



**Quality.
Timeliness.
Customer Service.**

A Leidos Company

PROJECT: Dashboard

TECHNICAL SPECIFICATIONS DOCUMENT

Created: 09/23/2022

Document Revision History

| Release No. | Date | Revision Description/ Requirement # | Modified By | Approved By | Approval Date |
|-------------|------------|--|--|-------------|---------------|
| V1.0 | 09/23/2022 | Initial TSD - | Francisco Guzman | | |
| V2.0 | 09/29/2022 | Added all of the information and diagrams of our dashboard | Bryan, Alvent, Adrian, Ashley, Karina | | |
| v3.0 | 10/06/2022 | Modified diagrams and details | Bryan, Alvent, Adrian, Ashley, Karina, Pokuong Lao | | |
| v4.0 | 10/20/2022 | Fixed database, ER diagrams, and mockups | Bryan, Alvent, Adrian, Ashley, Karina, Pokuong Lao, James, Razin | | |
| v.5.0 | 10/27/2022 | Added mockups and revised everything | Bryan, Alvent, Adrian, Ashley, Karina, Pokuong Lao | | |
| v.6.0 | 11/02/2022 | Fixed intro section, updated mockups, finalized report | Bryan, Alvent | | |

Table of Contents

| | |
|---------------------------------------|-----------|
| GENERAL INFORMATION | 3 |
| <i>Project Objective</i> | 3 |
| Overview | 3 |
| <i>System requirements (Tools)</i> | 4 |
| Tools Setup | 4 |
| SYSTEM DESIGN AND ARCHITECTURE | 5 |
| <i>Topology</i> | 5 |
| <i>Systems Architecture</i> | 6 |
| High level. | 6 |
| High-Level Sequence and Trigger Point | 6 |
| <i>Application Architecture</i> | 7 |
| High-level design | 7 |
| Error Types. | 7 |
| Interfaces. | 7 |
| Access Control. | 8 |
| Data Table Functionalities | 8 |
| Data Table Pagination | 8 |
| Sort by | 9 |
| Search | 9 |
| Error Filter | 10 |
| Current QTC Functionality | 10 |
| Design to Configure Services | 10 |
| Design to Generate Dashboard | 11 |
| <i>Technical Design</i> | 12 |
| INTEGRATION POINTS | 13 |
| Database | 13 |
| <i>Database Objects</i> | 13 |
| Core Tables. | 13 |
| Table Details. | 13 |
| <i>ER Diagram</i> | 14 |
| Appendix | |
| Mockup Design | 15 |
| Rendered Model and Design | 19 |
| References | 25 |
| Glossary | 25 |

1.0 GENERAL INFORMATION

1.1 PROJECT OBJECTIVE

The main objective of this project is to create a smart dashboard web application. Ideally, this dashboard aims to help consolidate errors for users and admins that have occurred in different systems. Additionally, service desk staff will be able to investigate user errors that are related to bad data easily. By knowing what the error is and why it occurred, they can then communicate with the user on how to fix their data input in a specific application. Lastly, they will also see system errors which they can forward to the developers so they can investigate and fix them quickly.

1.1.1 Overview

QTC is the largest provider of disability and occupational health examination services. Because of this, they have a wide array of applications and staff members that are entering data. With these applications and data input, it is inevitable that errors will occur. This smart dashboard will help consolidate those errors, especially in the case of a staff member entering bad data. By being able to see the errors easily, service desk staff can assist the people entering bad data to prevent it from happening again in the future. In regards to system errors, they are typically more critical as they may affect the whole infrastructure of an application. Being able to see these errors is necessary for the service desk to communicate with the developers. Some of our high-level objectives for this project are:

- Display both user and system errors separately
- Pull data from multiple databases
- Implement pagination of 200 records
- Integration with authorization module

1.1.2 Reason for building this custom solution:

We investigated other smart dashboard solutions that are popular on the market and that are catered towards businesses. Some of them are

1. Table
2. Yellowfin
3. Oracle Analytics Cloud
4. Domo

While all of these dashboards provide a feature to connect to a database to retrieve data and display it, they are not fully customizable, the user access control is very limited, and the credentials to databases would get stored under one centralized account corresponding to an email address which poses a major security risk.

The dashboards listed above require the database information to be stored within their system. If their system gets breached then the data is at risk of exposure. Having an in-house dashboard without storing any database credentials remotely is ideal for security reasons. Additionally, it will match the current look and feel of QTCs other web applications. This is not possible with the dashboards we listed.

1.2 SYSTEM REQUIREMENTS (TOOLS)

This smart dashboard will be developed using ASP.NET MVC, C#, Entity Framework using stored procedures, and SQL as the backend.

1.2.1 Tools Setup

1. SQL Server 2019
2. .NET 6.0
3. Oracle for enterprise printing integration
4. SQL Server for case management integration

2.0 SYSTEM DESIGN AND ARCHITECTURE

2.1.1 TOPOLOGY

Our system's architecture will follow the existing architecture that was set by the infrastructure team.

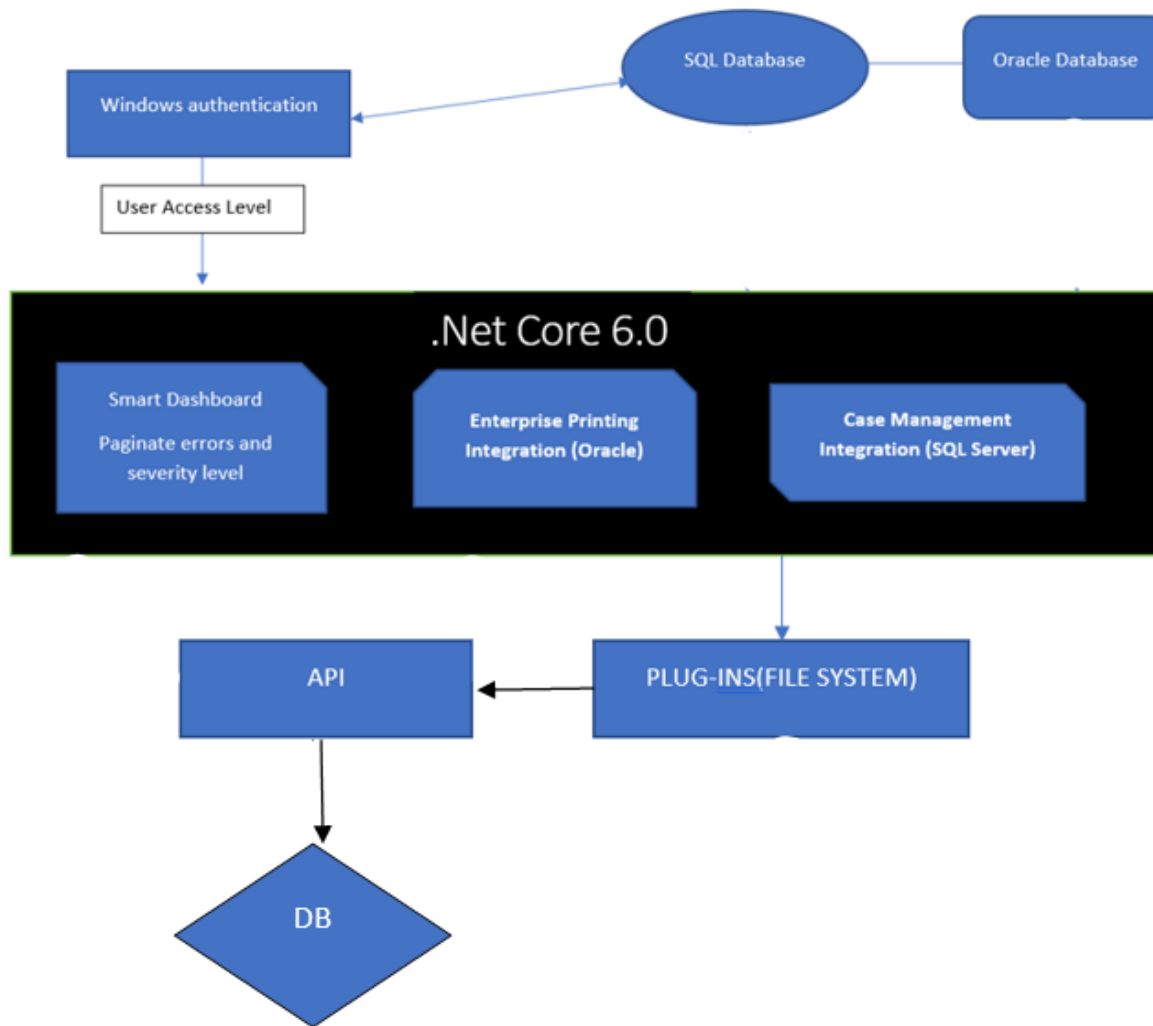


Figure 1: Topology

2.2 HIGH LEVEL.

Systems that will be interacting with the web application include Database and Assembly Plugins.

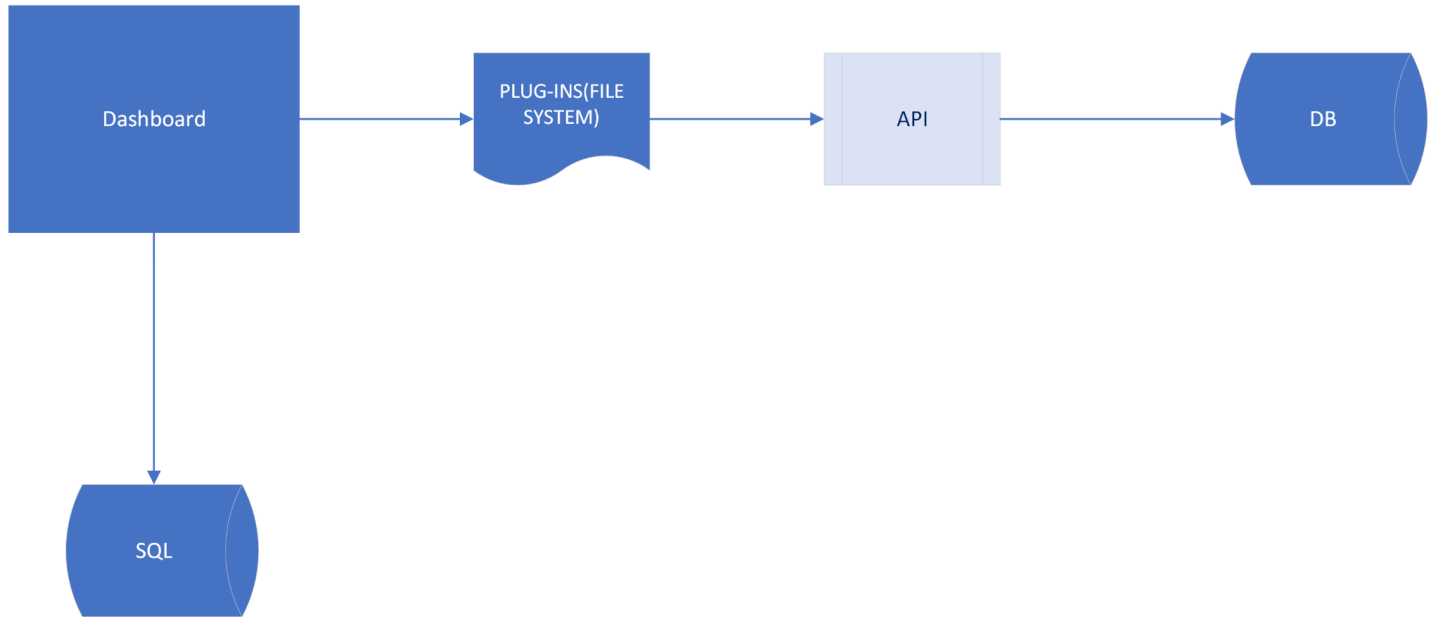


Figure 2: System Context

2.2.1 High-Level Sequence and Trigger Point

1. Upon start, the Dashboard will load all of the plugins
2. After the assemblies are loaded, the implementation for all of the tenants (Lines of business) will be injected.

More details will be determined for each module or application in their detailed design document

2.3 APPLICATION ARCHITECTURE

2.3.1 High-level design.

An application assembled interface that will contain all required classes and function declarations that will be used to read and pass information through a series of checkpoints. Authentication of injected objects will be verified at runtime. In addition, the collection will bring additional functionality to scan for errors. These objects will be referenced by the web application via dependency injection.

2.3.2 Error Types.

Our dashboard will display different types of errors based on the access the user has.

1. **User Errors:** This type of error will display an error made by the user either with incorrect data, nonexistent data, or access permission. Ex: The username and/or password does not match.
2. **System Errors:** This type of error will display to admin users if there were any network or server-side connection issues. Ex: A connection to the SQL database has failed. A network error has occurred while connecting to the server. Please verify your information is correct and SQL is configured for remote connections.
3. **Error Severity:** This type of error will display the severity for the user. The severity level will vary on the types of existing errors.

2.3.3 Interfaces.

Interfaces will be reusable which will contain core functionality and can also be implemented to trigger specific logic for a search. The following interfaces will be implemented:

- o **DashBoard:** This will allow the user to see all current data inserted and displayed to UIs. For Example: Based on user authentication, the records will be read, inserted, and displayed on the dashboard.
- o **Data Table:** This will allow the user to see all data entries that have been read and inserted from the database file. Allow filtering by searching and categorizing the table headers. Allows the user to control the number of results displayed. For Example, The user can enter a number and the table would start to reorganize itself from the smallest to the largest number. If the user only wanted to see 10 data entries on the page instead of 20 they would choose the number 10 in the dropdown list.
- o **Enterprise Printing Integration (Oracle):** This allows commercial tenant access and manages errors from the database.
- o **Case Management Integration (SQL Server):** This allows the user or admin to navigate and access data from the database with the following queue from the dashboard. The queue contains some required fields to allow access and searches. Users can acquire and manage errors via an API.

2.3.4 Access Control.

Access will be controlled by passing a parameter called user to type in the following way:

- o Access level administered via user type.
- o A user can have access to one or multiple applications.
- o Users that do not have access will not be able to view the application.
- o User access is based on the user type level.
 - User: basic access
 - Admin: administrative access

2.3.5 Data Table Functionalities

Sensitive/Classified Information

Bryan Gonzalez - Last Login 10/24/2022 10:43AM

VA - Application 1

Success! 200 more records have been pulled from the database.

Show 10 entries

☐ User Errors ☐ System Errors ☒ All

Search:

| Application Name | Layer | Module | Alert | Alert Team | Severity | Server Name | Error Code | Error Message | Error Date | User | View Record |
|------------------|---------|--------|-------|------------|----------|-------------|------------|--------------------|------------|-------|----------------------|
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | View |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | View |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | View |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | View |

Showing 1 to 10 of 200 entries

Previous 1 2 3 4 ... 20 Next

Pull 200 More Records

More Pages

Copyright © QTC Management, Inc. 2022

Figure 3: Data Table

2.3.5.1 Data Table Pagination

Data table will hold a default of 200 records that will be paginated by only showing 10 records per page. There will be pagination buttons at the top and bottom of the table that will allow a User/Admin to go back to a previous page, go to any of the visible page numbers displayed, or go to the next of the current page.

There will also be a button next to the pagination buttons, as shown in the bottom right of the table in [Figure 3](#) named “More Pages”, that will give the User/Admin the option to pull in a maximum of 200 more records. The additional records will be added to the default state the table was as. For example, if a User/Admin sorts the table by severity and decides to pull in an additional 200 records, the data table will reset back to its default state and will add the additional 200 to the end of the records that are already in the data table.

A User/Admin will have the option to change how many records are shown per page, this can be seen at the top left of the data table in [Figure 3](#); the default number is 10 per page. The number of button page numbers will change accordingly as well as text that displays the number of pages there are. The state of how many records are shown per page will not reset back to its default value if a user decides to pull 200 more records.

2.3.5.2 Sort by

Sorting by Error Date, from newest to oldest, will be the default display of the records. If a user decides to pull in 200 more records, the data table will reset back to its default sort.

User and Admin will be able to sort the data in the Data Table when clicking on one of the column headers below, and can be seen at the 1st row of the table in [Figure 3](#):

- | | |
|---------------------------------|-----------------------------------|
| ○ Application Name | ○ User (if available) |
| ○ Account Id (User only) | ○ Layer (Admin only) |
| ○ Severity | ○ Module (Admin only) |
| ○ Alert Team | ○ Alert (Admin only) |
| ○ Error Message | ○ Server Name (Admin only) |
| ○ Error Date | ○ Error Code (Admin only) |

Sort will look different for each column header. For “Error Date”, a user will be able to sort error dates by newest to oldest and vice versa. Alphabetically and vice versa for Application Name, Alert Team, User, Layer, Module, Alert, and Server Name. Numerically and vice versa for Account ID and Error Code. The severity will be sorted from lowest to highest and vice versa.

2.3.5.3 Search

A User/Admin will be able to search for record(s), at the top of the table as shown in [Figure 3](#), with any of the given values for all column headers mentioned in section 2.5.3.1; the format for the search input is comma-separated values to allow for a User/Admin to search for the specific record(s) in mind. The search will be applied to the records in the data table and also to the records in the database. For example, if a User/Admin is searching for a specific record, the data table will be searched first, if not there, then the database will be searched next. If the record(s) is found then it will be displayed to the user, if not a message will be shown that the record has not been found; as shown in [Figure 15](#) and [Figure 17](#).

2.3.5.4 Error Message

Depending on the size of the Error message, there will be a different UI for small-medium and large errors. For small-medium errors, the row of the record will simply expand, as shown in [Figure 10](#). For

larger errors, there will be a window that opens up that allows a User/Admin to see the message as shown in [Figure 9](#) and [Figure 18](#).

2.3.5.4 Error Filter

There will be an error filter above the data table, shown in [Figure 3](#), to allow for a User/Admin to filter the records based on the type of Error. There will be 3 options: User Errors, System Errors, and All. These options will be given as radio buttons and the default filter will be as option “All”. The records in the data table will be displayed based on the Error filters and will change accordingly. The error filter will reset back to its default value if a User/Admin decides to pull in an additional 200 records. Similar to section 2.3.5.2 sort by.

2.3.5.4 Help

Our Dashboard will also have a “Help” feature located in the sidebar, which can be seen in [Figure 14](#). This Help feature will provide an informational window to the User/Admin, shown in [Figure 16](#), to help a User/Admin understand our Dashboard. Currently our help window will show information on sorting, search filtering, pulling in more records, entries per page, and error types.

2.3.7 Current QTC Functionality

As of now QTC has no centralized location to display errors corresponding to bad user data input and system errors; for service desk staff to view and act on. This dashboard design will be the centralized place to display these errors providing a more efficient design instead of having two places to go for these errors.

2.3.8 Design to Configure Services

We will use the information that is being retrieved by the Authentication Module and search within our SQL Server Database to retrieve that user's saved information. That information stores the user, user type, and the databases they have access to, which then loads onto the smart dashboard.

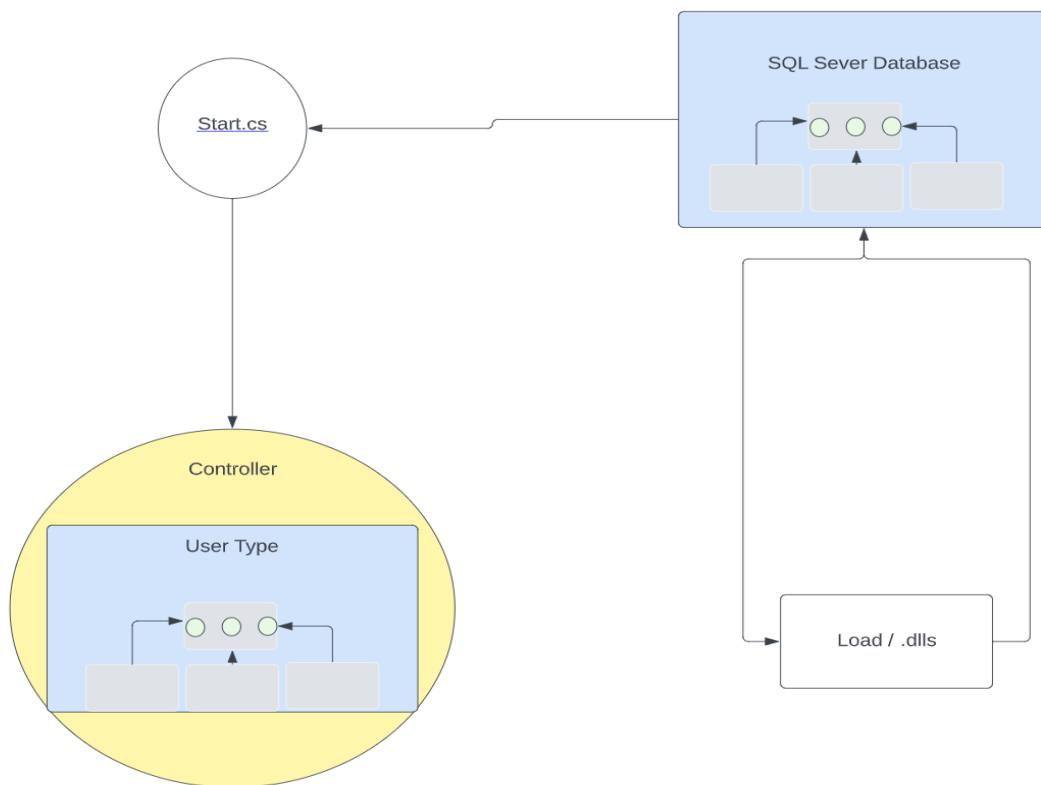


Figure 4: Design to Configure Services

2.3.9 Design to Generate Dashboard

Prior to generating the dashboard, the tenant's user name is passed to an API which makes a call to retrieve correlating configuration records. The user's information is passed to the controller which determines what access level they have based on their user type. Based on the user's access type they can view the dashboard with corresponding integration points.

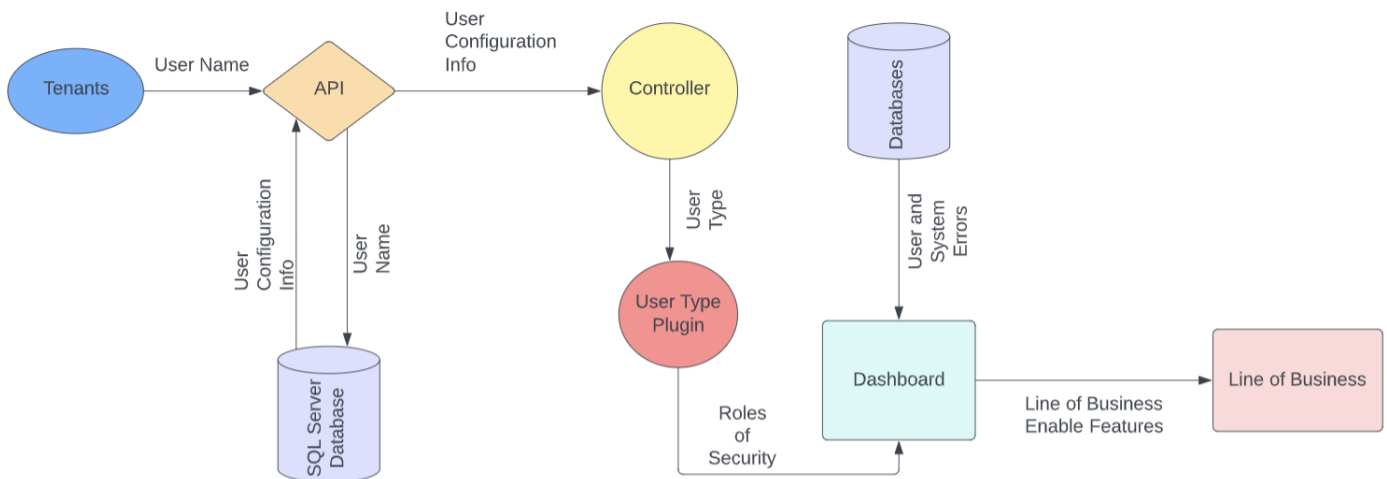


Figure 5: Design to Generate Dashboard

2.4 TECHNICAL DESIGN

Link to diagram:

[Figure 1: Topology](#)

[Figure 2: System Context](#)

[Figure 3: Data Table](#)

[Figure 4: Design to Configure Services](#)

[Figure 5: Design to Generate Dashboard](#)

[Figure 6: ER Diagram](#)

[Figure 7: Admin Interface - user interface via the dashboard](#)

[Figure 8: User Interface - user interface via the dashboard](#)

[Figure 9: Large Error - display a detailed and extensive error message](#)

[Figure 10: Small to Medium Error - display preview of error message](#)

[Figure 11: Closed Side Bar - minimalize sidebar](#)

[Figure 12: Dashboard - display paginated records \(admin level\)](#)

[Figure 14: Dashboard - display paginated records \(user level\)](#)

[Figure 13: Sidebar - collapsed sidebar](#)

[Figure 15: User Experience Design - display a message to consolidate user experience](#)

[Figure 16: Help Window - display functionality to accommodate the user experience](#)

[Figure 17: Additional Records - display a message to consolidate user experience](#)

[Figure 18: View Record- display record detail](#)

3.0 INTEGRATION POINTS

The dashboard will leverage multiple databases based on the user's records. Each integration point will be displayed as tabs with corresponding data.

4.0 DATABASE

The dashboard will pull from a SQL database that holds the user's records and configures them for display.

4.1 DATABASE OBJECTS

4.1.1 Core Tables.

TABLE [DASHBOARD].[Active]
TABLE [DASHBOARD].[Address1]
TABLE [DASHBOARD].[Address2]
TABLE [DASHBOARD].[ApplicationId]
TABLE [DASHBOARD].[ApplicationName]
TABLE [DASHBOARD].[City]
TABLE [DASHBOARD].[Country]
TABLE [DASHBOARD].[CreatedBy]
TABLE [DASHBOARD].[CreateDate]
TABLE [DASHBOARD].[Description]
TABLE [DASHBOARD].[IntegrationPoint]
TABLE [DASHBOARD].[LOBId]
TABLE [DASHBOARD].[Name]
TABLE [DASHBOARD].[State]
TABLE [DASHBOARD].[UpdatedBy]
TABLE [DASHBOARD].[UpdatedDate]
TABLE [DASHBOARD].[UpdatedDate]
TABLE [DASHBOARD].[UserId]
TABLE [DASHBOARD].[UserType]

4.1.2 Table Details.

- Table contains many different entities such as Organization, Application User Type, and Application all listed at the top grayed portion of each entity rectangle in the diagram below.
- Primary keys for entities are underlined, all primary keys will have "id" attached to the name
- In terms of cardinality our entities will have many to many cardinalities in the diagram below

4.2 ER DIAGRAM

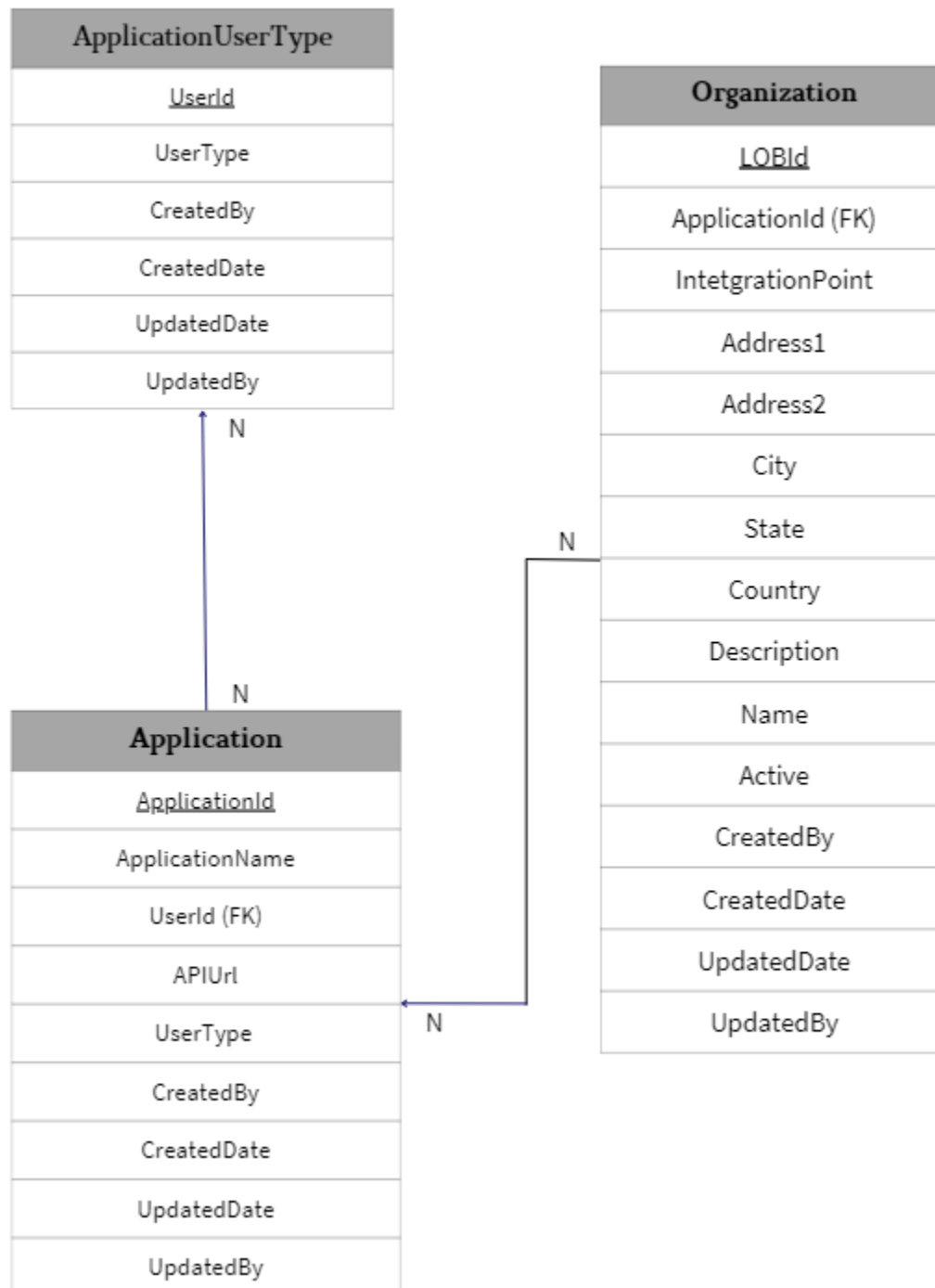


Figure 6: ER Diagram

5.0 APPENDIX

5.1 MOCKUP DESIGN

Link to Figma: [QTC MockUp – Figma](#)

5.1.1 Admin interface

Admin Interface

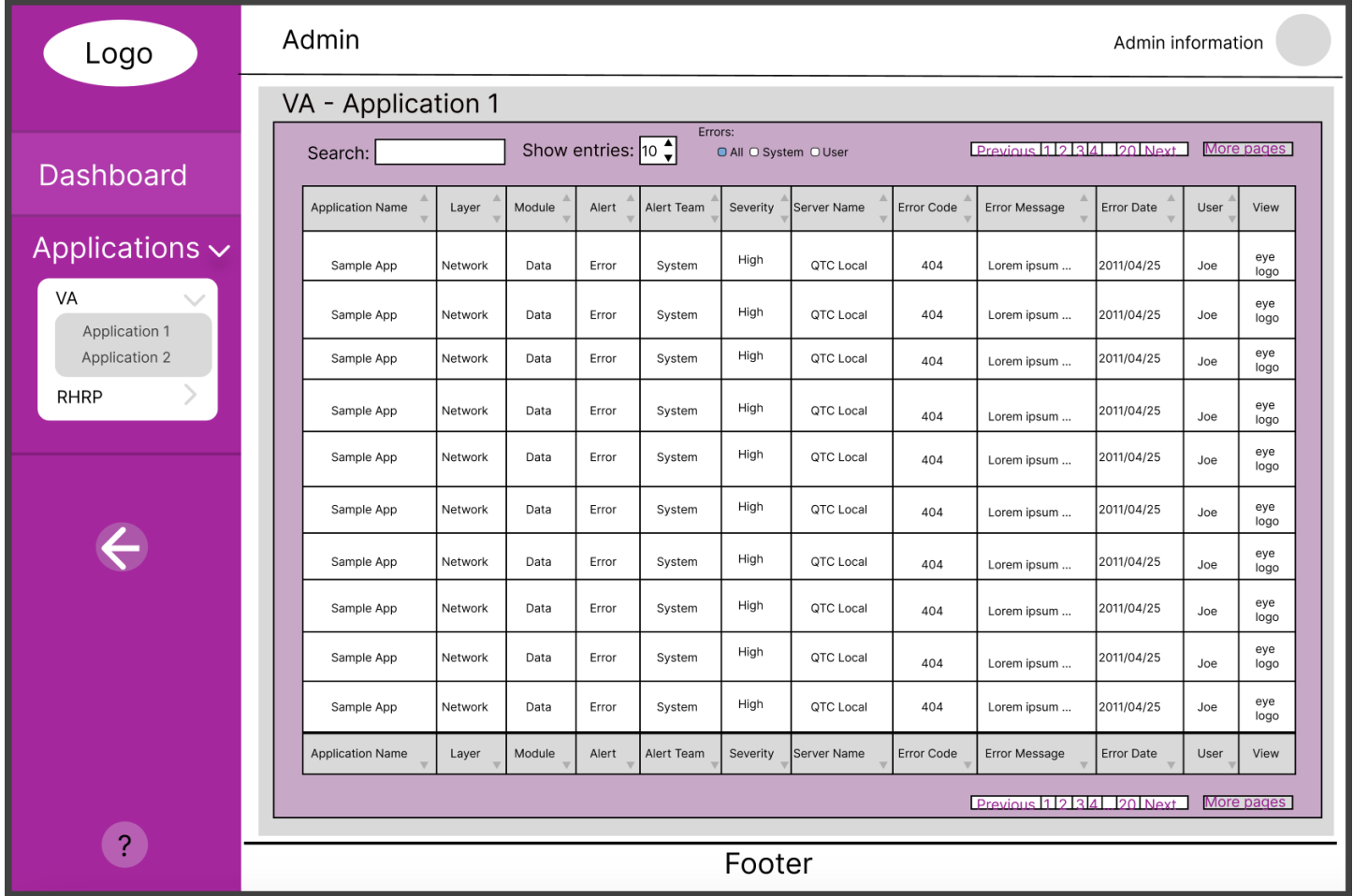


Figure 7: Admin Interface - user interface via the dashboard

5.1.2 User Interface

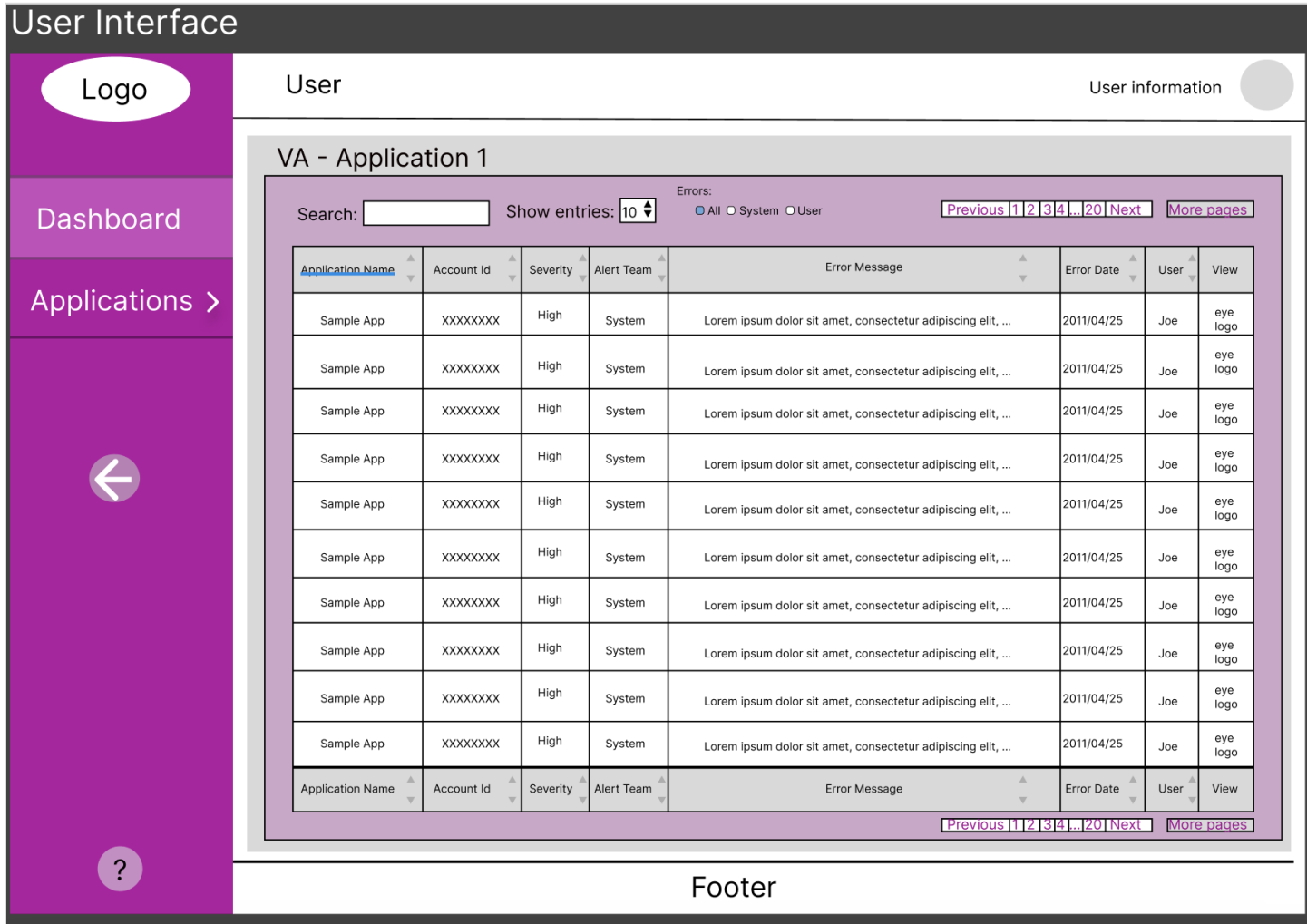


Figure 8: User Interface - user interface via the dashboard

5.1.3 Error Message Window

5.1.3.1 Large Error

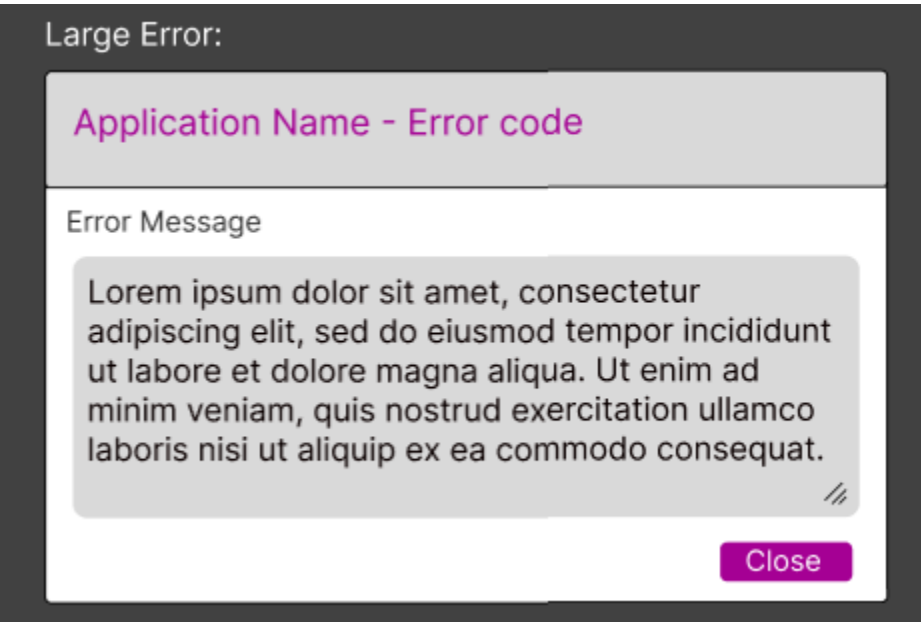


Figure 9: Large Error - display detailed and extensive error message

5.1.3.2 Small - Medium Error

Small - Medium Error:

| Application Name | Layer | Module | Alert | Alert Team | Severity | Server Name | Error Code | Error Message | Error Date | User | View |
|------------------|---------|--------|-------|------------|----------|-------------|------------|-----------------|------------|------|----------|
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Lorem ipsum ... | 2011/04/25 | Joe | eye logo |

| Application Name | Layer | Module | Alert | Alert Team | Severity | Server Name | Error Code | Error Message | Error Date | User | View |
|------------------|---------|--------|-------|------------|----------|-------------|------------|---|------------|------|----------|
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. | 2011/04/25 | Joe | eye logo |

Figure 10: Small to Medium Error - display preview of error message

5.1.4 Sidebar

Logo

User

User information

Dashboard

Applications

VA

RHFP

Application 1

Application 2

VA - Application 1

Search:

Show entries: 10

Errors: ☐ All ☐ System ☐ User

Previous 1 2 3 4 ... 20 Next

More pages

| Application Name | Account Id | Severity | Alert Team | Error Message | Error Date | User | View |
|------------------|------------|----------|------------|--|------------|------|----------|
| Sample App | XXXXXXXX | High | System | Lorem ipsum dolor sit amet, consectetur adipiscing elit, ... | 2011/04/25 | Joe | eye logo |
| Sample App | XXXXXXXX | High | System | Lorem ipsum dolor sit amet, consectetur adipiscing elit, ... | 2011/04/25 | Joe | eye logo |
| Sample App | XXXXXXXX | High | System | Lorem ipsum dolor sit amet, consectetur adipiscing elit, ... | 2011/04/25 | Joe | eye logo |
| Sample App | XXXXXXXX | High | System | Lorem ipsum dolor sit amet, consectetur adipiscing elit, ... | 2011/04/25 | Joe | eye logo |
| Sample App | XXXXXXXX | High | System | Lorem ipsum dolor sit amet, consectetur adipiscing elit, ... | 2011/04/25 | Joe | eye logo |
| Sample App | XXXXXXXX | High | System | Lorem ipsum dolor sit amet, consectetur adipiscing elit, ... | 2011/04/25 | Joe | eye logo |
| Sample App | XXXXXXXX | High | System | Lorem ipsum dolor sit amet, consectetur adipiscing elit, ... | 2011/04/25 | Joe | eye logo |
| Sample App | XXXXXXXX | High | System | Lorem ipsum dolor sit amet, consectetur adipiscing elit, ... | 2011/04/25 | Joe | eye logo |
| Sample App | XXXXXXXX | High | System | Lorem ipsum dolor sit amet, consectetur adipiscing elit, ... | 2011/04/25 | Joe | eye logo |
| Sample App | XXXXXXXX | High | System | Lorem ipsum dolor sit amet, consectetur adipiscing elit, ... | 2011/04/25 | Joe | eye logo |

Previous 1 2 3 4 ... 20 Next

More pages

?

Footer

Figure 11: Closed Side Bar - minimize sidebar

5.2 RENDERED MODEL AND DESIGN

5.2.1 Dashboard With Admin Access

Dashboard

VA

EPFS

Case Management

EProcess

RHRP

?

Sensitive/Classified Information

Bryan Gonzalez - Last Login 10/24/2022 10:43AM

VA - Application 1

Show 10 entries

☐ User Errors ☐ System Errors ☒ All

Search:

| Application Name | Layer | Module | Alert | Alert Team | Severity | Server Name | Error Code | Error Message | Error Date | User | View Record |
|------------------|---------|--------|-------|------------|----------|-------------|------------|--------------------|------------|-------|-------------|
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |

Showing 1 to 10 of 200 entries

Previous

1

2

3

4

...

20

Next

Pull 200 More Records

More Pages

Copyright © QTC Management, Inc. 2022

Dashboard

VA

RHRP

Intake

Compass

?

Sensitive/Classified Information

Bryan Gonzalez - Last Login 10/24/2022 10:43AM

RHRP - Application 1

Show 10 entries

☐ User Errors ☐ System Errors ☒ All

Search:

| Application Name | Layer | Module | Alert | Alert Team | Severity | Server Name | Error Code | Error Message | Error Date | User | View Record |
|------------------|---------|--------|-------|------------|----------|-------------|------------|--------------------|------------|-------|-------------|
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |

Showing 1 to 10 of 200 entries

Previous

1

2

3

4

...

20

Next

Pull 200 More Records

More Pages

Copyright © QTC Management, Inc. 2022

Figure 12: Dashboard - display paginated records (Admin level)

5.2.2 Dashboard with User Access

QTC

Dashboard

VA

EPFS

Case Management

EProcess

RHRP

Sensitive/Classified Information

Alvent Chang - Last Login 10/22/2022 10:40AM

User Queue

Show 10 entries

| Application Name | Account Id | Severity | Alert Team | Error Message | Error Date | User | Action |
|------------------|------------|----------|------------|--------------------|------------|-------|--------|
| Sample App | 4103 | High | Staff | Resource not found | 2011/04/25 | User1 | |
| Sample App | 4103 | High | Staff | Resource not found | 2011/04/25 | User1 | |
| Sample App | 4103 | High | Staff | Resource not found | 2011/04/25 | User1 | |
| Sample App | 4103 | High | Staff | Resource not found | 2011/04/25 | User1 | |
| Sample App | 4103 | High | Staff | Resource not found | 2011/04/25 | User1 | |
| Sample App | 4103 | High | Staff | Resource not found | 2011/04/25 | User1 | |
| Sample App | 4103 | High | Staff | Resource not found | 2011/04/25 | User1 | |
| Sample App | 4103 | High | Staff | Resource not found | 2011/04/25 | User1 | |

Showing 1 to 10 of 200 entries

Previous

1

2

3

4

...

20

Next

More Pages

Search:

Pull 200 More Records

Copyright © QTC Management, Inc. 2022

Figure 13: Dashboard - display paginated records (user level)

5.2.3 Sidebar

QTC

Dashboard

VA

RHRP

>

?

Sensitive/Classified Information

Bryan Gonzalez - Last Login 10/24/2022 10:43AM

VA - Application 1

Show

10

☐ User Errors ☐ System Errors ☒ All

Search:

entries

| Application Name | Layer | Module | Alert | Alert Team | Severity | Server Name | Error Code | Error Message | Error Date | User | View Record |
|------------------|---------|--------|-------|------------|----------|-------------|------------|--------------------|------------|-------|-------------|
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |

Showing 1 to 10 of 200 entries

Previous

1

2

3

4

...

20

Next

More Pages

Pull 200 More Records

Copyright © QTC Management, Inc. 2022

Figure 14: Sidebar - collapsed sidebar

5.2.4 User Experience

5.2.4.1 No Additional Records

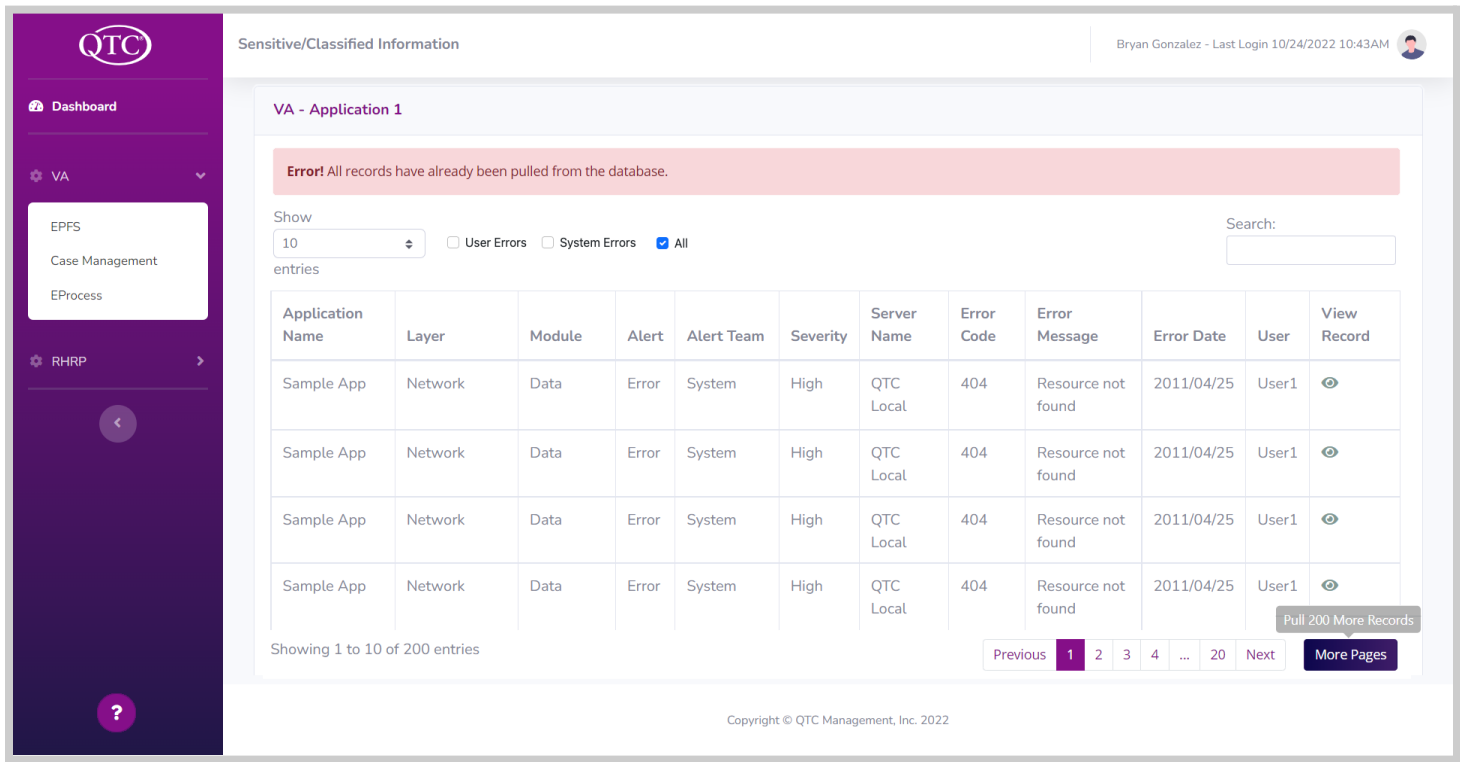


Figure 15: User Experience Design - display a message to consolidate user experience

5.2.4.2 Help Window

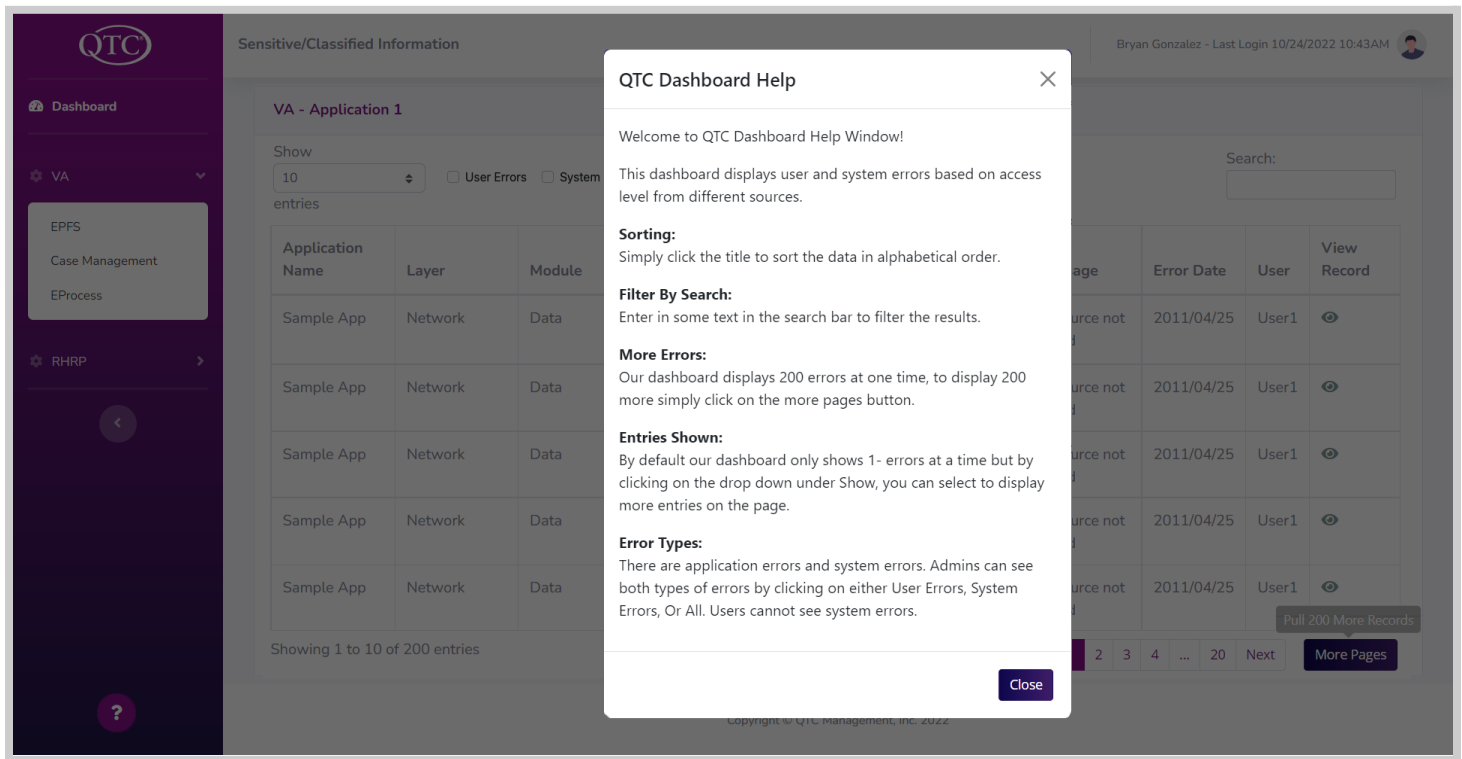


Figure 16: Help Window - display functionality to accommodate the user experience

5.2.4.3 Additional Records

The screenshot shows the QTC dashboard with a sidebar on the left containing 'Dashboard', 'VA', 'EPFS', 'Case Management', 'EProcess', 'RHRP', and a help icon. The main content area is titled 'Sensitive/Classified Information' and 'Bryan Gonzalez - Last Login 10/24/2022 10:43AM'. A green banner at the top of the main area says 'Success! 200 more records have been pulled from the database.' Below this, there's a 'Show' dropdown set to '10' and radio buttons for 'User Errors', 'System Errors', and 'All' (selected). A search bar is on the right. A table displays error records with columns: Application Name, Layer, Module, Alert, Alert Team, Severity, Server Name, Error Code, Error Message, Error Date, User, and View Record. The table shows four identical rows for 'Sample App' with a '404' error. At the bottom, it says 'Showing 1 to 10 of 200 entries' and has pagination controls (Previous, 1, 2, 3, 4, ..., 20, Next) and a 'More Pages' button. A tooltip 'Pull 200 More Records' is over the 'More Pages' button. Copyright © QTC Management, Inc. 2022 is at the bottom.

| Application Name | Layer | Module | Alert | Alert Team | Severity | Server Name | Error Code | Error Message | Error Date | User | View Record |
|------------------|---------|--------|-------|------------|----------|-------------|------------|--------------------|------------|-------|-------------|
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |
| Sample App | Network | Data | Error | System | High | QTC Local | 404 | Resource not found | 2011/04/25 | User1 | |

Figure 17: Additional Records - display message to consolidate user experience

5.2.4.4 View Record

This screenshot is similar to the previous one but includes a modal dialog box in the center. The dialog is titled 'Sample Application - Bad Email Address Entered' and contains the text: 'The email address entered "example@abbcom" is not in a valid format which is "example@abc.com"'. There is a 'Close' button at the bottom right of the dialog. The background content is dimmed. All other elements, including the sidebar, success message, table, and pagination, are identical to the previous screenshot. Copyright © QTC Management, Inc. 2022 is at the bottom.

Figure 18: View Record- display record detail

5.3 REFERENCES

- Exam File Manager's (EFM) Technical Specification Document

5.4 GLOSSARY

Smart Dashboard - consolidate and paginate errors for user

ASP.Net MVC - model-view-controller framework provides patterns-based ways to build dynamic websites.

.Net 6.0 - full stack framework and allows developers to build applications for cloud, web, desktop, mobile, gaming, etc...

C# - programming language that enables developers to build many types of secure and robust applications that run in .NET

Entity Framework - open-source ORM framework for .NET applications supported by Microsoft; allows developers to work with data using objects of domain-specific classes without focusing on the underlying database tables and columns where this data is stored

Stored Procedure - has the ability to automatically build native commands for the database based on your

LINQ-to-Entities or Entity SQL queries, as well as build the commands for inserting, updating, and deleting data.

Structured Query Language (SQL) - is a standardized programming language that is used to manage relational databases and perform various operations on the data in them.

SQL server 2019 - designed to solve challenges of the modern data professional including Store enterprise data in a data lake and offering SQL and Spark query capability overall data

Window server - supports enterprise-level management, data storage, applications, and communications

Window authentication - help you sign in to your accounts when you're using two-step verification

URL (uniform resource locators) - web pages also have unique addresses to help people locate them

Plugins - a software component that adds a specific feature to an existing computer program

A. ACRONYMS

DB Database

API Application Programming Interface