CC104 FINAL PROJECT

**Documentation:**

1. **INTRODUCTION**

* Welcome to our WMSU Parking Management System project, designed to simplify your parking experience and enhance the efficiency of parking management at Western Mindanao State University (WMSU). As we navigate through the bustling campus and busy streets, we understand the daily challenges of finding a convenient parking spot and managing parking spaces effectively.
* Our project is not just about technology; it's about making your life easier and contributing to a smoother, more enjoyable campus experience. Whether you're a student rushing to class, a faculty member with a busy schedule, or a visitor exploring our vibrant campus, we're here to optimize your parking journey.

1. **OBJECTIVES**

* Data Modeling: Designing database schemas, tables, relationships, and constraints to effectively manage parking-related data such as parking spaces, bookings, payments, and user details.
* Real-Time Updates: Implementing mechanisms for real-time updates and synchronization of parking data to provide accurate and up-to-date information for users and administrators.
* Optimized Query Performance: Fine-tuning database structures, indexing, and query optimization techniques to ensure fast retrieval and processing of parking-related data, optimizing system performance.
* Data Integrity and Security: Implementing measures to maintain data integrity, enforce constraints, and ensure secure storage, access, and transmission of parking system data.
* Scalability and Flexibility: Designing the database system to be scalable, adaptable, and capable of accommodating future expansion, increased data volume, and evolving requirements of the parking system.

1. **SCOPE AND LIMITATIONS**

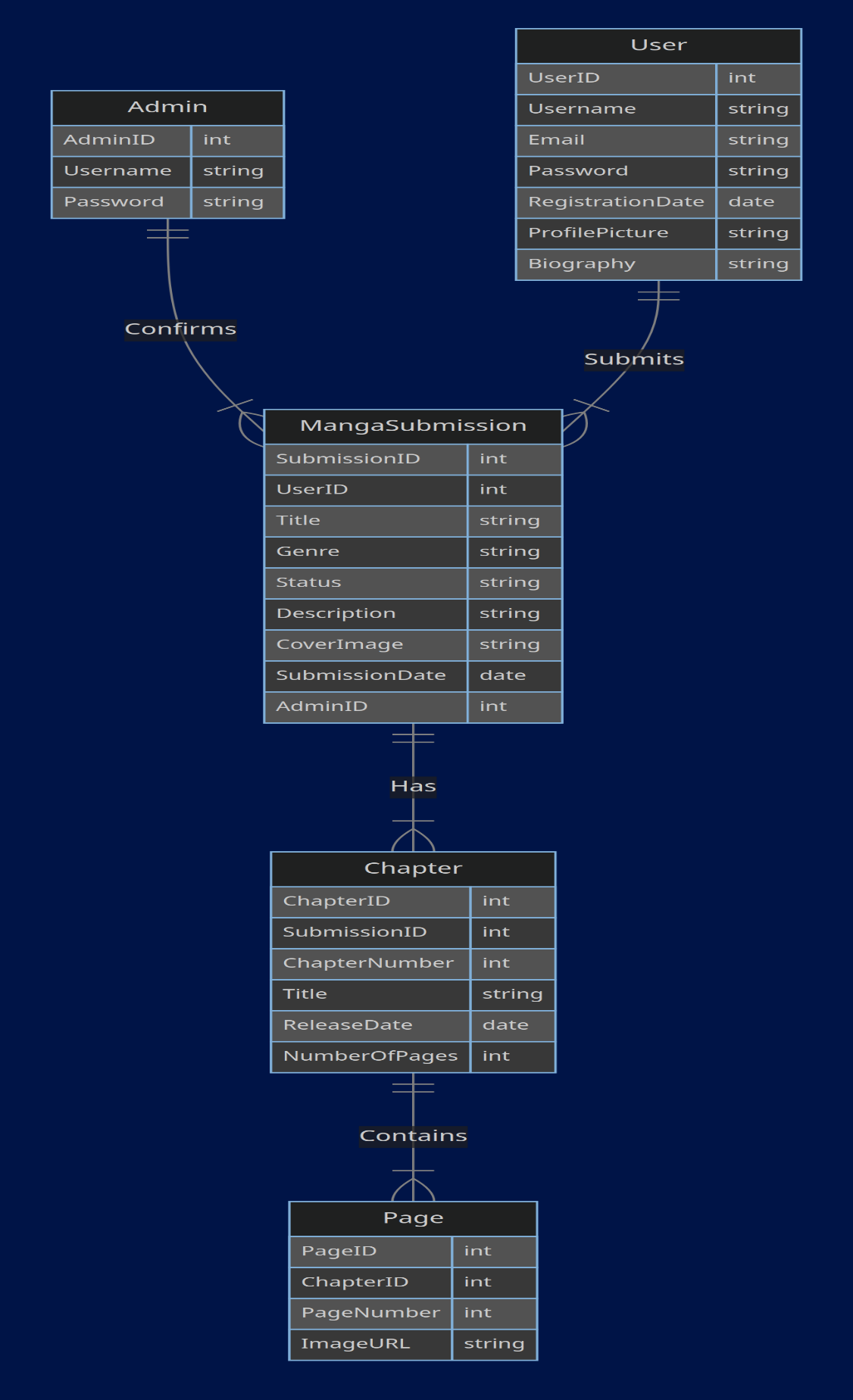
* **SCOPE**
* **Database Design:** The scope of this project includes designing a comprehensive database schema that efficiently manages parking-related data such as parking spaces, bookings, payments, user details, and operational logs.
* **Real-Time Updates:** Implementing mechanisms for real-time updates and synchronization of parking data to provide accurate and up-to-date information for users and administrators.
* **User Interface Integration:** Integrating the database system with a user-friendly interface, including mobile apps and web portals, to facilitate user interactions such as booking parking spaces, making payments, and accessing real-time parking information.
* **Security Measures:** Implementing robust security measures such as encryption, access controls, and audit logs to protect sensitive parking system data and ensure compliance with data protection regulations.
* **LIMITATIONS**
* **Hardware and Infrastructure:** The project does not encompass the procurement or implementation of physical hardware infrastructure such as sensors, gates, or surveillance systems. It focuses solely on the database system design and management aspects.
* **External Integrations:** While the project includes integrating the database system with user interfaces, it does not cover integrations with external systems or third-party services beyond the scope of parking management.
* **Advanced Analytics**: Although reporting and analytics functionalities are included, advanced data analytics techniques such as machine learning algorithms or predictive modeling are beyond the scope of this project.
* **Regulatory Compliance:** While security measures are implemented, the project does not address specific regulatory compliance requirements such as GDPR or PCI DSS compliance, which may require additional measures and certifications.
* **Scale and Performance:** While efforts are made to optimize query performance and scalability, the project does not guarantee optimal performance under extreme load conditions or for very large-scale deployments without further tuning and optimization efforts.

1. **DATABASE DESIGN**

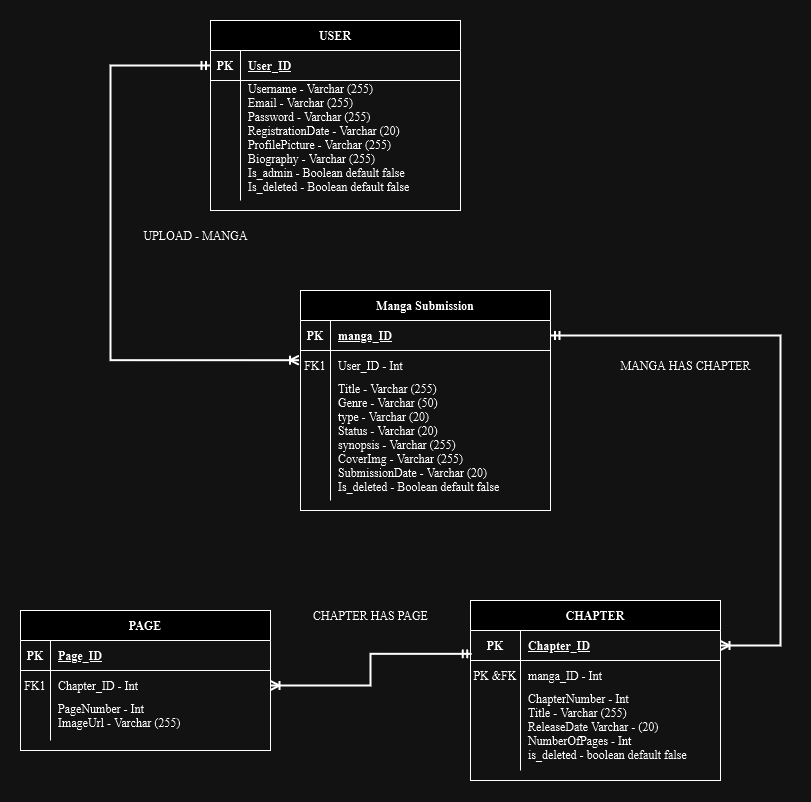
* **Conceptual Model**

**Entities:**

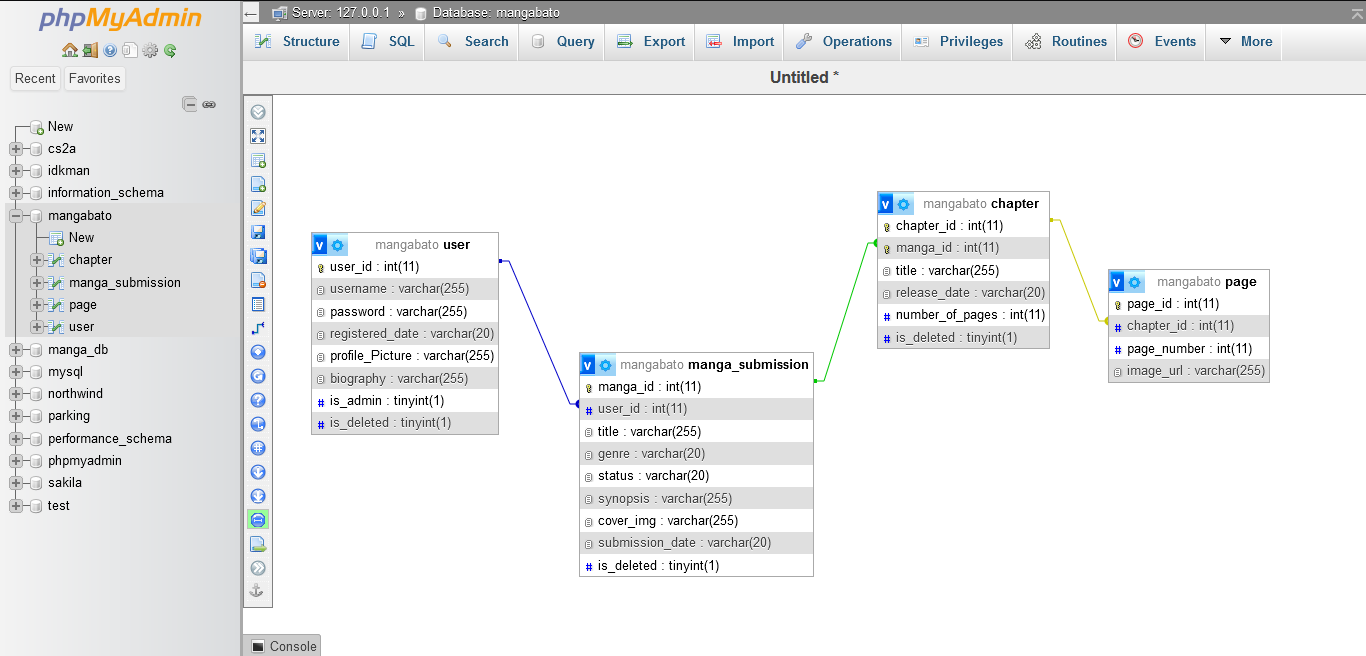
* **Admin Table**: Represents administrators who have access to the system and are responsible for confirming manga submissions.
* **User Table:** Represents registered users who can submit manga submissions and interact with the platform.
* **MangaSubmission Table:** Represents manga submissions made by users. It includes information about the manga, its status, and the admins who confirm the submissions.
* **Chapter Table:** Represents chapters within manga submissions. Each chapter belongs to a specific manga submission.
* **Page Table:** Represents individual pages within chapters. Each page belongs to a specific chapter.



* **Logical Model**



* **User and Manga Submission**
* Relationship: One-to-Many (1:M)
* A user can submit multiple manga submissions, but each manga submission is submitted by only one user.
* If a user is\_admin he can approve the status of the submission.
* **Manga Submission and Chapter**
* Relationship: One-to-Many (1:M)
* A manga submission can have multiple chapters, but each chapter belongs to only one manga submission.
* **Chapter and Page**
* Relationship: One-to-Many (1:M)
* A chapter can have multiple pages, but each page belongs to only one chapter.
* **Physical Design**
* **The designer tab of the DBMS you are using which shows the created tables and their connections**

****

* **Database structure**

**USER**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| User\_id | INT | Primary Key | User\_Id is the primary key of the table User |
| Username | VARCHAR(255) | NOT NULL | It refers to the name of the user |
| Password | VARCHAR(255) | NOT NULL | It refers to the password of the user |
| Email | VARCHAR(255) | NOT NULL | It refers to the email of the user |
| Registered\_date | VARCHAR(20) | NOT NULL | It refers to the date the user created the account |
| Profile\_picture | VARCHAR(255) | NOT NULL | Profile\_picture is a column where the profile picture is being stored |
| Biography | VARCHAR(255) | NULL | It refers to the description or biography of the user that he entered |
| Is\_admin | BOOLEAN | DEFAULT FALSE | Is\_admin is a status verification whether the user is admin. If he is an admin he has the authority to approve a manga submission or delete a user, manga, chapter, etc |
| Is\_deleted | BOOLEAN | DEFAULT FALSE | is\_deleted refers to when a user gets deleted. Instead of deleting the actual data. It flag the data as deleted (true) and filter it out. |

**MANGA SUBMISSION**

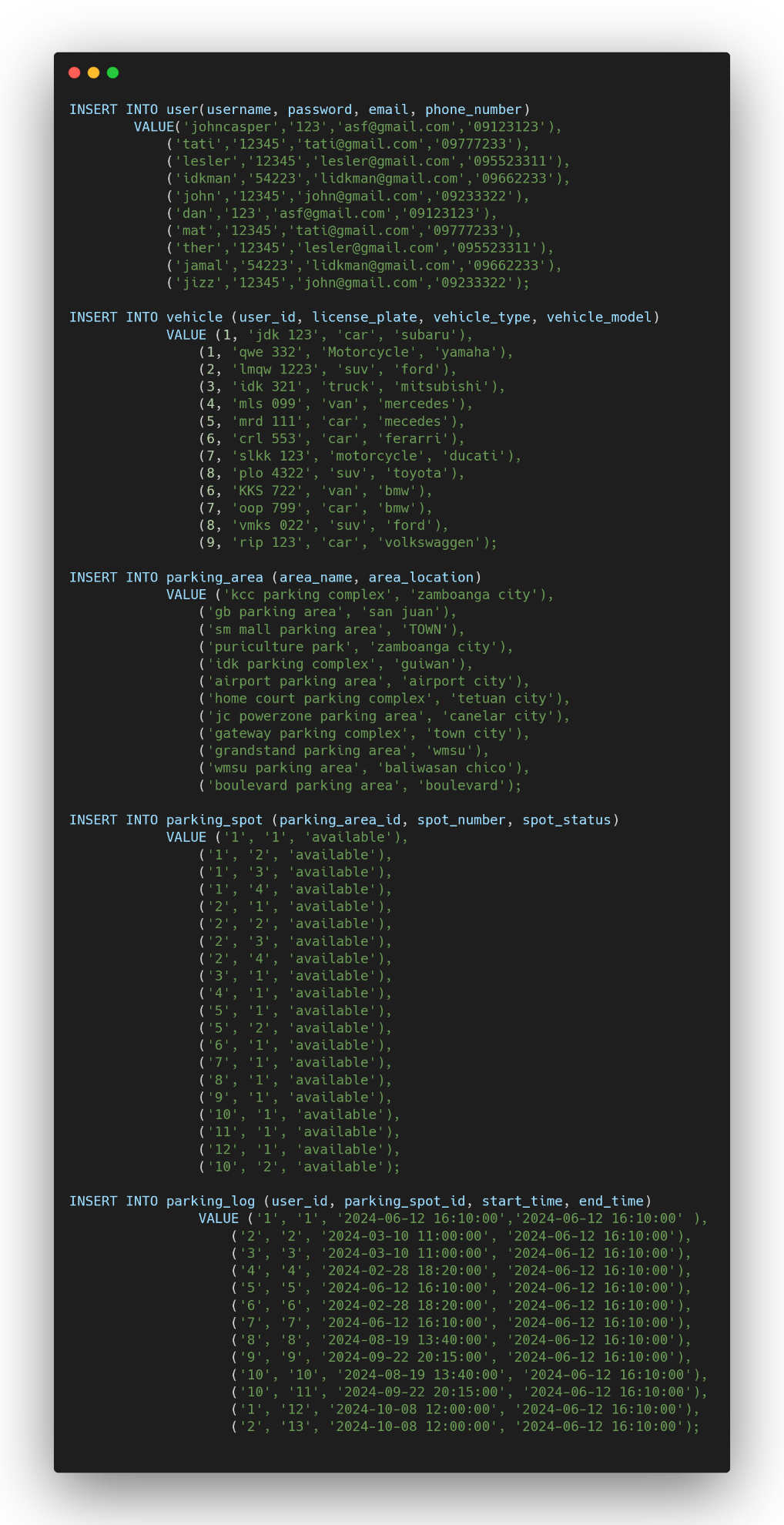
|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| manga\_Id | INT | Primary Key | Manga\_id is the primary key of the manga submission table |
| User\_Id | INT | Foreign key | User\_id is a foreign key associated to User table. It connects the user\_id to the user table |
| Title | VARCHAR(255) | NOT NULL | It refers to the title of the manga |
| Genre | VARCHAR(100) | NOT NULL | It refers to the genre of the manga |
| Type | VARCHAR(20) | NOT NULL | It refers to the type of manga. (Manga, manhwa, manhua) |
| Status | VARCHAR(20) | DEFAULT PENDING | It refers to the status of the manga. Whether it’s been pending, approved, declined by the admin |
| Synopsis | VARCHAR(20) | NOT NULL | It refers to the short description of the manga or the plot of the manga |
| Cover\_img | VARCHAR(255) | NOT NULL | It feres to the cover image of the manga that was uploaded by the user |
| Submission\_date | VARCHAR(20) | NOT NULL | It refers to the date and time of the user submit the manga for submission |
| Is\_deleted | BOOLEAN | DEFAULT FALSE | is\_deleted refers to when a user gets deleted. Instead of deleting the actual data. It flag the data as deleted (true) and filter it out. |

**CHAPTER**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| Chapter\_Id | INT | Composite Key,  Foreign Key | Chapter id is the primary key for the chapter table. |
| Manga\_id | Int | Composite Key,  Foreign Key | Manga\_id and the chapter id is a composite key for making the chapter table unique. It is also a foreign key referencing to the manga table |
| Title | VARCHAR(255) | NULL | It refers to the title of the chapter |
| Release\_date | VARCHAR(20) | NOT NULL | It refers to when a chapter been uploaded and it gets released. |
| Status | VARCHAR(20 | DEFAULT PENDING | Status refers to when the chapter been accepted/ declined or pending. |

**PAGE**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| Page\_id | INT | Primary Key | Page id is the primary key for the page table |
| Chapter\_id | INT | Foreign key | Chapter\_Id refers to the number of the chapter. Referencing the chapter table |
| Page\_number | INT | NOT NULL | It refers to the spot number of the parking spot |
| Image\_url | VARCHAR(255) | NOT NULL | Image\_url is where the image of the page is being stored. |

* **SQL SCRIPT**

