|  |  |
| --- | --- |
| **Functional Requirements Document(FRD)** | |
| **HIMS Upgrade(8.0)** | |
| **Project Code: 01049023** | |
| Prepared by | Prepared for |
| **Satra Services and Solutions**  **Private Limited** | **<client Name>**  **<Client Logo>** |
| **December 2023** | |

|  |
| --- |
| HIMS Upgrade(8.0) |

|  |  |  |  |
| --- | --- | --- | --- |
| **Quality Assurance Statement** | | | |
| **Prepared for** | | | **Prepared by**  Durga Konduru, Bhargav Turaga |
| **Report Name**  Functional Requirements Document(FRD) | | | **Reviewed by**  Durga Konduru, Rajshekar Gotimukul |
| **Project/Contract Number**  For Client:  For SATRA: 01049023 | | | **Approved for issue by**  Raj Mallela |
| **Date of Issue**  December 2023 | | | **Project Manager**  Durga Konduru |
| **Revisions** | | |  |
| 1 |  |  | |

**Table of Contents**

1 Summary 2

1.1 Background 2

1.2 Outline of the Inception Report 2

1.3 Objectives of the project 2

1.4 Scope of Services 2

1.5 Training and Knowledge Transfer 2

1.6 Methodology 2

1.7 Work Plan 2

2 Mobilisation 3

2.1 Project Office 3

2.2 Project Staff 3

2.3 Visa Processing(Not Applicable) 3

2.4 Off the Shelf System 3

2.5 Services/Facilities Provided by the Client 3

3 Overview of System Architecture 4

3.1 Technical Details 4

3.2 Approach and Methodology 4

4 Discussions with Stakeholders 5

4.1 Initial Discussions 5

4.2 Summary of Discussions 5

5 Software Customization and Implementation 6

5.1 Hardware & Software Requirements 6

5.2 System Design and Customisation Requirements 6

5.3 System Configuration 6

5.4 User Acceptance Testing 6

5.5 Deployment 6

5.6 Training and Technology Transfer 6

6 Focus Items 7

6.1 System Hosting 7

6.2 Timelines for Approvals/Feedback 7

6.3 Availability of Data/Information 7

6.4 Presentations/Discussions 7

6.5 Support Services 7

6.6 Out of Scope 7

7 Project Deliverables 8

7.1 Reports and Manuals 8

7.2 Key Deliverables 8

8 Work Plan and Staff Schedule 9

8.1 Work Plan 9

8.2 Team Composition 9

### Annex-I: Documents & Checklist

List of Tables

No table of figures entries found.

List of Figures

No table of figures entries found.

Glossary

# Summary

Background

<< Back ground of the project/ client details>>

SATRA Services and Solutions Pvt Ltd, India was awarded the consultancy services with Contract No..

**Key Contract Data:**

Contract sum : XXXXXXXXXXXXXXXXX

Contract signing : 21 September 2017

Commencement Date : 23 October 2017

Total Duration : 6 months + 6 months

Intended completion date : 22 October 2018

Outline of the Inception Report

<<Brief Description>>

This inception report covers in detail the following:

* Item 1
* Item 2

Objectives of the project

Scope of Services

Training and Knowledge Transfer

Methodology

Work Plan

# Functional Requirements

About HIMS 7.5

Current version of HIMS is built on the concept of Zero Coding which means no coding is needed and no build required for any implementation. From setting up geographical areas to implementing any kind of business logic, simple configuration is needed. Required screens are provided in current version to do the configuration. However, because of nature of the application some challenges were found while meeting very specific requirements of the user. Performance issues were found especially while fetching data from database. Though the application was designed on then trending technology, after 5 years (2017 to 2023), many of the features have become obsolete with the browser upgrades and increasing data needs.

Challenges

The objective of this upgrade is to address below challenges along with enhancements to make the application an enterprise product.

1. Resolve performance issues while bringing data into grids. Average time for grid to populate in current system is 20 sec though there is a limitation of 10000 records at application level. Grid with only 5 to 10 records is also taking same amount of time. It is expected to bring it to 5 sec maximum
2. Average navigation time from one page to other (load time) is 10 sec. In particular, dashboard pages take lot of time. It should be brought done to 3 secs or less as per industry standards.
3. Alters and notification mechanism are currently managed via emails. However, SMS, whatsapp messages and application (mobile/ web) notifications are not implemented. Should be implemented in new version.
4. There is no workflow in current system. Configurable workflow should be designed.
5. More enhancement to be given to data and map interaction.
6. Dynamic forms are at basic level. Should be improved to meet all data entry needs.

# System Configuration

Some of the tables, features and functionality in HIMS are coming up automatically when the application is deployed. It would be good if we can separate these tables and create only when certain operations are initiated. For example, Road Master and Link Master are to be created only when we say referenced DB.

## Reference Type

System Administrator should have the option to select whether an application is Unreferenced/ Fully Referenced / Semi(Partially) referenced. Default is Unreferenced.

### Unreferenced

Following features should be added only when it is selected as Referenced.

1. Full fledged LRMS module comes with Referenced application.
2. Create Road Master, Link Master, Link Summary and Road Summary Tables.
3. Create SysGeolevel1 to 10 tables.

It may not be necessary to delete them when it is switched back to Unreferenced, which can be debated further at later stage.

### Semi (Partially Referenced)

1. LRMS module is not provided. However RoadMaster table is created and accessed by Module Administrator under tools.
2. LinkMaster is not required.
3. Chainage validations should happen with respect to RoadMaster.

### Fully referenced

1. Current HIMS 7.5 version can be considered as Fully referenced application.
2. LRMS module and RIS module are given by default. Only RIS administrator should be able to access LRMS module. No other users should have access to LRMS module. We can think about having LRMS as one of the menus as well instead of a separate module ( depends on some of the RFPs, to be discussed further).

## GIS enabled

Following should be added/set only when the application is GIS enabled:

1. Create SysmapLayers, GPS Master tables.
2. Create Standard Process for preparing GPS Shape for the selected tables.
3. Enable an option in table and View definition as GPS enabled. That should create Shape, Latitude and Longitude columns automatically. When GPS is disabled, these columns can be removed.
4. GIS Menu navigation should happen only when GPS is enabled at system level.
5. Data Access in User Security should be enabled only when GIS is enabled for application.

## Survey year

This is currently used only for referenced data. To use for unreferenced data as well, it is proposed to rename to Datayear. If this requires huge efforts, atleast we should allow the user to change the caption. Say for example, if any invoices are to be verified, this column will be given caption as “Transaction year”. Transaction years can be something similar to “2023-24’ as well. Hence this column should be a text column with a Sequence column to identify the sequence of transactions. Parameter can be kept as “DatayearParam” throughout the application for easiness.

System should allow to define seasons or Quarters as well as sub items of years. In such case, That column can be named as “YearPortion” and caption to be changed as appropriate.

If a table or view has both “DataYear” and “YearPortion” columns, then two drop downs are to be displayed. If there is only Datayear column then, “YearPortion” should be invisible.

***Note:*** *DataYear and YearPortion are the key words and should not be used in anywhere.*

## Global Variables

These variables are used throughout the application in same format.

1. User ID: **UserParam** is the text to be used to extract user id for SPs or in any code.
2. Survey ID or current year or Data Year: This is used to differentiate year wise transactions or surveys. If any table has a column as “DataYear” then that can be used to filter data yearwise. Text to be used for that filter is DataYearParam.
3. YearPortionParam: for quarter or Seasons data.

***Note:*** *Survey ID column is used for multiple purposes. Please refer SurveyID section.*

1. Admin levels: Geo1Param – for 1st Geo Level, Geo2Param – For 2nd Geolevel. Similarly for other Geo levels.

# Object Configuration

Tables

Tables / datasets as called in HIMS, are going through following changes in HIMS 8.0 while retaining some of the existing features.

### Table Definition

Following are the attributes of a table introduced or modified in 8.0.

#### Read-only

Currently this feature is not working properly. Even if it is set as Read-only, it is appeared in Import templates and import dropdown. Edit/ + buttons are also shown in Grid. This needs to be reviewed and bugs are to be fixed.

#### Rows in a Grid

Current Setup for number of Rows in a Grid at System level is meeting the requirement and setting up for all tables. However, if there is any specific need for a particular table to increase the count, this new attribute can be used.

#### Document Upload

Current Document Upload attribute needs to be enhanced to take below sub attributes

1. Document type: In many of the cases, these documents types are custom values. However, it would be good, if a provision is given for a full SQL as well as custom values. No requirement of Row. Property here as document type is setup at table level.

Eg: Document Types: Experience letters, Contract Documents etc

Need to explore if the same can be given for Images as well.

2. Uploaded on: This needs to be populated when the document is shown on screen.

#### Display SQL

This attribute details the SQL text to be executed when the dataset is selected to display in grid. In 7.5, all columns that are defined in table are displayed in grid. Challenges experienced when user wants to see some informative columns that are not available in that table. To address this issue, display SQL attribute is defined. Now grid display can be controlled through various joins and conditions. UserID, SurveyID and Geographical parameters are available to use in SQL. Since the output is multiple rows and columns, a full SQL is to be given. User can override the visible feature as needed manually. All columns from the dataset will be displayed in Grid with captions as grid headers. If there is any column (calculated) with no name, a default caption of FLD1 will be used for grid display.

SQL Ex:

SELECT R.RoadId 'Road ID'

,R.RoadCode 'Road ID'

,R.RoadCategoryCode 'Category'

,R.Name 'Road Name'

,R.StartChainage 'Start Chainage'

,R.EndChainage 'End Chainage'

,R.Length 'Road Length'

,R.StartPoint 'Start Point'

,R.EndPoint 'End Chainage'

,R.OldRoadCode 'Old Road Code'

,R.Route 'Route'

,R.StartPointLatitude 'Start Lat'

,R.StartPointLongitude 'Start Long'

,R.EndPointLatitude 'End Lat'

,R.EndPointLongitude 'End Long'

,R.DivisionCode 'Div Code'

,R.IsCRN 'CRN'

FROM NM.RoadMaster R

Developer’s notes: First letter of the dataset name should be taken as alias name and all columns should be referred by that alias. Column Alias names are captions of the columns in definition. Invisible columns should not be appeared in this SQL. However, we don’t control if the user changes the SQL for any special reason.

Ex 2: To get only invoices pertaining to a logged in employee

SELECT I.PrjCode 'Project Code'

,I.Mlstn 'Mile Stone'

,I.InvNo 'Invoice No'

,I.InvDt ' Invoice Dt'

,I.InvAmt ' Invoice Amount'

,I.GST 'GST'

,I.AmtGST 'GST Amount'

,I.PmtRcv ' Payment Received'

,I.Rmrks 'Remarks'

,I.LstPymntDt 'Last payment Dt'

,I.Crrncy 'Currency'

,I.PndAmt 'Pending Amount'

,I.GSTRcv 'GST received'

,I.GSTPnd 'GST pending'

,I.InvSts 'Invoice Status'

,I.Particulars 'Particulars'

,I.ProjName 'Project Name'

,I.ConvRate 'Conversion rate'

FROM Invoice I

Inner join Project P on P.PrjCode=I.PrjCode

Inner join EmpMaster E on e.EmpCode=UserParam and P.PrjMgr=E.EmpCode

*Developer’s note: UserParam is used to know the logged in user. Row key word is not valid in this as there wont be any current row.*

#### GIS Enabled

To enable GIS Component to the table. This will add Shape, Lat and Long columns to a table. If the table is Section type, then both Start and End Lat and Longs are to be added and if the table is point type, only one of Lat and Long are to be added.

GIS enabled option should be enabled only if the system config is set as GIS Enabled= TRUE.

### Column definition

Following columns will be maintained as is.

#### Data Types

A new datatype QR Code will be added in column definition. This type is used to update QR codes for the given row information. Following additional attributes are to be filled when QR Code data type is selected.

1. QR Formula/ Generation Text: This attribute takes a partial SQL that is used to generate QR code for a selected column.

Eg: *select concat(row.ProjectCode;,' ', row.projectname;)*

Returns a text value. *From Table name* is not allowed.

Row represents current row to which edit operation is to be performed.

1. **; -** row.columnname should end with semicolon so as to read the text in .net. This is mandatory for a successful execution of SQL.
2. Generally this is not needed to join this dataset with another dataset. If needed, it is advised to join that table in Table SQL( please refer below sections for more details) such that one can access those columns.

When a data is edited or added, QR Code is generated with the above SQL and QR image is stored in application folder. The same path will be stored in respective row and QR column.1 table can have any number of QR code columns.

QR code can be viewed from Image Gallery. A separate tab will be created with the name of the QR crode column in Image Gallery.

#### Visible

There are no changes to this attribute from 7.5 to 8.0 at column definition. However, more emphasis is given when the dataset is displayed in a form. Reference data Form.

#### Validation Rule

This attribute is to validate the data entered in a column while editing or importing. Earlier there are 2 attributes min and max to validate number data only. That feature will be replaced by this new attribute. It takes partial SQL and returns True (1) or False(0)

Eg: Date of Birth should not be greater than current date:

Select case when row.DOB; > getdate () then 0 else 1 end

#### Validation Text

This works in conjunction to Validation rule. If the validation rule returns 0 then validation text is displayed as error message. If the validation returns 1 then no error.

Eg:

Select 'Dob should be less than Current date'

Row.Columnname; should be used as appropriate. A registration cant be done a minor. So validation text can be:

select Concat('User is ', Row.age; , 'years Old. So he/ she cant be registered')

#### Column Type

This is new to HIMS 8.0. It is proposed to have 2 types of columns as below:

1. Auto Value: This column holds an automatically incremented value or a calculated value from the current Row. Should be updated when a record is added or imported. These columns are used as Primary key columns or for displaying some calculations. So, all these columns are non-editable. If any requirement comes in future to edit these columns, decision can be made at that point. There can be multiple Auto Value columns in a system. Auto Value columns are to be updated by the system directly either when the form is saved or through import. These columns shouldn’t appear in import template and import column mapping.

Following sub-attributes will be there with Auto Value.

* 1. Formula: Formula to be used to get the required output. Output is to be properly setup through a SQL with respect to column datatype. Formatting is through a separate attribute, hence not necessary to handle here. Returns the result of SQL as a single value.

Eg: bridge Code:

select concat('Br:',Row.Roadcode;,':',round(Row.Chainage;/1000.000, 3))

* 1. CurrSeq: This attribute holds last used sequence for a column. Used in case of auto increment columns. Default initial value is zero.

Eg: Consent Form code:

Select Concat('0104 /',right(year(getdate()),2),20,' / ', CurrSeq;)

* 1. Increment by: This is also used in case of Auto increment columns to specify the number with which increment should happen. Sequence increases if a +ve number is given, decreases if a –ve number is given. Default is +1.

Eg: +1 to increase the Sequence by 1 for each record.

*` Developers Note: CurrSeq is the key word to be considered. Currently this value is reflected when the record is saved. It would be much helpful if the value is shown with an action, say for example “Show Value” instead of showing after saving the record.*

1. Input: this is the regular column type that takes an input in to the system. All input columns are editable columns either through form or through Import. These should appear in import template as well as in import column mapping.
2. Read-only column: This will be stored in DB but noit editable from front end.

#### Format Data

Currently formatting options are very limited. This needs to be enhanced as per industry standard. For example, commas are coming in US format only though most of the clients use Indian format. Would be good if excel formatting options are provided.

#### Default Value

If a default value is given, in case no value is supplied in import or edit/ add, this value will be updated to the database. Partial SQL is to be provided with Row. Phrase.

Views

These can be renamed as Custom Datasets for ease of understanding. Though it was discussed to deprecate Views concept, recent dashboard changes of RRAMS realized the importance of having views as below:

1. Once a view is created, we can add any further SQL clauses for filtering, sorting etc
2. Easy to manage with name instead of carrying the whole SQL.
3. Stored in DB, if needed most of the analysis can be done in backend

**Enhancements**

1. Would be good to save the definition (column names) of views.
2. Provision to hide certain columns. For example, if the same view is used in multiple locations, some columns may be hidden at some places.
3. Able to see Videos if Roadcode is attached.
4. Able to view in maps if lat longs are given.
5. Able to change colours and themes in maps
6. Able to attach a Map layer to this view such that, when map opened, the same layer is appeared on screen automatically.

Following columns are to be added to existing SysQueryDef table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Sno** | **Attribute Name** | **DataType** | **Remarks**  **Other notes** |
| 1 | ShowInMap | Boolean |  |
| 2 | Image | Boolean | Whether to display image from base table |
| 3 | Document | Boolean | Whether to display from base table |
| 4 | Video | Boolean | Whether to display image from base table |
| 5 | Global | Boolean |  |
| 6 | IsMobile | Boolean |
| 7 | MapImage | Text |
| 8 | BaseTable | Int | Data table number |

Following Child table needs to be defined to maintain the column definition of the views:

|  |  |  |  |
| --- | --- | --- | --- |
| **Sno** | **Attribute Name** | **DataType** | **Remarks**  **Other notes** |
| 1 | Id |  |  |
| 2 | DataType |  |  |
| 3 | DataTable |  |  |
| 4 | Name |  |  |
| 5 | Caption |  |  |
| 6 | Description |  |  |
| 7 | PrimaryKey | Boolean |  |
| 8 | Column Type |  | Editable/ Non Editable |
| 9 | Visible |  |  |
| 10 | DisplayFormat | Text | Format Code |
| 11 | Sequence | int | To display columns in a particular Seq |
| 12 | Formula |  | TBD |

Following valuation need to be done for views while updating data:

1. Should have primary key columns same as Table to be edited.
2. Editable column names should be same as table name
3. References etc. are to be inherited from base table.
4. Column Type can be Editable/ Non Editable.

Data Forms

Current data forms are to be enhanced to make them more useful. Following are various attributes at form level as well as at Form item level.

### Data Form

The following properties are to be set to a data form.

|  |  |
| --- | --- |
| Column Name | Remarks |
| Id |  |
| Name |  |
| Type | Holds 2 values, D: dataset, S : Stored Procedure. |
| DataSource | This holds either Data set ID or Stored procedure ID depending on Type. |
| HTMLSource | This holds complete HTML of the form. |

### Data Form Item

#### Control Types

In addition to existing control types which are dropdown, text, calendar, a new control Checkbox is added.

Following control types are recommended.

1. Section – Display the whole screen. Allow to add other controls in it.
2. Table – Display the whole screen. Allow to add other controls. Need to be reviewed later. This requires additional property as Data Source to populate the table.
3. HTML Layout – Display the whole scree with HTML control bar.
4. Email – Similar to a normal text box. However, add email properties such as text colour, @ validation, opening outlook when clicked etc.
5. Date – display date control
6. Time – display time control
7. Date & Time – Display date and Time control
8. Drop Down – Normal dropdown
9. List – for multi selection
10. Text Box – Single line
11. Text area – Multi line
12. Check box – Yes/ No. Need to see how Select All checkbox are to be displayed. For now, we can proceed with Y/ N.
13. Radio Button - multiple buttons as per the values. Values are to be populated either as custom or SQL similar to DropDown.
14. File Uploader – options to upload and download
15. URL - a text box with URL features
16. Formatted number – A text box with a specific format. For example, IP Address, Credit card number etc. Format can be defined in Sysformatdef table
17. Number – displays number with specified digits and decimals.
18. Currency – Displays currency Symbol
19. Signature – to upload signatures

#### Dropdown Type

Currently this dropdown accepts only datasets to fill in drop down. This is becoming a tedious job as complex queries can’t be used to fill, and dependencies cant be maintained in dropdowns. Further in case of very few dropdown values also, for example “male/ female”, a dataset is demanded in current version.

Dropdown Types:

* 1. Dataset – Provide Code & description from a dataset. Description is displayed in dropdown and code is used for inner processing. This can be replaced with SQL option to avoid multiple methods and redundancy.
  2. Custom – Add Custom values on the screen as needed by clicking + button.
  3. SQL – Get data from a SQL. This attribute takes full SQL with only 2 columns, Code and Description. This is used for generating dependency dropdowns.

Eg: to display only the districts existing in a selected Province below SQL to be used

Select Code, Desc from Districts where ParentCode=row.Province;

**Row** represents current row from which selected province code is to be taken.

**; -** row.columnname should end with semicolon so as to read the text in .net. This is mandatory for a successful execution of SQL.

*Developers’ Note:*

1. *For coding purpose, developers can take these two columns as Code and Desc****.***
2. *To make it generic to Radio button control as well, column name is changed as ListType and Listtext.*

#### Visible

This attribute should be a condition based one. Holds one of the 3 values, Yes/ No/ Conditional. If the condition returns Y, then the control is visible and if returns N, control is not visible. In case of Conditional, a partial SQL will be executed which returns either Y/ N. to be given.

Eg: Reason for reject column should appear only if approval status is “rejected”.

Select case when row.status;=’”rejected” then “Y” else “N” end

#### Format Columns

A table Sysdataformats to be defined with the following data.

|  |  |  |  |
| --- | --- | --- | --- |
| **Code** | **DataType** | **Format** | **Description** |
| N1 | Number | ###,### | 123,456 |
| N2 | Number | (###,###) | (123,456) - Negative value |
| N3 | Number | #,##,### | 1,23,456 - Indian System |
| N4 | Number | (#,##,###) | (1,23,456) - Indian, negative value |
| N5 | Number | ###.###.###.### | IP Address |
| N6 | Number | ####.####.####.#### | Credit card |
| N5 | Number | ##### | Fixed text. Ex:00015 |
| N6 | Number | (###) ##### ##### | Phone number |
| D1 | Date | dd-mm-yyyy | 20-01-2023 |
| D2 | Date | dd-mmm-yy | 20-Jan-23 |
| D3 | Date | dd-mmm | 20-Jan |
| D4 | Date | mmm-yy | Jan-23 |
| D5 | Date | month-yy | January-23 |
| D6 | Date | mm-dd-yyyy | 01-20-23 |
| D7 | Date | mmm-dd-yy | Jan-20-2023 |
| D8 | Date | mmm-dd | January-20 |
| D9 | Date |  | Friday, January 20, 2023 |
| D10 | Date | dd-mm-yy h:mm am/pm | 20-01-2023 5:00 |
| D11 | Date | dd-mm-yy h:mm | 20-01-2023 17:00 |
| D12 | Date | h:mm Am/ PM | 12:00:00 AM |
| G1 | Generic Code | Alphabet+Numeric+Special Char | |
| G2 | Generic Code | Alphabet |  |
| G3 | Generic Code | Alphabet+ Numeric |  |
| G4 | Generic Code | Alphabet+Special Char | |
| G5 | Generic Code | Capital+Alphabet+Numeric+Special Char | |
| G6 | Generic Code | Capital+Alphabet | |  |
| G7 | Generic Code | Capital+Alphabet+ Numeric | |  |
| G8 | Generic Code | Capital+Alphabet+Special Char | |

Format column of SysformItemsDef should be filled with one of the Codes to format data at display time.

A Column decimals to be added to SysFormItemsDef to specify number of decimals.

#### Additional Columns

New columns for display purpose can be added to a dataform. Most of these columns can be brought from Display SQL of the dataset. However, if further columns are needed, one can add here. These cant be edited. Only the SQL result is shown in them.

Eg:

Totals coulmn: this column is not stored in database. But just to give more information to the data entry user we can have on fly calculation of totals and show on the screen. Value for these columns to be updated when an action is taken. Need to check.

Column Type attribute holds 2 values, D - Dataset Driven, F- Form Driven. Additional columns should be of Form Driven Type. DisplaySQL column to be setup with SQL text for From Driven columns to display a value at run time.

#### Font attributes

Currently no font attributes are provided to columns. It is recommended to give Font type, size, colour, bold italics if not to a big list. Default needs to be set for all columns. CFontcolour is the attribute of Caption column and VFontColour is the attribute of Value column.

#### SysFormItemsDef

Following Table depicts the changes proposed for SysFormItemsDef table.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Sno** | **Group** | | **Attribute Name** | **Remarks** | **Other notes** |
| 1 | Generic | | FormDef | Form ID |  |
| 2 | Generic | | ColumnType | New column introduced to display on-fly values. It holds 2 values, D - Dataset driven, F - Form Driven. Default is dataset driven. |  |
| 3 | Generic | | ColumnName | In case of table type, column name is the column from dataset, taken from table definition. In case of form column, column name is blank. | Changes are proposed as per the HTML concept |
| 4 | Generic | | Caption |  |  |
| 5 | Generic | | LCaption | Need to check how language will be implemented |  |
| 6 | Generic | | ControlID | To be fetch in from HTML | Introduced to Support HTML Design concept |
| 7 | Generic | | Status | Active/ Inactive. If inactive, should not be appeared on the form and should not hold any data. |  |
| 8 | Position | ParentID | | This should be one of the items with in the form. Means, it goes with self join. Out of all, first section will not have Parent. All others should be having a table/ Section as containers/ Parents. It stores ID of the parent container. | These 2 attributes may become obsolete if HTML concept is finalised |
| 9 | Position | Sequence | | required to know order of columns |
| 10 | Validation | AllowNulls | | This specifies if it is mandatory or not. Holds 3 values, Y/ N/C |  |
| 11 | Validation | NullCond | | This needs to be populated only if AllowNulls is C. This column carries a SQL that returns Y/ N. If Y is returned, record can be saved even if no value is supplied. If N is returned, User must supply a value to the column. |  |
| 12 | Validation | Visibility | | Holds 3 values, Y/ N/C |  |
| 13 | Validation | VisibleCond | | Similar to NullCond |  |
| 14 | Validation | Editable | | Holds 3 values, Y/ N/ C. |  |
| 15 | Validation | EditableCond | | Similar to NullCond |  |
| 16 | Validation | Validation Rule | | Carries a Partial SQL to check the value entered in this column. Should be executed when one tabs out of the control. Should return Y/N.  If Y then data can be saved, if N then should display error text mentioned in Validation text Column. |  |
| 17 | Validation | Validation Text | | Carries a partial SQL that needs to be displayed when Validation Rule returns N |  |
| 18 | Populate | ControlType | | Various control types as per the list provided. There could be specific attributes for certain controls. Ex: Custom list | This needs to be populated from HTML as per the new approach. |
| 19 | Populate | DataSource | | This property is applicable to Table type control only. This carries a SQL to populate the table. |  |
| 20 | Populate | Tabletype | | R/ E. Read only or Editable. | Need to investigate further if a table is made editable. |
| 21 | Populate | DisplaySQL | | This is to be given only if ColumnType is F. Carries a normal SQL with parameters. |  |
| 22 | Populate | ListType | | Holds 2 values, S - SQL, C- Custom. Applicable to Radio buttons and Dropdowns |  |
| 23 | Populate | ListText | | Applicable to Radio buttons and Dropdowns.  Holds a SQL or custom list with 2 values. First column should be treated as Value/Code and Second value should be treated as display value SQL Ex: Select Code, Description from LkptCategorytype where BU=Row.BU; This returns the data from Categroy look up table matching with selected BU on the screen. |  |
| 24 | Populate | CustomList | | This column may be continued for some time until Developers get hold on the Definitions. For now, provide value, column format. Should check with developers.  Once done, this can be shifted to Listtext column itself, means, same column holds either SQL or customlist and depending Listtype, rendering should happen. |  |
|  | Format | DateFormat | | This column should be renamed as FormatCode and one of the codes to be stored in this column. |  |
|  | Format | Decimals | | Takes numbers from 0 to 10. Default is Zero. |  |
|  | Format | HTMLText | | In case of HTML type of control | Need to see if this control is useful if the Design HTML concept is introduced |
| 25 | Font/ Format | CFontItalics | | Boolean | If HTML concept is introduced, all these properties will become obsolete. |
| 26 | Font/ Format | CFontBold | | Boolean |
| 27 | Font/ Format | CFontUnderline | | Boolean |
| 28 | Font/ Format | CFontStrikethru | | Boolean |
| 29 | Font/ Format | VFontCode | | Good to maintain a SysFontsdef table for list |
| 30 | Font/ Format | VFontColour | | HTML Colour codes. Ex: #FF5733 |
| 31 | Font/ Format | VFontSize | | Any number |
| 32 | Font/ Format | VFontItalics | | Boolean |
| 33 | Font/ Format | VFontBold | | Boolean |
| 34 | Font/ Format | VFontUnderline | | Boolean |
| 35 | Font/ Format | VFontStrikethru | | Boolean |
| 36 | Obsolete | FKTblNm | | These 3 columns may be removed to accommodate conditional SQLs. Instead, add a column as DropDownType with values (S- SQL/C- Custom) |  |
| 37 | Obsolete | FKColVal | |  |  |
| 38 | Obsolete | FKColDisp | |  |  |
| 39 | Obsolete | isDeleted | | This column can be removed. No specific functionality as of now. |  |
| 40 | Obsolete | ColMerge | | These may be deleted as the forms concept is taking a different shape. |  |
| 41 | Obsolete | RowMerge | |  |  |
| 42 | Obsolete | RowCnt | |  |  |

Stored procedures

It is discussed to have 2 types of stored procedures. 1. Business Logic SPs (BL) 2. Reports.

1. Business Logic SPs: As the name suggests, they are mainly used to execute business logic. They take user parameters as well as System Parameters. Following system parameters are considered.
   1. UserID : Gives the ID of logged in user
   2. SurveyID: Survey year/ ID selected at the top most corner of the application

These SPs will not return any SQL output. These are executed from any data form page or as a part of the batch. Dependencies can be set from one SPs to other in case of running in a Batch. These can be used for workflow approvals.

1. Report SPs (RPTs): These are mainly for standard and custom reports. They are executed from Reports page following report definition.

Following items are the Improvements needed from older version to new version:

1. SP text should be manageable from application only. Currently there are limitations of text size resulting in managing the SQL through backend only. It would be good if Query analyser can be integrated in web application. This helps in syntax checking and viewing end results.
2. Parameters should be of 2 types, 1. SQL 2. Custom
3. SP parameters are to be further enhanced to have dependencies among columns.

Eg: State > District > Division etc.

1. Duplicate values are coming in parameters. Should be avoided.

Reports

HIMS will have 2 types of reports.

1. Standard reports – RDLC based reports with data and charts. Supports filters, formatting options.
2. Custom reports – These are SQL based reports that are displayed in grid. Supports column add/ remove, filters (on columns with and and or), sort and report title. Can be downloaded as PDF/ Excel. SQLs associated with tables are joined and displayed the results. SQL to be stored with the user. We can explore the option of public or private report. Public reports are visible to all and private reports are visible only to the current user.
3. System Reports – These are reports specific to HIMS system and come along with the application. One can run these reports from Admin module.

### Standard Reports

Following features are to be added to current Standard Reports:

1. Dependency Dropdowns: Currently we have to select the entire hierarchy to get the required result. It would be useful if the dropdowns can be filled even if the preceding parent is not selected.
2. Download RDLC: Currently there is no way to download the RDLC report that was created earlier. This is creating confusion and raising need for multiple copies when the report is to be edited. If a provision is given to open report designer from the application, that looks even better, though report designer is a desktop component.
3. Report should allow multiple data sets.

### Custom reports

Design/ config: following features are to be included in Design phase:

1. Remove current Query feature.
2. Make SP as data source to report. Queries are eliminated purposefully as they are not able to support parameters.
3. These rep
4. Following attributes are to be setup for Custom report:

*Note: Need to check how this data to be entered into system.*

Report Def:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sno** | **Group** | **Attribute Name** | **Remarks** | **Other notes** |
| 1 | Custom | Report Name |  |  |
| 2 | Generic | Report title |  |  |
| 3 | Generic | Sort by | Column names separated by commas |  |
|  |  |  |  |  |

Report Item Def:

One has to define this manually depending on the output of the SPs. Though it was discussed to prepare definition, it not practical to maintain definition when SQLs may change regularly.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sno** | **Group** | **Attribute Name** | **Remarks** | **Other notes** |
| 1 | Custom | Report ID |  |  |
| 2 | Generic | ColumnName | This column name should match with the column names of the SQL output from Stored procedure. SP should have names to each column. System should compare SP output with that of the definition for formatting captions etc |  |
| 3 | Generic | Caption |  |  |
|  |  | Sequence | This may be removed if the maintenance becomes complicated. Instead, one can follow the sequence coming from SP. |  |
|  |  | Format | Standard formats. This has to relationship with table definition. |  |

Report execution:

Once the report is defined, user can further configure following items as per his requirement. These are specific to user and runs only for that user. When a user saves a report, initially all columns in ReportItemdef are copied to ReportItemUserdef table.

ReportItemUserdef:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sno** | **Group** | **Attribute Name** | **Remarks** | **Other notes** |
| 1 | Custom | Report ID |  |  |
|  |  | UserID | Name of the user who saved the report. Only this will be executed when he runs which may be different for other users. |  |
|  |  | Report Title |  |  |
| 2 | Generic | ColumnID |  |  |
| 3 | Generic | Caption |  |  |
|  |  | Visible | Y/N |  |
|  |  | Format | If different from SP definition. |  |
|  |  | DisplaySequence | Specific to user |  |
|  |  | Filtertext | Specific to user |  |
|  |  | SortOrder | 1 asc,3 desc,5 desc or Roadid Asc, Roadname Desc (Names should match) |  |

### System Reports

1. List of users with their access levels. This should have a parameter for active and inactive users. Drill down report for data access.
2. List of Currently Connected users
3. List of processes still in active stage.
4. List of tables report by module
5. Table structure report
6. List of Views by module.
7. Tables relationship report
8. Object dependency report
9. Dashboard definition report by module

Workflow

Workflows are mainly for approval process and to assign the job to authority next in queue. Following terminology is used for workflow management.

1. Workflow: It refers to business flow that includes, various characters such as initiator, approver etc.
2. Workflow Item: An instance of a workflow is called workflow item. There could be multiple workflow items waiting for someone to be approved. For example, there could be 5 invoices waiting for approvals. These 5 are called workflow instances. Business flow behind Invoice is called workflow.
3. Manage Workflow: It refers to the process of approving or rejecting a workflow instance. This is done through Workflow management screen.

Eg: A user has 2 workflows assigned to him, 1. Invoice workflow, 2. MIS reports workflow. In each workflow, multiple workflow items may wait for approvals.

### Define Workflow

Similar to defining a table/ SP, a workflow definition page is used to define the workflow. This page should be managed by administrator only. Following system tables are needed for defining workflows.

1. Sys Workflow Definition

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Table Name:** | | SysWrkflwDef | | | |
| **Description:** | | Maintains master data of Workflow definition | | | |
| **Sno** | **Column Name** | | **Primary key** | **Remarks** | **Other notes** |
|  | WorkflowID | | Y |  |  |
|  | Name | |  |  |  |
|  | Description | |  |  |  |
|  | DisplaySQL | |  |  | SQL that gives information for approval |
|  | UpdateSQL | |  |  | SQL for updating data with current status. That table should have two columns as WorkflowStatus and Remarks. |
|  | Level | |  | No of approval levels. 1/2/3/4/5 |  |

1. Sys Workflow Level Definition

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Table Name:** | | SysWrkflwLvlDef | | | | |
| **Description:** | | Maintains details of various levels of approval | | | | |
| **Sno** | **Column Name** | **Primary Key** | | **Remarks** | | **Other notes** |
|  | WorkflowID | Y |  | |  | |
|  | Level | Y | In accordance with SysWorkflowDef table | |  | |
|  | Role |  | Initiator/ Approver/ Final Approver | | For now, initiator and approver will have similar functionality where they approve and assign to next level user. In case of Final Approver, no assigned to pop up is needed. All users can see the status of the Workflow instance once assigned to them. | |
|  | Assign to |  | Group1/ Group2 | | Let us keep assignee groups separate to make it common for all kinds of list. | |
|  | Action | Y | Approved/rejected | |  | |
|  | AssigntoHeadOnly |  | T/F | | If False selected, a dropdown appears with list of users in the selected Group. | |
|  | CanSkipTo |  | 3 | | Give a level number to which workflow approval can be skipped in case of nonavailability of People | |
|  | EmailTo |  |  | | To whom email should be sent | |
|  | CC |  |  | | List of CC emailIDs | |
|  | EmailTemplate |  |  | | Template to be followed. Need to detail further. | |
|  | Workflowstatus |  |  | | Status to be updated to the main record | |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **WorkflowID** | **Level** | **Role** | **Action** | **Assign to** | **AssigntoHeadOnly** | **CanSkipTo** | **Email to** | **CC** | **Email Template** | **WorkflowStatus** |
| 1 | 1 | Initiator | Approved | Helpdesk | Y |  | L2 Individual | [Helpdesk@satragroup.in](mailto:Helpdesk@satragroup.in) |  | Assigned to Helpdesk |
| 1 | 2 | Approver | Approved | Finance | Y |  | L3 Individual | [Accounts@satragroup.in](mailto:Accounts@satragroup.in) |  | Assigned to Finance |
| 1 | 3 | EndAction | Approved | Finance |  |  | L1 Individual | Accounts@satragroup.in, helpdesk@satragroup.in |  | Paid |
| 1 | 1 | Initiator | rejected | Helpdesk | Y |  | L2 Individual | [Helpdesk@satragroup.in](mailto:Helpdesk@satragroup.in) |  | Initiator Rejected |
| 1 | 2 | Approver | rejected | Finance | Y |  | L3 Individual | [Accounts@satragroup.in](mailto:Accounts@satragroup.in) |  | Helpdesk rejected |
| 1 | 3 | EndAction | Rejected | Finance |  |  | L1 Individual | Accounts@satragroup.in, helpdesk@satragroup.in |  | Finance rejected |

Sample data for Workflow level Def 1

### Define Groups

Assignee Groups: Following table(s) with data can be maintained for assignment purpose.

1. Sys Workflow Group Definition

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table Name:** | | SysWrkflwGrpDef | | |
| **Description:** | | Maintains details of various groups to be used for workflow | | |
| **Sno** | **Column Name** | **Primary Key** | **Remarks** | **Other notes** |
|  | GroupID | Y |  |  |
|  | Name |  |  |  |
|  | Description |  |  |  |
|  | Status |  | Active/Inactive |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **GrpID** | **Name** | **Description** | **Status** |
| 1 | Finance | Business Unit | Active |
| 2 | Helpdesk |  |  |
| 3 | HR |  |  |
| 4 | Accounts |  |  |
| 5 | Management |  |  |

Sample data of SysWrkflwGrpDef

1. Sys Workflow User Definition:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Table Name:** | | SysWrkflwUsrDef | | | |
| **Description:** | | Maintains details of various users to be involved in workflow | | | |
| **Sno** | **Column Name** | **Primary Key** | **Remarks** | **Other notes** | |
|  | ID |  |  |  |
|  | GroupID | Y |  |  |
|  | UserID | Y |  | Better use UserIDs from userdef table. |
|  | Display name |  |  | From userdef table |
|  | Position |  |  | As per the workfow |
|  | Group head |  | Y/N |  |

*Note: One user can be part of any number of groups.*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | GroupID | UserID | Display name | Position | Group head |
| 1 | 1 | Bharanin | Bharani Nandigam | Manager Accounts |  |
| 2 | 1 | AnilM | Anil Merugumala | Manager Accounts |  |
| 3 | 1 | RavitejaI | raviteja Injam | Accountant |  |
| 4 | 1 | Durgak | Durga Konduru | Director | Y |
| 5 | 1 | Rajm | Raj Mallela | Managing Director | Y |
| 6 | 2 | venkateshG | Venkatesh Gajjela | Executive network Engineer |  |
| 7 | 2 | DharmaC | Dharma Cirisinagandla | DGM |  |
| 8 | 2 | Alfredm | Alfred Marshall | GM | Y |
| 9 | 2 | Durgak | Durga Konduru | Director | Y |

**Sample data of SysWorkflowUserdef**

### Define Workflow Instances

1. SysWorkflowinstances

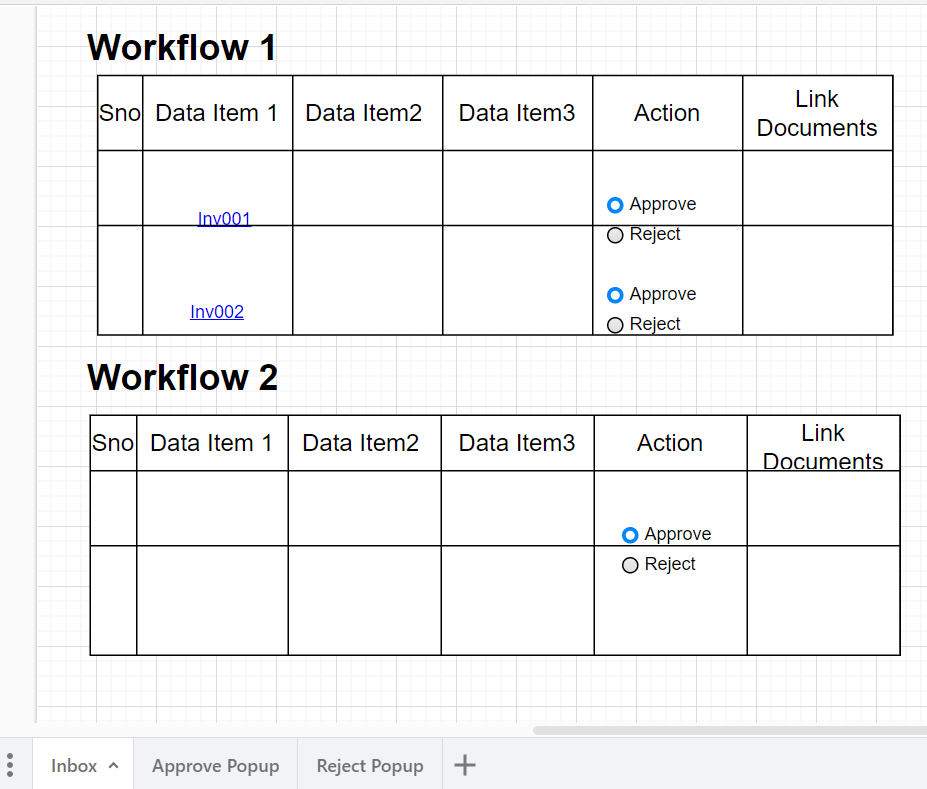
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table Name:** | | SysWrkflwInst | | |
| **Description:** | | Maintains details of workflow instances. This transaction data. | | |
| **Sno** | **Column Name** | **Primary Key** | **Remarks** | **Other notes** |
|  | ID |  |  |  |
|  | WrkflwID | Y |  |  |
|  | Status | Active/ Inactive | Active/Inactive | From userdef table |
|  | CurrentLevel |  | 1,2,3 etc | Details of the level it is assigned to |
|  | Initiatedon |  | DateTime |  |
|  | LastUpdatedon |  | DateTime |  |
|  | LastUpdatedBy |  |  |  |

1. Sys Workflow Instances History

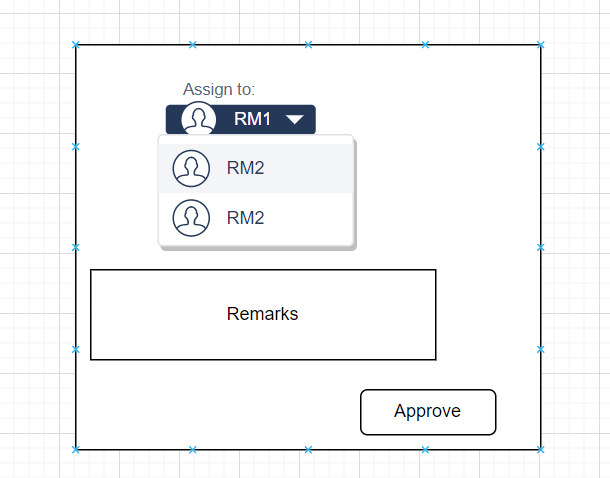
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Table Name:** | | SysWrkflwInstHstry | | | |
| **Description:** | | Maintains details of workflow instances and their history. This is history of transaction data. | | | |
| **Sno** | **Column Name** | | **Primary Key** | **Remarks** | **Other notes** |
|  | ID | |  |  |  |
|  | WrkflwID | | Y |  |  |
|  | InstanceID | | Y |  |  |
|  | Role | |  |  |  |
|  | Level | |  |  |  |
|  | StatusChanged to | |  | Approved/ rejected |  |
|  | ActedOn | |  | DateTime |  |
|  | Actedby | |  |  |  |
|  | Remarks | |  |  |  |

### Manage Workflow

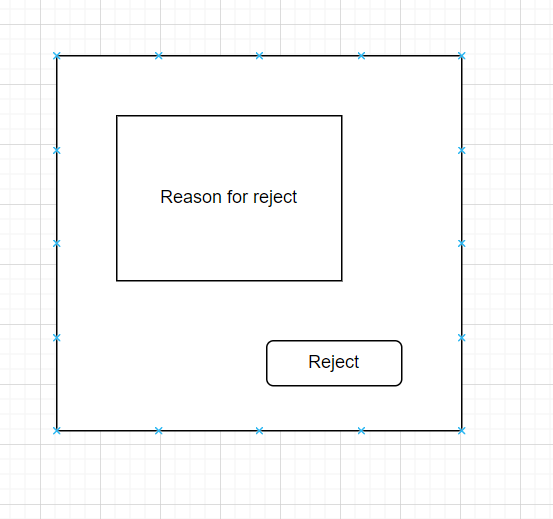
Following screen is used to manage workflow approvals and rejection. It would be good if this page can be opened somewhere from Dashboard screen, so that users doesn’t need to navigate much after logging into the system for approvals.



Sample Manage Workflow Screen



Approve Workflow Popup



Reject Workflow Popup

### Initiate Workflow

Workflow can be initiated when a record is saved. We may need to add an attribute as workflow item to a page property. It can be initiated when a process is run as well. Since processes are also run through Data form, we can use same technique for SPs also.

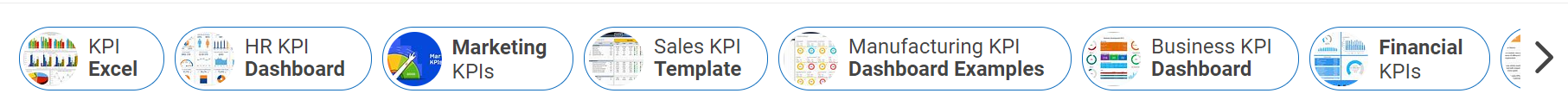
Dashboards

It is discussed to continue with current Dashboards (configurable) with some enhancements. Following feedback received from current dashboards. Enhancements should address these ones.

1. Not user friendly
2. Outdated charts with minimum functionality.
3. Drill downs not available.
4. Hiding or revealing certain information is not possible.
5. KPIs are not showing the data behind.
6. Link code is not necessary to be displayed in filters.

### KPIs

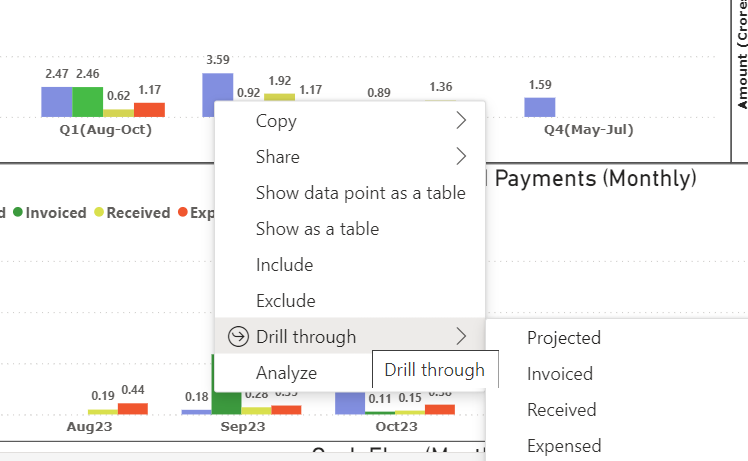
Current KPIs are to be enhanced with respect to presentation. Configuration part remains the same. However, configuration screen needs to be updated to make it more user friendly. User should be able to configure any number of KPIs and can be viewed by expanding them similar to below screen.



KPIs design and size need to be defined as per the current trends.

### Charts

Charts will have similar functionality, however, new enhanced controls are to be implemented to get more features such as drilldowns, filters within the chart etc. Below image reflects the expected result.



If user selects a specific bar, that should lead us to another chart in the same area with X axis changing to the selected item.

For example, a chart is displayed for Zonewise Carriageway type. Zones are taken on xaxis. If a bar pertaining to a zone is selected, and right clicked, it gives drill through options as District, Circle, Division and Subdivision. If division is selected, then X axis should replace zones with Subdivisions and data to be filtered for the selected zone (Bar). The same thing can happen on Y axis in case of Column charts. What columns can be considered for drilldown needs to be configured.

Current filter options exists as is. However, currently bar graphs have no filters on x axis. This needs to be implemented.

Charts will be standard height and width can be adjusted in percentage.

Maps

Following features are to be added to Maps in addition to what is given in version 7.5

1. Able to display Raster images.
2. Export Shape data to selected users. While exporting, it should export shape as well as data associated with that.
3. Export nonspatial data.
4. Ability to change themes at runtime.

Ref: <https://developers.arcgis.com/experience-builder/guide/theme-development/>

### Project Creation

This is similar to ArcMap concept where in, user stores his layers of interest in a specific object called Project. Map screen should have options to open a project, create project and Save project which are specific to that user only.

When a new project is selected, he can add as many layers as needed from SysMap Layers, external (Other web services) or local( for example, .CSVs from C:\ or D:\). Once the layers are added, user will save that project. When user logs in, he can open the already saved Map project to get all earlier settings.

Following are to be stored with the project:

1. Layer details including name, caption, type(internal - can be renamed as application / external/ local)
2. Order of Layers
3. Theming if any

# Access Levels

## Challenges in current version

HIMS 7.5 version supports access levels at module, data and at various operations such as Read/ Edit/ Define. However there is no concept of giving access to a specific data object such as , a selected table/ report etc. A user having readonly access to a module can be able to see all tables within that module.

Its been decided to further enhance this to bring the concept of Permissions and roles in HIMS 8.0 so as to provide access at data item level as well.

Below tables depict the functionality in HIMS 7.5.

|  |  |  |  |
| --- | --- | --- | --- |
| Object | Reader | Editor | Administrator |
| Tables | 1. Display in grid 2. Display in map 3. Export | 1. Add record 2. Modify record 3. Import   *Delete is not considered for now.* | 1. Add new table 2. Change definition 3. Delete definition |
| Reports | 1. Run report 2. Export to PDF |  | 1. Define new reports |
| Dashboards | 1. Accessible |  | 1. Define new dashboards |
| Views | 1. Accessible in grid  2. Export to PDF |  | 1. Define views |
| Queries | 1. Accessible in grid 2. Export to PDF |  | 1. Define views |
| Charts | 1. Accessible |  | 1. Define new charts |
| Groups | 1. Accessible |  | 1. Define new groups |
| Batch process |  |  | 1. Execute batch processes. |
| Tools |  |  | 1. Execute operations |

Object level Access in HIMS 7.5

|  |  |  |
| --- | --- | --- |
| Level | Hierarchy | remarks |
| Geo Level 1 - Zone | Yes |  |
| Geo level 2 - Circle | Yes |  |
| Geo level -3 District | Yes |  |
| Geo level -4 - Division | Yes |  |
| Geo level -5 Subdivision | Yes |  |
| Agency | No | Configurable |

Data level access in HIMS 7.5

## Proposed Solution

In HIMS 8.0, it is proposed to define User Access using permissions and roles. The advantage of this method is that one can give access to a selected object(s).

### Permission

Permission is one of the security data objects that maintains list of items for which the access cam be given to. In addition it also maintains type of access such as read only/ read & write / Execute (for batch processes) permission is applied on 1 module only which means, if multiple modules are to be accessed, multiple permissions are to be given.

One can create as many permissions as needed.

Create Permission page displays list of objects as applicable permissions as below:

|  |  |  |  |
| --- | --- | --- | --- |
| Object Type | Item Name | Access Level | Remarks |
| Table | TableName | Read Only/ R & W |  |
| Views/ reports/ Queries/ Charts/ Strip maps / Map/Stick Maps | View Name/ Reports Name/ Query Name | Read only |  |
| Dashboard | Not Applicable | Read Only |  |
| Batch process | Batch process Name | Execute |  |
| Tools | Tool Title | Execute |  |

Objects and access levels - HIMS 8.0

**SysPermissionDef**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Column Name | DataType | Lookup | Primary Key | Remarks |  |
| Name | Varchar(50) |  | Y | Don’t allow special characters other than “UnderScore” |  |
| Description | Varchar(100) |  |  | Comes as tooltip |  |
| Module |  |  | Y |  |  |

Table Structure - SysPermissionDef

**SysPermissionItemDef**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Column Name | DataType | Lookup | Primary Key | Remarks |
| PermissionID | int |  | Y | Don’t allow special characters other than “UnderScore” |
| ObjectType |  | Table/View/  Report/  Query/  Batch process/  Dashboard |  |  |
| ObjectID | Varchar(100) |  |  | Comes as tooltip |
| Read Only | Boolean |  |  |  |
| Read and Write | Boolean |  |  |  |
| Execute | Boolean |  |  |  |

Table Structure - SysPermissionItemDef

### Roles

Roles is a combination of many permissions of a selected module. Any number of permissions can be added to a role. Roles are also specific to module.

Module administrator is a special role which will not have any permissions attached to it. However, it gives access to define all objects. The title can be changed as appropriate.

**SysRoleDef**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Column Name | DataType | Lookup | Primary Key | Remarks |  |
| Name | Varchar(50) |  | Y | Don’t allow special characters other than “UnderScore” |  |
| Description | Varchar(100) |  |  | Comes as tooltip |  |
| Module |  |  | Y |  |  |

Table Structure - SysRoleDef

**SysRolePermMapping**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Column Name | DataType | Lookup | Primary Key | Remarks |  |
| RoleID | Varchar(50) |  | Y | Don’t allow special characters other than “UnderScore” |  |
| Permission ID | Varchar(50) |  | Y | Comes as tooltip |  |

Table Structure - SysRolePermMap

### User

In addition to current functionality, roles will be added to users.

**SysUserRoleMapping**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Column Name | DataType | Lookup | Primary Key | Remarks |  |
| UserID | Varchar(50) |  | Y |  |  |
| ModuleID |  |  | Y |  |  |
| RoleID |  |  | Y |  |  |

Table Structure - SysUserRoleMapping

### Groups

### Other items

In current system, system administrator is having complete access to the system as well as data. Idea is that he will be superior to all and can perform every operation. However, in reality, a person who has full rights on the data is someone in SE or abpve position who are not actually involved in performing these activities. Hence it is necessary to redefine the role of System Administrator to manage perform activities needed for smooth running of the system. For example, clearing old Batch processes, queues, scheduling jobs, defining objects etc which are not connected to any data access.

1. Initially, Module / System administrator can be confined to Manage Objects area. It is good to have a SQL developer for this position.
2. User creation and access to the module can be given at module level as well as at Application level. Result is same at either of the levels.
3. Application level should allow, cross module access.
4. Administrator can get access to Datasets and Views by giving access as Reader/ Editor. Data access concepts are same for module administrator, system administrator and super administrator. Means, data access depends on what levels are assigned to that administrator.
5. System should have provision to add other items like, Agency/ Business Unit to define user access. Master table name, and column name will be mentioned at system configuration.
6. Any table/ view/ query having the configured column will be filtered automatically with respect to user access.
7. System should have standard views to provide eloborative data with respect to user access. More details are provided in Views sections.

### Reports

Following reports are to be created for the system administrators and module administrators. Reports specific to module can be displayed at module as well.

1. List of Active users
2. List of Inactive users
3. List of users having access to a selected module(s) along with their permission levels.
4. List of read-only users to a module(s)
5. List of editors to a module(s)
6. List of module administrators(s)
7. List of users having access to a demo graphic area

# Error Logs

User Specific running logs to be displayed in a separate window which can be switched on/off by the user. All Error messages are to be properly numbered, such that one can search for those numbers instead of text for solutions.

# Help

Application should have help feature in a more interactive way. Currently, user manuals are attached to the application. But, it should be designed to search for key words or error numbers etc.

# Desktop features

1. Road Master creation
2. Link Master creation
3. LRMS may go as desktop feature
4. PMS may go as desktop feature

Advantages:

1. No Session outs
2. No need to login multiple times
3. Reduces load on the server.

# Functionality shortfalls

1. When a record is deleted from Master table, it should delete data from all child tables (where Master table primary key is set as foreign key in child).
   1. Need to alert user about what tables’ data will be deleted.
   2. Delete once confirmed.
2. Code columns (ex: Bridge Code etc) should not allow space as one of the characters.
3. In every object definition screen, a new tab Dependencies to be added.
   1. It should list out all other objects with in the database which use these objects.
   2. For avoiding complications, it is proposed to delete an object only of there are no dependencies. For example, if a table has a dependency View, one should delete View before deleting Table. This is proposed to avoid cascade dependencies.
4. For objects like queries, there should be a screen to add dependencies. Only these dependencies will be considered while delete the object.
   1. This is because, queries may be formed with complexity and developer can’t read the text to under the dependent objects.
5. A process to bulk delete the data to be given.
   1. This should be given to module administrators with a facility to restore the data(last action).
   2. Only one set to be maintained for restore option.
6. There should be only one active module administrator in a system.
   1. This is to keep control at one place.
7. Issues with dropdown:
   1. Currently dropdowns are filled with the data from a standard table. This is causing problem when a specific logic is to be executed. For ex: in F & F table, I would like to get only list of employees who has already left. Currently this is not possible. Even if other workarounds are found, for example updating a temporary table with relevant data, some of the scenarios are still kept pending.
8. Road Videos are displayed for only 1 year. Need to have a mechanism to store multiple years of Videos.
9. Provision to be given for system reports. Such that these reports will be added automatically at the time of application deployment. Eg: User Details.

# Grievance Management Application (GMA) – Not Completed

Following functionality is to be implemented in GMA module:

Module 1:

1. The system should make available, the entire data of all the complaints and the status of the complaints at any given time to departmental users/Citizens (through portal), through simple queries and aggregation, with the ability to search, filter and sort
2. The system should facilitate PHED officials to report all the complaints received and processed by them, along with computation of TATs, and analysis on reasons for delay
3. The system should facilitate citizens to file complaints/provide feedback on account of services provided by the PHED
4. The system should allow citizens to view and escalate the complaint to higher levels based on defined compliance rules
5. The system should ensure timely resolution of the complaints by allowing defined turnaround time (TAT) for different complaint types, and escalating issues that exceed expected resolution times.
6. The system should allow PHED to analyze the complaints with the help of ready to use MIS reports, queries and drill-down tabular reports & graphs, based on ageing combinations of: Complaint Category, division wise details, comparative charts, benchmarking amongst divisions etc.
7. Functional Architecture The PHED grievance system should act as a centralized database system for all the grievances. GMA should accept the input in the following scenarios.
8. Citizens lodge the complaint directly into the portal, which is downloaded by the PHED officials for further processing.
9. Complaints registered via e-mail, FAX, call, letter etc. should be registered through an appropriate interface in the application.
10. Complainants visit GMA to track the complaint Status
11. Functional Architecture (GMA The token number should be generated for all complaints.
12. Complaint Number Generation: The unique number for complaints should be of the format - /YY. The running serial number should be reset every year. This will ensure that the token number is always unique. It may be noted that a generic token number definition may be required, as a two letter code may be assigned for the division within the token number as a prefix. This is especially important as it may be desirable to keep running numbers for complaints within a division, yet be able to distinguish complaints at state level by division and year, as well as department. A minimum number of digits may also be specified for the complaint number (e.g.”001” and not 1).
13. Escalation: The system will escalate the complaint, in case it is not resolved in the specified time. However, the complainant may also choose to escalate, which can be allowed based on pre-defined rules.
14. User Profiles:
    1. Citizens:
       1. Lodge complaints
       2. Check status of complaints
       3. Escalate complaints
       4. Download registered complaints
       5. Take print- out of complaints
       6. Provide feedback on the resolution provided
    2. PHED:
       1. Receive complaints
       2. Process complaints
       3. Upload complaints
       4. Action on escalated complaints
       5. Generate MIS reports
    3. Grievance cell Users:
       1. Monitor escalated complaints
       2. Analysis on the grievance
       3. Analysis of grievance management of the Urban Sector
       4. Analysis of grievance by geography, and classes of grievances
       5. View grievance status and details
15. Grievance page:
    1. Access to the Grievance input form
    2. Upload / transfer of grievances to GMA portal
    3. A fully functional dashboard, customized for the user location, division and access rights, will provide:
       1. List of pending grievances, with provision to view, and update
       2. Interface to see feedback from portal
       3. MIS reports - service category-wise, service sub-category wise, overall, analytical reports, benchmarking of a particular division against others etc. Reports should have capabilities of graphical representations, aggregated tabular and drill down functionality
16. PHED’s User Interface & Dashboard:
    1. To escalate the grievances
    2. To view and update status & comments of grievances
    3. To be able to track and monitor the grievance handled by divisions
    4. Detailed MIS Reports - overall, Division wise, benchmarking of Divisions amongst each other. Detailed 45 Sr. Module Requirement Description report for each division (as described above)
17. 19 Citizens, Business, & other agencies 1. Access to Grievance input form. An option to prevent multiple complaints for a single issue in aspecific service area may also be considered for the complainant, based on authentication. 2. Access to view status of complaints 3. Access to escalate complaints 4. Ability to provide additional information, if needed by PHED For all citizens, 1. Basic statistics on department wise complaint inventory and resolution, average TATs, complaints types, and trends in complaint resolution.
18. 20 Grievance Resolution Workflow The status of the complaint should be REGISTERED & OPEN when the complaint is entered into the portal
19. 21 The system should change the status of the complaint to REJECTED when the PHED rejects the complaint.
20. 22 The system should change the status of the complaint to RESOLVED when the PHED resolves the complaint
21. 23 An option to assign a complaint as “NON RESOLVABLE”, with corresponding reason (any may be a date when the complaint may be resolved in the future), is required. This will be especially true in cases where capital expenditure or approvals are required for the provision of specific services in defined areas.
22. 24 The system should change the status of the complaint to ESCALATED when the complaint is escalated manually or automatically.
23. 25 The system should change the status to CLOSED when the request for closure is accepted by PHED. The final status – RESOLVED, NON-RESOLVABLE or REJECTED should also remain as a qualifier.
24. 26 The system should change the status to REOPEN when the ESCALATED complaint is accepted by the PHED
25. 27 The system should allow the complaint in the REOPEN state to be changed to RESOLVED, or any of the options listed above
26. 28 The system should allow the status to INPUT PENDING FROM COMPLAINANT when the PHED sought information from the complainant
27. 29 The system should change the status of the complaints in the RESOLVED or REJECTED state to CLOSED state if no action is taken on it for a stipulated time. The system shall send the reminders to concerned officers before taking any action. . The final status – RESOLVED, NON-RESOLVABLE or REJECTED should also remain as a qualifier 46 Sr. Module Requirement Description
28. 30 Complaint registration The system should allow citizens to enter the complaints directly into the portal.
29. 31 The system should allow citizens to register themselves on the portal. Verification of users will be done through the mechanism of sending activation codes on mobile / e-mail. Password generation rules will need to be defined. Password will be stored as digest. Forgot password and standard web features shall be enabled. The system should keep the history of complaints/feedback registered/provided & subsequent resolution provided by the registered users.
30. 32 The system should allow citizens to lodge a complaint without even registering themselves on the portal.
31. 33 The system should capture the following minimum details: - Complainant details including but not limited to name, address, district, block, panchayat, contact details, APL/BPL, profession etc. and complaint details including but not limited to complaint category/sub-category, first time complaint or escalation, type of complaint, details of complaint. System should allow the user to upload documents in .pdf, picture, .doc etc. format. In case the complaint is a repeat complaint, previous complaint details may also be captured.
32. 34 The system should generate Unique ID on successful registration of the complaint
33. 35 SMS or eMail should be sent to complainant, as well as the concerned officer of the PHED.
34. 36 An SMS should be sent to the complainant with the registration code along with very brief summary of complaint.
35. 37 A consolidated SMS should be sent to mid/senior level officers during working hours on a defined frequency. It may be noted that the status of all complaints will be available on-line.
36. 38 The system should provide an interface to the PHED divisions/state office to view/download the complaints.
37. 39 Complaint received by CGRC The system should allow cell to login to the portal and lodge a complaint on behalf of the citizens based on the complaint received by them through call, letter, FAX, personal visit etc.
38. 40 Escalation of complaint by System (Auto Escalation) The system should have an auto-escalation process, which would run on a scheduled basis
39. 41 The auto-escalation process should search for complaints with REGISTERED status with update date of the complaint more than the number of days defined in the master and send the complaints to the escalation engine. The escalation engine may 47 Sr. Module Requirement Description support multi-tier escalations and will provide for confirmation of receipt of the complaint, as well as provision to escalate to officers with temporary charge in case the primary resource is absent / not available.
40. 42 Escalation by PHED The system should allow the PHED grievance cell to escalate the complaints from the user, and mark them as priority.
41. 43 The system should mark the status of the complaints escalated by PHED as PHED ESCALATED and mark the complaint as HIGH priority complaint.
42. 44 Escalation Engine The system should identify the concerned user defined in the master on the basis of divisions, service type and functional area. This user should be set as the action user.
43. 45 The system should send a mail to the action user giving details of the complaint.
44. 46 The status of the complaint should be changed to ESCALATED.
45. 47 Action on Escalation The system should allow the action user to login to the portal and view the escalated complaints. At intervals, consolidated complaints – with new complaints and status of older complaints will be “pushed”, through e-mail to the action user. Copies may be marked to other officials based on rules defined in the workflow.
46. 48 The action user should be allowed to either accept the escalated complaint or request for closure of the complaint to PHED.
47. 49 The system should change the status of the complaint to REOPEN when the complaint is accepted by the action user.
48. 50 The system should send the complaint to the state admin user if the complaint is sent for closure.
49. 51 Closure Approval by PHED The system should allow the PHED user to either accept or reject the request sent for closure.
50. 52 If the request is rejected, the complaint status should remain escalated. If the request is accepted, the complaint status should change to CLOSED. A complete audit trail is expected to be maintained for all transactions.
51. 53 The system should send notification to complainant regarding closure of the complaint through SMS and e-mail
52. 54 SLA Engine The system should run an automated process, which would search for the complaints in the following state: ESCALATED, REOPEN, NON RESOLVABLE and PHED ESCALATED.
53. 55 The system should identify the complaint level of each complaint, on the basis of the PHED, product type, functional area and complaint type. Each complaint level will have a defined number of days associated with it to act on the 48 Sr. Module Requirement Description complaint in the master setup. This may also be customizable at type of service level.
54. 56 If the complaint status has remained in the current state for more number of days than the number of days defined in the SLA master, the system should notify the users defined in the SLA master. A reminder for such notification should be sent to the action user prior to the escalation.
55. 57 Check Status and View Details The system should allow the complainant to search for the complaint on the portal on the basis of Token Number. The compliant may also be visible to the user in case he has created a login profile, and either registered the complaint on the net, or provided his login details at the time of complaint.
56. 58 The system should display the details of the complaint along with the history of the complaint. For this purpose, the system should maintain an audit trail. The audit trail will maintain all information, including reminders and alert transmission, irrespective of whether they were acknowledged or not.
57. 59 Complaint Closure The system should allow PHED to close the complaint with status RESOLVED or REJECTED (or NON RESOLVABLE, as defined above)
58. 60 The system should allow change of status to CLOSED only if the complaint is in its current state for more than the defined period. This implies that closed status is possible only after an interval of no communication from either side.
59. 61 The system should allow PHED grievance cell to close the complaint with status RESOLVED or REJECTED (or NON RESOLVABLE based on the feedback of the complainant. Cell shall make calls to the complainant and register the feedback in the system.
60. 2 On closure, complaint shall be archived.
61. 63 On closure, system should send notification to citizens and concerned officials through SMS/eMail.
62. 64 Reports to PHED The system should have the facility to generate the following reports:- 1. Division/Sub-Division wise classification of complaints as per category/sub-category wise 2. Classification of complaints as per the complaint status 3. Classification of complaints as per the service and functional area 4. Complaint Details 5. Average Resolving Rate 6. Average Response & Resolution Rate 7. Repeat Complainants, frequency of repeat complaints 8. Analysis of complaints, which get Auto Escalated 9. Generate ad-hoc query / reports based on the user defined 49 Sr. Module Requirement Description parameters 10. All reports should generated in html, excel and “pdf” formats 11. Reports should also be available in graphical format 12. Reports on benchmarking of divisions amongst each other are also desired 13. Trend analysis and other MIS etc. are also necessary, and will include action user wise analysis. 65 Reports for Divisions The system should have the facility to generate defined reports, which can be “pushed” or e-mailed to defined users. All reports will be available on the portal. Audit reporting and software usage MIS will also be available 66 Configuration Management The system should have a facility to set up the following masters: - 1. PHED Master 2.Division Master 3. District Master 4. Block master 5. Panchayat Master 6. Hands pump master 7. Scheme Master 8. Service Type Master 9. Complaint Type Master 10. User Master 11. Masters to define resolution types, and codes 67 Mandatory Requirements The system should allow integration of the Grievance Management Application with the portals/applications of PHEDs (existing or to be come-up within contract period) through API or other suitable technology. 68 The system should support bi-lingual content, necessarily through Unicode. The portal shall be available in 2 languages - English and Hindi. 69 Other functional requirements: Supports generation of pre-defined and query based MIS reporting. 70 Allows to track/maintain history of a complainant 71 Maintain list of classifications, TATs and disposal status. 72 Online checking , query handling and notification of status 73 Provide automated dispatch of complaints to all divisions 74 Generate report of statutory obligations details based on Insurers 75 Facility to Monitor Grievances processes end-to-end 76 Ability to search details by any parameters 77 Ability to generate analysis of Grievances etc. 78 Ability to auto e-mail & SMS the complaint details to the complainant & PHED 79 GMA will allow nodal officer to view grievance status, action 50 Sr. Module Requirement Description taken, resolution provided, closure status, feedback given, escalation status etc. regarding their particular divisions/subdivisions 80 Multiple updates against a complaints status should be possible to be made by the authorized users to track the complete development on the complaint redress system. 81 Ability to import/export data from/to the other external systems. 82 Separate administration module for managing the secured logins 4. Functional specification of Mobile Application Sr. Functional Description 1. Solution should allow only authorized user to access the application. 2. Solution should have simple and user friendly user-interface with rich MIS reporting. 3. Solution should pop-up a message whenever a complaint is registered where a particular user has to take action. 4. Solution should allow user to search the complaint details by providing unique complaint ID 5. Solution should allow user to update the status of complaint. 6. Solution should have a local database in the mobile so as it can provide MIS in offline mode also. 7. Solution should allow user to register/update action/close the complaint. 8. Solution should allow users to click and upload photographs or attach documents while updating the status.

Technical Details

Approach and Methodology

# Bridge Management Application

Initial Discussions

Summary of Discussions

# Pavement Management Application

Hardware & Software Requirements

System Design and Customisation Requirements

System Configuration

User Acceptance Testing

Deployment

Training and Technology Transfer

# Linear reference Management

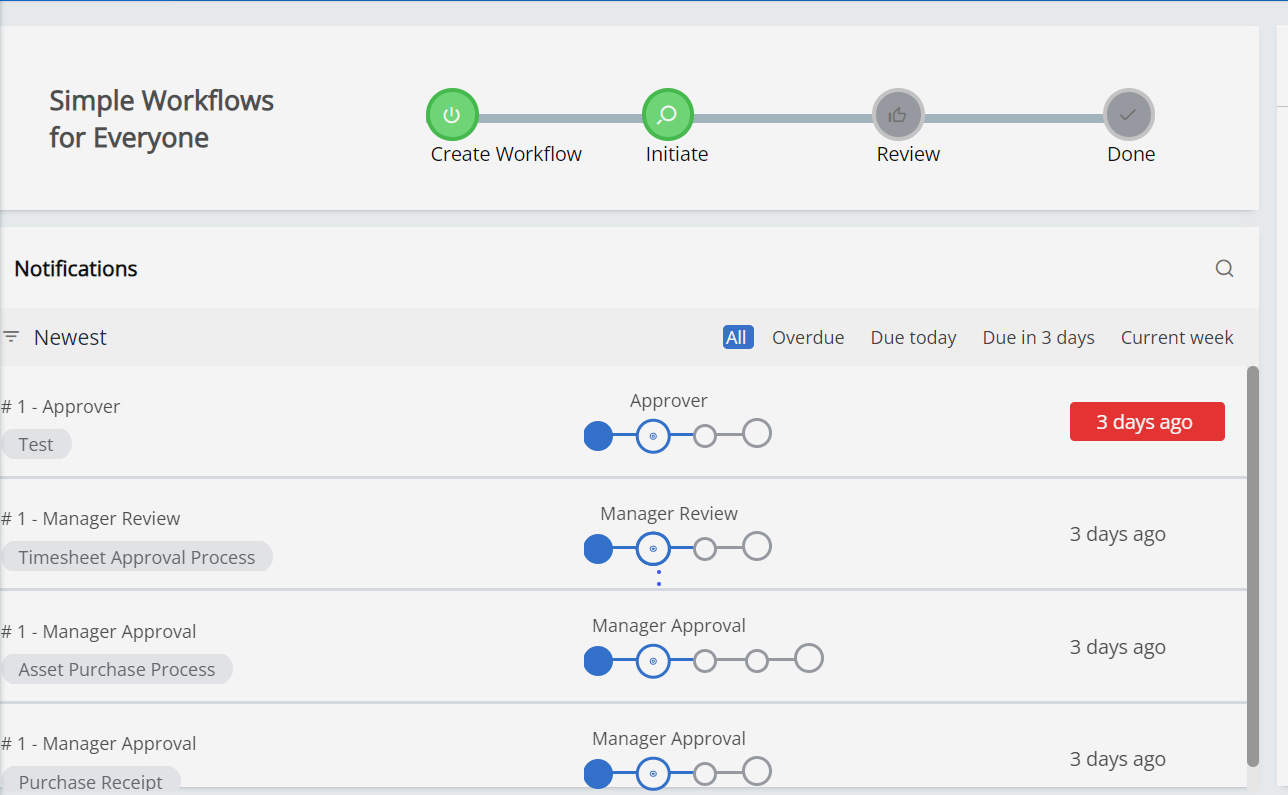
# Asset Valuation

# Workflow – reference Only

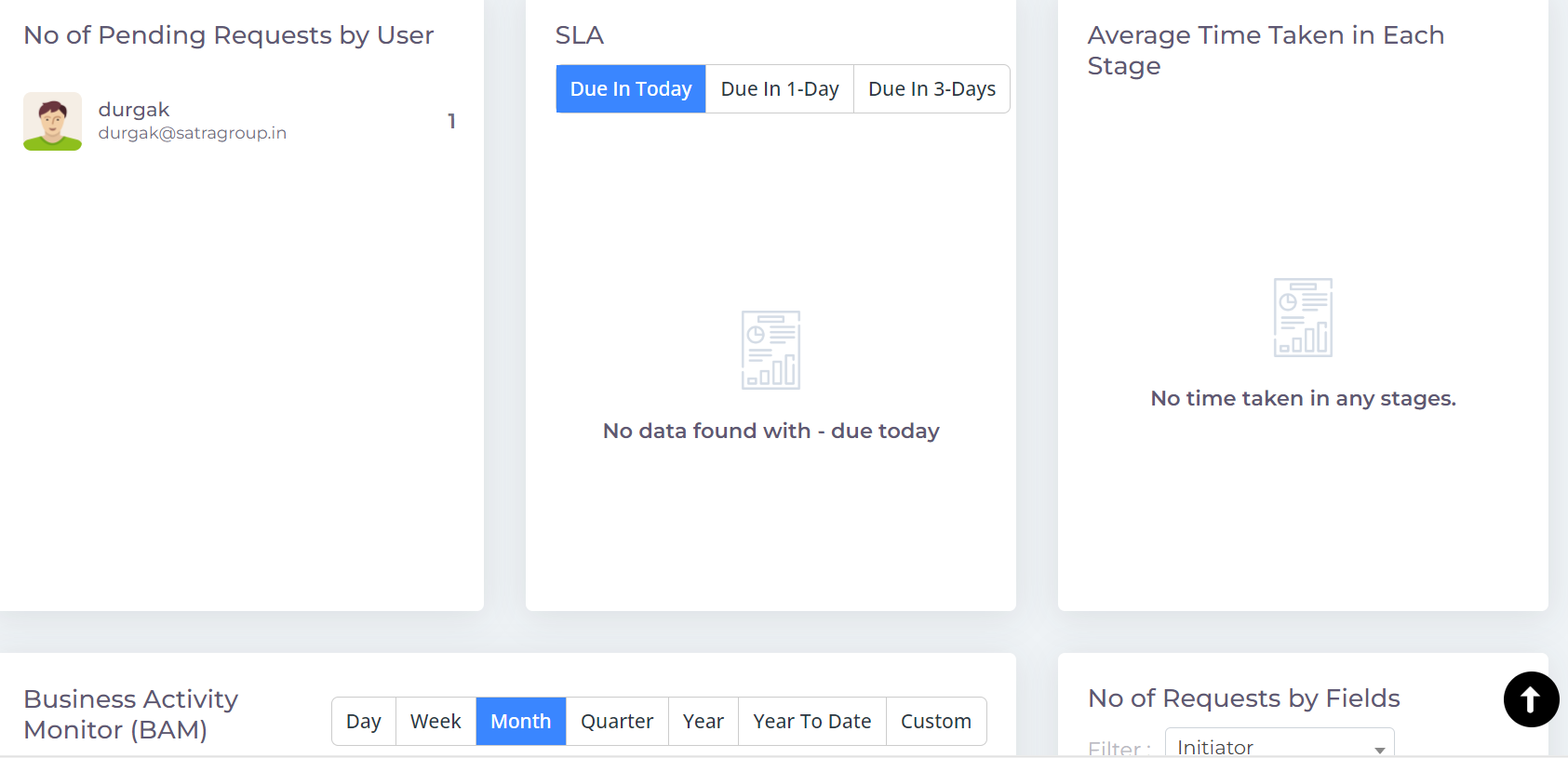
1. Initiate Workflow
2. Workflow dashboard

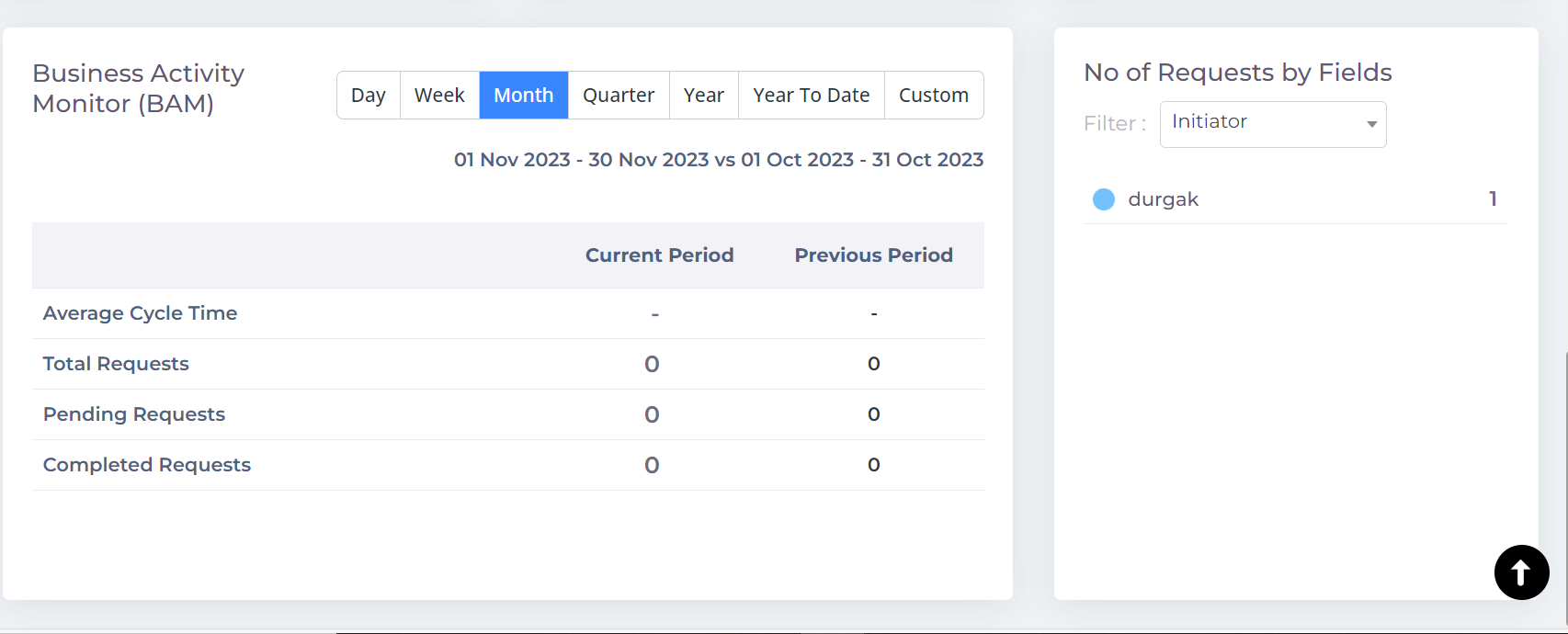
## Workflow dashboard

This is needed for administrator

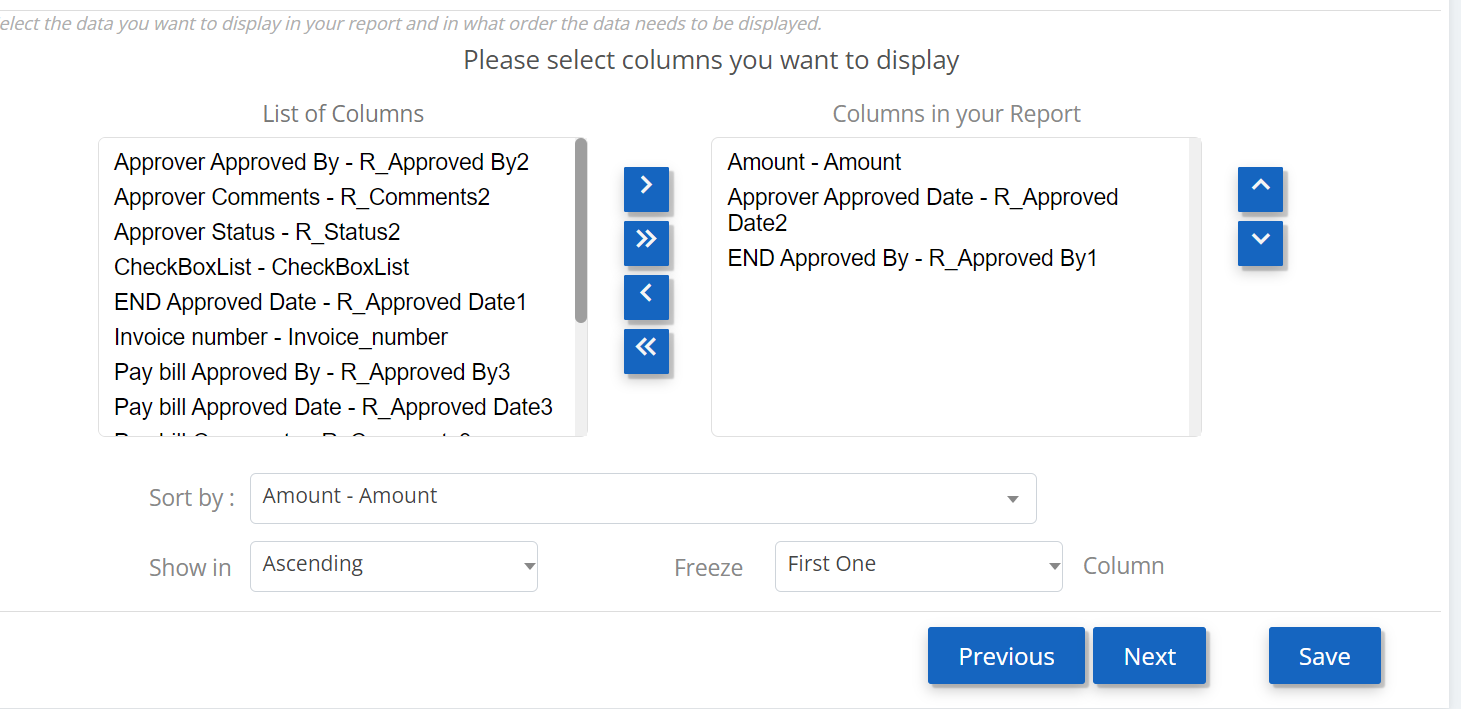




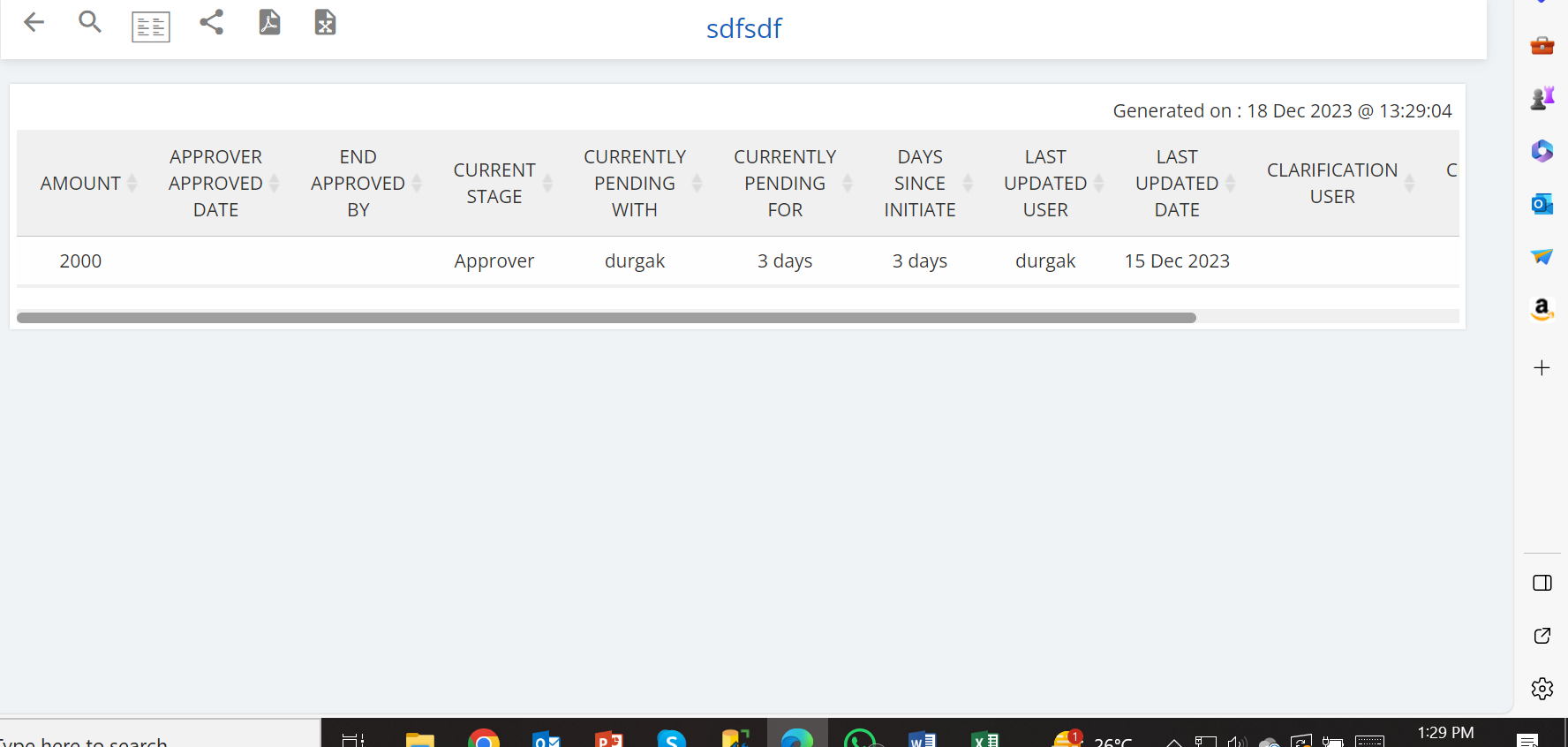


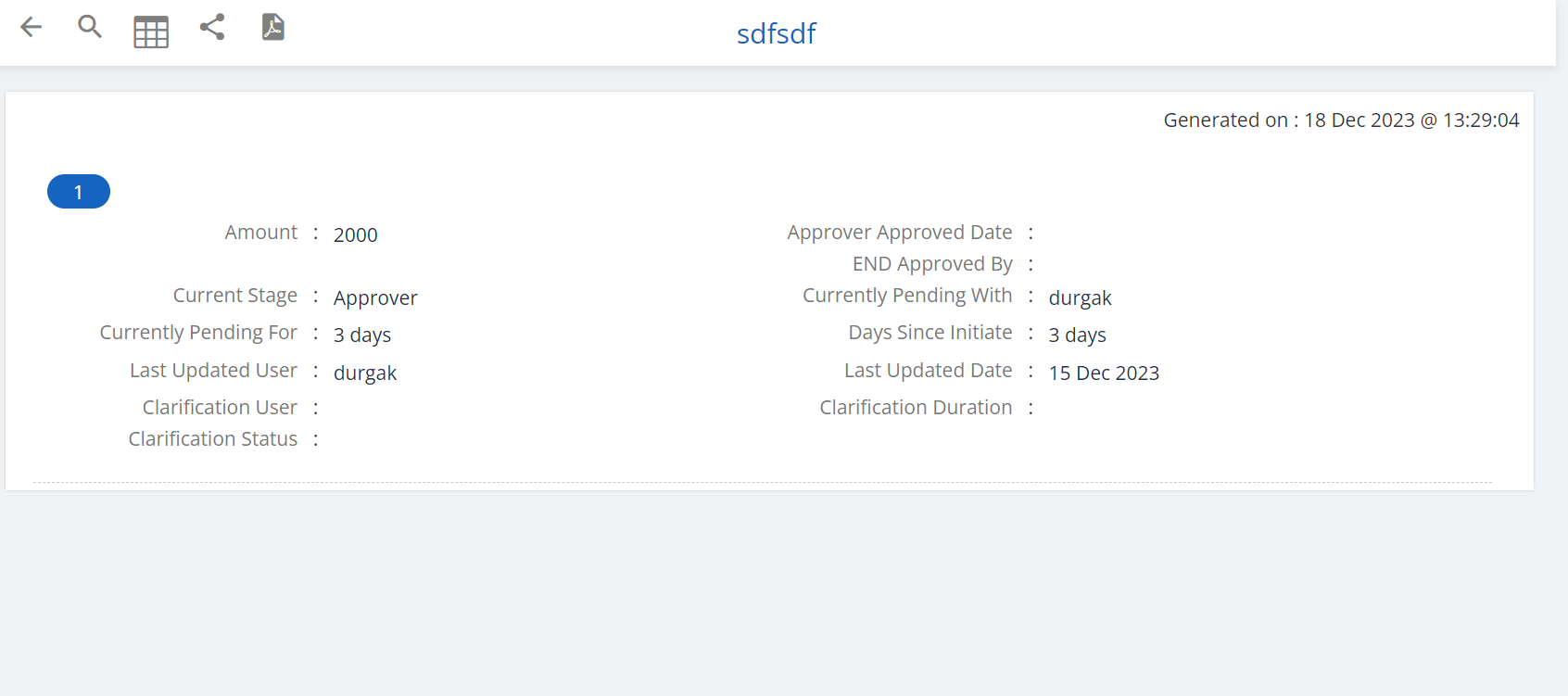


## Reports

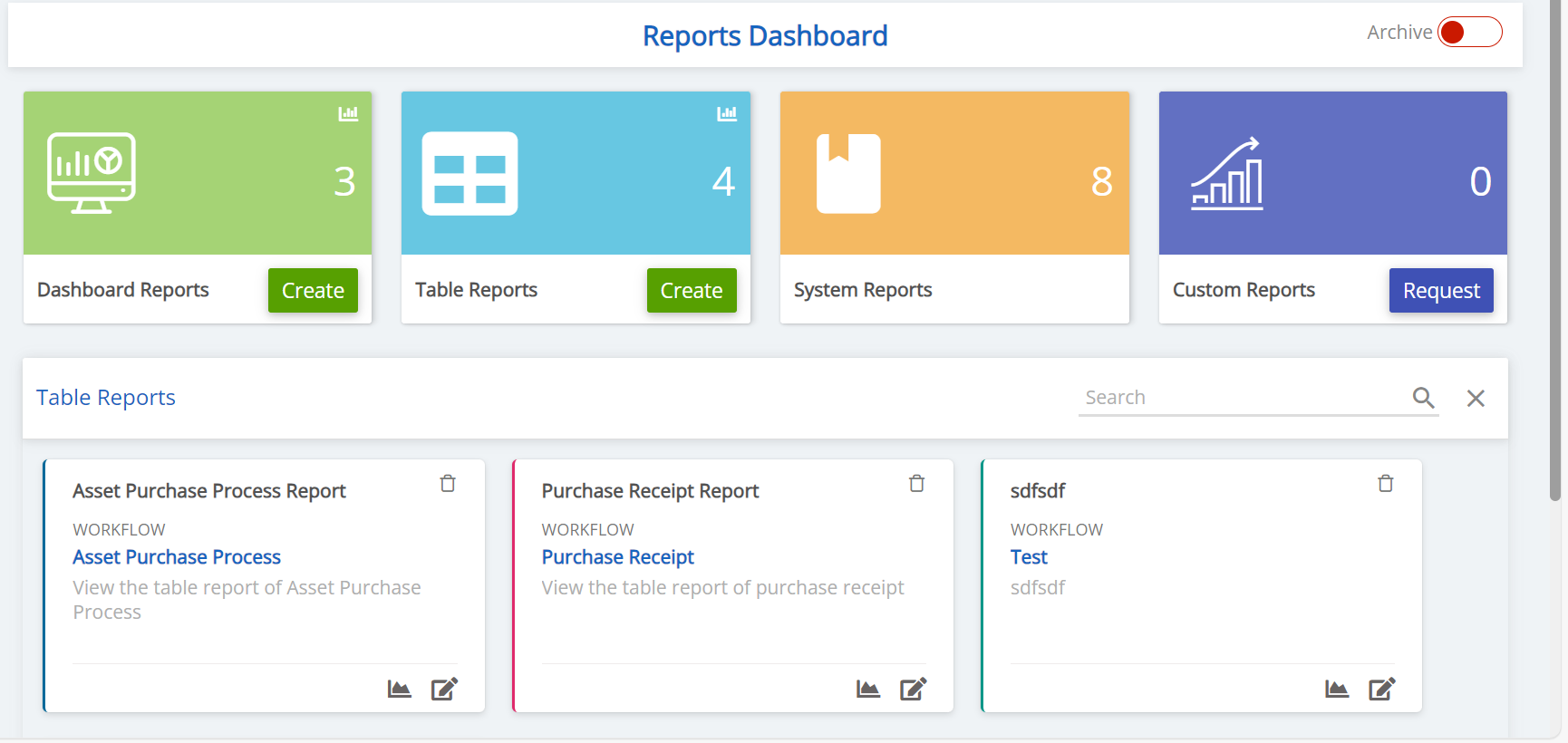


Freeze option to be explored.





Card View



# Key words

Following Keywords are recognized by the system and should not be used by the end user. System should have capability to identify these key words and through error if the user uses it. It would be good to give numbering to keywords as well so as to identify them easily.

|  |  |  |
| --- | --- | --- |
| KeyCode | KeyWord | Remarks |
| 10001 | Datayear |  |
|  | YearPortion |  |
|  | Geo1Param |  |
|  | Geo2Param |  |
|  | UserParam |  |
|  | DatayearParam |  |
|  | YearPortionParam |  |