



MINISTRY OF EDUCATION AND TRAINING

ĐẠI HỌC FPT

FPT UNIVERSITY

Capstone Project Document

Insurance Card

| Group 2 | |
|------------------------------|---|
| Group members | Đinh Quang Trung – SE60994 Nguyễn Hữu Phúc – SE60749 Phùng Quang Minh Trí – SE60746 Nguyễn Chí Kha – 60351 |
| Supervisor | Kiều Trọng Khánh |
| Ext. Supervisor | N/A |
| Capstone Project Code | MIC |

- Ho Chi Minh City, 12 May 2015 -

This page is intentionally left blank



CAPSTONE PROJECT REGISTER

Class: Duration time: from 11/05/2015.... To /2015.....

(*) Profession: <Software Engineer> Specialty: <ES> <IS>

(*) Kinds of person make registers: Lecturer Students

1. Register information for supervisor (if have)

| | Full name | Phone | E-Mail | Title |
|--------------|------------------|-------|--------------------|-------|
| Supervisor 1 | Kiều Trọng Khánh | | khanhkt@fpt.edu.vn | Mr. |

2. Register information for students (if have)

| | Full name | Student code | Phone | E-mail | Role in Group |
|-----------|-----------|--------------|-------|--------|---------------|
| Student 1 | | | | | |
| Student 2 | | | | | |
| Student 3 | | | | | |
| Student 4 | | | | | |

3. Register content of Capstone Project

(*) 3.1. Capstone Project name:

English: Insurance Card.

Vietnamese: Thẻ bảo hiểm

Abbreviation:

- MIC

- Building the application combining with mobile devices provides following services

+ Simulate with Motor Insurance Card

+ Multiple Insurance Card (optional if team members have good knowledge and skills)

(*) 3.2. Main proposal content (including result and product)

a) Theory and practice (document):

- Student should apply the software development process and the UML

- . Software artifacts include User Requirement, Software Requirement Specification, Architecture Design, Detail Design, System Implementation and Testing Document, Installation Guide, sources code, and deployable software packages
- . 3 tiers should be applied
- . Server side technique:
 - o Database design, OOA, OOD, OOP, MVC, Java or .Net technology, ...
- . Client side technique
 - o HTML5, CSS, JavaScript, JQuery, Ajax, Androids ...
- . Communication technique
 - o Exchange information and transfer data in effective in networks, communicating protocol between mobile device, ...
- . Research
 - o Algorithms, NFC ...
 - o Motor Insurance and Health Insurance, ... in Vietnam and developed country

b) Program:

- . Main functions

+ Motor Insurance Card

- Guest make contract of Motor Insurance with some constraints conditions depending his/her personal information
- The insured one can renewals the Motor Insurance with some constraints conditions
- The insured one can check the card information
- The traffic polices or Police Department can check specified Motor Insurance Card expired or not. They can update the punishment of violator
- ...

+ Multiple Insurance Card (optional)

- Apply to the Health Insurance, Life Insurance, ...

c) Other products:

- . All of management functions of the system must be implemented to support the operating system in best

4. Other comment (propose all relative thing if have)

N/A

HCM city, date 14/4/2015

Supervisor (If have)
(Sign and full name)

On behalf of Registers
(Sign and full name)

Table of Contents

| | |
|---|----|
| Table of Contents | 5 |
| List of Tables | 6 |
| List of Figures | 6 |
| Definitions, Acronyms, and Abbreviations | 7 |
| A. Introduction | 7 |
| 1. Project Information..... | 7 |
| 2. Current Situation..... | 7 |
| 3. Problem Definition | 7 |
| 4. Proposed Solution..... | 8 |
| 4.1. Feature functions | 8 |
| B. Software Project Management Plan..... | 9 |
| 1. Project organization..... | 9 |
| 1.1. Software Process Model | 9 |
| C. Software Requirement Specification..... | 9 |
| 1. User Requirement Specification | 9 |
| 1.1. Customer requirement..... | 9 |
| 1.2. Staff requirement | 10 |
| 1.3. Police requirement..... | 10 |
| 1.4. Admin requirement..... | 10 |
| 2. System Requirement Specification..... | 10 |
| 2.1. System Overview Use Case..... | 10 |
| 3. Conceptual Diagram..... | 13 |
| 4. Algorithms..... | 15 |
| 4.1. Contract State..... | 15 |
| 4.2. Notification..... | 16 |
| 4.3. Concurrency Control..... | 17 |
| D. Software Design Description | 19 |
| 1. System Architecture Design | 19 |
| 1.1. Web application architecture design..... | 19 |
| 1.2. Mobile Application architecture design | 19 |
| 2. Component Diagram..... | 20 |
| 3. Detailed Description | 21 |
| 1.1. Class Diagram..... | 21 |

| | |
|---------------------------------------|----|
| 1.2. Interactive Diagram..... | 23 |
| 1. Communication diagram | 26 |
| 2. Database Design | 27 |
| 2.1. Entity relationship diagram..... | 27 |
| 2.2. Physical Diagram..... | 28 |

List of Tables

| | |
|---|----|
| Table 1: Definitions, Acronyms, and Abbreviations | 7 |
| Table 2 Contract State Dictionary..... | 15 |
| Table 3 Contract State Flow | 16 |
| Table 4 Notification use cases | 17 |
| Table 5 Component Dictionary | 20 |
| Table 6 Class dictionary..... | 23 |

List of Figures

| | |
|---|----|
| Figure 1: Waterfall model | 9 |
| Figure 2: Overview Use Case..... | 11 |
| Figure 3 Conceptual diagram | 13 |
| Figure 4 Contract State Chart..... | 15 |
| Figure 5 Concurrency control flow | 18 |
| Figure 6 System architecture design..... | 19 |
| Figure 7 Component Diagram | 20 |
| Figure 8 Class Diagram..... | 21 |
| Figure 9 Sequence diagram - <Staff> Create new contract | 23 |
| Figure 10 Sequence diagram - <Staff> Cancel contract..... | 23 |
| Figure 11 Sequence diagram - <Customer> Renew contract | 24 |
| Figure 12 Sequence diagram - <Guest> Register new contract..... | 24 |
| Figure 13 <Police> Verify card validation..... | 25 |
| Figure 14 <Police> Add punishment information..... | 25 |
| Figure 15 Web Application Communication Diagram..... | 26 |
| Figure 16 Mobile Applications Communication Diagram | 26 |
| Figure 17 Entity relationship diagram..... | 27 |
| Figure 18 Physical diagram..... | 29 |

Definitions, Acronyms, and Abbreviations

| Name | Definition |
|------------|--------------------------|
| MIC | Motor Insurance Card |
| NFC | Near field communication |

Table 1: Definitions, Acronyms, and Abbreviations

A. Introduction

1. Project Information

- Project name: **Insurance Card**
- Project Code: **MIC**
- Product Type: **Website & Android Application**
- Start Date: **May 11th, 2015**
- End Date: **August 24th, 2015**

2. Current Situation

When participating in traffic, vehicle owners are required to have compulsory insurance (according to Article 6, Decree on compulsory insurance for civil liability of motor vehicle owners, Decree No. 103/2008/NĐ-CP by Vietnam Government). Therefore, vehicle owners buy insurance from insurance companies or its agents. They pay insurance premium by cash or in online website and receive an insurance certificate with a term of one year, the term can be shorter in some specific situation. When their insurance out of date, they must buy a new insurance, old certificate will be useless. Traffic police will read insurance certificate to check traffic participants.

3. Problem Definition

Below are disadvantages of current situation:

- **Forget insurance's expired date:** Vehicle owners usually keeps their insurance certificate in wallet or somewhere on their vehicle. However, except in cases of necessity, people are not often check their insurance so they could forget its expired date. An expired insurance is not good while it be revealed by traffic officers and could get worse in case of traffic accident.
- **Hard for traffic officers to check and verify insurance:** Traffic officers must read insurance certificate to check and verify vehicle owner's information. It can be difficult and hinder their work in some cases as at dark or handwriting illegible on insurance certificate.
- **No mechanism to renew old contract:** customers have to handy register new contract when the old one is expired, this is inconvenient for customers.
- **Insurance certificate made of paper:** It could be torn, wet, smudged and especially is counterfeited.

- **Claim/compensation process is ineffective** between customer and insurance company.
- **Difficult to track and manage number of traffic violations and collisions:** In current scenario, insurance companies almost impossible knows vehicle owner's history to adjust their insurance policy.

According to Vietnam's laws, motor vehicle owners must have insurance contract with fixed term and fixed fee for each type of vehicle.

4. Proposed Solution

Our proposed solution is to build an insurance NFC card system named "MIC system" to resolve the current situations and compatible with current laws, we also design the system to be scalable so we can deploy this system to a multiple insurance services company in future plan.

MIC system includes a web application and two mobile applications with following functions:

4.1. Feature functions

- Web application:
 - **Register insurance:** user can register a new insurance card with on website using online payment. A staff will contact the user to create contract and sends an insurance NFC card to him/her. If users already have a NFC card, they can use the website to renew current contract.
 - **Check card information:** user can login into the website and check for their card's information.
 - **Request compensation:** user can fill data into the sample fields and sends compensation request to the company.
 - **Make/manage contracts:** staff can make and manage contracts.
 - **Resolve compensation:** staff can receive and resolve compensation requests.
 - **Notify contract state:** system will send an email to notify the insured one when their insurance is expired.
 - **Notify compensation state:** system will send an email to info the insured one when their compensation were accepted or rejected.
- Insurance card printer (mobile app):
 - **Simulating NFC card printer:** staff can print NFC card.
- Insurance card checker (mobile app):
 - **Check card:** traffic police and Police Department can check specified motor insurance card expired or not.
 - **Update the punishment of violator:** traffic police and Police Department can update the punishment of violator to the card information.

B. Software Project Management Plan

1. Project organization

1.1. Software Process Model

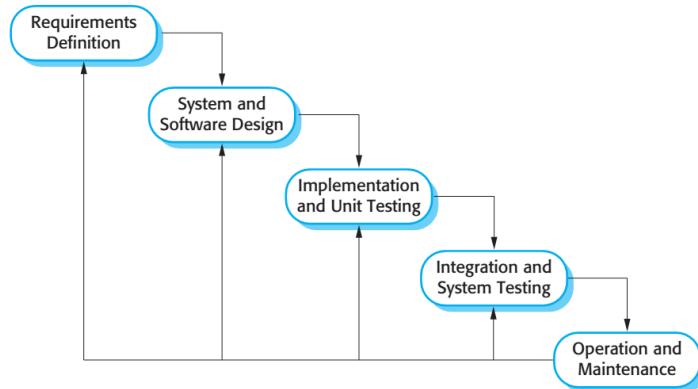


Figure 1: Waterfall model

Reference: Page 30, chapter 2, Software process model, SOFTWARE ENGINEERING 9th Edition, by Ian Sommerville.

We customize the waterfall model from the reference to make the process more capable with current situation of our team.

C. Software Requirement Specification

1. User Requirement Specification

1.1. Customer requirement

Customer is user who uses service of system. The customer can use some following functions:

- View history information include:
 - o View payment history
 - o View accident history
 - o View punishment history
- View compensation history
- View contract information
- View personal information
- Create new contract
- New card request
- Renew contract
- Cancel contract
- Request compensation

1.2. Staff requirement

Staff is people who works directly with system to track the information of customer or manages customer, staff can handle directly some problems if it happens from customer. Staff can use some following functions:

- View profile: they can change password of customer
- View customer information
- View card information
- Resolve new card request
- Resolve compensation requests
- Update contract type information
- Manage customer includes:
 - o View customer information
 - o Edit customer information
 - o Create new customer
- Manage contracts includes:
 - o Create new contract
 - o Update contract information
 - o Renew contract
 - o Cancel contract
 - o Update compensation history
 - o Update punishment history
 - o Update accident history
- Print NFC card for customer

1.3. Police requirement

Police is people who is interactive with system for checking information about customer's NFC card and handling in case the customer violates the traffic rule or make accident. Police can use some following functions:

- Verify card validation
- Get contract information
- Update punishment information

1.4. Admin requirement

Admin is people who manages staff. Administrator can use some following functions:

- Manage staff includes:
 - o Remove staff
 - o Add staff

2. System Requirement Specification

2.1. System Overview Use Case

2.1.1. Web Application

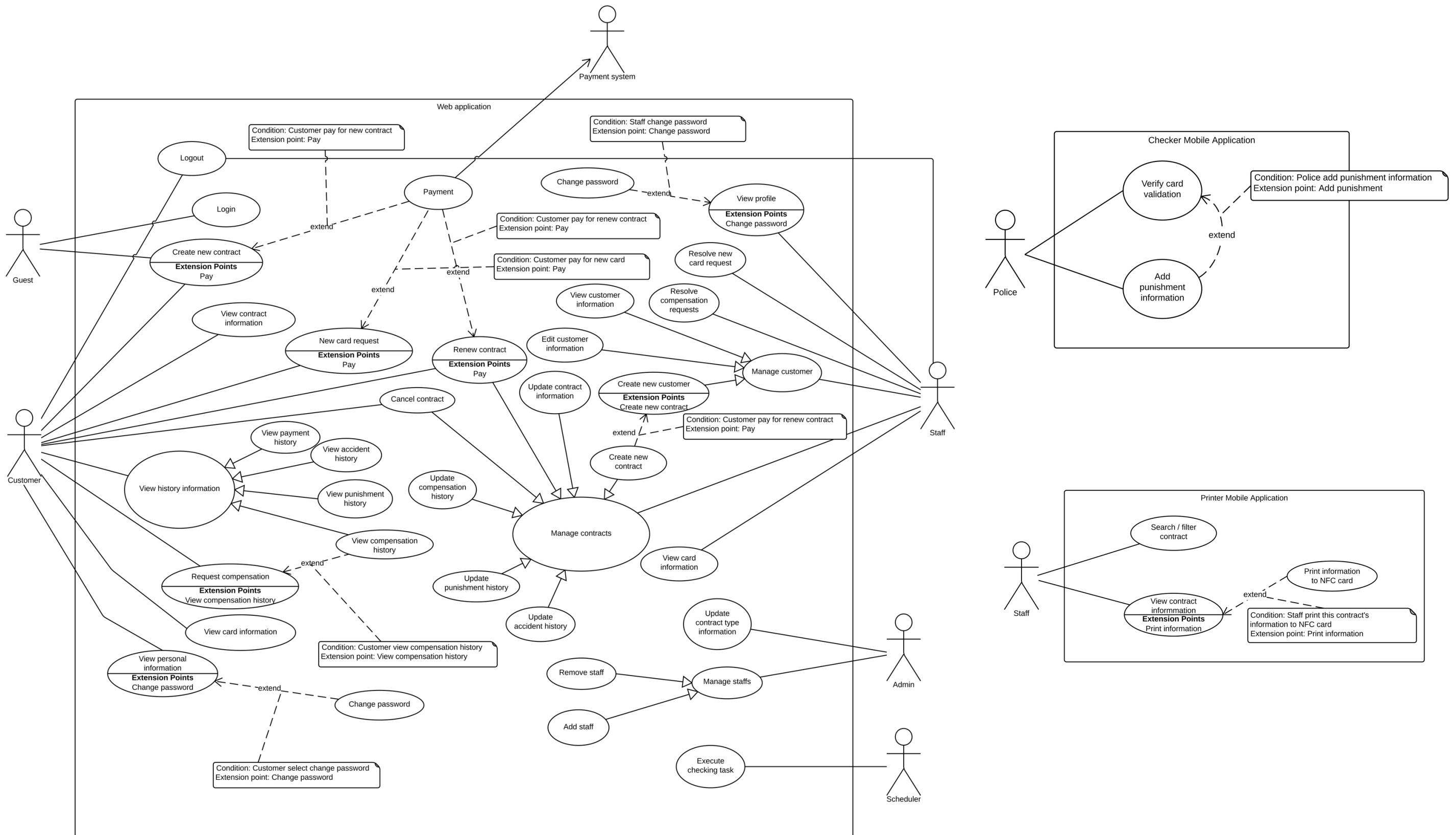


Figure 2: Overview Use Case

3. Conceptual Diagram

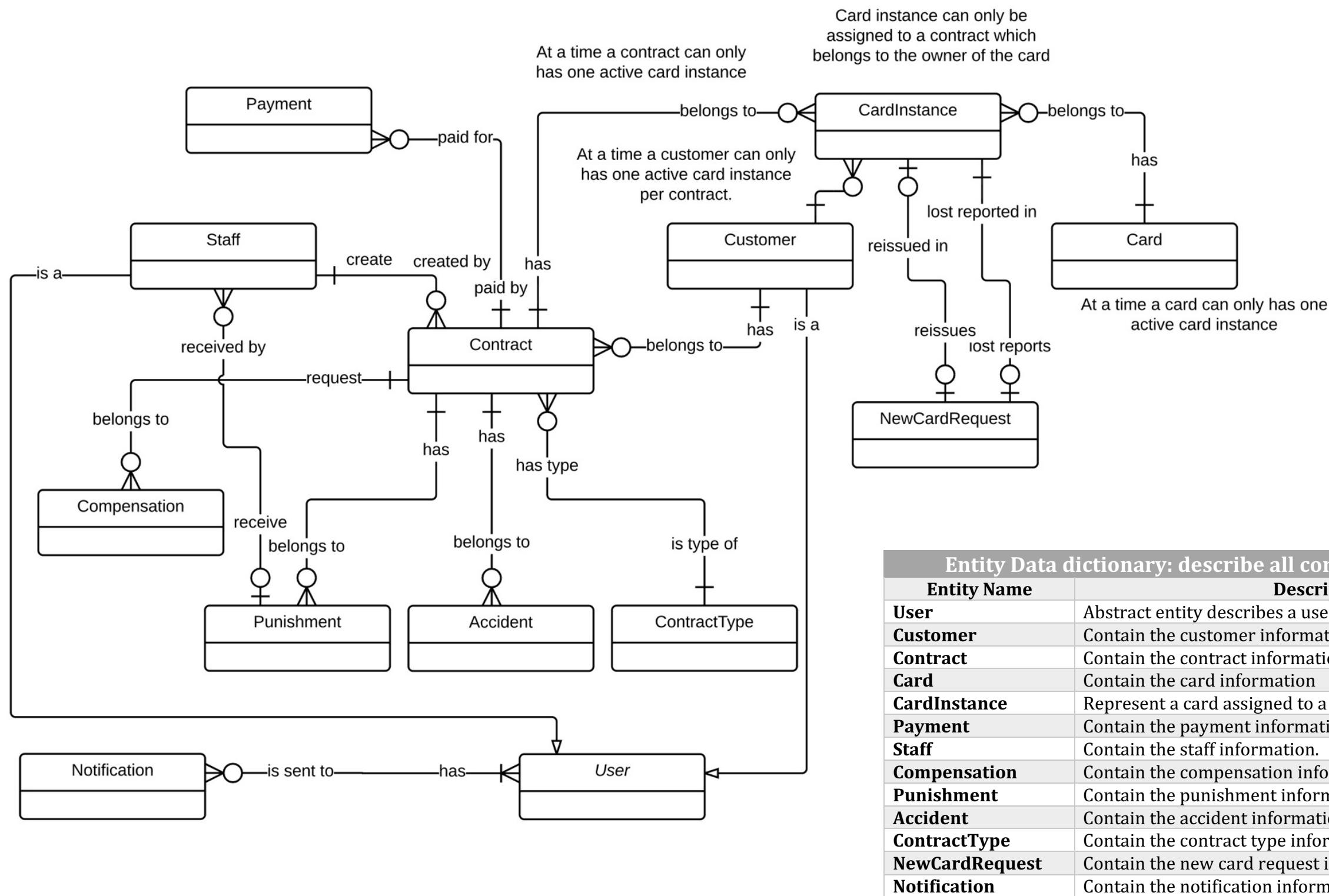


Figure 3 Conceptual diagram

This page is intentionally left blank

4. Algorithms

4.1. Contract State

The contracts in MIC system is complex and can be managed differently during the operation. The state chart bellow describes all the state of a contract.

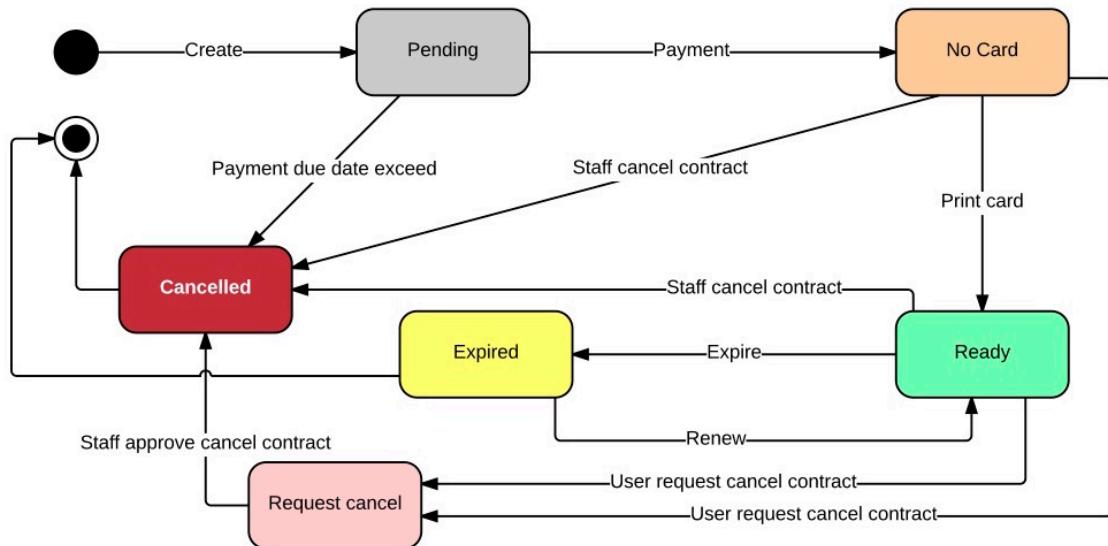


Figure 4 Contract State Chart

| State | Description |
|----------------|--|
| Pending | The contract is created and do not have payment or the start date is not come yet |
| No card | The contract had have payment and contract start date is arrived but have not assigned to a card |
| Ready | The contract is assigned with a card and ready to use |
| Expired | The contract due date is exceed and no longer valuable |
| Request cancel | The contract is requested to cancel by customer |
| Cancelled | The contract is cancelled and no longer valuable |

Table 2 Contract State Dictionary

| Stimulus | Description | State before | State after |
|----------|--|--------------|-------------|
| Create | Customer create new contract | N/A | Pending |
| Payment | Customer pay for the contract via PayPal or direct payment and contract start date is arrived. | Pending | No card |

| | | | |
|--------------------------------------|---|-------------------|----------------|
| Payment due date exceed | When the payment due date is exceed | Pending | Cancelled |
| Print card | Staff use mobile app to print the card for customer, the card ID is saved to the system | No card | Ready |
| Expire | When the contract due date is exceed, system will change the contract status | Ready | Expired |
| Renew | Customer renew contract via web application or direct payment | Expired | Ready |
| Staff cancel contract | Staff cancel contract via web application | No card, Ready | Cancelled |
| Customer cancel contract | Customer cancel contract via web application | No card, Ready | Request cancel |
| Staff approve cancel contract | Staff approve cancel request from customer via web application | Request cancel | Cancelled |

Table 3 Contract State Flow

4.2. Notification

4.2.1. Definition

Notification is a feature of MIC Web application to notify stakeholders when important events occurs.

4.2.2. Notification Methods

- Send email to receiver
- Show notification icon on web interface

4.2.3. Notification use cases

| Type | Trigger | Receiver | Notify method |
|------|--|-----------|---------------|
| 1 | Customer creates new contract | All staff | Web |
| 2 | Customer sends compensation request | All staff | Web |
| 3 | Customer sends new card request | All staff | Web |
| 4.1 | Contract is nearly expired (send first) | Customer | Web, Email |
| 4.2 | Contract is nearly expired (send 2nd time) | Customer | Web, Email |
| 4.3 | Contract is nearly expired (send 3rd time) | Customer | Web, Email |
| 5 | Contract is expired | Customer | Web, Email |

| | | | |
|----------|---|-----------|------------|
| 6 | Customer sends request cancel | All staff | Web |
| 7 | Contract is cancelled because payment due date exceed | Customer | Web, Email |
| 8 | Paid Pending Contract start date come | Customer | Web |
| 9 | Compensation resolved | Customer | Web, Email |

Table 4 Notification use cases

- For staffs, every notification has "status" to show that if the notification not resolved or when the notification is resolved and by whom. Except Type 1 notification has no status.
- All the notification can be handedly turned off by clicking Mark as read button
- When a notification is sent to all the staff, each staff has their own "Read" status. That's mean if staff A mark the status as read but staff B still see it as not read.

4.3. Concurrency Control

4.3.1. Definition

Concurrency control is a method to handle data which can be accessed and modified from many sources at the same time to prevent conflict and data loss.

4.3.2. Define problem

In web application, one contract might be edited by stakeholders at the same time, this situation may cause an inconsistent state of the contract. We have to find a solution for this problem.

4.3.3. Solution

We defined a list of entity that might be modified by many sources at a same time:

- **Contract:** can be edited/renew by stakeholders at a same time. This includes the information assigned with the contract: Payment, Compensation, Accident and Punishment information.

We use timestamp concurrency control method to solve the concurrency problem. Contract information have a "last modified" value, this value represents the last time the contract is modified. When update the contract information, a validation will run to check if the last modified is changed or not. If it is changed there will be error message to users.

4.3.4. Flow chart

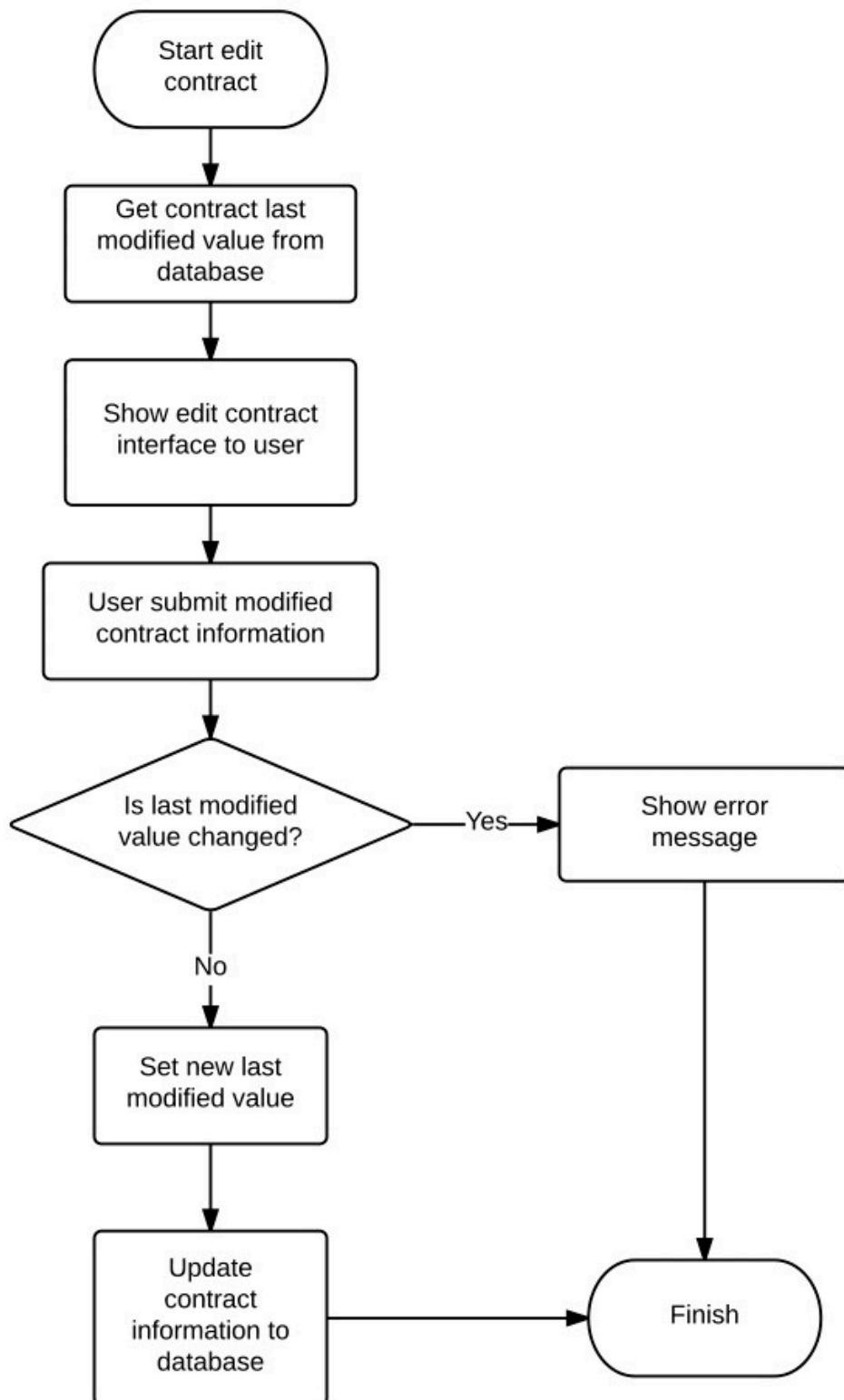


Figure 5 Concurrency control flow

D. Software Design Description

1. System Architecture Design

1.1. Web application architecture design

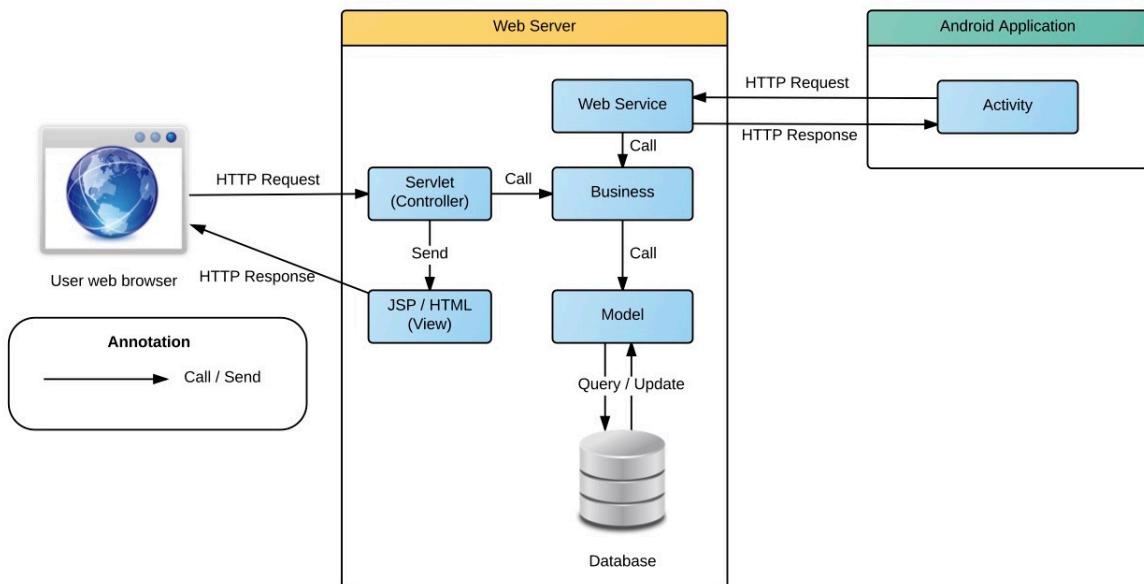
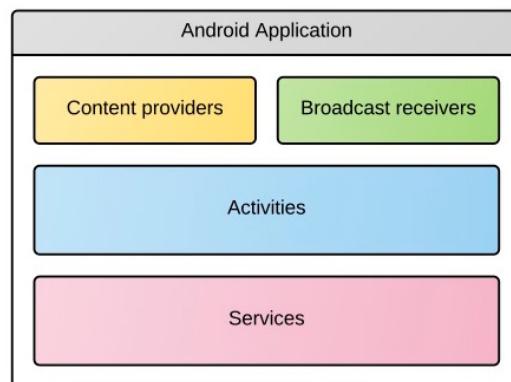


Figure 6 System architecture design

This diagram is referenced and modified from an original concept from: Chapter 6 Architecture Design, SOFTWARE ENGINEERING 9th Edition, by Ian Sommerville.

1.2. Mobile Application architecture design

The application is developed as an Android native application. In general, the application architecture conforms to Android architecture.



Reference: [Android Developer Guide - Application Fundamentals](http://developer.android.com/guide/components/fundamentals.html)

<http://developer.android.com/guide/components/fundamentals.html>

2. Component Diagram

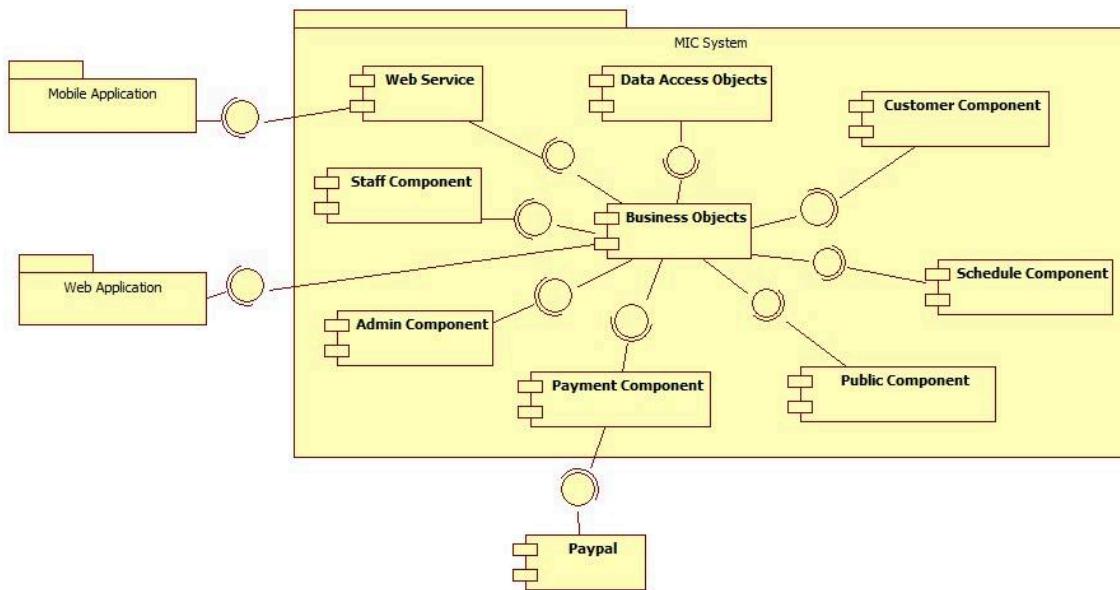


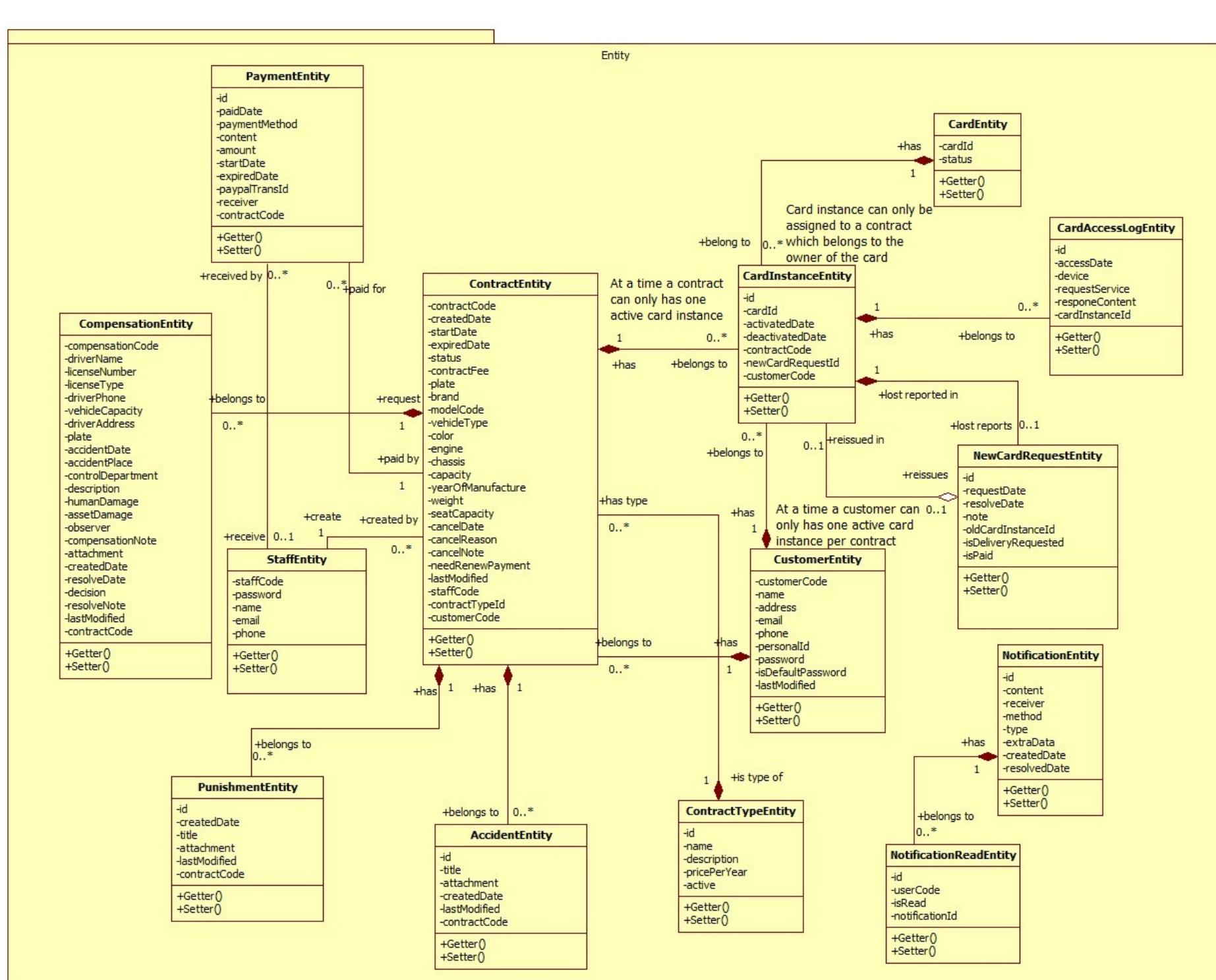
Figure 7 Component Diagram

| Component Dictionary: Describes components | |
|--|---|
| Web Application | Web application package: View, Controller |
| Mobile Application | Mobile application package |
| PayPal | Handle payment process with PayPal API |
| Payment Component | Component to handle payment process |
| Web Service | Provide API for mobile applications to interact with the system. |
| Staff Component | Component to handle staff activities in the system |
| Customer Component | Component to handle customer activities in the system |
| Public Component | Component to handle guest activities in the system |
| Admin Component | Component to handle admin activities in the system |
| Schedule Component | Component to handle scheduler in the system |
| Business Objects | Common objects to handle domain business operations for each components |
| Data Access Objects | Component to handle interaction between the system and database |

Table 5 Component Dictionary

3. Detailed Description

1.1. Class Diagram



| Class dictionary: describe Class | | |
|----------------------------------|--|--|
| Class Name | Mapping column with Conceptual diagram | Description |
| PaymentEntity | Payment | Contain the payment information. |
| CardEntity | Card | Contain the card information. |
| CardInstanceEntity | CardInstance | Contain the card instance information |
| CustomerEntity | Customer | Contain the customer information. |
| ContractEntity | Contract | Contain the contract information. |
| StaffEntity | Staff | Contain the staff information. |
| CompensationEntity | Compensation | Contain the compensation information. |
| PunishmentEntity | Punishment | Contain the punishment information. |
| AccidentEntity | Accident | Contain the accident information. |
| ContractTypeEntity | ContractType | Contain the contract type information. |
| NewCardRequestEntity | NewCardRequest | Contain the new card request information. |
| CardAccessLogEntity | N/A | Not exist in conceptual diagram. But needed in class diagram to contain the card access log information. |
| NotificationEntity | N/A | Not exist in conceptual diagram. But needed in class diagram to contain the notification information. |
| NotificationReadEntity | N/A | Not exist in conceptual diagram. But needed in class diagram to know what notifications is read. |

Figure 8 Class Diagram

This page is intentionally left blank

Table 6 Class dictionary

1.2. Interactive Diagram

1.2.1. Create new contract

Summary: this diagram show process of staff creates new contract

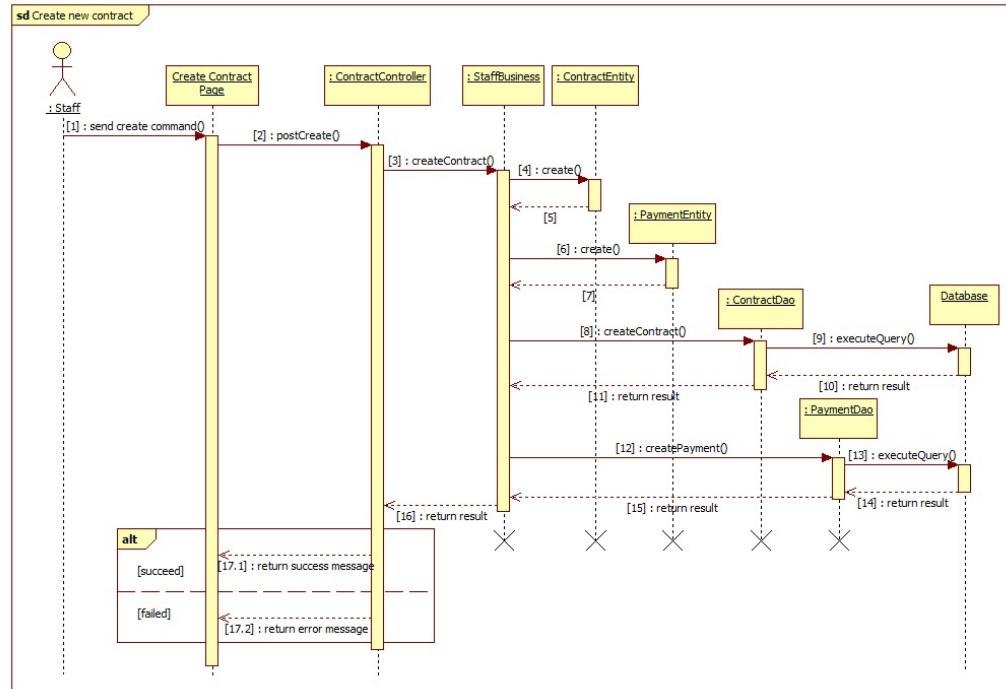


Figure 9 Sequence diagram - <Staff> Create new contract

1.2.2. Cancel contract

Summary: this diagram show process of staff cancels contract

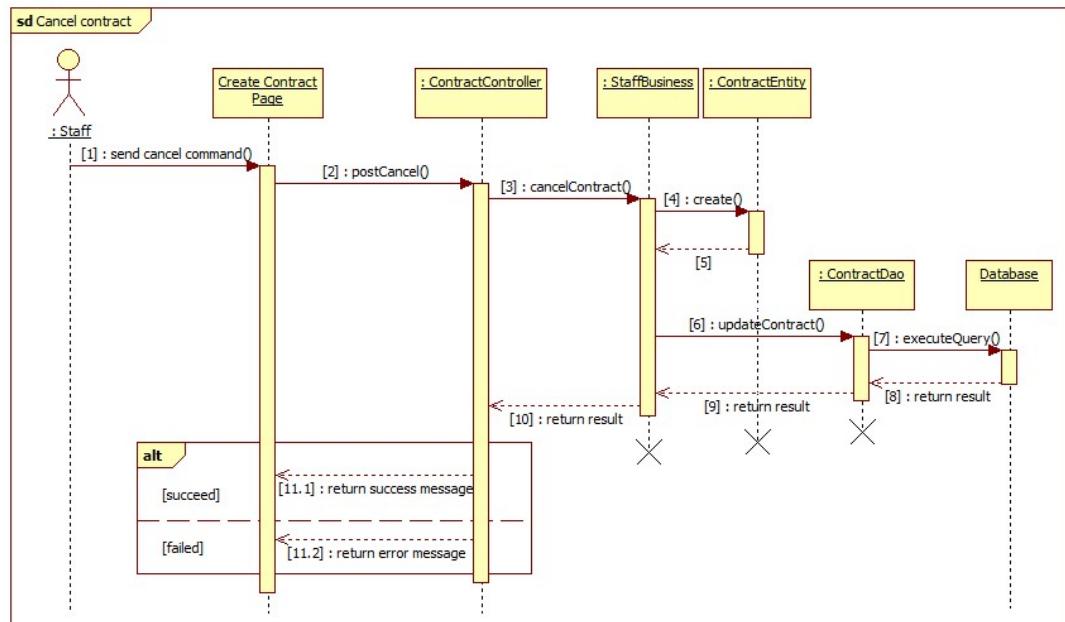


Figure 10 Sequence diagram - <Staff> Cancel contract

1.2.3. Renew contract

Summary: this diagram show process of customer renews contract

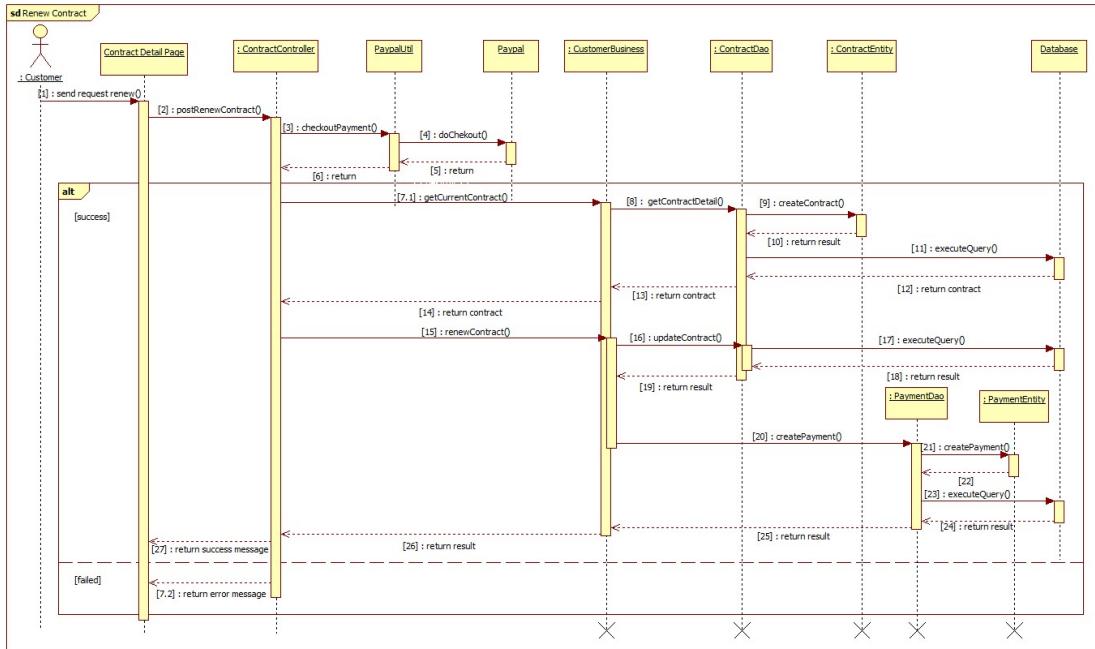


Figure 11 Sequence diagram - <Customer> Renew contract

1.2.4. Register new contract

Summary: this diagram show process of guest registers new contract

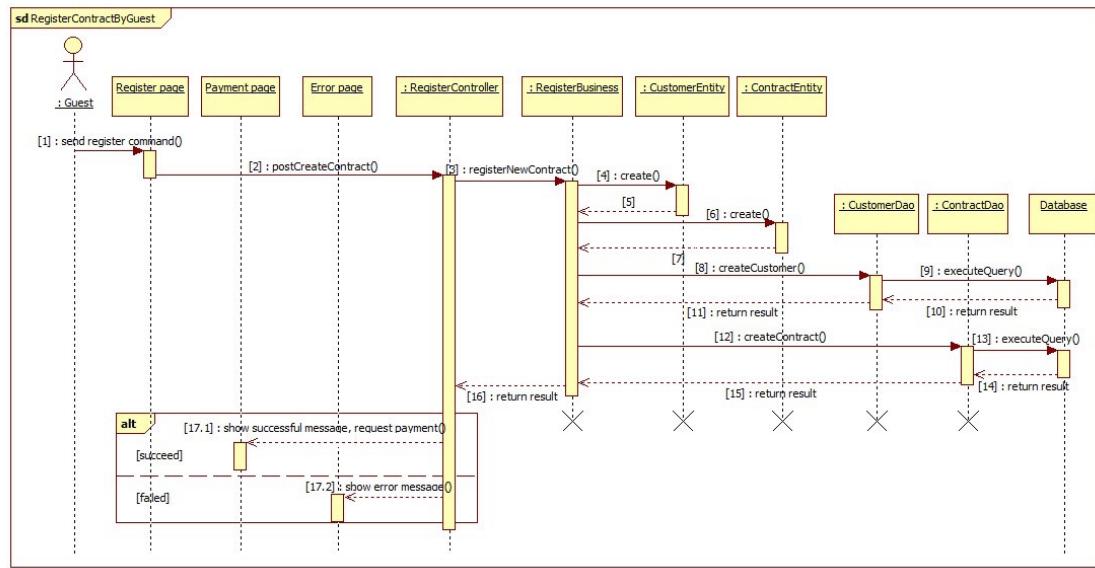


Figure 12 Sequence diagram - <Guest> Register new contract

1.2.5. <Police> Verify card validation

Summary: this diagram show process of police checks card validation.

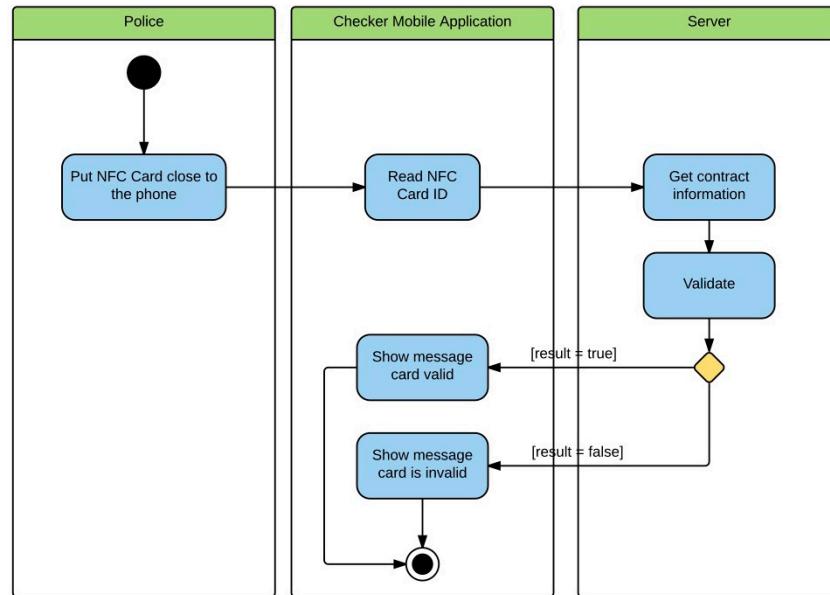


Figure 13 <Police> Verify card validation

1.2.6. <Police> Add punishment information

Summary: this diagram show process of police adds punishment information.

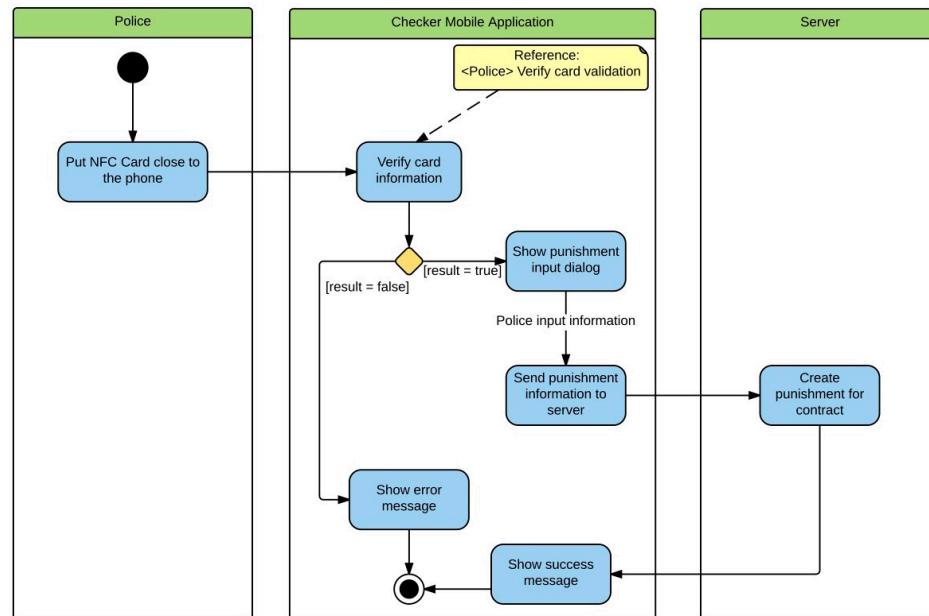


Figure 14 <Police> Add punishment information

Reference to full document for complete list of Interactive Diagram

1. Communication diagram

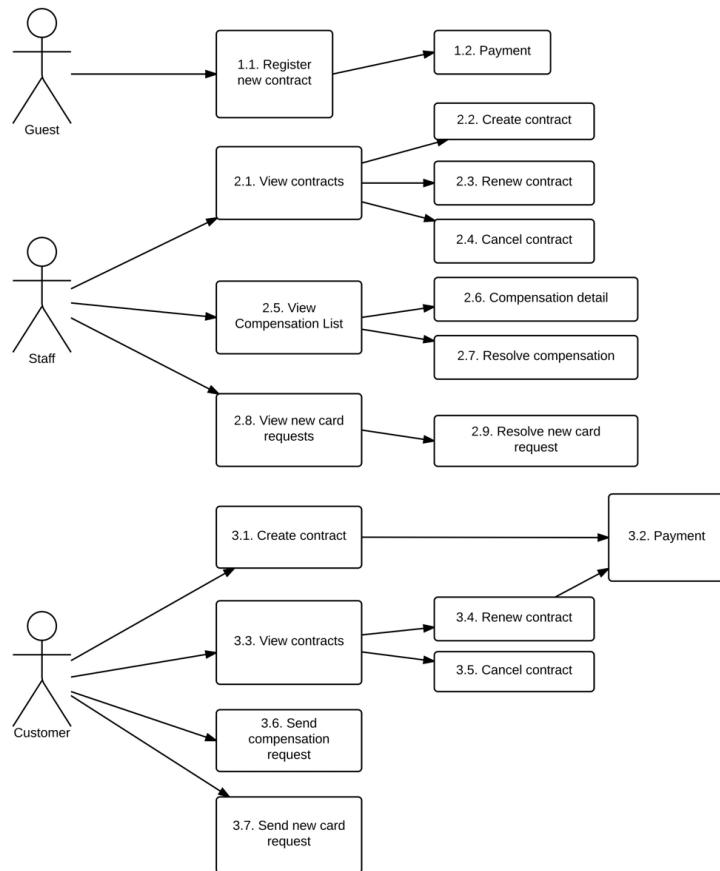


Figure 15 Web Application Communication Diagram

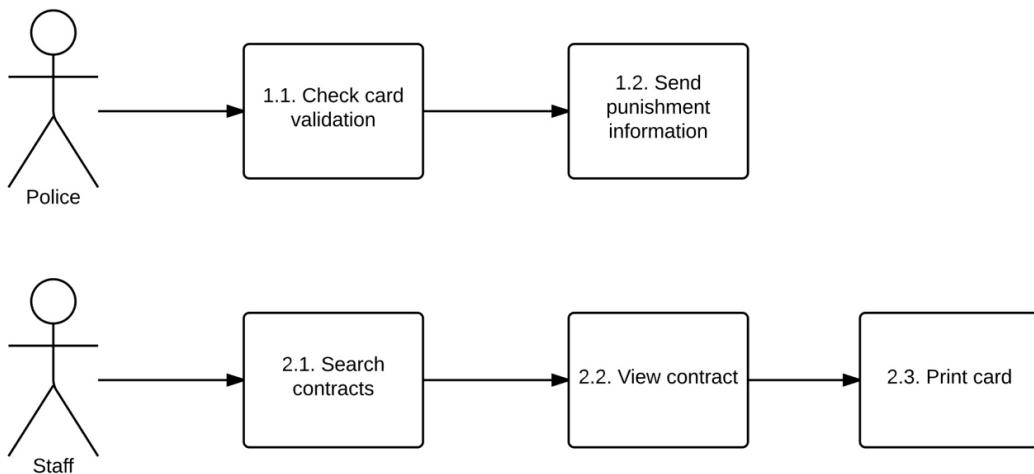
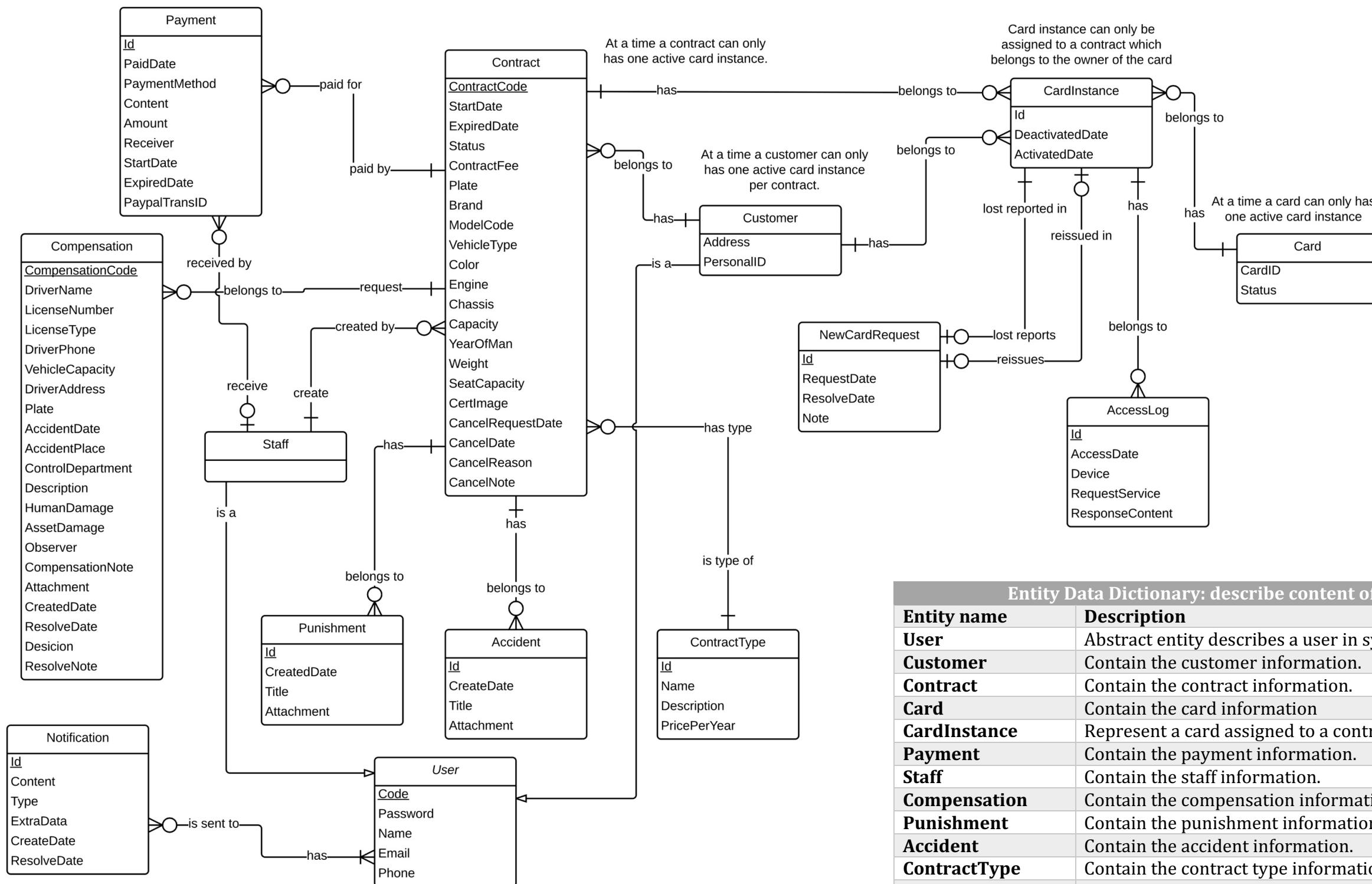


Figure 16 Mobile Applications Communication Diagram

2. Database Design

2.1. Entity relationship diagram



Entity Data Dictionary: describe content of all entities

| Entity name | Description |
|-----------------------|--|
| User | Abstract entity describes a user in system |
| Customer | Contain the customer information. |
| Contract | Contain the contract information. |
| Card | Contain the card information |
| CardInstance | Represent a card assigned to a contract |
| Payment | Contain the payment information. |
| Staff | Contain the staff information. |
| Compensation | Contain the compensation information. |
| Punishment | Contain the punishment information. |
| Accident | Contain the accident information. |
| ContractType | Contain the contract type information. |
| NewCardRequest | Contain the new card request information. |
| CardAccessLog | Contain the new card access log. |
| Notification | Contain the notification information |

Figure 17 Entity relationship diagram

This page is intentionally left blank

2.2. Physical Diagram

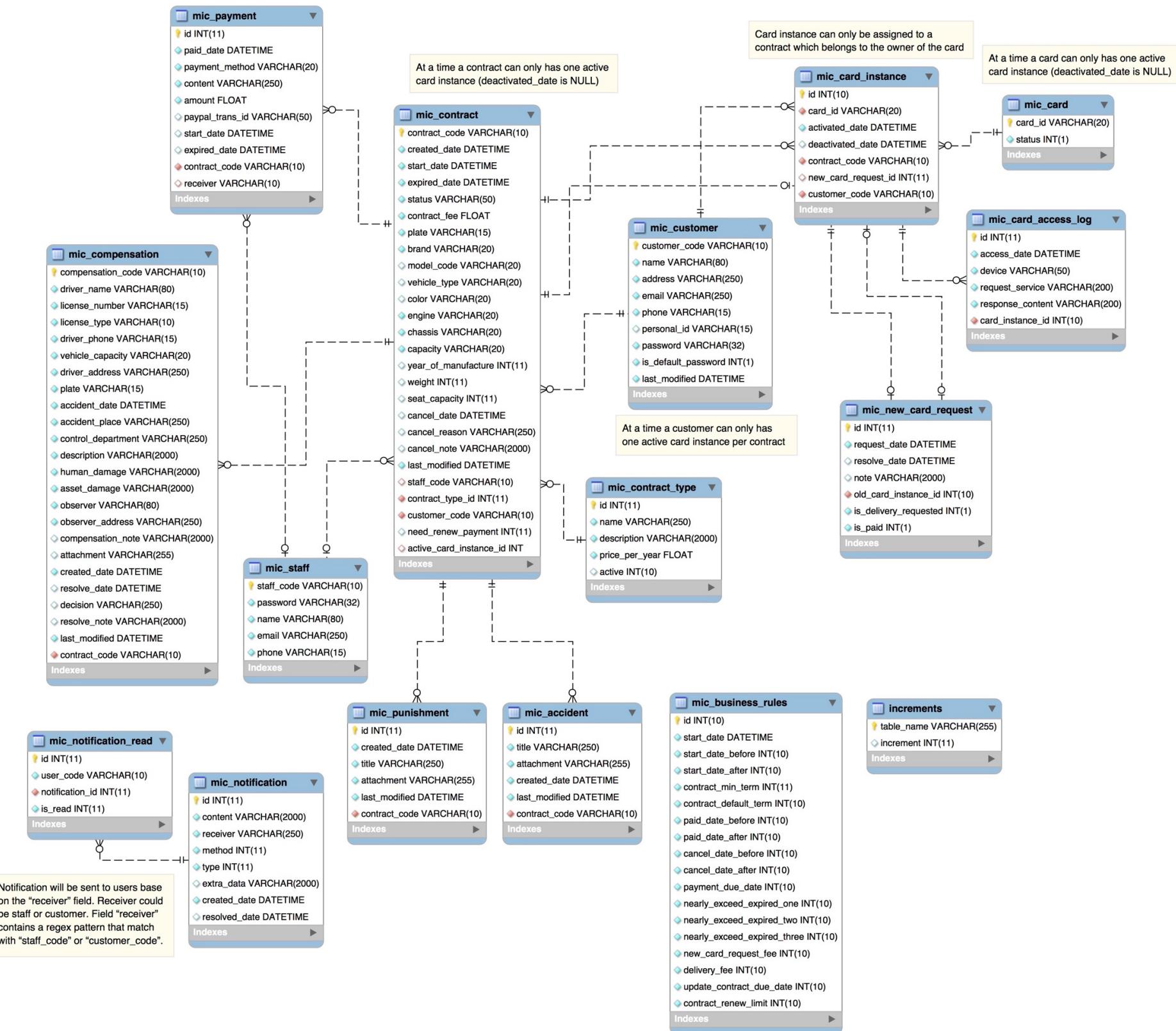


Figure 18 Physical diagram

This page is intentionally left blank