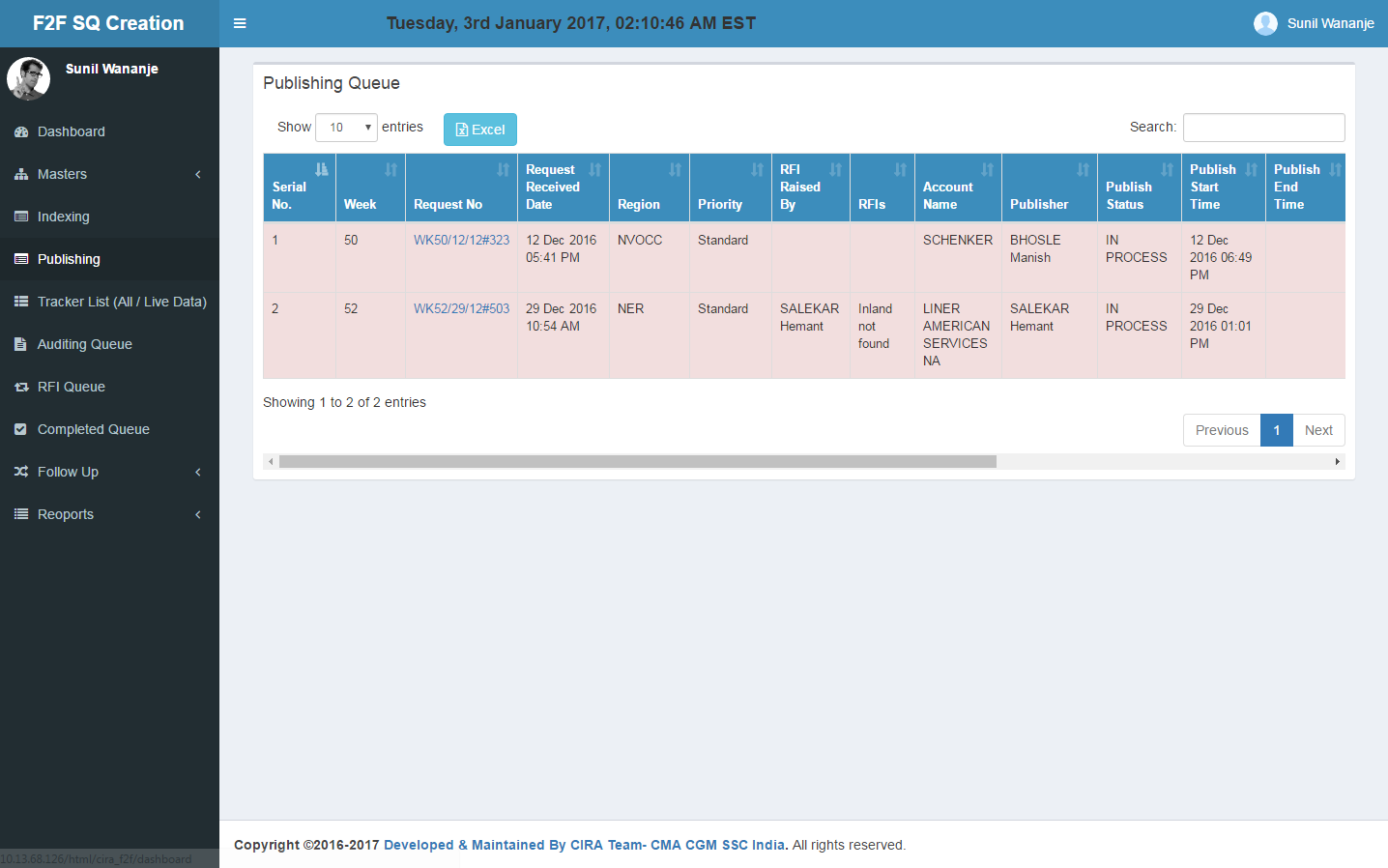


|  |  |
| --- | --- |
| **Class/***Controller Function* | **Query Used** |
| **App\Http\Controllers\IndexingController** /*create* | * select \* from `mst\_priority\_type` * select \* from `mst\_region` * select \* from `mst\_request\_type` * select \* from `mst\_office` |
| **Table Used** | * **mst\_priority\_type** table is used for fetching priority in dropdown i.e standard and urgent. * **mst\_region** table is used for fetching regions in dropdown. * **mst\_reuest\_type** table is used for fetching request type in dropdown. * **mst\_office** table is used for fetching registerd office in dropdown. |

## Publishing Queue

In this section all request we can see and after clicking on request no we can publish using User process form.

* Export to excel option is added for publishing queue.
* Unlock the request option for other user is added for admin user only.



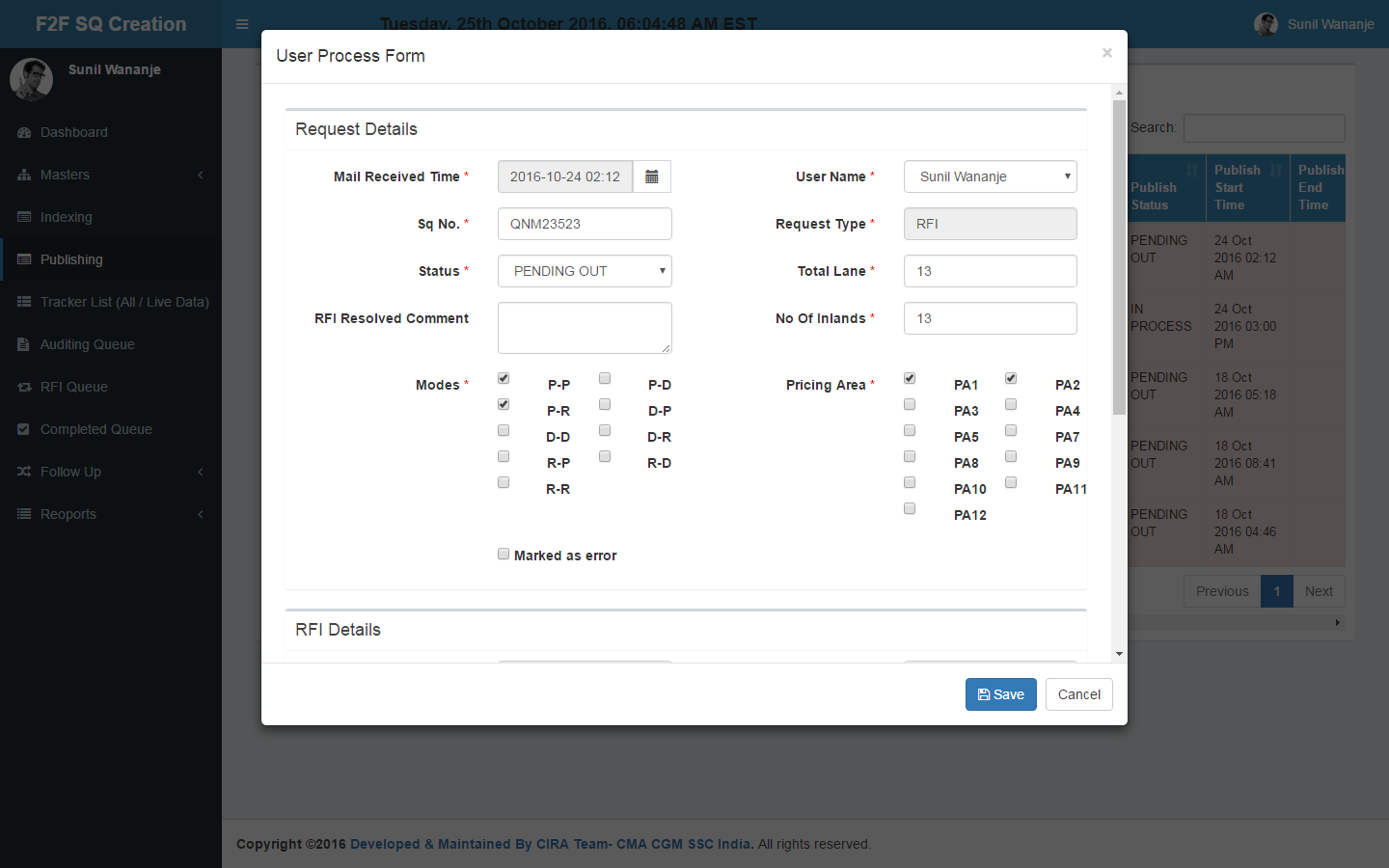
|  |  |
| --- | --- |
| **Class/***Controller Function* | **Query Used** |
| **App\Http\Controllers\PublishingController** /*index* | select `process\_queue`.\*, `indexing`.\*, `indexing`.`id` as `indexing\_id`, `mst\_rfi\_type`.`rfi\_type\_name` as `rfi\_name`, `mst\_error\_cat`.`name` as `err\_cat\_name`, `mst\_error\_type`.`name` as `err\_type\_name`, `users`.`name` as `rfi\_by\_name`, `err\_user`.`name` as `error\_done\_by\_name`, `publisher`.`name` as `publish\_by\_name`, `mst\_status`.`status\_name`, `process\_queue`.`id` as `process\_queue\_id` from `indexing` left join `process\_queue` on `indexing`.`id` = `process\_queue`.`indexing\_id` left join `mst\_modes` on `mst\_modes`.`id` = `process\_queue`.`mode\_id` left join `mst\_rfi\_type` on `mst\_rfi\_type`.`id` = `process\_queue`.`rfi\_type\_id` left join `mst\_status` on `mst\_status`.`id` = `process\_queue`.`status\_id` left join `mst\_error\_cat` on `mst\_error\_cat`.`id` = `process\_queue`.`error\_cat\_id` left join `mst\_error\_type` on `mst\_error\_type`.`id` = `process\_queue`.`error\_type\_id` left join `users` on `users`.`id` = `process\_queue`.`rfi\_raised\_by` left join `users` as `err\_user` on `err\_user`.`id` = `process\_queue`.`error\_done\_by` left join `users` as `publisher` on `publisher`.`id` = `process\_queue`.`publish\_by` where `mst\_status`.`status\_name` not in ('pending in', 'sent to audit', 'done', 'disregard') or `mst\_status`.`status\_name` is null order by `indexing`.`priority\_id` desc, `indexing`.`mail\_received\_time` desc |
| **Table Used** | * **indexing** is used for fetching all the indexer for fetching request no, region priority, request type. * **process\_queue** table is used for fetching publish data join with indexing. * **mst\_rfi\_type, mst\_error\_cat, mst\_error\_type, mst\_status, mst\_user, mst\_priority\_type** these table joining with **indexing** and **process\_queue** table to get their respective name for lsting. |

## User Process Form

After clicking on request no in publishing queue User Process Form will pop up.

Following point is added in enhancement.

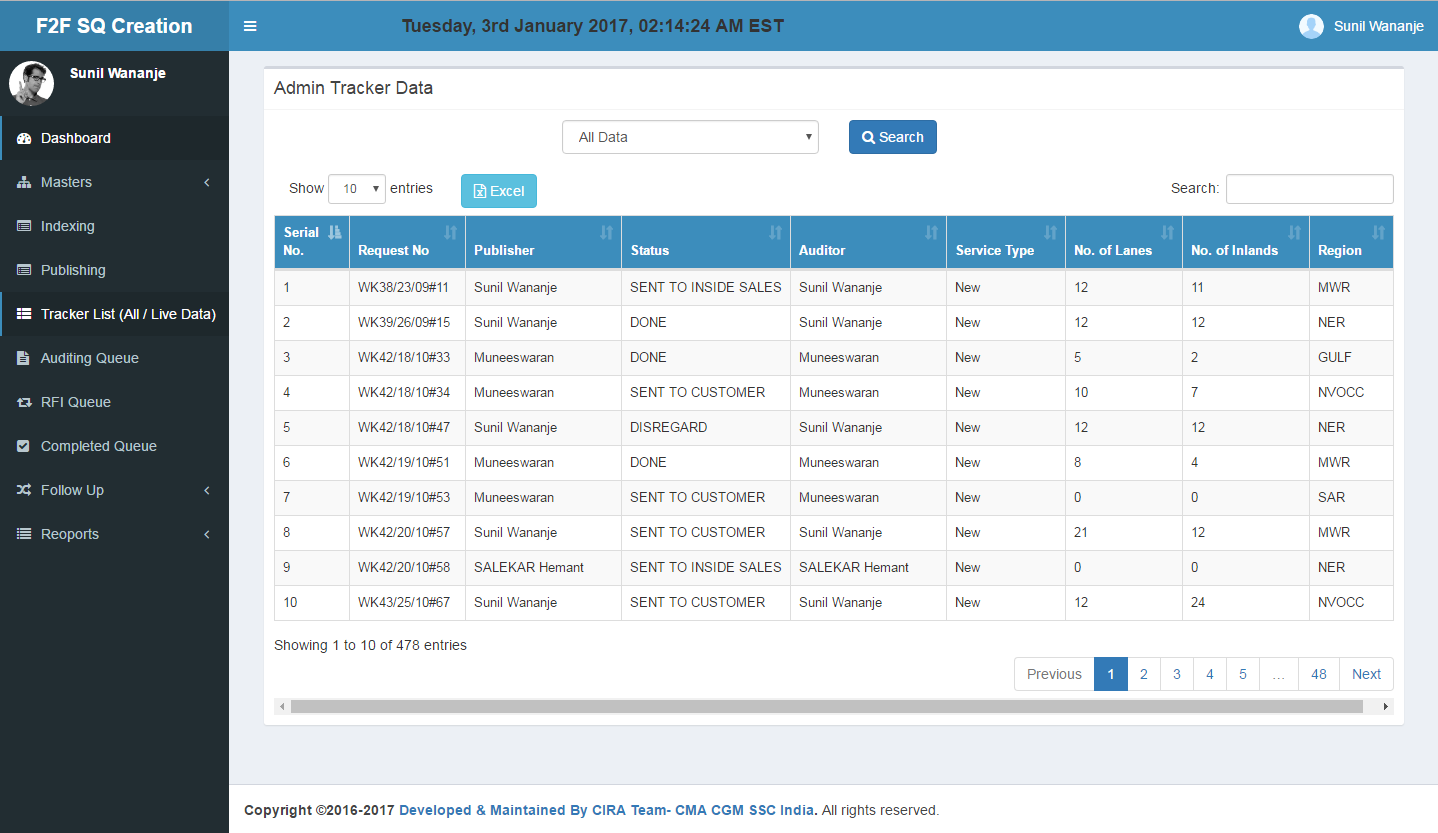
* ISR field is added when status is pending in.
* When the user selects "Sent to Audit", "Done" or "Disregard" in the UPF it is mandatory for the users to update "Total lanes" and "Modes".
* If the user selects the status as Done in UPF it is move to “Follow-up queue” and the same added in Daily Report => No. of Request sent to Customer



## All Data/Live Data

Admin Track is the screen used to track the activity status based on the user in Publishing and Auditing under the categorization of All Data & Live Data

* **ALL Data :** It consist of all the request status
* **LIVE Data :** It consist of the all the request which are "To Be Started", "In Process" and "Pending Out”

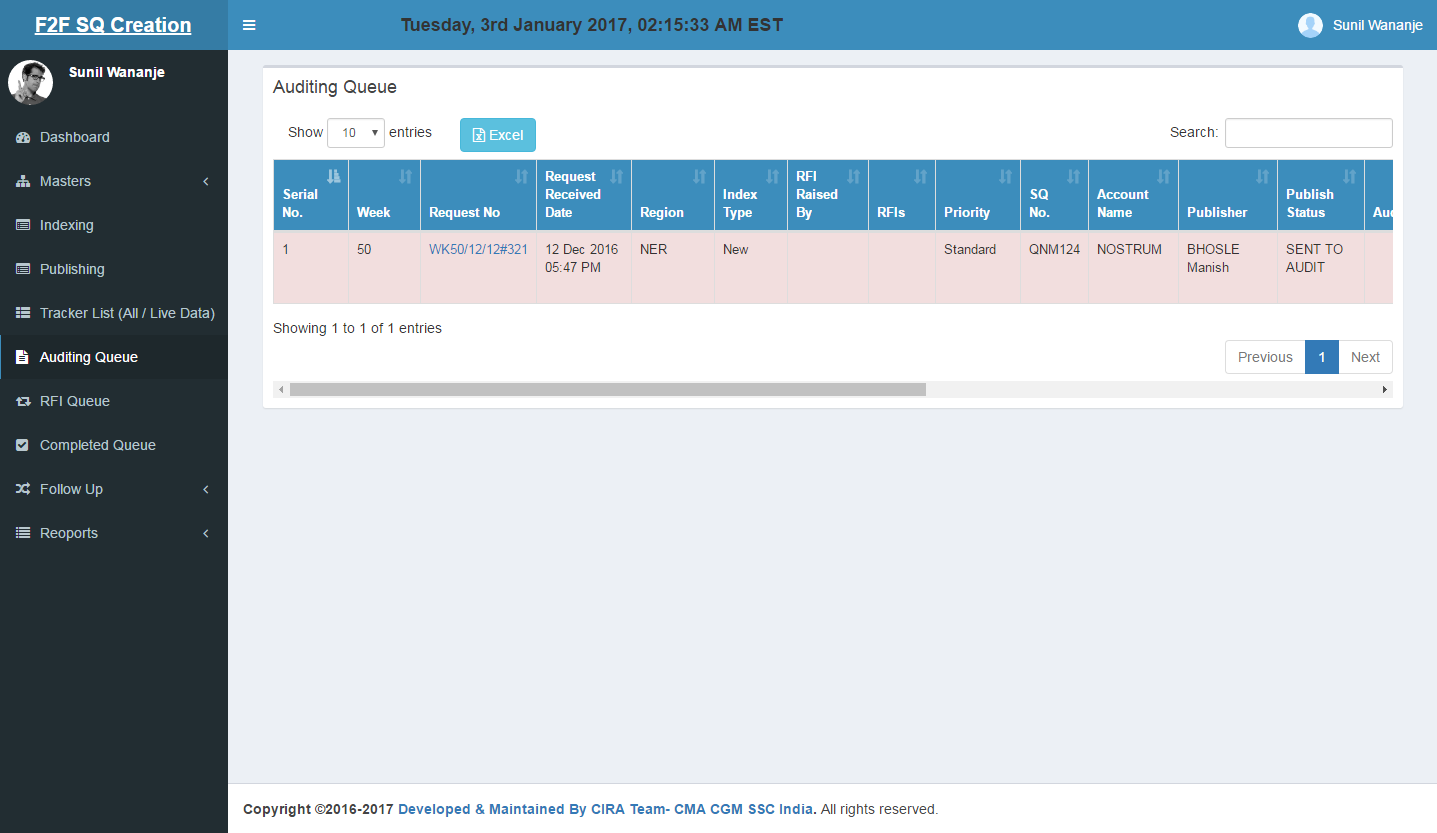


|  |  |
| --- | --- |
| **Class/***Controller Function* | **Query Used** |
| **App\Http\Controllers\PublishingController** /*index* | select `rfi\_queue`.\*, `mst\_region`.`name` as `region\_name`, `mail\_received\_time`, `indexing\_tat`, `request\_no`, `customer\_name`, `audit\_queue`.`audit\_rfi\_type\_id`, `audit\_queue`.`audit\_rfi\_description`, `audit\_queue`.`audit\_rfi\_raised\_by`, `process\_queue`.`rfi\_type\_id`, `process\_queue`.`rfi\_description`, `process\_queue`.`rfi\_raised\_by`, `auditor`.`name` as `audit\_rfi\_user`, `publisher`.`name` as `publish\_rfi\_user`, `mst\_rfi\_type`.`rfi\_type\_name` as `process\_rfi\_type`, `aq\_rfi`.`rfi\_type\_name` as `audit\_rfi\_type`, `pq\_status`.`status\_name` as `process\_status`, `aq\_status`.`status\_name` as `audit\_status` from `rfi\_queue` inner join `indexing` on `indexing`.`id` = `rfi\_queue`.`indexing\_id` left join `process\_queue` on `process\_queue`.`id` = `rfi\_queue`.`process\_queue\_id` left join `audit\_queue` on `audit\_queue`.`id` = `rfi\_queue`.`audit\_queue\_id` left join `mst\_region` on `mst\_region`.`id` = `indexing`.`region\_id` left join `users` as `auditor` on `auditor`.`id` = `audit\_queue`.`audit\_rfi\_raised\_by` left join `users` as `publisher` on `publisher`.`id` = `process\_queue`.`rfi\_raised\_by` left join `mst\_rfi\_type` on `mst\_rfi\_type`.`id` = `process\_queue`.`rfi\_type\_id` left join `mst\_status` as `aq\_status` on `aq\_status`.`id` = `audit\_queue`.`audit\_status\_id` left join `mst\_status` as `pq\_status` on `pq\_status`.`id` = `process\_queue`.`status\_id` left join `mst\_rfi\_type` as `aq\_rfi` on `aq\_rfi`.`id` = `audit\_queue`.`audit\_rfi\_type\_id` where `rfi\_queue`.`rfi\_status` = '1' |
| **Table Used** | * **indexing** is used for fetching all the indexer for fetching request no, region priority, request type. * **process\_queue** table is used for fetching publish data join with indexing. * **auditing** table is used for fetching auditing details joining with **indexing** and **process\_queue** table. * **mst\_rfi\_type, mst\_error\_cat, mst\_error\_type, mst\_status, mst\_user, mst\_priority\_type** these table joining with **indexing** , **process\_queue** and **auditing** table to get their respective name for lsting. |

## Auditing Queue

In this section all request which has status “**send to audit”** that only we can see and after clicking on request no we can do Audit process form.

* Export to excel option is added for auditing queue.
* Unlock the request option for other user is added for admin user only.

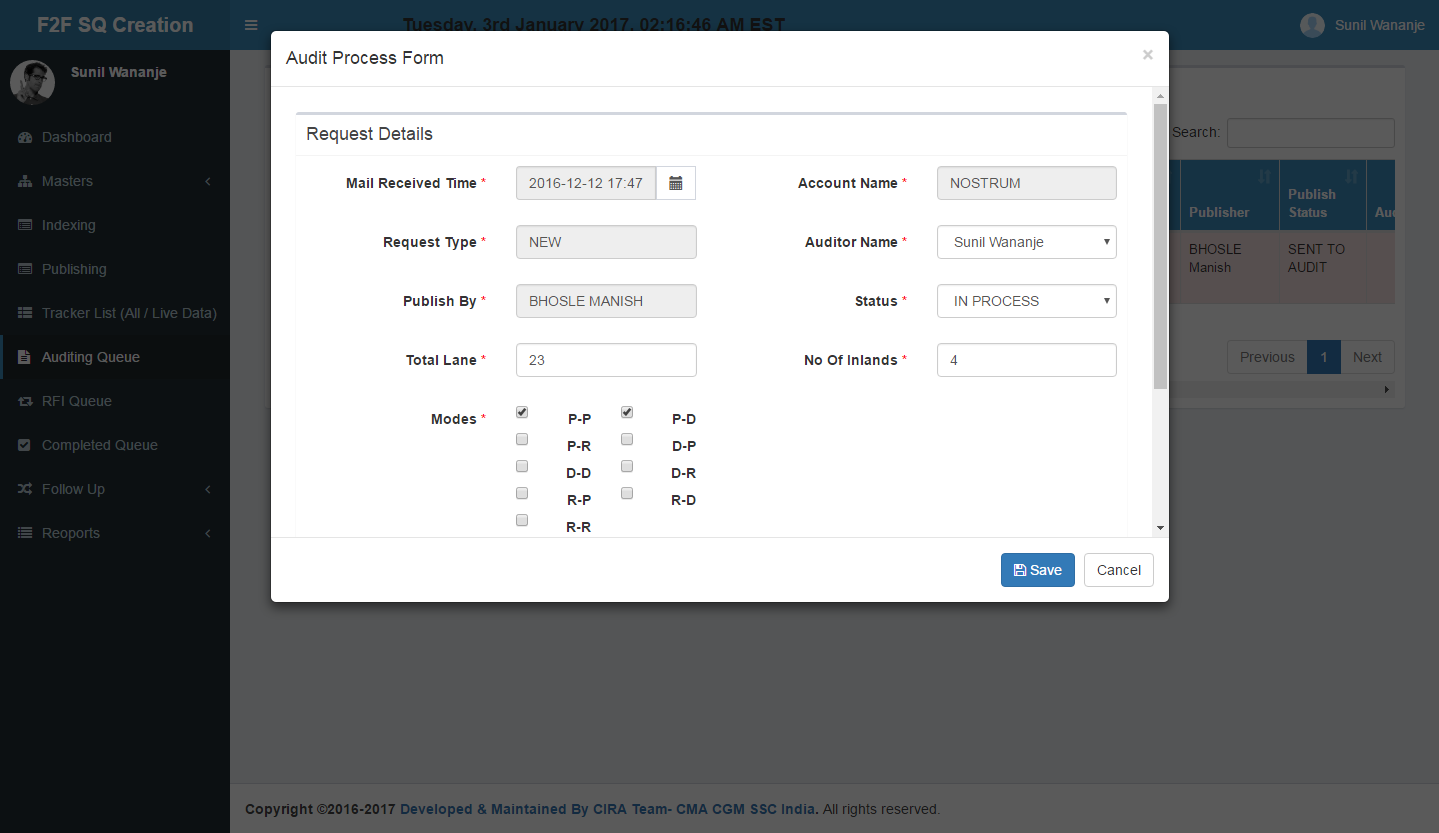


|  |  |
| --- | --- |
| **Class/***Controller Function* | **Query Used** |
| **App\Http\Controllers\AuditingController** /*index* | select `audit\_queue`.\*, `process\_queue`.`sq\_no`, `process\_queue`.`total\_lane`, `process\_queue`.`no\_of\_inlands`, `process\_queue`.`mode\_id`, `process\_queue`.`pricing\_area`, `partner\_code\_db\_id`, `process\_queue`.`publish\_by`, `indexing`.`id` as `indexing\_id`, `indexing`.`mail\_received\_time`, `indexing`.`indexing\_tat`, `indexing`.`request\_no`, `indexing`.`customer\_name`, `indexing`.`priority\_id`, `indexing`.`region\_id`, `indexing`.`office\_id`, `indexing`.`indexed\_by`, `indexing`.`request\_type\_id`, `mst\_rfi\_type`.`rfi\_type\_name` as `audit\_rfi\_name`, `mst\_error\_cat`.`name` as `audit\_err\_cat\_name`, `mst\_error\_type`.`name` as `audit\_err\_type\_name`, `auditor`.`name` as `auditor\_name`, `corrector`.`name` as `corrector\_name`, `publisher`.`name` as `publisher\_name`, `resolver`.`name` as `rfi\_resolver\_name`, `questioner`.`name` as `rfi\_questioner\_name`, `mst\_status`.`status\_name` as `audit\_status\_name`, `publish\_status`.`status\_name` as `publish\_status\_name`, `process\_queue`.`id` as `process\_queue\_id` from `indexing` inner join `process\_queue` on `indexing`.`id` = `process\_queue`.`indexing\_id` inner join `audit\_queue` on `audit\_queue`.`process\_queue\_id` = `process\_queue`.`id` inner join `users` as `publisher` on `publisher`.`id` = `process\_queue`.`publish\_by` left join `mst\_status` on `mst\_status`.`id` = `audit\_queue`.`audit\_status\_id` left join `mst\_status` as `publish\_status` on `publish\_status`.`id` = `process\_queue`.`status\_id` left join `users` as `auditor` on `auditor`.`id` = `audit\_queue`.`audit\_by` left join `users` as `corrector` on `corrector`.`id` = `audit\_queue`.`audit\_error\_done\_by` left join `users` as `resolver` on `resolver`.`id` = `audit\_queue`.`audit\_rfi\_resolved\_by` left join `users` as `questioner` on `questioner`.`id` = `audit\_queue`.`audit\_rfi\_raised\_by` left join `mst\_rfi\_type` on `mst\_rfi\_type`.`id` = `audit\_queue`.`audit\_rfi\_type\_id` left join `mst\_error\_cat` on `mst\_error\_cat`.`id` = `audit\_queue`.`audit\_error\_cat\_id` left join `mst\_error\_type` on `mst\_error\_type`.`id` = `audit\_queue`.`audit\_error\_type\_id` where `mst\_status`.`status\_name` not in ('pending in', 'sent to pricer', 'sent to customer', 'sent to inside sales', 'done', 'disregard') or `mst\_status`.`status\_name` is null order by `indexing`.`priority\_id` desc, `indexing`.`mail\_received\_time` desc |
| **Table Used** | * **auditing** table is join with **indexing** and **process\_queue**  to get auditing, indexing and publishing details. * **mst\_rfi\_type, mst\_error\_cat, mst\_error\_type, mst\_status, mst\_user, mst\_priority\_type** these table joining with **indexing** , **process\_queue** and **auditing** table to get their respective name for lsting. |

## Audit Process Form

After clicking on request no in audit queue Audit Process Form will pop up.

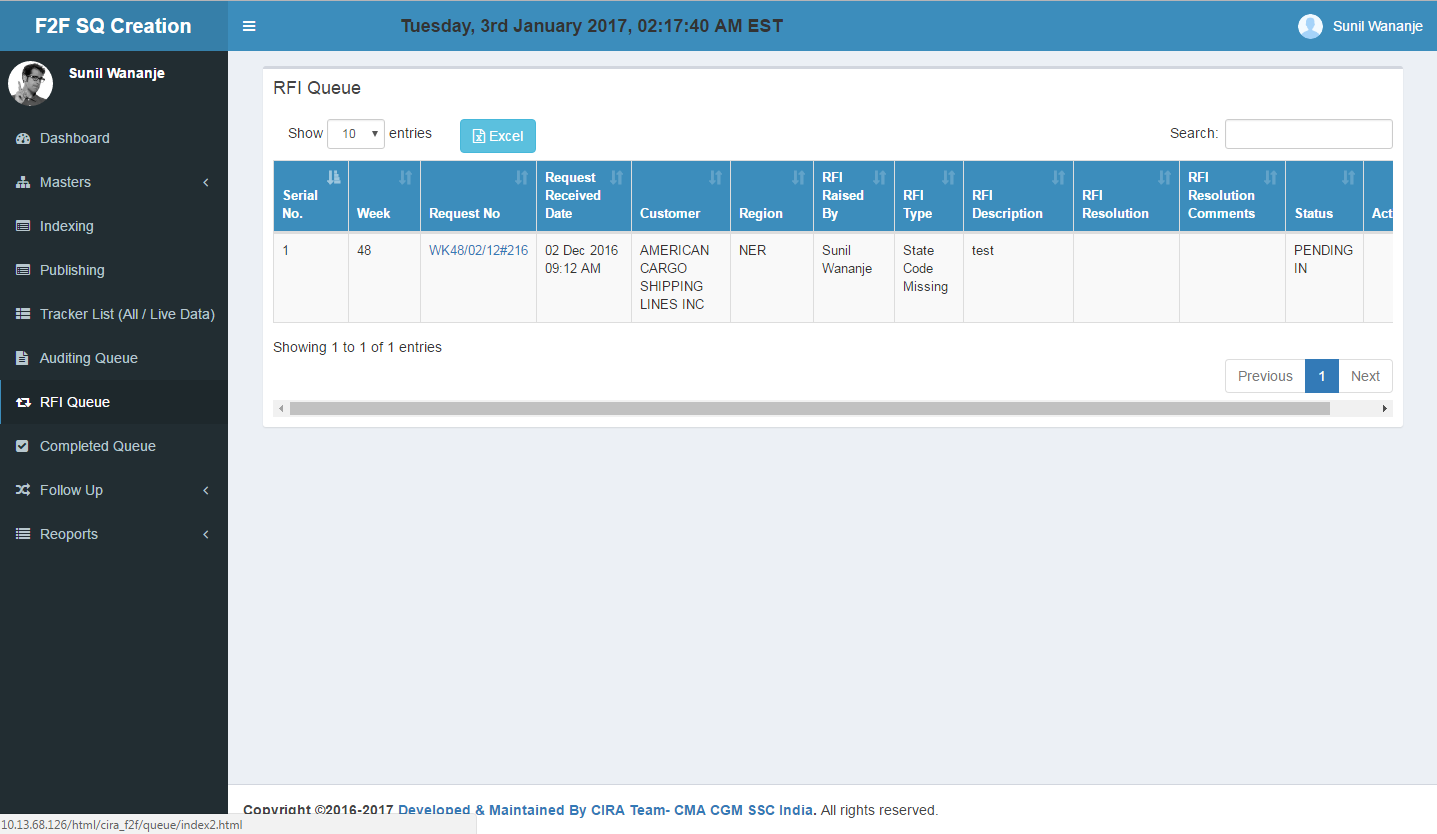
* ISR field is added if status is pending in
* Total lanes, No. Of inlands and Modes field is added in APF form



## RFI Queue

RFI raised from publishing queue and Auditing queue should be comes in RFI management screen

* Once received reply for RFI, Indexer should go to RFI queue => Click on UPF and make it Pending Out,
* Once the Indexer make the request Pending Out there must be a box to update the Reply Receive Time => As soon as the RFI is Pending Out, the request will move out of RFI Queue and will reflect in LIVE queue.
* Export to excel option is added.



|  |  |
| --- | --- |
| **Class/***Controller Function* | **Query Used** |
| **App\Http\Controllers\RfiController** /*index* | select `audit\_queue`.\*, `process\_queue`.`sq\_no`, `process\_queue`.`total\_lane`, `process\_queue`.`no\_of\_inlands`, `process\_queue`.`mode\_id`, `process\_queue`.`pricing\_area`, `partner\_code\_db\_id`, `process\_queue`.`publish\_by`, `indexing`.`id` as `indexing\_id`, `indexing`.`mail\_received\_time`, `indexing`.`indexing\_tat`, `indexing`.`request\_no`, `indexing`.`customer\_name`, `indexing`.`priority\_id`, `indexing`.`region\_id`, `indexing`.`office\_id`, `indexing`.`indexed\_by`, `indexing`.`request\_type\_id`, `mst\_rfi\_type`.`rfi\_type\_name` as `audit\_rfi\_name`, `mst\_error\_cat`.`name` as `audit\_err\_cat\_name`, `mst\_error\_type`.`name` as `audit\_err\_type\_name`, `auditor`.`name` as `auditor\_name`, `corrector`.`name` as `corrector\_name`, `publisher`.`name` as `publisher\_name`, `resolver`.`name` as `rfi\_resolver\_name`, `questioner`.`name` as `rfi\_questioner\_name`, `mst\_status`.`status\_name` as `audit\_status\_name`, `publish\_status`.`status\_name` as `publish\_status\_name`, `process\_queue`.`id` as `process\_queue\_id` from `indexing` inner join `process\_queue` on `indexing`.`id` = `process\_queue`.`indexing\_id` inner join `audit\_queue` on `audit\_queue`.`process\_queue\_id` = `process\_queue`.`id` inner join `users` as `publisher` on `publisher`.`id` = `process\_queue`.`publish\_by` left join `mst\_status` on `mst\_status`.`id` = `audit\_queue`.`audit\_status\_id` left join `mst\_status` as `publish\_status` on `publish\_status`.`id` = `process\_queue`.`status\_id` left join `users` as `auditor` on `auditor`.`id` = `audit\_queue`.`audit\_by` left join `users` as `corrector` on `corrector`.`id` = `audit\_queue`.`audit\_error\_done\_by` left join `users` as `resolver` on `resolver`.`id` = `audit\_queue`.`audit\_rfi\_resolved\_by` left join `users` as `questioner` on `questioner`.`id` = `audit\_queue`.`audit\_rfi\_raised\_by` left join `mst\_rfi\_type` on `mst\_rfi\_type`.`id` = `audit\_queue`.`audit\_rfi\_type\_id` left join `mst\_error\_cat` on `mst\_error\_cat`.`id` = `audit\_queue`.`audit\_error\_cat\_id` left join `mst\_error\_type` on `mst\_error\_type`.`id` = `audit\_queue`.`audit\_error\_type\_id` where `mst\_status`.`status\_name` not in ('pending in', 'sent to pricer', 'sent to customer', 'sent to inside sales', 'done', 'disregard') or `mst\_status`.`status\_name` is null order by `indexing`.`priority\_id` desc, `indexing`.`mail\_received\_time` desc |
| **Table Used** | * **rfi\_queue** table is join with **indexing** , **auditing** and **process\_queue**  to get one or more rfi for single request which comes from auditing as well as publishing.. * **mst\_rfi\_type, mst\_error\_cat, mst\_error\_type, mst\_status, mst\_user, mst\_priority\_type** these table joining with **indexing** , **process\_queue** and **auditing** table to get their respective name for lsting. |

## Completed Queue

Completed Queue consists of all the completed quote requests which moved from publishing and Auditing queue with all details

* Disregard status data also display in this queue.
* Export to excel option is added.



|  |  |
| --- | --- |
| **Class/***Controller Function* | **Query Used** |
| **App\Http\Controllers\RfiController** /*completedQueue* | select `indexing`.\*, `audit\_queue`.`audit\_rfi\_type\_id`, `audit\_queue`.`audit\_rfi\_description`, `audit\_queue`.`audit\_rfi\_raised\_by`, `audit\_queue`.`audit\_rfi\_end\_date`, `audit\_queue`.`audit\_start\_date`, `audit\_queue`.`audit\_end\_date`, `audit\_queue`.`oot\_remark` as `aq\_oot`, `audit\_queue`.`comments` as `aq\_comment`, `process\_queue`.`rfi\_type\_id`, `process\_queue`.`rfi\_description`, `process\_queue`.`rfi\_raised\_by`, `process\_queue`.`rfi\_end\_date`, `process\_queue`.`publish\_start\_date`, `process\_queue`.`publish\_end\_date`, `process\_queue`.`oot\_remark` as `pq\_oot`, `process\_queue`.`comments` as `pq\_comment`, `auditor`.`name` as `audit\_rfi\_user`, `publisher`.`name` as `publish\_rfi\_user`, `aq\_user`.`name` as `audit\_user`, `pq\_user`.`name` as `publish\_user`, `mst\_rfi\_type`.`rfi\_type\_name` as `process\_rfi\_type`, `aq\_rfi`.`rfi\_type\_name` as `audit\_rfi\_type` from `indexing` inner join `process\_queue` on `process\_queue`.`indexing\_id` = `indexing`.`id` left join `audit\_queue` on `audit\_queue`.`process\_queue\_id` = `process\_queue`.`id` left join `users` as `auditor` on `auditor`.`id` = `audit\_queue`.`audit\_rfi\_raised\_by` left join `users` as `publisher` on `publisher`.`id` = `process\_queue`.`rfi\_raised\_by` left join `users` as `aq\_user` on `aq\_user`.`id` = `audit\_queue`.`audit\_by` left join `users` as `pq\_user` on `pq\_user`.`id` = `process\_queue`.`publish\_by` left join `mst\_rfi\_type` on `mst\_rfi\_type`.`id` = `process\_queue`.`rfi\_type\_id` left join `mst\_rfi\_type` as `aq\_rfi` on `aq\_rfi`.`id` = `audit\_queue`.`audit\_rfi\_type\_id` left join `mst\_status` as `aq\_status` on `aq\_status`.`id` = `audit\_queue`.`audit\_status\_id` left join `mst\_status` as `pq\_status` on `pq\_status`.`id` = `process\_queue`.`status\_id` where `aq\_status`.`status\_name` in ('sent to customer', 'sent to pricer', 'done') or `pq\_status`.`status\_name` = 'done' |
| **Table Used** | * **indexing** , **auditing** and **process\_queue**  join these tables with publishin status is between ‘sent to pricer’, ‘disregard’ or status of auditing is between ‘sent to pricer’, ‘disregard’ and ‘sent to customer’. * **mst\_rfi\_type, mst\_error\_cat, mst\_error\_type, mst\_status, mst\_user, mst\_priority\_type** these table joining with **indexing** , **process\_queue** and **auditing** table to get their respective name for lsting. |

## Follow Up Queue

Follow up screen contains the pricer database . It enables the track of pricer chase mails (Reminder1, Remider2) status.

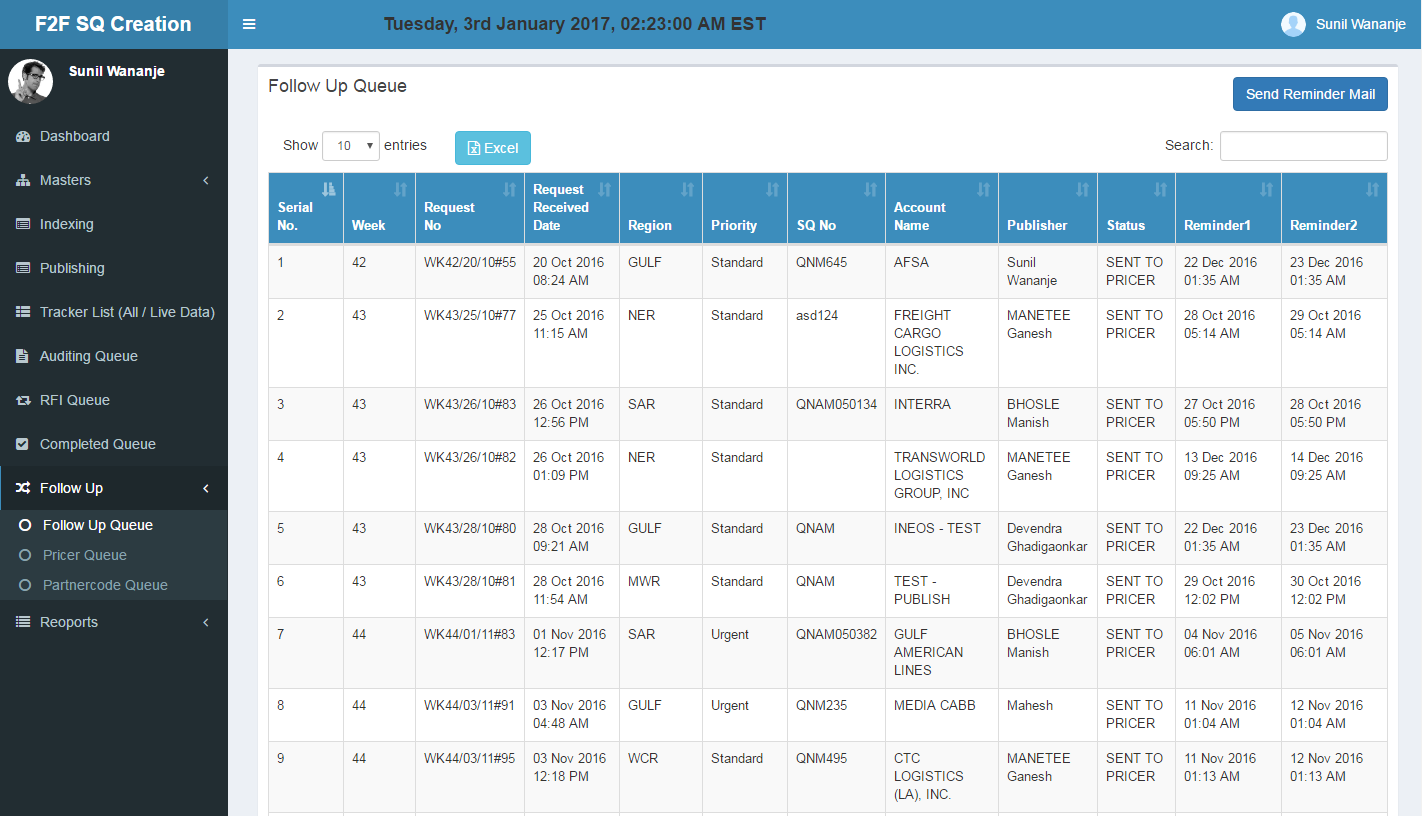
**Triggering Reminder Mail :**

Reminder mail triggered to the Respective User (Publisher) and Generic Inbox. Reminder 1 is next day and Reminder 2 is day after Tomorrow.

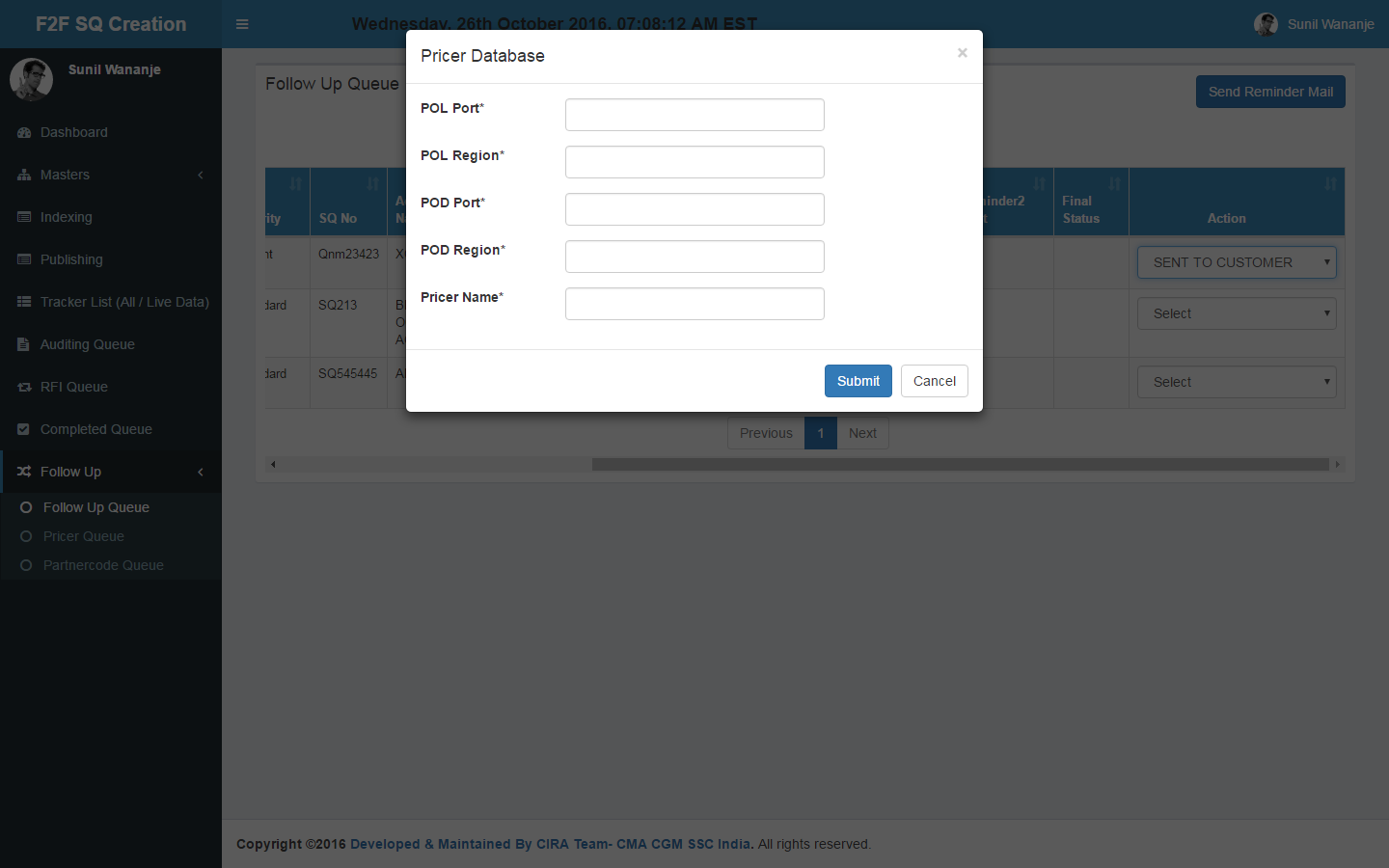
**Pricer Database:**

Pricer Database Form will Pop-up as soon as the Final Status is updated as "SENT TO CUSTOMER".

* Export to excel option is added.



**Pricer Database Form:**

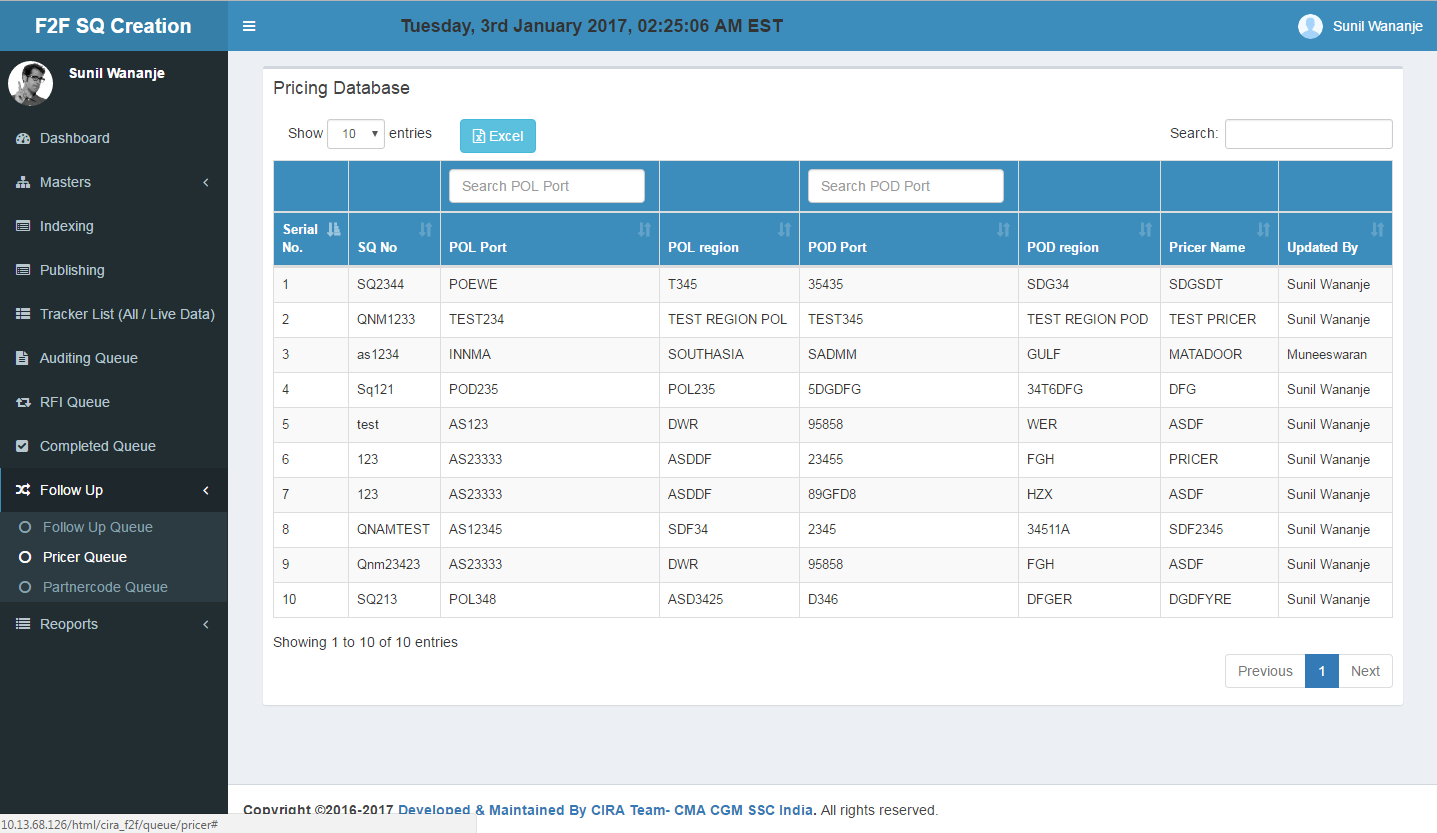


|  |  |
| --- | --- |
| **Class/***Controller Function* | **Query Used** |
| **App\Http\Controllers\FollowUpController**/*index* | select `audit\_queue`.`follow\_up\_date`, `audit\_queue`.`reminder\_1`, `audit\_queue`.`reminder\_2`, `audit\_queue`.`reminder1\_sent`, `audit\_queue`.`reminder2\_sent`, `audit\_queue`.`reminder1\_actual\_sent`, `audit\_queue`.`reminder2\_actual\_sent`, `process\_queue`.`sq\_no`, `process\_queue`.`publish\_by`, `audit\_queue`.`final\_status`, `publisher`.`name` as `publisher\_name`, `audit\_queue`.`id` as `audit\_queue\_id`, `indexing`.`id` as `indexing\_id`, `indexing`.`mail\_received\_time`, `indexing`.`indexing\_tat`, `indexing`.`request\_no`, `indexing`.`customer\_name`, `indexing`.`priority\_id`, `indexing`.`region\_id`, `indexing`.`request\_type\_id`, `mst\_status`.`status\_name` as `audit\_status\_name`, `process\_queue`.`id` as `process\_queue\_id`, `audit\_queue`.`id` as `audit\_status\_id` from `indexing` inner join `process\_queue` on `indexing`.`id` = `process\_queue`.`indexing\_id` inner join `audit\_queue` on `audit\_queue`.`process\_queue\_id` = `process\_queue`.`id` inner join `users` as `publisher` on `publisher`.`id` = `process\_queue`.`publish\_by` left join `mst\_status` on `mst\_status`.`id` = `audit\_queue`.`audit\_status\_id` where `mst\_status`.`status\_name` = 'sent to pricer' |
| **Table Used** | * Fetching all the request for this queue joining **indexing, process\_queue** and **auditing** table and whose status is ‘sent to pricer’ only. * **mst\_status, mst\_user, mst\_priority\_type** these table joining with **indexing** , **process\_queue** and **auditing** table to get their respective name for lsting. |

## Pricer Database

In pricer databse contain all pricer list which moved for sent to customer status

* Export to excel option is added.



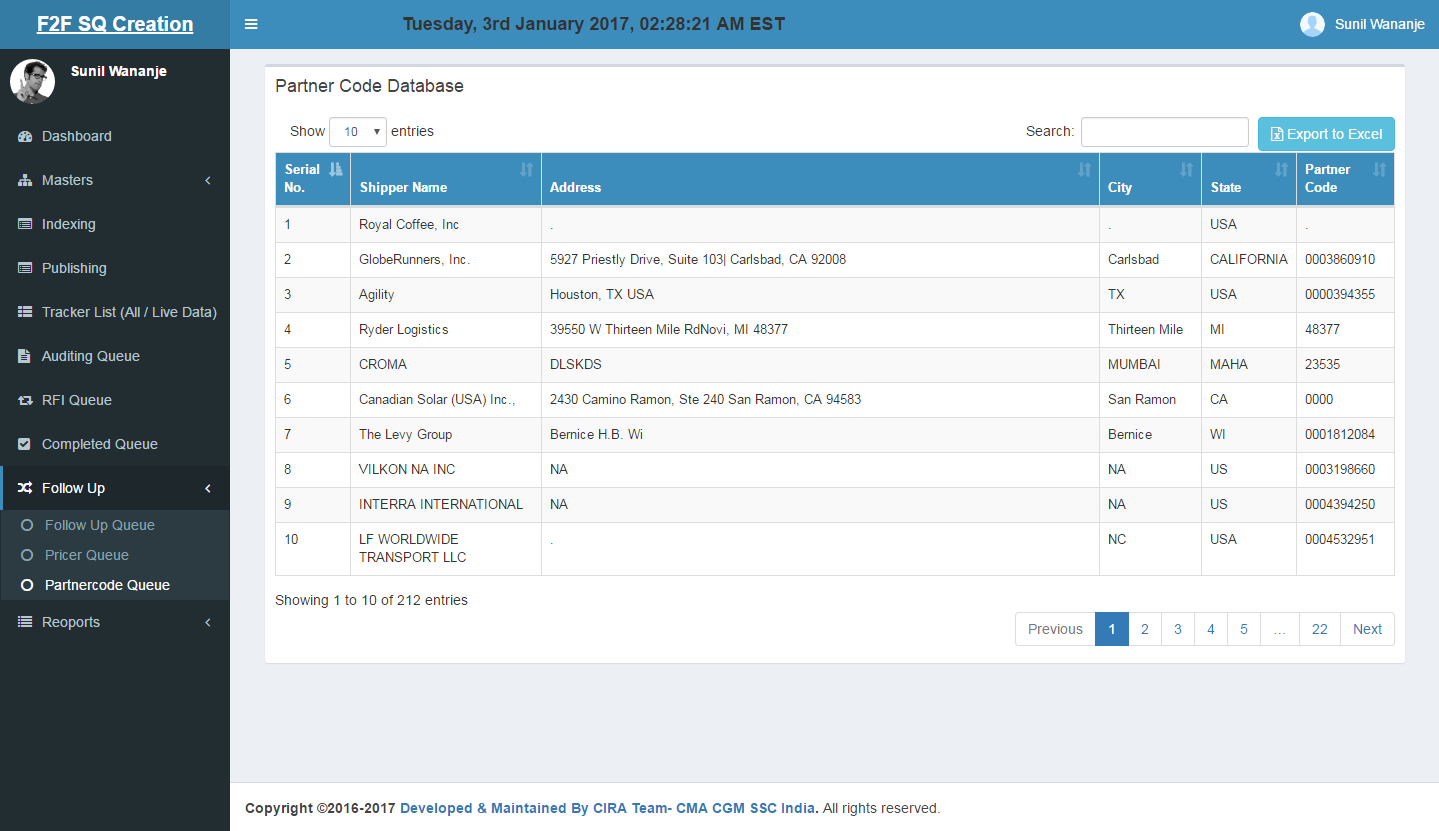
|  |  |
| --- | --- |
| **Class/***Controller Function* | **Query Used** |
| **App\Http\Controllers\FollowUpController**/*pricerData* | select `pricer`.\*, `users`.`name` as `updated\_user`, `process\_queue`.`sq\_no` from `Pricer` inner join `users` on `pricer`.`updated\_by` = `users`.`id` inner join `process\_queue` on `pricer`.`process\_queue\_id` = `process\_queue`.`id` |
| **Table Used** | * Joining **pricer, process\_queue** and **users** and get pricer data |

## Partner Code Database

In UPF we have a field, wherein User will update the Name of the Customer, Address, City, State and Partner Code.This data will be updated by the Users who are handling the requests.

This Data is linked to Partner Code Database .Here we show all partner code list.

* Export to excel option is added.



|  |  |
| --- | --- |
| **Class/***Controller Function* | **Query Used** |
| **App\Http\Controllers\FollowUpController**/*partnerCodeData* | select \* from `partner\_code\_db` |
| **Table Used** | * Fetching data from **partner\_code\_db** table. |

## Reports

### Daily Report

**Tables: indexing, auditing, process\_queue, rfi\_queue** and **mst\_status**

**Request Count:**

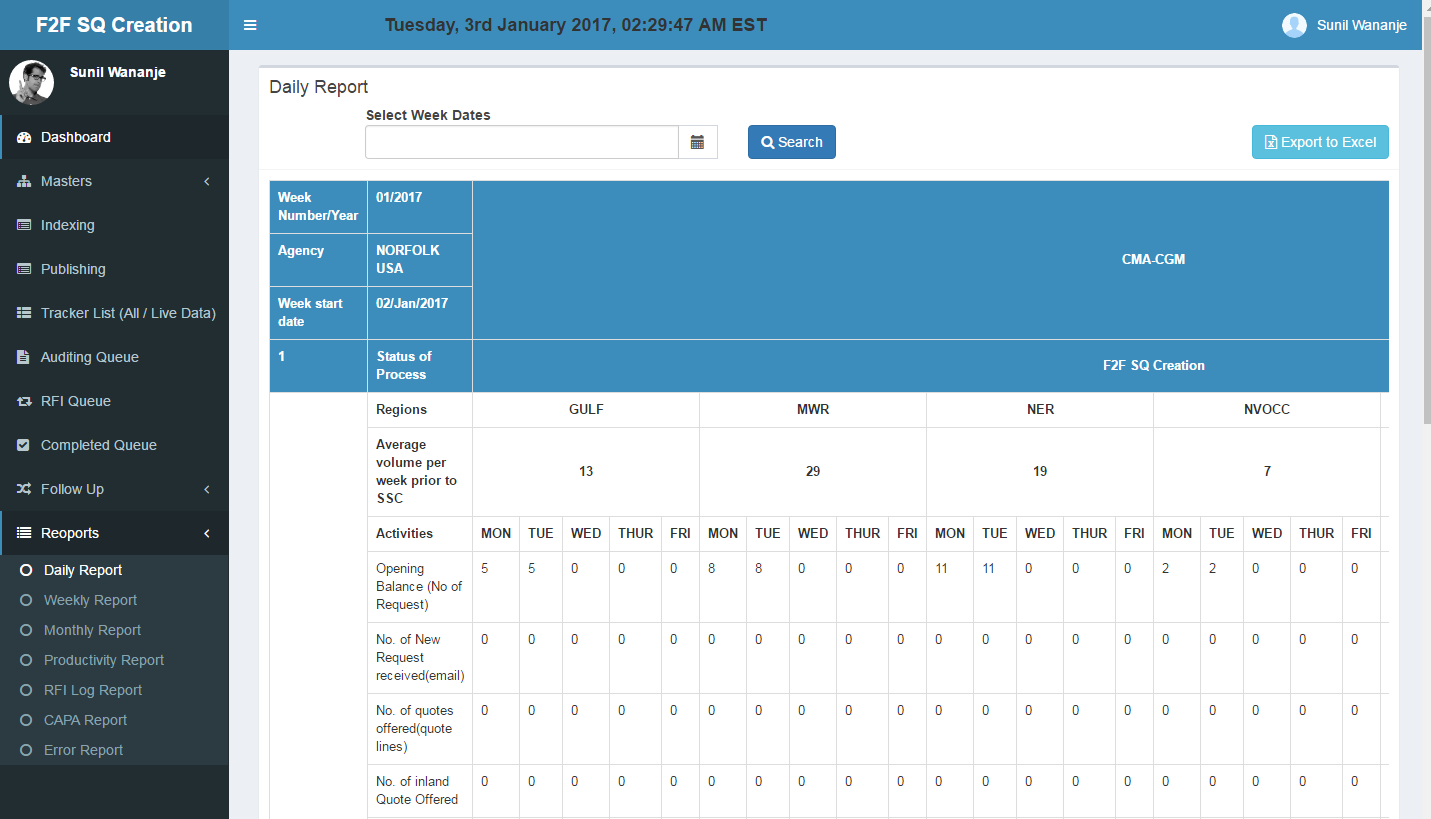
1. **Opening Balance (No of Request):** This is the “Closing balance” of last day i.e. for Monday the Opening balance will be Closing balance of Friday. For Tuesday  the “Opening Balance” will be Closing balance of Monday and so on.
2. **No. of New Request received(email):** Anything which is indexed on a particular day will reflect here i.e. if 100 mails are indexed on 8th November, 2016 same will reflect in this field.
3. **No. of quotes offered(quote lines):** Suppose if 100 request are quoted (status **DONE and DISREGARD** fromPublish queue and status **SENT TO PRICER**, **DONE and DISREGARD** from Audit queue),  on 8th November, 2016 which consist of **200 lanes** then **200** will reflect here.
4. **No. of inland Quote Offered:** Suppose if 100 request are quoted (status **DONE and DISREGARD** fromPublish queue and status **SENT TO PRICER**, **DONE and DISREGARD** from Audit queue), on 8th November, 2016 which consist of **110 inlands** then **110** will reflect here.
5. **No. of Request sent to Customer:** Suppose if 100 request are quoted (status **DONE and DISREGARD** fromPublish queue and status **SENT TO PRICER**, **DONE and DISREGARD** from Audit queue), on 8th November, 2016 then **100** will reflect here.
6. **Closing Balance (No of Request):** It is **Opening Balance (No of Request) + No. of New Request received(email) - Closing Balance (No of Request)**

**RFI Count:**

1. **Opening Balance (No of RFI):** This is the “Closing balance of RFI” of last day i.e. for Monday the Opening balance of RFI will be Closing balance of RFI from Friday. For Tuesday  the “Opening Balance of RFI” will be Closing balance of RFI from Monday and so on.
2. **No. of New RFI raised:** It is the number of time “Pending In” is done. Currently it is not calculating in the Report.
3. **No. of RFI resolved**: It is the number of time “Pending OUT” is done.
4. **No. of RFI pending resolution**: It is **Opening Balance (No of RFI)** **+** **No. of New RFI raised** **-** **No. of RFI resolved**

**Filters:**

* Date filter is added for getting previous week data.



### Weekly Report

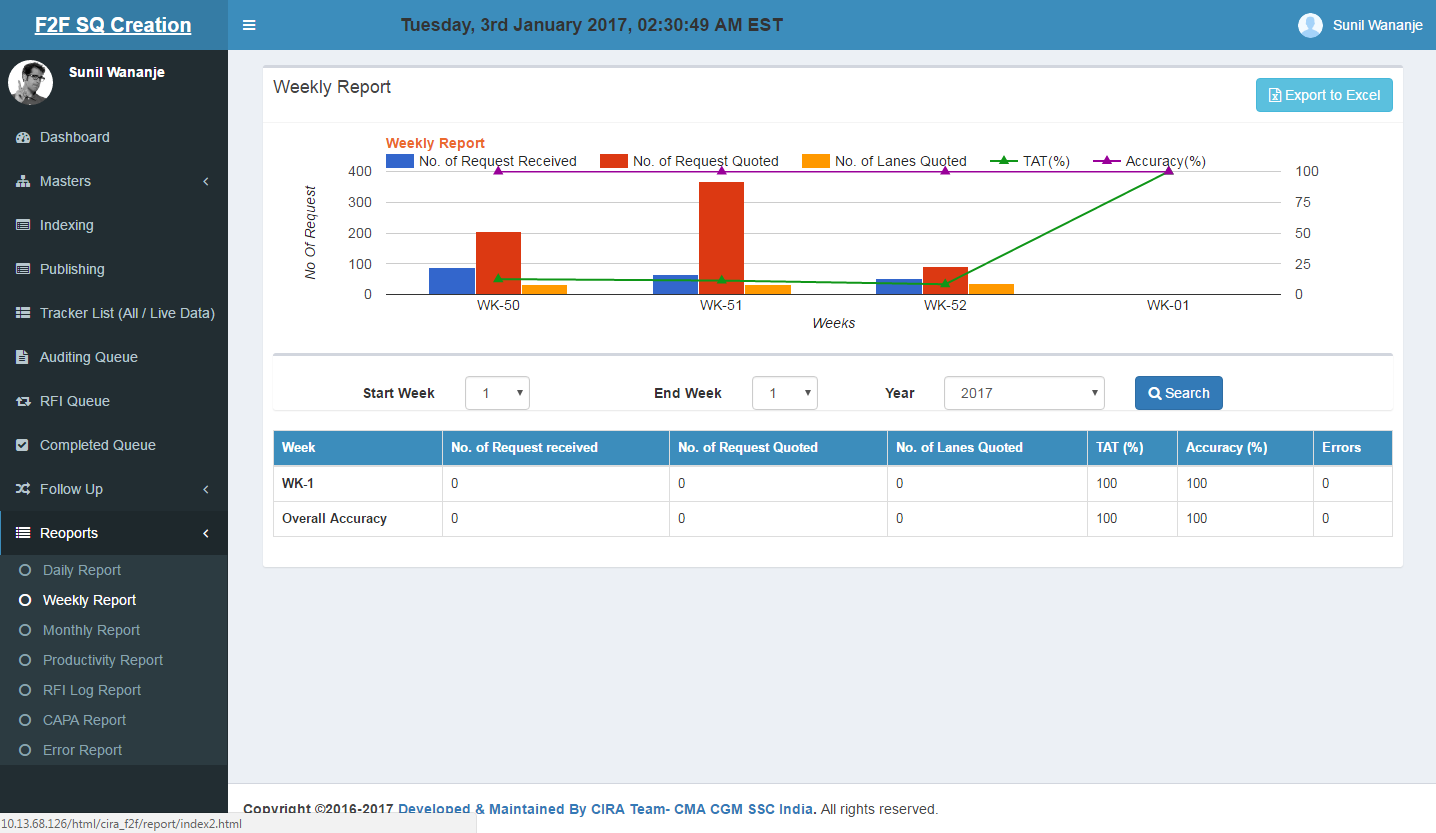
**Tables: indexing, auditing, process\_queue,** and **mst\_status.**

**Request Count:**

1. **No. Of Request Received:** Count the number of request received between start week and end week.
2. **No. Of Request Quoted :** Count the number of request quoted means whose publishing or auditing status is ‘sent to pricer’ or ‘sent to customer’.
3. **No. Of Lanes Quoted :** Sum the number of lanes of quoted request means whose publishing or auditing status is ‘sent to pricer’ or ‘sent to customer’.
4. **TAT(%) :** Claculate TAT % of those request whose out of tat is accepted by admin means ‘oot = 1’.
5. **Accuracy(%):** Calculate error % of all request of selected weeks.

**Filters:**

* Week from , week to and year filter is added for getting previous week data.



### Monthly Report

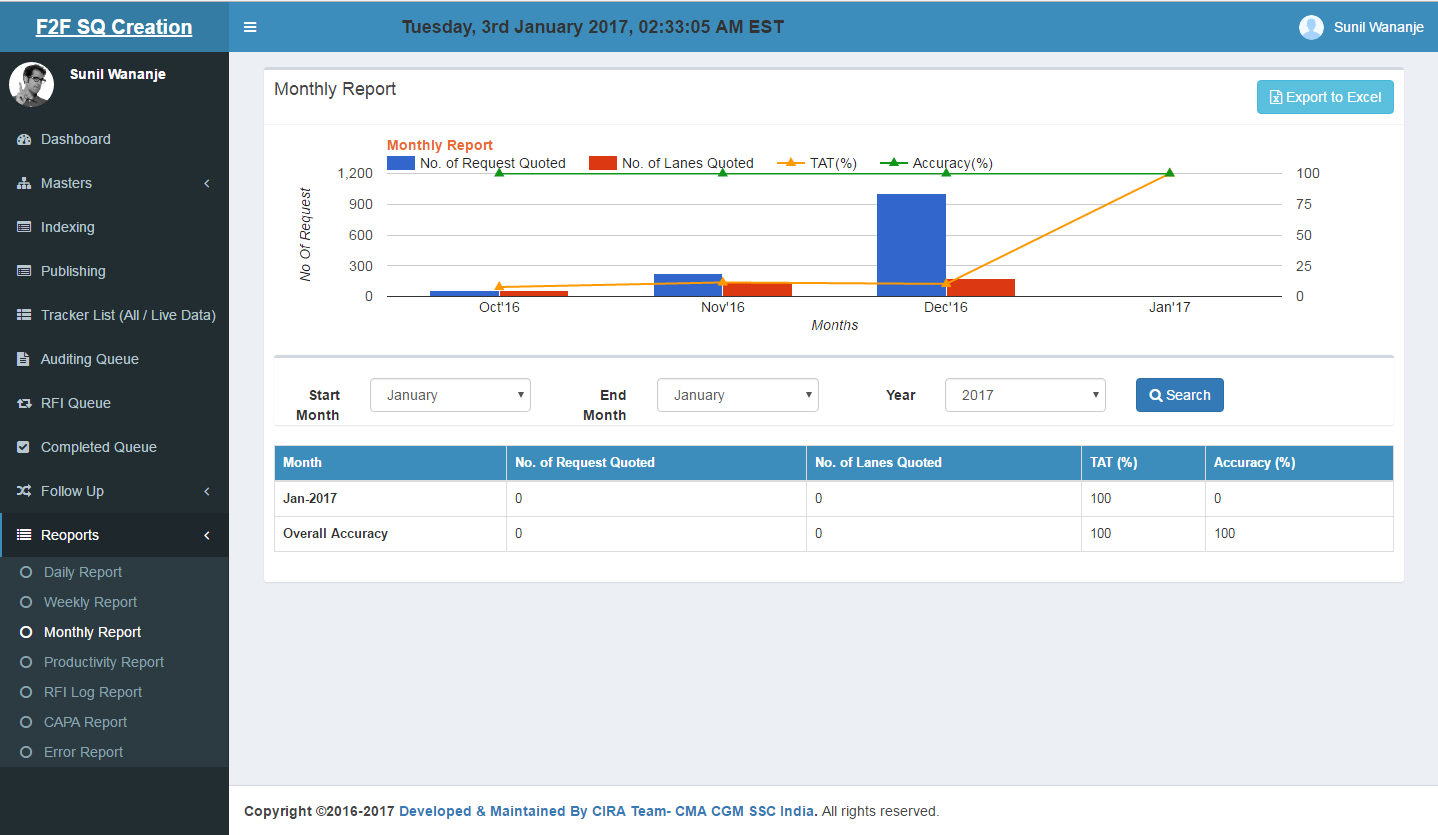
**Tables: indexing, auditing, process\_queue** and **mst\_status.**

**Request Count:**

1. **No. Of Request Quoted :** Count the number of request quoted means whose publishing or auditing status is ‘sent to pricer’ or ‘sent to customer’.
2. **No. Of Lanes Quoted :** Sum the number of lanes of quoted request means whose publishing or auditing status is ‘sent to pricer’ or ‘sent to customer’.
3. **TAT(%) :** Claculate TAT % of those request whose out of tat is accepted by admin means ‘oot = 1’.
4. **Accuracy(%):** Calculate error % of all request of selected months.

**Filters:**

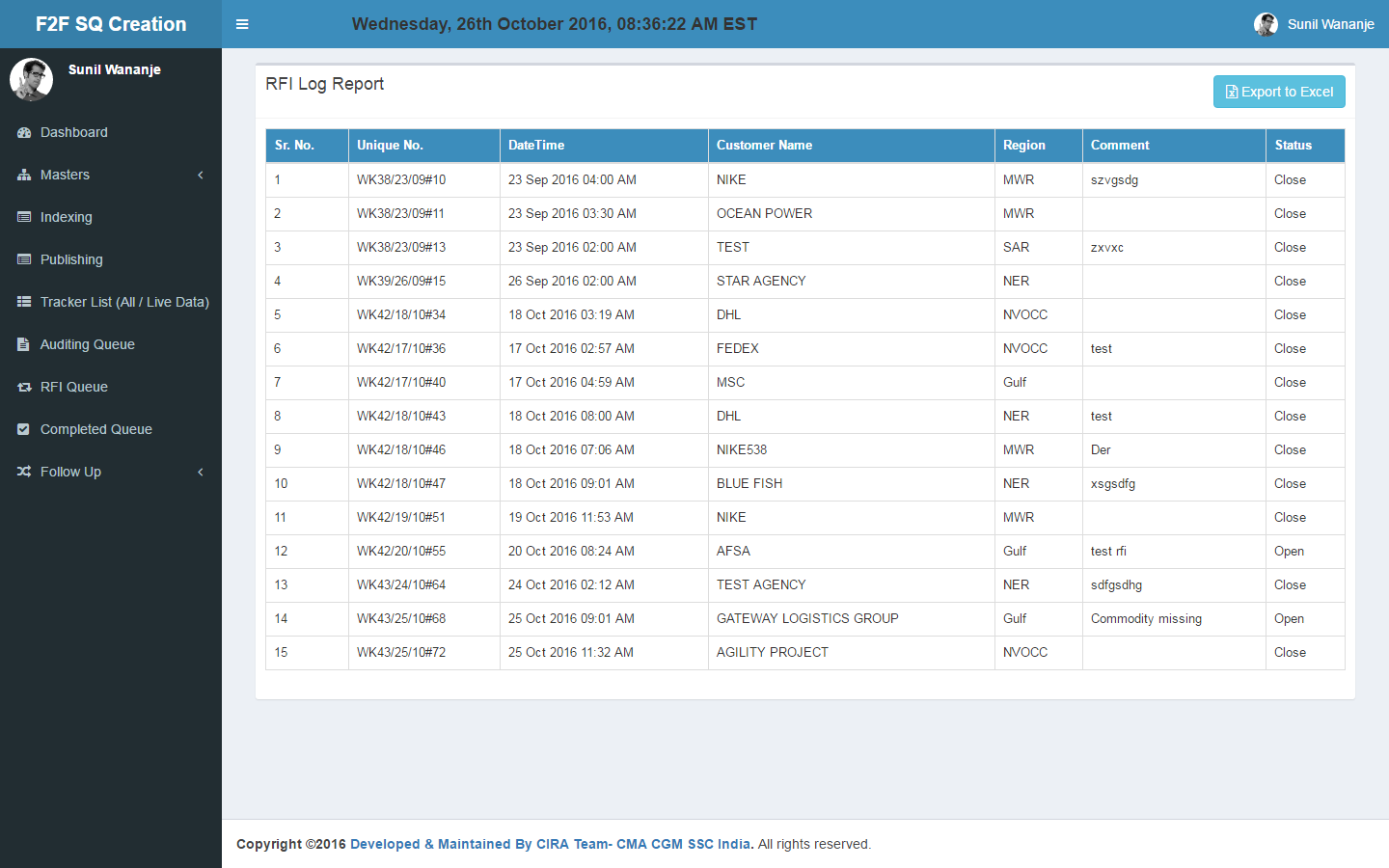
* Month from , Month to and year filter is added for getting previous month data.



### RFI Log Report

**Tables: indexing, auditing, process\_queue, rfi\_queue** and **mst\_status.**

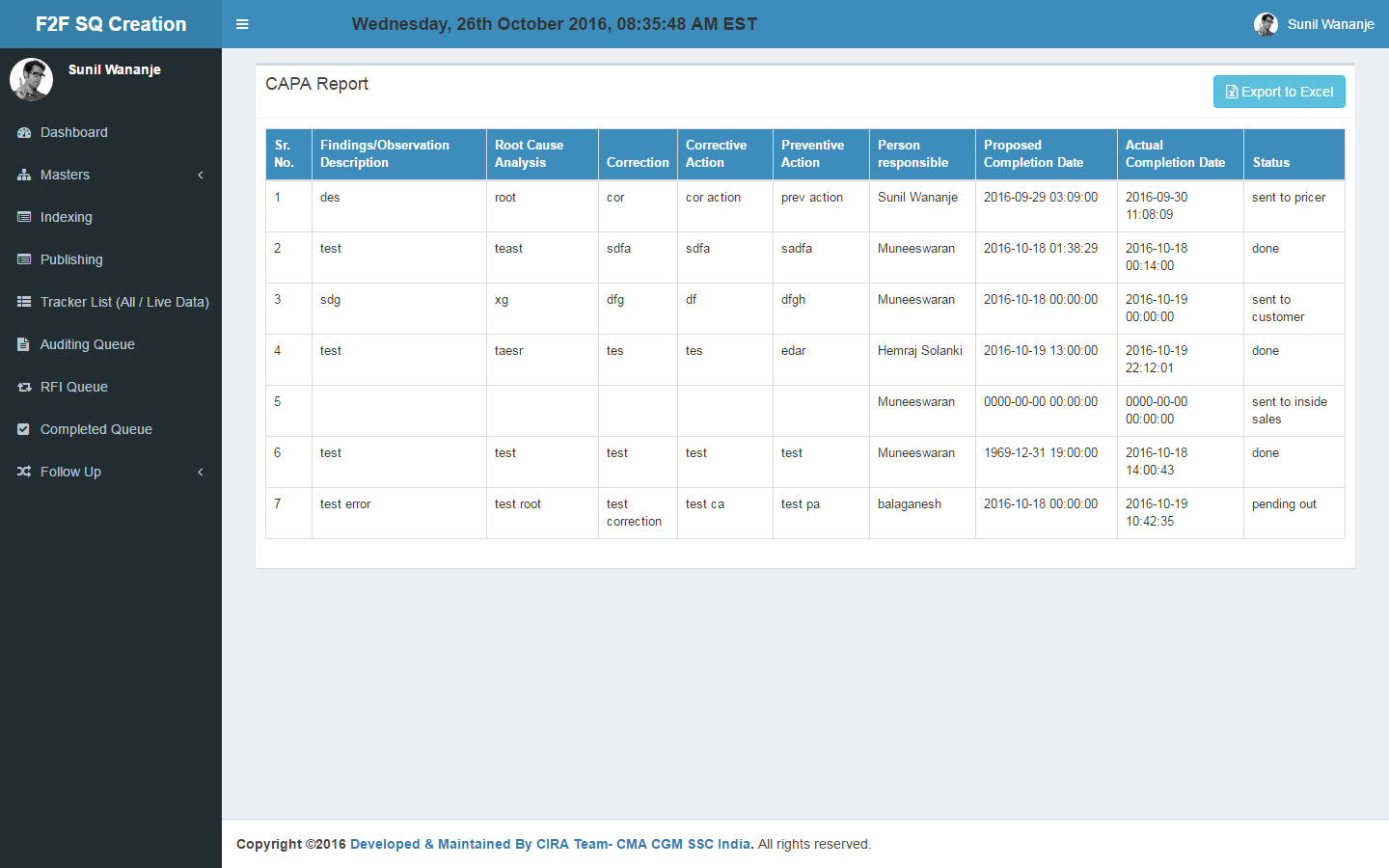
* Fetching all rfi which are closed as well as open.



### CAPA Report

**Tables: indexing, auditing, process\_queue** and **mst\_status.**

* Fetching all request whose corrective and preventive action is added.

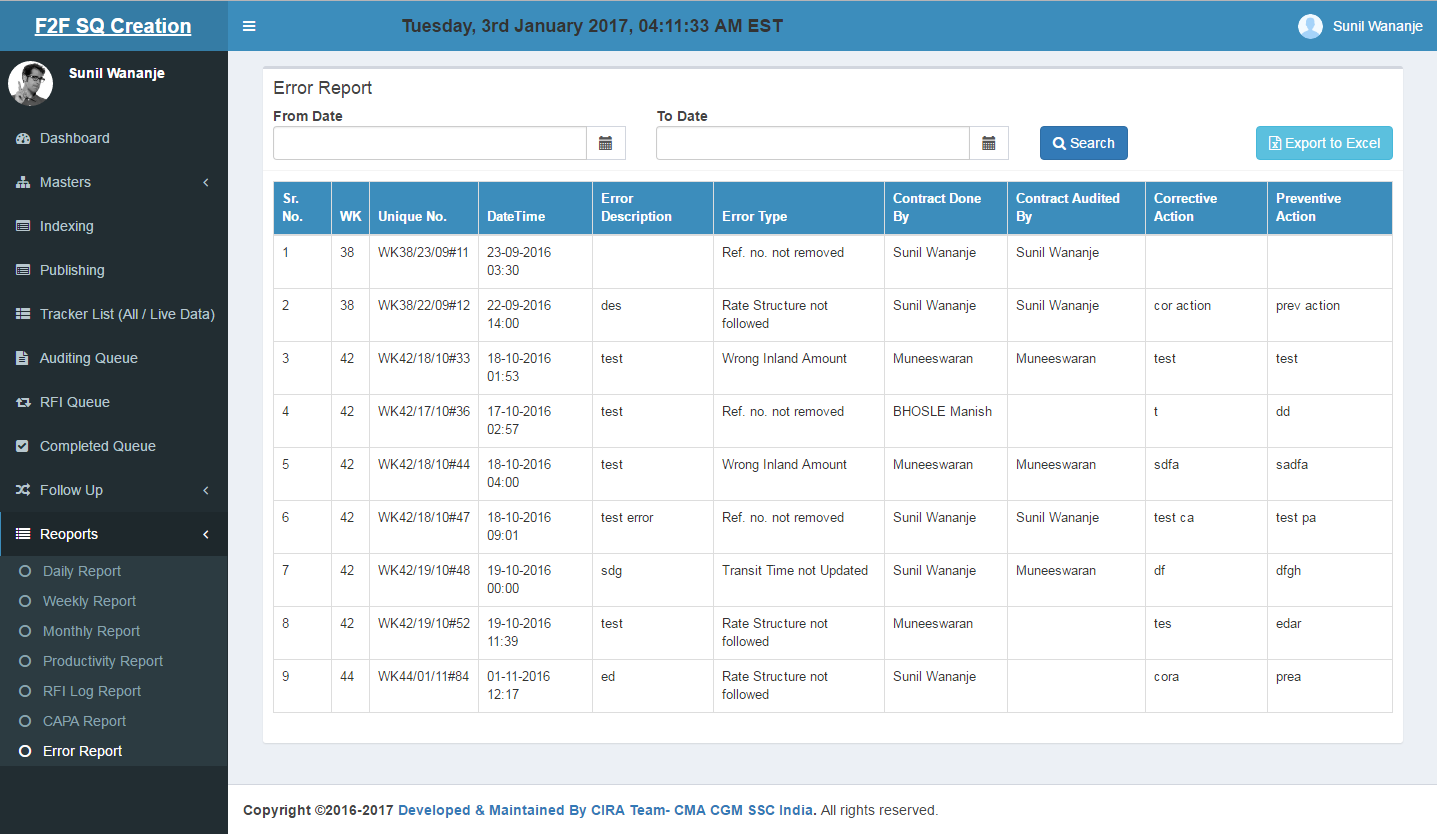


### Error Report

**Tables: indexing, auditing, process\_queue** and **mst\_status.**

**Filters:**

* Date from and date to filter is added for getting previous data.

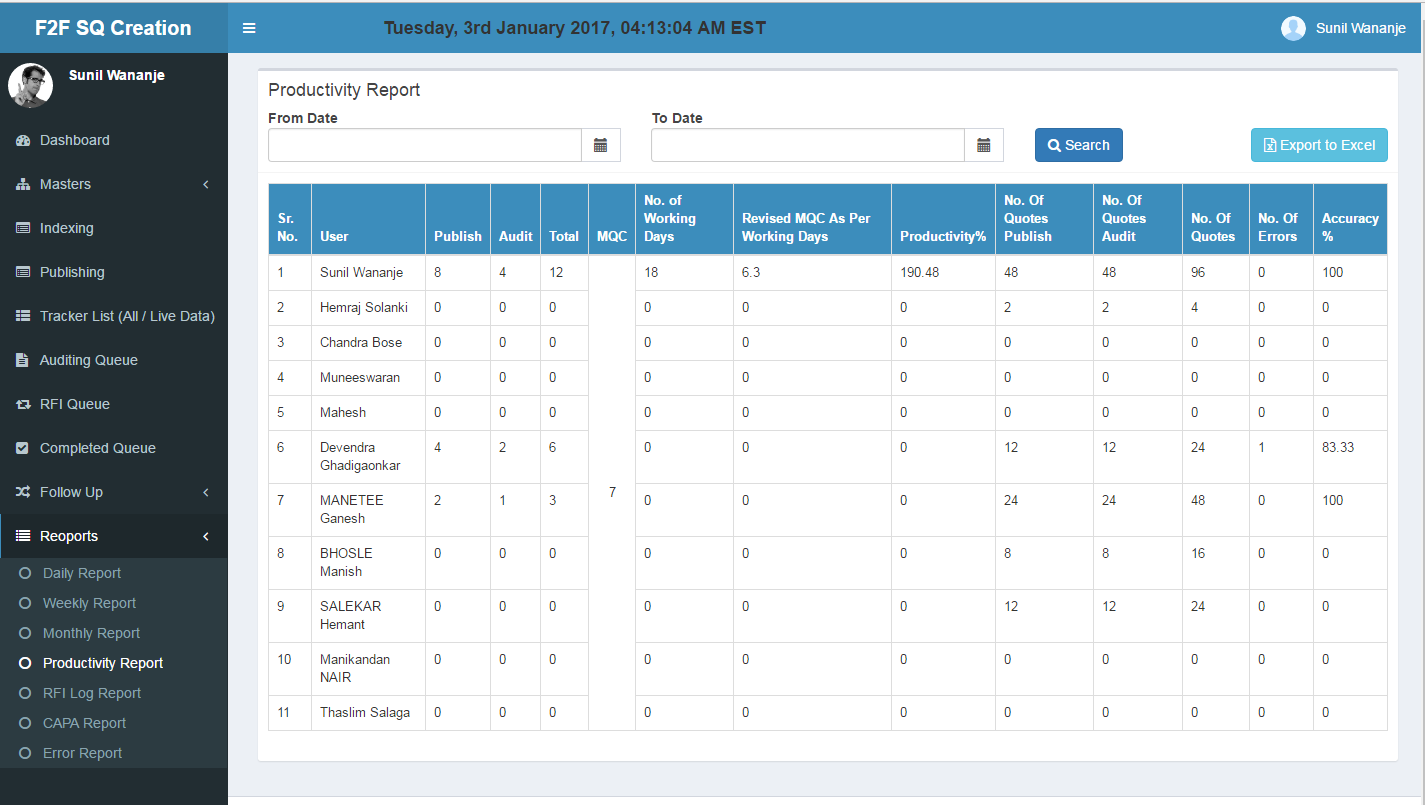


### Productivity Report

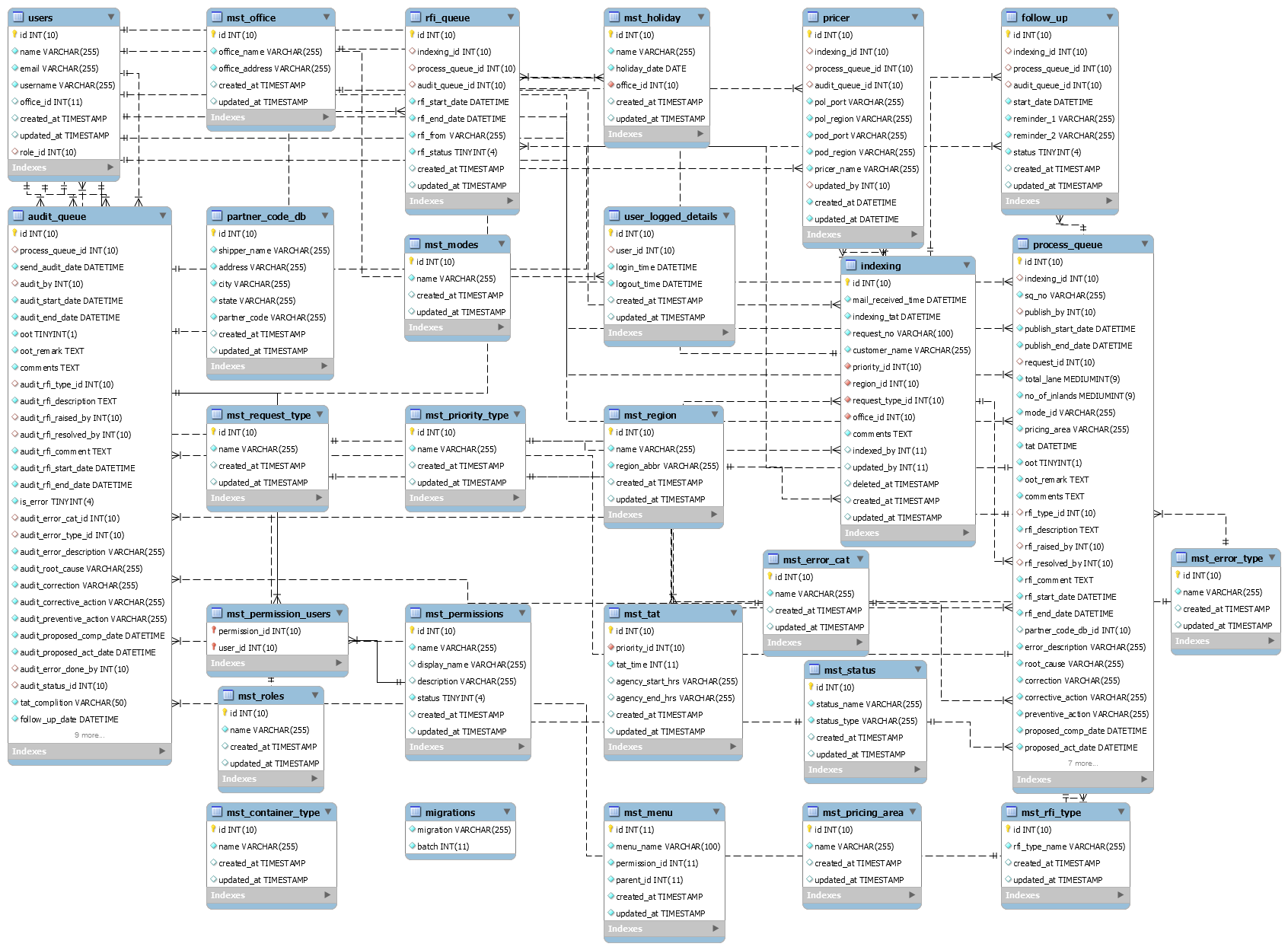
**Tables: indexing, auditing, process\_queue** and **mst\_status.**

**Filters:**

* Publish point and audit point is added.
* By default data is between 1st date of current month to current date excluding weekends and holidays



## Entity Relationship Diagram



### Stored Procedure: <stored procedure name>

*N/A*

# Interfaces

*N/A*

# Error Handling

*The debug option in your config/app.php configuration file determines how much information about an error is actually displayed to the user. By default, this option is set to respect the value of the APP\_DEBUG environment variable, which is stored in your .env file.*

*For local development, we should set the* ***APP\_DEBUG*** *environment variable to* ***true****. In production environment, this value should always be* ***false****. If the value is set to true in production, you risk exposing sensitive configuration values to your application's end users. If this variable is false then any errors occur in application it automatically forward to error page which shows* ***Something went wrong*** *message.*

*This application supports writing log information to single files, daily files, the* ***syslog****, and the* ***errorlog****. Currently we are following single file log system and log path is* ***storage/logs/laravel.log.*** *You can modify the log option in config/app.php configuration file. For example, if you wish to use daily log files instead of a single file, you should set the log value in your app configuration file to daily:*

*'log' => 'daily'.*

***In programming level, all errors handled by try and catch block.***

***For error we have created the following customized error pages that will be displayed to the user:***

1. ***For Session expiration we have created one page which shows “Session Expired!” message (i.e.*** ***resources/errors/TokenException.blade.php)***
2. ***If user have no permission then we show them “You don't have permission to access.... Please contact to admin.” On dashboard page i.e resources/view/dashboard.blade.php.***
3. ***For All other fatal errors we shows “Sorry Something went wrong”***

# Installation Requirements

*Need to Configure .env file which is present in folder Database and change the URL Config Lines Once the UAT Code is Move to Production. Update below mention variables in .env file as per our URL, database name, databse user name and database user password.*

