EMPLOYEE COLLABORATION PORTAL IN SHAREPOINT

A Master's Project
Presented to
Department of Computer and Information Sciences
SUNY Polytechnic Institute
Utica, New York

In Partial Fulfilment
Of the Requirements for the
Master of Science Degree

by
Sai Sandeep Soumithri Vempati

December 2016

EMPLOYEE COLLABORATION PORTAL IN SHAREPOINT

Master of Science Project in Computer and Information Sciences
Department of Computer Sciences
SUNY Polytechnic Institute

Approved and recommended for acceptance as a project in partial fulfilment of the requirements for the degree of **Master of Science in Computer and Information Sciences**

____

Date

____

Chen-Fu Chiang, Ph. D. (Adviser)

____

Jorge Novillo, Ph. D.

____

Mohamed Rezk, Ph. D.

EMPLOYEE COLLABORATION PORTAL IN SHAREPOINT

Declaration

I declare that this project is my own work and has not been submitted in any form for another degree or diploma at any university or other institute of tertiary education. Information derived from the published and unpublished work of others has been acknowledged in the text and a list of references is given.

VSandeep

Sai Sandeep Soumithri Vempati

Abstract

This project aims at developing a portal for a company's internal needs that include leave portal, a pre-sales dashboard and a document sharing list for the employees in SharePoint Online. SharePoint Online is web based Content Management System (CMS) provided by Microsoft. Microsoft introduced SharePoint in 2001 which was an instant winner. It had all the features that are needed for storage and collaboration. SharePoint later on evolved into two major versions, namely, On-premise and Cloud version. SharePoint the cloud version proved to be a feasible CMS for start-ups and small companies. As the usage of SharePoint Online has minimised the burden maintenance of servers and administration more companies started using SharePoint [5].

The utility of SharePoint has caught the attention of many companies lately. It has scaled up to, 75000 organisations having 160 million users [8]. The usage of SharePoint made companies develop portals that are interactive and act as platforms for collaboration and exchange of information. The workflow automation provided by SharePoint helps in simplifying the business process management. Web technologies can be used to develop the portal in a user friendly and responsive manner.

In this project, a portal is developed that mainly has three functionalities – a leave application platform, a dashboard for Presales and a list that helps sharing of information. The leave application feature is based on the workflow automation service provided by SharePoint in which the user can request concerned manager for a leave approval. The whole process of approval is automated in the portal. The Presales dashboard option helps in viewing data related to projects that can be used to develop reports by the Presales team of a company. The data is shown in various forms suitable for easy understanding using web parts in the dashboard. A list that demonstrates file approval is included in the portal.

Contents

List of Important Figures

Chapter 1

Introduction

1.1 Content Management System

Content Management System is a powerful and scalable tool that facilitates creation, storage and exchange of digital information. Content Management Systems are used for Enterprise content Management (ECM) and Web Content Management (WCM). ECM is used in organizations for exchange of content among the members and have an internal portal for collaboration operations such as approval processes, digital content management, providing user accounts and role-based access to the organisation's data, having different permission levels for content. WCM is used in providing collaborative writing of content. Generally, ECM has WCM features included in the interface. A Content Management System enables user to modify, add, create content. ECM web pages are often within the organisation's scope and are not accessible for public use. CMS also provides support to developers to create responsive and appealing web interfaces [13].

CMS consists of mainly two components CMA and CDA

- CMA (Content Management Application)
- CDA (Content Delivery Application)

CMA provides the functionalities of the GUI (Graphical User Interface) that enable the user to add, modify, create content from the page and user interface without the knowledge of web technologies. The CMA constitutes of the front-end user interface of CMS.
CDA provides the back-end relative to the CMA and help in rendering the changes done in the front end by the user [13].

1.2 Why SharePoint?

In any business organisation there is a need for a common platform to manage documents, interview process, leave applications, web services, task notifications, news bulletin, a platform for free communication and thought sharing, file sharing medium, a metadata management tool, a portal for web services, portals for HR, Project Management Strategies and other various departments, groups for maintaining contacts among the department staff and generate automated business processes for regular tasks are needed. Considering the much needed and generic requirements as stated above, SharePoint suffices the needs of a Business Organisation. SharePoint was initially used as an intra-organisational content management platform developed by Microsoft for its organisational needs. It was later developed into a product and marketed as Microsoft has seen its need in all the organisations in the technology based market.

1.3 What is SharePoint?

SharePoint is a browser based collaboration and document management platform from Microsoft. SharePoint has gained good appreciation among many upcoming small businesses and companies. SharePoint enables companies to maintain intranet portals and handle sharing and exchange of documents and other vital information through the enterprise content management facility. SharePoint's first version was released in 2001 by Microsoft as a simple document sharing and indexing software and has evolved dramatically due to increase in reception by several top business organisations. It started as a simple server management tool and evolved into a Cloud based all-in-one enterprise product having features like social networking, web designing capabilities and workflows integrations on lists and content types [14]. It makes IT professionals more productive and efficient to work on projects as it provides a portal for all the employees to collaborate and work. SharePoint is much appreciated for out of box features and web design editing features which reduces the need for professional developers and DBAs.

The various features of SharePoint (SP 2013) include:

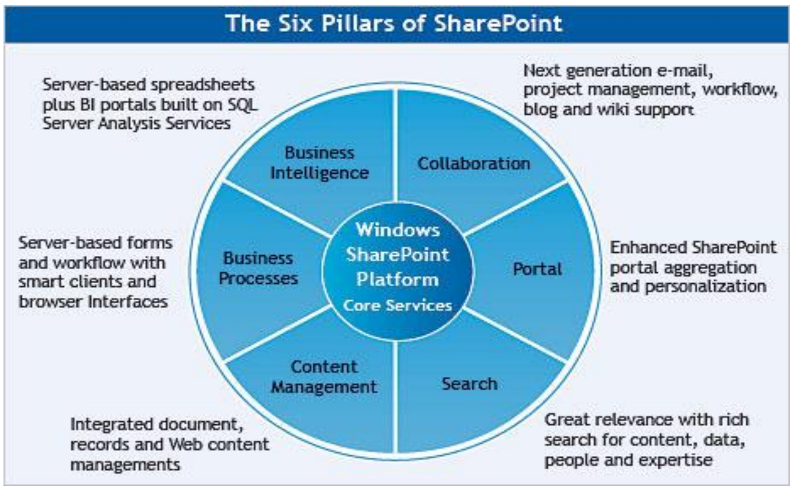

The Six Pillars of SharePoint
Server-based spreadsheets plus BI portals built on SQL Server Analysis Services
Next generation e-mail, project management, workflow, blog and wiki support
Business Intelligence
Collaboration
Windows SharePoint Platform
Core Services
Server-based forms and workflow with smart clients and browser Interfaces
Business Processes
Portal
Enhanced SharePoint portal aggregation and personalization
Content Management
Search
Integrated document, records and Web content managements
Great relevance with rich search for content, data, people and expertise

Figure 1: Features of SharePoint [1]

- Enterprise Content Management
- Business Process Management
- Business Intelligence
- Enterprise Search

- Enterprise Social Networking

Enterprise Content Management (ECM) - It includes all the content management features and functionalities that are used to create, customize and publish site collections and sites. Different types of branded sites can be developed using ECM based on the requirements of the organization. A branded site can be an intranet presence site, an internet business site, intranet site or an extranet site. Intranet presence site is a site that is targeted to the customers, partners and investors of a company. Internet business site is a site approach for promoting products and offers to companies. Intranet site is a website for making content available for sharing among the employees of an organization. Extranet site is used to provide access to information and content to remote employees.

Business Process Management (BPM) – It includes a streamlined way of automating a company's tasks with the workflows feature. Workflows for automated emails and task reporting and deadline driven work (task notifications) and approval of documents can be generated using SharePoint Designer. Business Process management services such as work automation services generating process models using Visio and various BPM related functions can be easily carried out using SharePoint.

Business Intelligence (BI) – Business intelligence refers to analysing and organization's raw data to make critical decisions. SharePoint helps in facilitating BI for generating reports using Excel services came that can be used by incorporated office 365 services. There are many third party applications that can also be used to perform business intelligence related work such as Bamboo. There are options for performing analysis of data using Excel service and Power view feature to publish and share different reports analyzed on the company data.

Enterprise Search – Search feature in SharePoint 2013 provides several query APIs, providing multiple ways to access search results that can return search results in a variety of custom solution types. The search feature in SharePoint is very advanced and easy to use. There are various components such as crawl components, query components and indexing components which make the Microsoft's search architecture framework a very powerful tool.

Chapter 2

SharePoint Server Architecture

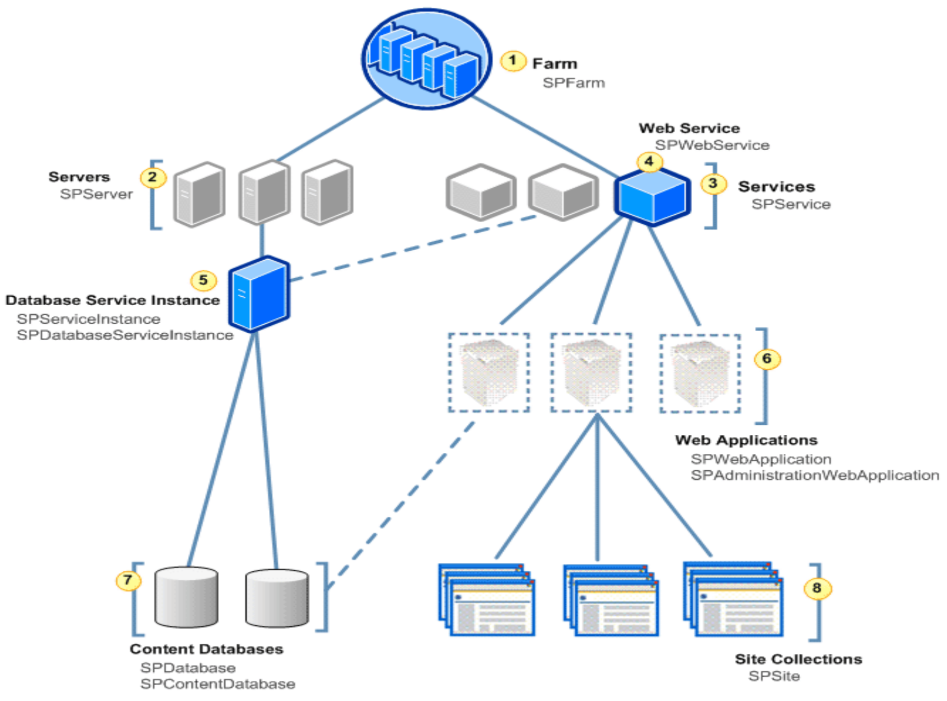


Figure 2: Server Architecture of SharePoint [2]

The figure 2 gives a basic idea about Server architecture of the SharePoint. It shows the setup and object hierarchy of the Windows SharePoint Services. The highest level of object in SharePoint topology is a Farm (SP Farm). There are one or more physical servers having SharePoint installed and all the servers are combined into a single SP Farm. An SP farm can be a combination of multiple severs or a single server. There are numerous servers that are used as SQL servers to store and retrieve data along with the SP servers. Generally, small organizations prefer having a single Farm so that the maintenance becomes simpler.

A **Farm** is a collection of many web services that are deployed on the SharePoint servers. Services that are deployed in the servers are referred as **service applications**. A service application provides a medium to share the various services deployed in the servers across the Farm. Few service applications are Business data connectivity service, Excel service, App management service, Visio integration service, Machine translation service, work automation service. Different functionalities of SharePoint are configured in service applications [17].

A **web application** is a logical unit for site collections. IIS (internet information services) website which is contained in the web application is used to host site collections. There can be multiple site collections under a web application. A Farm can host about 20 web applications [17].

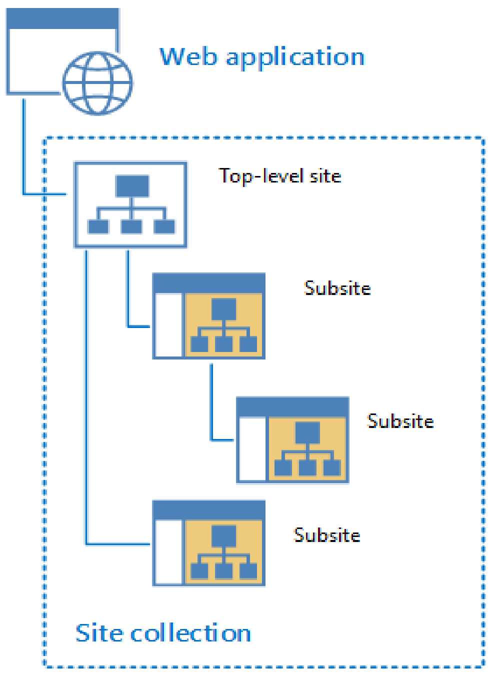


Figure 3: Web application structure [3]

A **Site collection** is a whole collection that consists of all the subsites under it. The collection of sub-sites is kept in a hierarchical fashion under a top level site. The top-level site can have links to subsites to make the sub-sites accessible to users. Example- An HR department can be a site collection which has the top level site an HR home page and the subsites can be leave request site, employee details website etc. About 750000 site collections can be created in a farm [17].

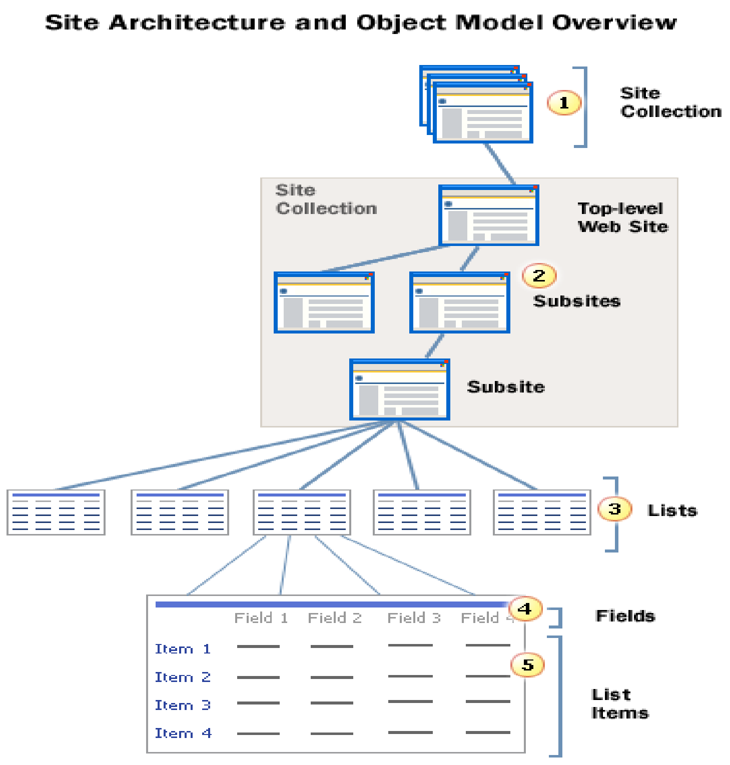
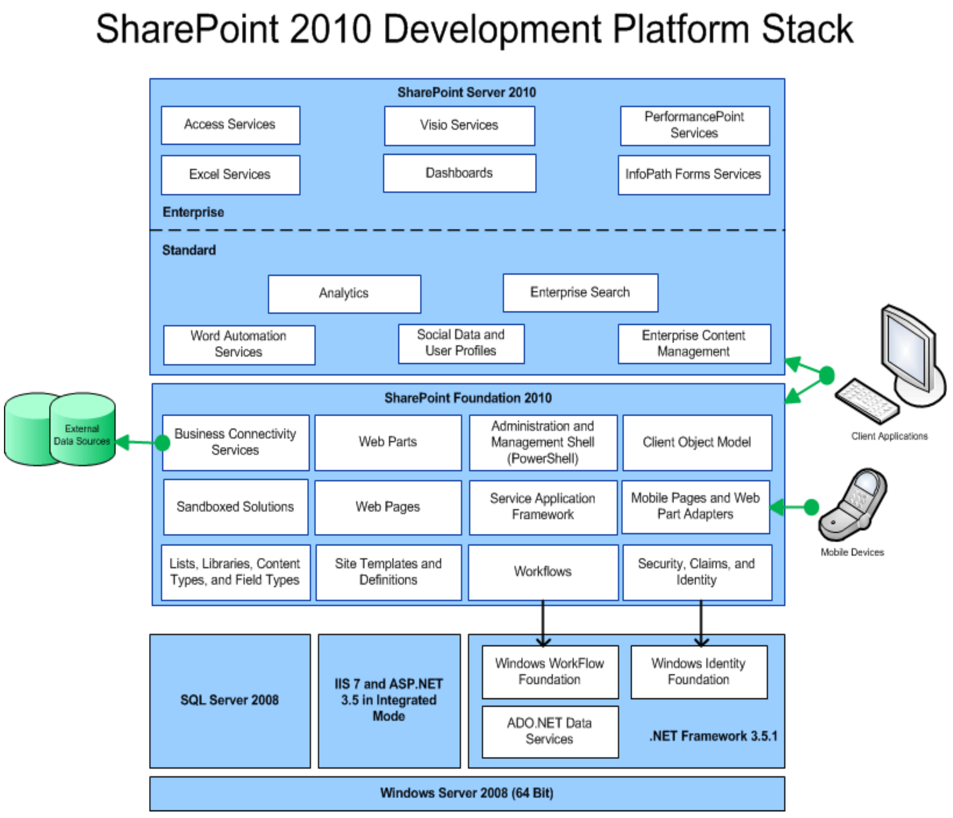


Figure 4: Site architecture and object model [3]

Figure 4 shows the site architecture which includes the objects that are base for every site in SharePoint. As elaborated above the site collection is generally the wholesome collection of Subsite. Each Subsite has a basic unit called list where data can be store just as in relational database. The list consists of two fundamental objects they are fields and list items. Fields are nothing but column entities as in the SQL tables and the list item refers to data in the table.

Subsite- Subsite is an independent site that has data that is contained in it that is shared to targeted users. Different permissions levels can be set up for a Subsite. For example, a group of employees can only view a particular Subsite in the site collection and the other employees are restricted from accessing the site. It simplifies classifying information [17].

List – A list can be termed as the fundamental object of SharePoint. A list consists of fields and list items. Fields are nothing but columns and list items are the data fields as in excel spreadsheets. A basic idea of lists in SharePoint is derived from the .xml model (excel spreadsheet). A **Content type** is a reusable set of columns (metadata). A content type can be a group of lists. A list can have multiple content types attached to it and vice-versa [17].



SharePoint 2010 Development Platform Stack
SharePoint Server 2010
Access Services
Visio Services
PerformancePoint Services
Excel Services
Dashboards
InfoPath Forms Services
Enterprise
Standard
Analytics
Enterprise Search
Word Automation Services
Social Data and User Profiles
Enterprise Content Management
SharePoint Foundation 2010
Business Connectivity Services
Web Parts
Administration and Management Shell (PowerShell)
Client Object Model
Sandboxed Solutions
Web Pages
Service Application Framework
Mobile Pages and Web Part Adapters
Lists, Libraries, Content Types, and Field Types
Site Templates and Definitions
Workflows
Security, Claims, and Identity
External Data Sources
Client Applications
Mobile Devices
SQL Server 2008
IIS 7 and ASP.NET 3.5 in Integrated Mode
Windows WorkFlow Foundation
Windows Identity Foundation
ADO.NET Data Services
.NET Framework 3.5.1
Windows Server 2008 (64 Bit)

Figure 5: Development platform Stack [4]

Figure 5 shows the development stack for the SharePoint 2010 platform. It gives brief understanding of the back end on SharePoint platform. There are few more advancements in the 2013 version which are used in the current project for developing the enterprise portal. In 2013 version feature for hosting cloud apps that can be developed using various development tools such as visual studio, NAPA office 365 development tools and SharePoint designer for editing master pages and creating workflows. SharePoint has made it simple for web developers who use web technologies such as HTML5, CSS, JavaScript, Angular JS, REST to develop solutions other than conventional Microsoft web technology (.net). The enterprise search feature is highly enhanced in the 2013 version, significant advancements are done to the Keyword Query Language (KQL) that is a search syntax for building search queries. Business connectivity services (BCS) to access data from external systems such as SAP, ERP are provisioned in SP 2013, Application services such as PowerPoint automation services , Excel and Access services were enhanced.

Chapter 3

Programming models in SharePoint 2013

There are many web technologies that can be used to develop interactive websites. Few popular and trending languages are JavaScript, HTML,CSS,ASP.NE T, PHP, Ruby on Rails, GO, Python. The object models in SharePoint facilitate using few of the technologies for making the web application hosted in SharePoint more interactive and creative. Mainly there are three programming models provisioned in SP 2013 that facilitate using the technologies supported by the SP platform. They are-

- SharePoint Server Object Model(SSOM)
- Client side object Model (CSOM and JSOM)
- REST API

3.1 SharePoint Server Side Object Model (SSOM)

Server Side object model is the first and the backbone for programming and development in SharePoint. SSOM is highly structured and helps referring to objects in like lists, sites to use them in the code. SSOM is used to write code when the application is being developed in ASP.NET in the SharePoint server. The languages supported in SSOM should belong to the .NET framework as it is a Microsoft product. SSOM is the fundamental model that is an assembly that provides classes and helps in mapping the components that the end user can see. SSOM can be understood by examining the figure 6.

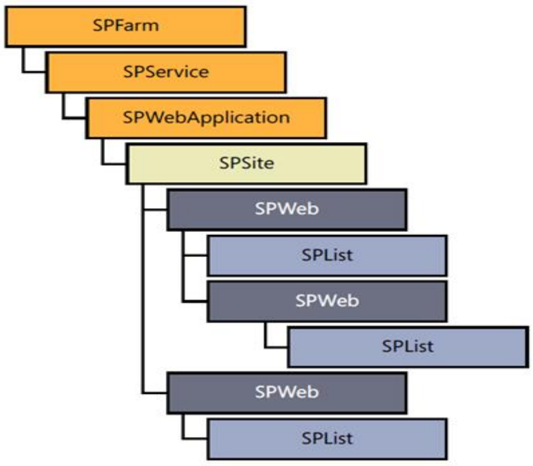


Figure 6: SharePoint Server Side Object Model [5]

The “Microsoft.Sharepoint.dll” is the core assembly of SSOM that is installed in the global cache comprising two main namespaces, namely, “Microsoft.Sharepoint” and “Microsoft.Sharepoint.Administration”. The “Microsoft.Sharepoint” namespace contains classes and objects that help in developing solutions at a site collection level, it helps site developers and application developers work effectively. It is used mainly to manipulate and tweak with the site architecture. SPList, SPWeb, SPSite, SPField are objects of the “Microsoft.Sharepoint” namespace. SPList represents a list; SPWeb represents a single site that includes all the features in the site. SPSite is used to represent a site collection. SPField is used to map field data in a list.

The “Microsoft.Sharepoint.Administration” namespace has classes and objects used for administration purposes .SP solutions that are developed at an administrative level are present here. SPFarm represents the whole server cluster; it is the highest object in the object hierarchy. SPService represents a server. SPWebApplication represents the web-application that is hosted in the IIS website. SPDatabase represents the SQL database that is associated with the web-application. The Server Object Model gets executed in the server side and must be deployed in the same farm.

The below snippet C# code gives an example of SSOM program for adding a list items specified by the user in the text boxes to a list [15]. Assuming that the studentDemo list consists of fields Name, Major, Age.

```
SPWeb ssomDemo = SPContext.Current.Web;
SPListItemCollection listitems1 = ssomDemo.Lists[“studentDemo”].Items;
SPListItem item = listItems1.Add();
item[“Name"] = TextBox2.Text;
item["Major"] = TextBox3.Text;
item["Age"] = Convert.ToInt32(TextBox4.Text);
item.Update();
}
```

Disadvantages of SSOM are that, only C# and VB can be used to code in SSOM and the development has to be done in the machine in which SharePoint is installed. Server-side object model is only used in on–premise version of SharePoint. For SharePoint online, CSOM is used as the main programming model.

3.2 Client-Side object model (CSOM)

In the SSOM, code runs at server front end whereas CSOM is used when the code has to run at the remote machine that is accessing SharePoint. The major advantage of CSOM is that coding can be done remotely with no need to connect to the server. CSOM provides client side application with access to a subset of SSOM which includes core objects such as sites, site collections, lists, list items etc. CSOM consists of three API's - the .NET Framework, the ECMA script (JavaScript) and the Silverlight application. Client side processing is preferred as it works faster than the Server side as the work gets distributed; the client only processes how to display the data and server only processes the data requests. The workflow of the client object model is shown in figure 7.

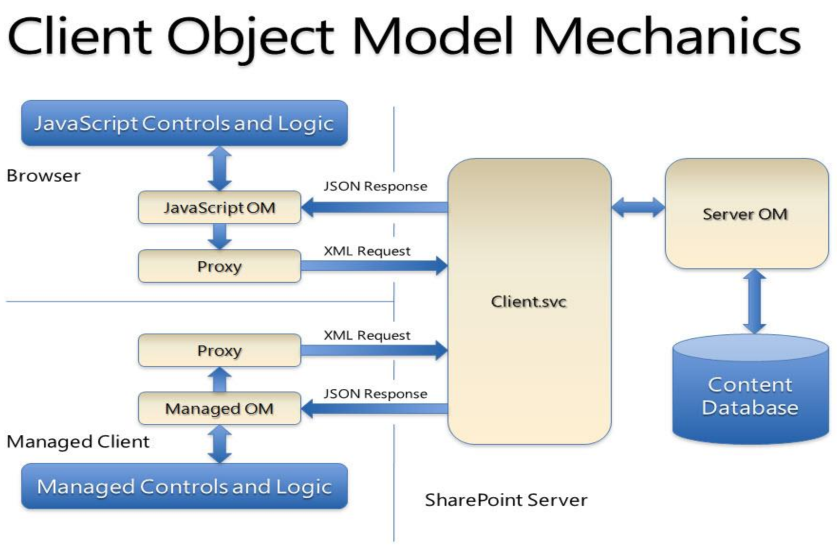


Figure 7: Client Object Model Mechanics [11]

CSOM can be used to perform the CRUD (create, read, update and delete) operations and generally querying of data is done using CAML queries. Using CSOM, synchronous and asynchronous operations can be performed. The CRUD operations are implemented on the list and the front end web parts.

Chapter 4

Motives and Project Idea

In the current market scenario, every company needs an outlook and introspection on their performance in sales, an internal collaboration portal for day to day functional activities, and a sharing platform for the exchange of information. Moreover, the project that is developed has to be a useful idea for the sake of any user and must be a learning experience. Hence, I have thought of an idea that consolidates all the needs of a company into a single accessible location. CMS is the only apt solution when there is a need for a multiple user logins/accounts and exchange of emails, a User Interface for interactive and functional needs such as creation of dashboards etc., approval of a higher tier employee, fancy looks for the web application and bulk supervised uploads and exchanges of files. One such CMS that is used all over the software industry is SharePoint. With this perspective, the project constitutes of a presales dashboard along with few pages that show various metrics, a leave application portal for the employees and a document sharing approval list.

Presales operations are performed in all product based and service based companies which must depend on sales data to understand the product's trend. In IT industries, the presales process is done before connecting with a customer in order to know the needs a customer expects and the performance of the previous project tenders. Data about the projects that were successfully acquired or lost, type of technologies used and other activities involved in sales relative to the company is analysed by the presales team. For this, they need a dashboard to view the data in a feasible and understandable way. Therefore, in this project a presales dash board is included.

For every company, the employees apply for leave, this process can be made easier with a portal where the employees can apply and check the leave history which comprises a list of leaves, date and day information. So, to have a feasible streamlined way of leave approval and application. There is a need for having a leave application portal in all the companies. There is also need for uploading and sharing of variety documents which is essential for productivity. Control over this data is necessary, like to share only approvable data, to have a content administrator for approval of data.

4.1 Technologies and Software’s used

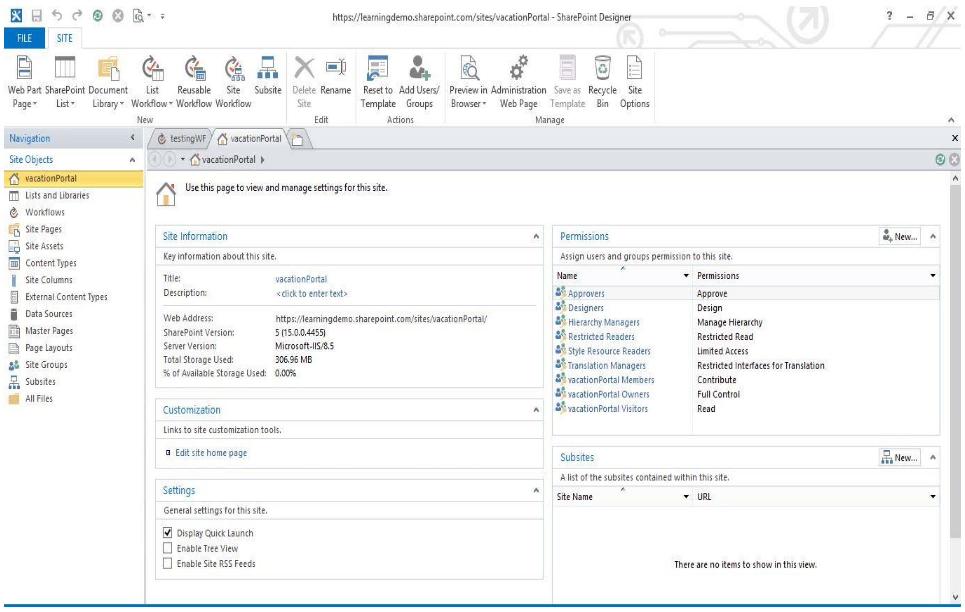
- SharePoint Office 365.
- SharePoint Designer.
- Asp.net to develop Master pages and page layouts.
- Html5, CSS3, Bootstrap.
- Angular JS to perform CRUD operations.

4.2 Master page and Page layouts

SharePoint master pages are the page layout format that is common all across the site used for development. The elements of the page that are common and have an unchanged position are predefined in the master page. Navigation bars, footers, side bars, headers are generally always the same in all the sites in the site collection, therefore, master pages can help in making designing process easier. The parts of site page that differs from each page to the other are known as Content pages such as web parts, lists, and data. The SharePoint online platform packs the content page and the master page into a single page while rendering it through the browser. Customizing the web pages to create a better user interface is termed as branding. Branding is done depending upon the application’s requirements and functionalities. SharePoint provides two out-of-the-box (inbuilt) master pages “Seattle.master”, “v4.master” and “Oslo.master”. A user can either edit these default master pages or create a new master page from scratch.

Creating a master page and page layouts involves these steps:

1. Connect the SharePoint designer to the site collection.



Use this page to view and manage settings for this site.

Site Information

Key information about this site.

Title:	vacationPortal
Description:	<click to enter text>
Web Address:	https://learningdemo.sharepoint.com/sites/vacationPortal/
SharePoint Version:	5 (15.0.0.4455)
Server Version:	Microsoft-IIS/8.5
Total Storage Used:	306.96 MB
% of Available Storage Used:	0.00%

Customization

Links to site customization tools.

Edit site home page

Settings

General settings for this site.

- [✓] Display Quick Launch
- [ ] Enable Tree View
- [ ] Enable Site RSS Feeds

Permissions

Assign users and groups permission to this site.

Name	Permissions
Approvers	Approve
Designers	Design
Hierarchy Managers	Manage Hierarchy
Restricted Readers	Restricted Read
Style Resource Readers	Limited Access
Translation Managers	Restricted Interfaces for Translation
vacationPortal Members	Contribute
vacationPortal Owners	Full Control
vacationPortal Visitors	Read

Subsites

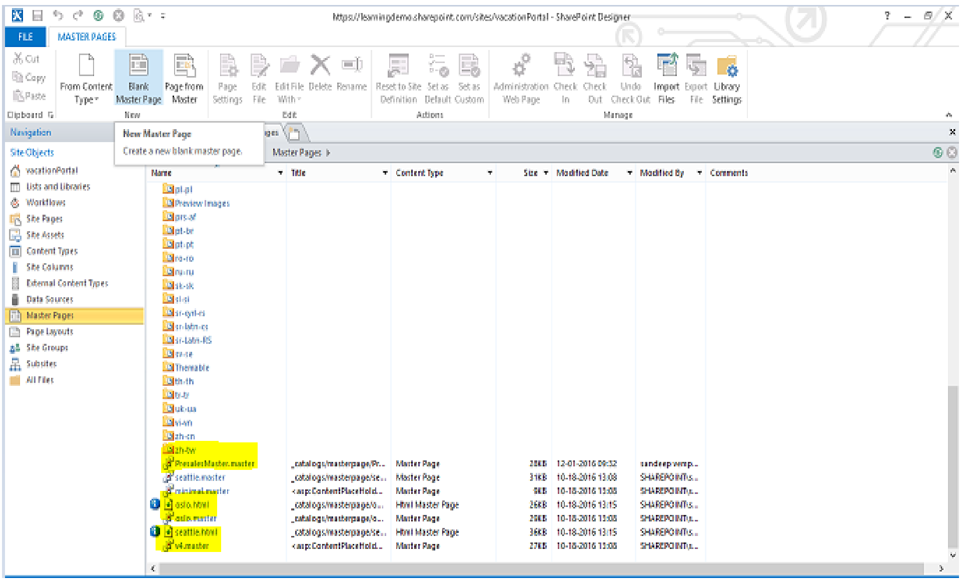
A list of the subsites contained within this site.

Site Name	URL

There are no items to show in this view.

Figure 8: SharePoint Designer

2. Select Master page from site objects navigation bar. The navigation bar contains the ‘Site Objects’ that are part of the site collection. For every site collection there is master page.



Name	Title	Content Type	Size	Modified Date	Modified By	Comments
pl-pl						
Preview Images						
prs-af						
pt-br						
pt-pt						
ro-ro						
ru-ru						
sk-sk						
sl-si						
sr-cyrl-cs						
sr-latn-cs						
sr-Latn-RS						
sv-se						
Themable						
th-th						
tr-tr						
uk-ua						
vi-vn						
zh-cn						
zh-tw						
PresalesMaster.master	_catalogs/masterpage/Pr...	Master Page	28KB	12-01-2016 09:52	sandeep.verap...	
seattle.master	_catalogs/masterpage/se...	Master Page	31KB	10-18-2016 13:08	SHAREPOINT\s...	
minimal.master	<asp:ContentPlaceHold...	Master Page	9KB	10-18-2016 13:08	SHAREPOINT\s...	
oslo.html	_catalogs/masterpage/o...	Html Master Page	26KB	10-18-2016 13:15	SHAREPOINT\s...	
oslo.master	_catalogs/masterpage/o...	Master Page	26KB	10-18-2016 13:08	SHAREPOINT\s...	
seattle.html	_catalogs/masterpage/se...	Html Master Page	36KB	10-18-2016 13:15	SHAREPOINT\s...	
v4.master	<asp:ContentPlaceHold...	Master Page	27KB	10-18-2016 13:08	SHAREPOINT\s...	

Figure 9: Master Pages List in SharePoint Designer

In our project the master pages developed are from the default master pages. By using adding required features to the “seattle.master” page we have created a Presales.master and “LeavePortal.master” pages. A master page can be created using the Blank Master Page on the screen shot above. Default top-level page of a site collection looks as shown below.

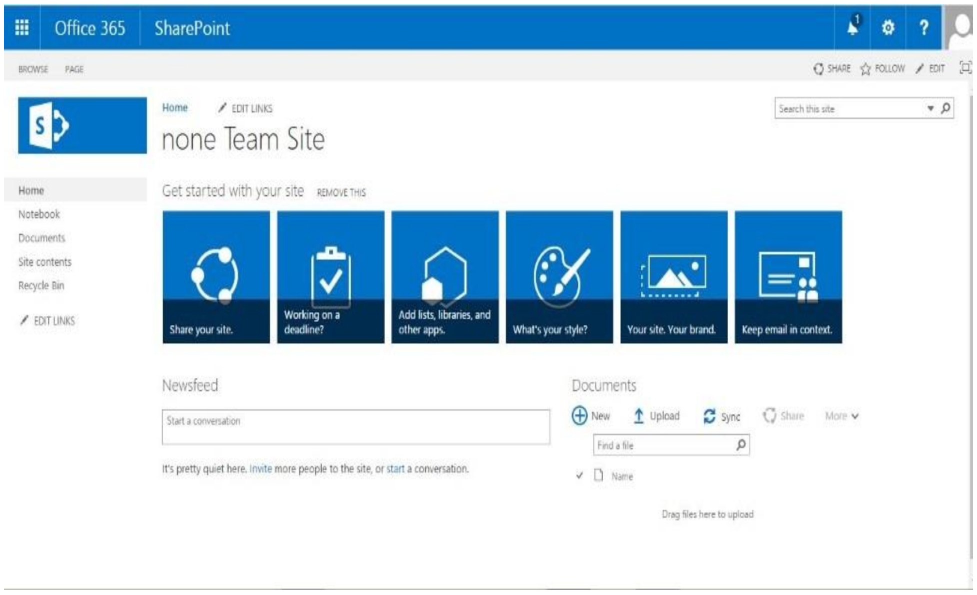


Figure 10: Default SharePoint Start page

The general layout of the web page in SharePoint is shown in figure 11:

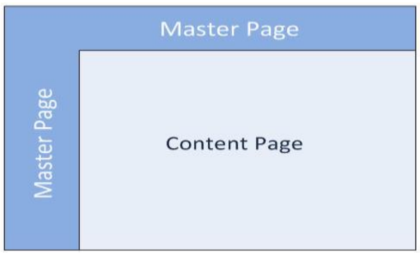


Figure 11: SharePoint Page Layout [12]

The elements such as company logo, side bars, search boxes, action menus, design and colors etc. are persistent when we navigate to the site in a company webpage. The master pages are used to define the layout, look and feel for the whole site collection pages. SharePoint shows the combination of the Master page elements and the content page elements in form of a responsive webpage when rendered through a browser. The content page part of the SharePoint page consists of web parts and content specific to the site page.

Content layout and placeholders-The content place holder help in defining the layout of the content page. The position and facility to add web parts to a page are provided by the content holder. There are by default 33 content holder in master page to define the functionalities of page in SharePoint.

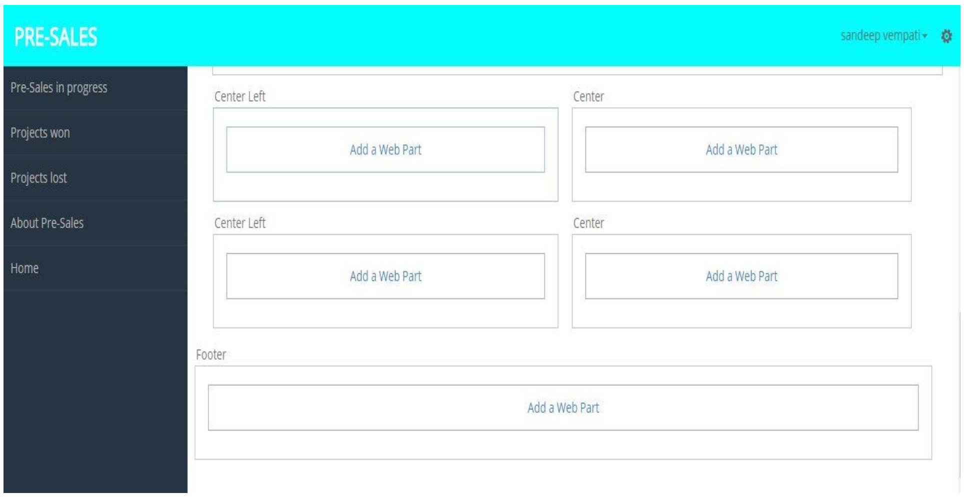


Figure 12: Content Place holders

The content place holder in the current project gives an idea of the layout for adding web parts to the master page.

Benefits of Master pages:

- Having a consistent view all through the site, this makes navigation easier.
- Look and feel of the website is consistent and easier to create new pages.
- Mitigates Design Issues
- Coding master pages makes job of a programmer easier.
- A power user can also create site pages form the existing master page.

4.3 Web Parts

The elements of the content page are known as web parts. Generally, there are inbuilt web parts in SharePoint. The user can add different web parts to the content page just by clicking on the content place holder and adding the web part or by adding the default web parts provided by the SharePoint.

Few commonly used In-built web parts provided by SharePoint are

- Lists
- Content Editors
- Image
- Page viewer web part
- Script editor

To add a web part to the content page. The following steps can be followed by the user:

1. Click on the settings icon on the right side of the SharePoint page and select edit page.
2. After clicking on the edit page option, the web part zones in which the web part can be placed can be seen. Click on the link add web part and the select the type of web parts we wish to add.

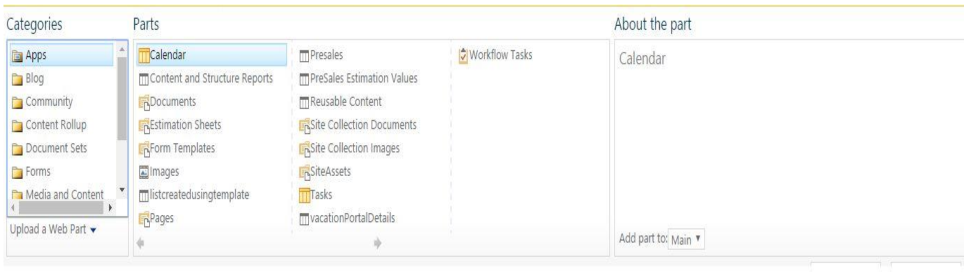


Figure 13: Add web parts window

4.4 Workflows

A series of tasks that produce an output related to a business process such as sequence of actions, automated exchange of items or documents is known as a workflow. SharePoint workflow are supported by Windows workflow Foundation (WF). The WF is designed and built based on the messaging functionality provided by Windows Communication Foundation (WCF). There are two types of workflows defined in a business environment [17].

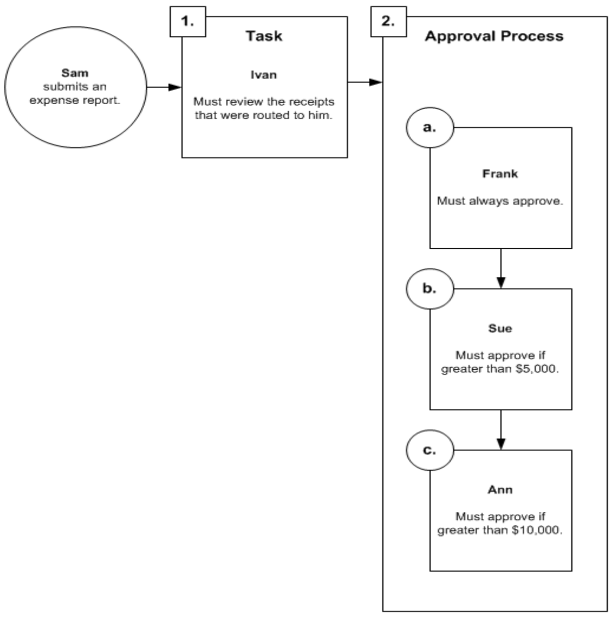

Sam submits an expense report.
1.
Task
Ivan
Must review the receipts that were routed to him.
2.
Approval Process
a.
Frank
Must always approve.
b.
Sue
Must approve if greater than $5,000.
c.
Ann
Must approve if greater than $10,000.

Figure 14: Workflow example [9]

- Sequential Workflow- Sequential workflow consist of series of steps that are performed in a standard order till the end of the process. All the steps in a sequential workflow are executed on after the other. Direction of execution order is towards the end of the process always no loops in the workflow process exist. Example of a sequential work flow is shown below [18].
- State Machine Workflow- State Machine Workflow has multiple steps and the order of the process is not specific. Any state can be the end state. The steps get executed asynchronously. Each process is independent to itself [18].

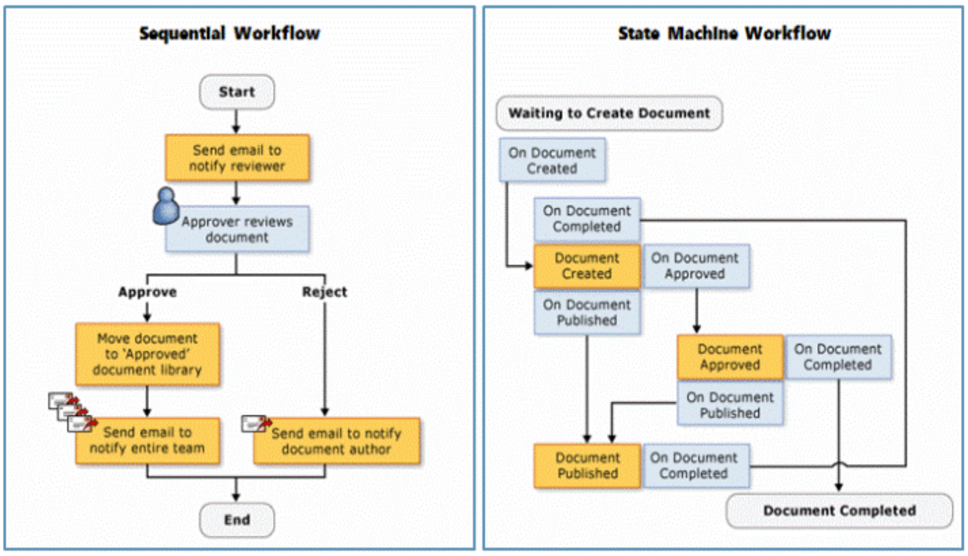


Figure 15: Types of Workflows [10]

Chapter 5

Implementation

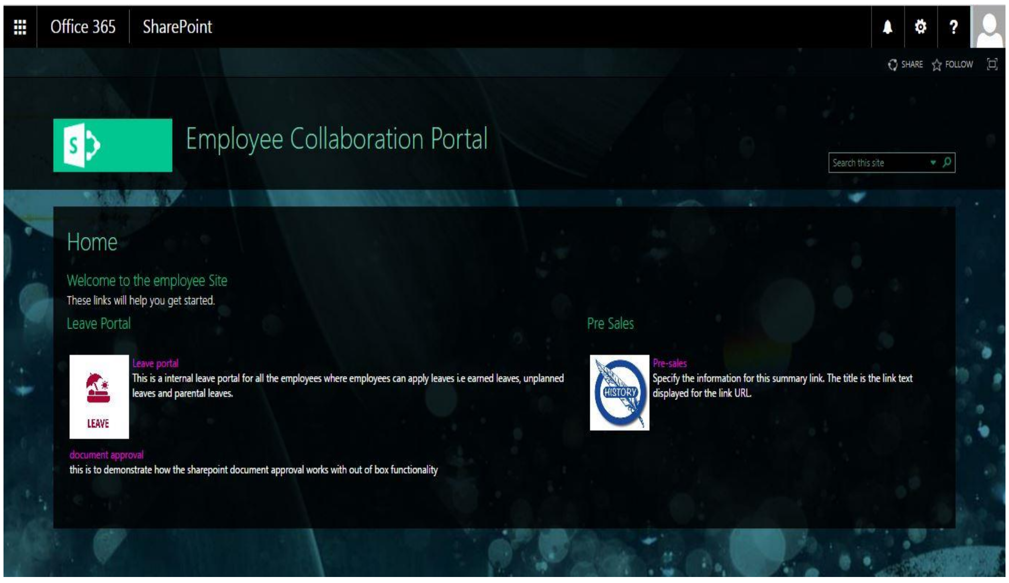
5.1 Employee Collaboration portal

The Employee Collaboration Portal is developed using SharePoint Online platform. The user accounts can be created using the SharePoint administrator account and the developer is by default the administrator (the administrator can give developer privileges to user too selectively). Every user is a considered to be an employee. The portal is targeted to all the employees of the company. The portal mainly has three functionalities –

- Presales Dashboard.
- Leave application management.
- Managed Document Sharing

Managed Document Sharing makes use of the in-built (out-of-box) SharePoint feature for document handling and sharing.

Figure 16 shows the home portal after an end user logs in with an Office 365 credential.



Home

Welcome to the employee Site

These links will help you get started.

Leave Portal

Leave portal

This is a internal leave portal for all the employees where employees can apply leaves i.e earned leaves, unplanned leaves and parental leaves.

document approval

this is to demonstrate how the sharepoint document approval works with out of box functionality

Pre Sales

Pre-sales

Specify the information for this summary link. The title is the link text displayed for the link URL.

Figure 16: Home Page

As we can see in figure 16, the tiles guide the users for accessing the functionalities. The employee can choose either of the options Leave portal and Presales.

Figure 17 shows the topology of the sites in the portal that is developed.

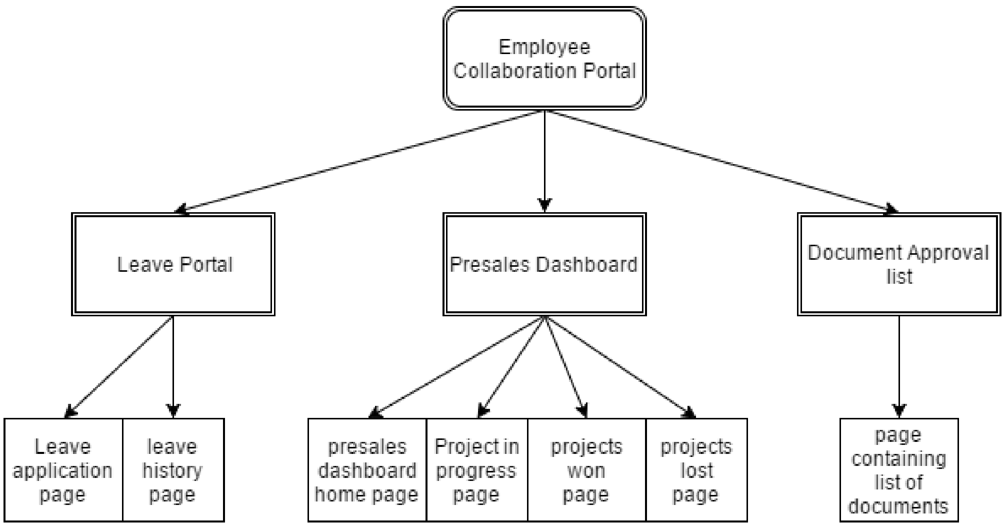
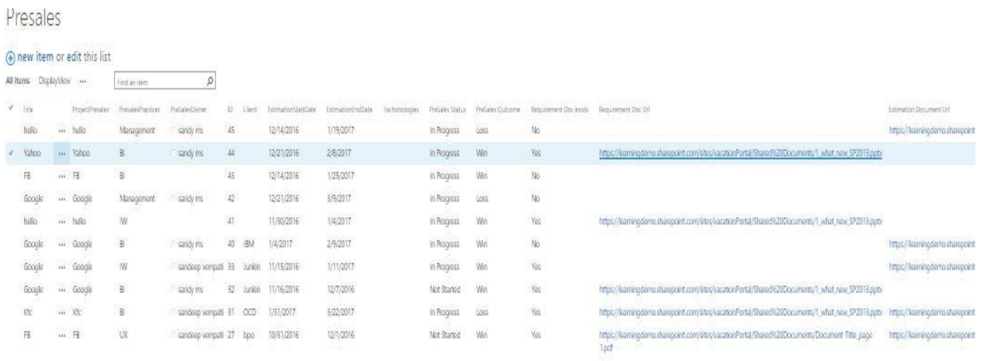


Figure 17: Site Topology

Site collections- There are fundamentally three site collections in this project. They are presales and leave application and company portal. The subpages under each site collection can be seen as shown in figure 17. Each collection has different master pages.

Lists- Few common types of lists created include calendar lists, contact lists, issue tracking, custom lists. All the lists used in the project are custom generated lists. The lists created in development of the company internal portal are:

<u>Presales List</u>: This list stores the project details and all the web parts in the presales dashboard use the data from this list to display content. The web parts that use the presales list include a presales graph, recent wins, a presales health check and project details.


Presales

new item or edit this list

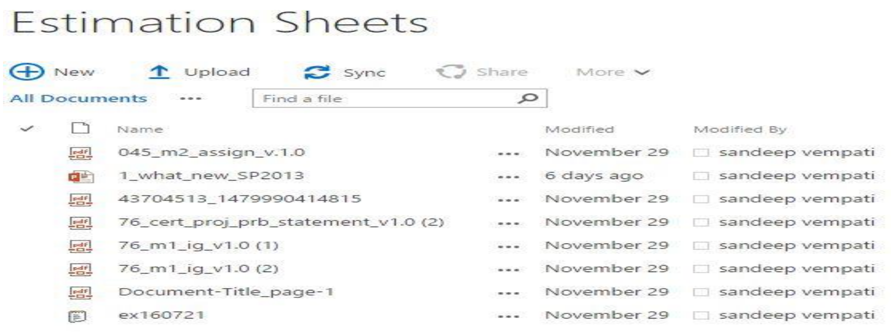
All Items DisplayView ··· Find an item

✓	Title		ProjectPresales	PresalesPractices	PreSalesOwner	ID	Client	EstimationStartDate	EstimationEndDate	Technologies	PreSales Status	PreSales Outcome	Requirement Doc exists	Requirement Doc Url	Estimation Document Url
	hello	···	hello	Management	sandy ms	45		12/14/2016	1/19/2017		In Progress	Loss	No		https://learningdemo.sharepoint
✓	Yahoo	···	Yahoo	BI	sandy ms	44		12/21/2016	2/8/2017		In Progress	Win	Yes	https://learningdemo.sharepoint.com/sites/vacationPortal/Shared%20Documents/1_what_new_SP2013.ppt	
	FB	···	FB	BI		43		12/14/2016	1/25/2017		In Progress	Win	No		
	Google	···	Google	Management	sandy ms	42		12/21/2016	3/9/2017		In Progress	Loss	No		
	hello	···	hello	IW		41		11/30/2016	1/4/2017		In Progress	Win	Yes	https://learningdemo.sharepoint.com/sites/vacationPortal/Shared%20Documents/1_what_new_SP2013.ppt	
	Google	···	Google	BI	sandy ms	40	IBM	1/4/2017	2/9/2017		In Progress	Win	No		https://learningdemo.sharepoint
	Google	···	Google	IW	sandeep vempati	33	Junkin	11/15/2016	1/11/2017		In Progress	Win	Yes		https://learningdemo.sharepoint
	Google	···	Google	BI	sandy ms	32	Junkin	11/16/2016	12/7/2016		Not Started	Win	Yes	https://learningdemo.sharepoint.com/sites/vacationPortal/Shared%20Documents/1_what_new_SP2013.ppt	
	Kfc	···	Kfc	BI	sandeep vempati	31	OCD	1/31/2017	3/22/2017		In Progress	Loss	Yes	https://learningdemo.sharepoint.com/sites/vacationPortal/Shared%20Documents/1_what_new_SP2013.ppt	https://learningdemo.sharepoint
	FB	···	FB	UX	sandeep vempati	27	bpo	10/31/2016	12/1/2016		Not Started	Win	Yes	https://learningdemo.sharepoint.com/sites/vacationPortal/Shared%20Documents/Document Title_page 1.pdf	https://learningdemo.sharepoint

Figure 18: Presales List

Presales Estimation Values List: This list stores the number of estimated hours and actual hours entered for every project.

Presales Estimation Sheets List: This list stores the uploaded estimation sheet and requirement document for every project.


Estimation Sheets

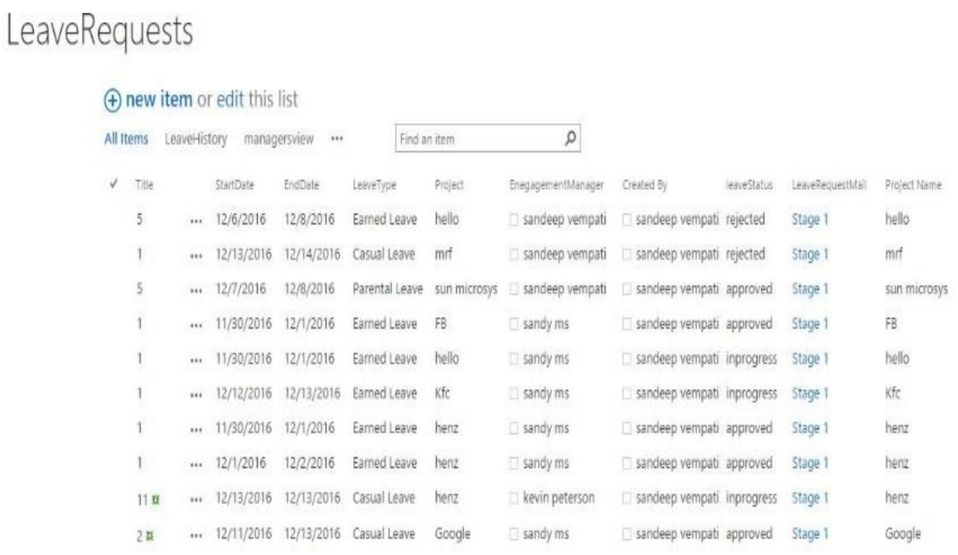
New Upload Sync Share More

All Documents ··· Find a file

✓		Name		Modified	Modified By
		045_m2_assign_v.1.0	···	November 29	sandeep vempati
		1_what_new_SP2013	···	6 days ago	sandeep vempati
		43704513_1479990414815	···	November 29	sandeep vempati
		76_cert_proj_prb_statement_v1.0 (2)	···	November 29	sandeep vempati
		76_m1_ig_v1.0 (1)	···	November 29	sandeep vempati
		76_m1_ig_v1.0 (2)	···	November 29	sandeep vempati
		Document-Title_page-1	···	November 29	sandeep vempati
		ex160721	···	November 29	sandeep vempati

Figure 19: Estimation Sheets List

Leave Request List: Leave request list is created to store data entered by the employee when a leave is applied and the workflow for the leave approval process is created on this list itself. A workflow can either be created on a list, a content type or a calendar list.



LeaveRequests

⊕ new item or edit this list

All Items   LeaveHistory   managersview   •••   Find an item

✓ Title		StartDate	EndDate	LeaveType	Project	EnegagementManager	Created By	leaveStatus	LeaveRequestMail	Project Name
5	•••	12/6/2016	12/8/2016	Earned Leave	hello	sandeep vempati	sandeep vempati	rejected	Stage 1	hello
1	•••	12/13/2016	12/14/2016	Casual Leave	mrf	sandeep vempati	sandeep vempati	rejected	Stage 1	mrf
5	•••	12/7/2016	12/8/2016	Parental Leave	sun microsys	sandeep vempati	sandeep vempati	approved	Stage 1	sun microsys
1	•••	11/30/2016	12/1/2016	Earned Leave	FB	sandy ms	sandeep vempati	approved	Stage 1	FB
1	•••	11/30/2016	12/1/2016	Earned Leave	hello	sandy ms	sandeep vempati	inprogress	Stage 1	hello
1	•••	12/12/2016	12/13/2016	Earned Leave	Kfc	sandy ms	sandeep vempati	inprogress	Stage 1	Kfc
1	•••	11/30/2016	12/1/2016	Earned Leave	henz	sandy ms	sandeep vempati	approved	Stage 1	henz
1	•••	12/1/2016	12/2/2016	Earned Leave	henz	sandy ms	sandeep vempati	approved	Stage 1	henz
11 ✱	•••	12/13/2016	12/13/2016	Casual Leave	henz	kevin peterson	sandeep vempati	inprogress	Stage 1	henz
2 ✱	•••	12/11/2016	12/13/2016	Casual Leave	Google	sandy ms	sandeep vempati	approved	Stage 1	Google

Figure 20: Leave Requests List

5.2 Use-case Diagrams

The Leave portal web application use case diagram:

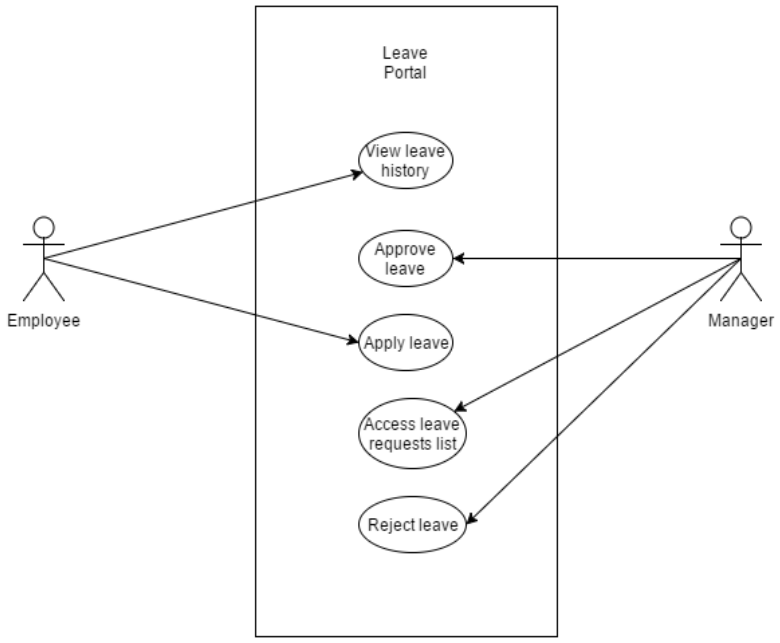


Figure 21: Use Case diagram for leave portal

Use case diagram for Presales web application**:**

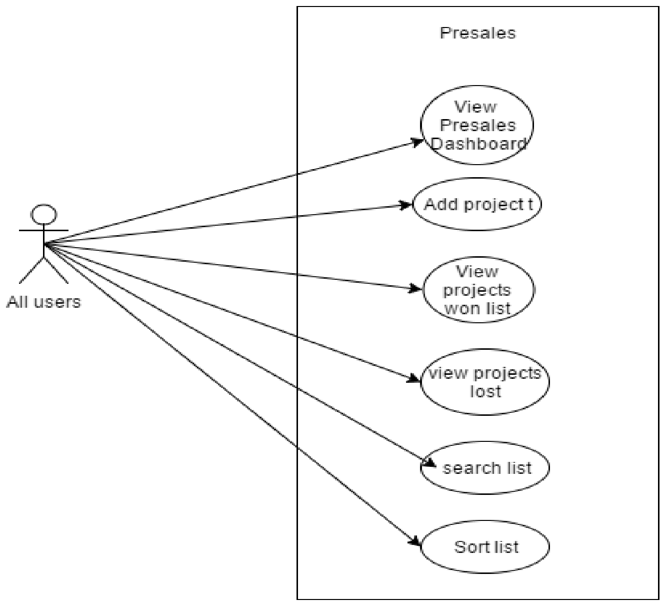


Figure 22: Use Case diagram for Presales

5.3 Leave Portal

Leave portal is a sub-site collection in the company portal. The tile that shows leave portal in the homepage enables the employee to apply for a leave by filling the details and sending a leave approval request to their concerned manger.

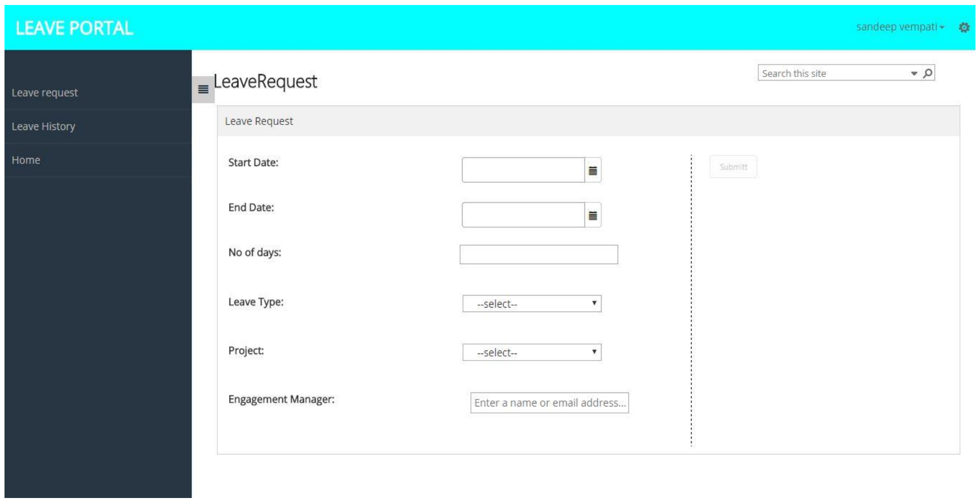
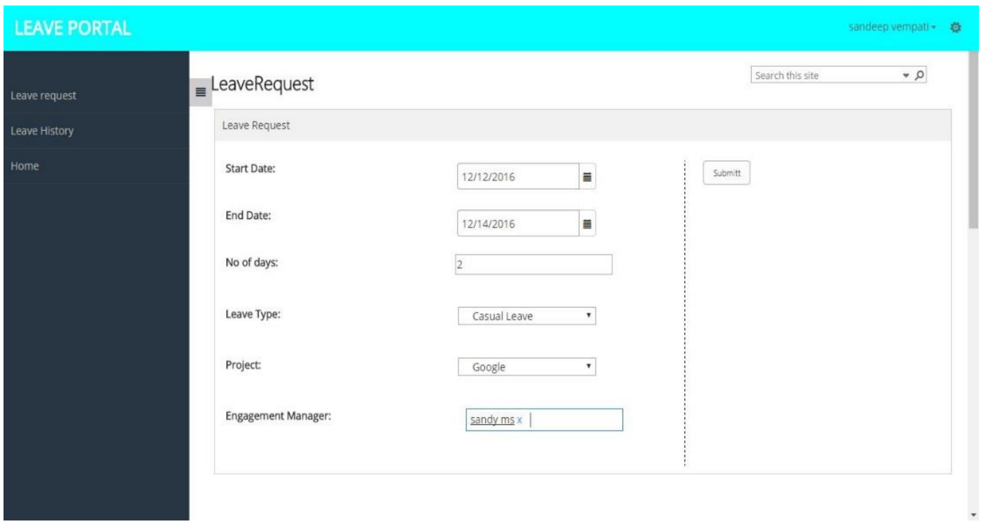


Figure 23: Leave Request Homepage

Figure 23 shows the homepage of the leave application portal which consist of a form where the employee can enter leave request details. The start date and end date fields can be filled by selecting the date picker that shows up on clicking the calendar button on the field. Number of days can be entered for the leave duration. The type of leave can be selected by using the dropdown list that has three options namely- casual leave, earned leave, parental leave. The project in which the employee works can be selected using the dropdown list and the Engagement Manager’s name is entered in the text box provided. The Engagement Manager name gets searched and prompts are shown whenever the employee starts typing by matching the name of users and the text typed.

After filling the data in the form the employee can click on submit button for sending an approval request to their concerned manger. As soon as the employee clicks on the submit

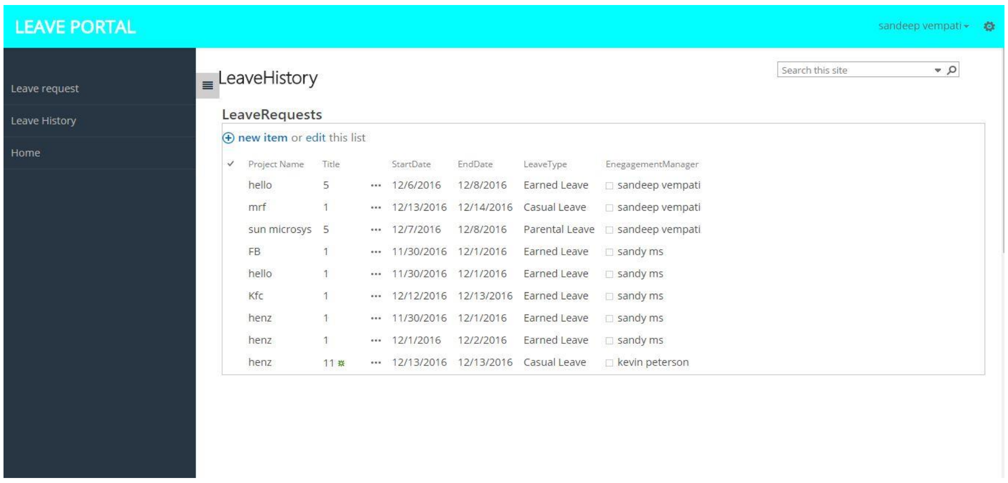


Leave Request

Start Date:	12/12/2016
End Date:	12/14/2016
No of days:	2
Leave Type:	Casual Leave
Project:	Google
Engagement Manager:	sandy ms ×

Figure 24: Leave Request Homepage

button the Leave History of the employees gets displayed. The user/employee can anytime click on the pane on the left side to access the leave history.



LeaveHistory

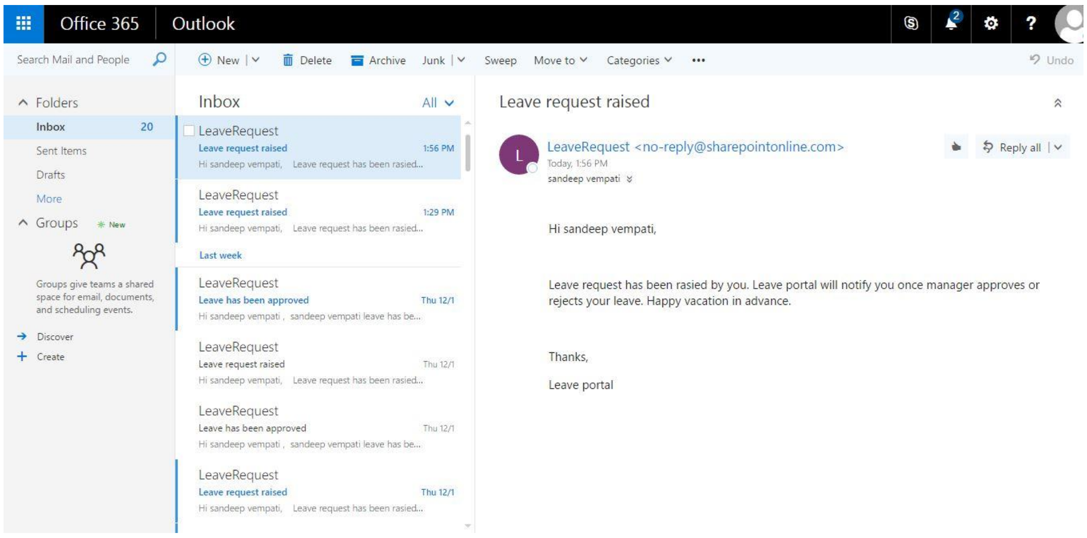
LeaveRequests

new item or edit this list

✓	Project Name	Title		StartDate	EndDate	LeaveType	EnegagementManager
	hello	5	•••	12/6/2016	12/8/2016	Earned Leave	sandeep vempati
	mrf	1	•••	12/13/2016	12/14/2016	Casual Leave	sandeep vempati
	sun microsys	5	•••	12/7/2016	12/8/2016	Parental Leave	sandeep vempati
	FB	1	•••	11/30/2016	12/1/2016	Earned Leave	sandy ms
	hello	1	•••	11/30/2016	12/1/2016	Earned Leave	sandy ms
	Kfc	1	•••	12/12/2016	12/13/2016	Earned Leave	sandy ms
	henz	1	•••	11/30/2016	12/1/2016	Earned Leave	sandy ms
	henz	1	•••	12/1/2016	12/2/2016	Earned Leave	sandy ms
	henz	11 ✳	•••	12/13/2016	12/13/2016	Casual Leave	kevin peterson

Figure 25: Leave History page

The list of leaves the employee has applied will be displayed in the Leave Requests list, as shown in the figure 25. After the employees places a leave request a workflow process gets initiated and the engagement manger gets approval request email and the user gets notified about the current stage of the approval as shown below.



LeaveRequest
Leave request raised 1:56 PM
Hi sandeep vempati, Leave request has been rasied...

LeaveRequest
Leave request raised 1:29 PM
Hi sandeep vempati, Leave request has been rasied...

Last week

LeaveRequest
Leave has been approved Thu 12/1
Hi sandeep vempati , sandeep vempati leave has be...

LeaveRequest
Leave request raised Thu 12/1
Hi sandeep vempati, Leave request has been rasied...

LeaveRequest
Leave has been approved Thu 12/1
Hi sandeep vempati , sandeep vempati leave has be...

LeaveRequest
Leave request raised Thu 12/1
Hi sandeep vempati, Leave request has been rasied...

Leave request raised

LeaveRequest <no-reply@sharepointonline.com>
Today, 1:56 PM
sandeep vempati

Hi sandeep vempati,

Leave request has been rasied by you. Leave portal will notify you once manager approves or rejects your leave. Happy vacation in advance.

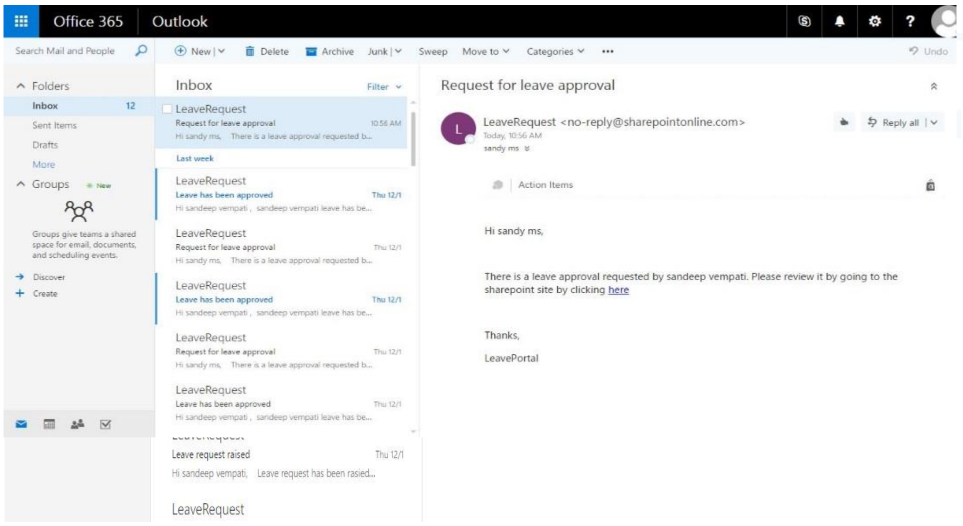
Thanks,

Leave portal

Figure 26: Workflow Email notification.

By this, the employee's application process ends, after the leave gets approved a notification email that encloses the response of the manager is received.

At the manager's end, the leave request email is received and can set the status of the leave request to approved /declined. The request email received by the manager is shown here.



LeaveRequest
Request for leave approval 10:56 AM
Hi sandy ms, There is a leave approval requested b...

LeaveRequest
Leave has been approved Thu 12/1
Hi sandeep vempati , sandeep vempati leave has be...

LeaveRequest
Request for leave approval Thu 12/1
Hi sandy ms, There is a leave approval requested b...

LeaveRequest
Leave has been approved Thu 12/1
Hi sandeep vempati , sandeep vempati leave has be...

LeaveRequest
Request for leave approval Thu 12/1
Hi sandy ms, There is a leave approval requested b...

LeaveRequest
Leave has been approved Thu 12/1
Hi sandeep vempati , sandeep vempati leave has be...

Leave request raised Thu 12/1
Hi sandeep vempati, Leave request has been rasied...

LeaveRequest

Request for leave approval

LeaveRequest <no-reply@sharepointonline.com>
Today, 10:56 AM
sandy ms

Hi sandy ms,

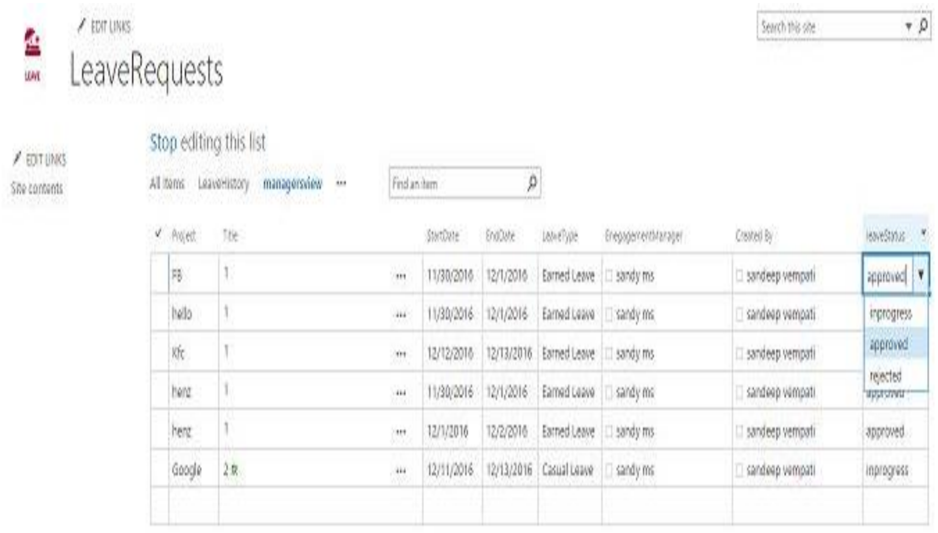
There is a leave approval requested by sandeep vempati. Please review it by going to the sharepoint site by clicking here

Thanks,

LeavePortal

Figure 27: Approval Email notification.

By clicking the link in the mail the manager can view the request and review it.

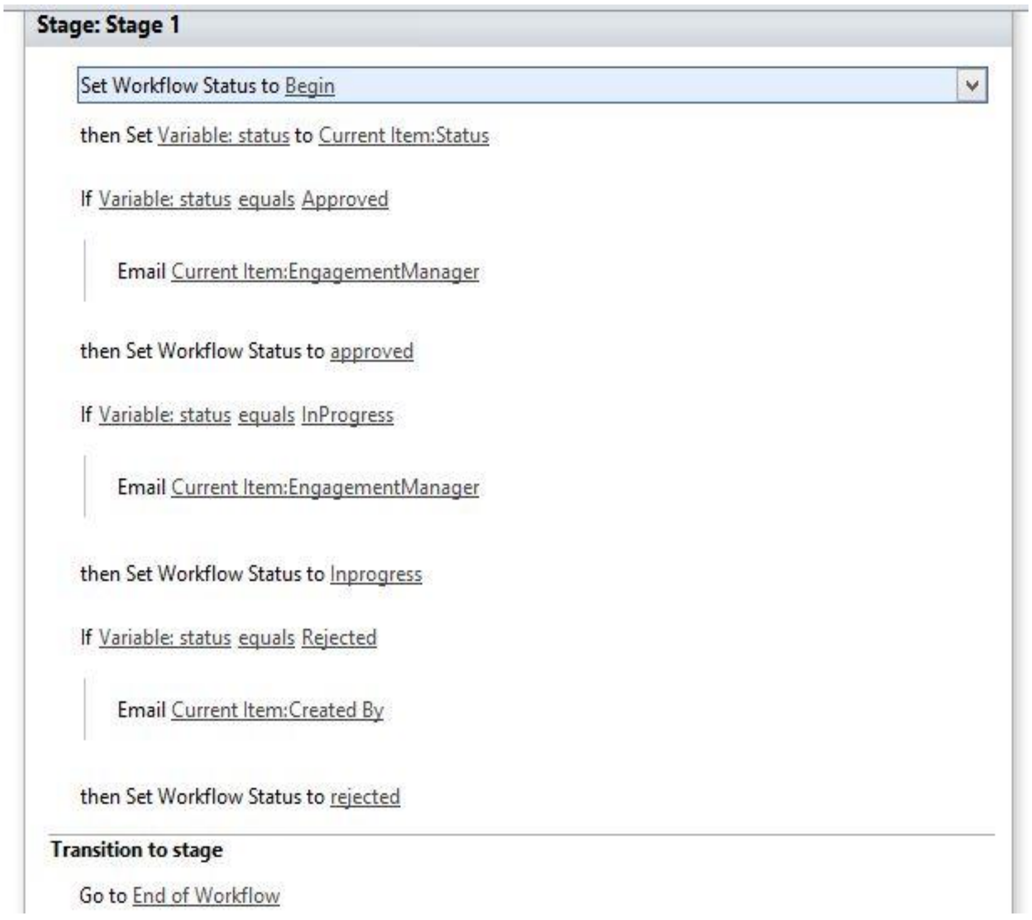


Project	Title		StartDate	EndDate	LeaveType	EngagementManager	Created By	leaveStatus
FB	1	...	11/30/2016	12/1/2016	Earned Leave	sandy ms	sandeep vempati	approved
hello	1	...	11/30/2016	12/1/2016	Earned Leave	sandy ms	sandeep vempati	[illegible]
Kfc	1	...	12/12/2016	12/13/2016	Earned Leave	sandy ms	sandeep vempati	[illegible]
henz	1	...	11/30/2016	12/1/2016	Earned Leave	sandy ms	sandeep vempati	approved
henz	1	...	12/1/2016	12/2/2016	Earned Leave	sandy ms	sandeep vempati	approved
Google	2	...	12/11/2016	12/13/2016	Casual Leave	sandy ms	sandeep vempati	inprogress

Figure 28: Leave Request List

The manager can select the option by editing the 'LeaveStatus' in the 'LeaveRequests' list as shown in figure 28. When the manager approves the leave an e-mail is sent automatically to the user as shown here. The details the user enters in the leave request page gets saved in the list- 'LeaveRequests' in backend as show in this screenshot.

The workflow for the leave request portal that helps in streamlining the process of approval is developed using the SharePoint designer. The workflow is created on the list that stores the data about the leave requests in the backend.



```
Stage: Stage 1
    Set Workflow Status to Begin
    then Set Variable: status to Current Item:Status
    If Variable: status equals Approved
        Email Current Item:EngagementManager
    then Set Workflow Status to approved
    If Variable: status equals InProgress
        Email Current Item:EngagementManager
    then Set Workflow Status to Inprogress
    If Variable: status equals Rejected
        Email Current Item:Created By
    then Set Workflow Status to rejected
Transition to stage
    Go to End of Workflow
```

Figure 29: Developed Workflow

5.4 Presales Dashboard

Presales dashboard has mainly three pages –

- The dashboard page
- The page that displays the number list of all projects
- The pages that display projects based on the status of the projects

List of pages for navigating in Presales site collection are –

- Presales Dashboard home
- All projects page
- Projects won
- Projects in progress
- Projects lost

The dashboard homepage looks as shown in figure 30:

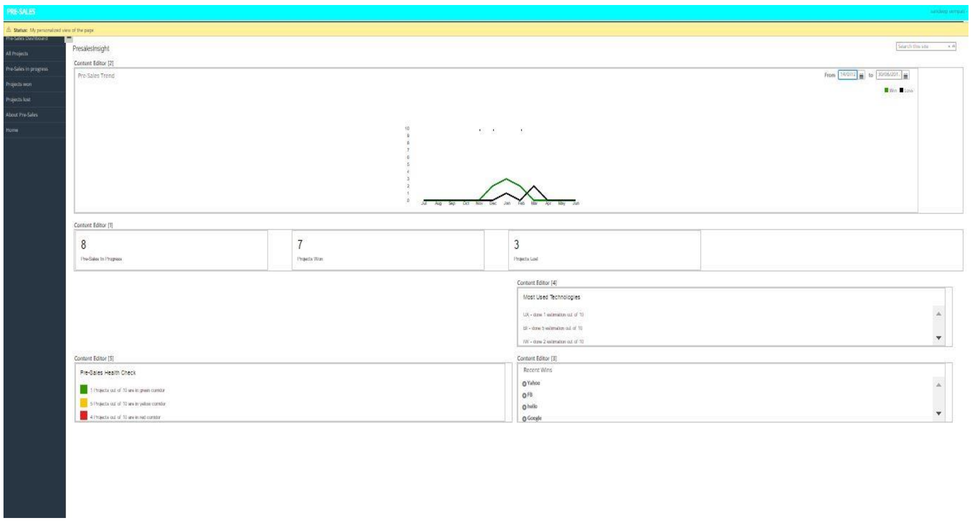


Figure 30: Presales Dashboard Homepage

The dashboard home has multiple web parts that get displayed based in the projects entered by the employee. Web parts in the dashboard are:

- Presales graph
- Web part that displays number of projects won, lost, in progress.
- Presales health check web part
- Most used technologies web part
- Recent wins web part.

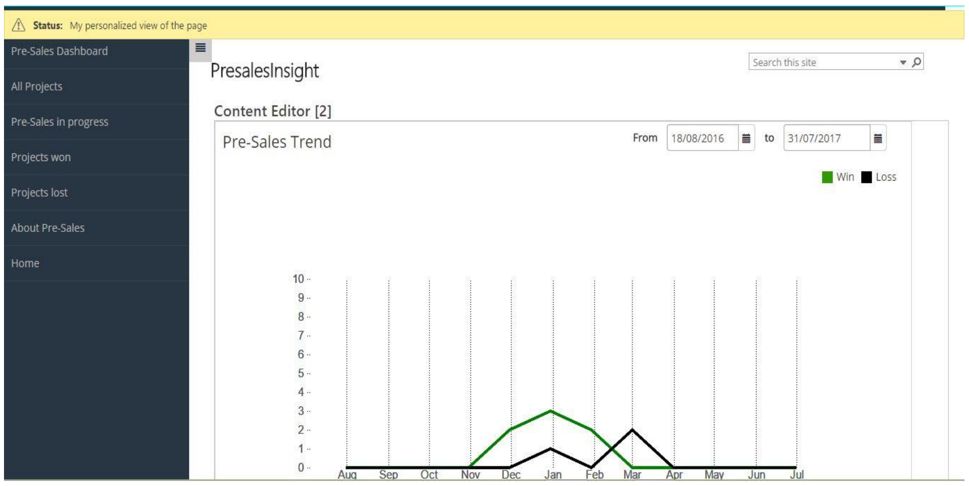
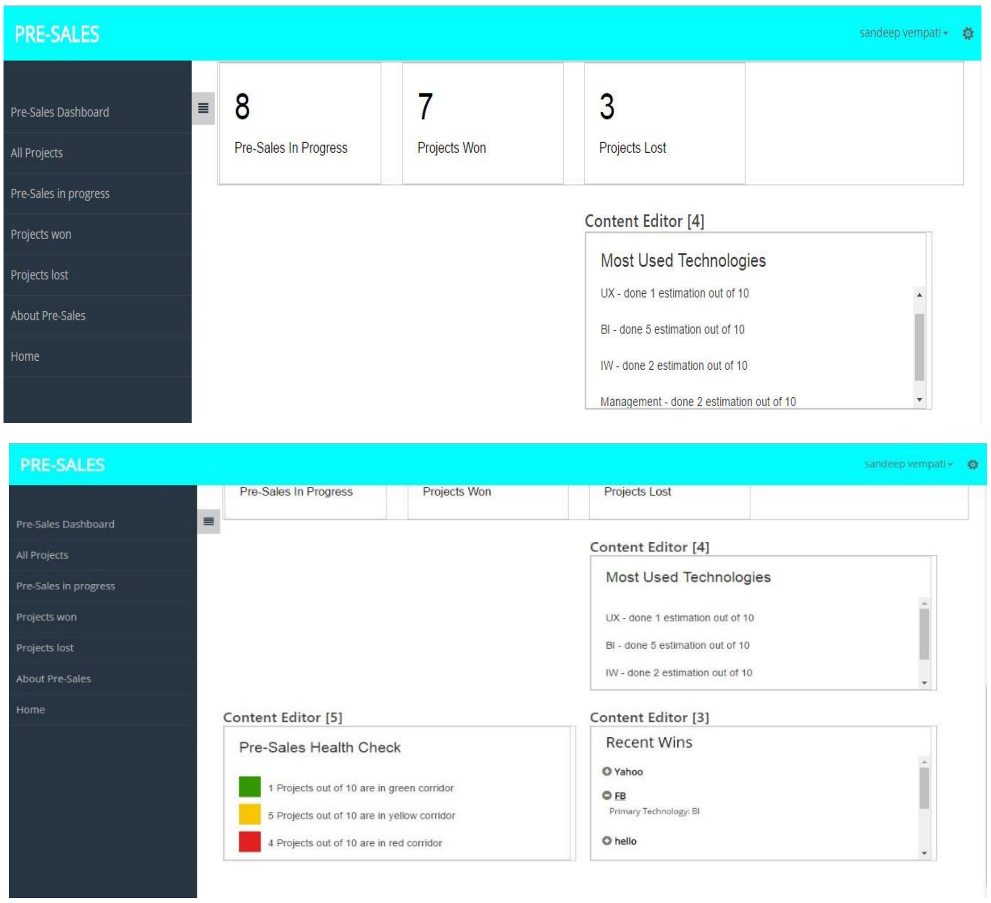


Figure 31: Presales Trend Graph



PRE-SALES

sandeep vempati

- Pre-Sales Dashboard
- All Projects
- Pre-Sales in progress
- Projects won
- Projects lost
- About Pre-Sales
- Home

8 Pre-Sales In Progress

7 Projects Won

3 Projects Lost

Content Editor [4]

Most Used Technologies

UX - done 1 estimation out of 10

BI - done 5 estimation out of 10

IW - done 2 estimation out of 10

Management - done 2 estimation out of 10

PRE-SALES

sandeep vempati

- Pre-Sales Dashboard
- All Projects
- Pre-Sales in progress
- Projects won
- Projects lost
- About Pre-Sales
- Home

Pre-Sales In Progress

Projects Won

Projects Lost

Content Editor [4]

Most Used Technologies

UX - done 1 estimation out of 10

BI - done 5 estimation out of 10

IW - done 2 estimation out of 10

Content Editor [5]

Pre-Sales Health Check

🟩 1 Projects out of 10 are in green corridor

🟨 5 Projects out of 10 are in yellow corridor

🟥 4 Projects out of 10 are in red corridor

Content Editor [3]

Recent Wins

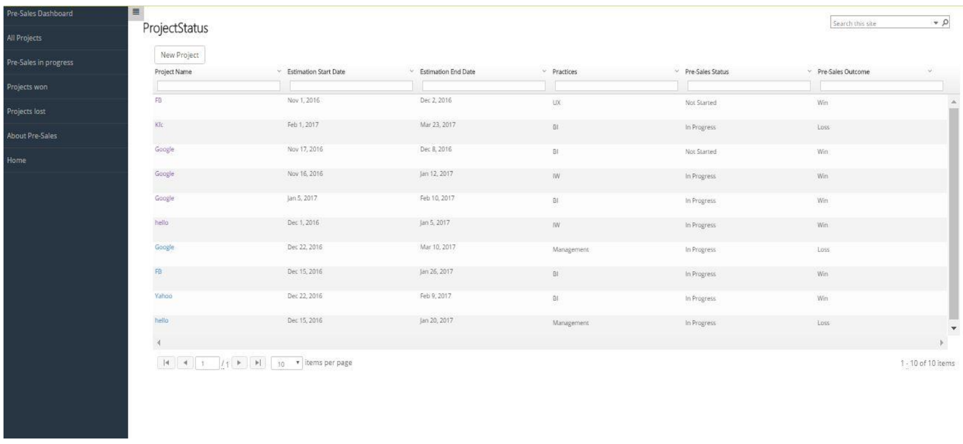
Yahoo

FB
Primary Technology: BI

hello

Figure 32: Presales Dashboard widgets

The **All Projects** page displays the projects that are undertaken by the company previously and an option for adding new projects as shown in figure 33.



- Pre-Sales Dashboard
- All Projects
- Pre-Sales in progress
- Projects won
- Projects lost
- About Pre-Sales
- Home

ProjectStatus

Search this site

New Project

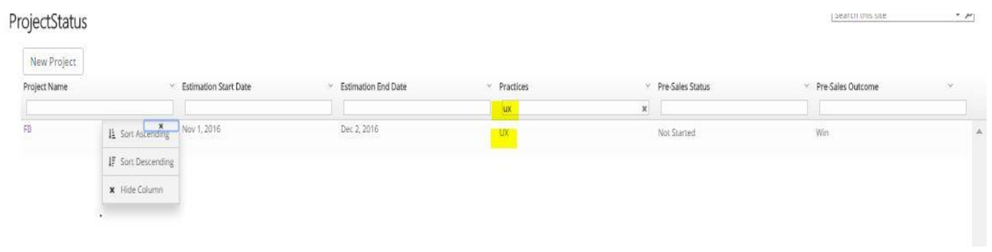
Project Name	Estimation Start Date	Estimation End Date	Practices	Pre-Sales Status	Pre-Sales Outcome
FB	Nov 1, 2016	Dec 2, 2016	UX	Not Started	Win
Kfc	Feb 1, 2017	Mar 23, 2017	BI	In Progress	Loss
Google	Nov 17, 2016	Dec 8, 2016	BI	Not Started	Win
Google	Nov 16, 2016	Jan 12, 2017	IW	In Progress	Win
Google	Jan 5, 2017	Feb 10, 2017	BI	In Progress	Win
hello	Dec 1, 2016	Jan 5, 2017	IW	In Progress	Win
Google	Dec 22, 2016	Mar 10, 2017	Management	In Progress	Loss
FB	Dec 15, 2016	Jan 26, 2017	BI	In Progress	Win
Yahoo	Dec 22, 2016	Feb 9, 2017	BI	In Progress	Win
hello	Dec 15, 2016	Jan 20, 2017	Management	In Progress	Loss

1 / 1 10 items per page

1 - 10 of 10 items

Figure 33: All Projects Page

There is an option for search below the column name to search for particular results. Additionally, the list can be sorted.



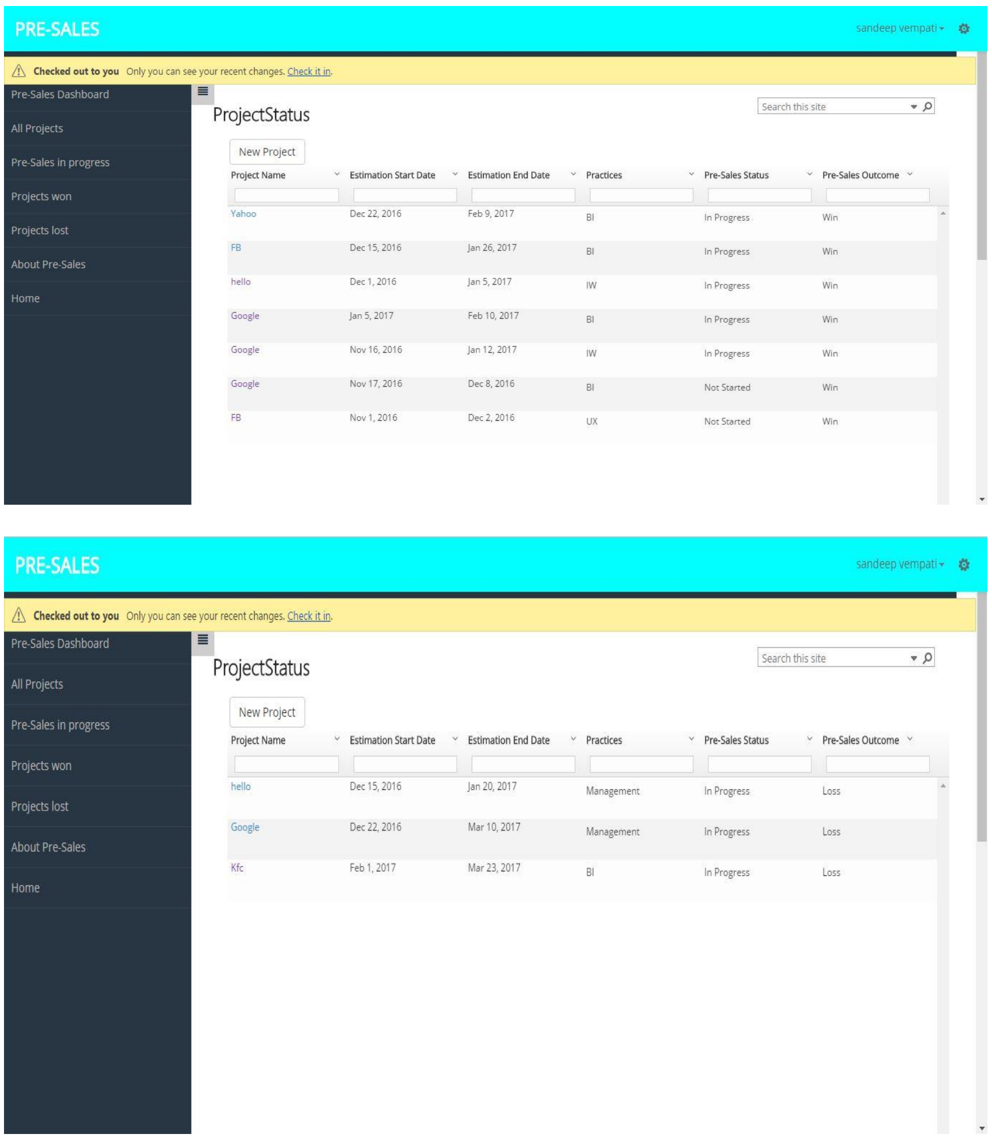
ProjectStatus

Search this site

New Project

Project Name	Estimation Start Date	Estimation End Date	Practices	Pre-Sales Status	Pre-Sales Outcome
			UX		
FB	Nov 1, 2016	Dec 2, 2016	UX	Not Started	Win

There are links added on the toggle bar pages that display only projects won, projects lost and homepage (company portal start page).



PRE-SALES

sandeep vempati

Checked out to you Only you can see your recent changes. Check it in.

ProjectStatus

Search this site

New Project

Project Name	Estimation Start Date	Estimation End Date	Practices	Pre-Sales Status	Pre-Sales Outcome
Yahoo	Dec 22, 2016	Feb 9, 2017	BI	In Progress	Win
FB	Dec 15, 2016	Jan 26, 2017	BI	In Progress	Win
hello	Dec 1, 2016	Jan 5, 2017	IW	In Progress	Win
Google	Jan 5, 2017	Feb 10, 2017	BI	In Progress	Win
Google	Nov 16, 2016	Jan 12, 2017	IW	In Progress	Win
Google	Nov 17, 2016	Dec 8, 2016	BI	Not Started	Win
FB	Nov 1, 2016	Dec 2, 2016	UX	Not Started	Win

PRE-SALES

sandeep vempati

Checked out to you Only you can see your recent changes. Check it in.

ProjectStatus

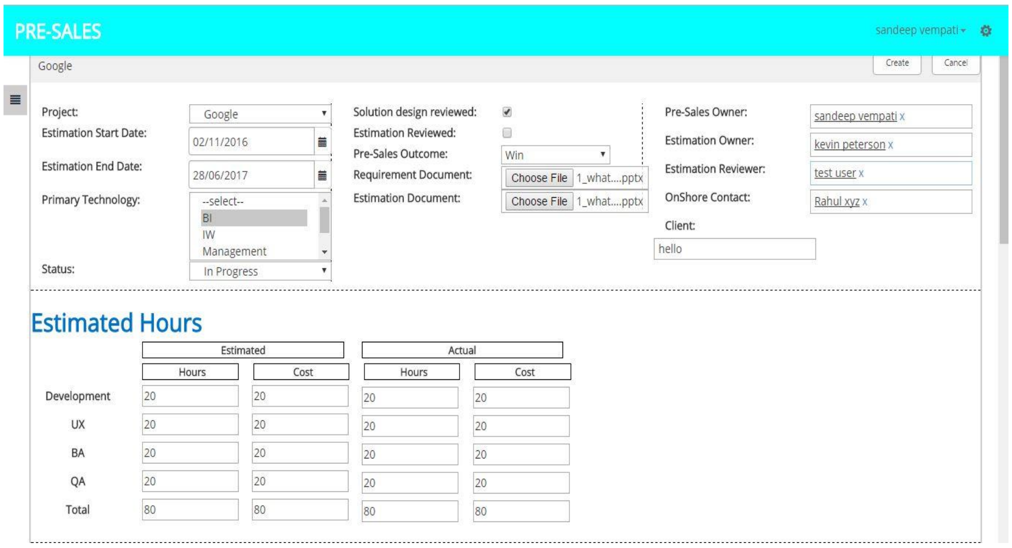
Search this site

New Project

Project Name	Estimation Start Date	Estimation End Date	Practices	Pre-Sales Status	Pre-Sales Outcome
hello	Dec 15, 2016	Jan 20, 2017	Management	In Progress	Loss
Google	Dec 22, 2016	Mar 10, 2017	Management	In Progress	Loss
Kfc	Feb 1, 2017	Mar 23, 2017	BI	In Progress	Loss

Figure 34: Projects Won and Projects Lost

On clicking the new project button the web part opens a form for entering project information and the information entered gets saved into the 'Presales' list in the backend and from that list the view web part displays the list of projects on the screen. Multiple options guide the user to enter project details.



Project:	Google
Estimation Start Date:	02/11/2016
Estimation End Date:	28/06/2017
Primary Technology:	--select-- / BI / IW / Management
Status:	In Progress
Solution design reviewed:	✓
Estimation Reviewed:	☐
Pre-Sales Outcome:	Win
Requirement Document:	Choose File 1_what....pptx
Estimation Document:	Choose File 1_what....pptx
Pre-Sales Owner:	sandeep vempati x
Estimation Owner:	kevin peterson x
Estimation Reviewer:	test user x
OnShore Contact:	Rahul xyz x
Client:	hello

Estimated Hours

	Estimated		Actual	
	Hours	Cost	Hours	Cost
Development	20	20	20	20
UX	20	20	20	20
BA	20	20	20	20
QA	20	20	20	20
Total	80	80	80	80

Figure 35: Add project page

Chapter 6

Appendix

6.1 Master page code

The master page is created by making some significant changes to the default master page seattle.master. Code added to the seattle master page is shown below.

```
<SharePoint:Scriptlink runat="server" Name="~site/SiteAssets/Presales/JS/ui-grid/ui-grid.js" Language="javascript"/>
<SharePoint:Scriptlink runat="server" Name="~site/SiteAssets/Presales/App/presalesApp.js"
Language="javascript"/>
<SharePoint:Scriptlink runat="server" Name="~site/SiteAssets/Presales/JS/app.js" Language="javascript"/>
<SharePoint:CssRegistration name="<%$SPUrl:~site/SiteAssets/Presales/js/ui-grid/ui-grid.css%>" runat="server" after="SharepointCssFile" />
 <SharePoint:SharePointForm runat="server" onsubmit="if (typeof(_spFormOnSubmitWrapper) != 'undefined') {return _spFormOnSubmitWrapper();} else {return true;}">
   <script type="text/javascript"> var submitHook = function () { return false; }; theFormtheForm._spOldSubmit = theForm.submit; theForm.submit = function () { if (!submitHook()) { this._spOldSubmit(); } }; </script>
   <SharePoint:AjaxDelta id="DeltaSPWebPartManager" runat="server">
      <WebPartPages:SPWebPartManager runat="Server"/>
   </SharePoint:AjaxDelta>
   <asp:ScriptManager id="ScriptManager" runat="server" EnablePageMethods="false" EnablePartialRendering="true" EnableScriptGlobalization="false" EnableScriptLocalization="true" />
   <SharePoint:AjaxDelta id="DeltaDelegateControls" runat="server">
      <SharePoint:DelegateControl runat="server" ControlId="GlobalNavigation" />
      <SharePoint:DelegateControl ControlId="GlobalSiteLink3" Scope="Farm" runat="server" Visible="false" />
   </SharePoint:AjaxDelta>
<div id="TurnOnAccessibility" style="display:none" class="s4-notdlg noindex">
   <a id="linkTurnOnAcc" href="#" class="ms-accessible ms-acc-button" onclick="SetIsAccessibilityFeatureEnabled(true);UpdateAccessibilityUI();document.getElementById('linkTurnOffAcc').focus();return false;">
   <SharePoint:EncodedLiteral runat="server" text="<%$Resources:wss,master_turnonaccessibility%>" EncodeMethod="HtmlEncode"/></a>
</div>
<div id="TurnOffAccessibility" style="display:none" class="s4-notdlg noindex">
   <a id="linkTurnOffAcc" href="#" class="ms-accessible ms-acc-button" onclick="SetIsAccessibilityFeatureEnabled(false);UpdateAccessibilityUI();document.getElementById('linkTurnOnAcc').focus();return false;">
   <SharePoint:EncodedLiteral runat="server" text="<%$Resources:wss,master_turnoffaccessibility%>" EncodeMethod="HtmlEncode"/></a>
</div>
<div class="s4-notdlg s4-skipribbonshortcut noindex">
   <a href="javascript:;" onclick="document.getElementById('startNavigation').focus();" class="ms-accessible ms-acc-button" accesskey="<%$Resources:wss,skipribbon_accesskey%>" runat="server">
   <SharePoint:EncodedLiteral runat="server" text="<%$Resources:wss,skipRibbonCommandsLink%>" EncodeMethod="HtmlEncode"/></a>
</div>
<div class="s4-notdlg noindex">
   <a href="javascript:;" onclick="document.getElementById('mainContent').focus();" class="ms-accessible ms-acc-button" runat="server">
```

```
    <SharePoint:EncodedLiteral runat="server" text="<%$Resources:wss,mainContentLink%>" EncodeMethod="HtmlEncode"/></a>
</div>
<div id="TurnOffAnimation" style="display:none;" class="s4-notdlg noindex">
    <a id="linkTurnOffAnimation" href="#" class="ms-accessible ms-acc-button" onclick="ToggleAnimationStatus();return false;">
    <SharePoint:EncodedLiteral runat="server" text="<%$Resources:wss,master_disableanimation%>" EncodeMethod="HtmlEncode"/></a>
</div>
<div id="TurnOnAnimation" style="display:none;" class="s4-notdlg noindex">
    <a id="linkTurnOnAnimation" href="#" class="ms-accessible ms-acc-button" onclick="ToggleAnimationStatus();return false;">
    <SharePoint:EncodedLiteral runat="server" text="<%$Resources:wss,master_enableanimation%>" EncodeMethod="HtmlEncode"/></a>
</div>

    <script type="text/javascript">
        function PresalesInsight() {
            window.location.href = _spPageContextInfo.webAbsoluteUrl + "/pages/PresalesInsight.aspx";
        }
        function ProjectStatus() {
            window.location.href = _spPageContextInfo.webAbsoluteUrl + "/pages/ProjectStatus.aspx?Status=";
        }
        function ProjectStatusInProgress() {
            window.location.href = _spPageContextInfo.webAbsoluteUrl + "/pages/ProjectStatus.aspx?Status=In Progress";
        }
        function ProjectStatusWin() {

            window.location.href = _spPageContextInfo.webAbsoluteUrl + "/pages/ProjectStatus.aspx?Status=Win";
        }
        function ProjectStatusLoss() {

            window.location.href = _spPageContextInfo.webAbsoluteUrl + "/pages/ProjectStatus.aspx?Status=Loss";
        }
        function AboutPresales() {
            window.location.href = _spPageContextInfo.webAbsoluteUrl + "/Pages/AboutPreSales.aspx";
        }
        function logoredirection() {
            window.location.href = _spPageContextInfo.webAbsoluteUrl + "/pages/PresalesInsight.aspx";
        }
        function homePage()
        {
        window.location.href ="https://learningdemo.sharepoint.com/sites/sales/Pages/default.aspx";
        }
</script>
<nav class="navbar-default navbar-side" role="navigation">
        <div id="sideNav" href=""><i class="fa fa-align-justify"></i></div>
            <div class="sidebar-collapse">
                <ul class="nav" id="main-menu">

                    <li>
                        <a class="navlinks " onclick="PresalesInsight()"> Pre-Sales Dashboard</a>
                    </li>
                    <li>
                        <a class="navlinks" onclick="ProjectStatus()"> All Projects</a>
                    </li>
                    <li>
                        <a class="navlinks" onclick="ProjectStatusInProgress()">Pre-Sales in progress</a>
                    </li>
                    <li>
                        <a class="navlinks" onclick="ProjectStatusWin()">Projects won</a>
                    </li>
                <li>
                    <a class="navlinks" onclick="ProjectStatusLoss()"> Projects lost</a>
                </li>
                <li>
                    <a class="navlinks" onclick="AboutPresales()"> About Pre-Sales</a>
                </li>
 <li>
                    <a class="navlinks" onclick="homePage()">Home</a>
                </li>
            </ul>

        </div>
```

6.2 Code for Presales Dashboard

Angular JS for Presales Graph web part:

```
/*global angular*/
(function () {
    'use strict';

    //Presales trend graph
    angular.module('preSalesApp').controller('PresalesTrendController', ['$scope', function ($scope) {

        var date = new Date();
        $scope.itemCol = [];
        $scope.data = [];
        $scope.sDate = new Date(date.getFullYear(), 0, 1);
        $scope.eDate = new Date(date.getFullYear(), 11, 31);
        $scope.sMaxDate = new Date(date.getFullYear(), 11, 31);
        $scope.eMinDate = new Date($scope.sDate.getFullYear(), $scope.sDate.getMonth(), $scope.sDate.getDate())
;
        var xLabelMax = new Date($scope.sDate.getFullYear() + 1, $scope.sDate.getMonth(), 0).getDate();
        $scope.eMaxDate = new Date($scope.sDate.getFullYear() + 1, $scope.sDate.getMonth() - 1, xLabelMax);

        $scope.startDateChange = function () {
            xLabelMax = new Date($scope.sDate.getFullYear() + 1, $scope.sDate.getMonth(), 0).getDate();
            $scope.eMinDate = new Date($scope.sDate.getFullYear(), $scope.sDate.getMonth(), $scope.sDate.getDate
());
            $scope.eMaxDate = new Date($scope.sDate.getFullYear() + 1, $scope.sDate.getMonth() - 1, xLabelMax);

            $scope.eDate = $scope.eMaxDate;
        };

        $scope.init = function () {
            var Context = SP.ClientContext.get_current();
            var web = Context.get_web();
            var oList = Context.get_web().get_lists().getByTitle('Presales');
            var camlQuery = new SP.CamlQuery();
            camlQuery.set_viewXml("<View><RowLimit>100</RowLimit><Query><OrderBy><FieldRef Name='ID
' Ascending='FALSE' /></OrderBy></Query></View>");
            var collImages = oList.getItems(camlQuery);
            Context.load(oList);
            Context.load(collImages, 'Include(PreSales_x0020_Outcome,EstimationEndDate,Solution_x0020_Design_
x0020_Url,Solution_x0020_Design_x0020_Revi,Estimation_x0020_Document_x0020_,Estimation_x0020_Revie
wed,Requirement_x0020_Doc_x0020_Url.Include(Url,Description),Requirement_x0020_Doc_x0020_exis,Estima
tionEndDate)');
            Context.executeQueryAsync(function () {
                var imagesEnumerator = collImages.getEnumerator();
                while (imagesEnumerator.moveNext()) {
                    var oImageItem = imagesEnumerator.get_current();
                    $scope.itemCol.push(oImageItem.get_fieldValues());
                }
                $scope.data = $scope.itemCol;
                $scope.$apply();
            }, function (sender, args) { alert(args.get_message()); });
        };

        SP.SOD.executeFunc('sp.js', 'SP.ClientContext', function () { $scope.init() });

        $scope.sDateOpen = function ($event) {
            $scope.status.sDateOpened = true;
        };

        $scope.eDateOpen = function ($event) {
            $scope.status.eDateOpened = true;
        };

        $scope.status = {
            sDateOpened: false,
            eDateOpened: false
        };
    }])

.directive('linearChart', ['$interval', 'dateFilter', '$window', function ($interval, dateFilter, $window) {

    function link(scope, element, attrs) {

        var dateToday = new Date();
        var fullData = [];
        var sDateMonth = 0;
        var eDateMonth = 11;
        var monthDomain = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11];
        var x, y, xLabels, line, graph, xAxis, yAxisLeft;
        var selStartDate = new Date(dateToday.getFullYear(), 0, 1);
        var selEndDate = new Date(dateToday.getFullYear(), 11, 31);

        scope.$watch(attrs.data, function (data) {
            fullData = data;
            if (fullData.length > 0)
                separateData();
        });

        scope.$watch(attrs.startDate, function (startDate) {
            selStartDate = startDate;
            sDateMonth = selStartDate.getMonth();
            separateData();
            redrawLineChart();
        });

        scope.$watch(attrs.endDate, function (endDate) {
            selEndDate = endDate;
            eDateMonth = selEndDate.getMonth();
            separateData();
            redrawLineChart();
        });

        var updatedWonData = [];
        var updatedLlossData = [];
        var wonData = [], lossData = [];
        var padding = 20;
        var pathClass = "path";
        var xScale, yScale, xAxisGen, yAxisGen, lineFun, m, w, h;
        var d3 = $window.d3;
        var rawSvg = element.find('svg');
        var svg = d3.select(rawSvg[0]);

    function separateData() {
        wonData = [];
        lossData = [];
        var janWon = 0, febWon = 0, marWon = 0, aprWon = 0, mayWon = 0, junWon = 0, julWon = 0, augWon = 0, sepWon = 0, octWon = 0, novWon = 0, decWon = 0;
        var janLoss = 0, febLoss = 0, marLoss = 0, aprLoss = 0, mayLoss = 0, junLoss = 0, julLoss = 0, augLoss = 0, sepLoss = 0, octLoss = 0, novLoss = 0, decLoss = 0;
        for (var i = 0; i < fullData.length; i++) {
            if (fullData[i].EstimationEndDate >= selStartDate && fullData[i].EstimationEndDate <= selEndDate)
                if (fullData[i].EstimationEndDate.getMonth() == '0') {
                    if (fullData[i].PreSales_x0020_Outcome == 'Win')
                        janWon++;
                    else if (fullData[i].PreSales_x0020_Outcome == 'Loss')
                        janLoss++;
                }
                else if (fullData[i].EstimationEndDate.getMonth() == '1') {
                    if (fullData[i].PreSales_x0020_Outcome == 'Win')
                        febWon++;
                    else if (fullData[i].PreSales_x0020_Outcome == 'Loss')
                        febLoss++;
                }
                else if (fullData[i].EstimationEndDate.getMonth() == '2') {
                    if (fullData[i].PreSales_x0020_Outcome == 'Win')
                        marWon++;
                    else if (fullData[i].PreSales_x0020_Outcome == 'Loss')
                        marLoss++;
                }
                else if (fullData[i].EstimationEndDate.getMonth() == '3') {
                    if (fullData[i].PreSales_x0020_Outcome == 'Win')
                        aprWon++;
                    else if (fullData[i].PreSales_x0020_Outcome == 'Loss')
                        aprLoss++;
                }
                else if (fullData[i].EstimationEndDate.getMonth() == '4') {
                    if (fullData[i].PreSales_x0020_Outcome == 'Win')
                        mayWon++;
                    else if (fullData[i].PreSales_x0020_Outcome == 'Loss')
                        mayLoss++;
                }
                else if (fullData[i].EstimationEndDate.getMonth() == '5') {
                    if (fullData[i].PreSales_x0020_Outcome == 'Win')
                        junWon++;
                    else if (fullData[i].PreSales_x0020_Outcome == 'Loss')
                        junLoss++;
                }
                else if (fullData[i].EstimationEndDate.getMonth() == '6') {
                    if (fullData[i].PreSales_x0020_Outcome == 'Win')
                        julWon++;
                    else if (fullData[i].PreSales_x0020_Outcome == 'Loss')
                        julLoss++;
                }
                else if (fullData[i].EstimationEndDate.getMonth() == '7') {
                    if (fullData[i].PreSales_x0020_Outcome == 'Win')
                        augWon++;
                    else if (fullData[i].PreSales_x0020_Outcome == 'Loss')
                        augLoss++;
                }
                else if (fullData[i].EstimationEndDate.getMonth() == '8') {
                    if (fullData[i].PreSales_x0020_Outcome == 'Win')
                        sepWon++;
                    else if (fullData[i].PreSales_x0020_Outcome == 'Loss')
                        sepLoss++;
                    }
                    else if (fullData[i].EstimationEndDate.getMonth() == '9') {
                        if (fullData[i].PreSales_x0020_Outcome == 'Win')
                            octWon++;
                        else if (fullData[i].PreSales_x0020_Outcome == 'Loss')
                            octLoss++;
                    }
                    else if (fullData[i].EstimationEndDate.getMonth() == '10') {
                        if (fullData[i].PreSales_x0020_Outcome == 'Win')
                            novWon++;
                        else if (fullData[i].PreSales_x0020_Outcome == 'Loss')
                            novLoss++;
                    }
                    else if (fullData[i].EstimationEndDate.getMonth() == '11') {
                        if (fullData[i].PreSales_x0020_Outcome == 'Win')
                            decWon++;
                        else if (fullData[i].PreSales_x0020_Outcome == 'Loss')
                            decLoss++;
                    }
            }
            wonData.push(janWon, febWon, marWon, aprWon, mayWon, junWon, julWon, augWon, sepWon, octWon, novWon, decWon);
            updatedWonData.push(wonData);
            lossData.push(janLoss, febLoss, marLoss, aprLoss, mayLoss, junLoss, julLoss, augLoss, sepLoss, octLoss, novLoss, decLoss);
            updatedLlossData.push(lossData);
            drawLineChart();
        }

        function setChartParameters() {
            m = [80, 80, 80, 80];
            w = 850 - m[1] - m[3];
            h = 400 - m[0] - m[2];
            updatedWonData = [];
            updatedLlossData = [];
            var xMaxDomain, xLabelMax;
            if (selStartDate.getFullYear() != selEndDate.getFullYear()) {
                for (var i = sDateMonth; i <= 11; i++) {
                    updatedWonData.push(wonData[i]);
                    updatedLlossData.push(lossData[i]);
                }
                for (var i = 0; i <= eDateMonth; i++) {
                    updatedWonData.push(wonData[i]);
                    updatedLlossData.push(lossData[i]);
                }
            }
            else {
                for (var i = sDateMonth; i <= eDateMonth; i++) {
                    updatedWonData.push(wonData[i]);
                    updatedLlossData.push(lossData[i]);
                }

            }
function formatCurrency(d) {
                return d;}
                if (selStartDate.getFullYear() != selEndDate.getFullYear())
                xMaxDomain = 12 - sDateMonth + eDateMonth + 1;
            else
                xMaxDomain = eDateMonth - sDateMonth + 1;
            xLabelMax = new Date(selEndDate.getFullYear(), selEndDate.getMonth() + 1, 0).getDate();
            xLabels = d3.time.scale().domain([new Date(selStartDate.getFullYear(), selStartDate.getMonth(), 0), new Date(selEndDate.getFullYear(), selEndDate.getMonth(), xLabelMax)]).range([0, w]);
            x = d3.scale.linear().domain([0, xMaxDomain]).range([0, w]);
        y = d3.scale.linear().domain([0, 10]).range([h, 0]);
        xAxis = d3.svg.axis().scale(xLabels).ticks(xMaxDomain).tickFormat(d3.time.format("%b")).tickSize(-
h).tickSubdivide(true);
        yAxisLeft = d3.svg.axis().scale(y).ticks(10).tickFormat(formatCurrency).orient("left");
        line = d3.svg.line()
            .x(function (d, i) {
                return x(i);
            })
            .y(function (d) {
                return y(d);
            });
    }

    function drawLineChart() {

        setChartParameters();

        var graph = svg.attr("width", w + m[1] + m[3])
        .attr("height", 350)
        .append("svg:g")
        .attr("transform", "translate(" + 120 + "," + m[0] + ")");

        graph.append("svg:g")
        .attr("class", "x axis")
        .attr("transform", "translate(0," + h + ")")
        .call(xAxis);

        graph.append("svg:g")
        .attr("class", "y axis")
        .attr("transform", "translate(-50,0)")
        .call(yAxisLeft);

        graph.append("svg:path")
        .attr("d", line(updatedWonData))
        .attr('stroke', 'green')
        .attr('class', 'wonline');

        graph.append("svg:path")
        .attr("d", line(updatedLlossData))
        .attr('stroke', 'black')
        .attr('class', 'lossline');
    }

    function redrawLineChart() {

        setChartParameters();

        svg.selectAll(".y.axis").call(yAxisLeft);

        svg.selectAll(".x.axis").call(xAxis);

        svg.selectAll("." + "wonline")
          .attr({
              d: line(updatedWonData)
          });
        svg.selectAll("." + "lossline")
          .attr({
              d: line(updatedLlossData)
          });

    }

}

return {
    scope: true,
    link: link,
    template: "<svg width='850' height='350'></svg>"
};
}]);

})();
```

6.3 Angular JS code for recent wins web part

```
/*global angular*/
(function () {
    'use strict';


//Recent Wins
angular.module('preSalesApp').controller('AccordionCtrl', ['$scope','spManager',function ($scope) {
  $scope.oneAtATime = true;


 }]);


})();
```

Angular JS code for Health Check web part:

```
/*global angular*/
(function () {
    'use strict';

    //health check

    angular.module('preSalesApp').controller('preSalesAppExtended', ['$scope', 'spManager', function ($scope, spManager) {

        $scope.data = [];
        $scope.redCor = 0;
        $scope.greenCor = 0;
        $scope.yellowCor = 0;
        $scope.itemsLength = 0;
        $scope.init = function () {
            var Context = SP.ClientContext.get_current();
            var web = Context.get_web();
            var oList = Context.get_web().get_lists().getByTitle('Presales');
            var camlQuery = new SP.CamlQuery();
            camlQuery.set_viewXml("<View><RowLimit>100</RowLimit><Query><OrderBy><FieldRef Name='ID' Ascending='FALSE' /></OrderBy><Where><And><Neq><FieldRef Name='PreSales_x0020_Outcome' /><Value Type='Choice'>Cancelled</Value></Neq><Neq><FieldRef Name='PreSales_x0020_Outcome' /><Value Type='Choice'>On Hold</Value></Neq></And></Where></Query></View>");
            var collImages = oList.getItems(camlQuery);
            Context.load(oList);
            Context.load(collImages, 'Include(PreSales_x0020_Outcome,Solution_x0020_Design_x0020_Url,Requirement_x0020_Doc_x0020_exis,Solution_x0020_Design_x0020_Revi,Estimation_x0020_Document_x0020_,Estimation_x0020_Reviewed,Requirement_x0020_Doc_x0020_Url.Include(Url,Description),Requirement_x0020_Doc_x0020_exis,EstimationEndDate)');
            Context.executeQueryAsync(function () {
                var imagesEnumerator = collImages.getEnumerator();
                var imagesList = [];
                while (imagesEnumerator.moveNext()) {
                    var oImageItem = imagesEnumerator.get_current();

                    imagesList.push(oImageItem.get_fieldValues());
                }
                $scope.data = imagesList;
                var todayDate = new Date();
                for (var i = 0; i < imagesList.length; i++) {

                    if (imagesList[i].Requirement_x0020_Doc_x0020_exis == false) {
                        $scope.redCor++;
                    }
                    else if ( imagesList[i].Estimation_x0020_Reviewed == true) {
                        $scope.yellowCor++;
                    }
                    else {
                        $scope.greenCor++;
                    }
                }
                $scope.$apply();

            }, function (sender, args) { alert(args.get_message()); });
        };
        SP.SOD.executeFunc('sp.js', 'SP.ClientContext', function () { $scope.init() });


    }]);


})();
```

6.4 Code for Estimation Values

```
function updateEstimationValues(Context, updatedEstimates, stream) {
          var oList = Context.get_web().get_lists().getByTitle('PreSales Estimation Values');
          var def = $q.defer();
          var oListItem = oList.getItemById(updatedEstimates.ID);
          oListItem.set_item('Estimated_x0020_Hours', updatedEstimates.Estimated_x0020_Hours);
          oListItem.set_item('Actual_x0020_Hours', updatedEstimates.Actual_x0020_Hours);
          oListItem.set_item('Estimated_x0020_Cost', updatedEstimates.Estimated_x0020_Cost);
          oListItem.set_item('Actual_x0020_Cost', updatedEstimates.Actual_x0020_Cost);
          oListItem.set_item('Streams', stream);
          var projectName = "-
1;#" + $scope.itemCol.PreSalesProject.Label + "|" + $scope.itemCol.PreSalesProject.TermGuid;
          oListItem.set_item('PreSalesProject', projectName);
          oListItem.set_item('Title', $scope.itemCol.PreSalesProject.Label);
          oListItem.update();

          Context.executeQueryAsync(function() {
              def.resolve();
          }, function() {
              alert("fails");
              def.reject("failed");
          });
          return def.promise;
      }
```

References

1. http://elegantsolutions.us/sharepoint-services/
2. https://msdn.microsoft.com/en-us/library/ms473633(v=office.12).aspx
3. http://sptechbytes.blogspot.com/2013/09/sharepoint-2013-server-object-model.html
4. https://msdn.microsoft.com/en-us/library/office/ms473633(v=office.14).aspx
5. http://en.share-gate.com/blog/start-learn-sharepoint-basics
6. https://technet.microsoft.com/en-us/library/dd309672(v=ax.50).aspx
7. https://msdn.microsoft.com/en-us/library/cc534997(v=office.12).as
8. https://en.wikipedia.org/wiki/SharePoint#Origins
9. https://msdn.microsoft.com/en-us/library/office/mt674607.aspx
10. https://technet.microsoft.com/en-us/library/fp179725.aspx
11. http://i.msdn.microsoft.com/dynimg/IC367367.jpg
12. https://support.office.com/en-us/article/Introduction-to-SharePoint-master-pages-dc9c4388-8dce-41b8-abb8-eeda2801b1a7
13. http://searchsoa.techtarget.com/definition/content-management-system
14. https://en.wikipedia.org/wiki/SharePoint
15. https://msdn.microsoft.com/en-us/library/office/ms467435(v=office.14).aspx
16. https://msdn.microsoft.com/en-us/library/ms473633(v=office.12).aspx
17. https://msdn.microsoft.com/en-us/library/office/jj163181.aspx
18. https://msdn.microsoft.com/en-us/library/office/ms468447(v=office.14).aspx