1. ABSTRACT

"**PlugH"** is a web platform where a company can hire services for creating awareness about health, conduct polls and survey on employees who are associated with the "PlugH" project.
The companies should be registered as clients to take various services provided under this project; the client can then use these services to analyze their employees. The clients can:

The PlugH involves 3 main modules:

- Admin View
- HR View
- Survey

The project has an Admin view where various services are created and assigned to client companies and PlugH also provides a HR view which is completely dynamic and dependent on the services undertaken by the client.

There are various services being provided such as health promotions, toolkits, surveys and polls, newsletters, merchandise and so on.

2. INTRODUCTION

The World Health Organization defines health as a state of complete physical, mental, and social well-being, and not merely the absence of disease or infirmity.

A healthy workplace environment is ideal when it comes to maintaining a positive outcome in a stressful atmosphere. The most important thing that influences employee motivation and happiness, and how productive and efficient they can be, all goes down to their working environment. A healthy workplace environment is good for the company as it could lead to bringing sales for business. A healthy workplace environment improves productivity and reduces costs related to absenteeism, turnover, workers' compensation, and medical claims.

Healthy does not only constitute as needing medical attention, it could also be the outlook of the workplace and even the surroundings. There are four aspects to look into when creating a healthy workplace environment:

- Workplace culture
- Physical environment and occupational health & safety
- Health and lifestyle practice
- Supportive workplace environment

The project "**PlugH**" focuses on creating such healthier atmosphere for the employees of an organization, the website is developed in order to create and lend services and ideas that in turn help the client organizations to create healthier work environments for their employees.

The Admin module offers the admin view where all the packages are created modified assigned and also viewed. It is also responsible for the creation of services that are provided in the packages. The client selects the services which they want in their packages.

The HR module is portal for the HR to login and avail the services being lent, in HR portal the services will be displayed according to the package purchased by the client.

The survey module is used to create surveys and polls that is one among the services being provided, the creation of the survey is taken care by the admin whereas the scheduling and conduction is taken care by the client through HR view.

2.1 WORKFLOW OF PROJECT:

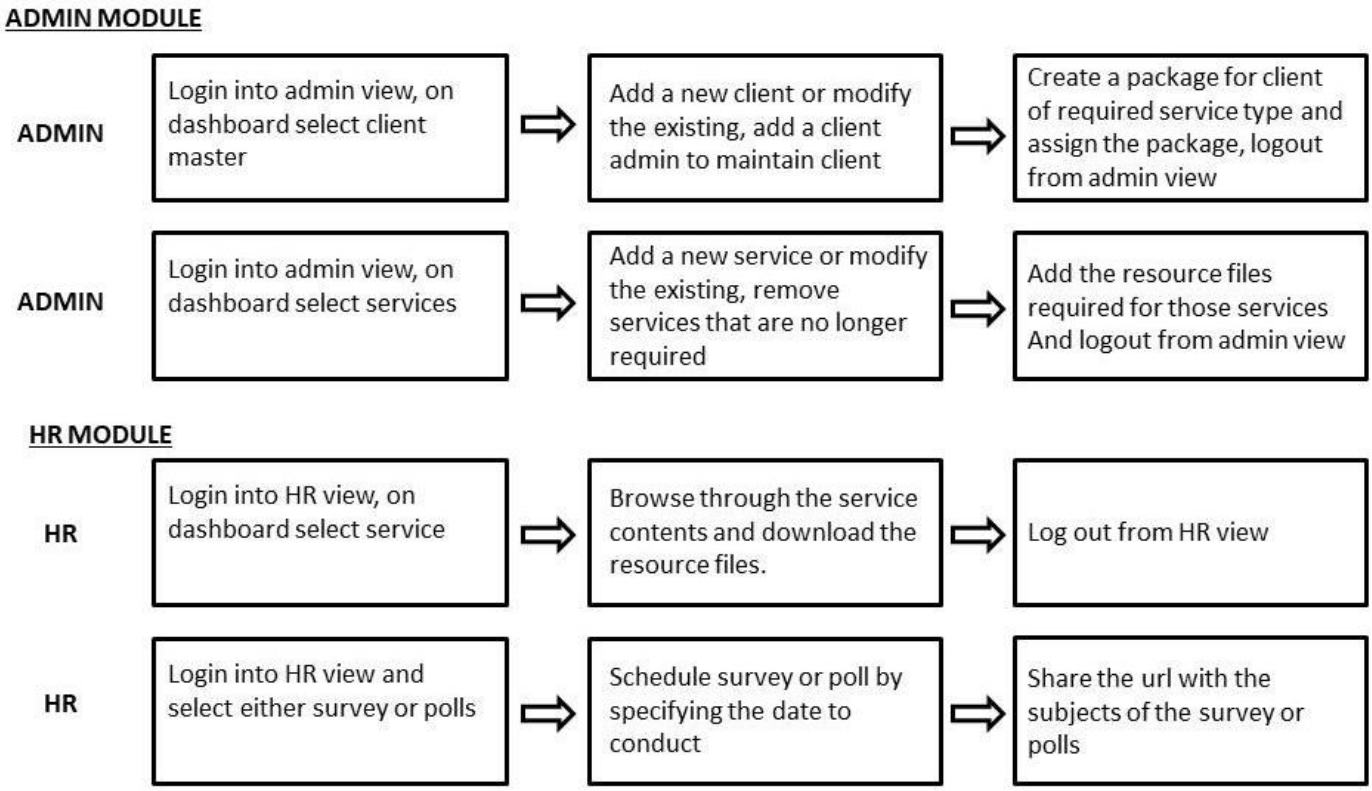


Fig 2.1 Workflow of the project

2.2 PROBLEM STATEMENT:

Today's stressful corporate life is so packed with loads of works that burdens the minds of those who are into it and becomes the root for most of the unhealthy conditions they develop both physically and mentally. The stress is such, an individual by the end of his/her twenties and by his/her mid-thirties would have issues like blood pressure, hypertension, and cholesterol hence so hence forth.

Therefore the main issue is having a platform for helping companies and organizations to deal with such issues their employees undergo more importantly to prevent them.

2.3 OBJECTIVE OF PROJECT:

The main objective of the project "PlugH" is to turn a workplace environment into healthy workplace environment and helps it to stay that way.

The project is built at both administrative end and user's (client) end but only the administrator is guaranteed the complete access whereas the user's (client) end is permitted only to read the contents and restricted to write any data or even append existing. The purpose of the project is to build a web-app that helps in process of converting the workplace environment into a healthier workplace environment. The project supports all the operations from creation to modification and removal of services that can be utilized by the organizations.

- It provides the facility to the clients who want to know how to turn their workplace healthier.
- It provides facility to the clients to know the potential risks and issues they might get due to their current workplace environment.
- It provides the full details about the services like type and related information.
- With the help of it we can improve productivity and reduce costs related to absenteeism, turnover, workers' compensation, and medical claims.
- The most important thing that influences employee motivation and happiness, and how productive and efficient they can be, all goes down to their working environment.
- It is ideal when it comes to maintaining a positive outcome in a stressful atmosphere. Deal with problems as soon as possible and regularly ask for feedback on how your workplace could be improved.

2.4 PROJECT SCOPE:

It may help collecting perfect management in details. In a very short time, the collection will be obvious, simple and sensible.

- Satisfies user requirement.
- Easy to understand by the user and operator.
- Easy to operate.
- Good user interface.
- Be expandable.

2.5 METHODOLOGY:

2.5.1 Agile model:

Agile SDLC model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. Every iteration involves cross functional teams working simultaneously on various areas like:

- Planning
- Requirements Analysis
- Design
- Coding
- Unit Testing and
- Acceptance Testing

At the end of the iteration, a working product is displayed to the customer and important stakeholders.

What is Agile?

Agile model believes that every project needs to be handled differently and the existing methods need to be tailored to best suit the project requirements. In Agile, the tasks are divided to time boxes (small time frames) to deliver specific features for a release. Iterative approach is taken and working software build is delivered after each iteration. Each build is incremental in terms of features; the final build holds all the features required by the customer.

Here is a graphical illustration of the Agile Model:

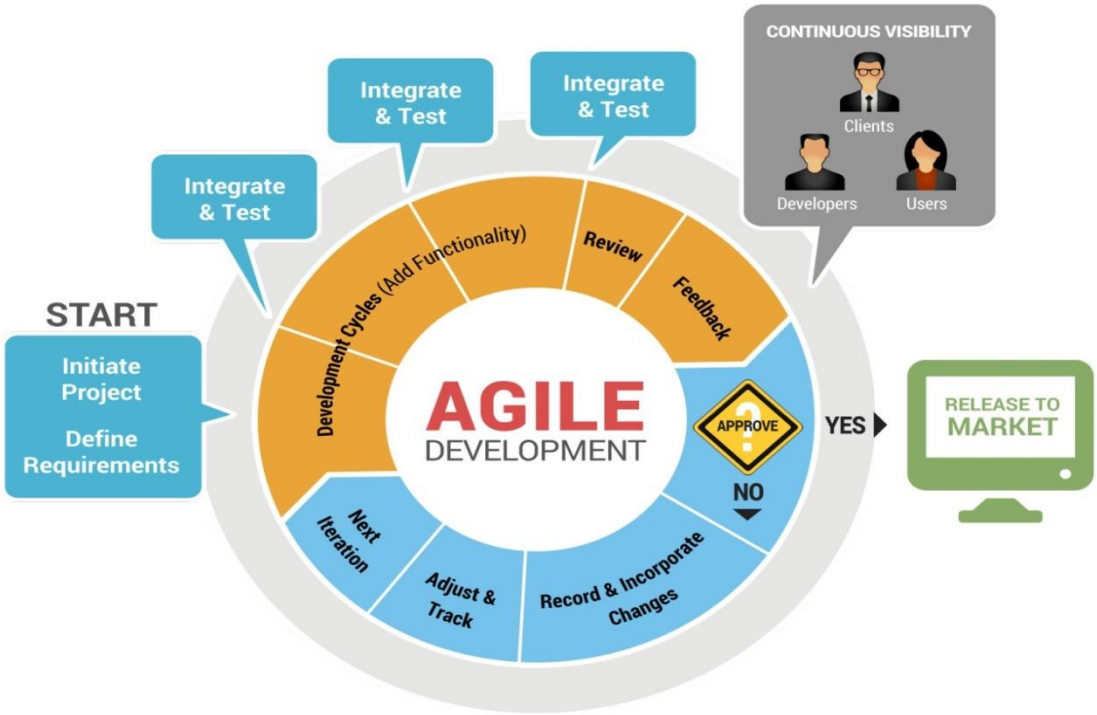


Fig 2.5.1 Different Stages in Agile Development:

The Agile thought process had started early in the software development and started becoming popular with time due to its flexibility and adaptability.

Agile Vs. Traditional SDLC Models

Agile is based on the adaptive software development methods, whereas the traditional SDLC models like the waterfall model is based on a predictive approach. Predictive teams in the traditional SDLC models usually work with detailed planning and have a complete forecast of the exact tasks and features to be delivered in the next few months or during the product life cycle.

Predictive methods entirely depend on the requirement analysis and planning done in the beginning of cycle. Any changes to be incorporated go through a strict change control management and prioritization.

Agile uses an adaptive approach where there is no detailed planning and there is clarity on future tasks only in respect of what features need to be developed. There is feature driven development and the team adapts to the changing product requirements dynamically.

The product is tested very frequently, through the release iterations, minimizing the risk of any major failures in future.

Customer Interaction is the backbone of this agile methodology, and open communication with minimum documentation are the typical features of agile development environment. The agile teams work in close collaboration with each other and are most often located in the same geographical location.

Agile Model - Pros and Cons

The advantages of the Agile Model are as follows:

- It is a very realistic approach to software development.
- Promotes teamwork and cross training.
- Functionality can be developed rapidly and demonstrated.
- Suitable for fixed or changing requirements
- Delivers early partial working solutions.
- Good model for environments that change steadily.
- Minimal rules, documentation easily employed.
- Enables concurrent development and delivery within an overall planned context.
- Little or no planning required, Easy to manage.
- Gives flexibility to developers.

The disadvantages of the Agile Model are as follows:

- Not suitable for handling complex dependencies.
- More risk of sustainability, maintainability and extensibility.
- An overall plan, an agile leader and agile PM practice is a must without which it will not work.
- Strict delivery management dictates the scope, functionality to be delivered, and adjustments to meet the deadlines.
- Depends heavily on customer interaction, so if customer is not clear, team can be driven in the wrong direction.
- There is a very high individual dependency, since there is minimum documentation generated.
- Transfer of technology to new team members may be quite challenging due to lack of documentation.

LIMITATIONS:

Sometimes You Can't Get What You Want

Clients will notified if the service they want is not available, whilst this can be annoying we will try and substitute your desired service for something of similar type and effect.

2.6 MODULE DESCRIPTION:

The project PlugH web-app has three main modules:

- Admin Module
- HR Module
- Survey Module

Admin Module:

- This module includes the login details of client, services created and assigned for the client.
- Company will create and register every client by providing accessing authority.
- Admin can view the details of various modules like HR View and Survey.
- Services are assigned to the clients based on their requirements.
- Admin can modify the services created and assigned for the clients.
- In admin view, admin will add user details and in admin view we have following sub modules given below:
    - Login
    - User
    - Client Master
    - Master
    - Service
    - Publishing
    - Ticket

Login Module:

The User has to register to login and the OTP (One Time Password) generated randomly and send to user email address to verify OTP to complete login. Once user registers the admin will approve his/her registration and fetch the services and the other details.

The user can create his

- User should able to login only after authenticated by the Staff/admin.
- User should able to login only with OTP send by Staff/admin after activation.
- Application has been set session.

User Module:

In User module, we are going to add user details like user name, role, email and phone number. After adding the user details in database, OTP(One Time Password) will generate via email and based on OTP user will login.

Client Master:

In Client Master, admin will add corporate details and Create Client Package. It has the following operations:

Client Profile:

In Client Profile we are going to add corporate details which consists company details, corporate Contact details and company branch facilities.

Client Admin:

We are going to add Client admin details which consists company name, admin display name, contact number and login-ID.

Create Client Package:

We are going to create the client package, In client package module we are fetching the company name from client profile and there will be a start date and end date, in a specified duration each company will select the service which they want and specify the number of services and select the services multiple time in given time but for survey we are going to select the number of services only once in a specified duration later we can't create the client package.

Master:

In master module we have sub modules they are:

Category:

In this sub module we are going to add category name and connect you to customers searching for the services you offer like addictive habits, demographics, eating habits etc.

Target Audience:

A target audience is the intended audience, it is a particular group of consumers within the predetermined target market, Businesses that have a wide target market will focus on a specific target audience for certain messages to send, such as Blue Collared Workers, Desk Workers, Frequent, Travellers, Graveyard Shift Workers, Long Standing Hours, Shift Workers, etc..

Occupational Hazard:

Occupational hazards are risks of illnesses or accidents in the workplace. In occupational hazard we are adding risks and its description.

Additional:

In additional master we are going to add the following information that are used by admin:

- Language
- Area of Expertise
- Industry
- Branch Facilities
- Merchandise Category.

Service:

In Service Module we are providing number of services like:

Survey:

In this sub module we are conducting survey for employees, The process involves asking people for information through a questionnaire, and questionnaire will be like single text, radio buttons, check box, dropdown, etc..

Polls:

Polls it is a process of finding out what people think about something by asking many people the same question like singe line questions.

Partner:

In Partner we have sub modules they are **Partner-Individual** and **Partner-Organisation**.

- **Partner-Individual** who joins with other individuals (partners) in an arrangement (partnership) and in this module we are adding Partner-Individual details such as Social-Media Presence, Service Area, Languages Known, Audience Comfortable with and Area of Expertise.
- **Partner-Organisation** is defined as an association of two or more persons to carry on as coowners of a business for profit, In Partner-Organisation we are adding adding Partner-Organisation details such as Social-Media Presence, Service Location, Area of Expertise and Contact Person Details.

Health Promotion:

Health Promotion is one of the service for client, where health promotion improves health behaviors, reduces the health Care Cost, improves employees health behaviors, etc., and they offer some of the health promotion for clients.

In this module we are adding Health Promotion details, External Links and Files.

Newsletter:

Newsletters include periodic updates, news, promotions and events communicated in print. While newsletters are useful in getting attention from prospects, their primary importance is in maintaining ongoing connections with contacts and existing customers. Companies send newsletters weekly, monthly or quarterly, depending on their goals. In newsletter we are going to update the news based on choosing the category, target-audience, occupational hazard. Based on this admin will update the news with description.

Toolkit:

Toolkit is one of the service for client, It is a set of tools can help you get more out of your innovation activities by providing you with concrete ways for managing different parts of innovation. The right tools allow you to consider different aspects of innovation, conducting the innovative Ideas.

Bite Size Ideas:

This module is used for getting the new ideas for clients and it is one of the service provided for clients. In Bite-size Ideas we are going to give new ideas with name and description by choosing category, target-audience and occupational hazard.

Merchandise-Master:

Merchandise means providing wholesale service to the clients like different types of categories, T-shirt, Mug, etc., for multiple events, In merchandise master we are going to add merchandise name, remarks and we also upload different types of files formats like pdf, jpg, png and cdr or ai.

Webinar:

Webinar is a seminar conducted online. With a webinar, share your presentation with your audience without physically being there. Live webinars happen in real-time and allow for much more interactivity as the audience can ask questions at any time and get an answer.

In this webinar we are providing seminar (meeting) name and description with specified date and time.

Publishing:

In publishing we have sub modules they are:

- **Health tip**: is one of the sub module in publishing, here we are going to add tip for the day like drink some water, especially before meals, get enough sleep, etc..
- **Health days:** In this module we are going provide important days based on date, day, write up and by uploading file like world health day, World Autism Awareness Day, International Yoga Day, etc.,

Ticket:

This module is used to raise the ticket. Like if we are facing any problem we can raise a ticket like by selecting the company name by which company they got ticket, and specify what the problem is by selecting the ticket type and with remarks, based on that the ticket will be solved however only one can raise a ticket either in Admin view or HR view.

Report:

In this we module we prepare a report in the client's required format regarding the packages and services.

HR Module:

- This Module is for the HR side representation of the services taken by the client.
- Here client can login and use the services taken by them from the company.
- The client can view the details of the services and assigned for them.
- The client can conduct polls and surveys to get information from their employees
- Browse tips and ideas related to the services bought by them.

In HR view we have following sub modules given below:

- Login
- Dashboard
- HCS Service
- Publishing
- Ticket

Login Module:

The HR has to register to login and the OTP (One Time Password) generated randomly and send to email address to verify OTP to complete login. Once he will register the admin will approve his registration and fetch the services and the other details.

- HR should login based on Company name, logo, Username and Password.
- HR should able to login only after authenticated by the Staff/admin.
- HR should able to login only with OTP send by Staff/admin after activation.
- Application has been set session.

Dashboard:

In dashboard we are going to display company name, company logo health tip, health days and all the PlugH services. In all the services we are displaying number of services scheduled and number of services allocated. In this dashboard we have sub modules they are:

- **Health Tip:** In the dashboard we are going to display(fetch) the number of tips which are added in admin. And in HR view we are going to display tips for the day and those tips which are displayed are not going to display more than once.
- **Health days**: Health days is nothing but important days which we are going to display all the important days which are added from admin side with specified date.

HCS Service

In HCS service we are fetching different services they are:

Survey:

In this module we are going to fetch the survey response, we are going to display all the surveys and we are also going to display the number of surveys allocated(scheduled), In that we are going to show, the number of surveys are active and number of surveys pending.

And In this module we are scheduling the survey based on date. These details will be displayed based on client package which has been done in admin, if not it will displayed as not scheduled.

Polls:

In this module we are going to fetch the polls, we are going to display only purchased polls and we are also going to display the number of polls allocated(scheduled), In we are going to show, the number of polls are active and number of polls pending. And In this module we are scheduling the polls based on date. These details will be displayed based on client package which has been done in admin, if not it will displayed as not scheduled.

Partner:

In this module we are going to display all partner directory i.e. partner-individual and partner-organization. We are display the number of partner directory allocated(scheduled), In that we are going to show, the number of partner directory are active and number of partner directory pending. These details will be displayed based on client package which has been done in admin, if not it will displayed as not scheduled.

Health Promotion:

In this module we are displaying all the health promotion details from admin which are added. Viewing this health promotion details HR will be going to use this service and based on specified company, the service details will be displayed, we can also download the uploaded files which are uploaded from admin side and display the External links given from admin side. And We are display the number of health promotion allocated (scheduled), In that we are going to show, the number of health promotion are active and number of health promotion pending. This details will be displayed based on client package which has been done in admin, if not it will displayed as not scheduled.

Newsletter:

In this module we are going to display all the newsletter details from admin which are added. By viewing this newsletter details HR will be going to use this service and based on specified company, the service details will be displayed, we can also download the uploaded files which are uploaded from admin side. And we display the number of newsletter allocated (scheduled), In that we are going to show, the number of newsletter are active and number of newsletter pending. This details will be displayed based on client package which has been done in admin, if not it will displayed as not scheduled.

Toolkit:

In this module we are going to display all the toolkit details from admin which are added. By viewing this toolkit details HR will be going to use this service and based on specified company, the service details will be displayed, we can also download the uploaded files which are uploaded from admin side. And we display the number of toolkit allocated (scheduled), In that we are going to show, the number of toolkit are active and number of toolkit pending. These details will be displayed based on client package which has been created in admin, if not it will displayed as not scheduled.

Bite Size Ideas:

In this module we are going to display all the bite size ideas details from admin which are added. By viewing this bite size ideas details, HR will be going to use this service and based on specified company, the service details will be displayed, we can also download the uploaded files which are uploaded from admin side. And we are display the number of bite size ideas allocated (scheduled), In that we are going to show, the number of bite size ideas are active and number of bite size ideas pending. These details will be displayed based on client package which has been done in admin, if not it will displayed as not scheduled.

Merchandise-Master:

In this module we are going to display all the merchandise details and the images from admin which are added. By viewing this merchandise details, HR will be going to use this service and based on specified company, the service details will be displayed, we can also download the uploaded files which are uploaded from admin side. And we display the number of merchandise allocated (scheduled), In that we are going to show, the number of merchandise are active and number of merchandise pending.

These details will be displayed based on client package which has been done in admin, if not it will displayed as not merchandise.

Webinar:

In this module we are going to display all the webinar details with specified date and time from admin which are added. By viewing this webinar details, HR will be going to use this service and based on specified company, the service details will be displayed. We are display the number of webinar allocated (scheduled), In that we are going to show, the number of webinar are active and number of webinar pending. These details will be displayed based on client package which has been done in admin.

Ticket:

This module is used to raise the ticket. Like if we are facing any problem we can raise a ticket like by selecting the company name by which company they got ticket, and specify what the problem is in ticket type and with remarks was. Based on the issue ticket will be solved.

This is similar to admin but only one can raise a ticket either in Admin view or HR view.

PLUGH DEMO:

A demo of HR view is shown to the HR.

Survey Module:

- In this module surveys and polls are created and assigned to the clients.
- The admin can also modify and delete the surveys and polls created.
- These surveys and polls are assigned client via client package.
- In HR view the HR will receive the survey and polls chosen by them and will then schedule them.

So as mentioned the survey module is again divided into two sub modules namely

Survey:

Here a survey is created which is related to certain aspects of health, which can be used by client organizations to analyze their employees. The survey can also be used collect information from the customers of the client organizations too. HR will be the mentor of the survey.

In the admin side the survey is created and it is of 2 types namely:

- **Anonymous survey:** Here details regarding the survey participants are not stored instead only their responses and the time of response is stored.
- **Known:** Here the primary details of the survey participants such as their name, email and contact numbers are also fetched and stored.

The survey is flexible and provides options to ask questions of following types:

- **Single Text questions**: Questions that accept a word or phrase of text as response.
- **Multiple Choices via checkboxes:** Questions that provide multiple options among which **one or more** can be chosen.
- **Multiple Choices via radio buttons:** Questions that provide multiple options among which **only one** can be chosen.
- **Date questions:** Questions that accepts dates as responses and these question are of 3 types:
  - Dates with No restriction.
  - Dates with restriction on past days.
  - Dates with restriction on future days.
- **Dropdown questions:** Questions that provides multiple options through dropdown menu.
- **Rating scale:** Questions for rating a product, environment, facilities or services.
- **Comment question:** Question that accepts one or more phrases or paragraphs i.e, briefly explained responses.

Polls:

A poll is similar to a survey however the poll contains a single question that in turn accepts either Yes or No as a response.

3. LITERATURE SURVEY

EXISTING SYSTEM:

The main concern is with reducing the unhealthy workplace environment (both mentally and physically) thus the employees have a less stressed and happy environment to work in. A web platform that helps in creating and providing such services is less in number and most of the platforms only provide information regarding such services and operations are carried by either stand-alone system applications or in traditional methodologies i.e., by papers.

PROPOSED SYSTEM:

The main advantage of this project is it is a web-platform hence it is not system dependent. The services are easily accessible by the clients through this portal and are efficient in providing them the required resources.

Each client is assigned with a admin to maintain their package and services offered to them and they also maintain the resources allocated. The admin responds to their queries and resolves issues through tickets.

The client specific approach and modules are developed in such a way that no data is shared outside the portal and clients can access only those that they have purchased through their package.

4. TECHNOLOGY USED

4.1.1 SCRIPTING LANGUAGE:

PHP

PHP is a general-purpose scripting language widely used as a server-side language for creating dynamic web pages. Though its reputation is mixed, PHP is still extremely popular and is used in over 75% of all websites where the server-side programming language is known.

PHP code is usually processed on a web server by a PHP interpreter implemented as a module, a daemon or as a Common Gateway Interface (CGI) executable. On a web server, the result of the interpreted and executed PHP code – which may be any type of data, such as generated HTML or binary image data – would form the whole or part of a HTTP response. Various web template systems, web content management systems, and web frameworks exist which can be employed to orchestrate or facilitate the generation of that response. Additionally, PHP can be used for many programming tasks outside of the web context, such as standalone graphical applications and robotic drone control. Arbitrary PHP code can also be interpreted and executed via command-line interface (CLI).

Features:

- HTTP authentication with PHP
- Cookies
- Sessions
- Dealing with XForms
- Handling file uploads
- Using remote files
- Connection handling
- Persistent Database Connections
- Safe Mode
- Command line usage — Using PHP from the command line
- Garbage Collection
- DTrace Dynamic Tracing

Object Orientation Programming:

Object Oriented Programming (OOP) is a concept where the programmer thinks of the program in "objects" (however abstract the objects may be) that interact with each other. In OOP, all the code associated with that object is in one place.

Procedural programming is about writing procedures or functions that perform operations on the data, while object-oriented programming is about creating objects that contain both data and functions.

Object-oriented programming has several advantages over procedural programming:

- OOP is faster and easier to execute
- OOP provides a clear structure for the programs
- OOP helps to keep the PHP code DRY "Don't Repeat Yourself", and makes the code easier to maintain, modify and debug
- OOP makes it possible to create full reusable applications with less code and shorter development time

4.1.2 FRAMEWORK:

CODEIGNITER

CodeIgniter is an Application Development Framework - a toolkit - for people who build web sites using PHP. Its goal is to enable you to develop projects much faster than you could if you were writing code from scratch, by providing a rich set of libraries for commonly needed tasks, as well as a simple interface and logical structure to access these libraries. CodeIgniter lets you creatively focus on your project by minimizing the amount of code needed for a given task.

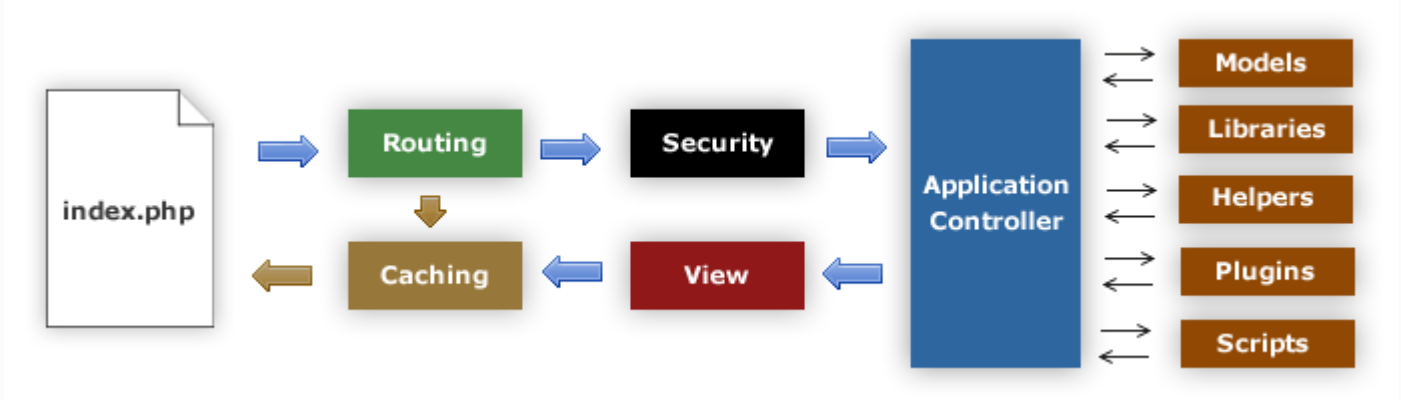
Supported Features:

- Model-View-Controller Based System
- Extremely Light Weight
- Full Featured database classes with support for several platforms.
- Query Builder Database Support
- Form and Data Validation
- Security and XSS Filtering
- Session Management
- Email Sending Class. Supports Attachments, HTML/Text email, multiple protocols (sendmail, SMTP, and Mail) and more.
- Image Manipulation Library (cropping, resizing, rotating, etc.). Supports GD, ImageMagick, and NetPBM

- File Uploading Class
- FTP Class
- Localization
- Pagination
- Data Encryption
- Benchmarking
- Full Page Caching
- Error Logging
- Application Profiling
- Calendaring Class
- User Agent Class
- Zip Encoding Class
- Template Engine Class
- Trackback Class
- XML-RPC Library
- Unit Testing Class
- Search-engine Friendly URLs
- Flexible URI Routing
- Support for Hooks and Class Extensions
- Large library of “helper” functions

Application Flow Chart

The following graphic illustrates how data flows throughout the system:



The index.php serves as the front controller, initializing the base resources needed to run CodeIgniter.

The Router examines the HTTP request to determine what should be done with it.

If a cache file exists, it is sent directly to the browser, bypassing the normal system execution.

Security. Before the application controller is loaded, the HTTP request and any user submitted data is filtered for security.

The Controller loads the model, core libraries, helpers, and any other resources needed to process the specific request.

The finalized View is rendered then sent to the web browser to be seen. If caching is enabled, the view is cached first so that on subsequent requests it can be served.

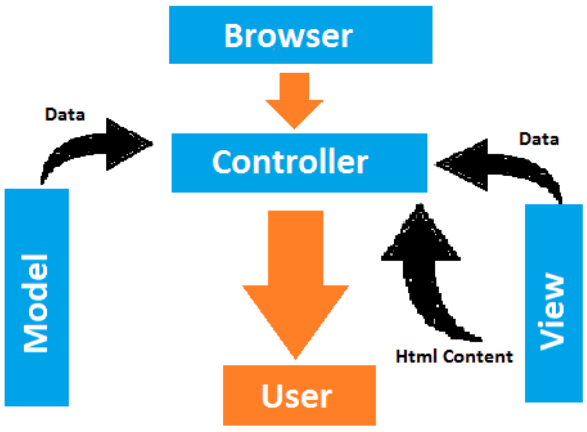
Model-View-Controller

CodeIgniter is based on the **Model-View-Controller** development pattern. MVC is a software approach that separates application logic from presentation. In practice, it permits your web pages to contain minimal scripting since the presentation is separate from the PHP scripting.

The Model represents your data structures. Typically your model classes will contain functions that help you retrieve, insert, and update information in your database.

The View is the information that is being presented to a user. A View will normally be a web page, but in CodeIgniter, a view can also be a page fragment like a header or footer. It can also be an RSS page, or any other type of "page".

The Controller serves as an intermediary between the Model, the View, and any other resources needed to process the HTTP request and generate a web page.



CodeIgniter has a fairly loose approach to MVC since Models are not required. If you don't need the added separation, or find that maintaining models requires more complexity than you want, you can ignore them and build your application minimally using Controllers and Views. CodeIgniter also enables you to incorporate your own existing scripts, or even develop core libraries for the system, enabling you to work in a way that makes the most sense to you.

4.1.3 BOOTSTRAP

Bootstrap is a free and open-source front-end library for designing websites and web applications. It contains HTML- and CSS-based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions.

Bootstrap is modular and consists of a series of less stylesheets that implement the various components of the toolkit. These stylesheets are generally compiled into a bundle and included in web pages, but individual components can be included or removed. Bootstrap provides a number of configuration variables that control things such as color and padding of various components.

Features

- Bootstrap 3 supports the latest versions of the Google Chrome, Firefox, Internet Explorer, Opera, and Safari (except on Windows). It additionally supports back to IE8 and the latest Firefox Extended Support Release (ESR).
- Since Bootstrap 2, the Bootstrap documentation has included a customization wizard which generates a customized version of Bootstrap based on the requested components and various settings.
- As of Bootstrap 4, Sass is used instead of less for the stylesheets.
- Each Bootstrap component consists of an HTML structure, CSS declarations, and in some cases accompanying JavaScript code.
- Grid system and responsive design comes standard with an 1170 pixel wide grid layout. Alternatively, the developer can use a variable-width layout. For both cases, the toolkit has four variations to make use of different resolutions and types of devices: mobile phones, portrait and landscape, tablets and PCs with low and high resolution. Each variation adjusts the width of the columns.

Stylesheets:

Bootstrap provides a set of stylesheets that provide basic style definitions for all key HTML components. These provide a uniform, modern appearance for formatting text, tables and form elements.

Re-usable components:

In addition to the regular HTML elements, Bootstrap contains other commonly used interface elements. The components are implemented as CSS classes, which must be applied to certain HTML elements in a page.

JavaScript components:

Bootstrap comes with several JavaScript components in the form of jQuery plugins. They provide additional user interface elements such as dialog boxes, tooltips, and carousels. They also extend the functionality of some existing interface elements, including for example an auto-complete function for input fields. In version 1.3, the following JavaScript plugins are supported: Modal, Dropdown, scroll spy, Tab, Tooltip, Popover, Alert, Button, Collapse, Carousel and type a head.

4.1.4 CSS:

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate CSS file, and reduce complexity and repetition in the structural content.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device.

The name cascading comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable.

Limitations

Some noted limitations of the current capabilities of CSS include:

Selectors are unable to ascend:

CSS currently offers no way to select a *parent* or *ancestor* of an element that satisfies certain criteria.CSS Selectors Level 4, which is still in Working Draft status, proposes such a selector, but only as part of the "complete" selector profile, not the "fast" profile used in dynamic CSS styling. A more advanced selector scheme (such as XPath) would enable more sophisticated style sheets. The major reasons for the CSS Working Group previously rejecting proposals for parent selectors are related to browser performance and incremental rendering issues.

Cannot explicitly declare new scope independently of position:

Scoping rules for properties such as z-index look for the closest parent element with a position: absolute or position: relative attribute. This odd coupling has undesired effects. For example, it is impossible to avoid declaring a new scope when one is forced to adjust an element's position, preventing one from using the desired scope of a parent element.

Pseudo-class dynamic behavior not controllable:

CSS implements pseudo-classes that allow a degree of user feedback by conditional application of alternate styles. One CSS pseudo-class, "hover", is dynamic (equivalent of JavaScript "onmouseover") and has potential for abuse (e.g., implementing cursor-proximity popups), but CSS has no ability for a client to disable it (no "disable"-like property) or limit its effects (no "no change"-like values for each property).

Cannot name rules:

There is no way to name a CSS rule, which would allow (for example) client-side scripts to refer to the rule even if its selector changes.

Cannot include styles from a rule into another rule:

CSS styles often must be duplicated in several rules to achieve a desired effect, causing additional maintenance and requiring more thorough testing. Some new CSS features were proposed to solve this, but (as of February, 2016) are not yet implemented anywhere.

Cannot target specific text without altering markup:

Besides the: first-letter pseudo-element, one cannot target specific ranges of text without needing to utilize place-holder elements.

4.1.5 HYPER TEXT MARKUP LANUGAGE:

HTML5 is a markup language used for structuring and presenting content on the World Wide Web. It is the fifth and current major version of the HTML standard.

It was published in October 2014 by the World Wide Web Consortium (W3C) to improve the language with support for the latest multimedia, while keeping it both easily readable by humans and consistently understood by computers and devices such as web browsers, parsers, etc. HTML5 is intended to subsume not only HTML 4, but also XHTML 1 and DOM Level 2 HTML.

HTML5 includes detailed processing models to encourage more interoperable implementations; it extends, improves and rationalizes the markup available for documents, and introduces markup and application programming interfaces (APIs) for complex web applications. For the same reasons, HTML5 is also a candidate for cross-platform mobile applications, because it includes features designed with low-powered devices in mind.

Many new syntactic features are included. To natively include multimedia and graphical content, the new <video>, <audio> and <canvas> elements were added, and support for scalable vector graphics (SVG) content and MathML for mathematical formulas.

To enrich the semantic content of documents, new page structure elements such as
<main>, <sections>, <article>, <header>, <footer>, <aside>, <nav> and <figure>, are
added. New attributes are introduced, some elements and attributes have been removed, and others such as <a>, <cite> and <menu> have been changed, redefined or standardized.

The APIs and Document Object Model (DOM) are now fundamental parts of the HTML5 specification and HTML5 also better defines the processing for any invalid documents.

HTML5 has a lot of features to offer. This article discussed some important features that you may find useful. We discussed the <audio> and <video> elements, <canvas> element, new input types, some form enhancements and custom data attributes. These are just a few of the features that HTML5 has to offer. Although these features are not supported equally in all the browsers, you can start using them today. Modern websites and development tools try to leverage the HTML5 features and knowing them is important for any web developer.

Advanced Features of HTML5 and CSS3:

- Editing content within the elements. You can edit content within an element by using an HTML attribute called content editable
- Form attribute required
- Regular expressions
- Figure tag
- SVG element
- Web fonts
- Gradient
- Animation and Transition.

4.1.6 JAVASCRIPT:

JavaScript is a lightweight, interpreted programming language. It is designed for creating network-centric applications. It is complimentary to and integrated with Java. JavaScript is very easy to implement because it is integrated with HTML. It is open and cross-platform.

JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

JavaScript was first known as LiveScript, but Netscape changed its name to JavaScript, possibly because of the excitement being generated by Java. JavaScript made its first appearance in Netscape 2.0 in 1995 with the name LiveScript. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other web browsers.

The ECMA-262 Specification defined a standard version of the core JavaScript language.

JavaScript is a lightweight, interpreted programming language.

- Designed for creating network-centric applications.
- Complementary to and integrated with Java.
- Complementary to and integrated with HTML.

Open and cross-platform.

Advantages of JavaScript:

The merits of using JavaScript are −

Less server interaction − You can validate user input before sending the page off to the server. This saves server traffic, which means less load on your server.

Immediate feedback to the visitors − They doesn't have to wait for a page reload to see if they have forgotten to enter something.

Increased interactivity − You can create interfaces that react when the user hovers over them with a mouse or activates them via the keyboard.

Richer interfaces −You can use JavaScript to include such items as drag-and-drop components and sliders to give a Rich Interface to your site visitors.

4.1.7 STRUCTURED QUERY LANGUAGE:

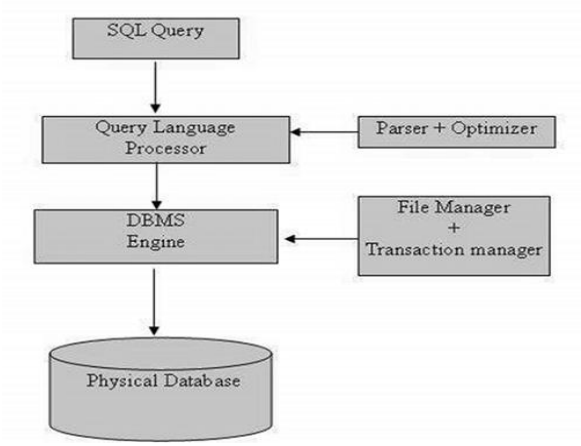
SQL is a database computer language designed for the retrieval and management of data in a relational database. SQL stands for Structured Query Language. SQL is a language to operate databases; it includes database creation, deletion, fetching rows, modifying rows, etc. SQL is an ANSI (American National Standards Institute) standard language, but there are many different versions of the SQL language.

SQL is the standard language for Relational Database System. All the Relational Database Management Systems (RDMS) like MySQL, MS Access, Oracle, Sybase, Informix, Postgres and SQL Server use SQL as their standard database language.

Also, they are using different dialects, such as −

- MS SQL Server using T-SQL,
- Oracle using PL/SQL,
- MS Access version of SQL is called JET SQL (native format) etc.
- SQL is widely popular because it offers the following advantages −
- Allows users to access data in the relational database management systems.
- Allows users to describe the data.
- Allows users to define the data in a database and manipulate that data.
- Allows to embed within other languages using SQL modules, libraries & pre-compilers.
- Allows users to create and drop databases and tables.
- Allows users to create view, stored procedure, functions in a database.
- Allows users to set permissions on tables, procedures and views.

SQL Architecture:



RDBMS:

RDBMS stands for Relational Database Management System. RDBMS is the basis for SQL, and for all modern database systems like MS SQL Server, IBM DB2, Oracle, MySQL, and Microsoft Access.

Table:

The data in an RDBMS is stored in database objects which are called as tables. This table is basically a collection of related data entries and it consists of numerous columns and rows.

4.1.8 AJAX:

Ajax (also AJAX short for "Asynchronous JavaScript And XML")is a set of web-development techniques using many Web technologies on the client side to create asynchronous Web applications. With Ajax, Web applications can send and retrieve data from a server asynchronously (in the background) without interfering with the display and behavior of the existing page. By decoupling the data interchange layer from the presentation layer, Ajax allows Web pages, and by extension Web applications, to change content dynamically without the need to reload the entire page. In practice, modern implementations commonly utilize JSON instead of XML due to the advantages of JSON being native to JavaScript.

Ajax is not a single technology, but rather a group of technologies. HTML and CSS can be used in combination to mark up and style information. The webpage can then be modified by JavaScript to dynamically display – and allow the user to interact with — the new information. The built-in XML Http Request object within JavaScript is commonly used to execute Ajax on webpages allowing websites to load content onto the screen without refreshing the page. Ajax is not a new technology, or different language, just existing technologies used in new ways.

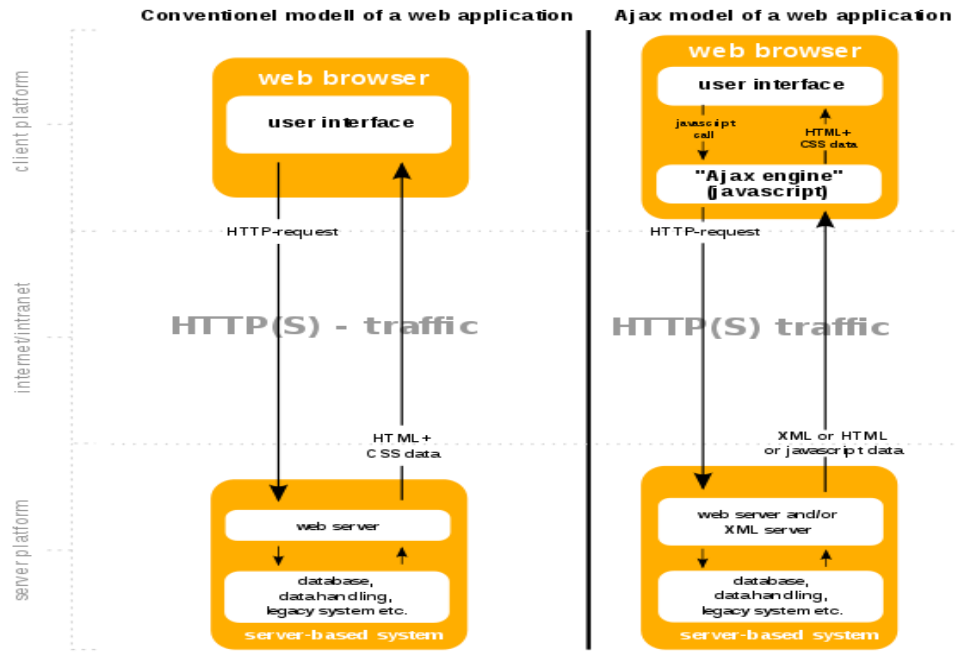
The term Ajax has come to represent a broad group of Web technologies that can be used to implement a Web application that communicates with a server in the background, without interfering with the current state of the page. In the article that coined the term Ajax, Jesse James Garrett explained that the following technologies are incorporated:

HTML (or XHTML) and CSS for presentation:

- The Document Object Model (DOM) for dynamic display of and interaction with data
- JSON or XML for the interchange of data, and XSLT for its manipulation
- The XML Http Request object for asynchronous communication
- JavaScript to bring these technologies together

Since then, however, there have been a number of developments in the technologies used in an Ajax application, and in the definition of the term Ajax itself. XML is no longer required for data interchange and, therefore, XSLT is no longer required for the manipulation of data. JavaScript Object Notation (JSON) is often used as an alternative format for data interchange, although other formats such as preformatted HTML or plain text can also be used.[16] A variety of popular JavaScript libraries, including JQuery, include abstractions to assist in executing Ajax requests.

Asynchronous HTML and HTTP (AHAH) involves using XML HTTP Request to retrieve (X)HTML fragments, which are then inserted directly into the Web page.



The conventional model for a Web Application and an application using Ajax.

4.2 HARDWARE AND SOFTWARE SPECIFICATION:

4.2.1 SOFTWARE REQUIREMENT SPECIFICATION

It specifies the hardware and software requirements that are required in order to run the application properly. The Software Requirement Specification (SRS) is explained in detail, which includes overview of this dissertation as well as the functional and non-functional requirement of this dissertation.

Functional Requirements:

Functional Requirement defines a function of a software system and how the system must behave when presented with specific inputs or conditions. These may include calculations, data manipulation and processing and other specific functionality. In this system following are the functional requirements: -

Admin Module:

The admin has to add users and clients through registered emails Ids, and must assign them their respective roles as well as tasks to be performed.

Administrator should be able to verify and authenticate the end user and client as well.

Admin will provide the decrypt key to the user.

Admin should have control to activate or deactivate the user and client.

Once account is activated by admin, the application will send the secret key via mail.

After receiving the file request from the user, admin will generate master key for encrypt and secret key for decrypt via mail.

Create services for clients and assign them to the client who purchases those services in their packages.

HR Module:

The HR must be capable of conducting surveys in specific days as scheduled by them.

The HR must be capable of accessing the resources provided to them.

Non – Functional Requirements:

Non – Functional requirements, as the name suggests, are those requirements that are not directly concerned with the specific functions delivered by the system. They may relate to emergent system properties such as reliability response time and store occupancy. Alternatively, they may define constraints on the system such as the capability of the Input Output devices and the data representations used in system interfaces. Many non-functional requirements relate to the system as whole rather than to individual system features. This means they are often critical than the individual functional requirements. The following non-functional requirements are worthy of attention.

The key non-functional requirements are:

1) **Security:** The system should allow secure communication between cloud server & user.
2) **Platform Independence:** The application should run on any platform without recompilation.
3) **Reliability:** The system should be reliable and must not degrade the performance of the existing system and should not lead to the hanging of the system.
4) **Response time:** The application response time should be quick.
5) **Scalability:** The system should provide optimal performance even if the user base grows dramatically.
6) **Maintainability:** The system should support incorporation of future requirements easily.
7) **Robustness:** The application should be fault tolerant with respect to illegal user/receiver inputs. Error checking has been built in the system to prevent system failure.

Preliminary Investigation:

The first and foremost strategy for development of a project starts from the thought of designing a mail enabled platform for a small firm in which it is easy and convenient of sending and receiving messages,

there is a search engine ,address book and also including some entertaining games. When it is approved by the organization and our project guide the first activity, i.e. Preliminary investigation begins.

The activity has three parts:

1. Request Clarification
2. Feasibility Study
3. Request Approval

Request Clarification:

After the approval of the request to the organization and project guide, with an investigation being considered, the project request must be examined to determine precisely what the system requires.

Here our project is basically meant for users within the company whose systems can be interconnected by the Local Area Network (LAN). In today's busy schedule man need everything should be provided in a readymade manner. So taking into consideration of the vastly use of the net in day to day life, the corresponding development of the portal came into existence.

Feasibility Analysis:

An important outcome of preliminary investigation is the determination that the system request is feasible. This is possible only if it is feasible within limited resource and time. The different feasibilities that have to be analyzed are:

1. Operational Feasibility
2. Economic Feasibility
3. Technical Feasibility

Operational Feasibility:

Operational Feasibility deals with the study of prospects of the system to be developed. This system operationally eliminates all the tensions of the Admin and helps him in effectively tracking the project progress. This kind of automation will surely reduce the time and energy, which previously consumed in manual work. Based on the study, the system is proved to be operationally feasible.

Economic Feasibility:

Economic Feasibility or Cost-benefit is an assessment of the economic justification for a computer-based project. As hardware was installed from the beginning & for lots of purposes thus the cost on project of hardware is low. Since the system is a network based, any number of employees connected to the LAN within that organization can use this tool from at any time. The Virtual Private Network is to be developed using the existing resources of the organization. So, the project is economically feasible.

Technical Feasibility:

According to Roger S. Pressman, Technical Feasibility is the assessment of the technical resources of the organization. The organization needs IBM compatible machines with a graphical web browser connected to the Internet and Intranet. The system is developed for platform Independent environment. Java Server Pages, JavaScript, HTML, SQL server and Web Logic Server are used to develop the system. The technical feasibility has been carried out. The system is technically feasible for development and can be developed with the existing facility.

Request Approval:

Not all request projects are desirable or feasible. Some organization receives so many project requests from client users that only few of them are pursued. However, those projects that are both feasible and desirable should be put into schedule. After a project request is approved, it cost, priority, completion time and personnel requirement is estimated and used to determine where to add it to any project list. Truly speaking, the approval of those above factors, development works can be launched.

4.2.2 SYSTEM-REQUIREMENTS:

Hardware Requirements:

RAM – 4GB

Processor – Intel i3

Hard Disk Space – Above 200GB

Software Requirements

Operating System - Windows 10 Server

Database – SQL Server

Platform – PHP

Framework - Codeigniter

Tools - Java Script, HTML, CSS, AJAX, Bootstrap

Web browser – Chrome, Mozilla.

5. DESIGN AND IMPLEMENTATIONS

Introduction:

Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application. Design is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity that will later be built. Beginning, once system requirement has been specified and analyzed, system design is the first of the three technical activities -design, code and test that is required to build and verify software.

The importance can be stated with a single word "Quality". Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess for quality. Design is the only way that we can accurately translate a customer's view into a finished software product or system. Software design serves as a foundation for all the software engineering steps that follow. Without a strong design we risk building an unstable system – one that will be difficult to test, one whose quality cannot be assessed until the last stage.

During design, progressive refinement of data structure, program structure, and procedural details are developed reviewed and documented. System design can be viewed from either technical or project management perspective. From the technical point of view, design is comprised of four activities – architectural design, data structure design, interface design and procedural design

System Workflow:

Design is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity that will later be built. The importance can be stated with a single word "Quality". Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess for quality. Design is the only way that we can accurately translate a customer's view into a finished software product or system.

Data Flow Diagrams:

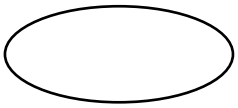
A data flow diagram is graphical tool used to describe and analyze movement of data through a system. These are the central tool and the basis from which the other components are developed. The transformation of data from input to output, through processed, may be described logically and independently of physical components associated with the system. These are known as the logical data flow

diagrams. The physical data flow diagrams show the actual implements and movement of data between people, departments and workstations.

DFD Symbols:

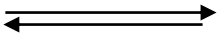
In the DFD, there are four symbols

- A square defines a source(originator) or destination of system data
- An arrow identifies data flow. It is the pipeline through which the information flows
- A circle or a bubble represents a process that transforms incoming data flow into outgoing data flows.
- An open rectangle is a data store, data at rest or a temporary repository of data.

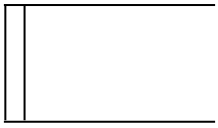


Process that transforms data flow

Source or Destination of data



Data flow



Data Store

Constructing A DFD:

Several rules of thumb are used in drawing DFD'S:

Process should be named and numbered for an easy reference. Each name should be representative of the process.

The direction of flow is from top to bottom and from left to right. Data traditionally flow from source to the destination although they may flow back to the source. One way to indicate this is to draw long flow line back to a source. An alternative way is to repeat the source symbol as a destination. Since it is used more than once in the DFD, it is marked with a short diagonal.

When a process is exploded into lower level details, they are numbered.

The names of data stores and destinations are written in capital letters. Process and dataflow names have the first letter of each work capitalized.

A DFD typically shows the minimum contents of data store. Each data store should contain all the data elements that flow in and out.

Questionnaires should contain all the data elements that flow in and out. Missing interfaces redundancies and like is then accounted for often through interviews.

Salient Features of DFD'S:

The DFD shows flow of data, not of control loops and decision are controlled considerations do not appear on a DFD.

The DFD does not indicate the time factor involved in any process whether the dataflow take place daily, weekly, monthly or yearly.

The sequence of events is not brought out on the DFD.

Types of Data Flow Diagrams

- Current Physical
- Current Logical
- New Logical
- New Physical

Rules Governing the DFD'S:

Process:

- No process can have only outputs.
- No process can have only inputs.
- If an object has only inputs than it must be a sink.
- A process has a verb phrase label.

Data Store:

- Data cannot move directly from one data store to another data store, a process must move data.
- Data cannot move directly from an outside source to a data store, a process, which receives, must move data from the source and place the data into data store
- A data store has a noun phrase label.

Source (or) Sink:

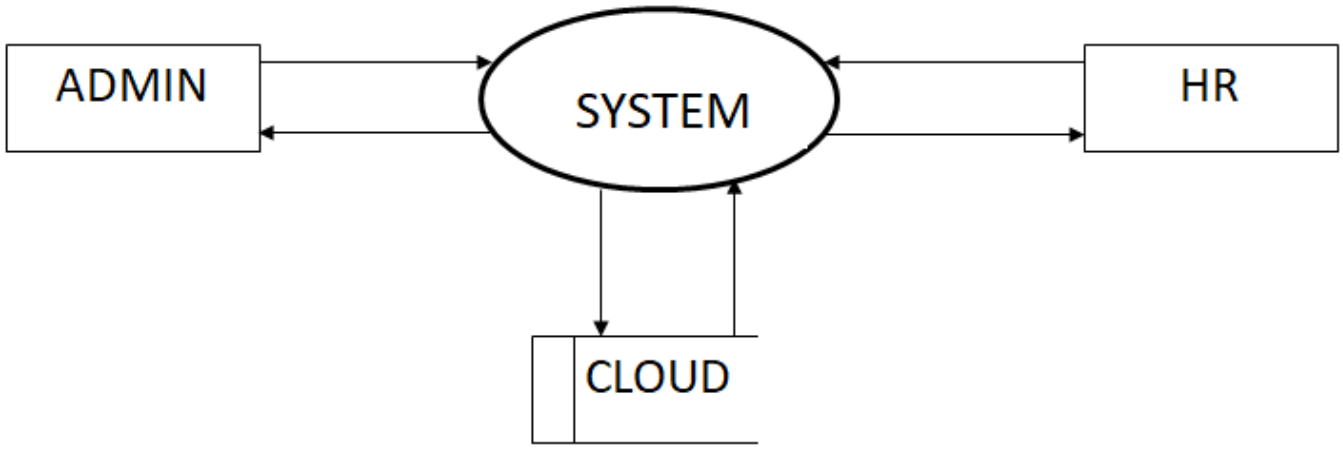
- The origin and/or destination of data.
- Data cannot move direly from a source to sink it must be moved by a process.
- A source and /or sink has a noun phrase land.

Data Flow:

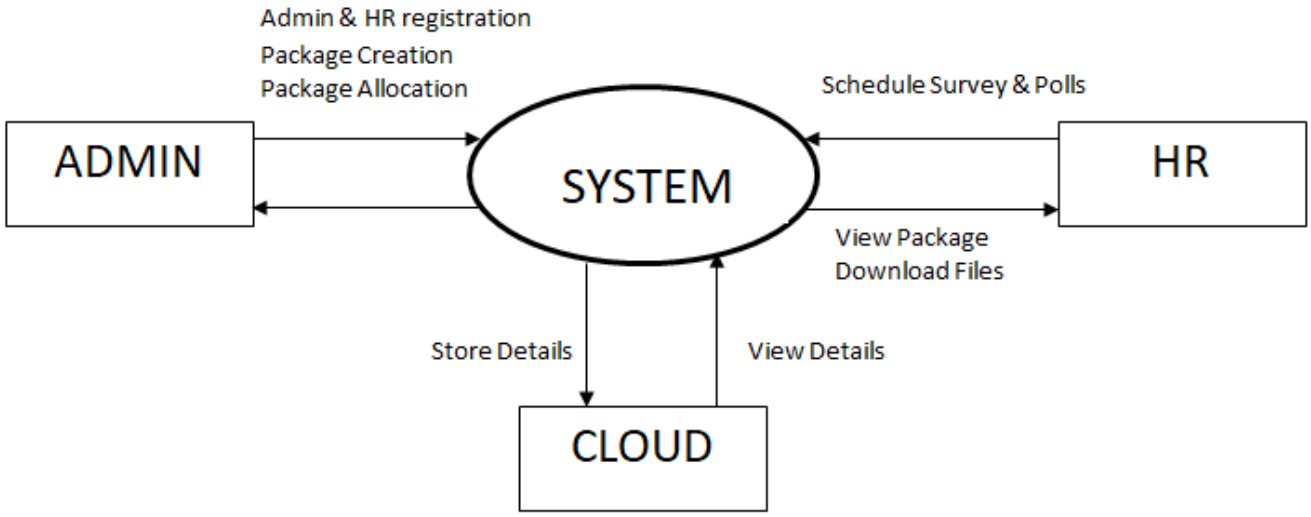
- A Data Flow has only one direction of flow between symbols. It may flow in both directions between a process and a data store to show a read before an update. The later is usually indicated however by two separate arrows since these happen at different type.
- A join in DFD means that exactly the same data comes from any of two or more different processes data store or sink to a common location.
- A data flow cannot go directly back to the same process it leads. There must be at least one other process that handles the data flow produce some other data flow returns the original data into the beginning process.
- A Data flow to a data store means update (delete or change).
- A Data flow from a data store means retrieve or use.

DFD:

Level – 0



Level – 1



Detailed Design:

Detailed design is also called as low-level design. During detailed design, the internal logic of various functionalities in the proposed system is decided. During this phase, further details of the data structures, the table designs and algorithmic design of each of the functionality is specified. It also involves the entity-relationship diagrams and the use-case diagrams that further explain the data flow in the system.

Detailed Design is carried out in different phases. The phases are as follows.

- Entity –Relationship (E-R)
- Database Design

ER Diagrams:

In software engineering, an entity-relationship model (ERM) is an abstract and conceptual representation of data. Entity-relationship modeling is a database modeling method, used to produce a type of conceptual schema or semantic data model of a system, often a relational database, and its requirements in a top-down fashion. Diagrams created by this process are called entity-relationship diagrams, ER diagrams, or ERDs.

The definitive reference for entity-relationship modeling is Peter Chen's 1976 paper. However, variants of the idea existed previously, and have been devised subsequently.

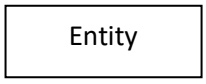
Overview:

The first stage of information system design uses these models during the requirements analysis to describe information needs or the type of information that is to be stored in a database. The data modeling technique can be used to describe any ontology (i.e. an overview and classifications of used terms and their relationships) for a certain area of interest. In the case of the design of an information system that is based on a database, the conceptual data model is, at a later stage (usually called logical design), mapped to a logical data model, such as the relational model; this in turn is mapped to a physical model during physical design.

The symbols used to design for ER-diagram:

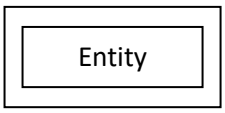
An Entity relationship Diagram is a data modeling technique that creates a graphical representation of the entities and the relationships between entities.

Entity:



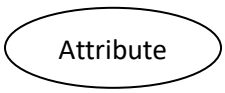
An entity is an object or concept about which you want to store information.

Weak Entity:



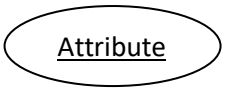
A weak entity is dependent on another entity to exist.

Attribute:



Attribute are the properties or characteristics of an entity.

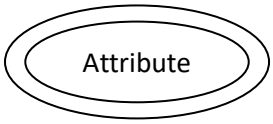
Key attribute:



A key attribute is the unique, distinguishing characteristic of the entity.

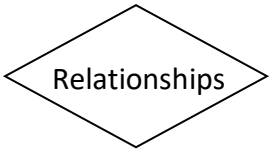
For example, an employee's social security number might be the employee's key attribute.

Multi-valued Attribute:



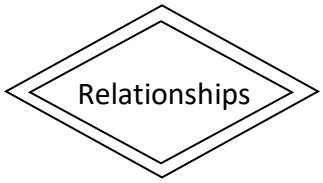
A multi-valued attribute can have more than one value. For example, an employee entity can have multiple skill values.

Relationships:



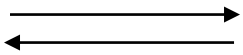
Relationships illustrate how two entities share information in the database structure.

Weak relationship:



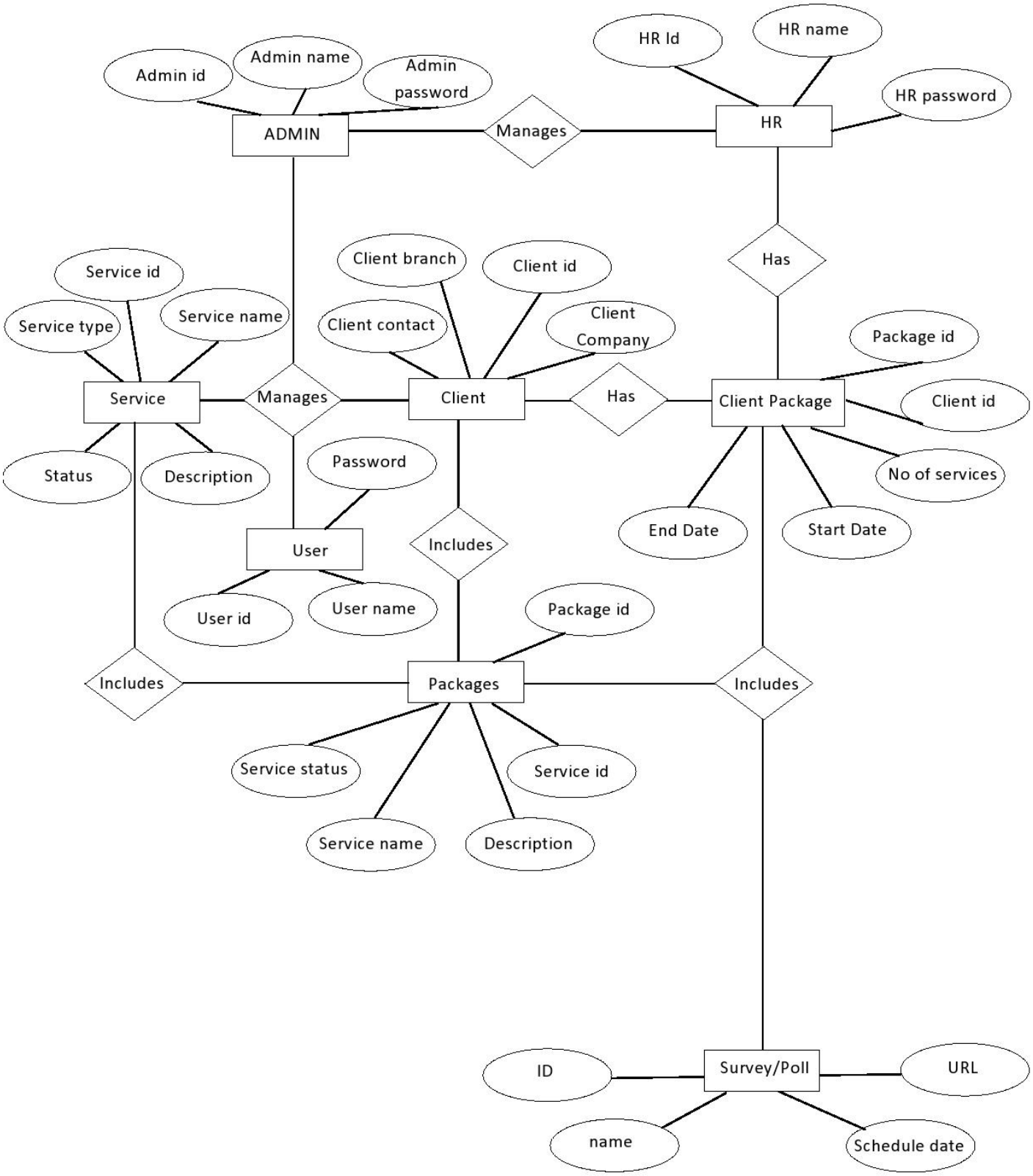
To connect a weak entity with others, you should use a weak relationship notation.

Flow Control:



To connect the different controls in the E-R Diagram

ER-DIAGRAM:


HR Id
HR name
Admin id
Admin name
Admin password
HR password
ADMIN
Manages
HR
Has
Service id
Client branch
Client id
Service type
Service name
Client contact
Client Company
Package id
Service
Manages
Client
Has
Client Package
Client id
Status
Description
Password
No of services
End Date
Start Date
User
Includes
User id
User name
Package id
Includes
Packages
Includes
Service status
Service id
Service name
Description
ID
Survey/Poll
URL
name
Schedule date

USE CASE ANALYSIS:

A use case analysis is a common technique used to identify the requirements of a system and the information used to both define processes used and classes which will be used both in the use case diagram and the overall use case in the development of a software system or program.

A use case analysis is the primary form for gathering usage requirements for a new software program or task to be completed. The primary goals of a use case analysis are designing a system from the user's perspective, communicating system behavior in the user's terms, and specifying all externally visible behaviors. Another set of goals for a use case analysis is to clearly communicate system requirements, how the system is to be used, the roles the user plays in the system, what the system does in response to the user stimulus, what the user receives from the system, and what value the customer or user will receive from the system.

Use case diagram:

Table Used In Database

Database Functions:

Database should have the facility to

- Insert Records,
- Update Records,
- Edit Records,
- Delete Records

Tables:

Table 5.1 admin_user:

Field Name	Data type	Constraints
Emp_ID	Int(10)	None
User_Role	Varchar (10)	Not Null
User_Fname	Varchar (10)	Not Null
User_Lname	Varchar (10)	Not Null
User_Mobile	Datetime	Not Null
User_Email	Varchar (50)	Not Null
User_OTP	Varchar (25)	Not Null
User_Gen_Pass	Varchar (100)	Not Null
User_New_Pass	Varchar (100)	Not Null
User_Status	Varchar (10)	Not Null

Table 5.2 corporate_details

Field Name	Data type	Constraints
company_id	int (10)	None
company_name	Varchar (255)	Not Null
Website	Varchar (255)	Not Null
Logo	Text	Not Null
company_display_name	Varchar (255)	Not Null

Gst	Varchar (255)	Not Null
Industry	Varchar (255)	Not Null
country	Varchar (255)	Not Null
State	Varchar (255)	Not Null
City	Varchar (255)	Not Null
address	Varchar (255)	Not Null
total_emp	Varchar (255)	Not Null
name_of_insurance_company	Varchar (255)	Not Null
tpa_name	Varchar (255)	Not Null
dependent_defination	Varchar (255)	Not Null
assisgn_plugh_manager	Varchar (255)	Not Null
client_manager_notes	Varchar (255)	Not Null
Status	Varchar (15)	Not Null

Table 5.2.1 corporate_contact_details:

Field Name	Data type	Constraints
company_id	Int(10)	None
contact_person_id	Int(10)	Not Null
contact_person	Varchar (255)	Not Null
designation	Varchar (255)	Not Null
email_id	Varchar (255)	Not Null
phone_number	Varchar (255)	Not Null
contact_person_status	Varchar (15)	Not Null

Table 5.2.2 corporate_branch_details:

Field Name	Data type	Constraints
company_id	Int(10)	None
branch_id	Int(10)	None
branch_country	Varchar (255)	Not Null
branch_state	Varchar (255)	Not Null

branch_city	Varchar (255)	Not Null
branch_contact_person	Varchar (255)	Not Null
branch_phone_number	Varchar (255)	Not Null
branch_email_id	Varchar (255)	Not Null
annual_health_check	Varchar (255)	Not Null
inhouse_facilty	Varchar (255)	Not Null
occupational_hazard	Varchar (255)	Not Null
shift	Varchar (255)	Not Null
branch_status	Varchar (10)	Not Null

Table 5.3 client_admin:

Field Name	Data type	Constraints
id	Int(10)	None
company_id	Int(10)	None
company_name	Varchar (100)	Not Null
admin_display_name	Varchar (100)	Not Null
first_name	Varchar (100)	Not Null
last_name	Varchar (100)	Not Null
contact_number	Varchar (15)	Not Null
login_id	Varchar (100)	Not Null
client_status	Varchar (10)	Not Null
new_password	Varchar (255)	Not Null
client_gen_pass	Varchar (255)	Not Null
client_otp	Varchar (255)	Not Null

Table 5.4 clientpackage_details:

Field Name	Data type	Constraints
cp_id	Int(10)	None
cp_company_name	Varchar (255)	Not Null
cp_sdate	date	Not Null

cp_edate	date	Not Null
cp_servicetype	Varchar (50)	Not Null
cp_noofservices	Varchar (255)	Not Null
cp_status	Varchar (11)	Not Null
cp_hr_status	Varchar (25)	Not Null

Table 5.5 category:

Field Name	Data type	Constraints
id	Int(10)	None
target_audience	Varchar (255)	Not Null
status	Varchar (255)	Not Null

Table 5.6 target_audience:

Field Name	Data type	Constraints
id	Int(10)	None
category_name	Varchar (255)	Not Null
status	Varchar (255)	Not Null

Table 5.7 occupational_hazard:

Field Name	Data type	Constraints
id	Int(11)	None
hazard	Varchar (255)	Not Null
description	Mediumtext	Not Null
status	Varchar (255)	Not Null

Table 5.8.1 language:

Field Name	Data type	Constraints
id	Int(10)	None

language	Varchar (255)	Not Null
status	Varchar (255)	Not Null

Table 5.8.2 expertise_area:

Field Name	Data type	Constraints
id	Int(10)	None
expertise	Varchar (255)	Not Null
status	Varchar (255)	Not Null

Table 5.8.3 industry:

Field Name	Data type	Constraints
id	Int(10)	None
industry	Varchar (255)	Not Null
status	Varchar (255)	Not Null

Table 5.8.4 branch_facilities:

Field Name	Data type	Constraints
id	Int(10)	None
branch_facility	Varchar (255)	Not Null
status	Varchar (255)	Not Null

Table 5.8.5 merchandise_category:

Field Name	Data type	Constraints
id	Int(10)	None
merchandise_category	Varchar (255)	Not Null
status	Varchar (255)	Not Null

Table 5.9 survey:

Field Name	Data type	Constraints
survey_id	Varchar (255)	None
survey_name	text	Not Null
survey_status	Varchar (10)	Not Null
survey_type	Varchar (25)	Not Null
has_file	Varchar (5)	Not Null

Table 5.9.1 survey_answers:

Field Name	Data type	Constraints
survey_answer_id	Int(10)	None
survey_id	Varchar (500)	None
survey_response_id	Int(10)	None
answer	text	Not Null
question_id	Int(11)	None

Table 5.9.2 survey_responses:

Field Name	Data type	Constraints
survey_response_id	Int(10)	None
survey_id	Varchar (500)	None
datetime	Datetime	None

Table 5.9.3 survey_file:

Field Name	Data type	Constraints
survey_fileID	Int(11)	None
survey_file	Varchar (500)	Not Null
survey_id	Varchar (255)	Not Null

Table 5.9.4 survey_particpants:

Field Name	Data type	Constraints
survey_participant_id	Int(11)	None
survey_participant_name	Varchar (255)	None
survey_participant_email	Varchar (255)	None
survey_participant_phno	text	None
survey_id	Varchar (255)	None

Table 5.9.5 question:

Field Name	Data type	Constraints
question_id	Int(11)	None
survey_id	Varchar (255)	None
question_type	Varchar (255)	Not Null
question_text	text	Not Null
is_required	Varchar (255)	Not Null
question_order	Int(11)	Not Null

Table 5.9.6 radio_options:

Field Name	Data type	Constraints
radio_id	Int(11)	None
question_id	Int(11)	Not Null
survey_id	Varchar (255)	Not Null
radio_txt	Varchar (255)	Not Null

Table 5.9.7 dropdown_options:

Field Name	Data type	Constraints
option_id	Int(11)	None
question_id	Int(11)	Not Null

survey_id	Varchar (255)	Not Null
option_txt	Varchar (255)	Not Null

Table 5.9.8 date_options:

Field Name	Data type	Constraints
date_id	Int(11)	None
survey_id	Varchar (255)	Not Null
question_id	Int(11)	Not Null
date_restriction	Varchar (15)	Not Null

Table 5.9.9 checkbox_options:

Field Name	Data type	Constraints
chkbx_id	Int(11)	None
question_id	Int(11)	Not Null
survey_id	Varchar (255)	Not Null
chkbx_txt	Varchar (15)	Not Null

Table 5.10 polls:

Field Name	Data type	Constraints
poll_id	Varchar (255)	None
poll_name	Varchar (500)	Not Null
poll_txt	Varchar (1000)	Not Null
poll_status	Varchar (15)	Not Null

Table 5.10.1 polls_response:

Field Name	Data type	Constraints
poll_response_id	Int(11)	None
poll_id	Varchar (50)	None
poll_response	Varchar (255)	None

Table 5.11 partner_ind_details:

Field Name	Data type	Constraints
prt_ind_id	Int (10)	None
name	Varchar (255)	Not Null
dob	date	Not Null
gender	Varchar (10)	Not Null
educational_qualification	Varchar (255)	Not Null
total_yoe	Int (3)	Not Null
pan_no	Varchar (20)	Not Null
email_id	Varchar (255)	Not Null
website	Varchar (255)	Not Null
gst	Varchar (255)	Not Null
best_time	Varchar (255)	Not Null
pre_contact	Varchar (255)	Not Null
primary_contact	Varchar (255)	Not Null
second_contact	Varchar (255)	Not Null
notes	Varchar (255)	Not Null
address	Varchar (255)	Not Null
long_lat	Varchar (255)	Not Null
country	Varchar (255)	Not Null
state	Varchar (255)	Not Null
city	Varchar (255)	Not Null
status	Varchar (20)	Not Null

Table 5.11.1 prt_ind_social_details:

Field Name	Data type	Constraints
social_id	Int(10)	None
platform	Varchar (255)	Not Null
url	Varchar (255)	Not Null
prt_ind_id	Int(10)	Not Null
social_details_status	Varchar (10)	Not Null

Table 5.11.2 prt_ind_service_details:

Field Name	Data type	Constraints
service_id	Int(10)	None
service_city	Varchar (255)	Not Null
prt_ind_id	Int(10)	Not Null
service_status	Varchar (10)	Not Null

Table 5.11.3 prt_ind_language_details :

Field Name	Data type	Constraints
language_id	Int(10)	None
languages	Varchar (255)	Not Null
rws	Varchar (255)	Not Null
prt_ind_id	Int(10)	Not Null

Table 5.11.4 prt_ind_audience_comfortable :

Field Name	Data type	Constraints
audience_id	Int(10)	None
audience_comfortable	Varchar (255)	Not Null
prt_ind_id	Int(10)	Not Null

Table 5.11.5 prt_ind_area_of_expertise:

Field Name	Data type	Constraints
area_id	Int(10)	None
areaofexpertise	Varchar (255)	Not Null
Yearsofexperience	Varchar (255)	Not Null
prt_ind_id	Int(10)	Not Null
area_status	Varchar (10)	Not Null

Table 5.12 partner_org_details:

Field Name	Data type	Constraints
prt_org_id	Int (10)	None
organization	Varchar (255)	Not Null
date_of_registration	date	Not Null
website	Varchar (10)	Not Null
email_id	Varchar (255)	Not Null
contact_person	Int (3)	Not Null
designation	Varchar (20)	Not Null
phone_number	Varchar (10)	Not Null
pan	Varchar (255)	Not Null
gst	Varchar (255)	Not Null
best_time	Varchar (255)	Not Null
pre_contact	Varchar (255)	Not Null
primary_contact	Varchar (255)	Not Null
second_contact	Varchar (255)	Not Null
notes	Varchar (255)	Not Null
country	Varchar (255)	Not Null
state	Varchar (255)	Not Null
city	Varchar (255)	Not Null
address	Varchar (255)	Not Null
status	Varchar (10)	Not Null

Table 5.12.1 prt_org_social_details:

Field Name	Data type	Constraints
social_id	Int(10)	None
Platform	Varchar (255)	Not Null
url	Varchar (255)	Not Null
prt_org_id	Int(10)	Not Null
social_status	Varchar (10)	Not Null

Table 5.12.2 prt_org_service_details:

Field Name	Data type	Constraints
service_id	Int(10)	None
service_area	Varchar (255)	Not Null
prt_org_id	Int(10)	Not Null
service_status	Varchar (10)	Not Null

Table 5.12.3 prt_ind_area_of_expertise:

Field Name	Data type	Constraints
area_id	Int(10)	None
areaofexpertise	Varchar (255)	Not Null
Yearsofexperience	Varchar (255)	Not Null
prt_ind_id	Int(10)	Not Null
area_status	Varchar (10)	Not Null

Table 5.12.4 prt_org_area_of_expertise:

Field Name	Data type	Constraints
aoe_id	Int(10)	None
Areaofexpertise	Varchar (255)	Not Null
yearsofexperience	Varchar (255)	Not Null
prt_org_id	Int(10)	Not Null
aoe_status	Varchar (10)	Not Null

Table 5.12.1 prt_org_contact_details:

Field Name	Data type	Constraints
contact_id	Int(10)	None
contact_contact_person	Varchar (255)	Not Null
contact_designation	Varchar (255)	Not Null

contact_phone_number	Int(10)	Not Null
contact_email_id	Varchar (10)	Not Null
prt_org_id	Int(10)	Not Null
contact_status	Varchar (10)	Not Null

Table 5.13 healthpro_details:

Field Name	Data type	Constraints
hp_id	Int(10)	None
hp_category	Varchar (255)	Not Null
hp_targetaudience	Varchar (255)	Not Null
hp_occupationalhazard	Varchar (255)	Not Null
hp_name	Varchar (10)	Not Null
hp_duration	Varchar (255)	Not Null
hp_description	text	Not Null
hp_status	Varchar (10)	Not Null

Table 5.13.1 hpexternal_details:

Field Name	Data type	Constraints
hpe_id	Int(10)	None
hpe_external	Varchar (255)	Not Null
hp_id	Int(10)	Not Null

Table 5.13.2 hp_file_details:

Field Name	Data type	Constraints
hpf_id	Int(5)	None
hp_file	text	Not Null
hp_id	Int(10)	Not Null

Table 5.14 newsletter:

Field Name	Data type	Constraints
ns_service_code	Int(10)	None
ns_category	Varchar (255)	Not Null
ns_target_audience	Varchar (255)	Not Null
ns_name	Varchar (255)	Not Null
ns_upload_file	Varchar (255)	Not Null
ns_description	Mediumtext	Not Null
ns_hazard	Varchar (255)	Not Null
ns_status	Varchar (10)	Not Null

Table 5.15 toolkit:

Field Name	Data type	Constraints
tk_id	Int(10)	None
tk_category	Varchar (255)	Not Null
tk_target_audience	Varchar (255)	Not Null
tk_hazard	Varchar (255)	Not Null
tk_name	Varchar (255)	Not Null
tk_description	Text	Not Null
tk_important_days	Varchar (255)	Not Null
tk_status	Varchar (10)	Not Null

Table 5.15.1 toolkitactivity:

Field Name	Data type	Constraints
tk_activity_id	Int(10)	None
toolkit_activity_number	Varchar (255)	Not Null
toolkit_description	Text	Not Null
toolkit_sop	Varchar (255)	Not Null
toolkit_announcement	Varchar (255)	Not Null

toolkit_annexure	Varchar (255)	Not Null
toolkit_collaterals	Varchar (255)	Not Null
toolkit_attach_survey	Varchar (255)	Not Null
toolkit_status	Varchar (255)	Not Null
tk_id	Int(10)	Not Null

Table 5.16 bitesizeideas:

Field Name	Data type	Constraints
id	Int(10)	None
choose_category	Varchar (255)	Not Null
target_audience	Varchar (255)	Not Null
occupational_hazard	Varchar (255)	Not Null
name	Varchar (255)	Not Null
ideas	Varchar (255)	Not Null
status	Varchar (255)	Not Null
hd_id	Int(11)	Not Null
hd_date	Date	Not Null
hd_day	Varchar (255)	Not Null

Table 5.17 merchandise_service:

Field Name	Data type	Constraints
merchandise_id	Int(10)	None
merchandise_name	Varchar (255)	Not Null
merchandise_audience	Varchar (255)	Not Null
merchandise_category_general	Varchar (255)	Not Null
merchandise_category	Varchar (255)	Not Null
merchandise_remarks	Varchar (255)	Not Null
merchandise_cdr_ai	Varchar (255)	Not Null
merchandise_jpg	Varchar (255)	Not Null
merchandise_png	Varchar (255)	Not Null

merchandise_pdf	Varchar (255)	Not Null
merchandise_status	Varchar (10)	Not Null

Table 5.18 webinar:

Field Name	Data type	Constraints
w_id	Int(10)	None
w_category	Varchar (255)	Not Null
w_target_audience	Varchar (255)	Not Null
w_hazard	Varchar (255)	Not Null
w_name	Varchar (255)	Not Null
w_description	Text	Not Null
w_date	date	Not Null
w_time	Varchar (255)	Not Null
w_status	Varchar (255)	Not Null

Table 5.19 healthtip:

Field Name	Data type	Constraints
id	Int(10)	None
tip_for_the_day	Varchar (255)	Not Null
date	date	Not Null
display_date	date	Not Null
flag	tinyint	Not Null
status	Varchar (255)	Not Null

Table 5.20 healthdays:

Field Name	Data type	Constraints
hd_id	Int(10)	None
hd_date	date	Not Null
hd_day	Varchar (255)	Not Null

hd_writeup	Varchar (255)	Not Null
hd_upload_file	Varchar (255)	Not Null
hd_status	Varchar (10)	Not Null

Table 5.21 ticket:

Field Name	Data type	Constraints
Ticket_ID	Int(10)	None
t_company_name	Varchar (50)	Not Null
ticket_type	Varchar (50)	Not Null
t_subject	Varchar (255)	Not Null
t_description	Varchar (255)	Not Null
t_remarks	Varchar (255)	Not Null
t_date	date	Not Null
t_status	Varchar (50)	Not Null
t_raisedby	Varchar (255)	Not Null

6. TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

6.1 Test Plan

Unit testing:

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

Integration testing:

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

Functional testing:

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

- Valid Input: identified classes of valid input must be accepted.
- Invalid Input: identified classes of invalid input must be rejected.

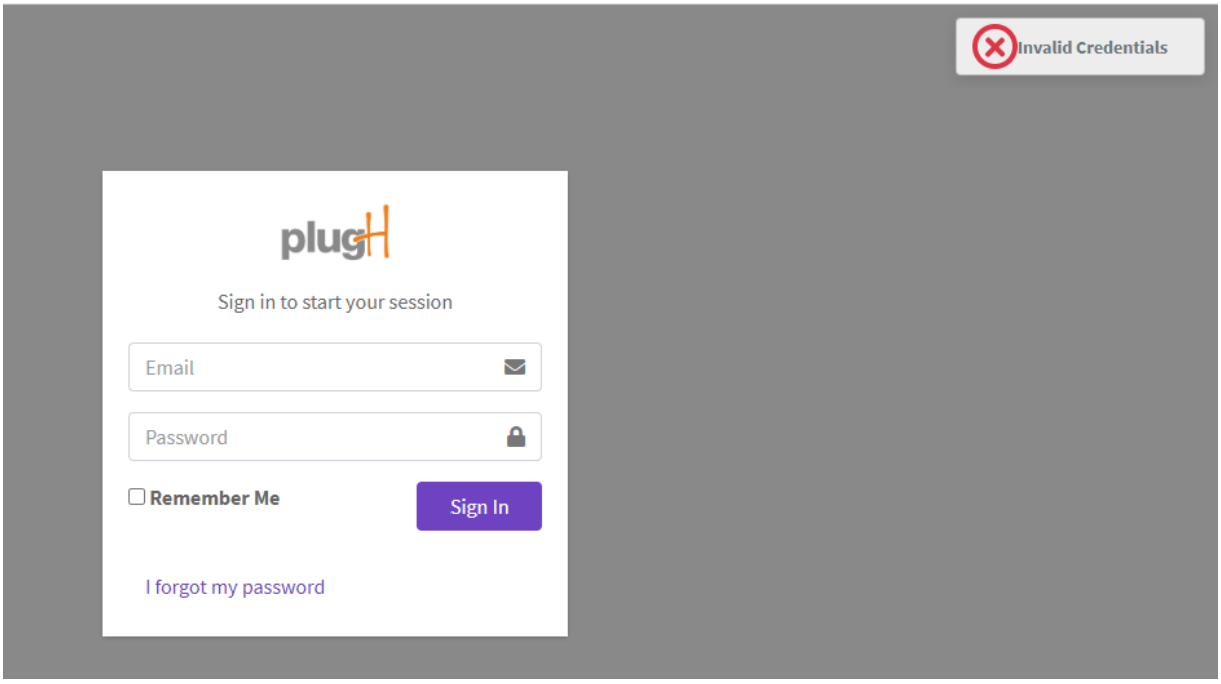


Fig 6.1.1 Invalid login credentials

- Functions: identified functions must be exercised.
- Output: identified classes of application outputs must be exercised.
- Systems/Procedures: interfacing systems or procedures must be invoked.
- Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

System Testing:

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

White Box Testing:

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

Black Box Testing:

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box. You cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works.

Test objectives:

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested:

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

Acceptance Testing:

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Bottom-up Integration:

This method begins the construction and testing with the modules at the lowest level in the program structure. Since the modules are integrated from the bottom up, processing required for modules subordinate to a given level is always available and the need for stubs is eliminated. The bottom up integration strategy may be implemented with the following steps:

- The low-level modules are combined into clusters into clusters that perform a specific Software sub-function.

- A driver (i.e.) the control program for testing is written to coordinate test case input and output.
- The cluster is tested.
- Drivers are removed and clusters are combined moving upward in the program structure.

OTHER TESTING METHODOLOGIES

User Acceptance Testing:

User Acceptance of a system is the key factor for the success of any system. The system under consideration is tested for user acceptance by constantly keeping in touch with the prospective system users at the time of developing and making changes wherever required. The system developed provides a friendly user interface that can easily be understood even by a person who is new to the system.

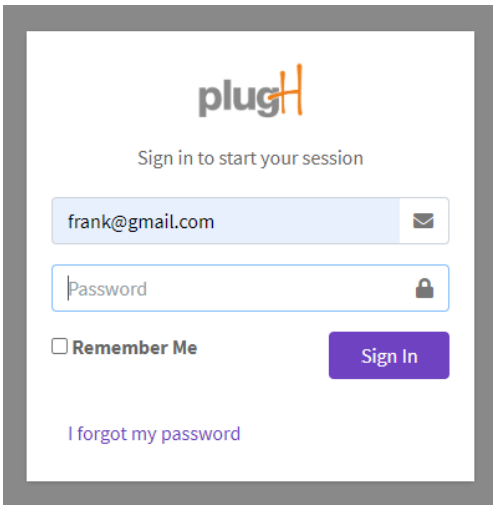


Fig 6.1.2 User Acceptance

Output Testing:

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in the specified format. Asking the users about the format required by them tests the outputs generated or displayed by the system under consideration. Hence the output format is considered in 2 ways – one is on screen and another in printed format.

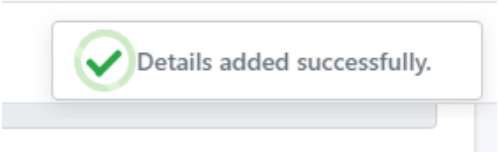


Fig 6.1.3 Output Testing

Validation Checking:

Validation checks are performed on the following fields.

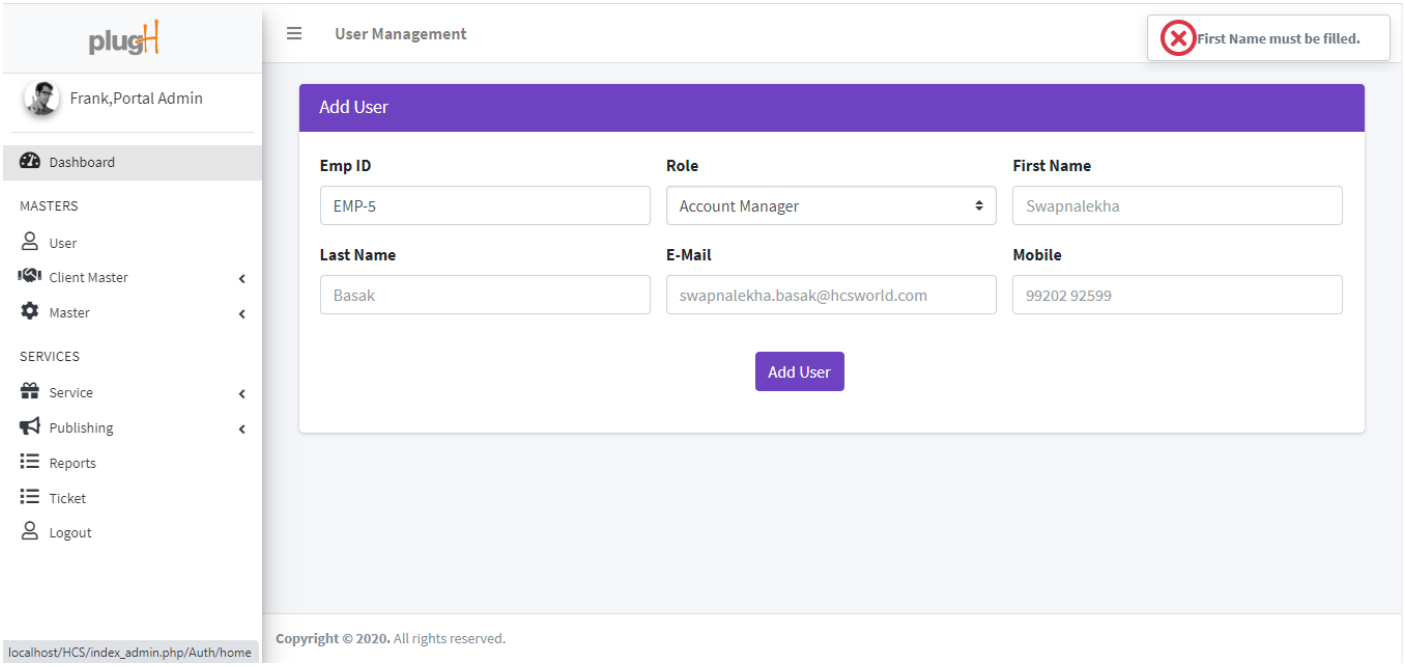


Fig 6.1.4 validation checking

Text Field:

The text field can contain only the number of characters lesser than or equal to its size. The text fields are alphanumeric in some tables and alphabetic in other tables. Incorrect entry always flashes and error message.

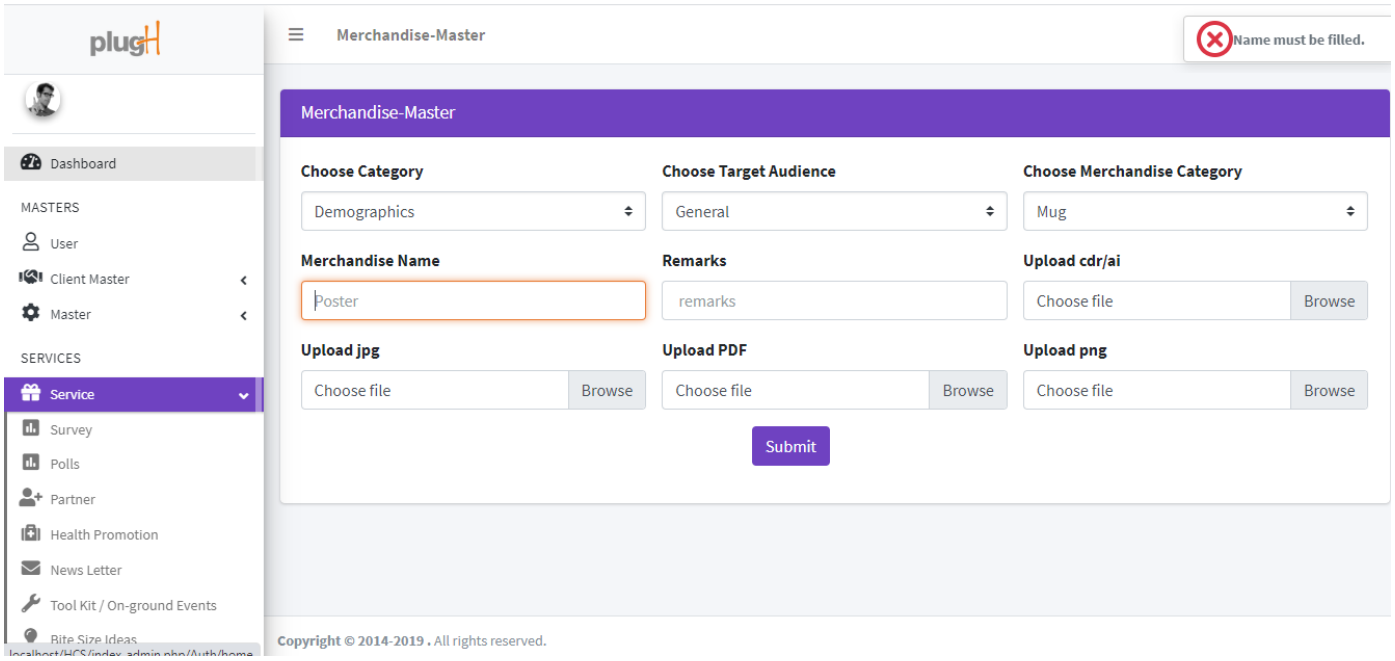


Fig 6.1.5 Text Field Validation

Numeric Field:

The numeric field can contain only numbers from 0 to 9. An entry of any character flashes an error message. The individual modules are checked for accuracy and what it has to perform. Each module is subjected to test run along with sample data. The individually tested modules are integrated into a single system. Testing involves executing the real data information is used in the program the existence of any program defect is inferred from the output. The testing should be planned so that all the requirements are individually tested.

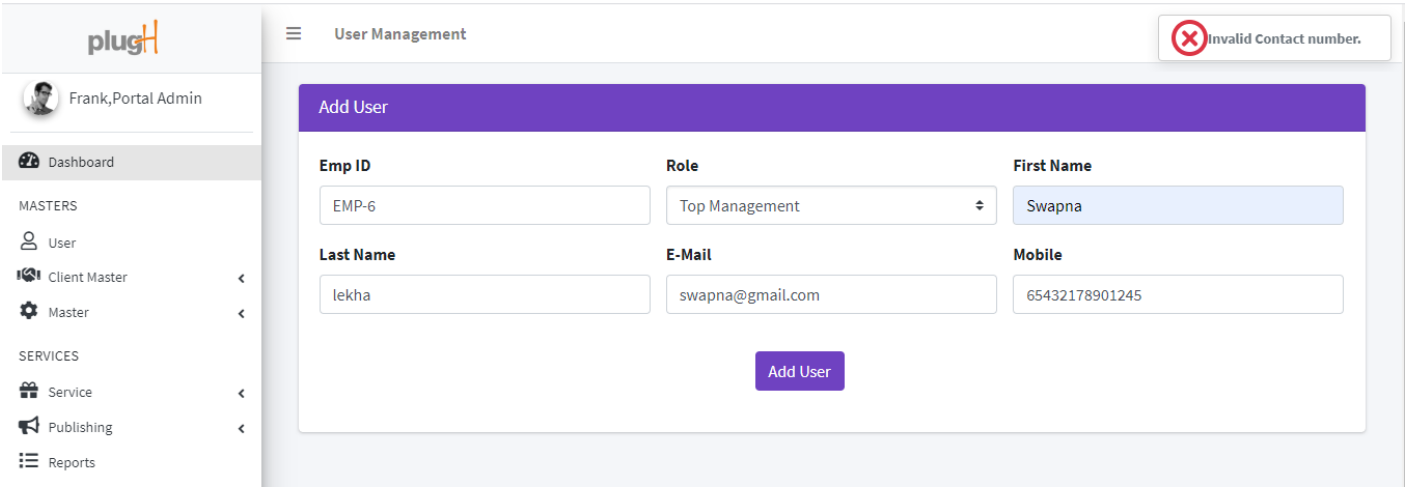


Fig 6.1.5 Numeric Field Validation

A successful test is one that gives out the defects for the in appropriate data and produces and output revealing the errors in the system.

USER TRAINING:

Whenever a new system is developed, user training is required to educate them about the working of the system so that it can be put to efficient use by those for whom the system has been primarily designed. For this purpose, the normal working of the project was demonstrated to the prospective users. Its working is easily understandable and since the expected users are people who have good knowledge of computers, the use of this system is very easy.

MAINTAINENCE:

This covers a wide range of activities including correcting code and design errors. To reduce the need for maintenance in the long run, we have more accurately defined the user's requirements during the

process of system development. Depending on the requirements, this system has been developed to satisfy the needs to the largest possible extent. With development in technology, it may be possible to add many more features based on the requirements in future. The coding and designing are simple and easy to understand which will make maintenance easier.

TESTING STRATEGY:

A strategy for system testing integrates system test cases and design techniques into a well-planned series of steps that results in the successful construction of software. The testing strategy must co-operate test planning, test case design, test execution, and the resultant data collection and evaluation .A strategy for software testing must accommodate low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high level tests that validate major system functions against user requirements.

Software testing is a critical element of software quality assurance and represents the ultimate review of specification design and coding. Testing represents an interesting anomaly for the software. Thus, a series of testing are performed for the proposed system before the system is ready for user acceptance testing.

6.2 Types of tests carried:

Slno	Test case Title	Description	Expected Outcome	Result
1	On select checkbox in grid view	The changes should update in the database which service is selected	Update Successfully	Passed
2	On unselect checkbox in grid view	Updating without select the checkbox	Update fails with an error 'Please select service'	Passed
3	Getting service details with valid service-ID	service should display with valid service -ID	Status should display with service details	Passed

4	On selecting the serviced type	In the grid view it should display the data of selected service type	It shows the data of the services of selected type in Grid view	Passed
5	On changing the serviced type	It should show the data of the services belonging to changed service type	It shows the data of the services of type changed in Grid view	Passed
6	No data in the database on not selecting service type	This will display the error message in the same page	It shows a message 'No data found'	Passed
7	On selecting type of date restriction as none	If admin selects none in the option for date tag	It displays the calendar with no restriction.	Passed
8	On selecting type of date restriction as future days	If admin selects future dates in the option for date tag	It displays the calendar with a restriction on past days	Passed
9	On selecting type of date restriction as past days	If admin selects past dates in the option for date tag	It displays the calendar with a restriction on past days	Passed

6.3 Screenshots:

6.3.1 ADMIN MODULE

6.3.1.1 USER LOGIN MODULE

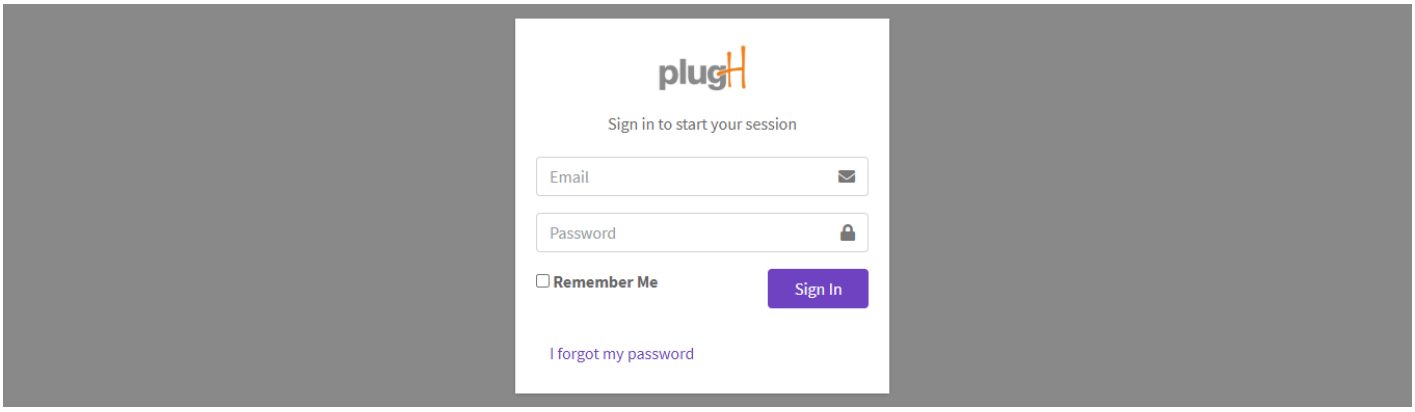


Fig 6.3.1.1.1 Admin Login

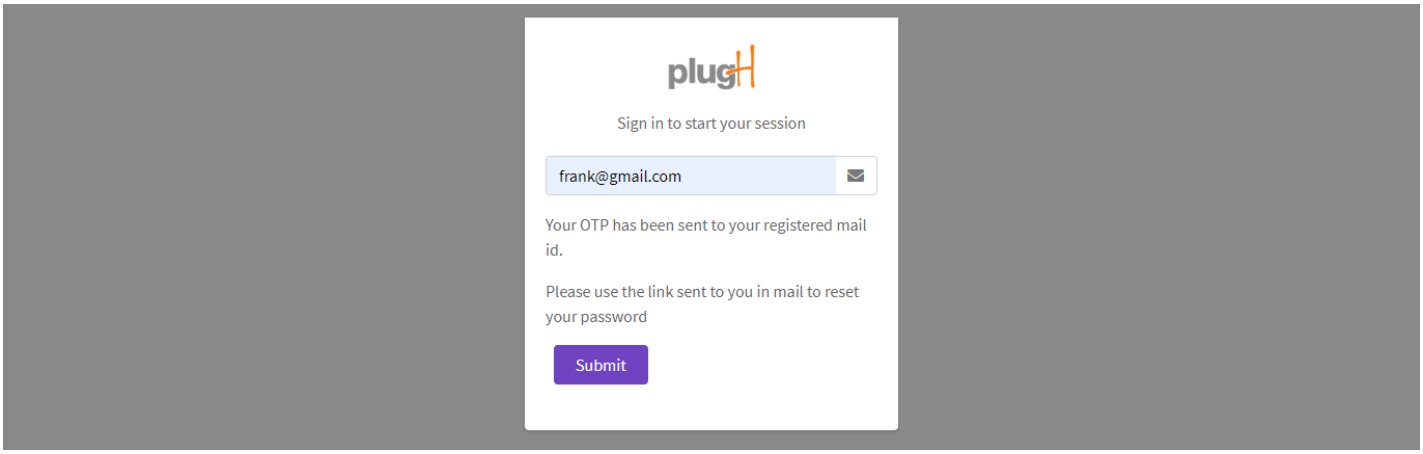


Fig 6.3.1.1.2 Forgot password

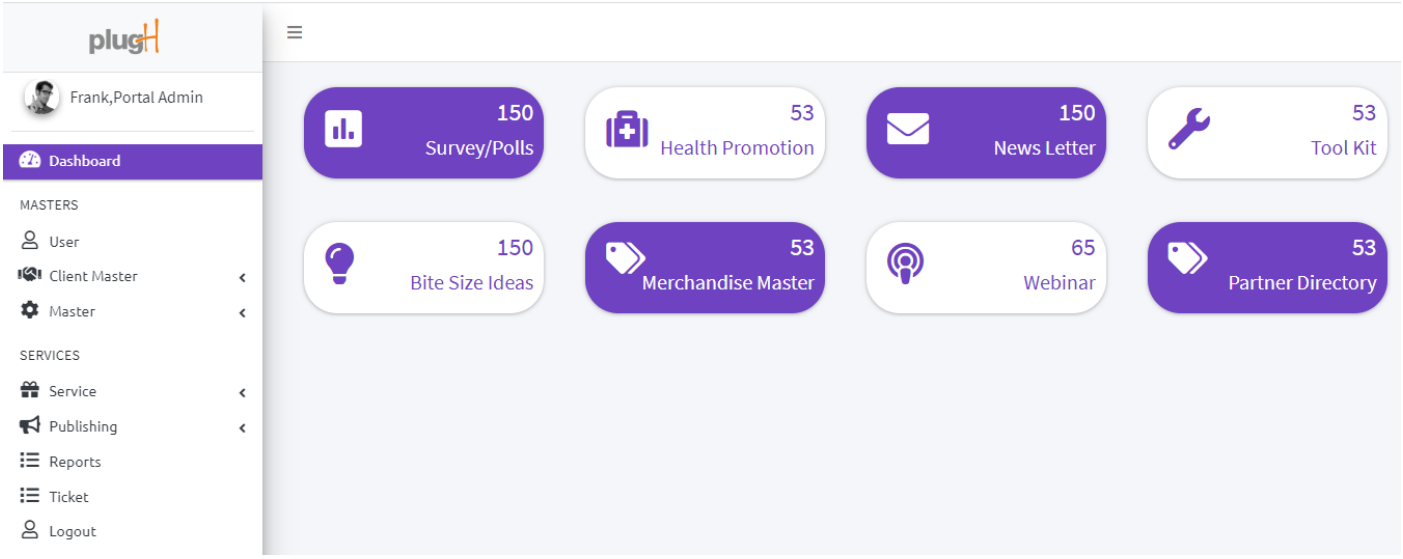


Fig 6.3.1.1.3 Admin Dashboard

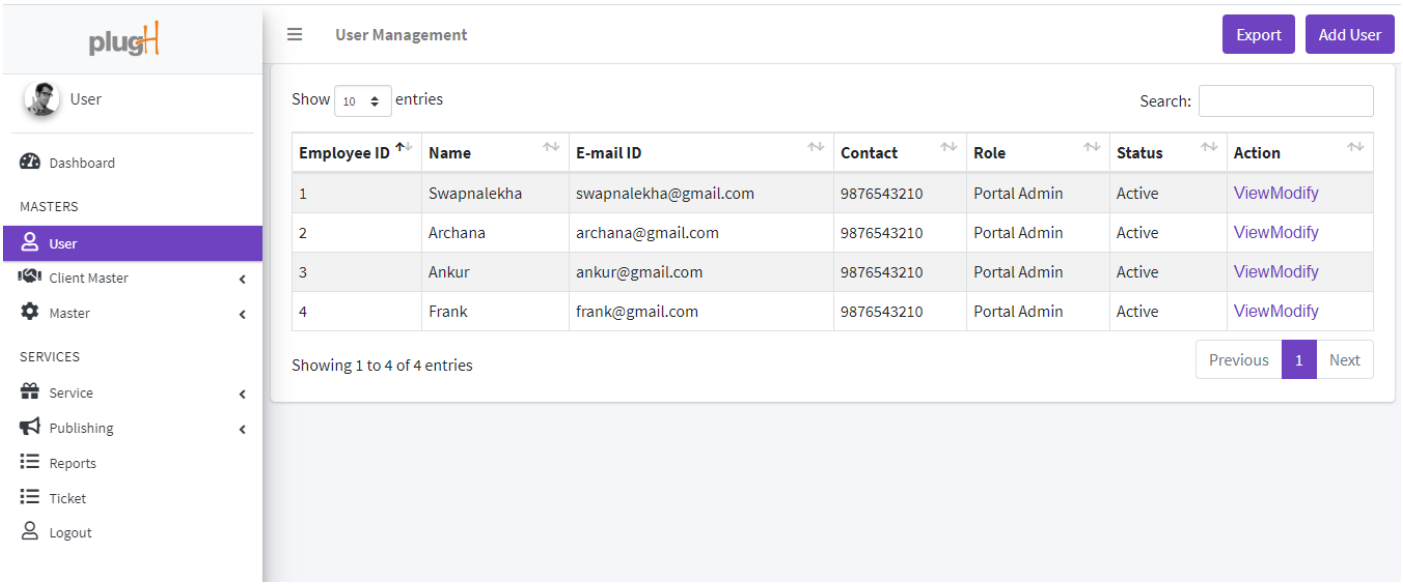


Fig 6.3.1.1.4 Admin users list

6.3.1.2 CLIENT MASTER

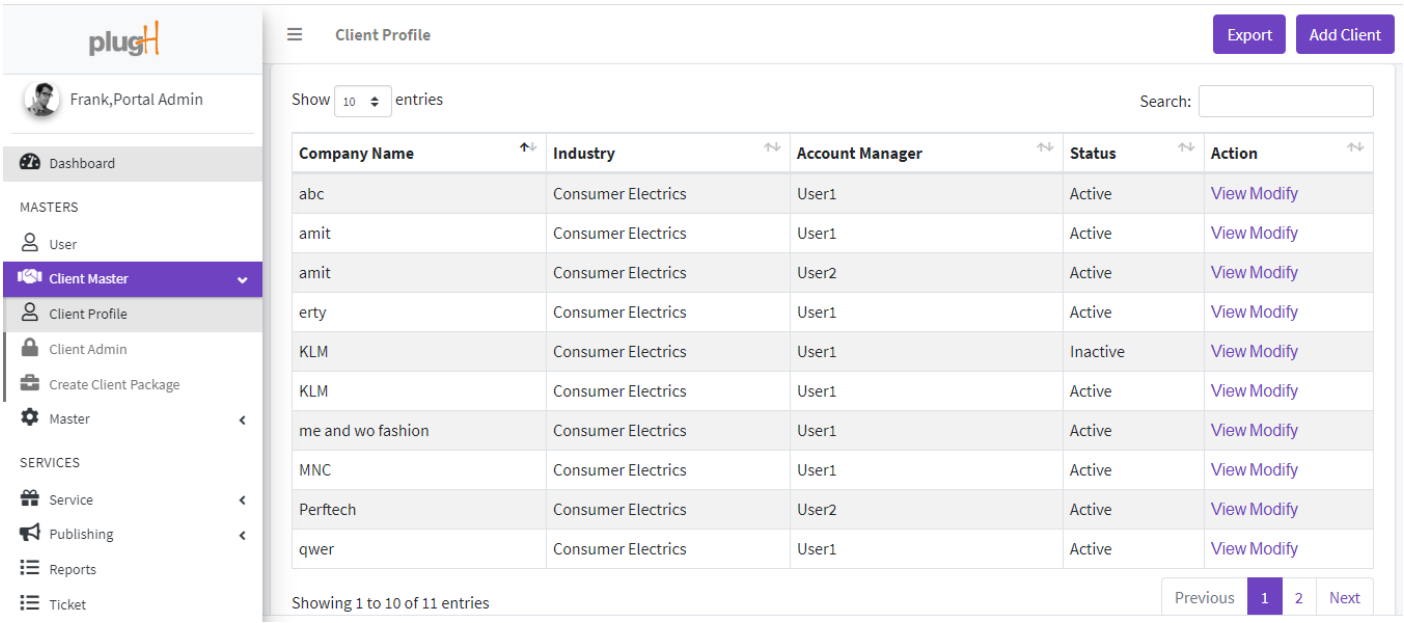


Fig 6.3.1.2.1 Client profiles

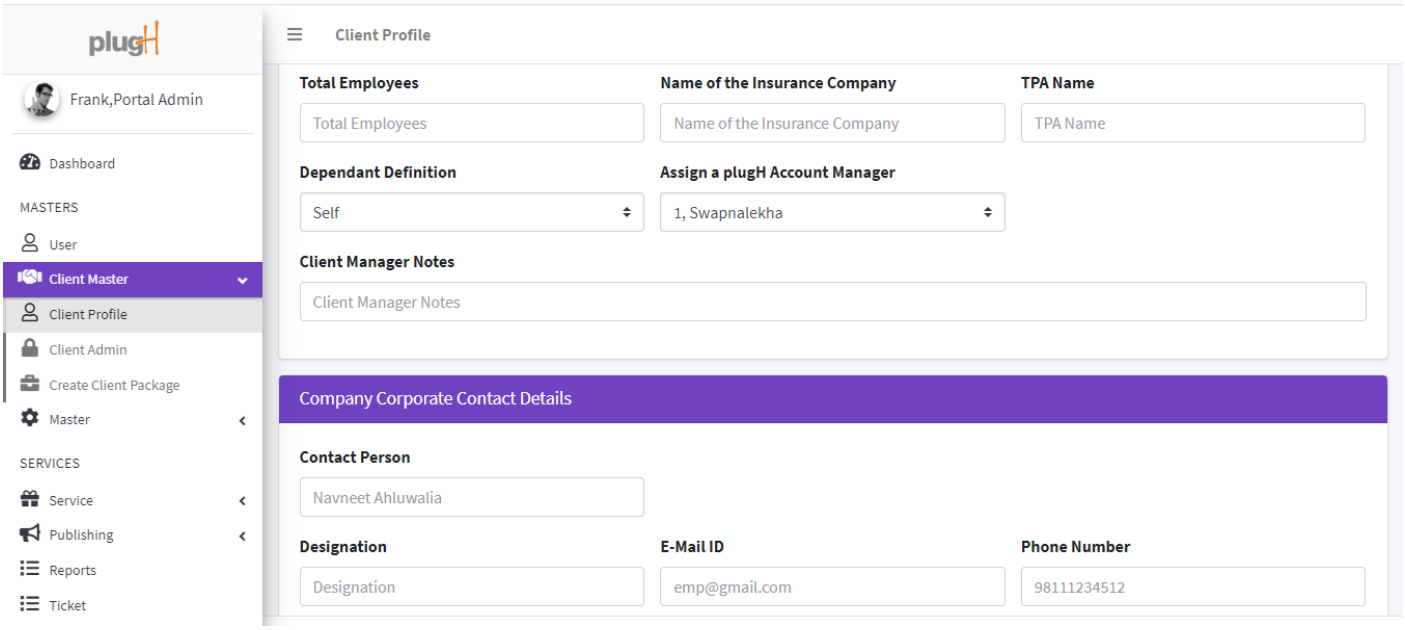


Fig 6.3.1.2.2 Add client profile

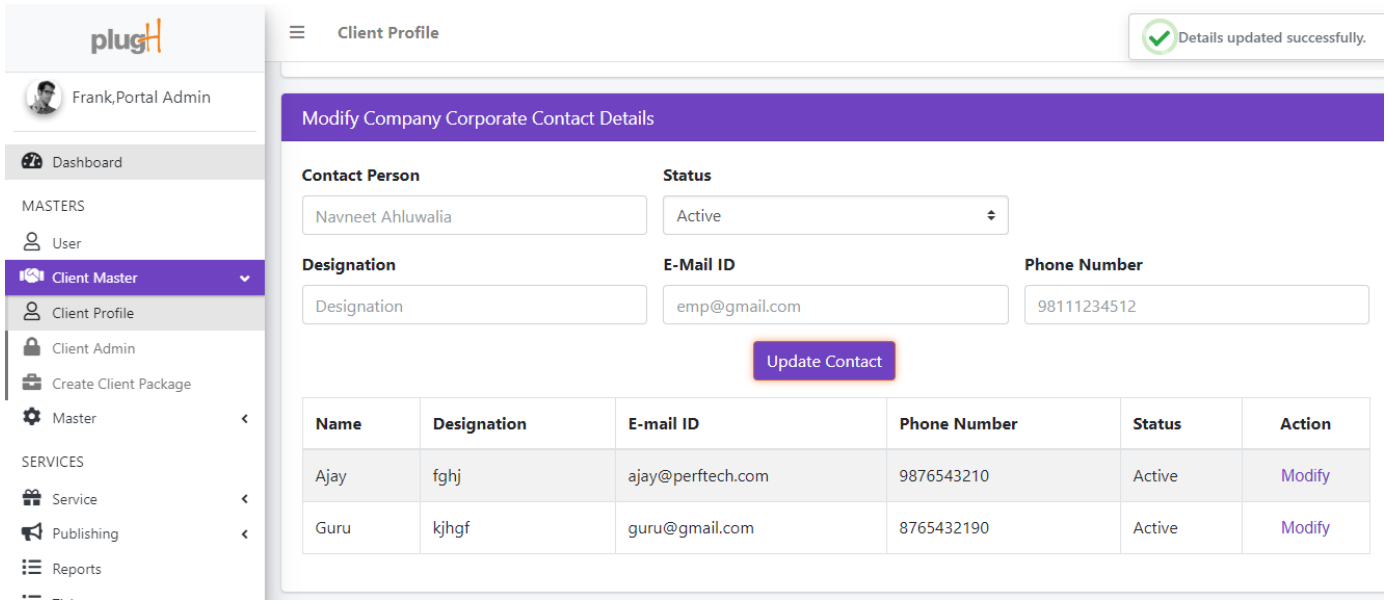


Fig 6.3.1.2.3 Update client profile

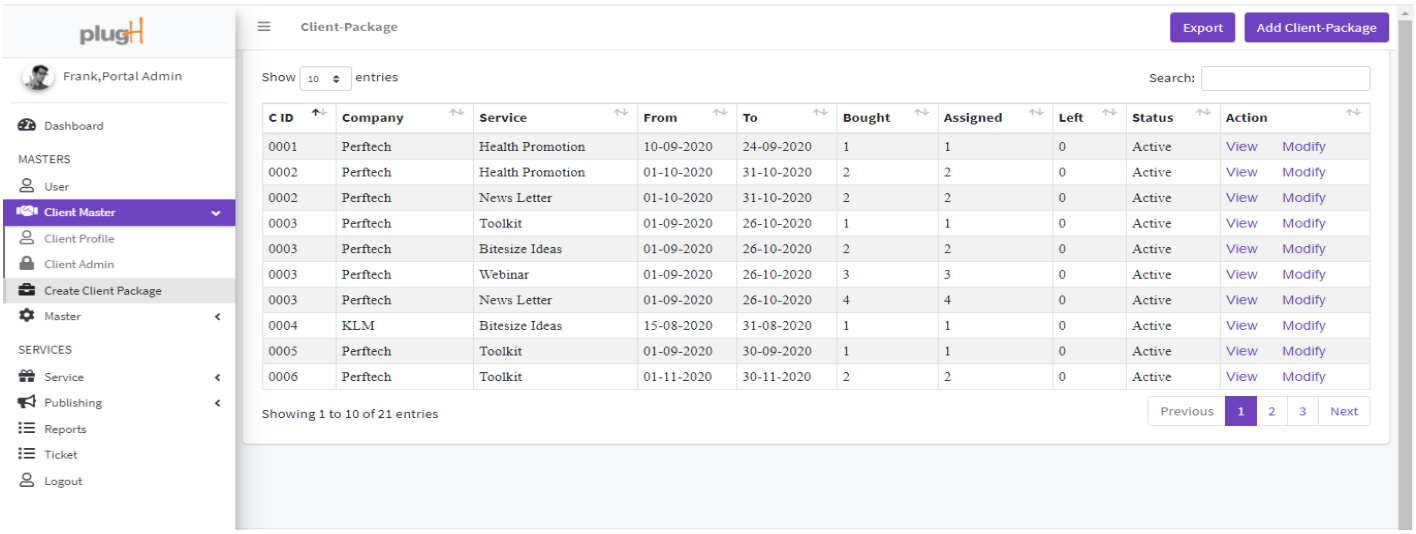


Fig 6.3.1.2.4 Client package list

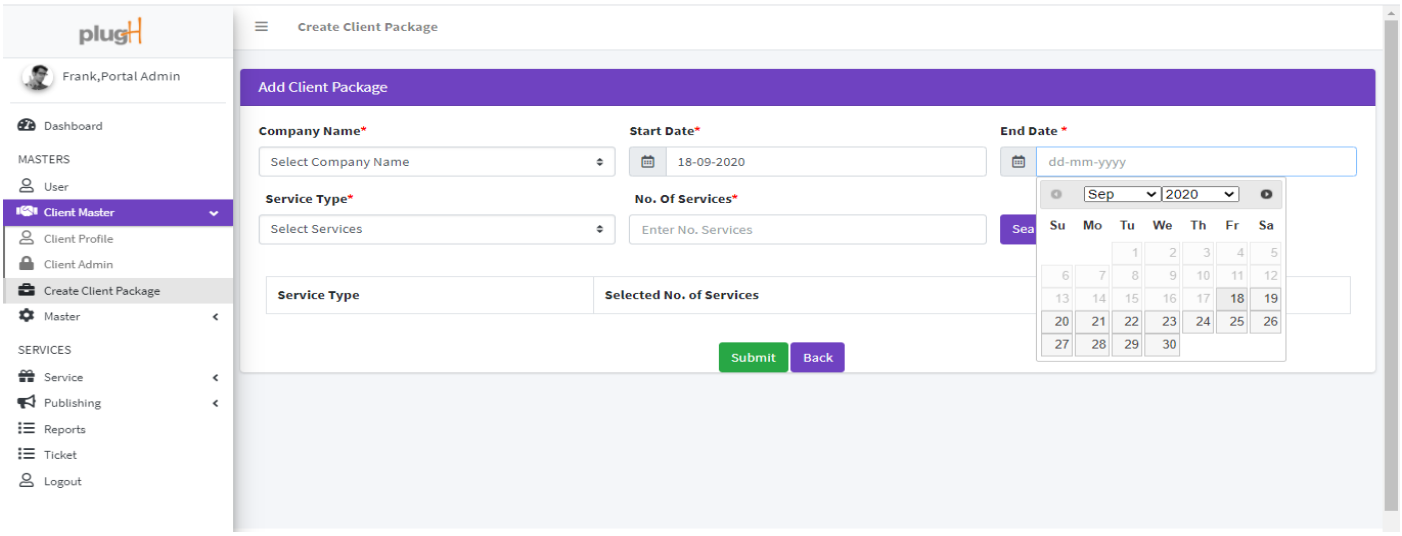


Fig 6.3.1.2.5 Add Client package

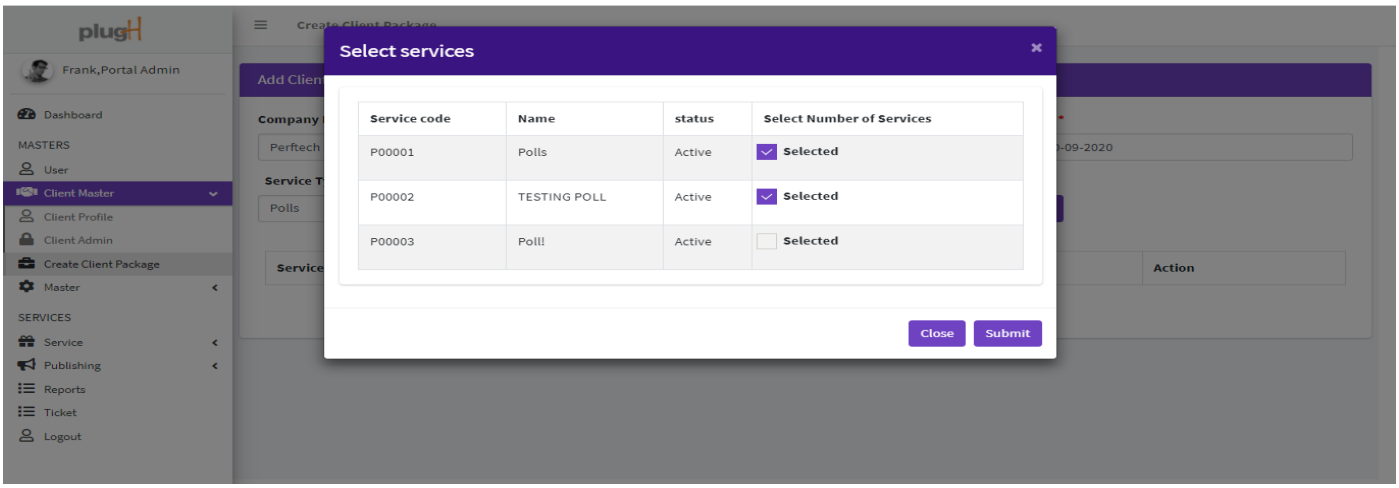


Fig 6.3.1.2.6 Add Client package

6.3.1.3 MASTER

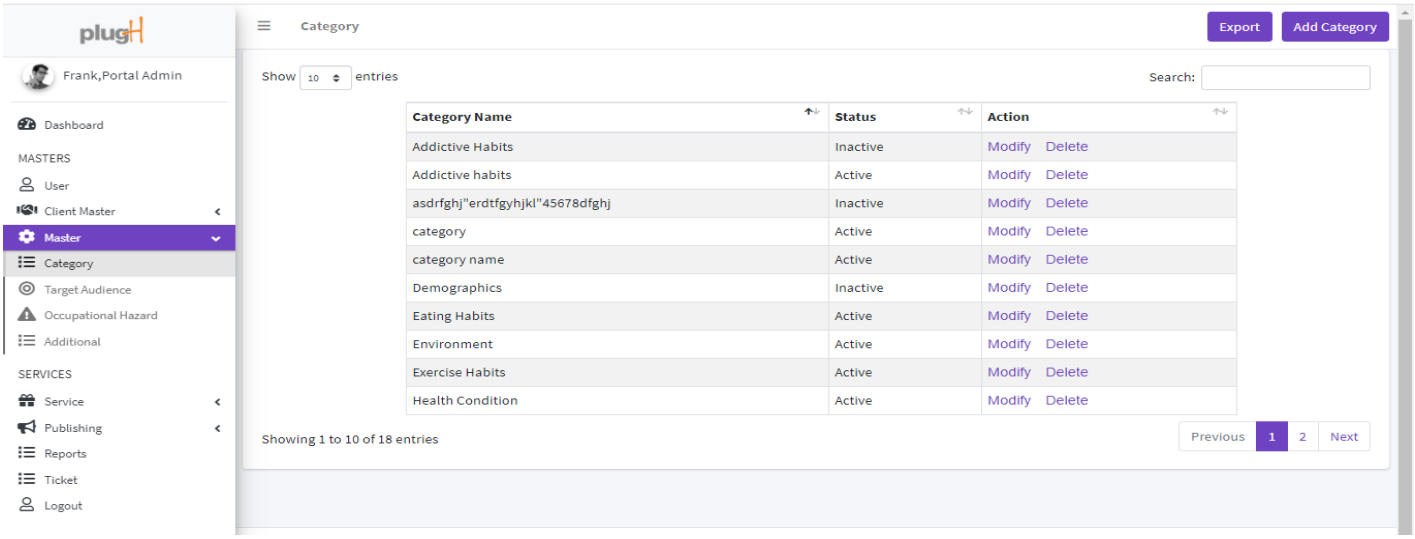


Fig 6.3.1.3.1 Category

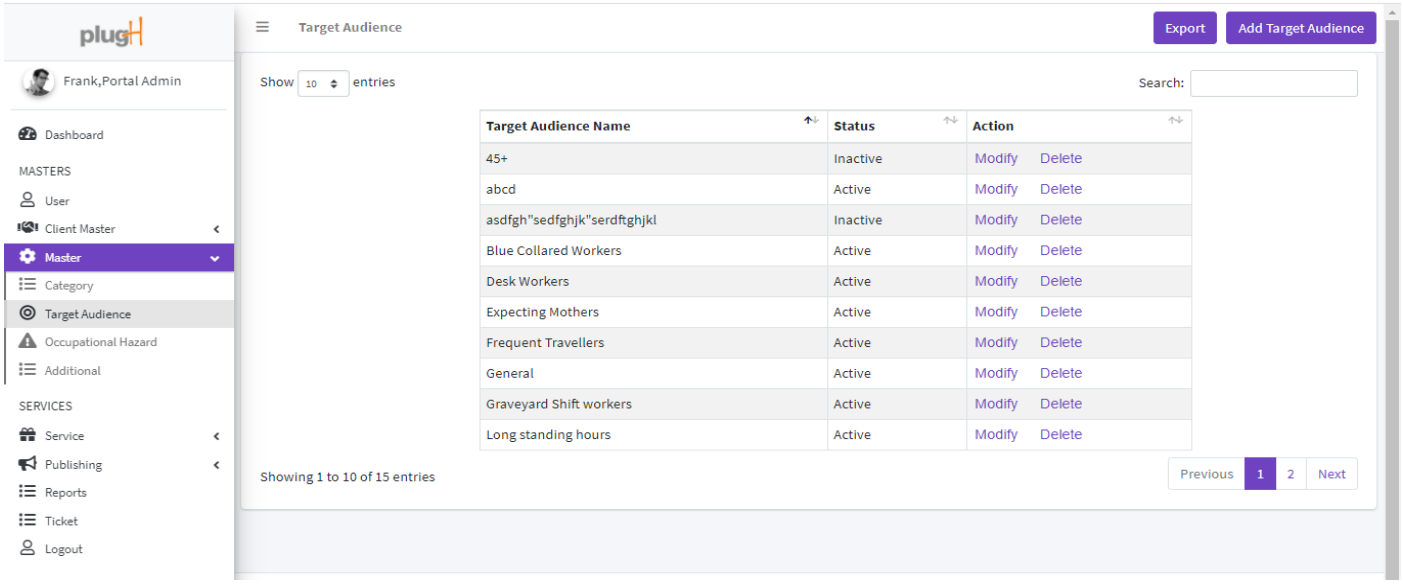


Fig 6.3.1.3.2 Target Audience

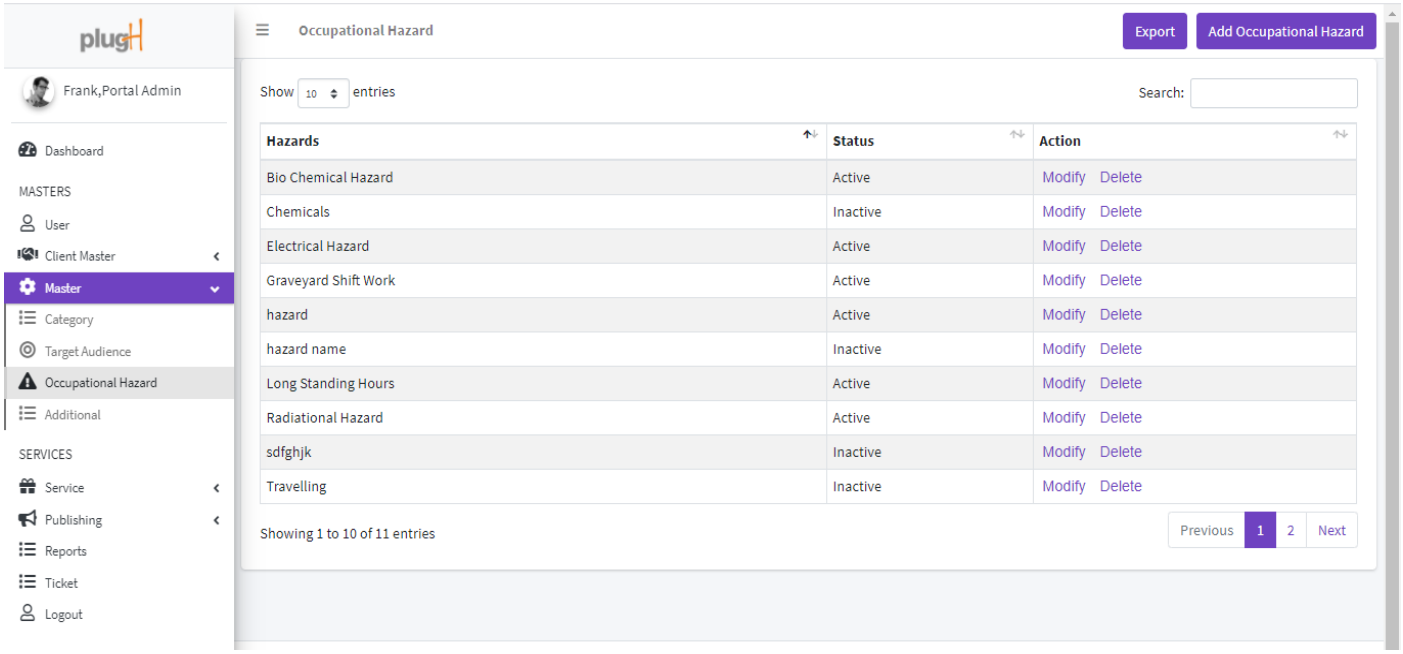


Fig 6.3.1.3.3 Occupational Hazard

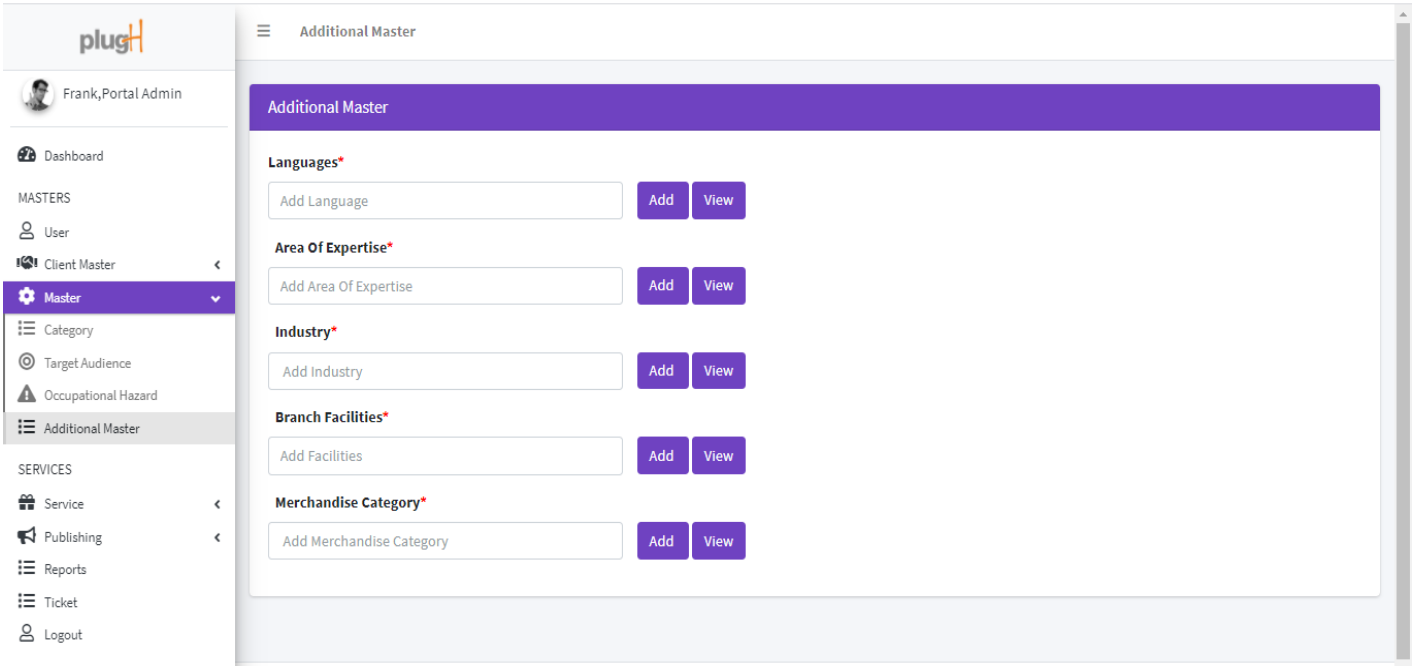


Fig 6.3.1.3.4 Additional Masters

6.3.1.4 SERVICE

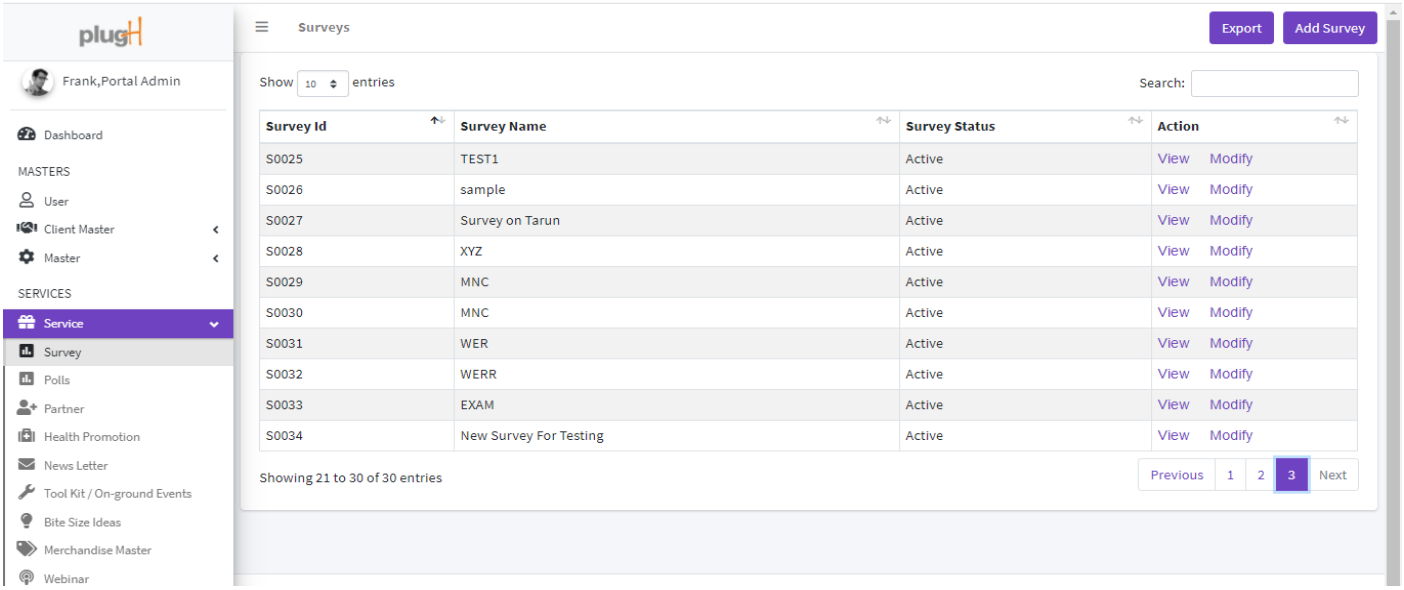


Fig 6.3.1.4.1 Survey

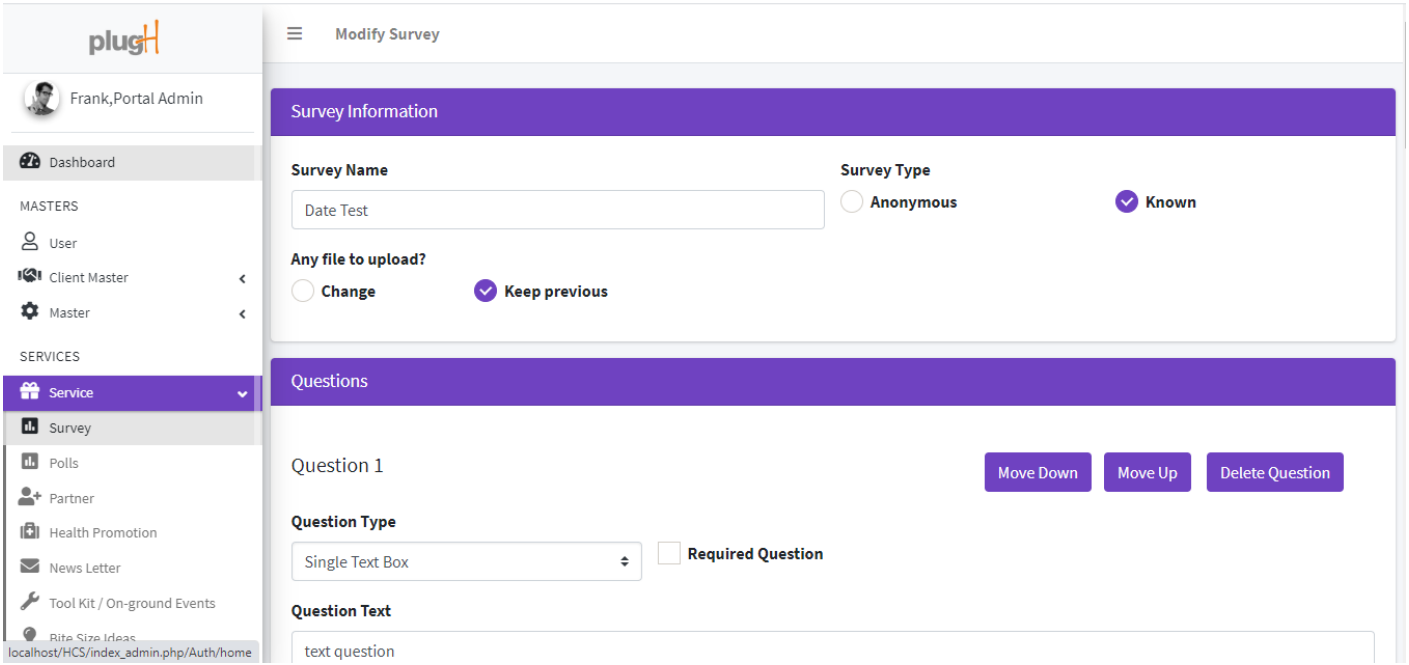


Fig 6.3.1.4.2 Modify Survey

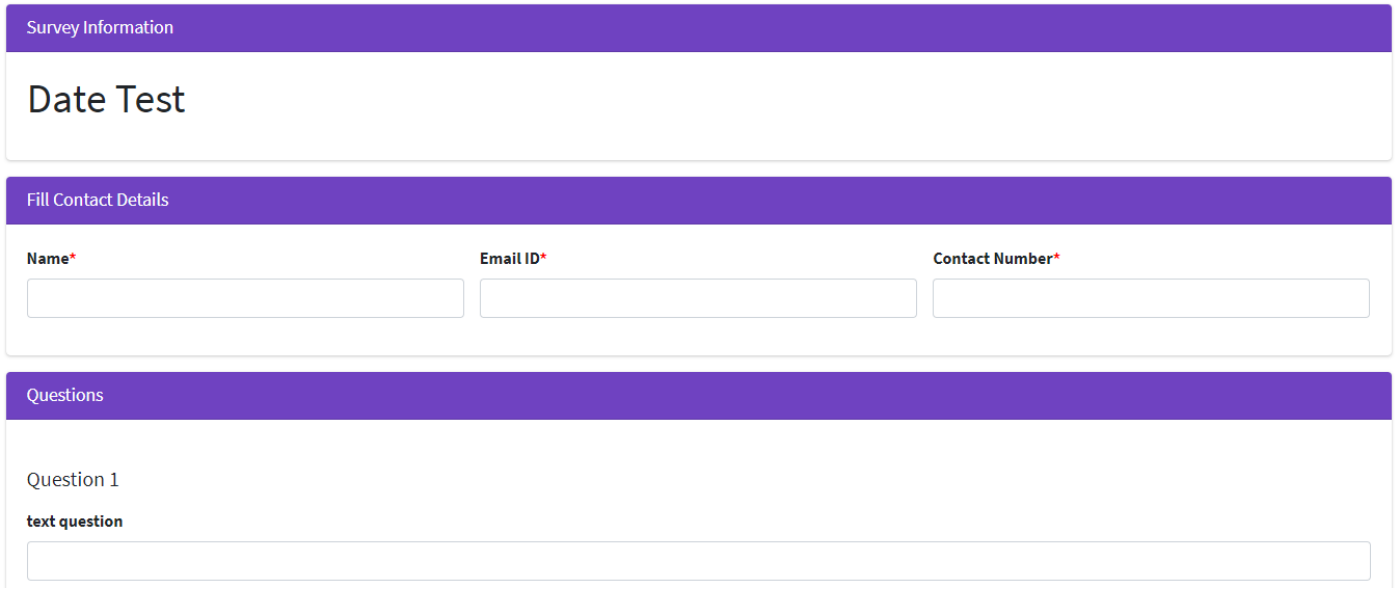


Fig 6.3.1.4.3 Take Survey

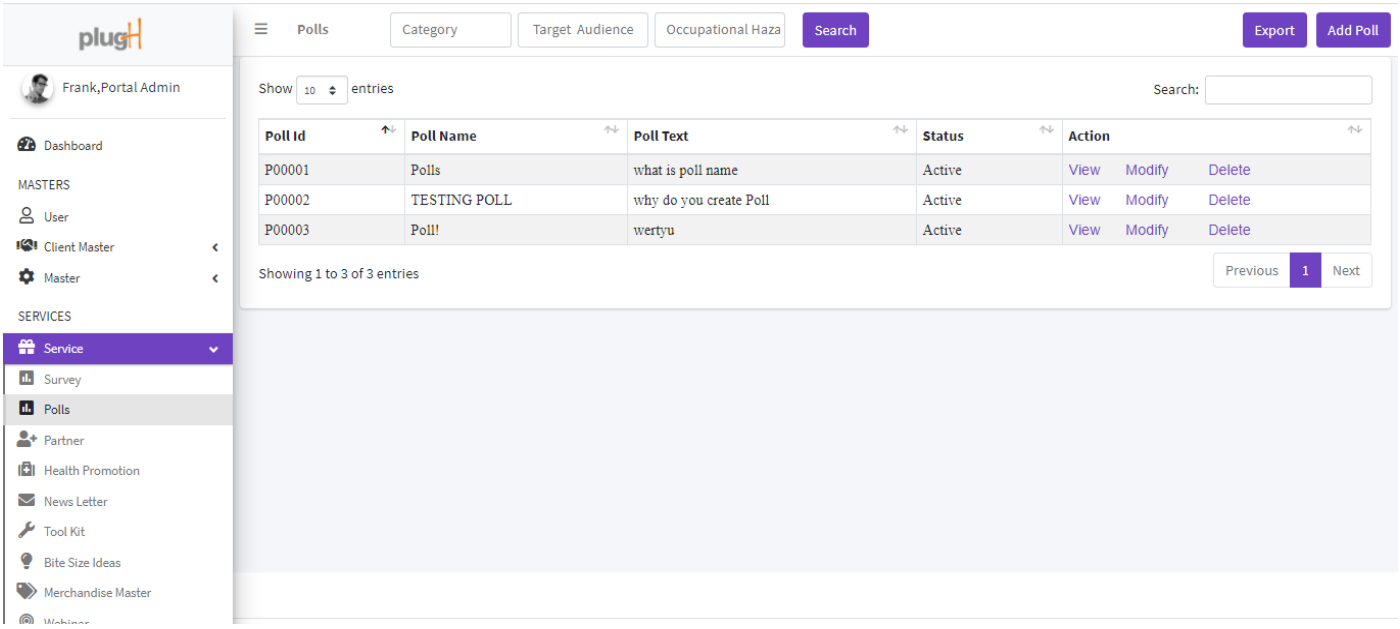


Fig 6.3.1.4.4 Polls

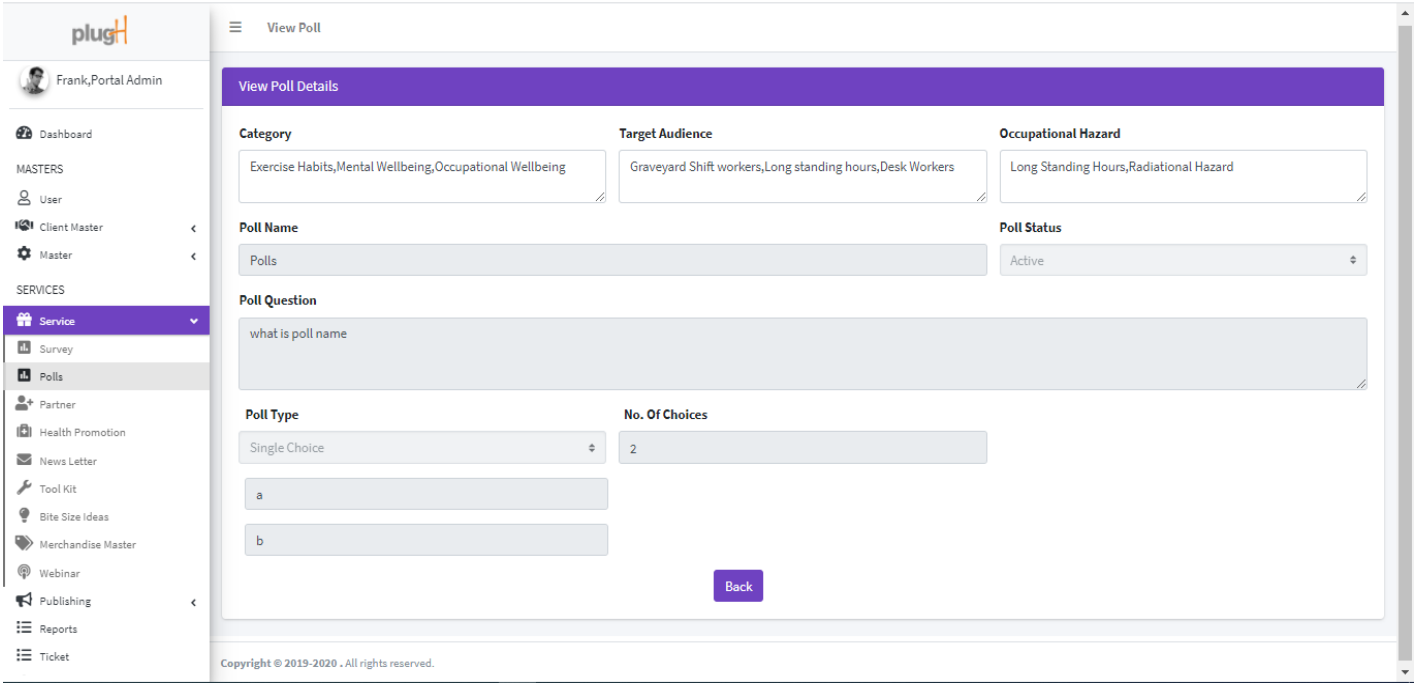


Fig 6.3.1.4.5 View Poll

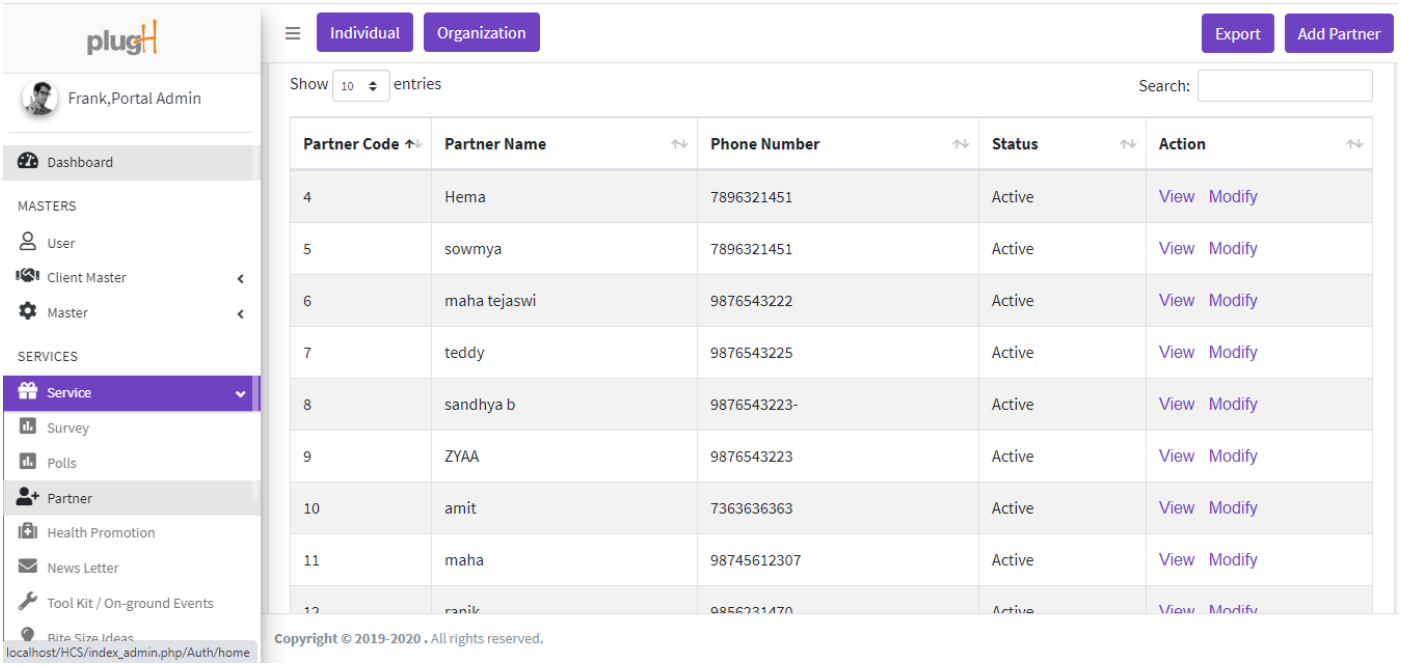


Fig 6.3.1.4.6 Partner Individual

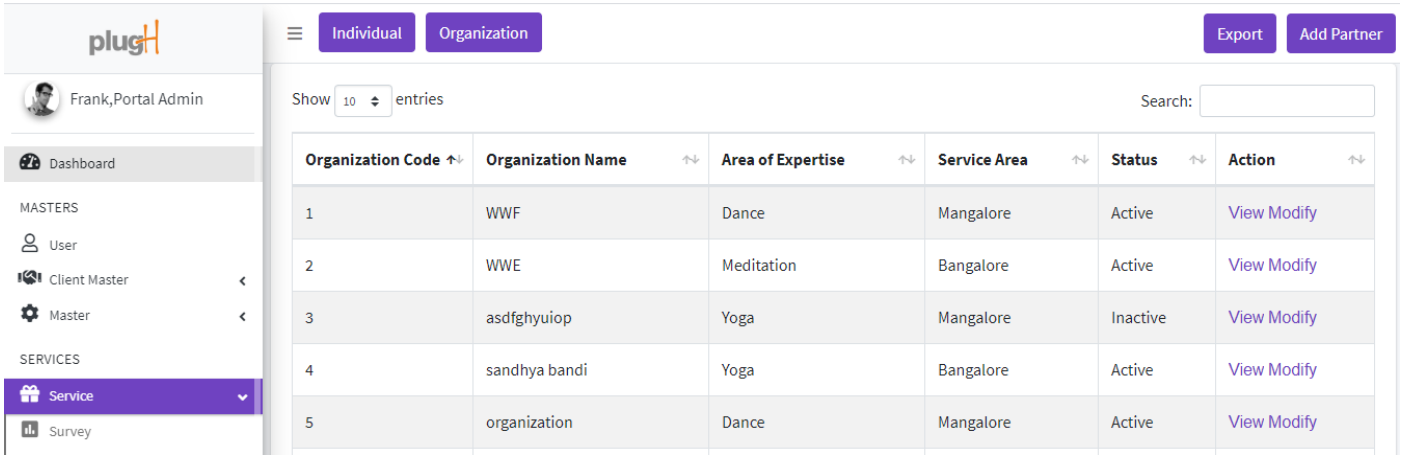


Fig 6.3.1.4.7 Partner Organisation

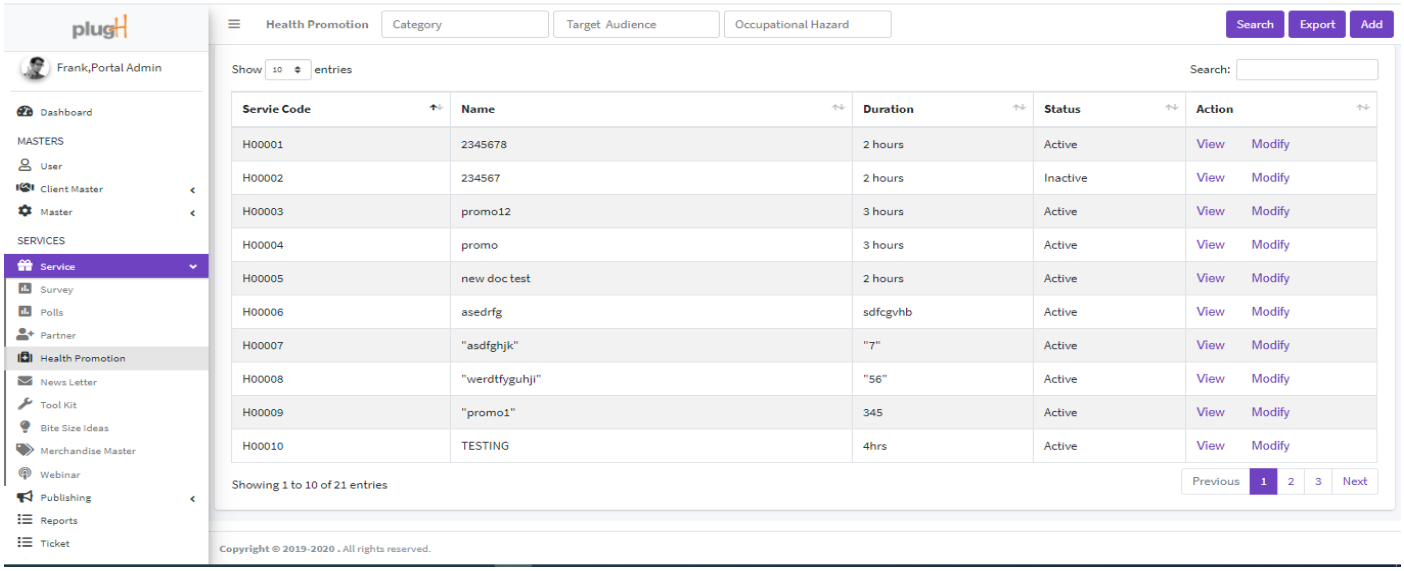


Fig 6.3.1.4.8 Health Promotion

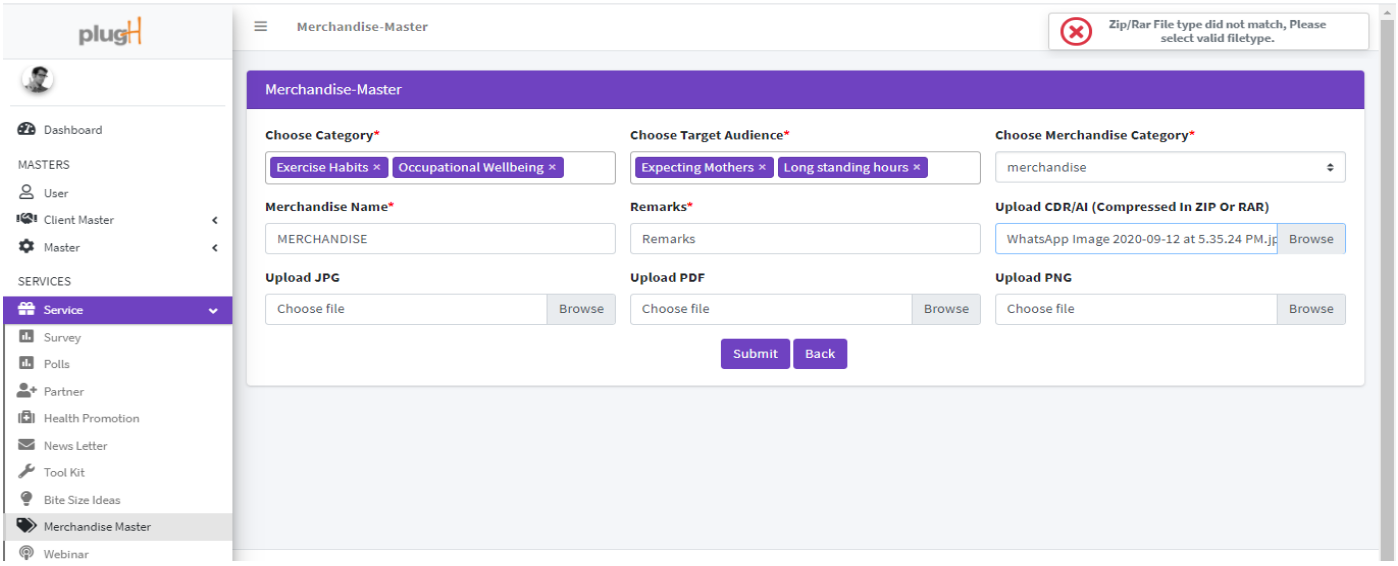


Fig 6.3.1.4.9 Merchandise Master

6.3.2 HR MODULE

6.3.2.1 LOGIN

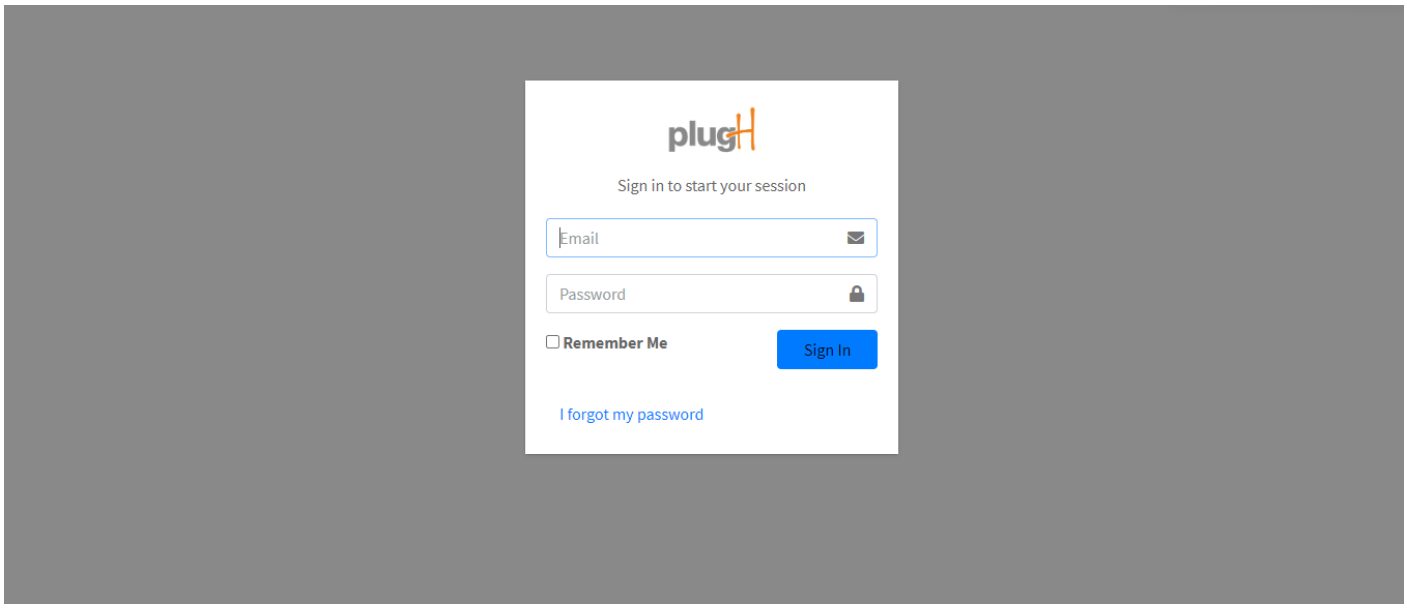


Fig 6.3.2.1 Login

6.3.2.2 DASHBOARD

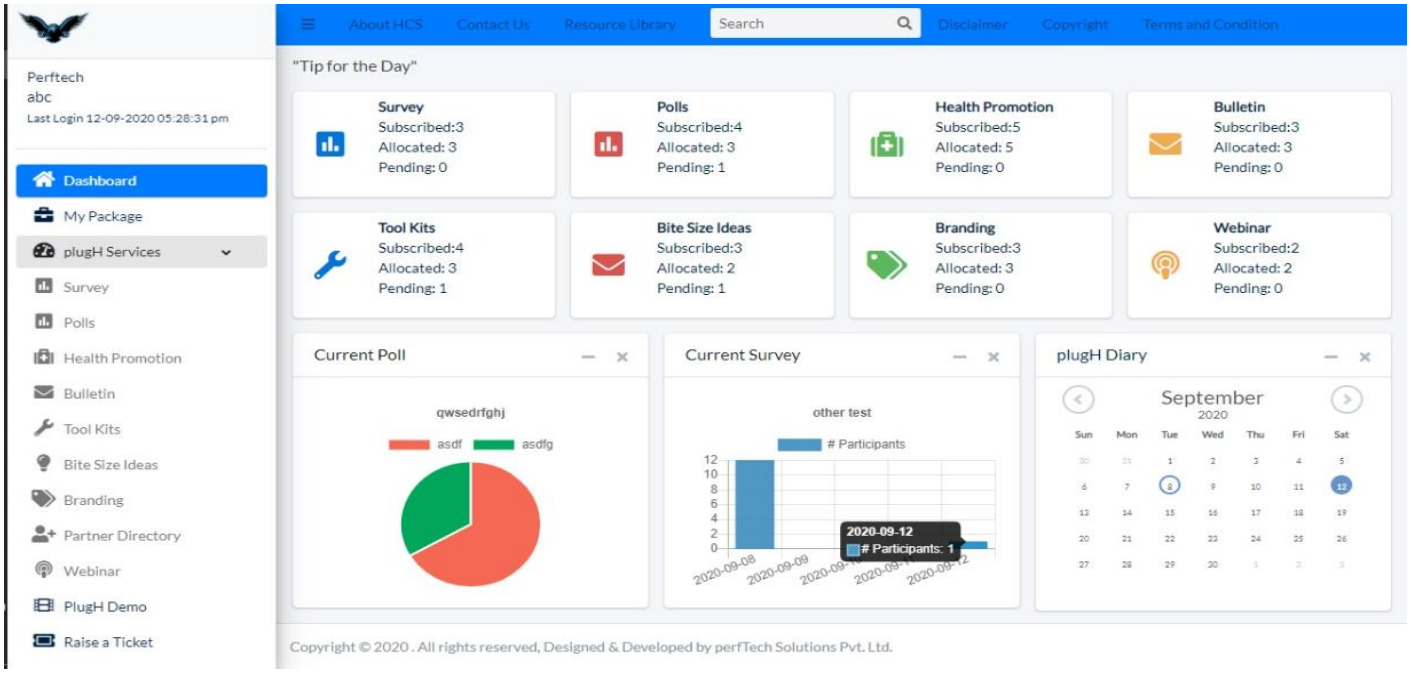


Fig 6.3.2.2 Dashboard

6.3.2.3 HCS SERVICE

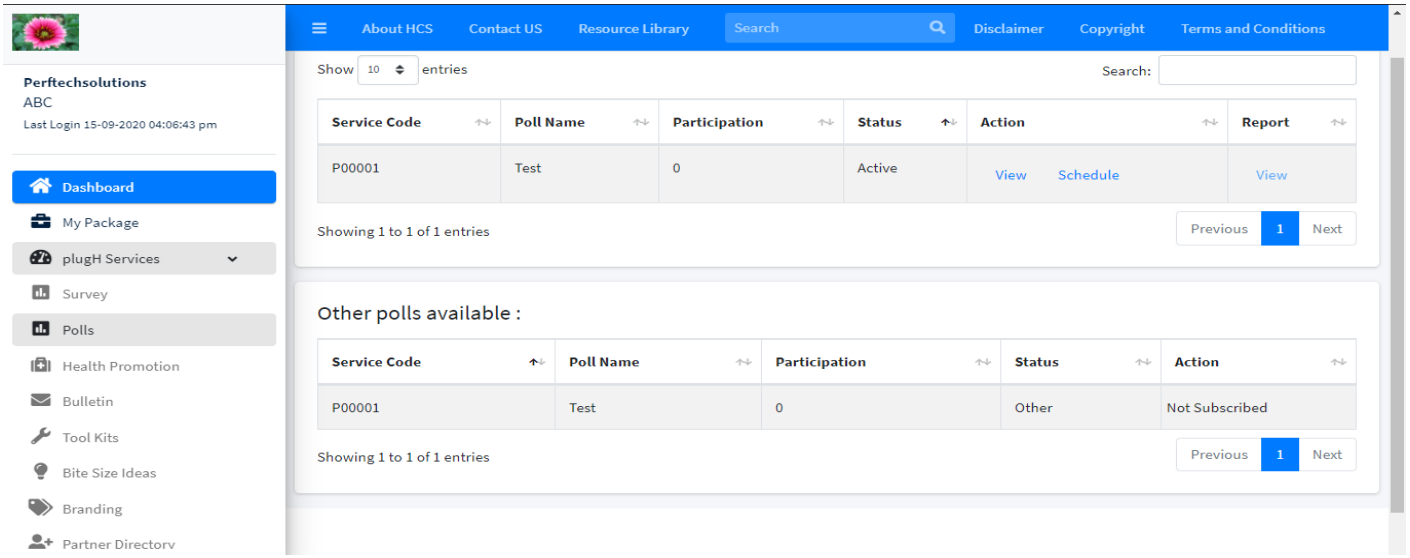


Fig 6.3.2.3.1 Polls

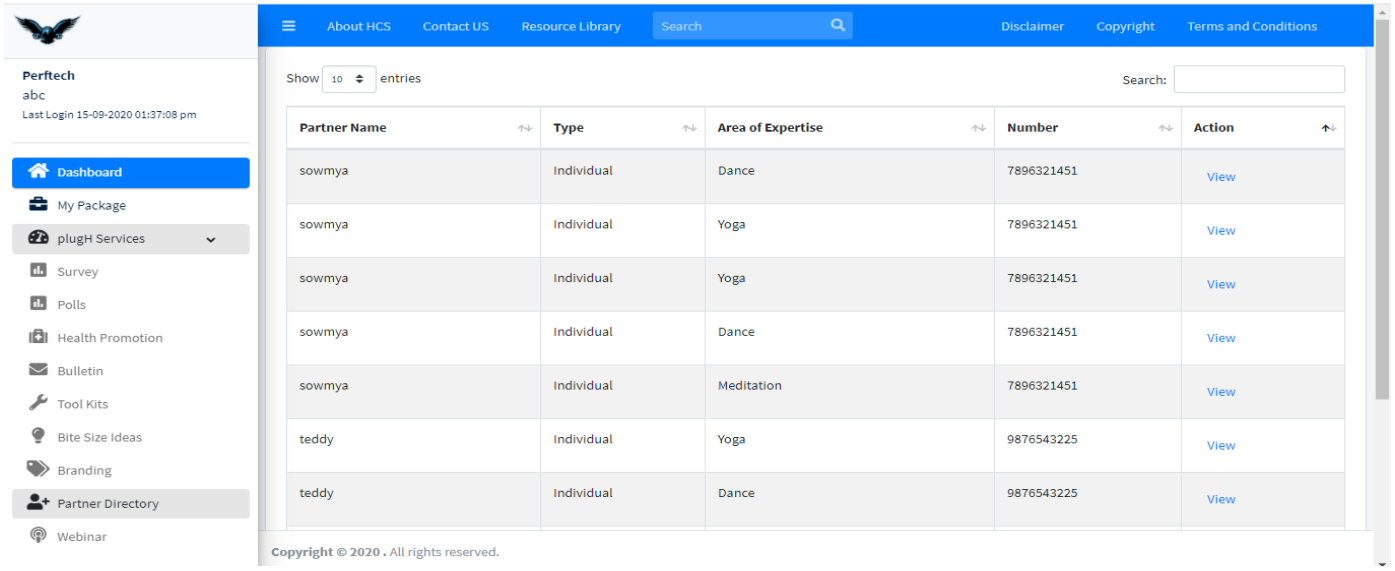


Fig 6.3.2.3.2 Partner

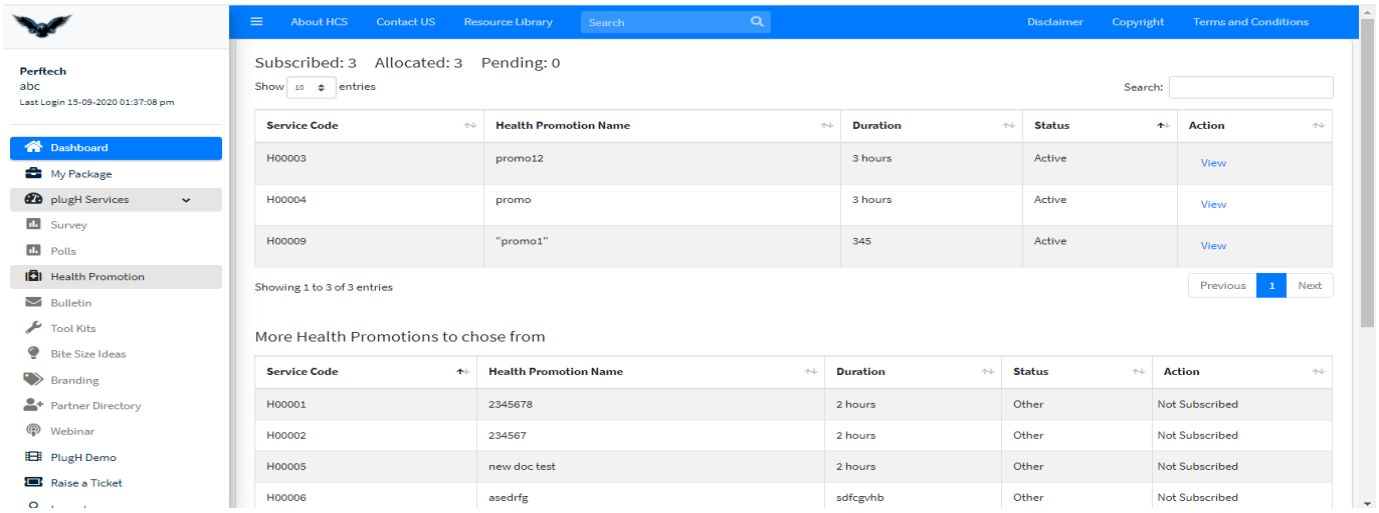


Fig 6.3.2.3.3 Health Promotion

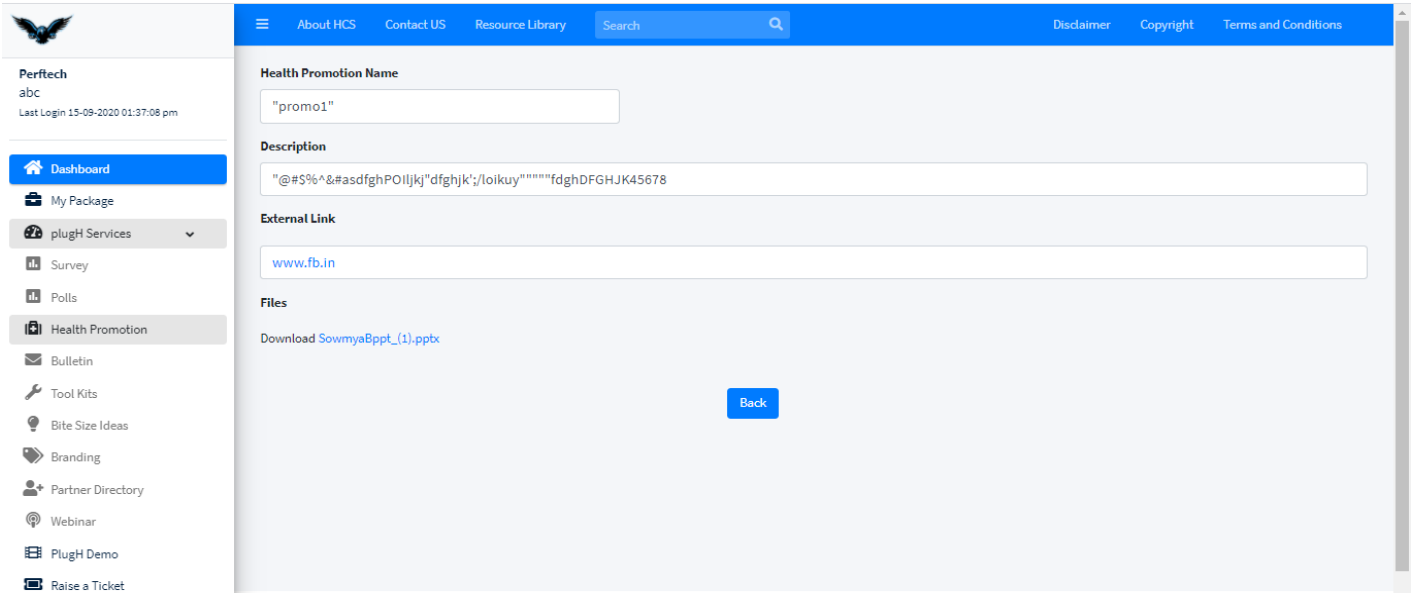


Fig 6.3.2.3.4 View Health Promotion

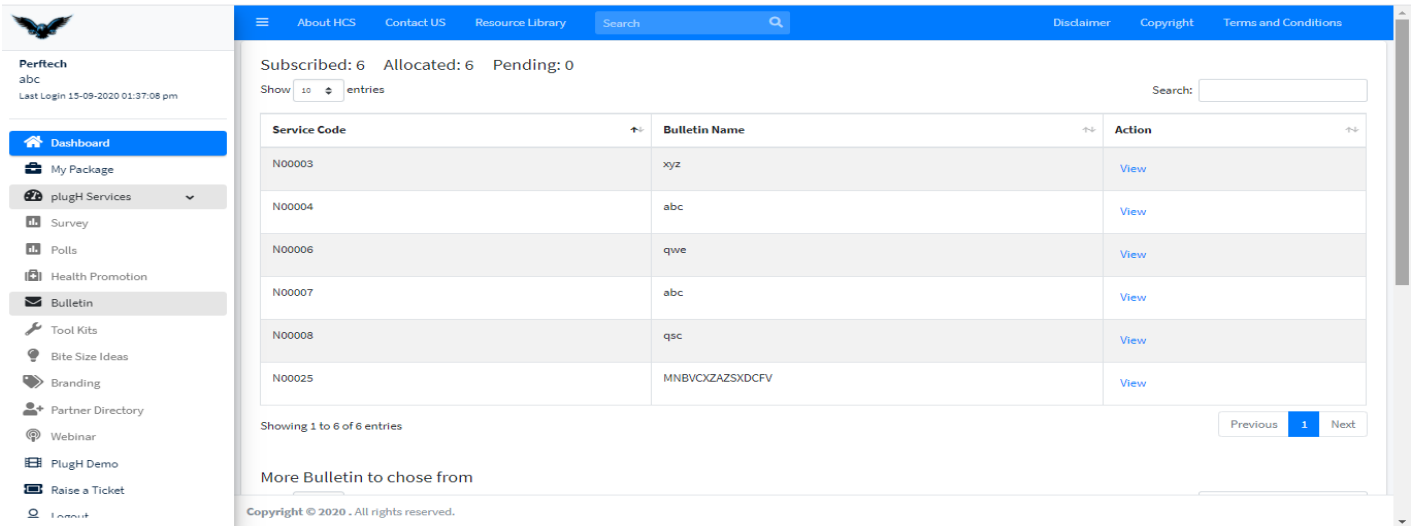


Fig 6.3.2.3.5 Newsletter [Bulletin]

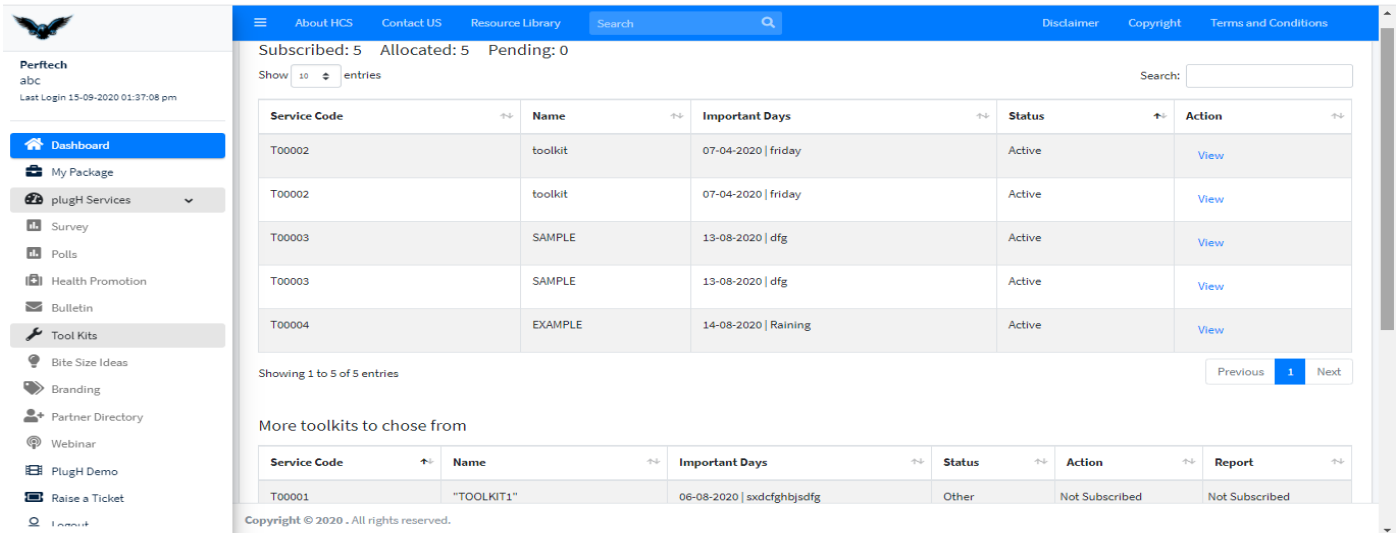


Fig 6.3.2.3.6 Toolkit

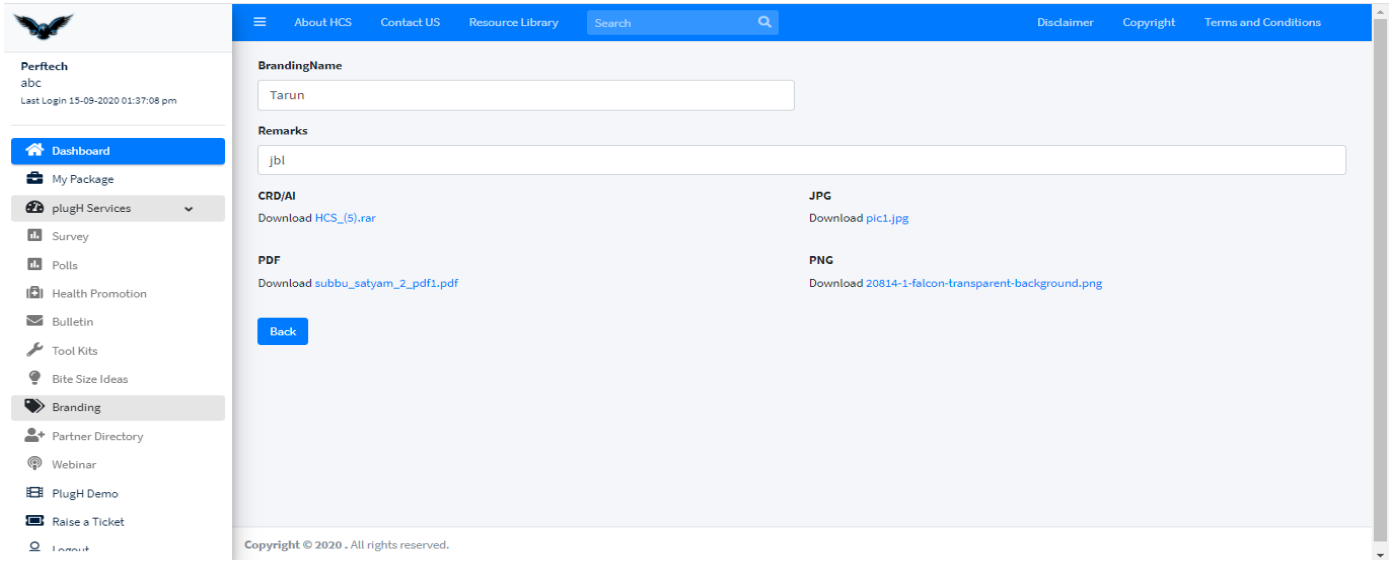


Fig 6.3.2.3.7 Merchandise-Master[Branding]

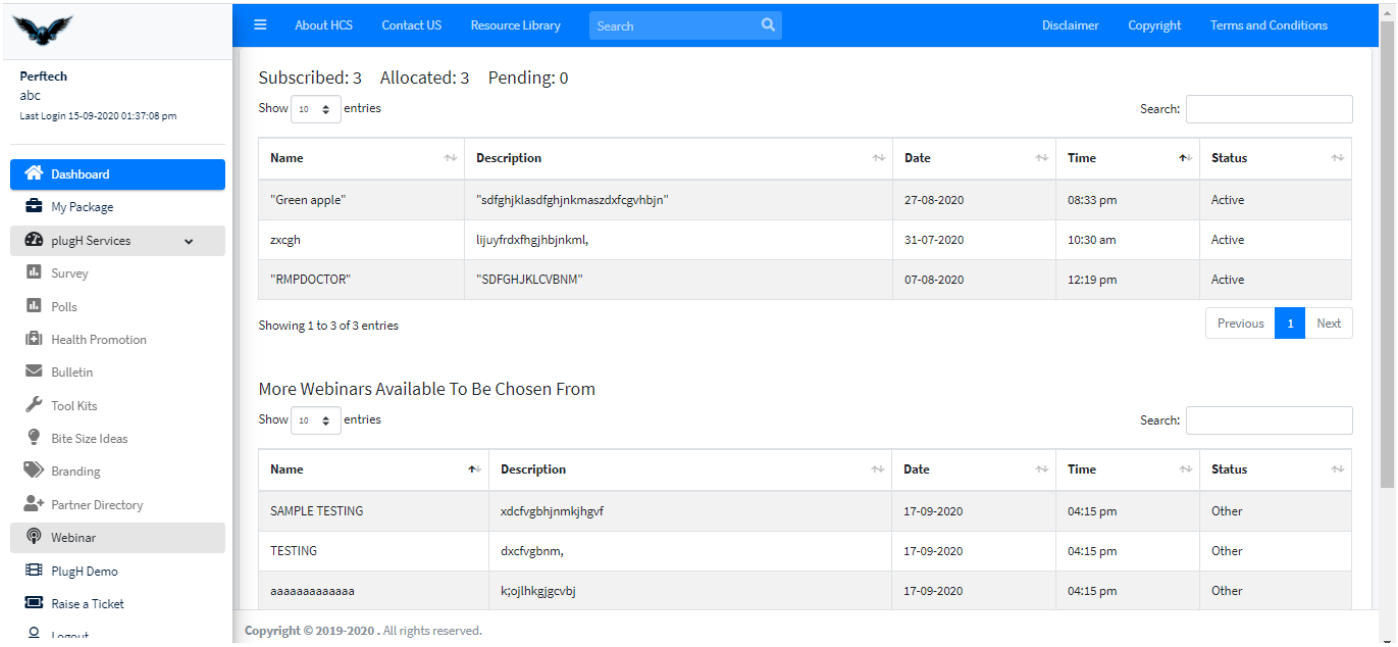


Fig 6.3.2.3.8 Webinar

7. VALIDATION

In this project I have used the validations for entering proper email format, entering only digits; the required fields cannot be left blank.

There are some validation controls are available. Using those controls, I have validated the textbox's and other controls.

The validation controls which I have been used are:

- Required field validator
- Regular Expression validator
- Range validator
- Comparison validator

Required Field Validator:

The required field validator is used to validate the particular text box in such a way that if any text box is left blank then it reminds you with a message that require fields can't be left blank.

Regular Expression:

This validation control can be used for validating the email formats and user defined expressions also it validates.

Range Validator:

This validator is used for giving the range for a text box using this I validate the phone number.

Comparison Validator:

This validator is used for compare two textboxes using this I validate the password.

Security Software

System security refers to various validations on data in form of checks and controls to avoid the system from failing. It is always important to ensure that only valid data is entered and only valid operations are performed on the system. The system employees two types of checks and controls:

Client-Side Validation:

Various client-side validations are used to ensure on the client side that only valid data is entered. Client-side validation saves server time and load to handle invalid data. Some checks imposed are:

It is used to ensure those required fields are filled with suitable data only. Maximum lengths of the fields of the forms are appropriately defined.

Forms cannot be submitted without filling up the mandatory data so that manual mistakes of submitting empty fields that are mandatory can be sorted out at the client side to save the server time and load.

Tab-indexes are set according to the need and taking into account the ease of user while working with the system.

Server-Side Validation:

Some checks cannot be applied at client side. Server-side checks are necessary to save the system from failing and intimating the user that some invalid operation has been performed or the performed operation is restricted. Some of the server-side checks imposed is:

- Server-side constraint has been imposed to check for the validity of primary key and foreign key.
- A primary key value cannot be duplicated.
- Any attempt to duplicate the primary value results into a message intimating the user about those values through the forms using foreign key can be updated only of the existing foreign key values.
- User is intimating through appropriate messages about the successful operations or exceptions occurring at server side.

Various Access Control Mechanisms have been built so that one user may not agitate upon another. Access permissions to various types of users are controlled according to the organizational structure. Only permitted users can log on to the system and can have access according to their category. User- name, passwords and permissions are controlled on the server side. Using server-side validation, constraints on several restricted operations are imposed.

8. CONCLUSION AND FUTURE SCOPE

The push to create a healthy work environment and increase employee wellness can be intense in big and small businesses. Increasingly, companies of all sizes and across all industries are improving their employee wellness programs. Companies that strive to make the work experience better for employees can attract more people, thus increasing competition among applicants. With more applicants to choose from, companies can find it easier to hire better quality employees overall.

A great company culture can keep employees productive and even help your company retain the best employees and happy workers are also more likely to stick around. This project is developed with an initiative to help every kind of organization from small scale to large scale in every industry.

This Project is humble venture to satisfy the needs to manage all the client details in the database securely. Several users friendly coding will be adopted. This project shall prove to be a powerful package in satisfying all the requirements of the clients. Customer satisfaction level measures will be high on quality of services provided; this project is error free, reliable and fast.

The experimental results and security analysis indicate that our scheme holds much higher security than the existing solutions with a reasonable overhead for cloud applications. To the best of our knowledge, until now this is the best web-app which is helpful for organisations to support and provide the methods and services for turning themselves into a workplace that ensures their employee's safety and occupational health.

The website is user friendly for both Admin and Client (HR) and can be beneficial to each other.

FUTURE SCOPE:

- The application will be made available for the current version of android and also IOS.
- The type of services being provided will be increased.
- The application will be extended to provide self-assessment for the client's employees.

9. BIBILOGRAPHY

- Software engineering, K K Agarwal, New age publisher, Third Edition, 2010.
- Software engineering, Rajib MALL, PHI Learning, Fifth Edition, 2010.
- The Complete Reference SQL, Paul Weinberg, James Groff, Andrew Oppel, Tata Mc Graw Hill Education Private Limited, Third Edition, 2010.
- HTML 4, Rick Darnell, Tech media, Second Edition, 2010.

Websites:

- https://codeigniter.com/userguide3
- https://www.tutorialspoint.com/php
- https://www.w3schools.com/
- https://www.geeksforgeeks.org/php