

Research Article

BI DIGITAL PORTFOLIO FOR VARIOUS STORE FRONTS

* Farhana Sethi

Global Data & Analytics Business Intelligence - Quality & Governance Manager with Schlumberger Oilfield, Texas, Houston since 2006

Received 15th August 2020; Accepted 20th September 2020; Published online 24th October 2020

ABSTRACT

BI Portfolio central cataloging helps business users to understand what BI Assets an organization has, where it is located and what stories it tells. It helps to prioritize work and resources. BI portfolio provides a central place where all business intelligence assets are cataloged. It offers over 3000 BI Assets, each BI Asset is align with one or more Storefront mapped with various Categories, tags and Key processes.

Keywords: Application packages, Design Concept, Turnkey Solution, Intranet Portal.

INTRODUCTION

BI Portfolio provides assured operational decisions with the powerful resource management capability. It Improve resource management efforts and drive business efficiency. It helps to monitor the business and get answers quickly with rich dashboards available on every device in the organization. Hence it improves decision-making confidence with real-time reports and accurate data and context readily available, everyone across the organization is able to make better decisions, faster. In our research we have designed, developed and implemented BI Portfolio application that provides a central place where all BI assets are cataloged. Each Storefront is link with BI portfolio using Key Processes and Categories, yet organized as an individual Catalog within the portfolio. BI Assets are associated with meaningful tags which provides strong search capabilities within each storefront. Users can easily drill down to the exact combination of what they want to explore. They can also mix and match with other filters that are available. To make all BI assets discoverable, it provides a taxonomy to the catalogs using tags. This allows users to explore effortlessly what is available in the organization. Every user have a personalized view of their analytics content, they can bookmark content. Store front saves the information what has been viewed by the user, what has been liked by the user and the most viewed and last added BI asset in the storefront. Further related information can also be linked for each Dash-board/Report like Access Request, Documentation, Ask & Learn, utilization Statistics and related Videos.

Application Features

There are some key features for the application that make it unique and useful, as mentioned below.

Catalog for all BI Assets

BI Portfolio is a central repository for all the BI Digital storefront in the organization. It contained many Storefronts based on the Product line, business unit and function. Each Storefront is link with BI portfolio using global Categories and tags, nevertheless organized as an individual Catalog within the portfolio. BI assets are associated

with meaningful tags which provides strong search capabilities within each storefront.

Storefront for the Product line and/or Business Group

Every Product line and /or Business unit in the organization is a strategic business unit that is a segmented group or department within a company that focuses on reaching a specific market or client. It may also focus on achieving a specific goal for the organization while keeping the shared and global organization vision. Storefront is a set of BI Assets that are pertained to a business unit or product line.

Personalization

Central catalog provided customization for every storefront such as adding a log for the business unit or product line, having a unique color/theme, categories, key process and owners for Key process and Product owners.

On the user's level

Users have a personalized view of their analytics content, they can star (bookmark) content. Store front saves the information what has been viewed by the user, what has been liked by the user and the most viewed and last added BI asset in the storefront.

Social

Business users and analytics team members can easily communicate issues or feedback using focused discussions per analytics asset such as Feedback, Reviews, and Support and feature requests.

Integrated with the ecosystem

It is a central BI analytics portal and set up as Single-Sign-On with viewer integration. It uses Domain Name System (DNS) and CNAME (Canonical Name record) that is easy to remember and searchable from central internal hub .it also provides authentication on the mobile integrated with Microsoft Intune which is a cloud-based service that focuses on mobile device management (MDM) for the organization.

Structure the BI-Portal using Tags

To make all analytics assets discoverable, it provides a taxonomy to the catalogs using tags. This allows business users to explore what is available in the organization.

*Corresponding Author: Farhana Sethi,

Global Data & Analytics Business Intelligence - Quality & Governance Manager with Schlumberger Oilfield, Texas, Houston since 2006.

Drill down to any combination
 Users can easily drill down to the exact combination of what they want to explore. They can also mix and match with other filters that are available, such as a filter to only select Power BI dashboards.

Supporting Content

Add supporting content to each BI asset, like: source data, a user guide, utilization metrics or a video walkthrough. This makes it substantially easier for new users to know their way around the BI asset, especially if it requires more context to understand the analytic assets.

Mobile View

Application supports mobile view using RWD. Responsive web design (RWD) is a web development approach that creates dynamic changes to the appearance of a website, depending on the screen size and orientation of the device being used to view it.

Integrated Viewer

It provides a view of all BI asset right from within Central Catalog, whether it is a hosted tableau dashboard, Power BI report, SAP report, a pdf report, an excel report or a custom dashboard. With central portfolio users can request for access by setting up Single-Sign-On with these other services (like Tableau), so there is no extra authentication step needed to open and view a BI Asset.

Application Architecture

BI Portfolio has been designed and developed in the Microsoft Azure Cloud using SharePoint Online as described in high level architecture diagram Figure 1 Microsoft Azure is a cloud computing service created by Microsoft that can be used for deploying and managing applications as well as services through a global network of Microsoft-managed data centers.

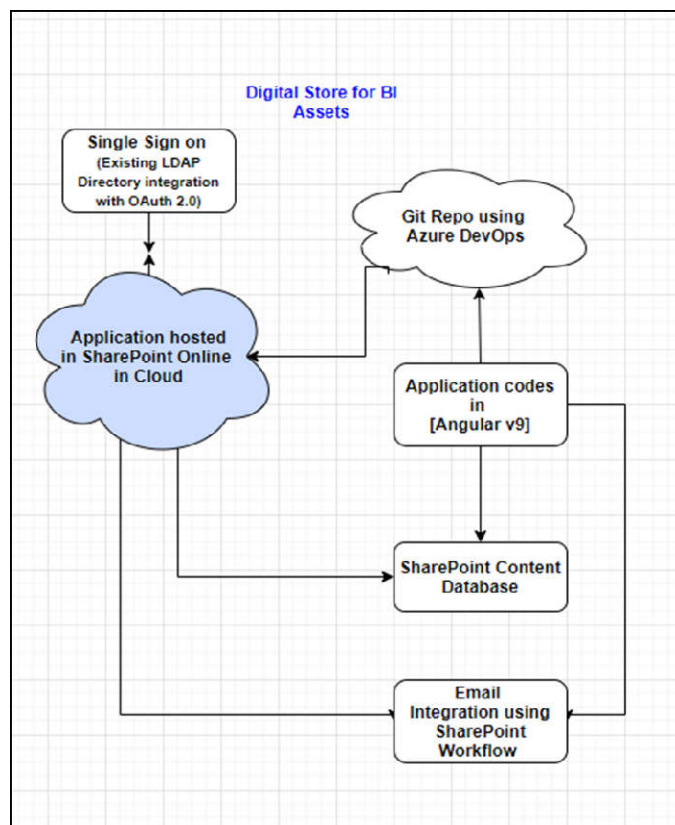


Figure 1: High-level Architecture diagram – BI Digital Portfolio

Key technology and components

Key technology and components of Digital store are as follows:

Web Framework

In our research we have considered and compare Angular, Angular JS and React for the selection of web framework to develop the application. There are several other web framework available such as Vue, Ember, Meteor, Polymer, Backbone, Aurelia and Mithril. In our re-search, we have not included the comparison of all other frameworks. Angular: Angular is a web framework and a platform for building mobile and desktop web applications using HTML and Typescript. Angular is written in Typescript (Typescript is an open-source programming language developed and maintained by Microsoft). Angular is used in many public-facing application like Google and You-Tube. It is a completely free framework helps to use HTML as a template language, creates RICH Internet Application and offers its developers a client-side application. We have used Version 9 in our application development, Angular 9 was released on February 6, 2020. This latest version moves all applications to use the Ivy compiler and runtime by default. Ivy is the code name for Angular’s next-generation compilation and rendering pipeline. Also it has been up-dated to work with Typescript 3.6 and 3.7. During the development, we discovered that Angular is at least five times faster than AngularJS due to a much better algorithm for data binding and a component-based architecture. The components of an Angular application are quite independent and self-sufficient, which makes them reusable and test friendly. Angular used Typescript and Angular JS uses JavaScript. The main advantage of Typescript over JavaScript is that Typescript is a superset of JavaScript. So Typescript designed for the development of a large program that Trans compile to JavaScript. But Typescript may be used to develop a JavaScript application for both client-side and server-side.

Angular VS. React

There is s detail comparison has been done between Angular and React as described in Table 1. To conclude our comparison, Angular is a full-fledged mobile and web development framework. React is a framework only for UI development, which can be turned into a full-fledged solution with the help of additional libraries. In the end, React vs Angular is all a matter of personal preference, a matter of skills and habits. We have used Angular v9 for the application development.

Microsoft SharePoint online

SharePoint is a cloud-based service offered from Microsoft that helps organizations share and manage content, knowledge, and applications to empower teamwork, quickly find information and seamlessly collaborate across the organization. In our research and deployment we have privileged enterprise infra-structure setup of SharePoint online in cloud to host the application that provides single sign on authentication using O365 with directory services, out of the box workflow development for email send and receive and host the database using SharePoint online Document library and SharePoint lists.

Modern Authentication

Modern Authentication is the term Microsoft uses to refer to their implementation of the OAuth 2.0 authorization framework for client/server authentication. By default, modern authentication is enabled for SharePoint online.

Table 1. Comparison between Angular and React Framework

Features	Angular	React
Brief Overview	Full-fledged open-sourced JS Framework managed by Google and the Angular developer community Both Web and Mobile	JS Library for UI development Managed By Facebook and open-source community of developers Both Web and Mobile
Universality	For mobile development, it needs to be incorporated with Cordova. Also cab connected with React Native.	In mobile development, however, a great share of work is done by Ionic. The counterpart of React Native is Native Script.
Self-Sufficiency	Angular is a full-fledged framework for software development Hence no additional libraries needed	Single and multiple-page apps React is a framework for UI development Extra libraries needed
Learning Curve	Angular is more complex to understand, there is a lot of unnecessary syntax, and component management is intricate. Long and difficult	React is minimalistic. However, it takes quite some time to learn how to set up a project. Fast and Simple with JS technical background
Community	Large	Large
Performance	Competitive Optimized with change detection	Competitive Optimized with Virtual DOM (Document Object Model)
Language	JS, Type Script	JS ES6+,JSX script
App Structure	Fixed and Complex	Flexible component-based view only
UI Components	Component-based Model, Controller, and view Built-in Material Design toolset	External Material-UI library and Dependencies
Directives	It supports dependency injection. Angular allows having different lifecycles for different stores.	React does not fully support dependency injection.
Dependency Injection	Incomprehensible without knowledge of Angular	Easily understood with knowledge of JavaScript
State Management	Supported Effective improved with NgRx or RxJS	Not Supported Effective Improved with Redux or MobX
Data Binding	Bidirectional Data is mutable	Unidirectional Data is immutable
Change Rendering	Real DOM, Change detection	Virtual DOM

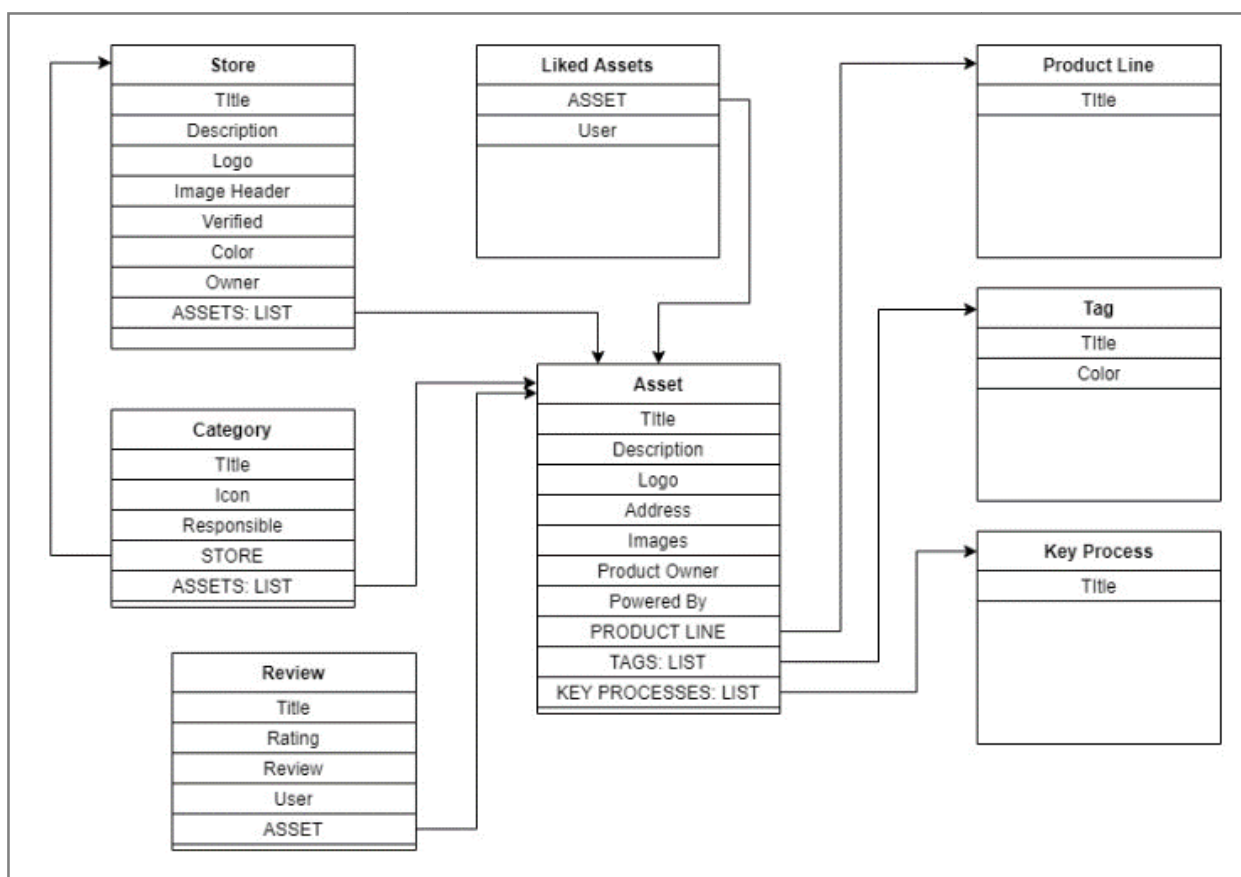


Figure 2. Column mapping for Data association

SharePoint content database

A content database is a database file that stores content for one or more site collections for SharePoint web application. The content can be pages, files, documents, images and much more. In our case, we have used SharePoint document library, lists and images to save the Catalog information.

SharePoint Online Workflow

Workflows help people to collaborate on documents and to manage project tasks by implementing business processes on documents and items in a SharePoint site. In our research and deployment we have set up custom workflow to send emails for Feedback and support form.

Visual studio Code

We have used Visual studio Code for code development. Visual Studio Code is a lightweight but powerful free- source code editor made by Microsoft. It has built-in source code control including Git support. The Visual Studio (VS) Code product itself ships under a standard Microsoft product license, as it has a small percentage of Microsoft-specific customizations. It's free despite the commercial license. We have all the good reasons to use VS Code for our research and tool deployment.

Data Structure

There are 8 main list has been used to embrace the catalog information. Every SharePoint list has columns to keep the information for the catalog as described in Table 2.

Table 2: SharePoint content database Column level detail

List Name	Stores
Column	Description
Title	Store Name
Description	Store Description
Verified	Store validation Status with Portfolio Owner
Color	Theme of individual Storefront
Owner	Name of the Storefront owner
Logo	Image logo
Image Header	Storefront image
Assets	Associated BI Assets
CoOwners	Name of the Storefront Co-owner
Priority	Sequence of Store display at home page
List Name	Product Line
Column	Description
Title	Name of Product Line
List Name	Tags
Column	Description
Title	Tag name
Color	Display Color for the tag
List Name	Categories
Column	Description
Title	Category name
Icon	Icon for Category
Responsible	Person responsible/Owner for the Category
Store	Associated Storefront
Assets	Associated BI Assets
List Name	Key Processes
Column	Description
Title	Key process Name
List Name	Help links
Column	Description
Title	Header text for Help
Link Address	URL for help
Store	Associated Storefront
List Name	Assets
Column	Description
Title	BI Asset Name
Description	BI Asset Description
Logo	BI Asset Logo
Address	BI Asset Address
Images	Image (e.g. Metro map)
Product Owner	Name of the BI Asset Owner
Powered By	Department Acronym
Tags	taxonomy to the catalogs
Product Line	Product Line Name
Documentation Link	Key manual for the BI Assets
Request Access Link	Access Request link
AskAndLearn	Question and Answers
Key Process	Key process Name
Usage Stats	Utilization statistic link for individual BI Asset
Video Url	Video link for individual BI Asset
Search Tags	Search tags for underline data
CoProduct Owner	Name for the Co product Owner
List Name	Reviews
Column	Description
Title	Comment from Reviewer
Rating	Rating from 1 to 5
Review	Reviewer Name
User	User who reviewed it
Asset	Associated BI Assets
List Name	Liked Assets
Column	Description
Title	BI Asset Name
User	User who liked the BI Asset
BI Asset	Asset Name

Mapping Techniques

A Smart mapping technique has been used to link BI Assets with the storefronts and other key features as described in Figure 2.

Folder Structure

The key guidelines for creating folder structure for an Angular Project is to organize the app such that you can locate code quickly and have a near-term view of implementation and a long-term vision. We have built the structure which focuses on a multiple-module architecture which in turn are divided into pages and a shared set of components with a large focus on scaling.

Code architecture

We have adopted the MVCS code architecture for the application development. The Model-View-Controller (MVC) is an architectural pattern that separates an application into three main logical components: the model, the view, and the controller. MVC is one of the most frequently used industry-standard web development framework to create scalable and extensible projects

Models - To manage the data of an application. The models will be anemic (they will lack functionalities) since they will be referred to the services.

Views/Templates - The page/GUI where the user interacts with the app.

Controllers - Links between services and views.

Services - Manages the SharePoint request data.

Angular Components

Components are the most basic building block of an UI in an Angular application. An Angular application is a tree of Angular components. Every page has multiple components and one component can belong to many pages. Below is an example of components breakdown and association with the Storefront landing Page as described in Figure 3.

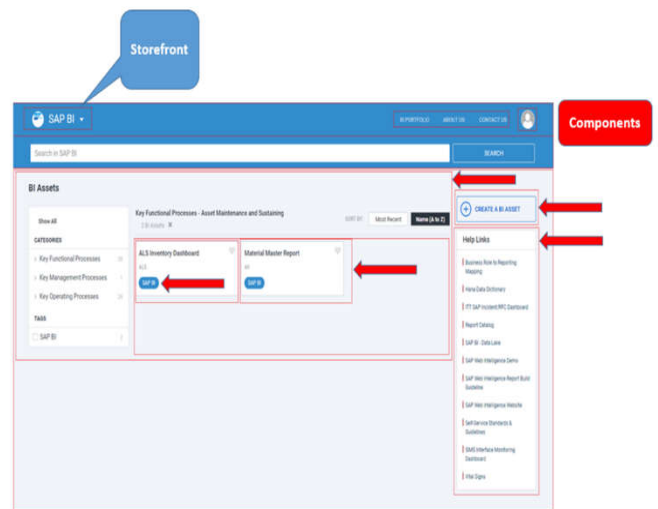


Figure 3. Components Association with the Web page

There are no more or less important components, all are important, but there are more complex or more used components.

Key Components

There are over 40 components being used to develop the application with all the required features in the scope. Top 5 key components has been identified and mentioned below.

- Asset Card
- Tag Card
- Storefront card
- Assets area in storefront page
- Search Bar

Inside the Components

Every Component has to be written in three languages.

HTML - Hypertext Markup Language (HTML)

Scss - Scss stands for Sassy Cascading Style Sheets and it's an extension of CSS.

TS – Type Script is an open-source programming language developed and maintained by Microsoft.

SharePoint Connection

SharePoint ship with an updated REST API that can be used to access and control nearly every aspect of a site collection: Lists, Libraries, Permissions, and Users are just some of the objects that have defined endpoints. Integrating Angular, a powerful client-side framework, gives developers and content managers another option for customizing the user experience. The most important library used in the project is to create and manage the connection between the application and the SharePoint structure.

Deployment technique

We used Minimum Viable Product (MVP) approach for product development strategy. A minimum viable product (MVP) is a version of a product with just enough features to satisfy early customers and provide feedback for future product development. Every MVP includes key elements of functionality, design improvement, reliability by rigorous testing and usability by making sure that each MVP was easy to use and intuitive. We have used Azure DevOps for code deployment. Azure DevOPS is a Software as a service (SaaS) platform from Microsoft that provides an end-to-end DevOps tool chain for developing and deploying software. Azure DevOps being used for code versioning and release pipeline using Git repository with Visual Studio code tool. A release pipeline is a conceptual process by which we take committed code into production. A release pipeline takes a build artifact, a result of the build process and deploys that to one or more environments. With source control, developers can collaborate, track and save their changes to a branch and these changes will be merged into the main branch and deployed to the higher-level environments (From development to Quality Assurance to Production), where it will also be tested and validated. Once the above mentioned steps are complete the developed codes be published into the development environment as described in Figure 4

Azure Release Pipeline Approval/Pre Deployment Conditions

Approvals and gates gives further control over the execution and success of the deployment pipeline. Each step in a release pipeline can be built with pre-deployment and post-deployment settings that can embrace waiting for authority users to manually approve or reject deployments, and testing and validating the quality checks with any

computerized systems until definite requirements are complete. Once the approvals are in place the Release would be promoted to higher environments and finally deployed in Production.

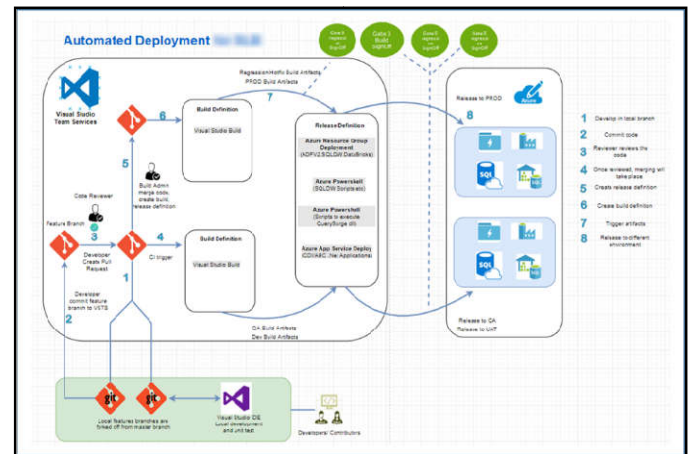


Figure 4. Automated deployment using Azure DevOps (Git) Repository

Output

Application in production has below key displays that will be interesting to show as described in Figure 4, 5, 6 for BI Portfolio Landing page, Inside the Storefront and Report Selection respectively.

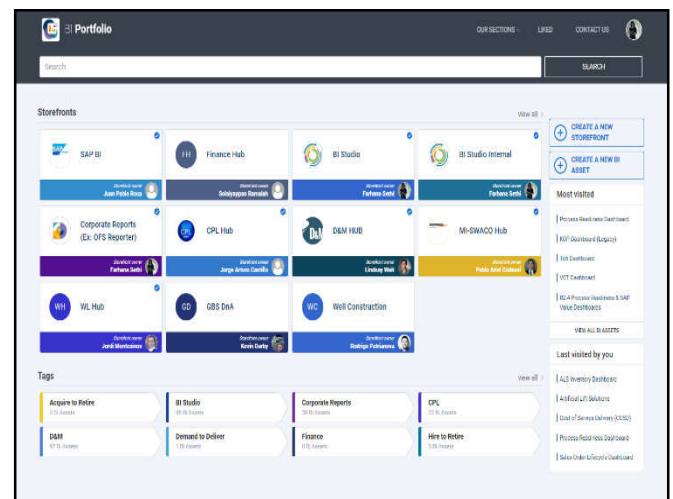


Figure 4. BI Portfolio landing page

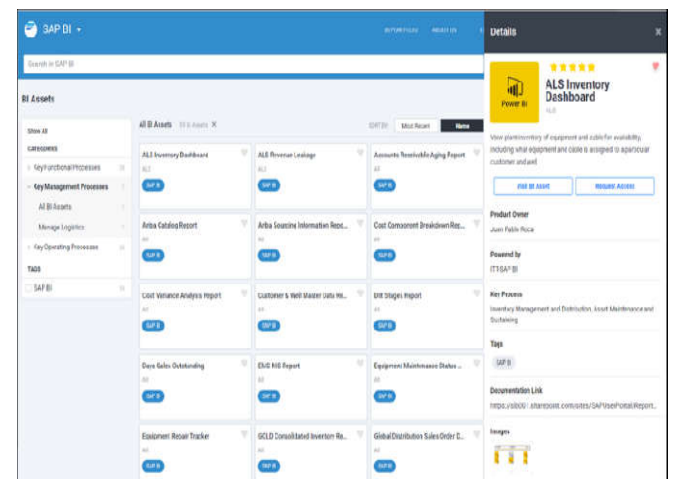


Figure 5. Inside the Storefront

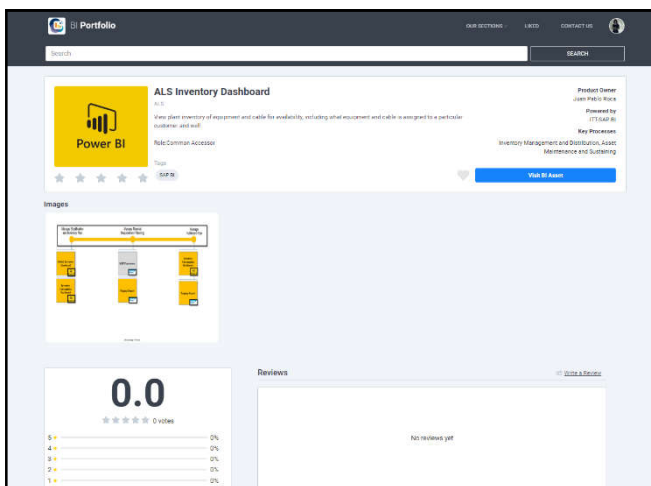


Figure 6. Report Selection

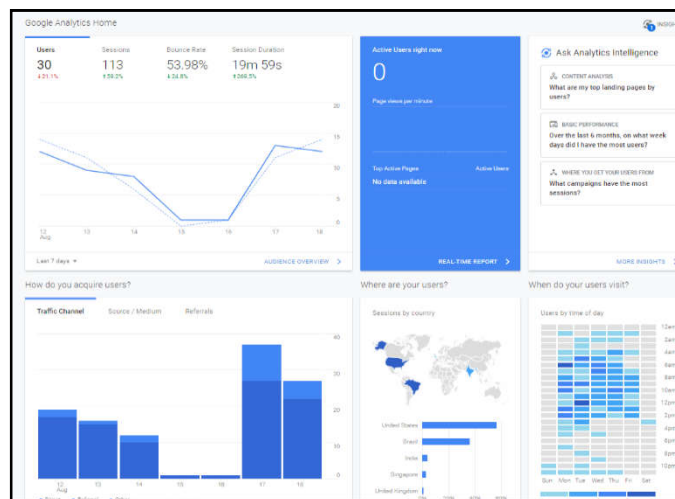


Figure 7. Overall traffic analysis from Google analytics

Utilization tracking

Utilization tracking for the Web application is one of the key strategy to get the adoption over the period of time. If you're not tracking enough data, you won't see how trends change over time or why some users behave one way versus another. You won't understand user engagement. Ultimately, you just won't have the knowledge to make smart product or marketing decisions. We have enabled and configured Utilization of the BI portfolio application due to the precise reasons as follows.

- To find out which storefront bring the most traffic and conversions.
- Determine where the best visitors are located in the worldwide.
- Learn what people are searching for on the application.
- Visualize what people click on the most.
- Uncover the top content. Which pages keep the visitors on the site the longest.
- To identify the worst performing pages.
- Determine where people abandon due to having multiple clicks
- We have used and compared SharePoint activity tracker and Google analytics to enable application tracking.

Microsoft SharePoint Analytics

We get a view into SharePoint activity using Microsoft SharePoint Analytics by looking at the Files and Users views. It is out of the box services available within Microsoft SharePoint Online in Cloud. The Pages view shows the number of unique pages visited by the users.

Google Analytics

1"Site owners using Google Analytics have the capacity to keep track of views over time, follow site visitors' movements in real time, pinpoint where site visitors leave, view demographics and maps, and understand the search terms that initially brought visitors to the website"1.In order for Google Analytics to function, we were required to put JavaScript tracking code on each page of our application. When a visitor comes to the application, the tracking code runs and sends tracking data to Google's data collection servers. In our analysis we decided to use Google analytics as it provides detail and comprehensive active user's trend over the period of time as described in Figure 7 below.

Security Overview

BI Portfolio is hosted in Microsoft SharePoint Online environment in Cloud, hence it is inherited the security framework what SharePoint offered. SharePoint has the default security groups which are the SharePoint objects that have "users" (Azure Active Directory Users and Azure Active Directory Groups by default) as members and come with their own settings.

There are 5 main layers of Security available in the BI Portfolio environment:

- Read:** Can view pages and list items and download documents.
- Contribute:** Can view, add, update, and delete list items and documents.
- Edit:** Can add, edit and delete lists; can view, add, update and delete list items and documents.
- Restricted Edit:** Can add, edit and delete specific rows from the list based on the defined role.
- Design:** Can view, add, update, delete, approve, and customize.
- Full Control:** Has full control.

There are three security groups. Members, Owners and Visitors. Users required to be added into a security group. Each security group has unique Permission level as described in Figure 8.

Security Groups	Permission Level					
	Read	Contribute	Edit	Restricted Edit	Design	Full Control
Members	✓	✓	✓	✗	✗	✗
Owners	✓	✓	✓	✓	✓	✓
Visitors	✓	✗	✗	✗	✗	✗
site collection Admins	✓	✓	✓	✓	✓	✓
Product owners	✓	✓	✗	✓	✗	✗
Storefront Owners	✓	✓	✗	✓	✗	✗
Product Co-Owners	✓	✓	✗	✓	✗	✗
Storefront Co-Owners	✓	✓	✗	✓	✗	✗

Figure 8. BI Portfolio Security Overview

Self-Service for Catalog maintenance

Self-service is the practice of serving oneself and empowered the clients. We have designed and built the application in a way that it required least support and maintenance from the BI Portfolio Admins and IT Staff. One of the key benefits of self-service to an organization is that they save on resources. Also Customers have better control and privilege on the Content and hence engagement and adoption will increase eventually. We have considered all those factors in the consideration and designed BI Portfolio application in such a manner where Storefront Owners, Product Owners and Co-Owners can maintain their assigned BI Assets on their own. Content is the essence of any digital application. The BI Portfolio catalog has to be current, consistent, and correct. There are three main components of BI portfolio that require to be updated and accurate all the time.

- Storefront
- BI Assets
- Mapping components

Every Storefront owner, Product Owner or Co-owner can add/edit. Update or delete their storefront information from the administrator console from the front end of the BI Portfolio application.

CONCLUSION

In this paper, we demonstrate that how BI Portfolio digital catalog can be developed for multiple storefront. The proposed method can be implemented to any large organization. The developed scripts has been implemented in real time in the organization. Furthermore, if the identified approach is also suitable for different processes, the scope of the analysis can be extended using a similar approach.

Acknowledgement

The author would like to thank the technical support from the department of IT in the Schlumberger Oil industry to gain the information and data gathered for program development.

Citation

1- E. J. Petersen and B. M. Martin, "Misuse, play, and dis-use: Technical and professional communication's role in understanding and supporting website owners' engagement with Google Analytics," 2015 IEEE International Professional Communication Conference (IPCC), Lime-rick, 2015, pp. 1-5, doi: 10.1109/IPCC.2015.7235786.

REFERENCES

- How to define a highly scalable folder structure for your Angular project- <https://itnext.io/choosing-a-highly-scalable-folder-structure-in-angular-d987de65ec7>
- SharePoint Connection Library - <https://www.npmjs.com/package/@cwi/sharepoint-connector>
- Microsoft SharePoint Document <https://docs.microsoft.com/en-us/sharepoint/introduction>
- Visual Studio Code:Version: 1.48.0 (user setup)
- Visual Studio Code Keyboard <https://www.arungudelli.com/microsoft/visual-studio-code-keyboard-shortcut-cheat-sheet-windows-mac-linux/>
- Microsoft Azure DevOps <https://azure.microsoft.com/en-us/services/devops/>
