***E-Project Semester 3***

**Student information**

Full name: Doan Duc Bao

Contact number: 0972055909

Email: baoddth2003002@fpt.edu.vn

Class: T2007E

Link github project: <https://github.com/psawn/ProjectSem3FPT>

**Acknowledgment**

To able to complete this graduation thesis, I would like to express my sincerest thanks to the teachers of FPT that trained and provided me with valuable knowledge and experiences during my study period

I also would like to thank my adviosr who had enthusiastically guided me in the process of making my graduation thesis.

Finally, I would like to thank my family and my friends who always by my side to encourage, give suggestions and create the conditions for me to complete this graduation thesis.

However, because of limited time and knowledge, this thesis inevitably has weaknesses, I really want to receive all reviews to make this project become better. I sincerely thank.

Ha Noi, March 5th, 2022

Doan Duc Bao

**Overview**

The main goal of this project is building an application to manage insurance company information data. The program is a tool to help administrators store and access data about insurances, policies and customers contact information. The program helps customers have accurate information and an overview of data which are managing by the company, therefore helping the administrator to easily collect information as well as build a plan to develop the company’s brand in the future.

The main content of the thesis includes the following parts.

**Chapter 1: Application development platform and technology**

This chapter mentions the platform as well as the technologies for developing application:

* Research about RESTful API
* Research about NodeJS platform
* Research about Express FrameWork
* Research about libraries in npm tool

**Chapter 2: System analysis and design**

This chapter focuses on survey, requirements analysis, and system design

* Introduction and survey of system requirements.
* Analyze and describe the function of system.
* Dive into database design, interface design

**Chapter 3: Summary and evaluation**

* State the results of thesis implementation, evaluate achieved and unachieved scores.
* Give the directions for system development in the future and conclude.

# APPLICATION DEVELOPMENT PLATFORM AND TECHNOLOGY

## Servey API

* API is acronym for Application Programming Interface. It allows connection and data exchange between two separate software systems. A software system can embed APIs consisting of function or sub-rountines that can be run by another software system.

## ASP.Net Core Web API

* ASP.NET Web API is a framework for building HTTP services that can be accessed from any client including browsers and mobile devices. It is an ideal platform for building RESTful applications on the .NET Framework.



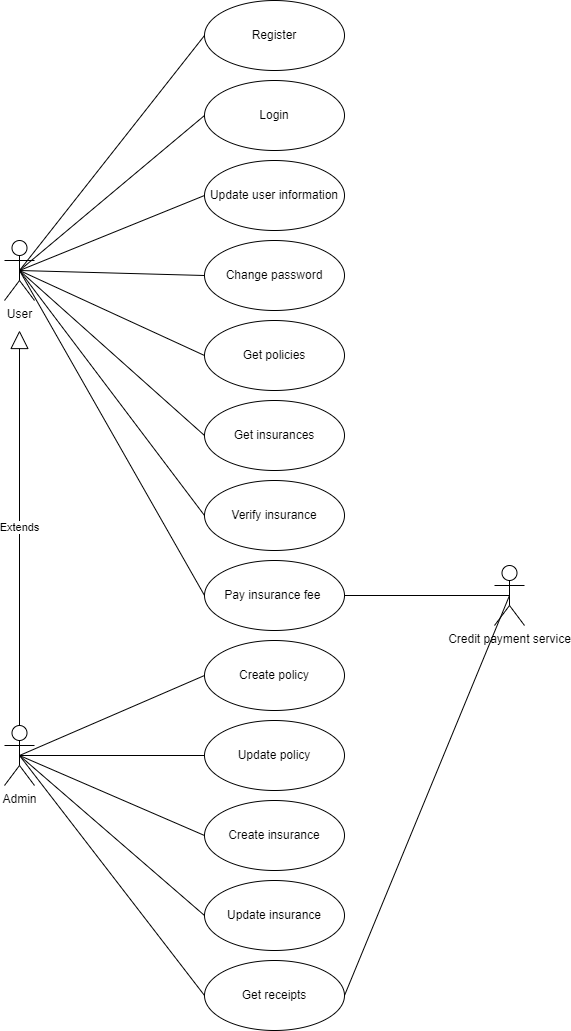
## Vue JS

* Vue is a progressive framework for building user interfaces. Unlike other monolithic frameworks, Vue is designed from the ground up to be incrementally adoptable. The core library is focused on the view layer only, and is easy to pick up and integrate with other libraries or existing projects. On the other hand, Vue is also perfectly capable of powering sophisticated Single-Page Applications when used in combination with modern tooling and supporting libraries.



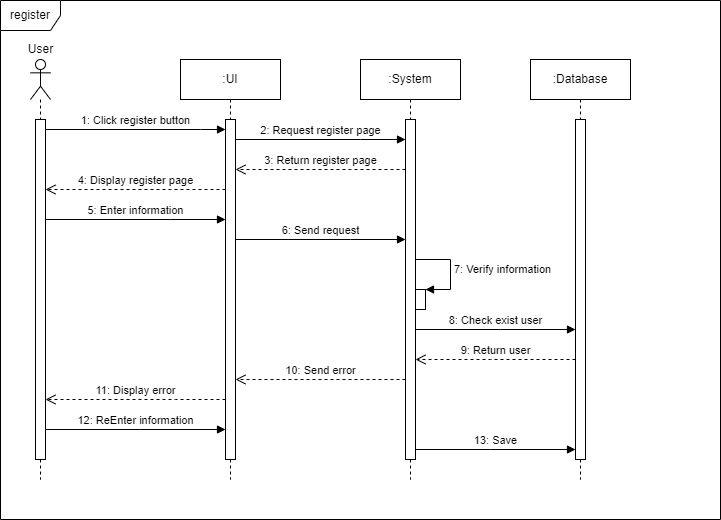
# SYSTEM ANALYSIS AND DESIGN

## Use case

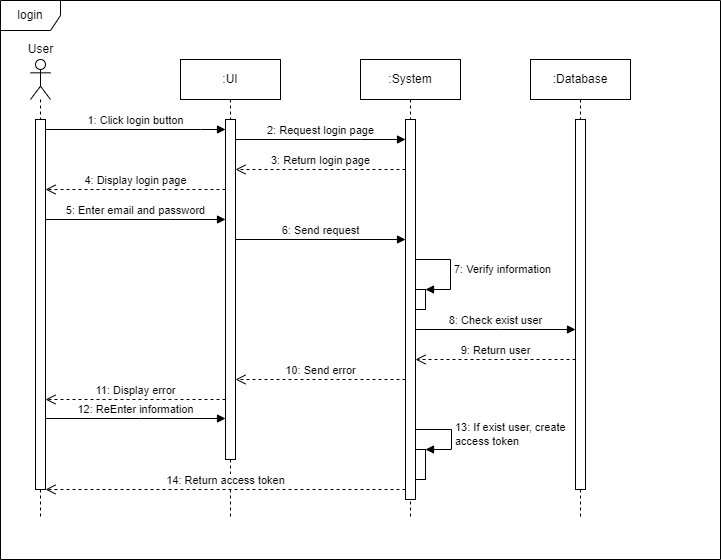


## Sequence diagram

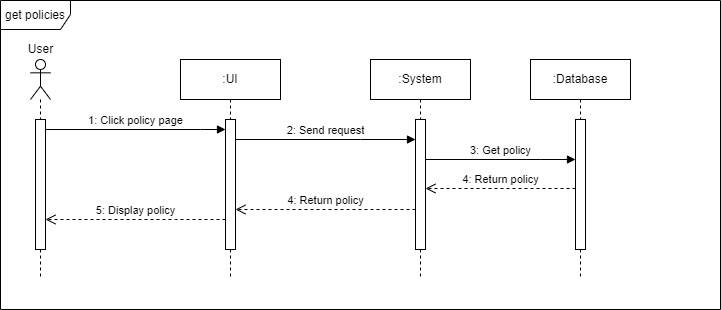
1. Register



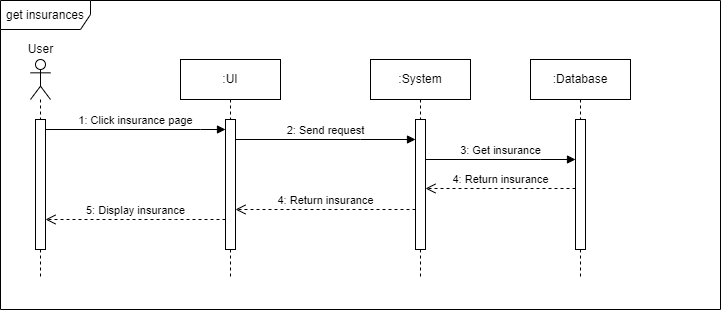
1. Login



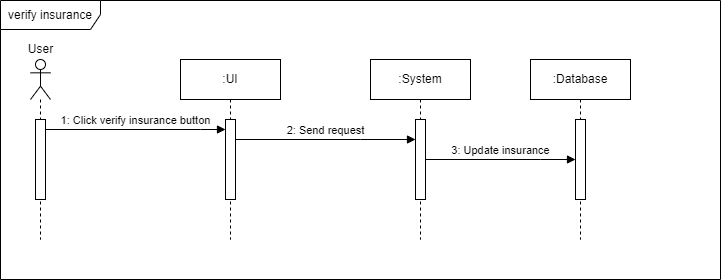
1. Get policies



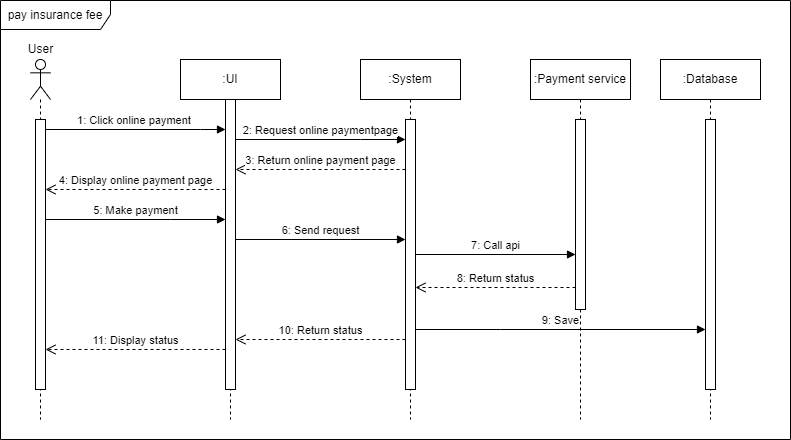
1. Get insurances



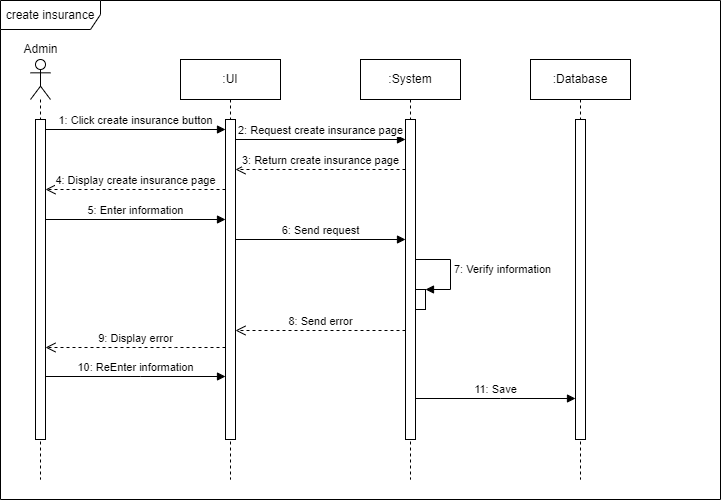
1. Verify insurances



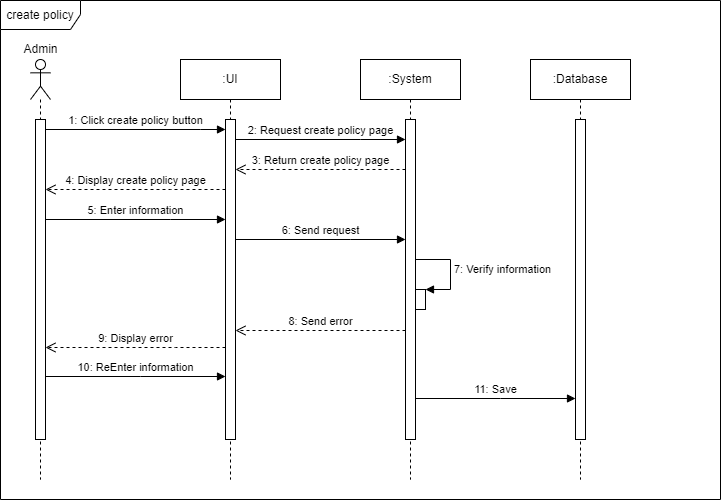
1. Pay insurance fee



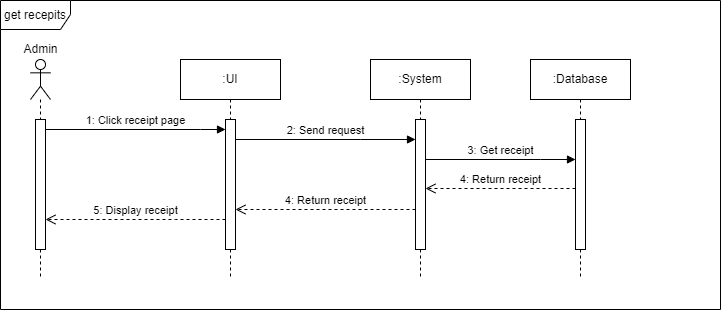
1. Create insurance



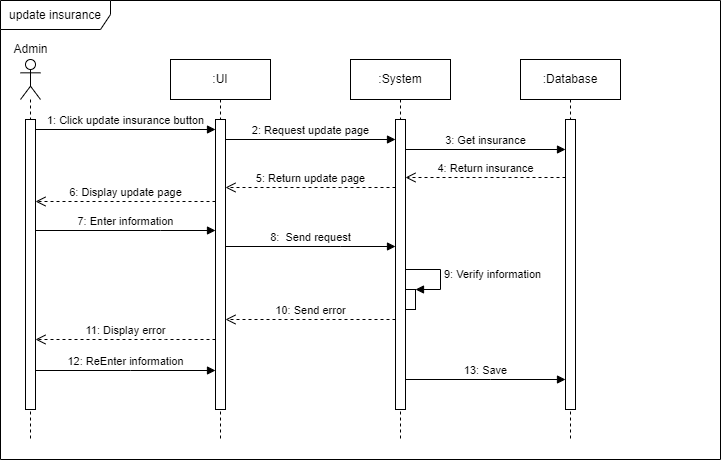
1. Create policy



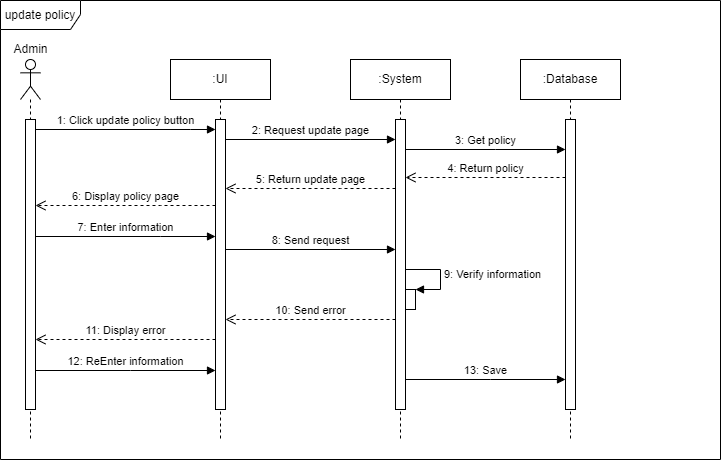
1. Get receipts



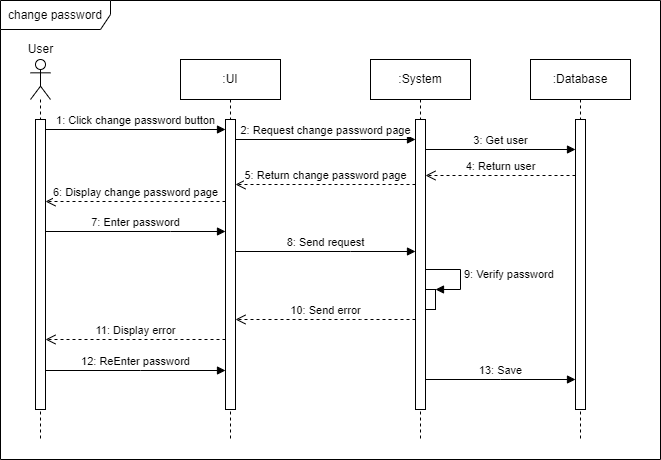
1. Update insurance



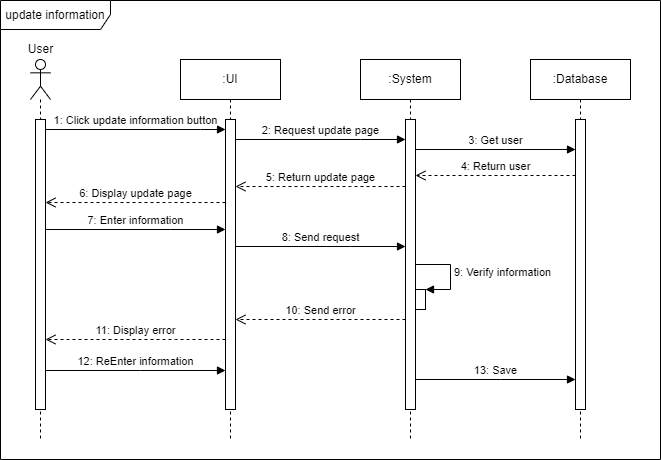
1. Update policy



1. Change password



1. Update user information



## Database design



## Database description

1. Description of Users table

Table name: Users

Description: Manage user information

|  |  |  |
| --- | --- | --- |
| **Field** | **Description** | **Properties** |
| Id | The identifier of each user | * primary key * uuid |
| Email | The email of user | * string * not null * unique |
| Password | The password of user | * string * not null |
| FirstName | The first name of user | * string |
| LastName | The last name of user | * string |
| PhoneNumber | The phone number of user | * string |
| Address | The address of user | * string |
| Role | The role of user | * string |
| CreatedDate | The time user crated | * datetime |
| ModifiedDate | The time user modified | * datetime |

1. Description of HomeInsurances table

Table name: HomeInsurances

Description: Manage home insurance information

|  |  |  |
| --- | --- | --- |
| **Field** | **Description** | **Properties** |
| Id | The identifier of each home insurance | * primary key * uuid |
| Address | The address of home | * string * not null |
| LandOwnershipId | The identifier of land ownership for home | * string * not null |
| HomeCharacteristic | The home’s characteristic | * string * not null |
| EffectiveDate | The start time of insurance | * datetime * not null |
| ExpireDate | The end time of insurance | * datetime * not null |
| CreatedDate | The time insurance created | * string |
| Role | The role of user | * string |
| CreatedDate | The time insurance crated | * datetime |
| ModifiedDate | The time insurance modified | * datetime |
| Remark | The remark of insurance | * string |
| Status | The status of insurance | * string |
| IsVerified | Insurance is verified by user or not ? | * string |
| Token | Token for insurance verify | * string |
| UserId | The identifier of user belong to insurance | * uuid * not null |
| HomePolicyId | The identifier of home policy belong to insurance | * uuid * not null |
| ReceiptId | The identifier of receipt belong to insurance | * uuid |

1. Description of LifeInsurances table

Table name: LifeInsurances

Description: Manage life insurance information

|  |  |  |
| --- | --- | --- |
| **Field** | **Description** | **Properties** |
| Id | The identifier of each life insurance | * primary key * uuid |
| FirstName | The first name of beneficiary | * string * not null |
| LastName | The last name of beneficiary | * string * not null |
| Relationship | The relationship between beneficiary and user | * string * not null |
| Image | Link of beneficiary’s image | * string * not null |
| EffectiveDate | The start time of insurance | * datetime * not null |
| ExpireDate | The end time of insurance | * datetime * not null |
| CreatedDate | The time insurance created | * datetime |
| ModifiedDate | The time insurance modified | * datetime |
| Remark | The remark of insurance | * string |
| Status | The status of insurance | * string |
| IsVerified | Insurance is verified by user or not ? | * string |
| Token | Token for insurance verify | * string |
| UserId | The identifier of user belong to insurance | * uuid * not null |
| LifePolicyId | The identifier of life policy belong to insurance | * uuid * not null |
| ReceiptId | The identifier of receipt belong to insurance | * uuid |

1. Description of VehicleInsurances table

Table name: VehicleInsurances

Description: Manage vehicle insurance information

|  |  |  |
| --- | --- | --- |
| **Field** | **Description** | **Properties** |
| Id | The identifier of each vehicle insurance | * primary key * uuid |
| PlateNumber | The plate number of vehicle | * string * not null |
| EngineNumber | The engine number of vehicle | * string * not null |
| ChasisNumber | The chasis number of vehicle | * string * not null |
| Type | The type of vehicle | * string * not null |
| EffectiveDate | The start time of insurance | * datetime * not null |
| ExpireDate | The end time of insurance | * datetime * not null |
| CreatedDate | The time insurance crated | * datetime |
| ModifiedDate | The time insurance modified | * datetime |
| Remark | The remark of insurance | * string |
| Status | The status of insurance | * string |
| IsVerified | Insurance is verified by user or not ? | * string |
| Token | Token for insurance verify | * string |
| UserId | The identifier of user belong to insurance | * uuid * not null |
| VehiclePolicyId | The identifier of vehicle policy belong to insurance | * uuid * not null |
| ReceiptId | The identifier of receipt belong to insurance | * uuid |

1. Description of MedicalInsurances table

Table name: MedicalInsurances

Description: Manage vehicle insurance information

|  |  |  |
| --- | --- | --- |
| **Field** | **Description** | **Properties** |
| Id | The identifier of each medical insurance | * primary key * uuid |
| FirstName | The first name of beneficiary | * string * not null |
| LastName | The last name of beneficiary | * string * not null |
| Relationship | The relationship between beneficiary and user | * string * not null |
| Image | Link of beneficiary’s image | * string * not null |
| EffectiveDate | The start time of insurance | * datetime * not null |
| ExpireDate | The end time of insurance | * datetime * not null |
| CreatedDate | The time insurance crated | * datetime |
| ModifiedDate | The time insurance modified | * datetime |
| Remark | The remark of insurance | * string |
| Status | The status of insurance | * string |
| IsVerified | Insurance is verified by user or not ? | * string |
| Token | Token for insurance verify | * string |
| UserId | The identifier of user belong to insurance | * uuid * not null |
| MedicalPolicyId | The identifier of medical policy belong to insurance | * uuid * not null |
| ReceiptId | The identifier of receipt belong to insurance | * uuid |
| HospitalId | The identifier of hospital providing service | * uuid * not null |

1. Description of Hospitals table

Table name: Hospitals

Description: Manage hospitals information

|  |  |  |
| --- | --- | --- |
| **Field** | **Description** | **Properties** |
| Id | The identifier of each hospital | * primary key * uuid |
| Name | The hospital’s name | * string * not null |
| Address | The hospital’s address | * string * not null |

1. Description of HomePolicies table

Table name: HomePolicies

Description: Manage home policiy information

|  |  |  |
| --- | --- | --- |
| **Field** | **Description** | **Properties** |
| Id | The identifier of each home policy | * primary key * uuid |
| Type | The type name of policy | * string * not null |
| Content | The content of policy | * string * not null |
| AmountPaid | The amount of monthly payment | * int * not null |
| CreatedBy | The identifier of user create policy | * string |
| ModifiedBy | The identifier of user modify policy | * string |
| CreatedDate | The time policy crated | * datetime |
| ModifiedDate | The time policy modified | * datetime |
| ReleasedDate | The time policy released | * datetime |
| IsReleased | The policy is released or not ? | * boolean |

1. Description of MedicalPolicies table

Table name: MedicalPolicies

Description: Manage medical policiy information

|  |  |  |
| --- | --- | --- |
| **Field** | **Description** | **Properties** |
| Id | The identifier of each medical policy | * primary key * uuid |
| Type | The type name of policy | * string * not null |
| Content | The content of policy | * string * not null |
| AmountPaid | The amount of monthly payment | * int * not null |
| CreatedBy | The identifier of user create policy | * string |
| ModifiedBy | The identifier of user modify policy | * string |
| CreatedDate | The time policy crated | * datetime |
| ModifiedDate | The time policy modified | * datetime |
| ReleasedDate | The time policy released | * datetime |
| IsReleased | The policy is released or not ? | * boolean |

1. Description of LifePolicies table

Table name: LifePolicies

Description: Manage life policiy information

|  |  |  |
| --- | --- | --- |
| **Field** | **Description** | **Properties** |
| Id | The identifier of each life policy | * primary key * uuid |
| Type | The type name of policy | * string * not null |
| Content | The content of policy | * string * not null |
| PersonClaim | The amount of money for compensation for person | * int * not null |
| AmountPaid | The amount of monthly payment | * int * not null |
| CreatedBy | The identifier of user create policy | * string |
| ModifiedBy | The identifier of user modify policy | * string |
| CreatedDate | The time policy crated | * datetime |
| ModifiedDate | The time policy modified | * datetime |
| ReleasedDate | The time policy released | * datetime |
| IsReleased | The policy is released or not ? | * boolean |

1. Description of VehiclePolicies table

Table name: VehiclePolicies

Description: Manage vehicle policiy information

|  |  |  |
| --- | --- | --- |
| **Field** | **Description** | **Properties** |
| Id | The identifier of each vehicle policy | * primary key * uuid |
| Type | The type name of policy | * string * not null |
| Content | The content of policy | * string * not null |
| PersonClaim | The amount of money for compensation for person | * int * not null |
| VehicleClaim | The amount of money for compensation for vehicle | * int * not null |
| AmountPaid | The amount of monthly payment | * int * not null |
| CreatedBy | The identifier of user create policy | * string |
| ModifiedBy | The identifier of user modify policy | * string |
| CreatedDate | The time policy crated | * datetime |
| ModifiedDate | The time policy modified | * datetime |
| ReleasedDate | The time policy released | * datetime |
| IsReleased | The policy is released or not ? | * boolean |

# SUMMARY AND EVALUATION

## Achieve result

After finish this graduation thesis, I have some the following achieve result:

* Build a website to manage policies, insurances and customer information of Seacom company step by step. This is a tool which can help admin have the most overview of company’s business activities. As a result, they will have an effective business strategy for the development in the future.
* The system has efficiency and convenience because admin can access the system from anywhere at anytime, as long as they are authorized for an account in the system to operate.

Although the achieved results above are quite positive, due to time and knowledge limitations, the system is still not really optimized in performance. Some function can be developed more streamlined, efficient and friendly. Functions may be further developed in future updates.

## Development direction

In the future, the system can have the following development direction:

* Optimize the system’s database in term of database design
* Develop some advanced reporting features, report export.
* Commercialize the system.
* Finish missing system functions due to limited time.

## Summary

After a period of research, the system is relatively complete. The actual requirements analysis has helped me to identify the right features for the system as well as choose the right technologies to build the right applications for the needs of customers who are medium and small companies with not too large amount of data and users. However, due to limitations in terms of time , knowledge and andexperience, the study inevitably contains errors and limitations. I look forward to receiving all your comments and suggestions to improve the system.