

Hobby Store Online Platform

Software Requirements Specifications

Version 1.0.0

10/21/2018

Harry Archer
Saif Mustaf
Nolan Honey
Jan Farun
Nickolas Di Domenico

Prepared for
Capstone Project 1
Instructor: Anjana Shah.
Fall 2018

Revision History

Date	Description	Author	Comments
25-10-18	First completed version	Nolan Honey	First draft
02-11-18	Added Nickolas to the group	Nolan Honey	Added new member

Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date
Nolan Honey	Nolan Honey	Team Lead/Stakeholder	05-11-2018

Table of Contents

REVISION HISTORY	1
DOCUMENT APPROVAL	1
1. INTRODUCTION	4
1.1 PURPOSE	4
1.2 SCOPE	4
1.3 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS	4
1.4 REFERENCES	5
1.5 OVERVIEW	5
2. GENERAL DESCRIPTION	5
2.1 PRODUCT PERSPECTIVE	5
2.2 PRODUCT FUNCTIONS	5
2.3 USER CHARACTERISTICS	6
2.4 GENERAL CONSTRAINTS	6
2.5 ASSUMPTIONS AND DEPENDENCIES	6
3. SPECIFIC REQUIREMENTS	7
3.1 EXTERNAL INTERFACE REQUIREMENTS	7
3.1.1 <i>User Interfaces</i>	7
3.1.2 <i>Hardware Interfaces</i>	7
3.1.3 <i>Software Interfaces</i>	7
3.1.4 <i>Communications Interfaces</i>	7
3.2 FUNCTIONAL REQUIREMENTS	7
3.2.1 <i>Functional Requirement: Card Stock & Price: UI and Database</i>	7
3.2.2 <i>Functional Requirement: Store Credit Page System</i>	8
3.3 USE CASES	9
3.3.1 <i>Use Case</i>	9
3.4 NON-FUNCTIONAL REQUIREMENTS	10
3.4.1 <i>Performance</i>	10
3.4.2 <i>Reliability</i>	10
3.4.3 <i>Availability</i>	10
3.4.4 <i>Security</i>	10
3.4.5 <i>Maintainability</i>	10
3.4.6 <i>Portability</i>	10
3.5 DESIGN CONSTRAINTS	11
3.6 LOGICAL DATABASE REQUIREMENTS	11
4. ANALYSIS MODELS	11
4.1 SEQUENCE DIAGRAMS	11
4.2 STATE-TRANSITION DIAGRAMS (STD)	14
4.3 DATA FLOW DIAGRAMS (DFD)	16
4.4 NORMALIZED DATA MODEL (NDM)	18
4.5 CLASS DIAGRAM	19
5. CHANGE MANAGEMENT PROCESS	19

Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of the Hobby Store Online Marketing System. It will explain the in depth purpose, features, interface of the system, the constraints under which it must operate and how to use the platform. SRS is mainly prepared for developers and stakeholders of this platform and will be the main document guiding the development process. Also it will be shared with the Capstone Professor, Anjana Shah.

1.2 Scope

This platform is being developed for local toronto business The Sword and Board that sells MTG game cards. The Online Marketing Platform(OMP) will help The Sword and Board to manage their inventory, employees and customers. It will have a database that will hold cards, employees and customers information. It will allow platform administrators like employees and manager to add/remove cards and inform customers what cards are currently for sale or trade. Friendly U/I will be developed to help users to interact with the platform. Managers will have control over access to this platform by adding employees to it, that way only people that are supposed to have access to sensitive information will be allowed to view or edit informations. Managers will also have view to how store is doing financially, by having access to store's sales and refunds that will be tracked by the platform. Customers will be able to buy and reserve cards online because, at this point they can only do that by actually going to the store by themselves. Customers will be able to view and edit their store account, keep up to date all their personal informations and view their store credit. This OMP will fill the communication gap between The Sword and Board store and it's customers.

1.3 Definitions, Acronyms, and Abbreviations

- MTG - Magic The Gathering
- U/I - User Interface
- OMP – Online Marketing Platform
- SRS - Software Requirements Specification
- S&B - The Sword and Board

1.4 References

IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

Reference File Name	Version	Description
Software Requirements Specification Template	1	Template provided by professor Anjana Shah
Project Vision	1	Project Vision document is part of Sprint 1 of this project
Project Plan and Team Charter	1	Project Plan and Team Charter is part of Sprint 2 of this project

1.5 Overview

The remaining parts of the SRS contain specifics regarding all aspects and angles of the OMP, including but not limited to UI design, customer interaction, systems and features, constraints, diagrams and external resources. The SRS is organized from the most general topics to the most specific, finally concluded with diagrams.

2. General Description

The next chapter, the Overall Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish context for the technical requirements specification in the next chapter.

2.1 Product Perspective

This product aims to achieve all the goals of other competing MTG OMP softwares. In addition the platform looks to evolve the typical OMP into a more diverse, modular program that can serve more hobby stores needs. This will be achieved by allowing the modular parts of the software to be enabled and disabled at the will of the business.

2.2 Product Functions

The provided software shall be capable of three things.

1. The product shall contain a system for tracking and managing customers store credit.
2. The product shall contain a system for tracking and managing a stores stock of MTG cards.
3. The product shall contain a system to scrap prices from competitors, which shall be used as a reference to display the stores prices on MTG cards.

2.3 User Characteristics

Software users fall under two groups: Customers and employees. Both of these groups shall interact with different parts of the systems that are directly related to each other. Employees shall expect a simple interface that is easy to navigate and quick to get them to where they need to go in the software. Employees don't have time to jump through different links to find the information that they came looking for. Customers on the other hand want certainty and easy navigation. They want certainty that stock levels won't change, or that prices won't change randomly change or that those numbers are correct. These expectations shall be taken into account when making the project in that for employees they will never need to click more than twice to get what they want, and for customers, a mission statement will be clearly indicated on the pages relating to these expectations.

2.4 General Constraints

The software being produced depend on two third party sources. The source of all cards come from the card manufacturers own database. Card prices also come from third party retailers and price indexes.

The stakeholder will not be implementing all of the developed software. Since the software will be developed modularly, this should not be an issue.

Card prices are not dependant on card stock. These two metrics may move independently of each other.

2.5 Assumptions and Dependencies

The system shall be written in some sort of stack, requiring server space. Access to a server is assumed and depended on.

Implementers of the product will have internet access and thus working computers.

These are critical dependencies and without access to them, the product is virtually useless.

3. Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

There will be two main user interfaces for both the customers and employees. These interfaces relate to the card OMS and the store credit interfaces. The customer sides will be view only interfaces which will contain all relevant information to them. The employee side of the interfaces shall contain everything the customer side of the interfaces contain, as well as the ability to edit any of the information displayed to the customers.

3.1.2 Hardware Interfaces

The hardware interface will be whatever internet browser capable devices customers and employees use to access the system. These can range in physical size from handheld devices to typical desktop computers.

3.1.3 Software Interfaces

The software interfaces shall be where a business manager may change core functionality of the software as well as change the permission levels of certain accounts. (E.G adding/removing employees, banning/unbanning customers, etc).

3.1.4 Communications Interfaces

Any internet enabled device with keyboard and mouse/touch gesture input.

3.2 Functional Requirements

3.2.1 Card Stock & Price: UI and Database

3.2.1.1 Introduction

The OMP for short is the main purpose of this product. In short, this will be a way to manage stock on the backend for a business, and a way to view prices and said stock on the front end for customers.

3.2.1.2 Inputs

Users may use a search bar function to search for specific cards or they may use the more advanced search feature which takes certain characteristics of a card(such as edition, colour, other game related statistics, set it was released in, formats of play it is legal in etc) as parameters. This would return all matching cards which fit these constraints as well as their stock levels and/or price.

3.2.1.3 Processing

All requests will be processed on the server side. Card searches will use the GET HTML method. Card prices will be preprocessed at a low impact time of day and appended to the database then as to not constantly ping external servers repeatedly and redundantly.

3.2.1.4 Outputs

Outputs will simply be what the server finds that matches the input. These will be displayed in a table format, separating key information into logical columns. Additionally an image will be provided of the card with which customers can crosscheck and make sure that that is what they were searching for. Also a text version of the information on the card will be provided for indexing purposes.

3.2.1.5 Error Handling

When an error is encountered, the use will be returned to the main search page where an error message will be displayed below the search function.

3.2.2 Store Credit Page System

3.2.2.1 Introduction

The store credit page is a core business feature for this style of store. Since hobby stores buy many used MTG cards and games from customers, they need a more cost effective way to award these customers money that isn't expensive gift cards. A store credit page will be a database that maintains the account holder's name, maintains an account balance and maintains a record of interaction along with their dates. Pages should be editable by employee's and viewable with security by customers.

3.2.2.2 Inputs

On a secured log-in web-page customers may view their store credit page which will be automatically fetched. Access is restricted to logged-in customers. Customers can only see their own store credit pages.

Employee's will have a search bar with which they can type in a customer's name. This will bring up a page which will contain the customers name and current store credit balance. The employee will have a few separate fields to fill in including the amount of credit to be added, the amount of credit to be removed and comments surrounding the transaction. Automatically with this submission the employee's name will be submitted (as they would have to be logged into the system to access the employee resources) as well as the date of the transaction. Alternatively if the customer doesn't have a store credit page yet, beside the search bar, a "make new account" button will exist to instantiate a new customer's account. This would require the customer to create a password, submit the first and last names as well as allow the employee to set any starting account transactions(for example, adding store credit right away).

3.2.2.3 Processing

Client side forms will be linked to a server-side database. The POST method will be used to log users of the Store Credit System in as well as when their account is created. The POST method will also be used when employees are submitting new transactions. The GET method will be used when the customers pages are loaded into their account page.

3.2.2.4 Outputs

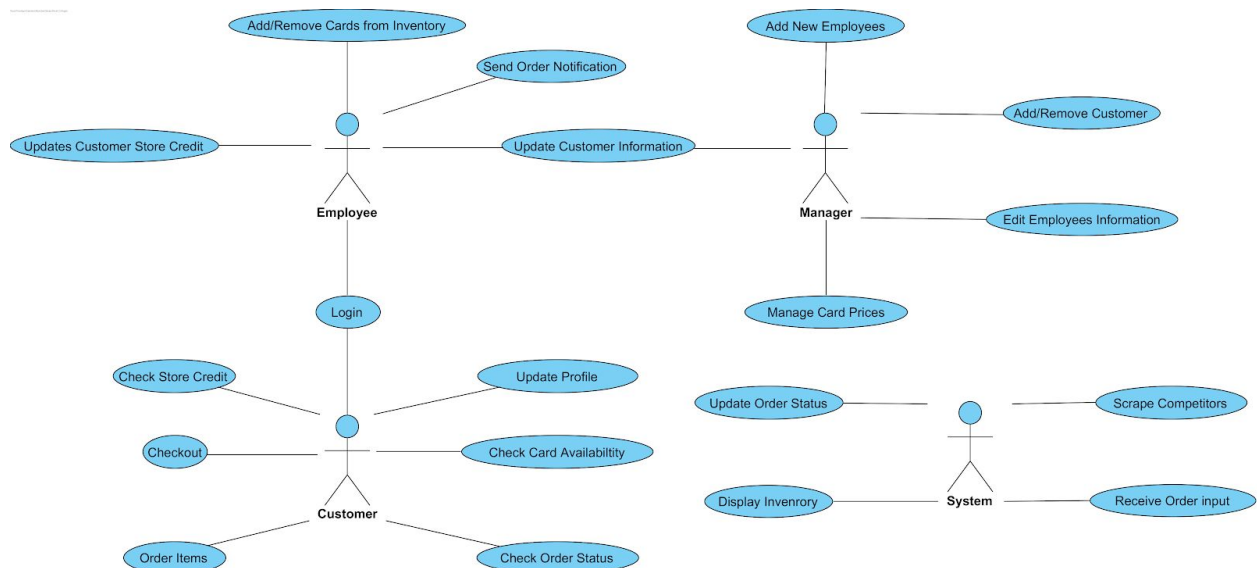
There is one output for the store credit system. This is the customer's page that is stored in the database. On the page, users of the system will see the account holders name, dates with correlated transaction records as well as the employee that enabled the transaction. This information is read only.

3.2.2.5 Error Handling

When a user not found error is encountered by an employee, the system will redirect the employee to the page with the search for user box. For a customer, if their user page isn't found when they log in, they will get a box in the place the store credit system is usually displayed indicating the error message. If any other non-fatal errors are encountered a console message will be displayed and the user will be redirected to where they previously were in the software.

3.3 Use Cases

See attached image to view a larger version:



3.4 Non-Functional Requirements

3.4.1 Performance

Card searches must be instantaneous. Having searches take any longer than one second means the system is inefficient and not doing a good job at replacing the previous slow system.

3.4.2 Reliability

Maintaining reliability is a two step process. First server updates for the card database are done daily and automatically at an off peak hour to grab any price updates. Additionally a background script will run once every few hours when the system isn't in use to check if any prices have significantly changed and update those accordingly. The second point of maintaining reliability is that if system updates aren't able to be done either through error or other reasons, the previous most accurate prices will be presented. Employee's will be notified when they log in, when the last database updates occurred.

3.4.3 Availability

Access to the card stock database will be limited to visitors of the website and employee's viewing the system locally. The services will be available 24/7 except during the time the server is updating for obvious reasons. Access to the store credit system will be restricted to logged in users and employees only. Anyone can make a store credit page by asking an employee in store. One will automatically be created when a customer is awarded store credit.

3.4.4 Security

Access to customer accounts will be password protected. Store credit balances will be stored in a hashed format that will make any security breaches less impactful. Employee access will only exist at the stores location through an offline portal with a two-factor hashed key system. This should be the most secure system reasonably obtainable. All changes to the store credit database are logged so any mistakes or unapproved changes can be edited by the administrator.

3.4.5 Maintainability

Many aspects of the system are self-maintaining. As mentioned before the card database will have regular automated updates to keep cards and prices current. The store credit system is quite simple and shouldn't need any major maintenance. Occasional database integrity checks will be done. Caching of historical data may be performed every few years to reduce database data size.

3.4.6 Portability

Since this is a web based system, all of the services available online for the customers will be infinitely portable, as long as one has an internet connection. The employee tools will only be available locally on the store computer and is therefore not portable.

3.5 Design Constraints

The primary implementers (S&B) do not currently want to list their cardstock. This is for two main reasons.

1. Counting and adding all their cards to the database would take thousands of hours to complete, and they currently do not have the time or money for that large of an endeavor.
2. Not implementing the stock module means that people will have to go in, or contact the store to enquire about the cards they were interested in. This increases the likelihood that employees can upsell to the customers.

3.6 Logical Database Requirements

Two databases shall be used.

The first one, for the cards will be mirrored from the card manufacturer themselves who provide an API to stores for this very use. Due to this, all of the databases design is already done, the software will just have to be able to access it and append information.

The second database, for store credit tracking will be quite simple. Only entrenched customers will generally have store credit, as it is only awarded for events and for collection trade-ins so only a small fraction of the customer base shall need to be in the database. The customers shall be stored using their full names as a searchable key in the database. Attached to their names will be their current account balance, as well as past transactions with notes regarding what the transaction was for and the dates on which the transactions were processed. Data will be stored in plaintext for the most part, storing account balances in a hashed format shall be discussed with *Will a database be used? If so, what logical requirements exist for data formats, storage capabilities, data retention, data integrity, etc.*

4. Analysis Models

List all analysis models used in developing specific requirements previously given in this SRS. Each model should include an introduction and a narrative description. Furthermore, each model should be traceable the SRS's requirements.

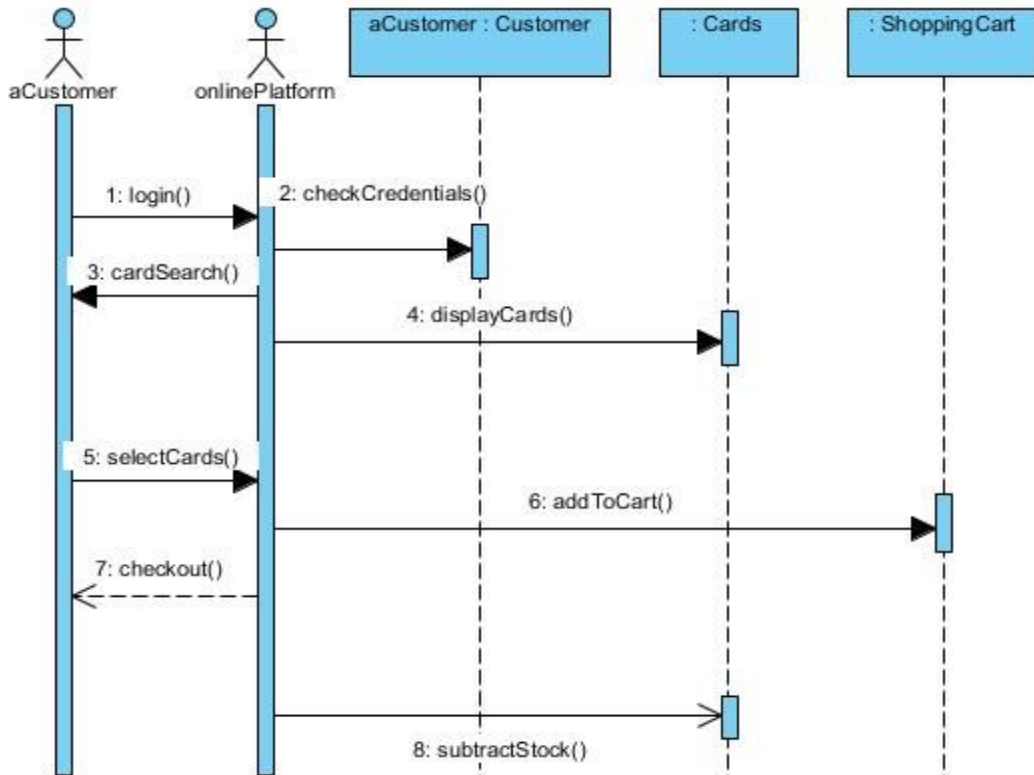
4.1 Sequence Diagrams

Sequence Diagram #1

This diagram depicts the sequence that could be made by the Customer if they were to log into the online platform with their own account. Once they pass the login with the right credentials they will have the ability to search for the specific cards they want or view all cards in the database. Once the database displays all the cards, the Customer can select the cards they want and add those selected cards to their own shopping cart. From there, the customer will be

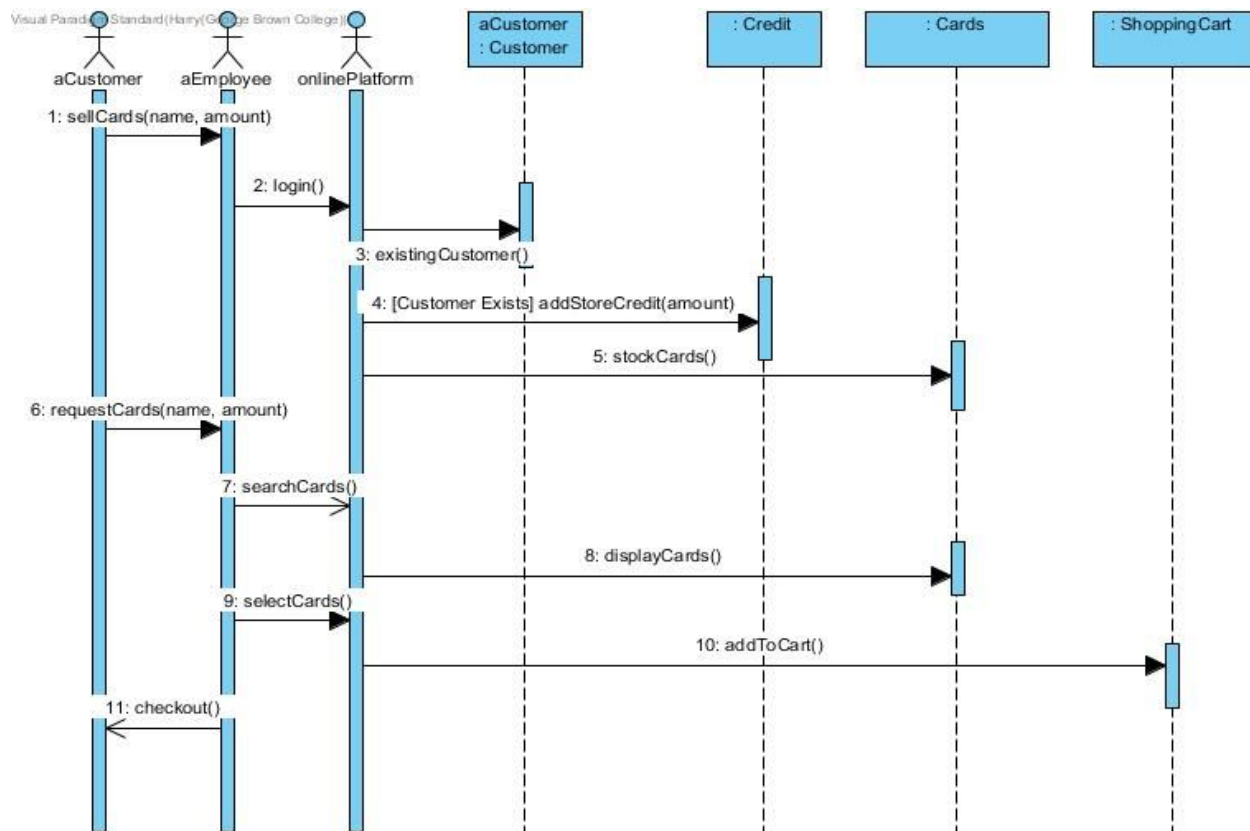
prompted to proceed to checkout. Once confirmed, the selected cards will have their stock reduced by however many cards were bought by the customer.

UML Paradigm Standard (Harry George Brown College)



Sequence Diagram #2

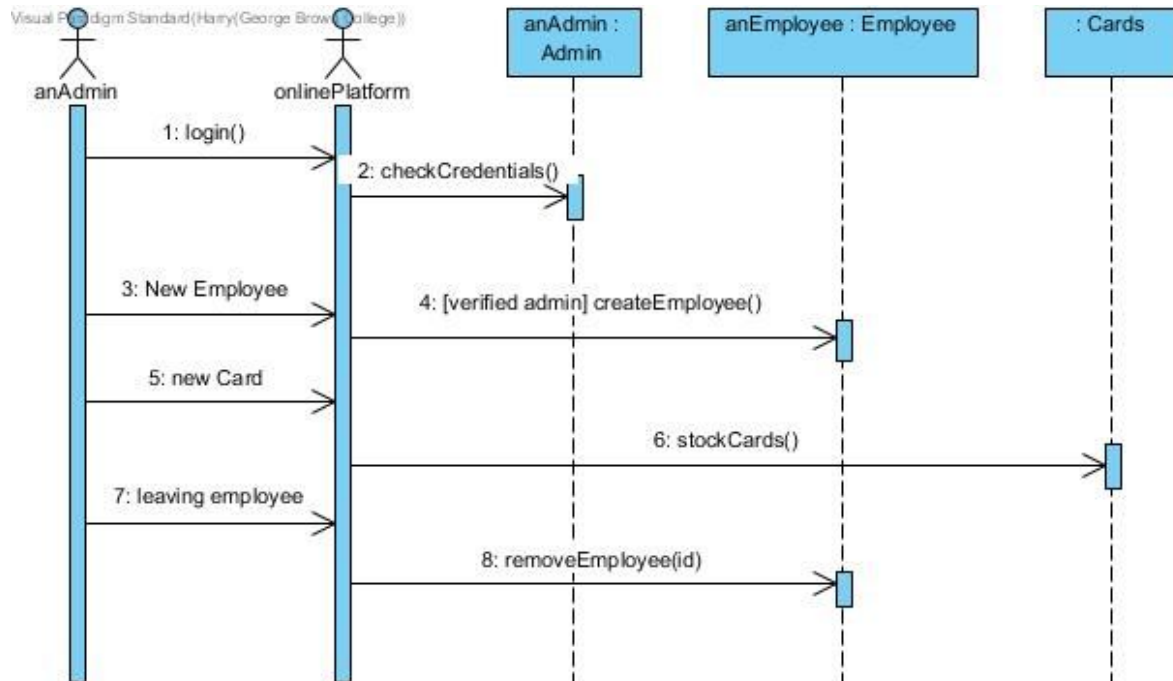
This Sequence diagram is in the context of the Customer coming to the store and interacting with the Employee. Assuming that the Customer is already in the database, the Customer can sell their cards to the store and in return they can earn store credit on their account. When the customer sells their cards to the store it can be added to the database by the Employee. The customer can request to buy cards and the employee can search up the cards and add them to the shopping cart for the total price that must be paid. Employee can prompt the customer what method they want to pay with, whether with store credit plus money, or without store credit.



Sequence Diagram #3

This diagram is in the Context of the admin using the platform. As you know the admin can add new employees and edit the database when he sees fit. So in this sequence the admin will login with their credentials. Let's say that a new employee is added to the team. The admin can create an account for them in the database with a corresponding id. If customer comes in with a new set cards and the admin is in attendance to help the customer, they can add those cards to the

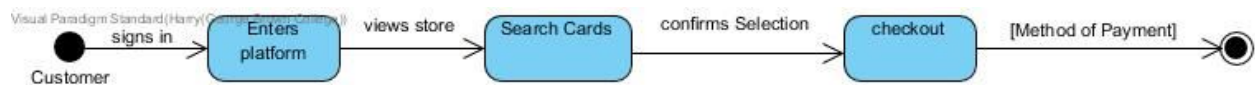
database also. If there is an Employee leaving the company the admin has the authority to remove the Employee's account with their corresponding ID from the database.



4.2 State-Transition Diagrams (STD)

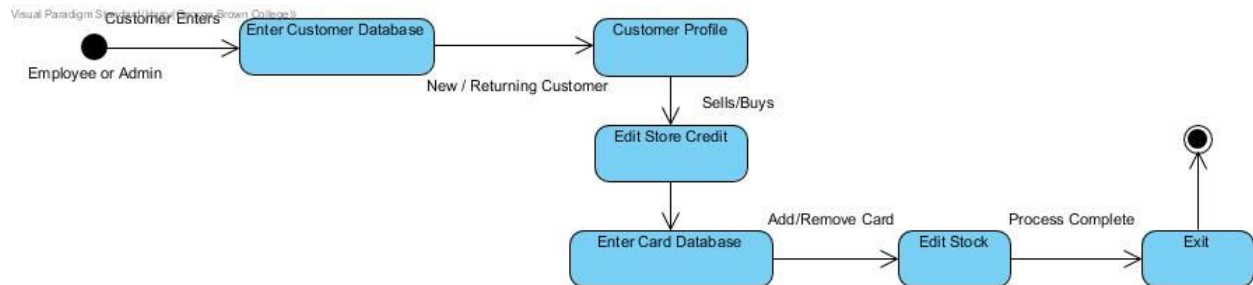
State Transition Diagram 1

This State Transition Diagram describes the Customer moving through the states of buying MTG Cards. The Customer signs in to the platform. Once they have entered they can view the store and search cards in the database. Customer will then move on to the checkout state after confirming their selection of cards. Finally, they will choose their method of payment which can be through credit card, payment in store or by using their store credit.



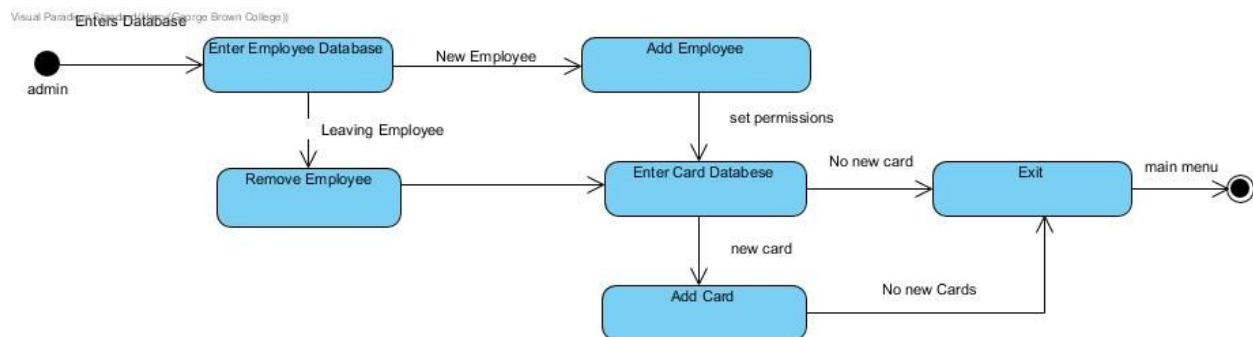
State Transition Diagram 2

This State Transition can be in the context of both Employee or Admin roles editing the customer database and card database. In this context the Customer enters the store, then the employee/admin will enter the customer database and enter the customer profile. From there the Employee has the option to add store credit if the customer is selling their cards or take out store credit that the customer plans to use to buy a card(s). Then the Employee can add or remove the card from the stock and exit the platform.



State Transition Diagram 3

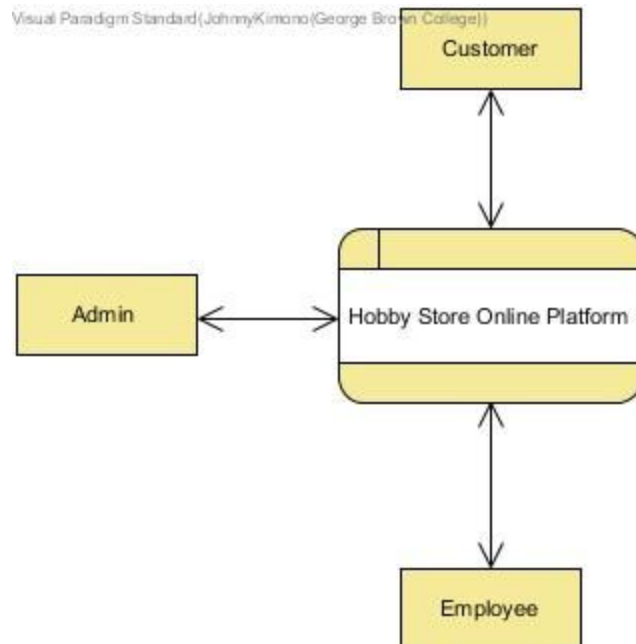
This is in the Context of the Admin using the Employee database and card database. When there is a new Employee the Admin will login to the Employee database and enter the State of adding an employee profile. In the incident that there is a leaving employee, the Admin will enter the state of removing the employee. After the Admin has set the permissions for an Employee, they can enter into the card database. If they have a new card to add they will proceed to the state of adding cards to the database. If there are no new cards then they will proceed it exit.



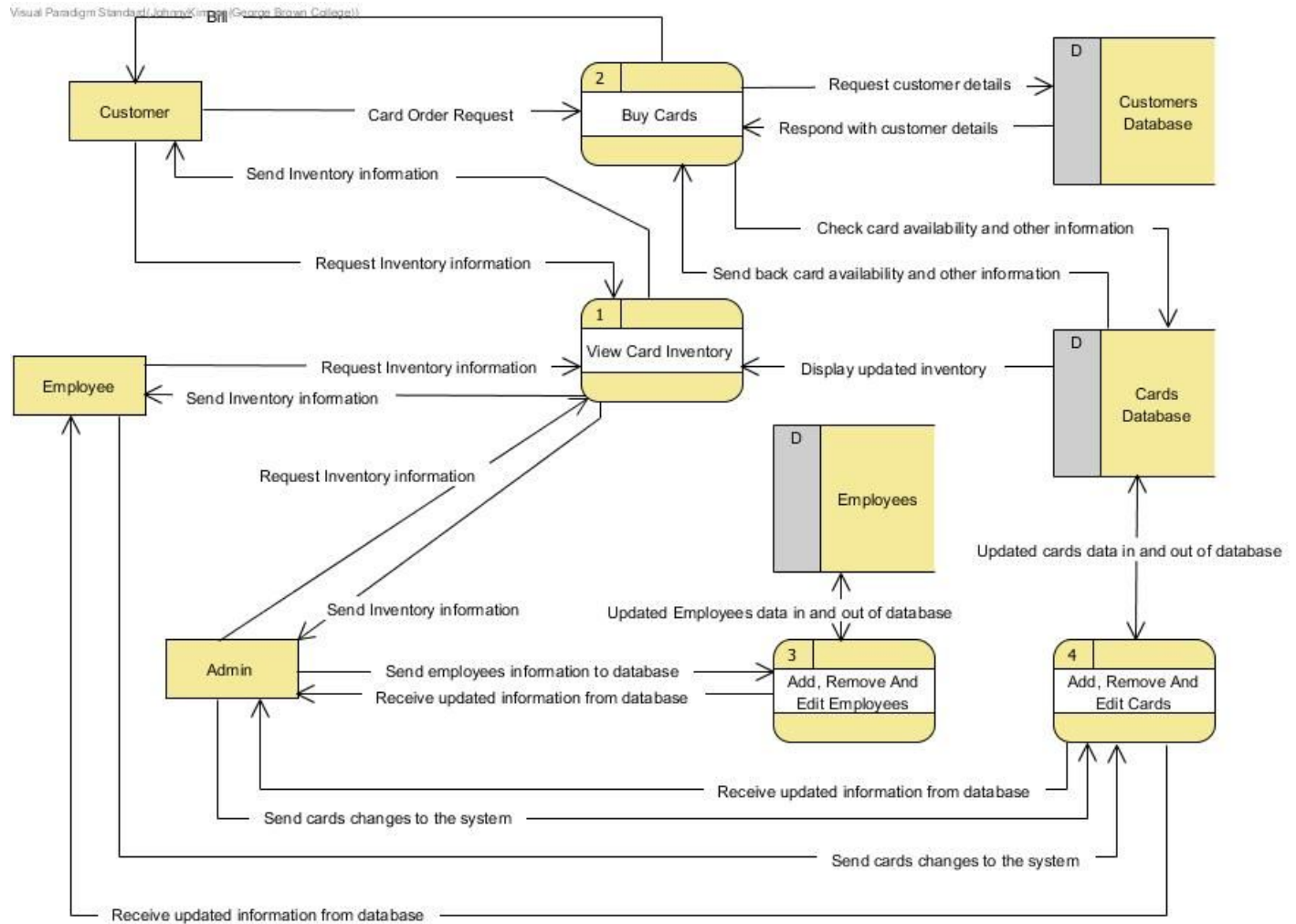
4.3 Data Flow Diagrams (DFD)

This data flow diagrams show how external entities send and receive data from Platform and how platform's databases are updated to keep data integrity in good condition and up to date. In our level one diagram, we have 3 main entities that interact with our platform Admin, Customer and Employee. We also have four processes that handle data received from external entities and from platforms databases. First process "View Card Inventory" which basically allows external entities to view store's inventory, what is available for purchase ,quantity and prices. This process gets data from Cards Database and keeps all entities updated with store's inventory. Second process "Buy Cards" handles Customer's request to buy cards. Customer sends card order buy cards process checks database for information about customer, database responds with details back.This process also checks cards availability in Cards Database and database responds with cards detail. Finally if all customers information are provided need to process buy, shipping and cards are available, bill is send to customer. Third process "Add, Remove And Edit Employees" allows Admin to update database with current employees so that, only people that should have access to platforms sensitive information have that access, like current Employees and Admins. Forth process "Add, Remove And Edit Cards" allows Admins and Employees to update cards information in Cards Database so that platform can be up to date with newly acquired cards, sold cards and price changes of current inventory.

Context DFD

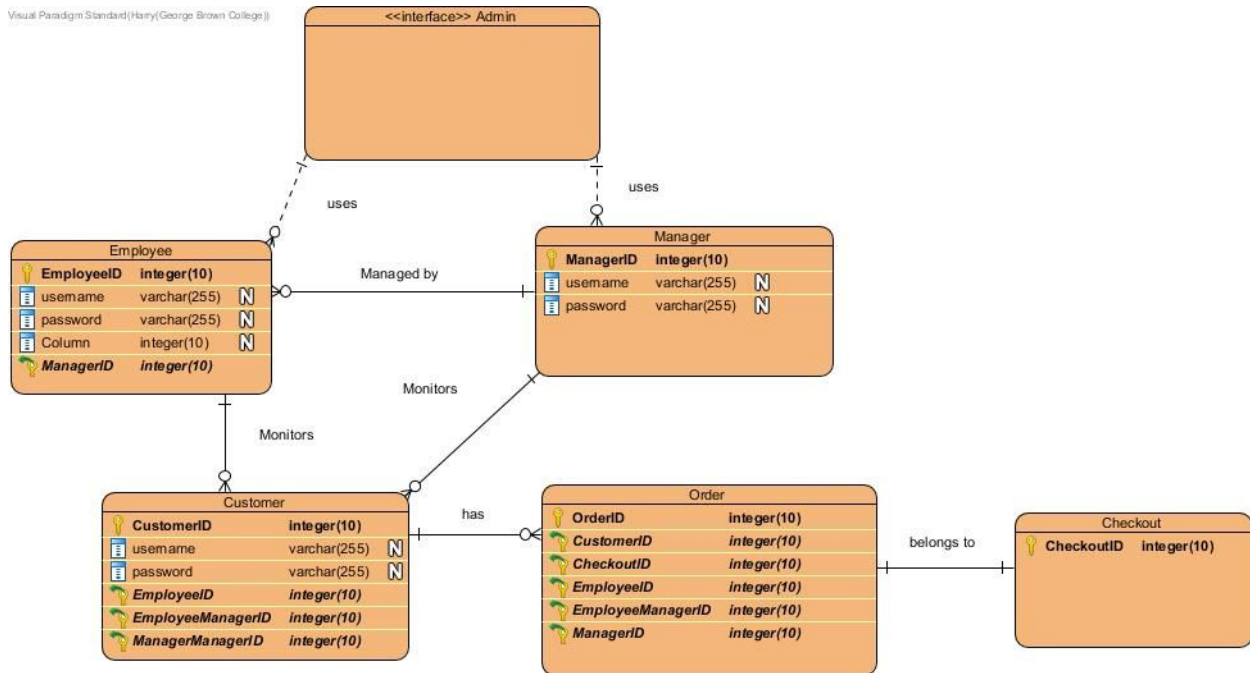


Level 1 DFD

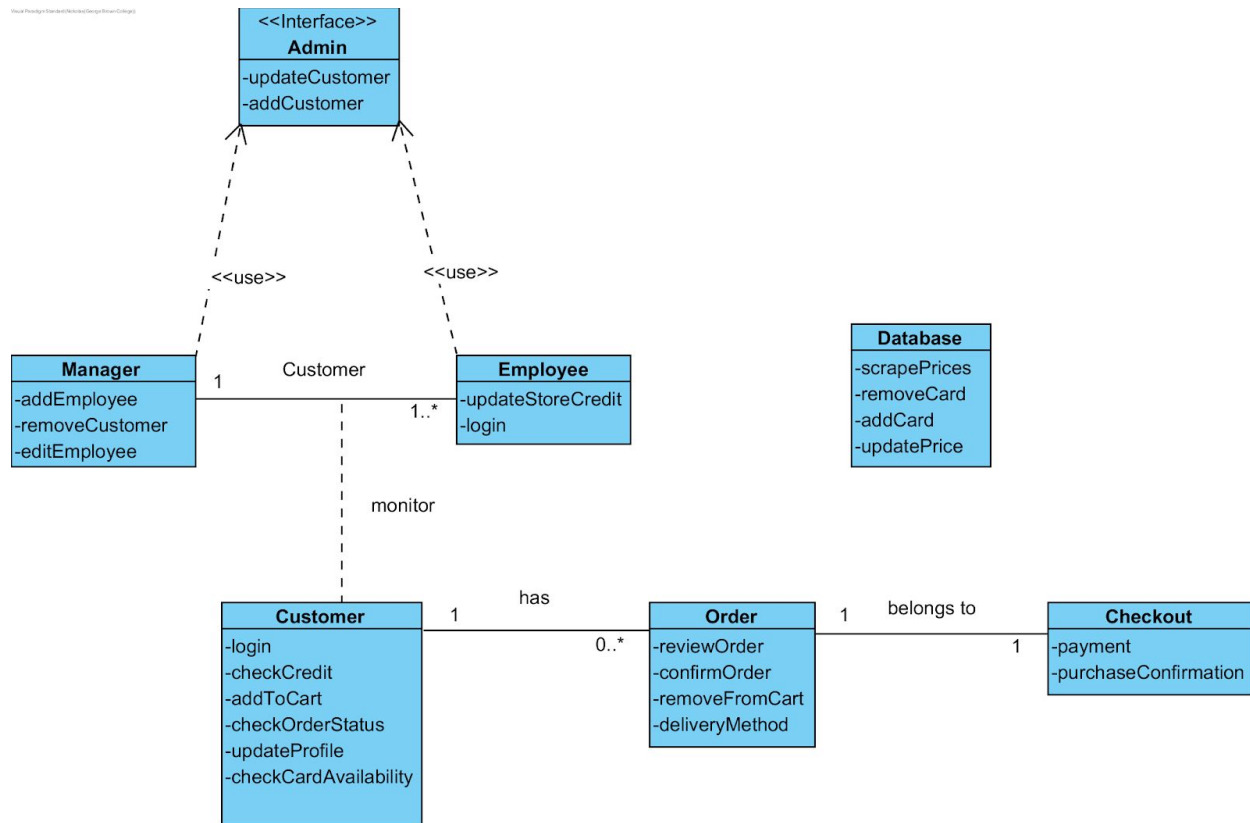


4.4 Normalized Data Model

Visual Paradigm Standard (Harry George Brown College)



4.5 Class Diagram



5. Change Management Process

Changes to the SRS may be proposed by any member of the group or stakeholder team. Nolan will have the final say on whether the changes get approved and implemented. Nolan or another group member appointed at that time will make the changes. The Updated SRS will be resubmitted by Nolan.