**ERP PROJECT:-**

**Technologies used:-**

**Backend:-** Codeigniter(3.0) with hmvc.

**Frontend:-** Html5, css 3, javaScript, jQuery (1.12.4),bootstrap 3.3.6,

**Template** :- Admin ssb material (link- <https://gurayyarar.github.io/AdminBSBMaterialDesign/>)

**What is erp?**

Enterprise resource planning or ERP software is a suite of applications that manages core business processes, such as sales, purchasing, accounting, human resource, customer support, CRM and inventory. It’s an integrated system as opposed to individual software designed specifically for business process.

**Modules used** :-

* Super\_admin (make admin).
* Home (for dashboard and admin).
* Profile (for change username, password and self profile).
* Login.
* Account.
* Item.
* Purchase.
* Sales
* Crn ( for add customer vendors)
* Category
* Service
* Staff
* Report
* Ticket
* Log
* Setting
* Quotation
* Payroll
* Leave
* Group
* Sms
* Email
* Task

**Online url :**- <erp.enextit.com>

**Database url :- erp.enextit.com/phpmyadmin**

**Project Description :-** BasicallyThis is Erp project. In this mainly crm (customer relationship manager, Inventory management system, Ticket management system, Payroll management system, Employee management system, leave management system are included.

We can check sales report, purchase report, invoice report, receipt report, payment report, customer & vendor ledger report, stock report date wise .

**Authorization User-** In this project 3 type of users

* Super admin
* Admin
* Employee

**Super admin**:- Super admin only one , its role create new seller panel, if any customer need this so super admin add seller company and details and seller credential is created automatically in db, and if sms service is active so credential send to customer by sms.

**Admin:-**

* Admin can be multiple of same seller or same company. Admin have all the right of edit, delete, create.
* Admin can approve and reject purchase and sales order.
* Admin can create employee and give him credential.
* Admin can see attendance report and take attendance.
* Admin can create another admin and give some responsibility to him.
* Admin can change his username & password.

**Employee :-**

* Employee have a role of add purchase and sales & services in inventory.
* Employee can check his attendance and payroll information.
* Add ticket if any complain arise.
* Solve ticket and add log in ticket.
* Employee can change his username & password.

**Generic model for developer**

in Codeigniter framework location:- project\_name/application/core/MY\_Model

to make code reusable and curd operation easily we make this generic model library,in this we have to pass only parameter with its function name and just extends this model then this will work.

This model is directly extends by CI\_Model (class name is case sensitive in linux server so mbe carefull).

All the table name of whole project displayed here by protected access identifier, so if in future table name will change so we can change here, so no code has been disturbed

**Functions**

* **select($table\_name,$condition,$content\_display,$order)**

Use to fetch data array for table by condition, $content\_display is used to display what data we want for table.

* **insert($table\_name,$params)**

Use to insert data in table.

* **delete($table\_name,$condition)**

Use to delete data by condition in table

* **update($condition,$table\_name,$params)**

Use to update whole row of table

* **select\_id($table\_name,$condition=null,$content\_display,$order=null)**

fetch only one row with condition

* **update\_col($table\_name,$condition,$params)**

Update column by condition and update particular column by $params

* **report($table\_name,$display\_contents,$condition,$start\_date='',$end\_date='',$status='',$coloumn='created\_at')**

give report of table by start date to end date condition wise

* **data\_between\_date($table\_name,$display\_contents,$condition,$interval\_unit,$table\_row='created\_at',$group\_by='',$order\_by='')**

fetch data between date and interval unit like(1 month, 1 year, 7 days).

* **data\_current($table\_name,$display\_contents,$condition,$interval\_unit,$table\_row='created\_at',$start\_date='',$end\_date='')**

fetch data of current month and current year.

* **data\_between\_date\_count($table\_name,$display\_contents,$condition,$interval\_unit='',$start\_date='',$end\_date='')**

fetch number of rows count in table with date andinterval unit like(1 month, 1 year, 7 days).

* **data\_current\_count($table\_name,$display\_contents,$condition,$interval\_unit,$start\_date='',$end\_date='')**

fetch number of rows count in table with date of current year month

* **data\_sum\_between\_date($table\_name,$display\_contents,$condition,$interval\_unit,$coloumn='created\_at',$start\_date='',$end\_date='')**

sum data of table with specific row of between two dates.

* **data\_sum\_current($table\_name,$display\_contents,$condition,$interval\_unit,$coloumn='created\_at',$start\_date='',$end\_date='')**

sum data of table with specific row of between two dates and current year or current month.

For hmvc what is apply on this project-:

* Download hmvc plugin for codeigniter and paste in application/third\_party/.
* Create folder modules in Application/

In application/config/config.php add ($config['modules\_locations'] = array(APPPATH.'modules/' => '../modules/');

Create new module in project (hmvc):-

Go to Application-> modules->Add new folder ->give name(example- admin) lower case

->create 3 folder (controllers,models,views)

* Under controllers create new file with same controller and modules name. controllers file name should start. Example (Admin.php) .
* After create file in controller, extends class with MX\_Controller.
* In this project I create MY\_Controller in application/core which extends with MX\_Controller .we can add authentication process and the process which is common to all the modules.
* Class name always start with capital letter(it is rule)

**NOTE:** If you not follow this structure you phase 404 error because linux server is case sensitive.

**Application/views folder->**

*dashboard :-use to display dashboard.*

*header.php:- all html,head ,css file are here, this view is common to all.*

*Index.php :-* in this file header and footer are included

*$data['\_view'] =’file name’; (example-add\_sales);*

*$this->load->view(‘index',$data);*

Modules description :-

Method format in controller- snake\_case(example add\_sales())

Login module :- used to login, logout, forgot password.

Home Module:- After login if (authorization id=2 and authorization id=3) Home module dashboard is open.

Accounting in project

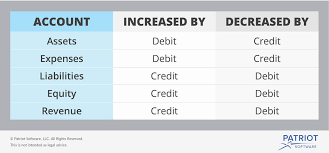
In this project accounting is done by double entry

In ledger group table in database accounting ledger group



Here parent id donate that particular group is a child of id this(example capital account parent\_id 14 so it is child of liabilities.

So accounting debit credit is based on this law



So use this theorem use double entry .

Equity=total assets- total expenses.

It all be work in balance sheet, account transactions, .