

P4 Planning Doc - Tessitura Report Catalog

App Title

Tessitura Report Catalog

Description

Tessitura Report Catalog is a web application to manage report documentation for all in-house custom reports developed for Tessitura CRM application.

Tessitura (<http://tessituranetwork.com>) is the core business application we use in our company for online ticket selling, venue management, fundraising, customer relationship management and other ecommerce related operations. To meet our end users' demand we do lot of inhouse customizations, and created hundreds of custom reports and utilities. The one main challenge our end users and developers face is keeping track of all of these customizations. There are online documentation for all the canned reports and other standard Tessitura functionalities but nothing for the in-house custom solutions. Tessitura Report Catalog web application will meet this demand by creating an online repository for all the in-house reports which then we can hook up with the main Tessitura application so that a user can pull a report documentation on demand using context sensitive help from a Tessitura client application.

This will also help our in-house developers to find the relevant system documentation for troubleshooting and further enhancement purposes.

If there is enough time, the application may also have the features to view all favorite reports by a user; view and save a report detail page as PDF; view and save a searched report listing page as PDF; option to add, edit and view data dictionary/glossary page(s) which will help users & developers to use common lingo; a contact page, and a page to add and update system/reference data without relying on database interface.

Target Audience

The users and developers of Tessitura CRM application. The initial plan is to implement it in my current organization. If it becomes successful and after a couple of months of usage when everything will become streamlined I may share the application with other Tessitura licensee organizations to be locally installed and used if anyone is interested. Currently there are around 600 Tessitura licensees around the globe.

Essential Features

- Add a new report
- View list of reports with elaborated search/filtering options
- View a specific report detail
- Edit an existing report
- Add a report revision history by a developer
- Edit/delete a report revision history by a developer
- Add comments to a report; rate and mark a report as favorite by a user

- Edit/delete comments to a report, update rating and mark/unmark a report as favorite by a user

Non-essential Features

- View “My Favorite” reports by a user
- Add data dictionary/glossary for commonly used terms
- Edit data dictionary/glossary for commonly used terms
- Contact page with send message via email option
- View and save a report list as PDF
- View and save a report detail page as PDF
- Interface to add system or reference data
 - Report Framework (SSRS, InfoMaker, Other)
 - Code Location (SP, Embedded SQL, Other)
 - Report Category
 - User Group/ Roles

To do

[] Setup

- [] Create app
- [] Version control in Github
- [] Deploy to production (Digital Ocean) server
- [] Set up subdomain p4.guddi.ca

[] Homepage

- [] Welcome/intro note
- [] Logo/Company Name/Application Title
- [] Link to the “Listing page of reports with elaborated filtering options”
- [] Top level menu and submenu with link to other pages

[] Users

- [] Login
- [] Signup (eventually will be disabled and replaced by the “Add Users” screen)
- [] Page to add users (separately built to manage internal users)

[] Reports

- [] Page to add a new report
- [] Page to edit an existing report
- [] Listing page of reports with elaborated filtering options
- [] Page to add comments, rate a report and mark as favorite
- [] Page to edit comments, rate a report and mark as favorite
- [] Page to add revision history

☐ Page to edit revision history

☐ Finalize

☐ Test all features

☐ Observe someone else use the application

☐ Validate, spell check

☐ Update Github reports

☐ Final deployment on live site

Route Plan (Essential Features)

Purpose	Method	Route Name	Action	URI
Homepage	GET	index	index	/
Listing of reports	GET	reports.index	index	/reports/
Show a report detail	GET	reports.show	show	/reports/show/{reportId?}
Show form to add a new report	GET	reports.create	create	/reports/create
Process form to add the report	POST	reports.store	store	/reports/create
Show form to edit a report	GET	reports.edit	edit	/reports/{reportId?}/edit
Process form to edit the report	PUT	reports.update	update	/reports/{reportId?}/edit
Show form to edit a report by adding comments/delete comments	GET	comments.edit	edit	/comments/{reportId?}/edit
Process form to edit the report by adding comments	PUT	comments.update	update	/comments/{reportId?}/edit
Process form to delete a comment	DELETE	comments.destroy	destroy	/comments/{commentId?}
Show form to edit a report by adding	GET	revisions.edit	edit	/revisions/{reportId?}/edit

revision history				
Process form to edit the report by adding revision history	PUT	revisions.update	update	/revisions/{reportId?}/edit
Process form to delete a revision history	DELETE	revisions.destroy	destroy	/revisions/{revisionId?}

(Attention **Kate**: I broke the database object naming conventions suggested in the class here. Had a discussion in Piazza with **Susan** on this. Here is the Piazza discussion link: <https://piazza.com/class/iqiwxw3sex3r2?cid=155>. Thanks!

Update (Nov 29, 2016): I thought about it. Since it will be a completely separate DB from our business app, I changed my plan to stick to Laravel convention instead - at least for the duration of this course, so that I can take advantage of some Laravel built-in functionalities.)

Database Tables

Table name: reports

Description: Will store individual report doc

Fields:

- (int, primary key, auto_increment) **id**
- (varchar(255)) **name**
- (text) **description**
- (varchar(255)) **tess_report_id**
- (varchar(255)) **definition file**
- (text) **sql_proc**
- (varchar(255)) **database**
- (varchar(255)) **keywords**
- (text) **note_general**
- (text) **note_technical**
- (int) **category_id**
- (int) **framework_id**
- (int) **type_id**
- (boolean) **schedulable**
- (boolean) **inhouse**
- (boolean) **verified**
- (boolean) **published**
- (datetime) **first_implementetation_dt**
- (datetime) **last_update_dt**
- (boolean) **discontinued**
- (boolean) **active**
- (timestamp) **created_at**
- (timestamp) **updated_at**

- (int) **created_by**
- (int) **updated_by**

Table name: screenshots

Description: Will store screenshot file names for a report

Fields:

- (int, primary key, auto_increment) **id**
- (int) **report_id**
- (varchar(255)) **file_name**
- (varchar(255)) **file_type**
- (varchar(255)) **caption**
- (varchar(255)) **description**
- (boolean) **active**
- (timestamp) **created_at**
- (timestamp) **updated_at**

Table name: categories

Description: Will store report category names

Fields:

- (int, primary key, auto_increment) **id**
- (varchar(255)) **name**
- (varchar(255)) **description**
- (boolean) **default**
- (boolean) **active**
- (timestamp) **created_at**
- (timestamp) **updated_at**

Table name: frameworks

Description: Will store report frameworks

Fields:

- (int, primary key, auto_increment) **id**
- (varchar(255)) **name**
- (varchar(255)) **description**
- (boolean) **default**
- (boolean) **active**
- (timestamp) **created_at**
- (timestamp) **updated_at**

Table name: types

Description: Will store report types

Fields:

- (int, primary key, auto_increment) **id**
- (varchar(255)) **name**
- (varchar(255)) **description**
- (boolean) **default**
- (boolean) **active**
- (timestamp) **created_at**
- (timestamp) **updated_at**

Table name: tessareas

Description: Will store Tessitura area for reports

Fields:

- (int, primary key, auto_increment) **id**
- (varchar(255)) **name**
- (text) **description**
- (boolean) **default**
- (boolean) **active**
- (timestamp) **created_at**
- (timestamp) **updated_at**

Table name: report_tessarea

Description: Pivot table between reports and tessareas

Fields:

- (int, primary key, auto_increment) **id**
- (int) **report_id**
- (int) **tessarea_id**
- (boolean) **active**
- (timestamp) **created_at**
- (timestamp) **updated_at**

Table name: comments

Description: Table to store a user comments.

Fields:

- (int, primary key, auto_increment) **id**
- (int) **report_id**
- (int) **user_id**
- (varchar(255)) **description**
- (datetime) **comment_dt**
- (boolean) **active**
- (timestamp) **created_at**
- (timestamp) **updated_at**
- (timestamp) **deleted_at**

Table name: ratings

Description: Table to store a user comments.

Fields:

- (int, primary key, auto_increment) **id**
- (int) **report_id**
- (int) **user_id**
- (int) **rating**
- (boolean) **favorite**
- (boolean) **active**
- (timestamp) **created_at**
- (timestamp) **updated_at**
- (timestamp) **deleted_at**

Table name: revisions

Description: Table to store a revision history.

Fields:

- (int, primary key, auto_increment) **id**
- (int) **report_id**
- (int) **user_id**
- (varchar(255)) **description**
- (timestamp) **revision_dt**
- (boolean) **active**
- (timestamp) **created_at**
- (timestamp) **updated_at**
- (timestamp) **deleted_at**

Table name: glossaries

Description: Will store data dictionary/glossary items.

Fields:

- (int, primary key, auto_increment) **id**
- (varchar(255)) **term**
- (text) **definition**
- (boolean) **active**
- (timestamp) **created_at**
- (timestamp) **updated_at**
- (int) **created_by**
- (int) **updated_by**

Table name: users (Laravel built-in)

Description: Table to store users

Fields:

- (int, primary key, auto_increment) **id**
- (varchar(255)) **name**
- (varchar(255)) **email**
- (varchar(255)) **password**
- (varchar(100)) **remember_token**
- (timestamp) **created_at**
- (timestamp) **updated_at**

Table name: password_resets (Laravel built-in)

Description: Table to store users

Fields:

- (varchar(255)) **email**
- (varchar(255)) **token**
- (timestamp) **created_at**

Table name: roles

Description: Table to store application roles.

Fields:

- (int, primary key, auto_increment) **id**

- (varchar(255)) **name**
- (varchar(255)) **description**
- (boolean) **default**
- (boolean) **active**
- (timestamp) **created_at**
- (timestamp) **updated_at**

Table name: role_user

Description: Pivot table between the roles table and the users table.

Fields:

- (int, primary key, auto_increment) **id**
- (int) **role_id**
- (int) **user_id**
- (boolean) **active**
- (timestamp) **created_at**
- (timestamp) **updated_at**

Table name: appobjects

Description: Will store object details for this web application

Fields:

- (int, primary key, auto_increment) **id**
- (varchar(255)) **name**
- (text) **description**
- (boolean) **default**
- (boolean) **active**
- (timestamp) **created_at**
- (timestamp) **updated_at**

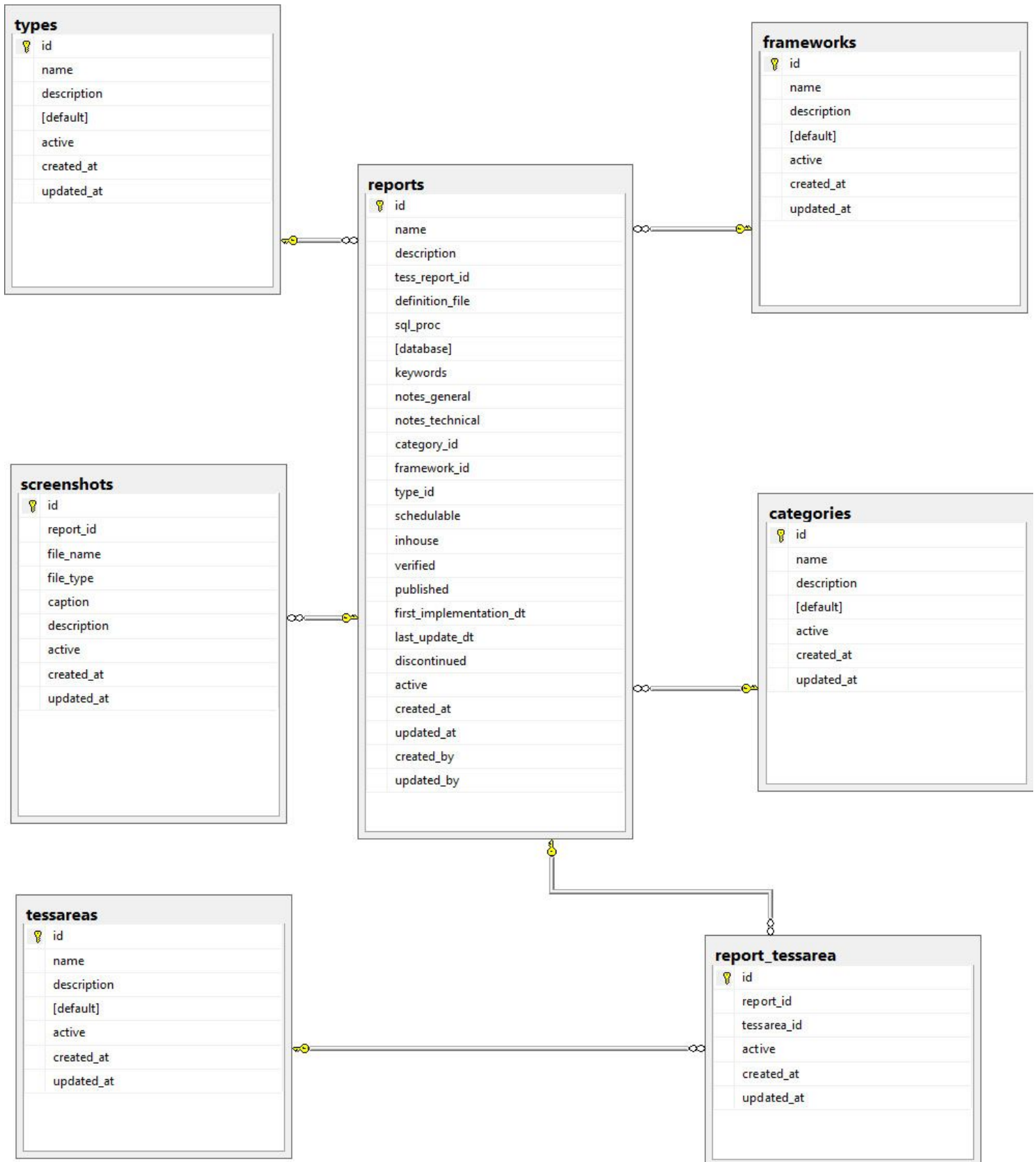
Table name: appobject_role

Description: Pivot table between appobjects and roles for authorization purpose

Fields:

- (int, primary key, auto_increment) **id**
- (int) **appobject_id**
- (int) **role_id**
- (boolean) **create**
- (boolean) **read**
- (boolean) **update**
- (boolean) **delete**
- (timestamp) **created_at**
- (timestamp) **updated_at**

Tess Reports - Core Attributes



glossaries

id
term
definition
active
created_at
updated_at
created_by
updated_by

Tess Reports - Glossaries, Revisions and User Feedback

ratings

id
report_id
user_id
rating
favorite
active
created_at
updated_at
deleted_at

comments

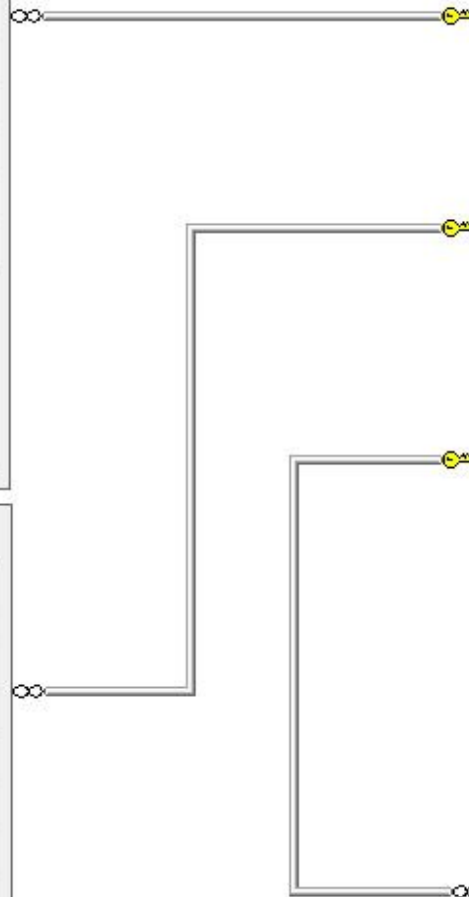
id
report_id
user_id
description
comment_dt
active
created_at
updated_at
deleted_at

reports

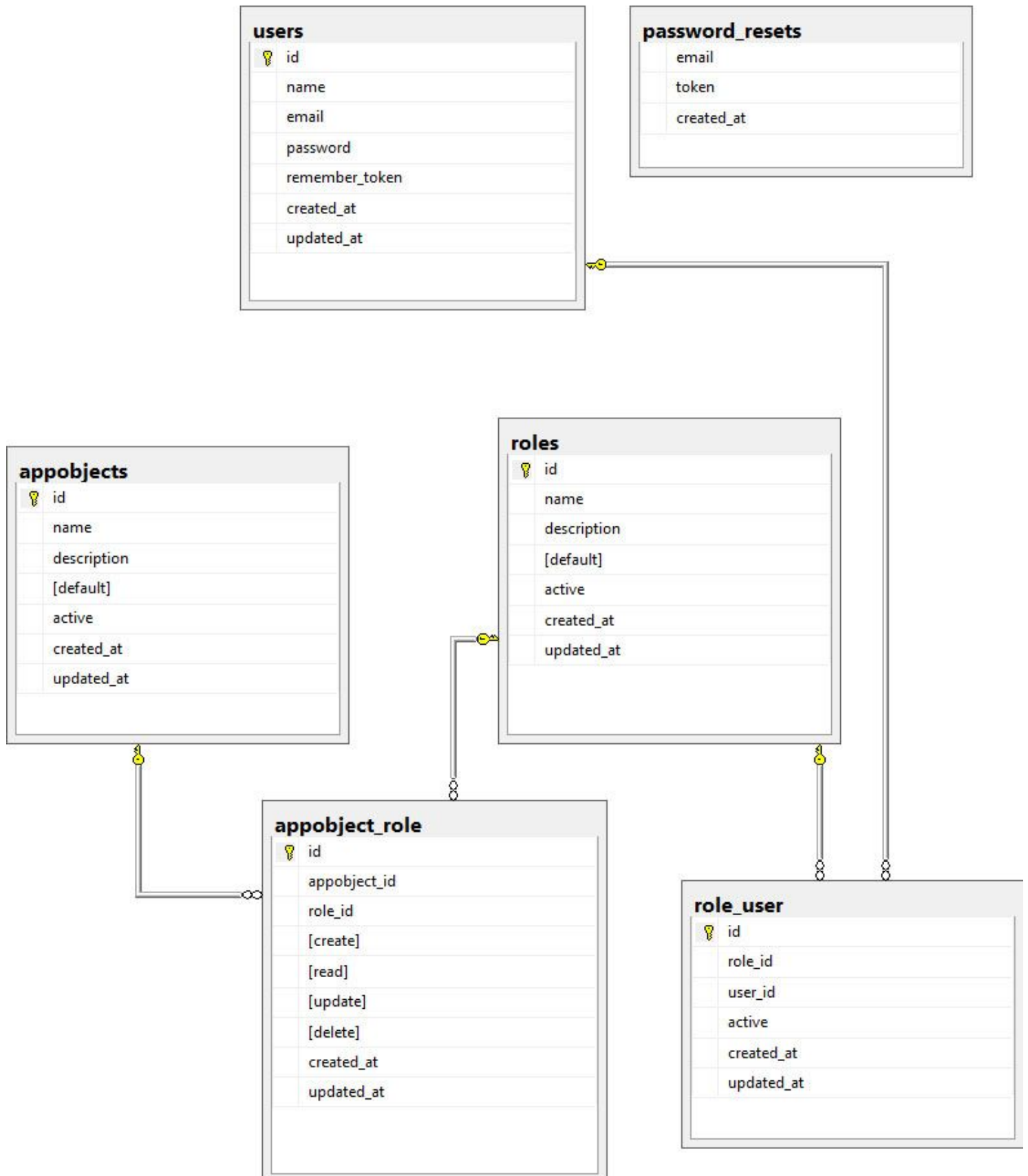
id
name
description
tess_report_id
definition_file
sql_proc
[database]
keywords
notes_general
notes_technical
category_id
framework_id
type_id
schedulable
inhouse
verified
published
first_implementation_dt
last_update_dt
discontinued
active
created_at
updated_at
created_by
updated_by

revisions

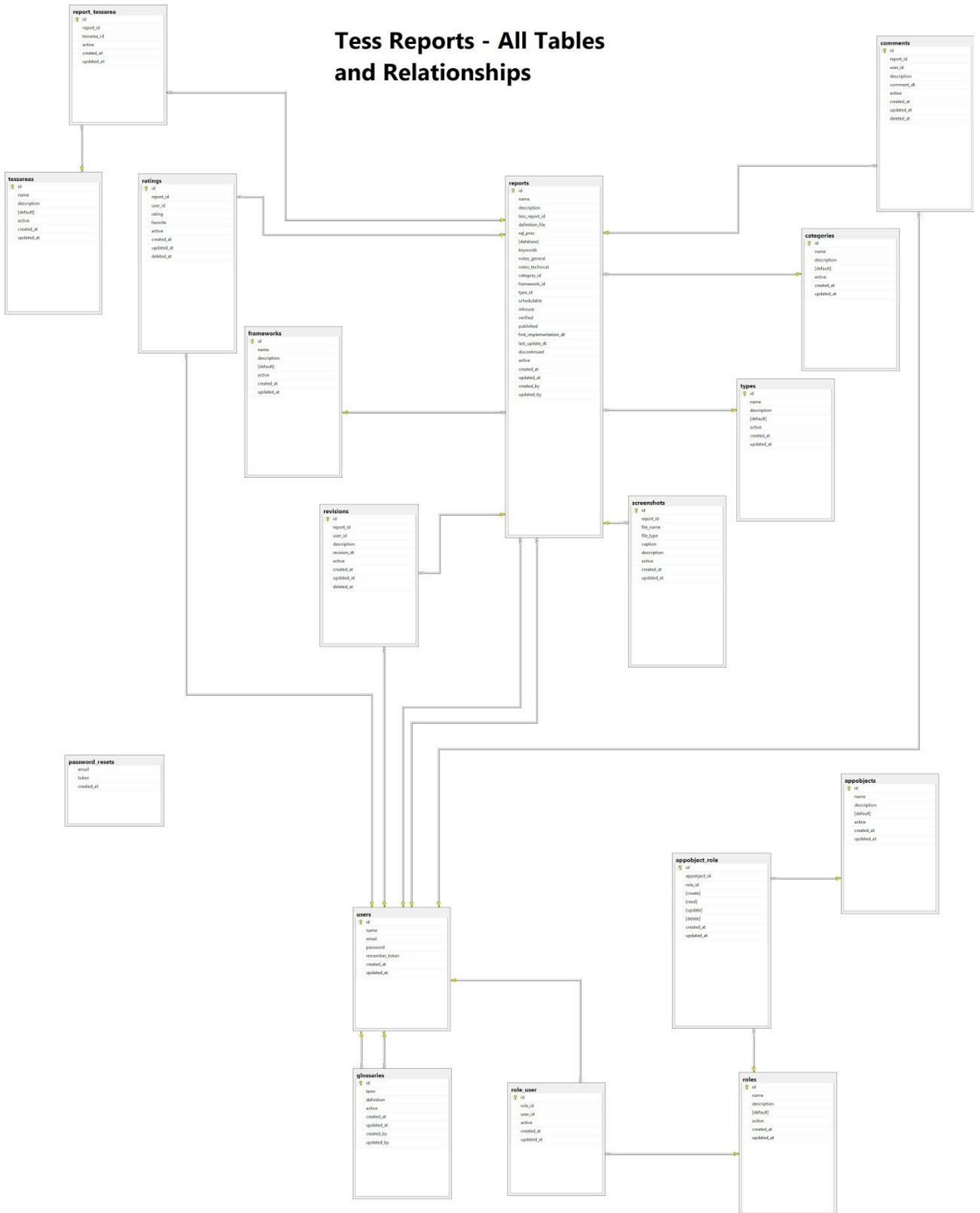
id
report_id
user_id
description
revision_dt
active
created_at
updated_at
deleted_at



Tess Reports - Security, Authentication and Authorization



Tess Reports - All Tables and Relationships



Misc

1. General viewing of a report will not be password protected to accommodate linking from the core business application (yet to finalize this decision). All other functionalities will be password protected.
2. All system/reference data are quite static and hardly will require an up-date. Even if they ever get updated the frequency will be very low - may be once in a couple of years. This update will be done by the developers. So no front end entry/update screen is necessary at this point. That's why I added them as non-essential features.
3. Though for the class project purpose I will keep a signup screen for the users eventually there will be only login screen for the users but no sign-up screen as the application won't remain open to public. Only designated users will have access to this application. The users will be added by the admin staff.
4. Screenshot file names in the screenshots table will be saved with file extension but without any folder location. All screenshots will be pulled from a particular folder of the application.
5. Keywords to easily find a report will be saved as a comma separated value string in the reports table for simplicity purpose instead of putting them in a separate table.
6. Since Laravel built functionalities don't support role based security, I extended this part to add my own role based security. Look at the "Tess Reports - Security, Authentication and Authorization" diagram above for details.