

ACKNOWLEDGEMENT

The internship opportunity I had with AlmaMate info Tech was a great chance for learning and professional development. Therefore, I consider myself as a very lucky individual as I was provided with an opportunity to be a part of it. I am also grateful for having a chance to meet so many wonderful people and professionals who led me through this internship period.

I express my deepest thanks to Mr. Shiv Shankar, Sr. Executive-Learning & Development at AlmaMate for taking part in useful decision & giving necessary advices and guidance and arranged and taking time out to hear, guide and keep me on the correct path and allowing me to carry out my project at their esteemed organization and extending during the training and all facilities to make project easier. I choose this moment to acknowledge his contribution gratefully.

It is my radiant sentiment to place on record my best regards, deepest sense of gratitude to Pranav sir (assistant professor) for their careful and precious guidance which were extremely valuable for my study both theoretically and practically.

I perceive as this opportunity as a big milestone in my career development. I will strive to use gained skills and knowledge in the best possible way, and I will continue to work on their improvement, in order to attain desired career objectives. Hope to continue cooperation with all of you in the future.

INTRODUCTION

Project management system is a software that is required for analyzing keeping track and enhancing the project developing process.

Company has to go through a lot of project and has to handle a lot of team and member while working on a project making their way bulky and obsolete sometime when the work crosses over a limit of human boundation,in order to keep a track of a lot of project and stuff and to make them more efficient ,economical, we requires a project management system and so does this project is all about keeping a track of lots of project and how do they function.

The function of project management system is

To keep tracks of

- 1 NO of projects
- 2 Employee engaged in project
- 3 track of project
- 4 Invoice record and generation
- 5 Report and analysis

LIMITATONS OF CURRENT SYSTEM :-

Data redundancy

It means that same data fields appear in many different files and often in different formats. In manual system, it poses quite a big problem because the data has to be maintained in large volumes but in our system, this problem

can be overcome by providing the condition that if the data entered is duplicate, it will not be entered, otherwise, updating will take place.

Difficulty in accessing the data

In manual system, searching information is time consuming but in our system, any information can be accessed by providing the primary key.

Unsatisfactory security measures

In manual system, no security measures were provided but in this system, password security has been provided. The person can access the system by providing the correct password otherwise he is denied the access.

This all is solved while using and working through a salesforce thus our project solves the problem of redundancy, security and it's quite easy to monitor data.

PROJECT MANAGEMENT

Project management is the discipline of initiating, planning, executing, controlling, and closing the [work](#) of a [team](#) to achieve specific goals and meet specific success criteria. A [project](#) is a temporary endeavor designed to produce a unique product, service or result with a defined beginning and end (usually time-constrained, and often constrained by funding or [deliverables](#)) undertaken to meet unique goals and objectives, typically to bring about beneficial change or added value. The temporary nature of projects stands in contrast with [business as usual \(or operations\)](#), which are repetitive, permanent, or semi-permanent functional activities to produce products or services. In practice, the [management](#) of these two systems is often quite different, and as such requires the development of distinct technical skills and management strategies.

The primary challenge of project management is to achieve all of the project goals within the given constraints. This information is usually described in a user or project manual, which is created at the beginning of the development process. The primary constraints are [scope](#), time, quality and budget. The secondary and more ambitious challenge is to [optimize](#) the [allocation](#) of necessary inputs and integrate them to meet pre-defined objectives.

TOOLS AND TECHNOLOGIES USED : -

TECHNOLOGY USED: -

SALESFORCE

Salesforce helps you manage your customer relationships, integrate with other systems, and build your own applications.

Salesforce is one of the feature of cloud computing which help us to work in a more efficient and friendly manner.

It provide us a platform where we are no longer dependent on

Bulky software downloaded in our pc's,all we need is a system and internet.

Features of salesforce are

- [Manage Relationships and Data With Records](#)

Much of your Salesforce data is stored in individual *records*, and organized within objects. For example, the Account object presents all of your account records. If the Acme company is one of your accounts, you'll have an account record for Acme.

- [Find Information with Search](#)

Salesforce search gives you the power to find information faster. Search is available via the Salesforce site, a Salesforce1 mobile app, or a custom search implementation built on the Salesforce platform.

- [Run a Report](#)

To run a report, find it on the Reports tab and click the report name. Most reports run automatically when you click the name. If you're already viewing a report, click Run Report to run it immediately or schedule a future run. In Lightning Experience, the Run Report button only appears after you save a report.

- [Your Access to UI Elements, Records, and Fields](#)

Your administrator can customize many different areas to secure your company's data. Users in Professional, Enterprise, Unlimited, Performance, and Developer Edition organizations can control other users' access to their data by sharing records individually with colleagues.

Q- WHAT IS SALESFORCE?

Salesforce helps you manage your customer relationships, integrate with other systems, and build your own applications.

Salesforce includes the following products and services.

Available in: SalesforceClassic

Salesforce Applications

Salesforce includes prebuilt applications (or “apps”) for customer relationship management Your Salesforce edition (CRM) ranging from sales force automation to partner relationship management, marketing, determines which features and customer service, and functionality you can access.

Force.com Platform

The Force.com platform is the first platform as a service (PaaS), enabling developers to create and deliver any kind of business application entirely on demand and without software. The

platform also includes easy to use point-and-click customization tools to help you create solutions for your unique business requirements, without any programming experience.

Cloud	A Salesforce name for a loose federation of features that help you accomplish certain types activities, such as
-------	---

selling products, supporting your customers, or collaborating with your coworkers. Two common examples you'll come across are *Service Cloud* and *Sales Cloud*.

Cloud Computing	Technology that enables Internet-based services that let you sign up and log in through a browser. Salesforce delivers its service in the cloud. Other familiar cloud computing services include Google Apps and Amazon.com.
Software as a Service (SaaS)	Software delivered not by traditional means (such as on disk) but in the cloud, as a service. There's nothing to download or install, and updates are automatic.
Trust	<p>A Salesforce term for its company-wide commitment to building and delivering the most secure, fast, and reliable cloud-based service available.</p> <p>Created to enhance customer success, trust.salesforce.com is a systems status website giving Salesforce customers and the community access to real-time and historical system performance information and updates, incident reports and maintenance schedules across all its key system components.</p> <p>trust.salesforce.com is free to all members of the Salesforce community.</p>
App	Short for <i>application</i> . A collection of components such as tabs, reports, dashboards, and Visualforce pages that address specific business needs. Salesforce provides standard apps, which you can customize, such as Sales and Call Center. You can customize the standard apps to

match the way you work.

Edition	One of several bundles of Salesforce products and services, each geared toward a different set of business needs. All Salesforce editions share the same look and feel, but they vary by feature, functionality, and pricing.
Object	<p>A definition of a specific type of information you can store in Salesforce. For example, the Case object lets you store information about customer inquiries. For each object, your organization will have multiple, specific records.</p> <p>Salesforce comes with lots of standard objects, but you can create custom objects, as well.</p>
Organization	A deployment of Salesforce that has a defined set of licensed users. Your organization includes all of your data and applications.

Term Definition

Record	A collection of fields that store information about a specific item of a specific type (represented by an object), such as a contact, an account, or an opportunity. For example, you might have a contact record to store information about Joe Smith, and a case record store information about his training inquiry.
--------	---

Release	<p>Salesforce releases new products and features three times per year, and releases are identified by season—Winter, Spring, and Summer—along with the calendar year. <i>Example:</i> Winter '15.</p> <p>For every Salesforce release, the Salesforce release notes include new features and products that are generally available or in beta release, plus all changes to existing features and products. You can find the release notes when you search for “Release Notes” in the Salesforce Help.</p>
---------	---

Salesforce	The name of the Salesforce cloud computing CRM service.
------------	---

Salesforce	The company name.
------------	-------------------

The Salesforce features listed here are those that are also available in Database.com. The features are organized by general product area. Some features are only available through the API. These are noted where they appear.

Administration

Audit Trail

Company Information

Delegated Administrator and Login As

Enhanced Profile Management

Field-Level Security

Login History

Mass Transfers

New Profile UI

Permission Sets

Profile

Apex (Libraries are available only for features supported in Database.com.)

API

API Clients

Group Layouts

Group Trigger and Group Member Triggers

Groups

Hashtags and Topics

Quick Actions

Rich Link Previews in Feed

Triggers for Feed Comments and Feed Items

Content

Custom Settings

Customer Domains

Data

Ability to mass transfer records of custom objects

Data Loader

Declarative App Builder and Custom Schema

Custom Objects and Fields

Force.com

Formula Fields

Rich Text Area

Roll-up Summary Fields

Validation Rules

Developer Tools

Force.com IDE

Divisions

Enhanced Profile User Interface

Enabling

Field Level Security

Field/Object Perms

Object and Tab settings – Objects only

Search

System Perms

Viewing and editing app perms

Viewing and editing Login Hours

Viewing and editing Login IP Ranges

Viewing and editing Service Provider Access

Viewing Profile Lists

Field Accessibility (for custom objects)

Field Dependency Matrix

Fileforce

Localization and Globalization

Multi-Currency • Timezones

My Personal Information

Change My Password

Editing personal information

Permission Sets

Ability to search permissions

Viewing, Editing, and Creating

Profile List Views

Creating

Editing

Restricting Logins

Hours and IP Ranges on Profiles

Sandbox

Developer Sandbox (for an additional fee)

Full Copy Sandbox (for an additional fee)

Setup Only Sandbox (for an additional fee)

Search

Personal Tags

Public Tags

Security

Connected Apps (OAuth)

Encrypted Custom Fields

General Security

Identity Feature

Inbound Single Sign-On

Password and Login Policies

Retrieving forgotten passwords

Single Sign-On

Sharing

Apex Managed Sharing

Criteria Based Sharing Rules

High Volume Portal User

Ownership Based Sharing Rules

Public Groups

Queues

Role Hierarchy

Sharing Model and Manual Sharing

Topics

Topic and TopicAssignment Triggers

User Setup

Creating and editing Personal and Public Groups

Send Activation Link

Users

Expiring Passwords

Manage Users

Managing Roles

Monitoring User Events (logins and training)

Resetting Passwords

Role List page

Set Up Password Expiration

Setting Login Restrictions

Viewing the Public Group Related List

Weekly Export

Workflow

Outbound messages (as both immediate and time-dependent actions)

Time-Based Workflow (Workflow Queue)

Workflow Field Updates and Outbound Messages

Workflow rules

Role Fields

SCOPE OF PROJECT

- Manage Client Projects.
- Track Invoices.
- Track Bills of a Project.
- Track no. of Days Worked on Project.
- Track no. of Hours Worked on Project.
- Create Invoice to be Send to Client.
- Track the Balance Amount Left on Project.
- Approval of project via hierarchy.
- Backup of Data.

S.R.S

A software requirements specification (SRS) is a comprehensive description of the intended purpose and environment for [software](#) under development. The SRS fully describes what the software will do and how it will be expected to perform.

An SRS minimizes the time and effort required by developers to achieve desired goals and also minimizes the development cost. A good SRS defines how an [application](#) will interact with system [hardware](#), other programs and human users in a wide variety of real-world situations. Parameters such as operating speed, [response time](#), [availability](#), [portability](#), maintainability, [footprint](#), security and speed of recovery from adverse events are evaluated. Methods of defining an SRS are described by the [IEEE](#) (Institute of Electrical and Electronics Engineers) specification 830-1998.

System development life cycle

Definition

A software process model or a software engineering is an abstract representation of a software process. It is a software development strategy that encompasses the process, methods and tools layers plus the generic phases namely definition phase, development phase and support phase. A process model is chosen based on the nature of the project and application, the methods and tools to be used and the controls and deliverables that are required.

Definition

A software process model or a software engineering is an abstract representation of a software process. It is a software development strategy that encompasses the process, methods and tools layers plus the generic phases namely definition phase, development phase and support phase. A process model is chosen based on the nature of the project and application, the methods and tools to be used and the controls and deliverables that are required.

Spiral Model

The spiral model, also known as the spiral lifecycle model, is a systems development lifecycle model used in information technology (IT). This model of development combines the features of the prototyping model and the waterfall model. The spiral model is favored for large, expensive, and complicated projects.

The steps in the spiral model iteration can be generalized as follows:

1. The system requirements are defined in as much detail as possible. This usually involves interviewing a number of users representing all the external or internal users and other aspects of the existing system.
2. A preliminary design is created for the new system. This phase is the most important part of "Spiral Model". In this phase all possible (and available) alternatives, which can help in developing a cost effective project are analyzed and strategies to use them are decided. This phase has been added specially in order to identify and resolve all the possible risks in the project development. If risks indicate any kind of uncertainty in requirements, prototyping may be used to proceed with the available data and find out possible solution in order to deal with the potential changes in the requirements.

3. A first prototype of the new system is constructed from the preliminary design. This is usually a scaled-down system, and represents an approximation of the characteristics of the final product.
4. A second prototype is evolved by a fourfold procedure:
 1. Evaluating the first prototype in terms of its strengths, weaknesses, and risks.
 2. Defining the requirements of the second prototype.
 3. Planning and designing the second prototype.
 4. Constructing and testing the second prototype.

Analysis

1. REQUIREMENT ANALYSIS:

Requirements are a feature of a system or description of something that is capable of doing in order to fulfill the system's purpose. It provides the appropriate mechanism for understanding what the customer wants, analyzing the needs, assessing feasibility, negotiating a solution, specifying the solution unambiguously, validating the specification and managing the requirements as they are translated into an operational system. Requirement Analysis is a task done under software engineering and software design. While requirements engineering specifies software's operational characteristics i.e. function, data behavior, indicates software's interface constraints, requirements analysis let the software engineer (called analysis) to refine the software allocation and construct models of data, functional and behavioral domains. Moreover, requirements analysis provides software developer with a representation of data, function and behavior that can be converted to data, architectural, interface and

componentlevel designs. At last, we can say that the requirement specification makes available, the developer and the customer, a means to assess quality, once the software has been built.

Software requirements analysis can be categorized into four areas of effort, as follows-

Evaluation and synthesis, Modeling, Specification, Review

The analyst starts with the studies of system specification and the software project plan. It is then important to understand the software in a system context. Also, the review of the software scope, used to generate planning estimate, is necessary. Next, communication for analysis must be established, so as to ensure problem recognition. The reason behind is to recognize the basic problem elements perceived by customer.

The next major area of effort for analysis is problem evaluation and solution synthesis. The engineer (or analyst) must define all data objects that are extremely observable. He must evaluate the content and flow of information. Also, he must define and describe all software functions, understand software behavior in the context of the system affected events, establish the characteristics of system interface, and uncover additional design constraints.

After evaluating the current problems and desired information (i.e., input and output), the engineer and analyst synthesizing one or more solutions. Initially, the data objects, processing functions and the system behavior are defined in detail. Once establishing this information, the analyst then considers basic architectures for implementation. Thus the process of evaluation and synthesis proceeds until both analyst and the customer are sure that software can be adequately specified for subsequent development steps.

During the evaluation and synthesis activity, the analyst creates the system model so as to better understand data and control flow, functional processing, operational behavior and the information content. The model provides a base for software design and the creation of specifications for the software.

2. REQUIREMENT SPECIFICATIONS:

A Software Requirements Specification (SRS) is a complete description of the behavior of the system to be developed. It includes a set of use case that describes all the interactions that the users will have with the software. Use cases are also known as Functional Requirements. Non-Functional Requirements are requirements which impose constraints on the design or implementation (such as performance requirements, quality standards, or design constraints).

i. FUNCTION REQUIREMENTS

Performance Requirements

User Satisfaction: - The system is such that it stands up to the user expectations.

Response Time: -The response of all the operation is good. This has been made possible by careful programming.

Error Handling: - Response to user errors and undesired situations has been taken care of to ensure that the system operates without halting.

Safety and Robustness: - The system is able to avoid or tackle disastrous action. In other words, it should be fool proof. The system safeguards against undesired events, without human intervention.

Portable: - The software should not be architecture specific. It should be easily transferable to other platforms if needed.

User friendliness: - The system is easy to learn and understand. A native user can also use the system effectively, without any difficulties.

Design constraint

There are a number of factors in the client's environment that may restrict the choices of a designer. Such factors include standards that must be followed, resource limits, operating environment, reliability and security requirements and policies that may have an impact on the design of the system. An SRS (Software Requirements Analysis and Specification) should identify and specify all such constraints.

Standard Compliance: - This specifies the requirements for the standards the system must follow. The standards may include the report format and accounting properties.

Hardware Limitations:- The software may have to operate on some existing or predetermined hardware, thus imposing restrictions on the design. Hardware limitations can include the types of machines to be used, operating system available on the system, languages supported and limits on primary and secondary storage.

Reliability and Fault Tolerance: - Fault tolerance requirements can place a major constraint on how the system is to be designed. Fault tolerance requirements often make the system more complex and expensive. Requirements about system behavior in the face of certain kinds of faults are specified. Recovery requirements are often an integral part here, detailing what the system should do if some failure occurs to ensure certain properties. Reliability requirements are very important for critical applications.

Security: - Security requirements are particularly significant in defense systems and database systems. They place restrictions on the use of certain commands, control access to data, provide different kinds of

access requirements for different people, require the use of passwords and cryptography techniques and maintain a log of activities in the system.

ii. HARDWARE REQUIREMENTS

For the hardware requirements the SRS specifies the logical characteristics of each interface b/w the software product and the hardware components. It specifies the hardware requirements like memory restrictions, cache size, the processor, RAM size etc... those are required for the software to run.

Minimum Hardware Requirements

Processor Pentium III Hard disk drive 40 GB RAM 128 MB Cache 512 kb

Preferred Hardware Requirements

Processor Pentium IV Hard disk drive 80 GB RAM 256 MB Cache 512 kb

iii. SOFTWARE REQUIREMENTS

Any window based operating system with DOS support are primary requirements for software development. Windows XP, FrontPage and dumps are required. The systems must be connected via LAN and connection to internet is mandatory.

It should also contain compatible browsers. An account or license of Salesforce platform is necessary.

Other requirements:

Software should satisfy following requirements as well:-

SECURITY PORTABILITY CORRECTNESS EFFICIENCY FLEXIBILITY TESTABILITY
EUSABILITY

iv. NON-FUNCTIONAL REQUIREMENTS

Security:

The system use SSL (secured socket layer) in all transactions that include any confidential customer information. The system must automatically log out all customers after a period of inactivity. The system should not leave any cookies on the customer's computer containing the user's password. The system's back-end servers shall only be accessible to authenticated management.

Reliability:

The reliability of the overall project depends on the reliability of the separate components. The main pillar of reliability of the system is the backup of the database which is continuously maintained and updated to reflect the most recent changes. Also the system will be functioning inside a container. Thus the overall stability of the system depends on the stability of container and its underlying operating system.

Availability:

The system should be available at all times, meaning the user can access it using a web browser, only restricted by the down time of the server on which the system runs. A customer friendly system which is in access of people around the world should work 24 hours. In case of a of a hardware failure or database corruption, a replacement page will be shown. Also in case of a hardware failure or database corruption, backups of the database should be retrieved from the server and saved by the Organizer. Then the service will be restarted. It means 24 x 7 availability.

Maintainability:

A commercial database is used for maintaining the database and the application server takes care of the site. In case of a failure, a re-initialization of the project will be done. Also the software design is being done with modularity in mind so that maintainability can be done efficiently.

Supportability:

The code and supporting modules of the system will be well documented and easy to understand. Online User Documentation and Help System Requirements.

3. DESIGN PHASE

The design phase involves converting the informational, functional, and network requirements identified during the initiation and planning phases into unified design specifications that developers use to script programs during the development phase.

Program designs are constructed in various ways. Using a top-down approach, designers first identify and link major program components and interfaces, then expand design layouts as they identify and link smaller subsystems and connections. Using a bottom-up approach, designers first identify and link minor program components and interfaces, then expand design layouts as they identify and link larger systems and connections. Contemporary design techniques often use prototyping tools that build mock-up designs of items such as application screens, database layouts, and system architectures. End users, designers, developers, database managers, and network administrators should review and refine the prototyped designs in an iterative process until they agree on an acceptable design. Audit,

security, and quality assurance personnel should be involved in their view and approval process. During this phase, the system is designed to satisfy the functional requirements identified in the previous phase. Since problems in the design phase could be very expensive to solve in the later stage of the software development, a variety of elements are considered in the design to mitigate risk. These include:

- Identifying potential risks and defining mitigating design features.
- Performing a security risk assessment. Developing a conversion plan to migrate current data to the new system.
- Determining the operating environment.
- Defining major subsystems and their inputs and outputs.
- Allocating processes to resources.
- Preparing detailed logic specifications for each software module.

The result is a draft System Design Document which captures the preliminary design for the system. Everything requiring user input or approval is documented and reviewed by the user. Once these documents have been approved by the Agency CIO and Business Sponsor, the final System Design Document is created to serve as the Critical/Detailed Design for the system. This document receives a rigorous review by Agency technical and functional representatives to ensure that it satisfies the business requirements. Concurrent with the development of the system design, the Agency Project Manager begins development of the Implementation Plan, Operations and Maintenance Manual, and the Training Plan.

4. IMPLEMENTATION PHASE

The development phase involves converting design specifications into executable programs.

Effective development standards include requirements that programmers and other project participants discuss design specifications before programming begins. The procedures help ensure programmers clearly understand program designs and functional requirements.

Programmers use various techniques to develop computer programs. The large transaction-oriented programs associated with financial institutions have traditionally been developed using procedural programming techniques. Procedural programming involves the line-by-line scripting of logical instructions that are combined to form a program. Effective completion of the previous stages is a key factor in the success of the Development phase.

The Development phase consists of:-

- Translating the detailed requirements and design into system components.
- Testing individual elements (units) for usability.
- Preparing for integration and testing of the IT system. Integration and Test Phase.

5. INTEGRATION AND TEST PHASE

Subsystem integration, system, security, and user acceptance testing is conducted during the integration and test phase. The user, with those responsible for quality assurance, validates that the functional requirements, as defined in the functional requirements document, are

satisfied by the developed or modified system. OIT Security staff assesses the system security and issue a security certification and accreditation prior to installation/implementation.

Multiple levels of testing are performed, including:-

- Testing at the development facility by the contractor and possibly supported by end users.
- Testing as a deployed system with end users working together with contract personnel.
- Operational testing by the end user alone performing all functions.

Requirements are traced throughout testing, a final Independent Verification & Validation evaluation is performed and all documentation is reviewed and accepted prior to acceptance of the system.

6. IMPLEMENTATION PHASE

This phase is initiated after the system has been tested and accepted by the user. In this phase, the system is installed to support the intended business functions. System performance is compared to performance objectives established during the planning phase. Implementation includes user notification, user training, installation of hardware, installation of software onto production computers, and integration of the system into daily work processes. This phase continues until the system is operating in production in accordance with the defined user requirements.

7. OPERATION AND MAINTENANCE PHASE

The system operation is ongoing. The system is monitored for continued performance in accordance with user requirements and needed system modifications are incorporated. Operations continue as long as the system can be effectively adapted to respond to the organization's needs. When modifications or changes are identified, the system may reenter the planning phase.

The purpose of this phase is to:-

- Operate, maintain, and enhance the system.
- Certify that the system can process sensitive information.
- Conduct periodic assessments of the system to ensure the functional requirements continue to be satisfied.
- Determine when the system needs to be modernized, replaced, or retired.

TESTING: -

Testing is the process of exercising software with the intent of finding errors and ultimately correcting them. The following testing techniques have been used to make this project free of errors.

Content Review

The whole content of the project has been reviewed thoroughly to uncover typographical errors, grammatical error and ambiguous sentences.

Navigation Errors

Different users were allowed to navigate through the project to uncover the navigation errors. The views of the user regarding the navigation flexibility and user friendliness were taken into account and implemented in the project.

Unit Testing

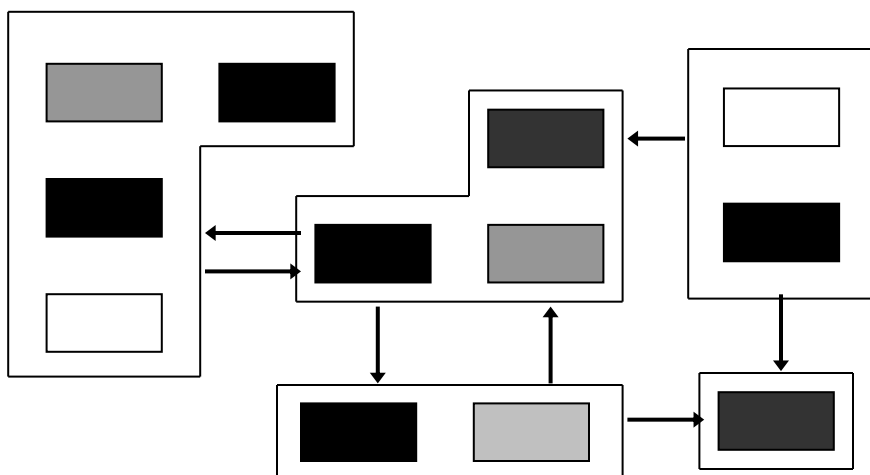
Focuses on individual software units, groups of related units.

Unit – smallest testable piece of software.

A unit can be compiled /assembled / linked/loaded; and put under a test harness. Unit testing done to show that the unit does not satisfy the application and /or its implemented software does not match the intended designed structure.

Integration Testing

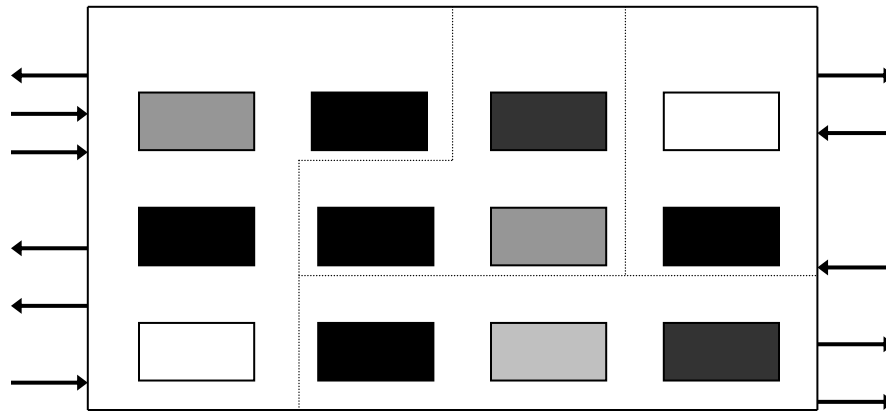
Focuses on combining units to evaluate the interaction among them. Integration is the process of aggregating components to create larger components. Integration testing done to show that even though components were individually satisfactory, the combination is incorrect and inconsistent.



System Testing

Focuses on a complete integrated system to evaluate compliance with specified requirements (test characteristics that are only present when entire system is run)

A system is a big component. System testing is aimed at revealing bugs that cannot be attributed to a component as such, to inconsistencies between components or planned interactions between components. Concern: issues, behaviors that can only be exposed by testing the entire integrated system (e.g., performance, security, recovery)



each form encapsulates (labels, texts, grid etc.). Hence in case of project in V.B. form are the basic units. Each form is tested thoroughly in term of calculation, display etc.

Regression Testing

Each time a new form is added to the project the whole project is tested thoroughly to rectify any side effects. That might have occurred due to the addition of the new form. Thus regression testing has been performed.

White-Box testing

White-box testing (also known as clear box testing, glass box testing, transparent box testing and structural testing) tests internal structures or workings of a program, as opposed to the functionality exposed to the end-user. In white-box testing an internal perspective of the system, as well as programming skills, are used to design test cases. The tester chooses inputs to exercise paths through the code and determine the appropriate outputs. This is analogous to testing nodes in a circuit, e.g. in-circuit testing (ICT).

While white-box testing can be applied at the unit, integration and system levels of the software testing process, it is usually done at the unit level. It can test paths within a unit, paths between units during integration, and between subsystems during a system-level test.

Though this method of test design can uncover many errors or problems, it might not detect unimplemented parts of the specification or missing requirements.

Techniques used in white-box testing include:

API testing (application programming interface) – testing of the application using public and private APIs

Code coverage – creating tests to satisfy some criteria of code coverage (e.g., the test designer can create tests to cause all statements in the program to be executed at least once)

Fault injection methods – intentionally introducing faults to gauge the efficacy of testing strategies

Mutation testing methods

Static testing methods

Code coverage tools can evaluate the completeness of a test suite that was created with any method, including black-box testing. This allows the software team to examine parts of a system that are rarely tested and ensures that the most important function points have been tested. Code coverage as a software metric can be reported as a percentage for:

Function coverage, which reports on functions executed

Statement coverage, which reports on the number of lines executed to complete the test

100% statement coverage ensures that all code paths, or branches (in terms of control flow) are executed at least once. This is helpful in ensuring correct functionality, but not sufficient since the same code may process different inputs correctly or incorrectly.

Black-box testing

Black-box testing treats the software as a "black box", examining functionality without any knowledge of internal implementation. The tester is only aware of what the software is supposed to do, not how it does it. Black-box testing methods include: equivalence partitioning, boundary value analysis, all-pairs testing, state transition tables, decision table testing, fuzz testing, model-based testing, use case testing, exploratory testing and specification-based testing.

Specification-based testing aims to test the functionality of software according to the applicable requirements. This level of testing usually requires thorough test cases to be provided to the tester, who then can simply verify that for a given input, the output value (or behaviour), either "is" or "is not" the same as the expected value specified in the test case. Test cases are built around specifications and requirements, i.e., what the application is supposed to do. It uses external

descriptions of the software, including specifications, requirements, and designs to derive test cases. These tests can be functional or non-functional, though usually functional.

Specification-based testing may be necessary to assure correct functionality, but it is insufficient to guard against complex or high-risk situations.

One advantage of the black box technique is that no programming knowledge is required. Whatever biases the programmers may have had, the tester likely has a different set and may emphasize different areas of functionality. On the other hand, black-box testing has been said to be "like a walk in a dark labyrinth without a flashlight." Because they do not examine the source code, there are situations when a tester writes many test cases to check something that could have been tested by only one test case, or leaves some parts of the program untested.

This method of test can be applied to all levels of software testing: unit, integration, system and acceptance. It typically comprises most if not all testing at higher levels, but can also dominate unit testing as well.

Testing part involved in project: -

Salesforce Basics Salesforce Editions

Force.com Developer Console - Google Chrome

https://ap2.salesforce.com/_ui/common/apex/debug/ApexCSIPage

route1.apxc Log executeAnonymous @10/9/2016, 7:12:01 AM

Execution Log

Timestamp	Event	Details
07:12:01:102	USER_DEBUG	[24]DEBUG()

☐ This Frame ☐ Executable ☒ Debug Only ☐ Filter [Click here to filter the log](#)

Logs, Tests, and Problems

Windows taskbar: Windows Start button, Task View, Internet Explorer, File Explorer, Microsoft Word, Excel, PowerPoint, Outlook, Chrome, and system tray icons (network, volume, battery). System clock: 7:14 AM, 10/9/2016.

Force.com Developer Console - Google Chrome

https://ap2.salesforce.com/_ui/common/apex/debug/ApexCSIPage

File Edit Debug Test Workspace Help < >

route1.apxc Log executeAnonymous @10/9/2016, 7:12:01 AM

Execution Log

Timestamp	Event	Details
07:12:01:002	USER_INFO	[EXTERNAL] 00528000004FMrk sahilsharma@railforce.in Pacific Standard Time GMT-07:00
07:12:01:002	EXECUTION_ST...	
07:12:01:002	CODE_UNIT_ST...	[EXTERNAL] execute_anonymous_apex
07:12:01:020	VARIABLE_SCO...	[1] r gettrain true false
07:12:01:020	HEAP_ALLOCATE	[72] Bytes:3
07:12:01:020	HEAP_ALLOCATE	[77] Bytes:152
07:12:01:020	HEAP_ALLOCATE	[342] Bytes:408
07:12:01:020	HEAP_ALLOCATE	[355] Bytes:408
07:12:01:020	HEAP_ALLOCATE	[467] Bytes:48
07:12:01:021	HEAP_ALLOCATE	[139] Bytes:6
07:12:01:021	HEAP_ALLOCATE	[EXTERNAL] Bytes:3
07:12:01:021	STATEMENT_EX...	[1]
07:12:01:021	STATEMENT_EX...	[1]
07:12:01:021	HEAP_ALLOCATE	[1] Bytes:156
07:12:01:021	HEAP_ALLOCATE	[1] Bytes:15
07:12:01:021	HEAP_ALLOCATE	[1] Bytes:2
07:12:01:021	METHOD_ENTRY	[1] 01p28000009n5vX gettrain.gettrain()
07:12:01:021	STATEMENT_EX...	[1]
07:12:01:021	STATEMENT_EX...	[1]
07:12:01:021	HEAP_ALLOCATE	[50] Bytes:5
07:12:01:021	HEAP_ALLOCATE	[56] Bytes:5
07:12:01:021	HEAP_ALLOCATE	[64] Bytes:7
07:12:01:021	SYSTEM_MODE...	false
07:12:01:021	HEAP_ALLOCATE	[1] Bytes:5

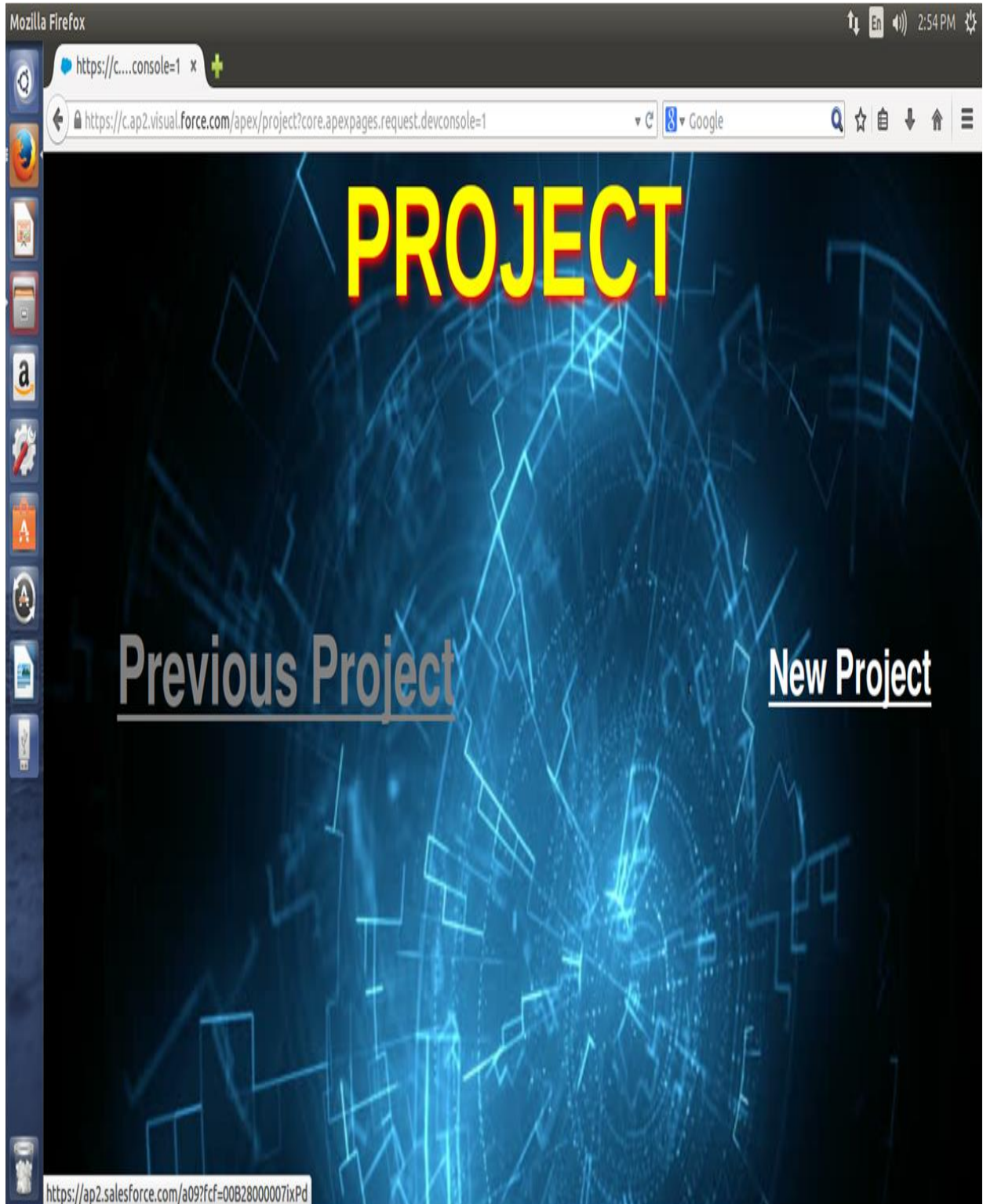
☐ This Frame ☐ Executable ☐ Debug Only ☐ Filter

Logs, Tests, and Problems

Windows taskbar icons: Windows, File Explorer, Microsoft Edge, Word, Excel, PowerPoint, Visual Studio Code, and Google Chrome. System tray shows network, volume, and battery status. Date and time: 7:13 AM 10/9/2016.

SCREENSHOTS





Projects ~ Salesforce - Developer Edition - Mozilla Firefox

Projects ~ Salesforc... x

https://ap2.salesforce.com/a09?fcf=00B28000007ixPd

Google

ms vpk Setup Help Sales

Home Chatter Campaigns Leads Accounts Contacts Opportunities Forecasts Contracts Orders Cases Solutions Products Reports Dashboards Candidates Job Posting Sites

trial Edit Delete Create New View

Change Owner

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Other All

Action	project Name	Project id no	Record ID
<input type="checkbox"/> Edit Del	Cab management	a-008474	a092800000lhzyf
<input type="checkbox"/> Edit Del	CAB MANAGEMENT	a-008475	a092800000lIRR3
<input type="checkbox"/> Edit Del	Hospital Management	a-008473	a092800000lhzya
<input type="checkbox"/> Edit Del	Hospital Management	a-008477	a092800000lRjH
<input type="checkbox"/> Edit Del	jai bholenath ki	a-008470	a092800000l3iHe
<input type="checkbox"/> Edit Del	Mata ji	a-008471	a092800000lHeO
<input type="checkbox"/> Edit Del	movie management	a-008479	a092800000lgiMI
<input type="checkbox"/> Edit Del	Railway Management	a-008478	a092800000lKSBF
<input type="checkbox"/> Edit Del	school management	a-008472	a092800000lhtP9
<input type="checkbox"/> Edit Del	Student Management	a-008476	a092800000lRIY

1-10 of 10 0 Selected

« Previous Next »

Page 1 of 1

The screenshot shows a web browser window with the address bar displaying `https://ap2.salesforce.com/a09/e?retURL=%2Fa09%2Fo`. The page title is "Project Edit" and the main heading is "New Project". A "Help for this Page" link is visible in the top right corner. The form is titled "Project Edit" and includes "Save", "Save & New", and "Cancel" buttons. A red bar at the top of the form area indicates "Required Information". The form fields are organized into two main sections: "INFO" and "PROJECT COMPLETE INFO".

INFO Section:

- project Name:** A text input field.
- client:** A text input field with a magnifying glass icon for search.
- starting date:** A date input field showing "7/30/2016".
- status:** A dropdown menu currently set to "--None--".
- Group:** A text input field with a magnifying glass icon for search.

PROJECT COMPLETE INFO Section:

- company_name:** A text input field.
- project type:** A dropdown menu currently set to "--None--".
- sub status:** A dropdown menu currently set to "--None--".

A rich text editor is located between the "starting date" and "company_name" fields, featuring a toolbar with options for bold, italic, underline, strikethrough, link, unlink, bulleted list, numbered list, and indent. The editor area is currently empty.



Clients - Salesforce - Developer Edition - Mozilla Firefox

https://ap2.salesforce.com/a0A2fcf=00B28000007ZT6u

Search... Search

ms ypk Setup Help Sales

Home Chatter Campaigns Leads Accounts Contacts Opportunities Forecasts Contracts Orders Cases Solutions Products Reports Dashboards Candidates Job_Posting_Sites Job Applications

All Edit Delete Create New View

Change Owner

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z Other All

Action	Client ID
<input type="checkbox"/> Edit Del	C-101001
<input type="checkbox"/> Edit Del	C-101002
<input type="checkbox"/> Edit Del	C-101003
<input type="checkbox"/> Edit Del	C-101004
<input type="checkbox"/> Edit Del	C-101005
<input type="checkbox"/> Edit Del	ia ambe maa
<input type="checkbox"/> Edit Del	ia bholaath ki

1-7 of 7 0 Selected Previous Next Page 1 of 1

client Edit: New client ~ Salesforce - Developer Edition - Mozilla Firefox

client Edit: New client... x

https://ap2.salesforce.com/a0A/e?retURL=%2Fa0A%2Fo&RecordType=01228000000ubWn&ent=0128000001Ypi9

Search...

ms vpk Setup Help Sales

Home Chatter Campaigns Leads Accounts Contacts Opportunities Forecasts Contracts Orders Cases Solutions Products Reports Dashboards Candidates Job Posting Sites Job Applications

client Edit

New client

client Edit Save Save & New Cancel

Client info Required information

Record Type Individual

First Name

Last name

Billing address

pin code

email

Mobile no

project detail

Project name

project_detail

Information

Owner ms vpk

Save Save & New Cancel

Copyright © 2000-2016 salesforce.com, inc. All rights reserved. Privacy Statement Security Statement Terms of Use 508 Compliance

Accounts: Home ~ Salesforce - Developer Edition - Mozilla Firefox

Accounts: Home ~ Sal... x

https://ap2.salesforce.com/a0G/o

MSVPK

Search... Search

ms vpk Setup Help project details

Home Chatter Employee Projects Clients Groups Invoices Accounts PMSS +

Create New...

Accounts Home

Help for this Page

Recent Items

- ms vpk
- C-101004
- I-101001
- Cab management
- fubbu deepk
- paras suraj
- school management
- JBK
- C-101003
- msvpk2

Recycle Bin

View: trial Go! Edit | Create New View

Recent Accounts

New Recently Viewed

project name
school management
trj
a

Copyright © 2000-2016 salesforce.com, inc. All rights reserved. | [Privacy Statement](#) | [Security Statement](#) | [Terms of Use](#) | [508 Compliance](#)

Account Edit: New Account ~ Salesforce - Developer Edition - Mozilla Firefox

Account Edit: New Ac... x

https://ap2.salesforce.com/a0G/e?retURL=%2Fa0G%2Fo

Create New...

Account Edit

New Account

Help for this Page

Account Edit

Save Save & New Cancel

project info

project name project id

project cost

installment 2

last payment recieved 2 due date 2 [7/29/2016]

installment 1

First Installment due date1 [7/29/2016]

last payment recieved 1

one time payment

total cost due date [7/29/2016]

Recieved

installment 3

last payment recieved 3 due date 3 [7/29/2016]

Information

Owner ms vpk

Save Save & New Cancel

Recent Items

- ms vpk
- C-101004
- I-101001
- Cab management
- fubbu deepk
- paras suraj
- school management
- JBK
- C-101003
- msvpk2

Recycle Bin

Copyright © 2000-2016 salesforce.com, inc. All rights reserved. | Privacy Statement | Security Statement | Terms of Use | 508 Compliance

Employee: Home ~ Salesforce - Developer Edition - Mozilla Firefox

Employee: Home ~ S... x

https://ap2.salesforce.com/a07/o

ms vpk Setup Help project details

MSVPK

Home Chatter Employee Projects Clients Groups Invoices Accounts PMSS

Create New...

Employee Home

Help for this Page

View: All Go! Edit | Create New View

Recent Items

- ms vpk
- C-101004
- I-101001
- Cab management
- fubbu deepk
- paras suraj
- school management
- JBK
- C-101003
- msvpk2

Recycle Bin

Recent Employee

New

Recently Viewed

First Name
Richa
Priya
Kar-wai
Yimou
Xiang
riva
ashutosh
arva
fubbu
paras
Show 25 items

Copyright © 2000-2016 salesforce.com, inc. All rights reserved. | [Privacy Statement](#) | [Security Statement](#) | [Terms of Use](#) | [508 Compliance](#)

Employee Edit: New Employee ~ Salesforce - Developer Edition - Mozilla Firefox

Employee Edit: New ... x

https://ap2.salesforce.com/a07/e?retURL=%2Fa07%2Fo

Recent Items

- ms vpk
- C-101004
- I-101001
- Cab management
- fubbu deepk
- paras suraj
- school management
- JBK
- C-101003
- msvpk2

Recycle Bin

Employee Edit

Save Save & New Cancel

Personal info

= Required Information

Salutation --None--

First Name

Last name

Gender --None--

Role

Education

Date Of Birth [7/29/2016]

Address

Correspondence address

e-mail

Image id

Professional

Date Of Joining [7/29/2016]

No. Of Projects Handled

Qualification

No. Of Projects Pending

Strong point

No. Of Projects Completed

Weak point

Comment

Min Pay

Max pay

Information

Owner ms vpk

Save Save & New Cancel

Groups: Home ~ Salesforce - Developer Edition - Mozilla Firefox

Groups: Home ~ Sal... x

https://ap2.salesforce.com/a0B/o

ms vpk Setup Help project details

MSVPK

Home Chatter Employee Projects Clients **Groups** Invoices Accounts PMSS +

Create New...

Groups Home Help for this Page

View: All Go! Edit | Create New View

Recent Items

- ms vpk
- C-101004
- I-101001
- Cab management
- tubbu deepk
- paras suraj
- school management
- JBK
- C-101003
- msvpk2

Recycle Bin

Recent Groups

New Recently Viewed

Group Name
2
3
1

Copyright © 2000-2016 salesforce.com, Inc. All rights reserved. | [Privacy Statement](#) | [Security Statement](#) | [Terms of Use](#) | [508 Compliance](#)

Group Edit: New Group ~ Salesforce - Developer Edition - Mozilla Firefox

Group Edit: New Gr... x

https://ap2.salesforce.com/a0B/e?retURL=%2Fa0B%2Fo

Recent Items

- ms vpk
- C-101004
- I-101001
- Cab management
- fubbu deepk
- paras suraj
- school management
- JBK
- C-101003
- msvpk2

Recycle Bin

Group Edit

Save Save & New Cancel

Info

= Required Information

Group Number Group Specialisation

Group Name Leader Name

Project

member details

Member1 Member2

Role1 Role2

Working Days1 Working Days2

Member3

Role3

Working Days3

previous records

Comments

Information

Owner ms vpk

Save Save & New Cancel

LIMITATION

Limited user

Here we only got to work with a max of five user id tus restricting our scope of adding more departments or field in it.this limitation can be overcome by purchasing more license from salesforce.

Self tracking of previous client

We can not trackdown the previous client that hav a relation with company.

Bill generation

In term of billing we could not extract a bill of past records .

Limited permission

The permission provided while working in salesforce software were limited

FUTURE SCOPE

the it sector is growing so is the requirement of apps.

with more upgradation and enhancement it will come on it.

There is always be a project upcoming in the world so is the need of managing it.

Hence the need of project management system is always there and will be always there in market.