**Health Insurance Management Portal**

**Application Design**

**V1.0**

****

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Prepared By** | **Reviewed by** | **Approved By** |
| **Name** | Sandhya B |  |  |
| **Role** | SA | Project Leader | Project Manager |
| **Signature** |  |  |  |
| **Date** | 05/16/2013 |  |  |

Table of Contents

[1.0 Introduction 3](#_Toc356570072)

[1.1 Purpose of this document 3](#_Toc356570073)

[1.2 Scope 3](#_Toc356570074)

[1.2.1 In Scope 3](#_Toc356570075)

[1.2.2 Out Scope 3](#_Toc356570076)

[1.3 Intended Audience 3](#_Toc356570077)

[1.4 Definition & Acronyms 3](#_Toc356570078)

[1.5 References 3](#_Toc356570079)

[2.0 Conventions and Standards Followed 5](#_Toc356570080)

[3.0 Assumptions, Dependencies and Risks 5](#_Toc356570081)

[*4.0* Use Case Realization 6](#_Toc356570082)

[4.1 Flow of Events – Design 6](#_Toc356570083)

[4.1.1 Admin Login Sequence 6](#_Toc356570084)

[4.1.2 Employee Login Sequence 6](#_Toc356570085)

[4.1.3 Adding Employee Sequence 7](#_Toc356570086)

[4.1.4 Apply for Policy Sequence 7](#_Toc356570087)

[4.1.5 Adding Policies on Employee Sequence 8](#_Toc356570088)

[*5.0* Object Model 10](#_Toc356570089)

[5.1 System Object Model 10](#_Toc356570090)

[5.2 UI Controls 11](#_Toc356570091)

[5.3 UI Design 13](#_Toc356570092)

[5.3.1 Add/Edit Employee 13](#_Toc356570093)

[5.3.2 View Employee 14](#_Toc356570094)

[5.3.3 Add Insurance 14](#_Toc356570095)

[5.3.4 Add Policy 14](#_Toc356570096)

[5.3.5 Add Policy on Employee 15](#_Toc356570097)

[5.3.6 Employees based on Policies 15](#_Toc356570098)

[5.3.7 Policy Request Details 15](#_Toc356570099)

[5.4 Object Descriptions 16](#_Toc356570100)

[6.0 Database Design 20](#_Toc356570101)

[6.1 Identification of Tables 20](#_Toc356570102)

[6.2 Table Structure 21](#_Toc356570103)

[7.0 References 25](#_Toc356570104)

[8.0 Change Log 26](#_Toc356570105)

# Introduction

## Purpose of this document

*<The detailed design contains the design of every module / sub-system. This covers the following:*

* *Design of common / generic components*
* *Module / Sub-system design*
* *Unit / Class / Program Specifications*

*From a content consideration, the above three have been combined in this document. However depending on the scale of design or projects, from manageability and configurability point of view, the above three may be split and maintained>*

## Scope

### In Scope

1. Administrative user interface
2. Employee user interface

* The administrative user interface concentrates on the consistent information that is practically, part of the organizational activities and which needs proper authentication for the data collection. The interfaces help the administrations with all the transactional states like Data insertion, Data deletion and Data updating along with the extensive data search capabilities.
* The operational or generic user interface helps the users upon the system in transactions through the existing data and required services. The operational user interface also helps the ordinary users in managing their own information in a customized manner as per the assisted flexibilities.

### Out Scope

* Policy payments/transactions

## Intended Audience

*The intended audience for this document is*

* *Developers, testers, other associates in the project*
* *Client*
* *Delivery Assurance Group, Process Engineering group >*

## Definition & Acronyms

## References

Health Insurance Management\_BRD\_V1.0

# Conventions and Standards Followed

*Java Coding Standards*

# Assumptions, Dependencies and Risks

**Assumptions**:

* Any changes to the design post sign-off would be taken as a Change Request.

**Dependencies**

NA

**Risk**

* Any delay in obtaining the sign-off for the design would have an impact in the schedule of the project

# Use Case Realization

## Flow of Events – Design

### Admin Login Sequence

Time

Admin

Home Page

Database

Admin Home Page

Use URL

Press login button

If Yes Goes to its Home Page

If No Come Back to Home Page

Validate if NO

### Employee Login Sequence

Database

Employee Home Page

Home Page

Company Employees

Use URL

Press login button

Validate if NO

If No Come Back to Home Page

If Yes Goes to its Home Page

Time

### Adding Employee Sequence



### Apply for Policy Sequence

Home Page

Login Page

Employee home

Employee

Use URL

Press Login Button

If Yes Goes toEmpoloye Home Page

Validate If NOT

Come Back to Login Page

Request for policy

Database

Confirm Page

Click on Link for request Page

Press Button for Saving Data

if Validation NOT OK

Back to add policy page

If OK Then go to Confirmation Page

### Adding Policies on Employee Sequence

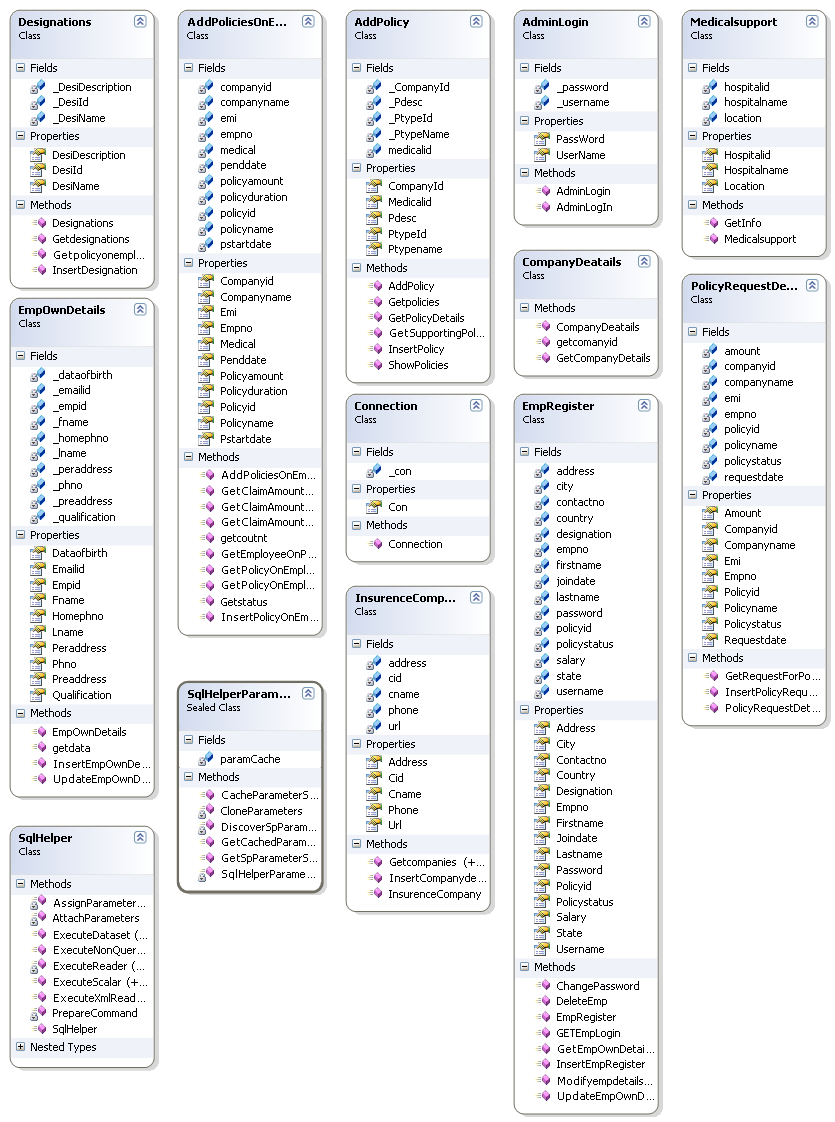


Derived Requirements

*<Description that collects all requirements, such as non-functional requirements, on the use-case realizations that are not considered in the design model, but that need to be taken care of during implementation>*

# Object Model

## System Object Model



## UI Controls

**Add/Edit Employee**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Data Element** | **Control type** | **Default Values** | **Editable Field** | **Restrictions** | **Mandatory/**  **Non Mandatory** | **Data Type** | **Data**  **Size** |
| Designation | Textbox | No | Yes | Only Alphabets | Mandatory | Varchar | 50 |
| Join Date | Textbox with Calendar control | No | Yes | Valid date format | Mandatory | datetime |  |
| Salary | Textbox | No | Yes | Numeric |  | money |  |
| First Name | Textbox | No | Yes | Only Alphabets | Mandatory | Varchar | 30 |
| Last Name | Textbox | No | Yes | Only Alphabets | Mandatory | Varchar | 30 |
| User Name | Textbox | No | Yes | Only Alphabets | Mandatory | Varchar | 50 |
| Password | Textbox | No | Yes | Alphanumeric | Mandatory | Varchar | 50 |
| Address | Textarea | No | Yes |  | Mandatory | Varchar | 150 |
| Contact No | Textbox | No | Yes | Valid Date | Mandatory | Date | - |
| State | Textbox | No | Yes | Only Alphanumeric | Mandatory | Varchar | 50 |
| Country | Textbox | No | Yes | Only Alphabets | Mandatory | Varchar | 50 |
| City | Textbox | No | Yes | Only Alphabets | Mandatory | Varchar | 50 |

**Add Insurance Company**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Data Element** | **Control type** | **Default Values** | **Editable Field** | **Restrictions** | **Mandatory/**  **Non Mandatory** | **Data Type** | **Data**  **Size** |
| Company Name | Textbox | No | No | Alphanumeric | Mandatory | Varchar | 50 |
| Address | Text area | No | No |  | Mandatory | Varchar | 150 |
| Phone no | Textbox | No | No | Only Numeric | Mandatory | Varchar | 20 |
| Company URL | Textbox | No | No |  |  | Varchar | 50 |

**Add Policy**

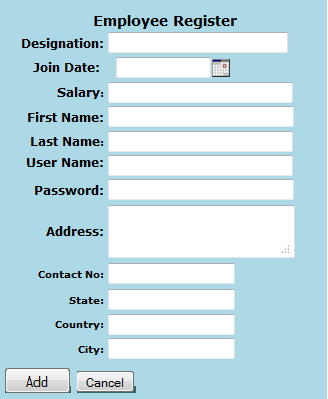
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Data Element** | **Control type** | **Default Values** | **Editable Field** | **Restrictions** | **Mandatory/**  **Non Mandatory** | **Data Type** | **Data**  **Size** |
| Policy Type Name | Textbox | No | No | Alphanumeric | Mandatory | Varchar | 50 |
| Policy Desc | Text area | No | No |  | Mandatory | Varchar | 150 |
| Policy Amount | Textbox | No | No | Numeric | Mandatory | Money |  |
| Policy EMI | Textbox | No | No | Numeric | Mandatory | Money |  |
| Company Id | Dropdown | Yes | No |  | Mandatory | Int |  |
| Medical Id | Textbox | N o | No | Alpha Numeric | Mandatory | Varchar | 50 |

**Add Policy on Employee**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Data Element** | **Control type** | **Default Values** | **Editable Field** | **Restrictions** | **Mandatory/**  **Non Mandatory** | **Data Type** | **Data**  **Size** |
| Emp Id | Textbox | No | No | Numeric | Mandatory | int |  |
| Policy Id | Dropdown | Yes | No |  | Mandatory | int |  |
| Policy Name | Textbox | Yes | No | Alphanumeric | Mandatory | Varchar | 50 |
| Policy Amount | Textbox | Yes | No | Numeric | Mandatory | Money |  |
| Policy EMI | Textbox | Yes | No | Numeric | Mandatory | Money |  |
| Policy Start Date | Textbox with Calendar control | No | No | Date format | Mandatory | Datetime |  |
| Policy End Date | Textbox with Calendar control | No | No | Date format | Mandatory | Datetime |  |
| Policy Duration | Textbox |  | No | Only numeric |  | int |  |
| Company Id | Dropdown | Yes | No |  | Mandatory | Int |  |
| Company Name | Textbox | Yes |  | Alpha Numeric |  | Varchar | 50 |
| Medical Id | Textbox | Yes | No | Alpha Numeric | Mandatory | Varchar | 50 |

## UI Design

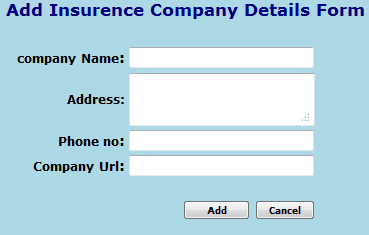
### Add/Edit Employee



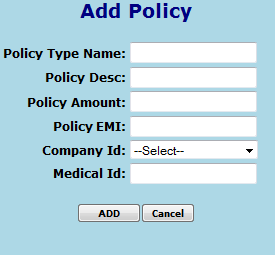
### View Employee

****

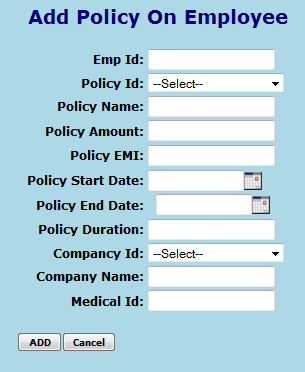
### Add Insurance

****

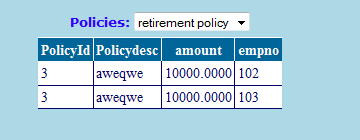
### Add Policy



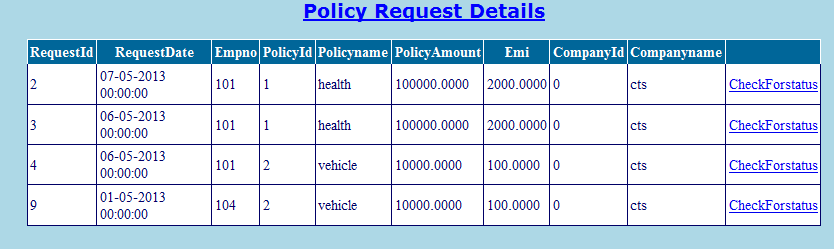
### Add Policy on Employee

****

### Employees based on Policies

****

### Policy Request Details

****

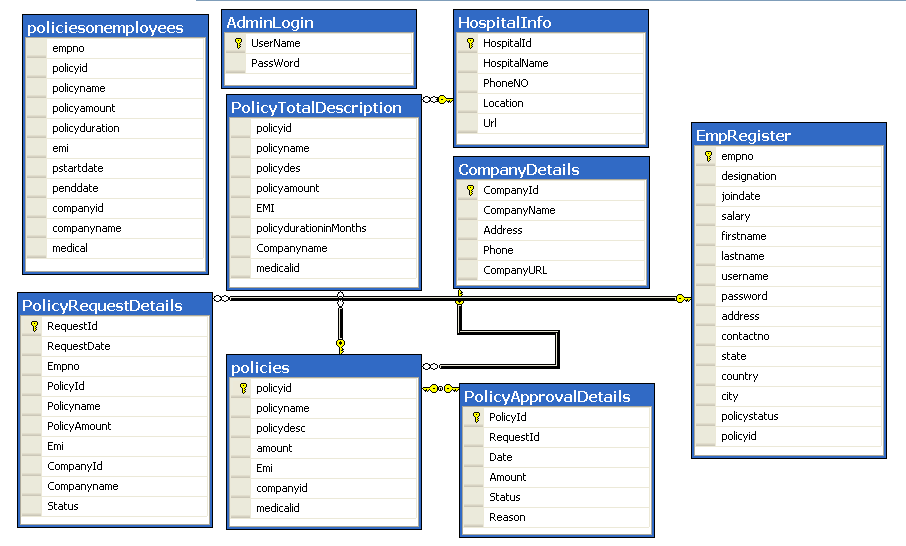
## Object Descriptions

|  |  |  |
| --- | --- | --- |
| **Class** | **Method** | **Program Logic** |
| UserController | **doGet**  **Parameters** HttpServletRequest request, HttpServletResponse response  **Return Type**  *void* | * Get the ‘action type’ from the request. * If the ‘action type’ is ‘edit’ call the UpdateEmpOwnDetails   () method of the EmpOwnDetails class by passing the employee id whose information needs to be updated as the input parameter.   * If the ‘action type’ is ‘listUser’ call the getData() method of the EmpOwnDetails class. * Set the Employee List obtained in request. * Retrieve the Request policies list by calling the GetRequestForPolicies() method of PolicyRequestDetails class. * Set the Policy Request List obtained in request. * Based on the Action type forward the response to the appropriate page using a RequestDispatcher. * If there is an error while performing the operation redirect to the Error page. |
| **doPost**  **Parameters**  HttpServletRequest request, HttpServletResponse  **Return Type**  void | * Get the ‘action type’ from the request. * Get the User details from request. * Set the Employee details obtained in the EmployeeTO object. * If the userid obtained from request is null call the addEmp() method else call the updateEmp() method of EmployeeBO class. * Retrieve the employee list by calling the getAllEmp() method of the EmployeeBO class. * Set the employee list in request. * Forward the response to the view page using a RequestDispatcher if the operation is successfully completed. Else redirect to the error page. |
| EmpRegister BO | InsertEmpRegister    **Parameters**  EmployeeTO employee  **Return Type**  void | * Call the InsertEmpRegister() of EmpRegisterBO by passing the employee. |
| **deleteEmp**  **Parameters**  int empId  **Return Type**  void | * Call the deleteEmp() of EmpRegister by passing the empId. |
| **UpdateEmpOwnDetails**  **Parameters**  EmployeeTO employee  **Return Type**  void | * Call the UpdateEmpOwnDetails() of EmpRegister by passing the empId. |
| **getAllEmp**  **Parameters**  None  **Return Type**  List<EmployeeTO> | * Call the getAllEmp() of EmpRegister |
| **getEmpById**  **Parameters**  int userId  **Return Type**  EmployeeTO | * Call the getEmpById () of EmpRegister by passing the user id. |

|  |  |  |
| --- | --- | --- |
| InsuranceCompanyBO | **doGet**  **Parameters**  HttpServletRequest request, HttpServletResponse response  **Return Type**  void | * Retrieve the Insurance company list by calling the getCompanies() method of InsuranceCompany class. * Set the Department List obtained in request. * Forward the response to the appropriate page using a RequestDispatcher. * If there is an error while performing the operation redirect to the Error page |
| **doPost**  **Parameters**  HttpServletRequest request, HttpServletResponse response  **Return Type**  void | * Retrieve the Company name from request. * Call the InsertCompanyDetails() method of InsuranceCompanyBO class by passing the Company details as the input parameter. |
| AddPolicyBO | **doGet**  **Parameters**  HttpServletRequest request, HttpServletResponse response  **Return Type**  void | * Retrieve the Policies list by calling the getPolicies() method of AddPolicy class. * Set the Policy List obtained in request. * Forward the response to the appropriate page using a RequestDispatcher. * If there is an error while performing the operation redirect to the Error page |
| **doPost**  **Parameters**  HttpServletRequest request, HttpServletResponse response  **Return Type**  void | * Retrieve the PolicyDetails from request. * Call the InsertPolicy() method of AddPolicyBO class by passing the Policy details as the input parameter. |
| AddPoliciesOnEmployeesBO | **doGet**  **Parameters**  HttpServletRequest request, HttpServletResponse response  **Return Type**  void | * Retrieve the Employee list by calling the getEmployeeonPolicy() method of AddPoliciesonEmployees class. * Set the Employee List obtained in request. * Forward the response to the appropriate page using a RequestDispatcher. * If there is an error while performing the operation redirect to the Error page |
| **doPost**  **Parameters**  HttpServletRequest request, HttpServletResponse response  **Return Type**  void | * Retrieve the Company name from request. * Call the InserPolicyonEmployee() method of AddPoliciesonEmployees class by passing the Company details as the input parameter. |
| PolicyRequestDetailsBO | **doGet**  **Parameters**  HttpServletRequest request, HttpServletResponse response  **Return Type**  void | * Retrieve the Policy Request list by calling the getRequestforPolicies() method of PolicyRequestDetails class. * Set the Policy Request List obtained in request. * Forward the response to the appropriate page using a RequestDispatcher. * If there is an error while performing the operation redirect to the Error page |
| **doPost**  **Parameters**  HttpServletRequest request, HttpServletResponse response  **Return Type**  void | * Call the insertPolicyRequest() method of PolicyRequestDetails class by passing the Request details as the input parameter. |

# Database Design

## Identification of Tables



## Table Structure

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Table Name | Column Name | Column Discription | Data Type | Data Size | Primary Key | Other Constraint |
| EMPLOYEES | EMPNO | Column is used for storing employee id. | int | 11 | YES | Not Null |
| DESIGNATION | Column is used for storing employee designation | Varchar | 50 |  | Not Null |
| JOINDATE | Column is used for storing employment joining date | datetime |  |  | Not Null |
| SALARY | Column is used for storing salary of the employee | money |  |  |  |
| FIRSTNAME | Column for storing the first name. | Varchar | 50 |  | Not Null |
| LASTNAME | Column is used for storing last name | Varchar | 50 |  | Not Null |
| USERNAME | Column is used for storing Username of employee for login | Varchar | 50 |  | Not Null |
| PASSWORD | Column is used for storing password of employee | Varchar | 50 |  | Not Null |
| ADDRESS | Column is used to store address of employee | Varchar | 150 |  | Not Null |
| CONTACTNO | Column is used for storing phone number of the employee | Varchar | 50 |  | Not Null |
| STATE | Column is used for storing state of the employee | Varchar | 50 |  | Not Null |
| COUNTRY | Column is used for storing country of the employee | Varchar | 50 |  | Not Null |
| CITY | Column is used for storing city of the employee | Varchar | 50 |  | Not Null |
| POLICYSTATUS | Column is used for storing policy status of the employee | Varchar | 30 |  |  |
| POLICYID | Column is used for storing policy id of the employee | int |  |  | FOREIGN KEY (POLICIES.POLICYID ) |
| ADMINLOGIN | USERNAME | Column is used for storing Admin user name | Varchar | 50 | Yes | NOT NULL |
| PASSWORD | Column is used for storing Admin user password | Varchar | 50 |  | NOT NULL |
| COMPANYDETAILS | COMPANYID | Column is used for storing Company id | int |  | Yes | NOT NULL |
| COMPANYNAME | Column is used for storing company name | Varchar | 50 |  | NOT NULL |
| ADDRESS | Column is used for storing address of the company | Varchar | 150 |  |  |
| PHONE | Column is used for storing phone number of the company | Varchar | 20 |  |  |
| COMPANYURL | Column is used for storing URL of the company | Varchar | 50 |  |  |
| POLICIES | POLICYID | Column is used for storing policy id | Int |  | Yes | NOT NULL |
| POLICYNAME | Column is used for storing policy name | Varchar | 50 |  |  |
| POLICYDESC | Column is used for storing description of a policy | Varchar | 150 |  |  |
| AMOUNT | Column is used for storing amount of the policy | money |  |  |  |
| EMI | Column is used for storing emi amount of the policy | money |  |  |  |
| COMPANYID | Column is used for storing company id for that policy | int |  |  |  |
| MEDICALID | Column is used for storing medical id of the policy | Varchar | 50 |  |  |
| POLICIESONEMPLOYEES | EMPNO | Column is used for storing employee number | Varchar | 10 |  |  |
| POLICYID | Column is used for storing policy id | int |  |  |  |
| POLICYNAME | Column is used for storing policy name | Varchar | 50 |  |  |
| POLICYAMOUNT | Column is used for storing amount of the policy | money |  |  |  |
| POLICYDURATION | Column is used for storing duration of the policy | Decimal | 7,2 |  |  |
| EMI | Column is used for storing emi anount of the policy | Decimal | 7,2 |  |  |
| PSTARTDATE | Column is used for storing start date of the policy | Datetime |  |  |  |
| PENDDATE | Column is used for storing end date of the policy | Datetime |  |  |  |
| COMPANYID | Column is used for storing company id | Varchar | 30 |  |  |
| COMPANYNAME | Column is used for storing company name | Varchar | 50 |  |  |
| MEDICALID | Column is used for storing medical id | Varchar | 50 |  |  |
| POLICYAPPROVALDETAILS | POLICYID | Column is used for storing policy id | Int |  | Yes | NOT NULL |
| REQUESTID | Column is used for storing request id of the policy mapping by employee | int |  |  |  |
| DATE | Column is used for storing date requested by the employee | Datetime |  |  |  |
| AMOUNT | Column is used for storing amount | money |  |  |  |
| STATUS | Column is used for storing status of the request | Char | 3 |  |  |
| REASON | Column is used for storing reason for rejection | Varchar | 50 |  |  |
| POLICYREQUESTDETAILS | REQUESTID | Column is used for storing request id | Int |  | Yes | NOT NULL |
| REQUESTDATE | Column is used for storing requested date for policy mapping | Datetime |  |  |  |
| EMPNO | Column is used for storing employee number | Int |  |  | NOT NULL |
| POLICYID | Column is used for storing policy id | int |  |  | FOREIGN KEY (POLICIES.POLICYID ) |
| POLICYNAME | Column is used for storing policy name | Varchar | 50 |  |  |
| POLICYAMOUNT | Column is used for storing policy amount | Money |  |  |  |
| EMI | Column is used for storing emi amount | money |  |  |  |
| COMPANYID | Column is used for storing company id | int |  |  |  |
| COMPANYNAME | Column is used for storing company name | Varchar | 50 |  |  |
| STATUS | Column is used for storing status | Varchar | 50 |  | NOT NULL |
| POLICYTOTALDESCRIPTION | POLICYID | Column is used for storing policy id | int |  |  | FOREIGN KEY (POLICIES.POLICYID ) |
| POLICYNAME | Column is used for storing policy name | Varchar | 50 |  |  |
| POLICYDESC | Column is used for storing policy description | Varchar | 250 |  |  |
| POLICYAMOUNT | Column is used for storing policy amount | Money |  |  |  |
| EMI | Column is used for storing policy emi amount | Money |  |  |  |
| POLICYDURATIONINMONTHS | Column is used for storing duration of policy in months | Int |  |  |  |
| COMPANYNAME | Column is used for storing name of the company | Varchar | 50 |  | NOT NULL |
| MEDDICALID | Column is used for storing medical id | Varchar | 50 |  | NOT NULL |

# References

# Change Log

*Please note that this table needs to be maintained even if a Configuration Management tool is used.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version Number | Changes made | | | |
| V1.0 | *Created by Sandhya B* | | | |
| V1.1 | *<If the change details are not explicitly documented in the table below, reference should be provided here>* | | | |
| Page no | Changed by | Effective date | Changes effected |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| V1.2 | *<If the change details are not explicitly documented in the table below, reference should be provided here>* | | | |
| Page no | Changed by | Effective date | Changes effected |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |