College Bytes, LLC.

**Class Bytes**

**Technical Manual**

Table of Contents

[Introduction 3](#_Toc450474649)

[Installation 3](#_Toc450474650)

[Installation Using Minimum System Requirements 3](#_Toc450474651)

[Installation Using Recommended System Requirements 4](#_Toc450474652)

[Final System Configuration 5](#_Toc450474653)

[Backup Procedures 5](#_Toc450474654)

[Administrator System Access 7](#_Toc450474655)

[System Processes 7](#_Toc450474656)

[Context Level Data Flow Diagram 7](#_Toc450474657)

[Level 0 Data Flow Diagram 8](#_Toc450474658)

[Process 1.0 Level 1 DFD 10](#_Toc450474659)

[Process 2.0 Level 1 DFD 10](#_Toc450474660)

[Process 3.0 Level 1 DFD 11](#_Toc450474661)

[Process 4.0 11](#_Toc450474662)

[Process 5.0 11](#_Toc450474663)

[Process 6.0 Level 1 DFD 12](#_Toc450474664)

[Process 7.0 Level 1 DFD 12](#_Toc450474665)

[Process 8.0 Level 1 DFD 13](#_Toc450474666)

[System Database Entity Relationship Diagrams 14](#_Toc450474667)

[Production Version 14](#_Toc450474668)

[Alpha Version 15](#_Toc450474669)

[Troubleshooting 16](#_Toc450474670)

# Introduction

The Class Bytes System from College Bytes, LLC is a dynamic scheduling and information system designed specifically for use by students in a collegiate setting. With the Class Bytes System, users can manage an unlimited number of classes, instructors, assignments, and grades, and be able to access, modify, view, or delete any of their class-related information from anywhere that they have access to an internet connection.

# Installation

Minimum System Requirements:

* 1 x Server, configured as a web server with an SQL DBMS and PHP interpreter preinstalled
* 1 x Dedicated high-bandwidth internet connection
* Class Bytes System database
* Class Bytes System web front-end package

Recommended System Requirements:

* 2 x Web server, both configured as web servers with PhpMyAdmin preinstalled
* 1 x Database server, configured with MySQL preinstalled
* 1 x Local DNS server, configured for load balancing
* 1 x Dedicated backup server
* 1 x Dedicated high-bandwidth static internet connection
* Class Bytes System database
* Class Bytes System web front-end package

## Installation Using Minimum System Requirements

**Note:**

*The Class Bytes System is based on a MySQL server with a PHP interpreter. Any installation of the system should have a server with a dedicated internet connection utilizing a static IP address. The use of DDNS on a dynamic IP address is feasible, though highly discouraged. The server will need to be configured as a web server, and have the latest versions of MySQL and PhpMyAdmin installed. The use of other SQL database management systems or PHP interpreters have not been tested, and are not advised for use.*

It is highly encouraged that the server to which the Class Bytes System is installed to be dedicated solely as a web server with MySQL and the PhpMyAdmin interpreter. Once the server has been properly configured and is accessible from the internet via a system operator specified domain, installation of the Class Bytes System is fairly straightforward. First, the database will need to be imported into the MySQL server. Once the database has been successful imported, the entire HTML and PHP web front-end will need to be loaded onto the web server. The individual web pages in the front end are coded using relative addressing, so there is no need to change any addressing on the pages. Please note, the minimum system requirements do not specify any backup system. Although not technically required for system operation, a backup system is strongly recommended. See Figure 1 below for the system diagram utilizing the minimum systems requirements.

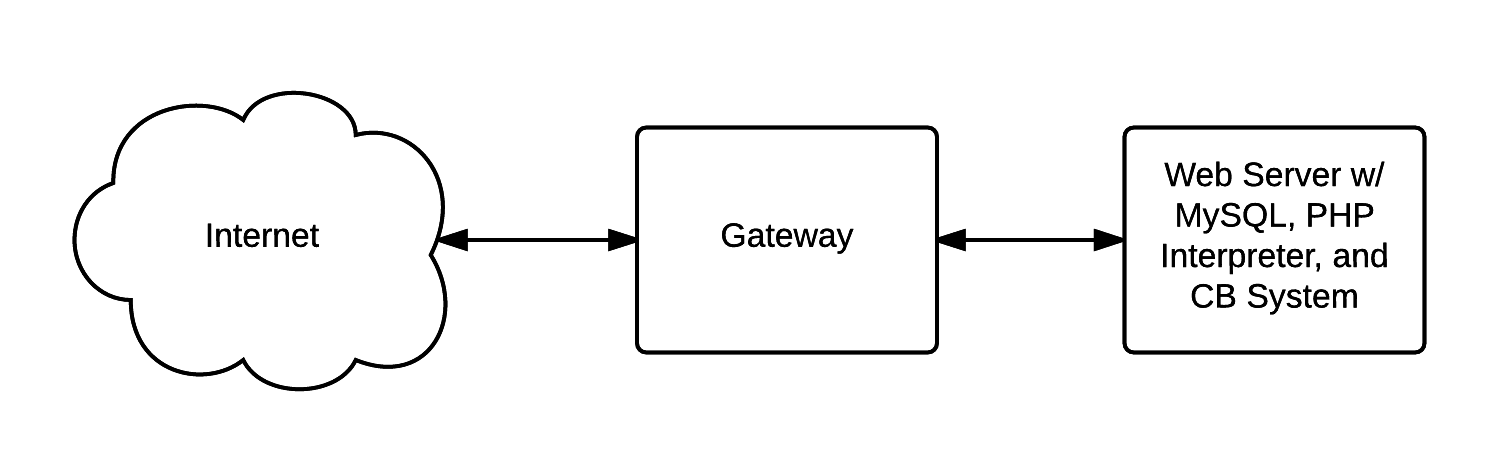


Figure 1 - Minimum System Requirements

## Installation Using Recommended System Requirements

**Note:**

*The Class Bytes System is based on a MySQL server with a PHP interpreter. Any installation of the system should have a server with a dedicated internet connection utilizing a static IP address. The use of DDNS on a dynamic IP address is feasible, though highly discouraged. The server will need to be configured as a web server, and have the latest versions of MySQL and PhpMyAdmin installed. The use of other SQL database management systems or PHP interpreters have not been tested, and are not advised for use.*

Installation of the Class Bytes System using the recommended system requirements will require the system operator to configure two identical web servers for connection to the internet through a dedicated internet connection with a single high-bandwidth static IP address. Additionally, an internal DNS server will need to be configured for web server load balancing. Load balancing will prevent web server overload and will increase system availability. A database server will also need to be loaded with MySQL and configured to interface with both of the web servers. Finally, a backup server will need to be configured to backup one of the web servers and also serve as a backup for the database server. System backup is discussed more fully in the Backup Procedures section. See Figure 2 below for the system diagram utilizing the recommended systems requirements.

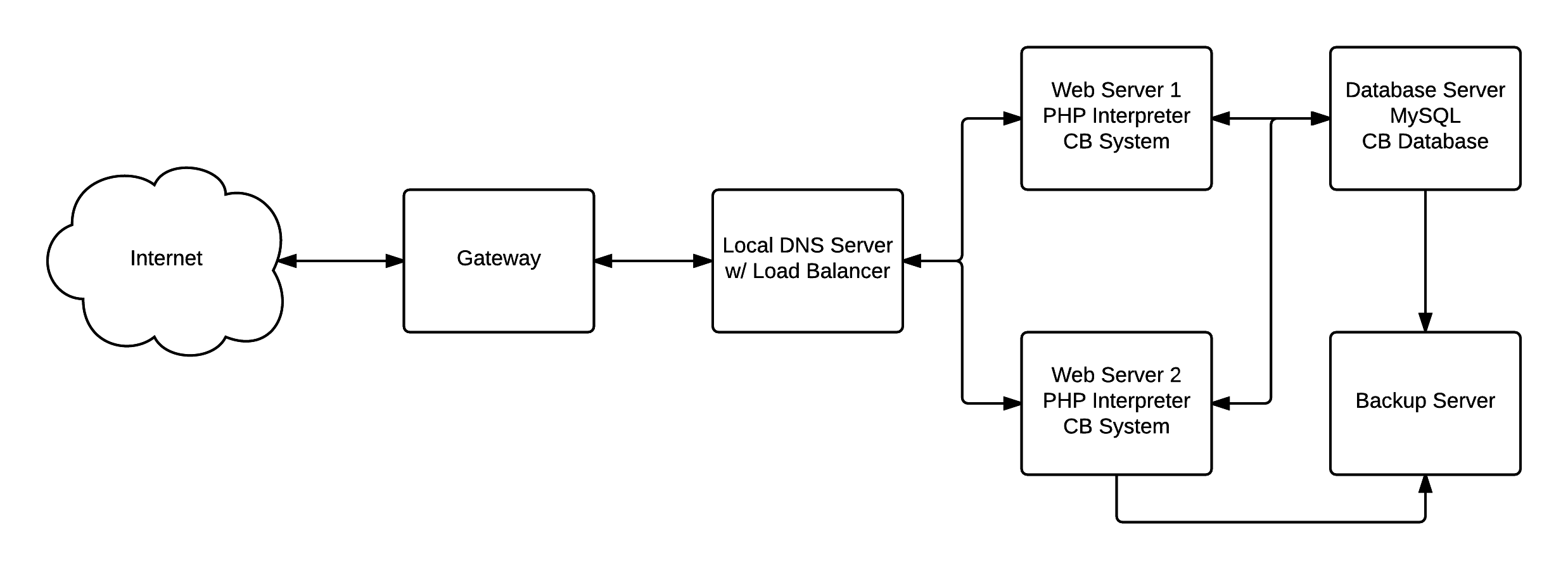


Figure 2 - Recommended System Requirements

## Final System Configuration

Prior to bringing the system online, the payment processing module will need to be configured with the proper payee information for the installing organization. Currently, this is accomplished by editing the PHP code controlling the payment system, located in the “creditcard.php” file. See Figure 3 below for more information.

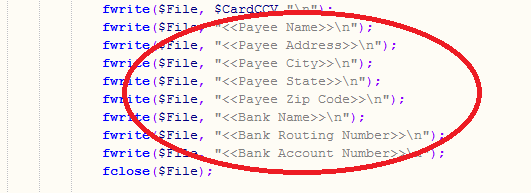


Figure 3 - Payment Configuration

# Backup Procedures

To ensure maximum system reliability, performance, and uptime, a backup system is highly encouraged for any Class Bytes System installation. These backup procedures are assuming that the system operator is utilizing the recommended system requirements. The use of a stand-alone backup server is recommended for use in backing up the entire system. As the web servers are identical in nature, and are managed by the DNS server’s load-balancing system, only one of these servers needs to be backed up, as if one fails while in operation, the other can take over while a replacement server is installed. This will result in some network slowdowns, but will still produce a high uptime rate. Also, if updates to the system are required, they can be performed on one server at a time, taking that server offline to install and configure the update while the other remains in operation. Additionally, unless an update to the Class Bytes System is installed, the content of the web servers will not change in any meaningful way. It is recommended that a full system backup of one of the web servers be accomplished at a time of low system usage at least one a month, or upon the installation of any system update.

The most critical point of failure in the system is the database server, as it houses all of the user’s information, and will render the system unusable should it fail. Therefore, it is recommended that the backup server be configured to provide backup services in a three-fold manner:

* 1st of every month – Full database backup
* Midnight each day – Differential database backup
* Every 15 minutes – Transaction log backup

Should the database server fail, it will result in system downtime. However, by applying the full system backup, the differential backup and then the transaction log backup to a replacement server, the Class Bytes System should be able to be recovered quickly, with little to no loss of data. If the database server were to fail, it is also possible to notify users of the brief system interruption by posting the details to the web servers, which will remain unaffected by a database server failure.

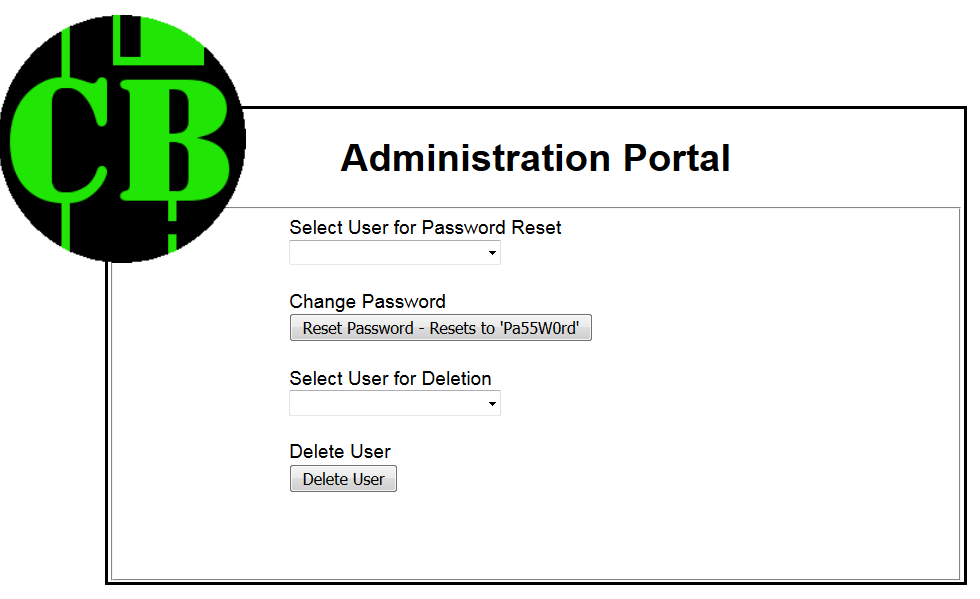
# Administrator System Access

There exists in the Class Bytes System an Administrator Portal. Through this portal, system administrators may add new system administrators (feature to be included in upcoming revision), conduct password resets for users, and delete users and their associated data from the system. Access to the administrator portal is through a link on the system main page, in the lower left-hand corner. The default administrator username and password are as follows:

Username: admin

Password: Pa55W0rd

Upon system installation, and before placing the system into operation, this username and password MUST be changed to maintain system security integrity.



# System Processes

## Context Level Data Flow Diagram

The view of the Class Bytes System from a context level shows how the system interacts with both the user (Student), and the external entity (Credit Card Processor), including all relevant flows of data between the entities. See Figure 4 below.

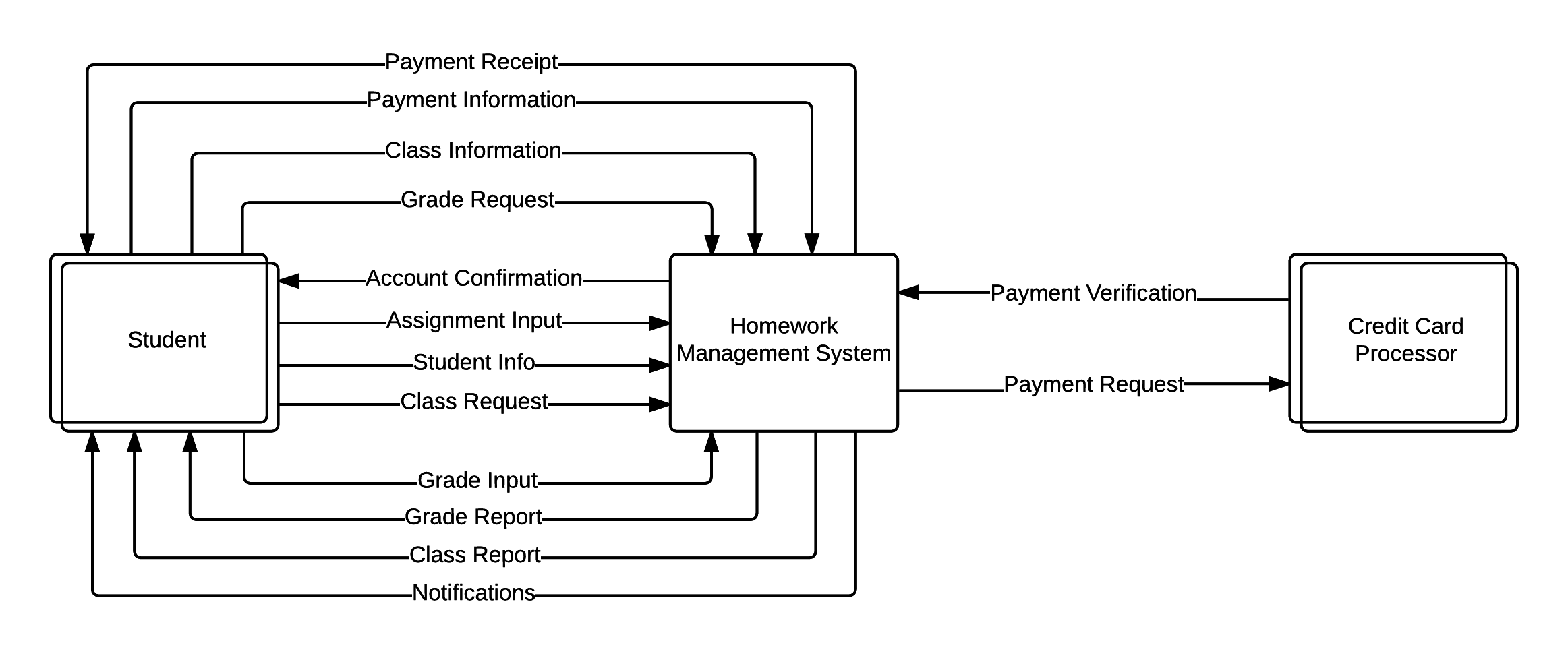


Figure 4 - Context Level DFD

## Level 0 Data Flow Diagram

The level zero data flow diagram shows how each individual module of the Class Bytes System works within the system and interfaces with each other. Further descriptions of each process are listed below. See Figure 5 below.

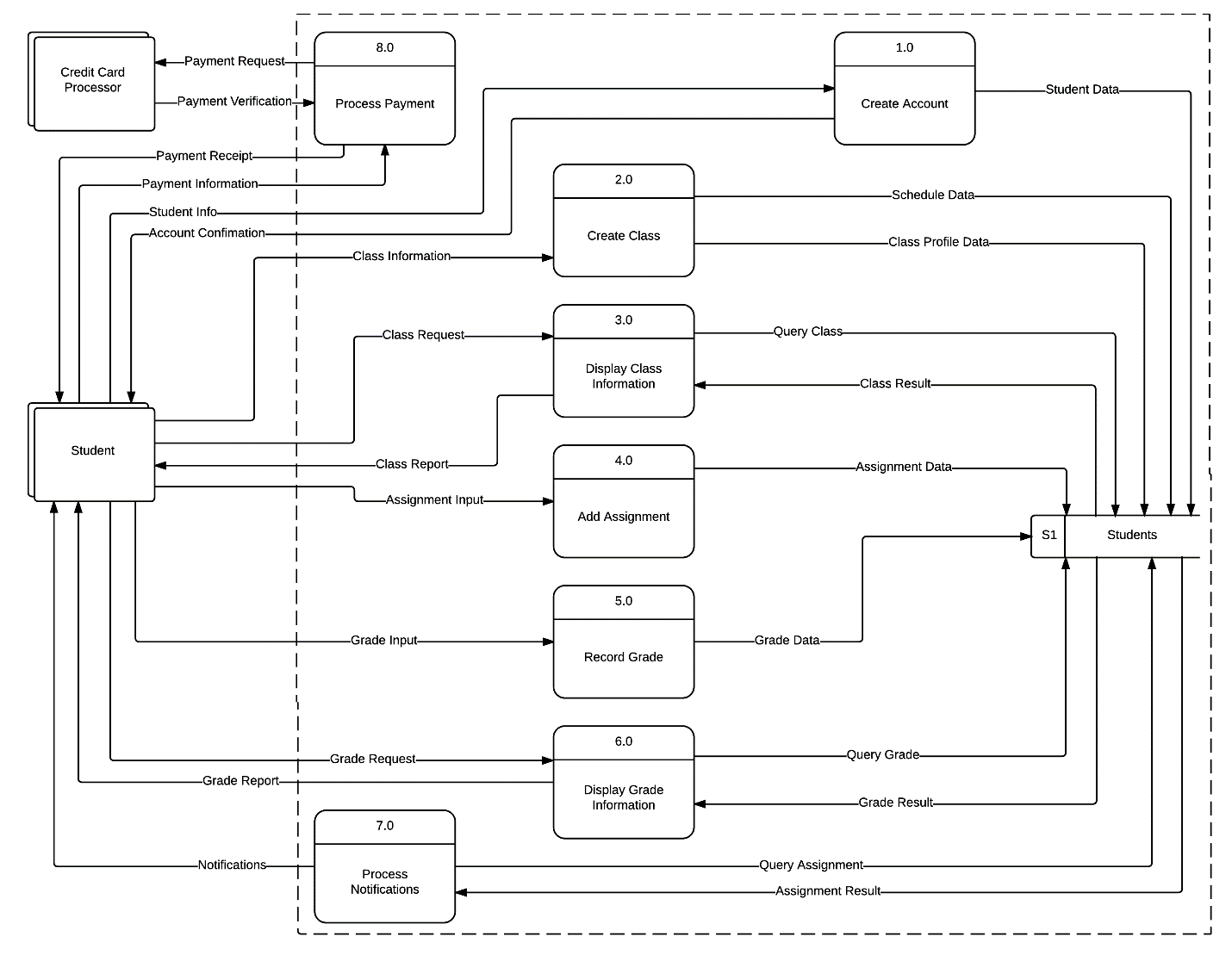


Figure 5 - Level 0 DFD

### Process 1.0 Level 1 DFD

The 1.0 process, Create Account, details how the user’s information is taken in to the system. This process is utilized whenever a new user registers their user account to the system for the first time. Their information is taken in, checked for validity, and if confirmed valid, placed into the database. A successful validity check will also generate a confirmation to the user. The user sees this confirmation in the form of moving on to the login page of the system. Any failure of this process will generate an appropriate error message for the user. Process 1.0 is detailed in Figure 6 below.

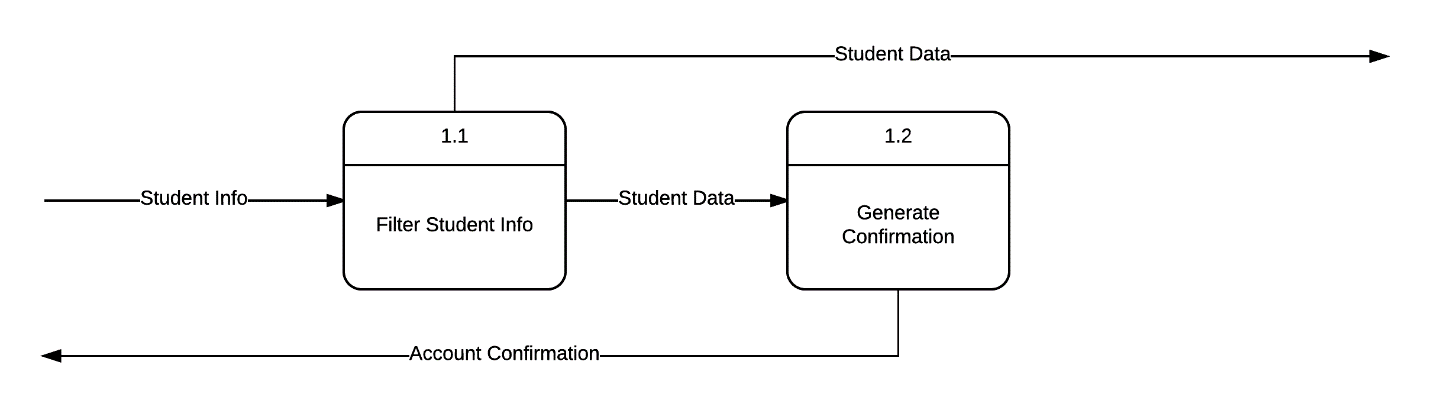


Figure 6 - Process 1.0

### Process 2.0 Level 1 DFD

The 2.0 process, Create Class, details how a user’s class information is taken into the system. This process is utilized any time the user adds a new class to their account. The class information submitted by the user is checked for validity, split into two separate flows of data and then stored in the appropriate tables in the database. A successful validity check will result in the data being entered into the database. Any failure of this process will generate an appropriate error message for the user. Success is indicated via a confirmation message to the user. Process 2.0 is detailed in Figure 7 below.

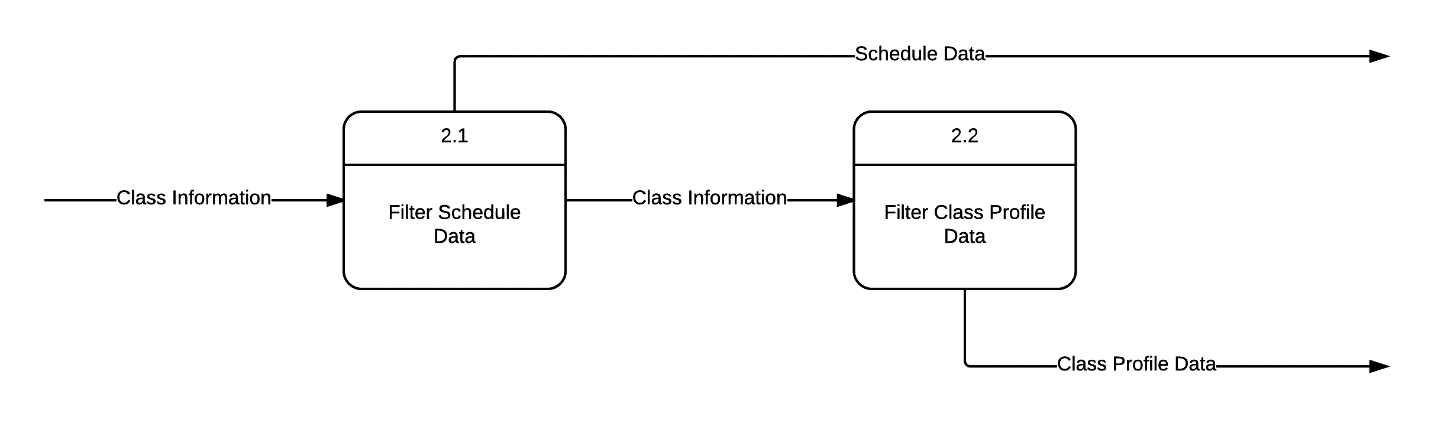


Figure 7 - Process 2.0

### Process 3.0 Level 1 DFD

The 3.0 process, Display Class Information, details how a user’s request for class data is processed by the system. This process is utilized automatically any time the user visits a page within the system that displays class-related date, or explicitly when a user selects a specific class to view information on that class. Any failure of this process will generate an appropriate error message for the user. Success is indicated by the user successfully viewing the information they requested. Process 3.0 is detailed in Figure 8 below.

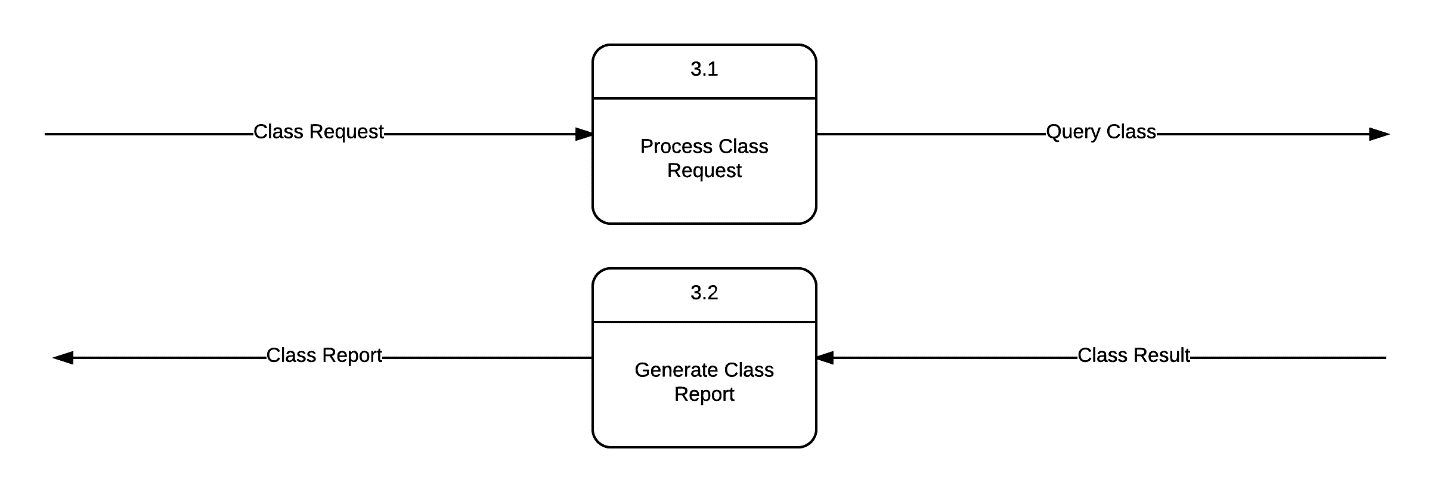


Figure 8 - Process 3.0

### Process 4.0

The 4.0 process, Add Assignment, details how a user’s assignment data is entered into the system. The process is invoked whenever a user chooses to add an assignment related to a specific class registered to their account. The user’s input assignment information is checked for validity, and if confirmed valid, entered into the appropriate tables in the database. If the user-supplied information is invalid, the user will be shown an appropriate error message. Success is indicated to the user by way of a confirmation message. Process 4.0 is detailed in the Level 0 DFD above.

### Process 5.0

The 5.0 process, Record Grade, details how a user’s grade data is entered into the system. The process is invoked whenever a user chooses to add a grade to an existing assignment in their account. The user submits a completion Boolean and a percentage grade, and this information is checked for validity, and if confirmed valid, entered into the assignments table in the database. If the user-supplied information is invalid, the user will be shown an appropriate error message. Success is indicated to the user by way of a confirmation message. Process 5.0 is detailed in the Level 0 DFD above.

### Process 6.0 Level 1 DFD

The 6.0 process, Display Grade Information, details how a user’s request for grade data is processed by the system. This process is utilized automatically any time the user visits a page within the system that displays grade-related date, or explicitly when a user selects a specific class to view grade information on that class. The process involves calculations of the user’s grades based on any weighting scheme that they may have entered into the system. Any failure of this process will generate an appropriate error message for the user. Success is indicated by the user successfully viewing the correct grade information they requested. Process 6.0 is detailed in Figure 9 below.

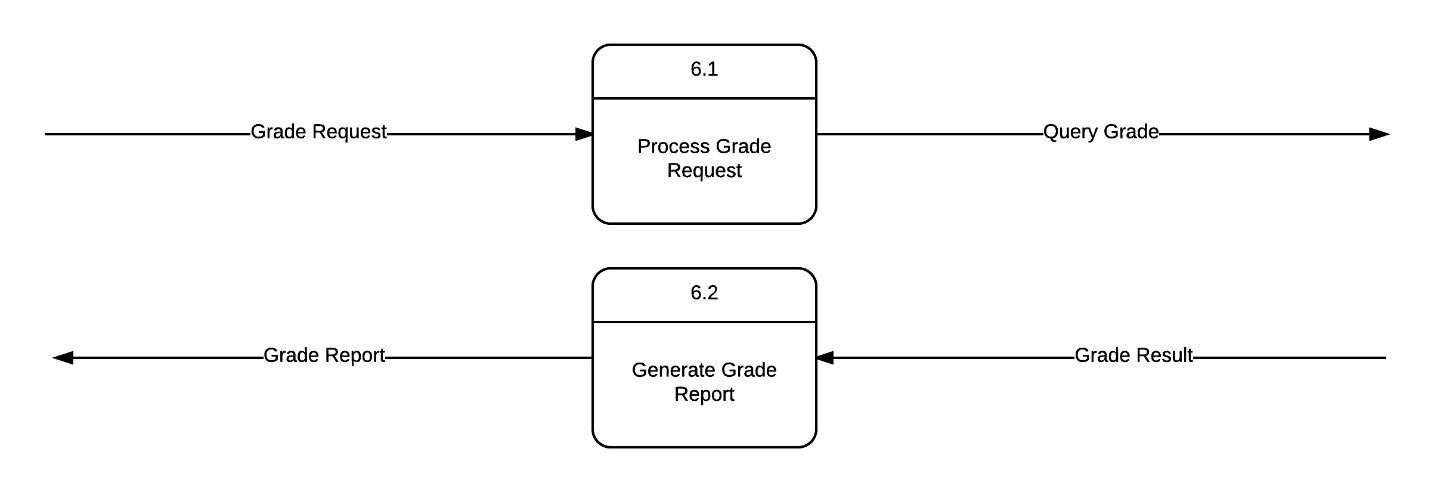


Figure 9 - Process 6.0

### Process 7.0 Level 1 DFD

The 7.0 process, Process Notifications, details how information regarding coming due or past due class obligations is relayed to the user. This process occurs automatically any time a page that contains notification information is opened or refreshed. The process compares the user-entered due dates of all class obligations in the database against the system clock, and displays those that meet notification criteria to the user. Any failure of this process will generate an appropriate error message for the user. Success is indicated by the user successfully viewing the correct system notifications. Process 7.0 is detailed in Figure 10 below.

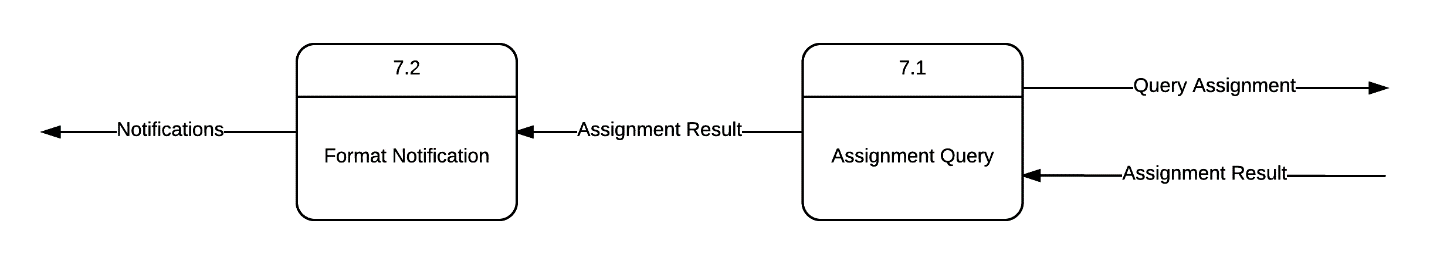


Figure 10 - Process 7.0

### Process 8.0 Level 1 DFD

Process 8.0, Process Payment, details how user-supplied information is translated into a credit-card processor payment request, and how this request is sent to the credit card processor, and how the confirmation or declination from the credit card system is processed by the system. User payment information is taken into the system, checked for validity, and if confirmed valid, packaged into a file for transmittal to the credit card processor. If the user-entered information is invalid, the system will display an appropriate error message to the user. Payment confirmation and declination messages are processed through the system and displayed for the user. Success is indicated if the payment file is created and transmitted to the credit card processor, the appropriate confirmation or declination is received and displayed to the user, and the generation of a receipt for the user (currently unavailable in Alpha release).

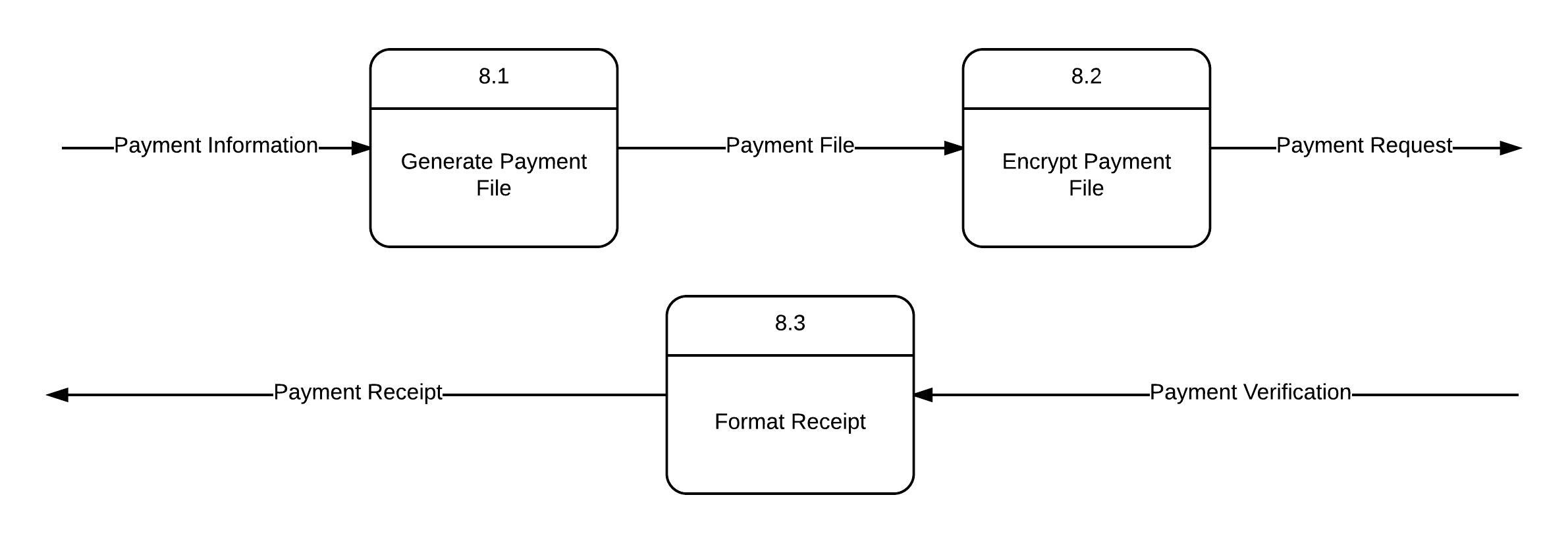


Figure 11 - Process 8.0

# System Database Entity Relationship Diagrams

## Production Version

Figure 12 below details the Class Bytes System database design for use in the production version.

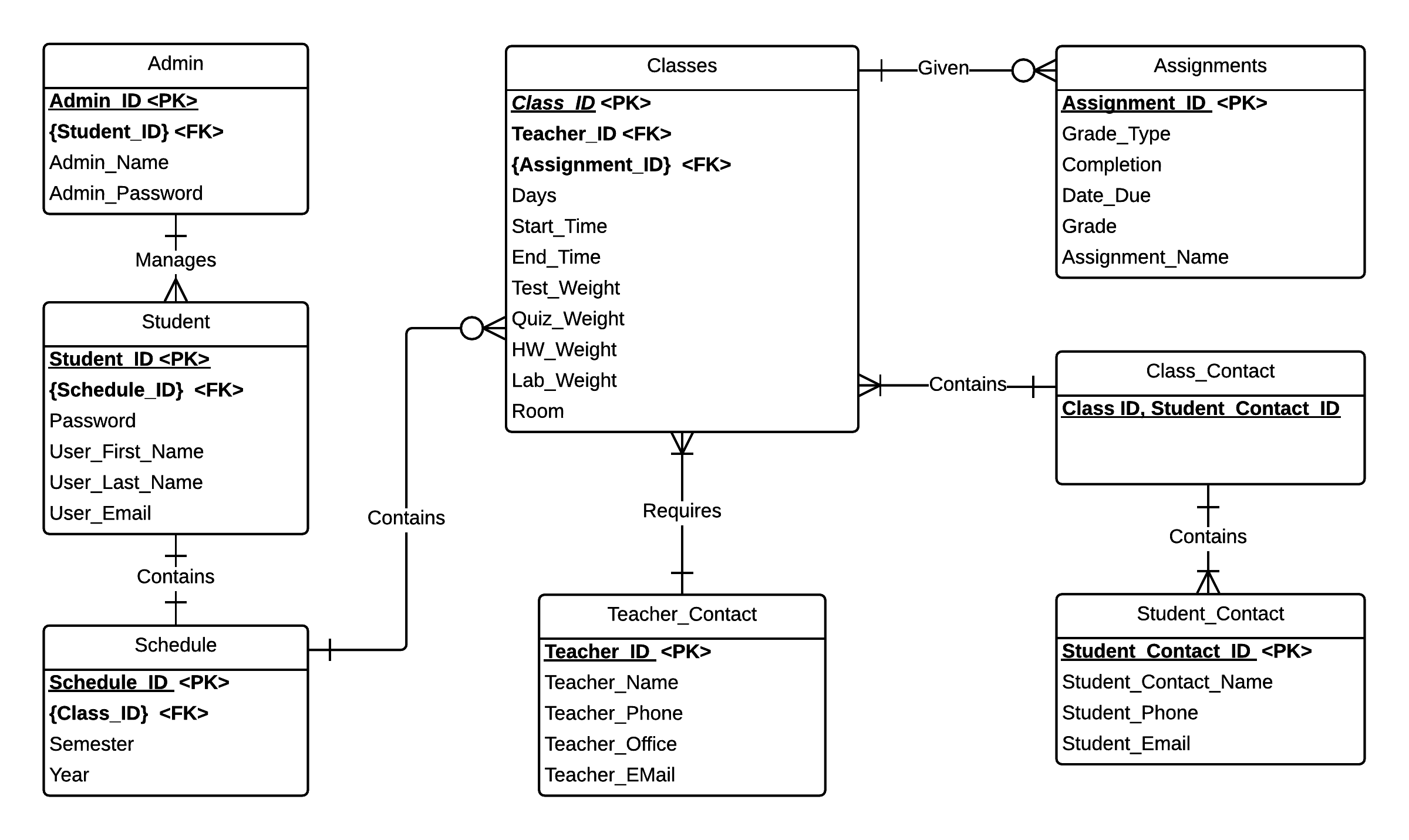


Figure 12 - Production ERD

## Alpha Version

Figure 13 below details the Class Bytes System database design used for the Alpha release of the system. This database design is ONLY for testing purposes, as while it provides the exact same functionality as the production version, the production version will operate with a much higher performance margin. DO NOT use this database design for any deployed Class Bytes System.

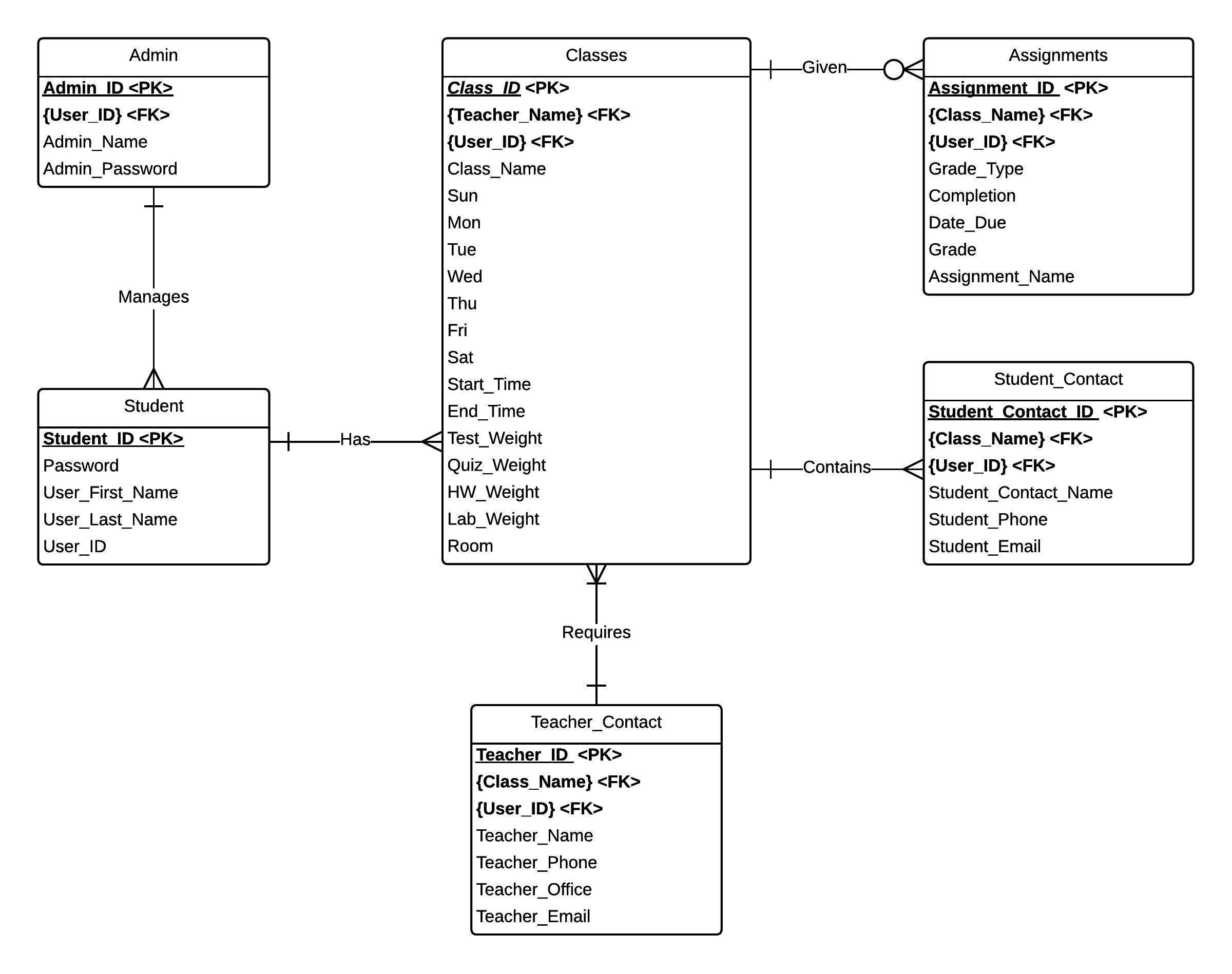


Figure 13 - Alpha Release ERD

# Troubleshooting

Problem: System is not accessible through the internet

Solutions: Confirm proper operation of both web servers

Confirm proper operation of the internal DNS server

Confirm proper operation and continuity of interconnecting hardware

Confirm proper operation of access point

Call “Mike” at Time Warner Internet

Turn system off and on again

Clear your internet browsing cache

Problem: Web front end is accessible, but neither accepts nor returns data

Solutions: Confirm proper operation of the database server

Confirm proper operation of the database server interconnecting hardware

Confirm proper operation of PHP interpreter software on web servers (uncommon)

Confirm validity of HTML / PHP code on web servers (uncommon)

Problem: System is returns errors upon data entry

Solutions: Confirm proper operation of the database server

Confirm proper operation of the database server interconnecting hardware

Confirm database has sufficient drive space available

Confirm proper operation of PHP interpreter software on web servers (uncommon)

Confirm validity of HTML / PHP code on web servers (uncommon)

Problem: User cannot login to system

Solutions: Confirm user is using the correct credentials

Confirm user is present in the database

Confirm proper operation of the database server interconnecting hardware

Confirm proper operation of PHP interpreter software on web servers (uncommon)

Confirm validity of HTML / PHP code on web servers (uncommon)

Problem: Payments are not being posted to operator’s accounts

Solutions: Confirm that the payment module is operation correctly

Confirm that payment information has been correctly configured within the “creditcard.php” file

For any additional problems, or to report system errors or suggest system improvements, please feel free to contact us at 1(800) CLS-BYTS (1(800) 257-2987) Monday thru Friday 9:00 AM to 5:00 PM EST, or anytime by email at [support@classbytes.com](mailto:support@classbytes.com), and we will be happy to assist you.