

GROUP ASSIGNMENT

TECHNOLOGY PARK MALAYSIA

**CT042-3-1-IDB**

INTRODUCTION TO DATABASES

HAND OUT DATE: 7th APRIL 2021

HAND IN DATE: 3rd June 2021

WEIGHTAGE: 40%

TEAM MEMBERS:

* KEVIN MATTHEW ADYAN (TP058466)
* MARCELL AGUNG WAHYUDI (TP058650)
* SELVAN NICHOLAS (TP058084)
* WATARU SHINZATO (TP061217)

Table of Contents

[1.0 INTRODUCTION 1](#_Toc73560219)

[2.0 FUNCTIONALITY OF THE CURRENT SYSTEM 1](#_Toc73560220)

[3.0 DATABASE AND DATABASE MANAGEMENT SYSTEM 2](#_Toc73560221)

[3.1 DISADVANTAGES OF FILE-BASED SYSTEM 6](#_Toc73560222)

[3.2 ADVANTAGES OF DATABASE AND DBMS, FUNCTIONS OF DBMS 7](#_Toc73560223)

[3.3 JUSTIFICATION FOR THE DATABASE MANAGEMENT SYSTEM SELECTED 9](#_Toc73560224)

[4.0 BUSINESS RULES & NORMALISATION 10](#_Toc73560225)

[4.1 BUSINESS RULES 10](#_Toc73560226)

[4.2 NORMALISATION 11](#_Toc73560227)

[4.2.1 DEFINITION 11](#_Toc73560228)

[4.2.2 PROCESS INVOLVED 11](#_Toc73560229)

[5.0 ENTITY RELATIONSHIP DIAGRAM 26](#_Toc73560230)

[5.1 CONCEPUTAL DATABASE DESIGN (ERD-ENTITY RELATIONSHIP DIAGRAM) 26](#_Toc73560231)

[5.1.1 STEPS INVOLVED 26](#_Toc73560232)

[5.1.2 ERD 29](#_Toc73560233)

[5.2 LOGICAL DATABASE DESIGN (EERD-ENHANCED ENTITY RELATIONSHIP DIAGRAM) 30](#_Toc73560234)

[5.2.1 STEPS INVOLVED 30](#_Toc73560235)

[5.2.2 EERD 31](#_Toc73560236)

[6.0 CONCLUSIONS 34](#_Toc73560237)

[7.0 WORKLOAD DISTRIBUTION 35](#_Toc73560238)

[8.0 REFERENCES 37](#_Toc73560239)

# INTRODUCTION

In this assignment, the team has to develop a database for managing the flight reservation system for airline companies residing in Malaysia. The term ‘database’ means a collection of objects, encompassing tables, views, functions, stored procedures, triggers, etc. (Tutorials Point, 2019). It is used to store data in a structured way. Database is managed by a complex system called Database Management System (DBMS). A DBMS also provides protection of data from unauthorised users.

To build a database, the first step that needs to be resolved is identifying the business rules and requirements from the scenario given. Then, an Entity Relationship Diagram (ERD) needs to be designed as the database foundation. An ERD consists of entities/objects, each with their attributes/characteristics, with relationships drawn between entities. Next, the ERD should be transformed into an Enhanced ERD (EERD), which will be the base for creating the database schema, the structure of database tables. Finally, Data Definition Language (DDL) and Data Manipulation Language (DML) derived from the Structured Query Language (SQL) can then be used to construct the real database. DDL deals with creating, dropping, or altering the database tables with its attributes, while DML is applied for inserting, deleting, updating the rows/tuples in the database tables.

The assignment team comprises four members, Mr. Kevin Matthew Adyan as the group leader, Mr. Marcell Agung Wahyudi, Mr. Selvan Nicholas, and Mr. Wataru Shinzato as the members of the group.

# FUNCTIONALITY OF THE CURRENT SYSTEM

The team is required to identify the current functionality of the system. In analysing the scenario given regarding the Malaysian Airlines Reservation System, the team found out a few points or conditions that are to be noted. First, the system revolves around four airline companies operating in Malaysia: Core Airways, Echo Airline, Peak Airways, and Spark Airways. Second, the airlines offer domestic routes among six states in Malaysia, including Negeri Sembilan, Pahang, Perak, Sabah, Sarawak, and Wilayah Persekutuan Kuala Lumpur, where in each state, a booking office resides.

In case of the flight travel, it is mandatory that two pilots, which consists of a pilot and co-pilot, be present to control the plane. Each plane could also consist of up to three flight attendants. Additionally, a pilot could also become a senior pilot if he or she has a total flying hour of 20.000.

The airlines accommodate almost every customer that comes from any state in Malaysia. Customers who want to travel are required to book the flights and book more than one flight. Customers who already booked the flights will get a ticket containing information related to the flights, such as flight number, time of departure, class seat, and any critical information required.

To sum it all up, the current Malaysia Airlines Reservation System would start the process when customers book their flight at a booking office in one of the six states in Malaysia. The customers are allowed to book more than one flight if they want to. After the booking process is done, the booking ticket could then be issued. In the booking ticket, information related to the flight is printed, including the flight booking status, which could be either Booked, Cancelled, or Scratched. Following the confirmation of the ticket, the flight then could be conducted with a requirement of two pilots and a maximum of 3 flight attendants.

Implementing this system requires a detailed and centralised database in all the booking offices to run smoothly. In accordance with that, the team would be designing a neat and clear central air-reservation database.

# DATABASE AND DATABASE MANAGEMENT SYSTEM

Database is a collection of data that is managed in a way to suffice specific conditions (DICODING INTERN, 2020). The data stored in a database are interconnected, which makes them easier to manage. Through the management system provided, users can easily find, save, and delete information. In addition, database can also be defined as a system that functions to collect files, archives, or tables connected and stored on various electronic media (DICODING INTERN, 2020).

Databases can be divided into many types (Tutorialspoint, 2020). Based on the location of data, databases can be divided into:

1. Centralised Database

In this database, information is kept at a centralised location, making it so users from various places can retrieve that information.

1. Distributed Database

This database is the exact opposite of the centralised database concept. The data in this database are not located in one place but distributed among various locations in an organisation. These places are interconnected with the help of communication links which helps them to access the distributed data easily.

1. Cloud Database

It is a database that enables users to manage, retrieve and store data via a cloud platform, meaning that it can be accessed over the Internet (Rackspace Technology, n.d.).

Based on the cost of database, databases can be divided into:

1. Commercial Database

This database is a paid variety of massive databases maintained by commercial businesses explicitly designed for customers, which usually consist of large enterprises.

1. Open-source Database

Contrary to commercial database, this database comes with little to no cost and its source code are available to the public. However, support is only available from the community (GeeksforGeeks, 2020).

Based on the cost of database, databases can be divided into:

1. Personal Database

Data in this database are collected and stored on a personal computer. These databases are usually small and can be handled easily. These data are commonly used by similar departments and can be accessed by only a few groups of people.

1. Enterprise Database

This type of database manages large amounts of data. It usually can be used by many users simultaneously, increasing the efficiency and productivity of the organisation (JavaTPoint, 2020).

Based on its structure, databases can be divided into:

1. Relational Database

This type of database has the characteristics of having a collection of tables where information gets stored into a category which has already been defined before. The Structured Query Language (SQL) is the typical user interface (UI) and application program interface (API) for this kind of database.

1. NoSQL Database

It is a database used for large sets of distributed data (Schaefer, n.d.). NoSQL Database typically refers to any non-relational database. NoSQL Database stores relationships, but they store the data differently than relational databases do. For example, NoSQL can store data in document (similar to JSON documents), graph, or key-value format.

1. Object-Oriented Database

This database is a group of object-oriented and relational database (C# Curator, 2019). The data in these databases are represented and stored in the shape of objects.

DataBase Management System or DBMS is a system or software designed specifically to manage databases and perform operations on user-requested data (Guru99, 2019). DBMS is an intermediary between users and databases. To interact with a DBMS, users can use the database language that the DBMS application uses. Database language usually consists of various kinds of instructions formulated in a way that allows DBMS to process them.

There are four types of DBMS models (Guru99, 2019), which are:

1. Hierarchical Model

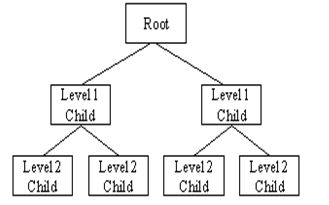


Figure 3.1 Hierarchical Model (Panwar, 2020)

In this DBMS model, data is managed in a formation similar to a tree. Data are stored in a top-down or bottom-up format. In addition, the data are represented as a parent-child relationship, which means that a parent may have many children, but a child may only have one parent.

1. Network Model

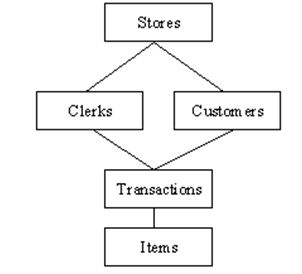


Figure 3.2 Network Model (Panwar, 2020)

This DBMS model enables each child to have more than one parent, hence allowing to fulfil the requirement such as modelling more sophisticated relationships such as many-to-many relationships. This model can be pictured as entities managed in a graph that are able to be reached through different routes.

1. Relational Model

Diagram

Description automatically generated

Figure 3.3 Relational Model (Caseau, 1991)

This model is the most commonly used model since it is one of the simplest to use. The relational model works by normalising data in the columns and rows of the tables.

1. Object-Oriented Model

Diagram

Description automatically generated

Figure 3.4 Object-oriented Model (Caseau, 1991)

In this model, data are stored in the shape of objects. The objects possess some properties and belong to a class.

## DISADVANTAGES OF FILE-BASED SYSTEM

Graphical user interface, diagram

Description automatically generated

Figure 3.5 Comparison between file-based system and database management system (Bédard & Henriques, 2002)

Before the advent of databases, people managed data through the use of file-based systems. A file-based system is a cluster of programs that enables users to retrieve data. Each application inside this system interprets and organises its own data. Consequently, there will be limitations to how data are utilised or moved (Computing Students, n.d.).

The drawbacks of the file-based system make it a legacy system in the database domain in the 21st century. Firstly, the same data can exist and be duplicated in different files, leading to the wastefulness of memory space. Furthermore, the integrity of the data could be questioned. Secondly, as a result of data redundancy, it is also possible that the data are not always consistent. For example, when updating a value, the database in other departments containing that value is not updated, thus making the value outdated in some database. This condition is often defined as data inconsistency (Computing Students, n.d.).

In addition, in terms of data sharing, these data are all distributed over various files and they may be in different formats. For instance, different departments using different programming languages to access data. This makes it difficult when other departments want to access the database and, therefore, results in an inconvenient experience while working with the database.

The file-based system also has what is called the Atomicity Problems (Thakur, 2015). If there is any operation on the database, it has to be atomic. ‘Atomic’ means it has to happen all together or not at all. For instance, updating only parts of the database and leaving the other part not updated could eventually cause a severe error in the system. Hence, the system should either successfully perform the update or fail the whole operation.

The file-based system also produces Concurrent Access Anomalies, which occurs when more than one user has permission to access the data all at once. In this case, it may produce anomalies, and users can also cause data conflicts.

Ultimately, file-based systems haven’t come up with a solution to security problems. It hadn’t possessed sufficient access control to users, which could lead to unauthorised modification and access of data.

## ADVANTAGES OF DATABASE AND DBMS, FUNCTIONS OF DBMS

In the 1960s, through the increased performance of computers and growing usage of computers in businesses, database also started to gain popularity. There are several advantages that it provides compared to the file-based system (Castro, 2018).

Database system is fast and convenient to use. It provides the ability to select data into one organized group quickly. It also enables multi-user to access all database at once, which provides comfortable access to many users.

Another significant advantage of database is its data security. With the use of database language, the database system has been made safely. These databases can only be accessed by parties who have permission to access the database. In addition, it provides a centralised data control, which means the database doesn’t need more than one server. One centralised server is enough to store data that can be accessed by many users.

As a result, by using database, the maintenance cost can be cut down further as having only one centralised database is enough for a big company that needs to collect data concisely. This makes companies save money since they don’t need different places to store data. Moreover, databases allow even small branches in a remote area to access the central data.

Databases are managed with the help of DBMS. DBMS provides a diverse method to store and access data. It also possesses robust functions to store and access data in an efficient way. DBMS servers also has the ability to efficiently balance the demand of multiple application accessing the same data using a load-balancer. Moreover, it offers data integrity, consistency, and security (Castro, 2018). DBMS provides precise and consistent data in the database. Since there’s no data redundancy in DBMS, the data is very consistent. Changes needed to be done in the database will be reflected to all users accessing the database. In terms of security, only authorised users have permission to access the database. In addition, DBMS has an automatic backup and recovery system so users don’t need to concerned about the data in case of a crash or system failure.

DBMS creates and manages complex formats required for data storage. DBMS will help transform the form of data that the user entered into the required data structures for presentation. DBMS supplies data access through query languages (Database Access Languages). Most DBMS supports Structured Query Language (SQL). To access the database in the servers, Application Programming Interfaces can be utilised. This communication interface enables end users to access the database using various web browsers (THIRU, n.d.).

DBMS also enforces user security and data privacy. Not only that, it also provides multi-user access control. DBMS is devised with advanced algorithms to make sure many users can access the database in real time without corrupting the database integrity.

Furthermore, DBMS enables the searching of required data component structures and relationships using data dictionaries, which could facilitate users and remove the hassle of having to code complex relationships in each program.

DBMS also comes with a backup and recovery system which is critical to preserving database’s integrity, particularly in big companies having a lot of data which is vital to the operation of the company. That being said, DBMS enforces integrity rules, meaning it strives to maximize data consistency and minimize data redundancy.

## JUSTIFICATION FOR THE DATABASE MANAGEMENT SYSTEM SELECTED

Figure 3.6: Microsoft SQL Server Management Studio Screenshot (Ghanayem, 2019)

The DBMS used to manage the assignment’s database is Microsoft SQL Server Management Studio (SSMS) 18. SSMS is an Integrated Development Environment (IDE) that provides means of connecting with Microsoft SQL Server as the database engine (Erkec, 2020). However, in 2021, it is only compatible to be run on 64-bit Windows 8.1 and 64-bit Windows 10 OS.

SSMS is a type of relational database management system. Its latest version, version 18, was launched in 2019. A significant improvement in this version is that it is now half the size of its predecessor, version 17 (Nethi, 2019), leaving more storage space for users. SSMS 18 also fully supports the latest SQL Server version 2019. Regardless of versions, SSMS provides backward compatibility up to SQL Server 2008 version, which reduces hassle when there is no other way but to work with older SQL server versions.

Microsoft SSMS is free to use so it is very suitable for database learners. However, its database engine, MS SQL server, is free only for development purposes, but paid when it is used for commercial purposes. SSMS can be downloaded from the Microsoft documentation website, docs.microsoft.com. It has a simple installation process with just a few clicks. The setup file will then take some time to install some packages included in the download package. However, as SSMS is just an IDE, users should also take note that the required database engine or services should already be installed and prepared before using SSMS.

SSMS 18 comes with a simple and friendly UI/UX. It provides an object explorer section on the left to help users quickly find and manage SQL server instances. It also delivers a lot of object and custom SQL templates which is helpful for users having common tasks. Additionally, SSMS also provides database design tools for users to design diagrams, tables, and queries. It has a smart query editor as well, assisted by IntelliSense, which could increase scripting efficiency by aiding users in scripting, such as pointing on errors. Not only that, SSMS made scripting and querying easy as different keyword types are highlighted and coloured differently.

Overall, SSMS is a robust DBMS tool, having been developed continuously by Microsoft since 2005 (Talmage & Delaney, 2005), offering various tools and features to support developers in building efficient, secure, and functional databases.

# BUSINESS RULES & NORMALISATION

## BUSINESS RULES

Business rules are the criteria to make a decision in business by an organisation or individual. It is an important component in all businesses to make sure the business structure and order are kept intact. Furthermore, it is beneficial for identifying entities and relationship in designing databases.

From the scenario, there are several rules that can be extracted. First, in the second paragraph of the scenario given, it is stated that the flight crew for every flight should consist of two pilots and at most three flight attendants. Additionally, the crew is assumed to be able to serve in more than one flight.

Next, in the third paragraph, the customer’s attribute is discussed. A customer can have zero or more house phone numbers and unique email addresses. In booking a flight, they should be able to book more than one flight.

Below is the list of business rules found from the scenario:

1. Each flight travel must have two pilot.
2. Each flight travel must have at least one and up to three flight attendants.
3. Each pilot may fly one or more flights.
4. Each customer has zero or more house phone numbers.
5. Each customer has zero or more email addresses.
6. Each customer may book one or more flights.
7. Each email address is used only by one customer.

## NORMALISATION

### DEFINITION

Normalisation is a process, which transforms any complex database table into smaller management tables. Normalisation is required to ensure that the table stores related data only, not duplicated data. It is performed to ultimately build a database free of redundancy.

### PROCESS INVOLVED

The normalisation process has 3 phases:

Phase 1: 1NF-First Normal Form

Rule: Remove all repeating groups of attributes found in the complex database table (0NF / unnormalised database table). The result will be multiple database tables.

Phase 2: 2NF-Second Normal Form

Rule: Remove all partial dependencies if a non-primary key attribute found in the 1NF is not entirely dependent on its primary key.

Phase 3: 3NF-Second Normal Form

Rule: Remove all transitive dependencies from the 2NF (2ND Normal Form) or from the tables which are already in the 2NF. Transitive dependencies are formed when a value is indirectly dependent to another value in that table.

**Below are the normalised tables:**

1. AIRLINE\_T
2. BOOKING\_OFFICE\_T
3. FLIGHT\_T
4. PILOT\_T
5. FLIGHT\_ATTENDANT\_T
6. CUSTOMER\_T
7. FLIGHT\_BOOKING\_T
8. ATTENDANT\_SCHEDULE\_T
9. PILOT\_SCHEDULE\_T
10. CUST\_EMAIL\_T
11. CUST\_HPHONE\_NUM\_T
12. CUST\_BOOKING\_PLACE\_T

**AIRLINE\_T (air\_flight\_num\_pk,** air\_company**)**

**PRIMARY KEY: air\_flight\_num\_pk**

|  |  |
| --- | --- |
| Name of Table: | **AIRLINE\_T** |
| Primary Key: | **air\_flight\_num\_pk** |
| **air\_flight\_num\_pk** | air\_company |
| EA1709 | Echo Airline |
| SA1865 | Spark Airways |
| PA2098 | Peak Airways |
| CA8760 | Core Airways |
| AA1100 | Alpha Airways |
| BA1201 | Beta Airline |
| CA1351 | Charlie Airways |
| DA1442 | Delta Airways |
| FA1666 | Foxtrot Airline |
| GA1771 | Garuda Airline |

**BOOKING\_OFFICE\_T (bo\_id\_pk,** bo\_state, bo\_city, bo\_street, bo\_postcode, bo\_phone\_num**)**

**PRIMARY KEY: bo\_id\_pk**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Name of Table: | BOOKING\_OFFICE\_T | | | | |
| Primary Key: | bo\_id\_pk | | | | |
| **bo\_id\_pk** | bo\_state | bo\_city | bo\_street | bo\_postcode | bo\_phone\_num |
| PRK6868 | Perak | Ipoh | 143 Leboh Perajurit 3 Garden East | 31400 | 0556375098 |
| NSN6969 | Negeri Sembilan | Seremban | No. 8 Millennia Business Centre Jln Tan Sri Manickavasagam | 70200 | 0626989333 |
| PHG7070 | Pahang | Kuantan | B230 1St Floor Jln Air Putih | 25300 | 052681214 |
| SBH7171 | Sabah | Kota Kinabalu | 1 Wisma New Far East Taman Far East | 88300 | 052822931149 |
| SRW7272 | Sarawak | Kuching | 1 Jalan Padungan Utara | 93100 | 0622247778 |
| PKL7373 | Wilayah Persekutuan Kuala Lumpur | Kuala Lumpur | 12 Lorong Medan Tuanku Satu | 50300 | 0362033984 |
| KDH6969 | Kedah | Kuala Kedah | 14 Lorong Air Masin | 35000 | 08461345144 |
| TRG3210 | Terengganu | Kuala Terengganu | No. 14 Jalan Sultan Ismail | 14500 | 08216549211 |
| KLT0590 | Kelantan | Kota Bharu | No 69. Jalan Dusun Raja | 46000 | 08124657981 |
| PNG1340 | Pulau Pinang | Georgetown | No. 11 Lbh Muntri | 10200 | 082111110808 |

**FLIGHT\_T (fli\_num\_pk,** fli\_business\_seat, fli\_economy\_seat, fli\_capacity, fli\_departure\_time, fli\_arrival\_time, fli\_departure\_date, fli\_arrival\_date, fli\_origin, fli\_destination, air\_flight\_num\_pk**)**

**PRIMARY KEY: fli\_num\_pk**

**FOREIGN KEY:** air\_flight\_num\_pk

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Name of Table: | FLIGHT\_T | | | | | | | | | |
| Primary Key: | fli\_num\_pk | | | | | | | | | |
| Foreign Key: | air\_flight\_num\_pk | | | | | | | | | |
| **fli\_num\_pk** | fli\_business\_seat | fli\_economy\_seat | fli\_capacity | fli\_departure\_date | fli\_departure\_time | fli\_arrival\_date | fli\_arrival\_time | fli\_origin | fli\_destination | air\_flight\_num\_pk |
| 1023 | 38 | 250 | 288 | 2021-06-19 | 10:45 | 2021-06-19 | 11:20 | Kuala Lumpur | Sarawak | EA1709 |
| 1134 | 20 | 160 | 180 | 2021-06-28 | 12:55 | 2021-06-28 | 14:00 | Kuala Lumpur | Sabah | SA1865 |
| 1454 | 34 | 212 | 246 | 2021-06-30 | 14:30 | 2021-06-30 | 16:30 | Negeri Sembilan | Perak | PA2098 |
| 1256 | 16 | 170 | 186 | 2021-07-10 | 18:50 | 2021-07-10 | 19:30 | Kuala Lumpur | Negeri Sembilan | CA8760 |
| 1324 | 15 | 180 | 195 | 2021-05-22 | 17:30 | 2021-05-22 | 18:30 | Sarawak | Kuala Lumpur | AA1100 |
| 1242 | 31 | 217 | 248 | 2021-06-12 | 16:30 | 2021-06-12 | 17:30 | Sabah | Kuala Lumpur | BA1201 |
| 2157 | 24 | 198 | 222 | 2021-05-24 | 17:00 | 2021-05-24 | 18:45 | Pahang | Kuala Lumpur | CA1351 |
| 1214 | 26 | 196 | 222 | 2021-07-30 | 20:45 | 2021-07-30 | 22:45 | Kuala Lumpur | Pahang | DA1442 |
| 1432 | 16 | 160 | 176 | 2021-08-09 | 22:00 | 2021-08-09 | 23:45 | Perak | Kuala Lumpur | FA1666 |
| 1953 | 14 | 179 | 193 | 2021-05-27 | 17:30 | 2021-05-27 | 19:30 | Negeri Sembilan | Sabah | GA1771 |

**PILOT\_T (pil\_id\_pk**, pil\_staff\_fname, pil\_staff\_lname, pil\_gender, pil\_dob, pil\_age, pil\_year\_of\_experience, pil\_position, pil\_flying\_hours, pil\_salary, air\_flight\_num\_pk**)**

**PRIMARY KEY: pil\_id\_pk**

**FOREIGN KEY:** air\_flight\_num\_pk

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Name of Table: | PILOT\_T | | | | | | | | | |
| Primary Key: | pil\_id\_pk | | | | | | | | | |
| Foreign Key: | air\_flight\_num\_pk | | | | | | | | | |
| **pil\_id\_pk** | pil\_staff\_fname | pil\_staff\_lname | pil\_gender | pil\_dob | pil\_age | pil\_experience | pil\_position | pil\_flying \_hours | pil\_salary | air\_flight\_num\_pk |
| 10-3765 | Alice | Smith | F | 1995-07-27 | 25 | 4 | CC | 2,021 | 250,000 | EA1709 |
| 05-4057 | Bernard | Wilson | M | 1977-01-01 | 44 | 19 | C | 16,810 | 900,000 | SA1865 |
| 11-1019 | Charlie | Miller | M | 1966-04-08 | 55 | 30 | C | 25,789 | 1,200,000 | PA2098 |
| 07-0870 | Dan | Johnson | M | 1984-08-05 | 36 | 17 | CC | 15,968 | 400,000 | CA8760 |
| 01-6969 | Macel | Gunga | M | 1981-10-13 | 39 | 17 | C | 16,312 | 900,000 | AA1100 |
| 02-7777 | Vinke | Thewma | M | 1975-04-02 | 46 | 23 | CC | 8,012 | 350,000 | BA1201 |
| 12-4231 | Sukimi | Shidani | F | 1980-08-17 | 40 | 18 | CC | 9,901 | 375,000 | CA8760 |
| 51-3241 | Vansel | Colasni | F | 1989-12-13 | 31 | 11 | C | 10,231 | 760,000 | DA1442 |
| 23-6523 | Nipip | Wayanseti | M | 1991-03-22 | 30 | 10 | C | 14,023 | 800,000 | FA1666 |
| 21-6924 | Yanri | Tinmar | F | 1978-07-30 | 43 | 20 | CC | 3,502 | 275,000 | GA1771 |

**FLIGHT\_ATTENDANT\_T (fa\_id\_pk,** fa\_fname, fa\_lname, fa\_gender, fa\_position, fa\_salary, fa\_phone\_num, fa\_street, fa\_city, fa\_state, fa\_postcode, fa\_country, air\_flight\_num\_pk**)**

**PRIMARY KEY : fa\_id\_pk**

**FOREIGN KEY:** air\_flight\_num\_pk

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Name of Table: | FLIGHT\_ATTENDANT\_T | | | | | | | | | | | |
| Primary Key: | fa\_id\_pk | | | | | | | | | | | |
| Foreign Key: | air\_flight\_num\_pk | | | | | | | | | | | |
| **fa\_id\_pk** | fa\_fname | fa\_lname | fa\_gender | fa\_position | fa\_salary | fa\_phone\_num | fa\_street | fa\_city | fa\_state | fa\_postcode | fa\_country | air\_flight\_num\_pk |
| 1-235 | Catherine | Smith | F | FA | 100,000 | 6024154233 | Jalan Teknologi 5 | Kuala Lumpur | Wilayah Persekutuan Kuala Lumpur | 57000 | Malaysia | EA1709 |
| 0-564 | Ann | Ross | F | CP | 250,000 | 6026933777 | Jalan Tanjung Chat 4 | Kota Bharu | Kelantan | 15400 | Malaysia | SA1865 |
| 1-333 | Mike | Hunt | M | FA | 150,000 | 6223897534 | Jalan Benteng Jaya 16 | Kota Tangerang | Banten | 15111 | Indonesia | PA2098 |
| 0-565 | John | Taylor | M | FP | 200,000 | 6521245618 | 678A Jurong West Street 64 | Singapore | Singapore | 641678 | Singapore | CA8760 |
| 9-325 | Vinke | Wansei | F | CP | 250,000 | 6519452014 | Jalan Dermaga 69 | Kota Samarinda | East Kalimantan | 75242 | Indonesia | AA1100 |
| 4-562 | Kelmic | Kom | M | SP | 225,000 | 6598217168 | Jalan Setiabudhi 8 | Kota Bandung | Jawa Barat | 40141 | Indonesia | BA1201 |
| 9-936 | Nathanyo | Cenvi | F | FA | 150,000 | 62821450142 | Grusiva Niyamasubar St. | City of Seria | Belait | 2733 | Brunei | CA1351 |
| 2-252 | Rihen | Kelmai | M | FA | 125,000 | 62965748351 | Jalan Soekarno Hatta 61 | Kota Balikpapan | East Kalimantan | 76214 | Indonesia | DA1442 |
| 6-693 | Terpe | Yenbri | M | FP | 190,000 | 65789451345 | Jalan Bung Tomo 4 | Kota Surabaya | East Java | 60245 | Indonesia | FA1666 |
| 6-969 | Ranzah | Nihaf | F | FA | 100,000 | 62212691611 | Jalan Shonda Makurmpad | Kota Malang | East Java | 65113 | Indonesia | GA1771 |

**CUSTOMER\_T (cust\_id\_pk**, cust\_fname, cust\_lname, cust\_street, cust\_city, cust\_state, cust\_postcode, cust\_country, cust\_hphone\_num, cust\_gender**)**

**PRIMARY KEY: cust\_id\_pk**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Name of Table: | CUSTOMER\_T | | | | | | | |
| Primary Key: | cust\_id\_pk | | | | | | | |
| **cust\_id\_pk** | cust\_fname | cust\_lname | cust\_street | cust\_city | cust\_state | cust\_postcode | cust\_country | cust\_gender |
| 550106-12-5821 | Dig | Beck | Seksyen Bb 11, Bukit Beruntung | Rawang | Selangor | 48300 | Malaysia | M |
| 930906-14-7043 | Ben | Dover | No. 8 Jln Air Panas | Kuala Lumpur | Wilayah Persekutuan Kuala Lumpur | 53200 | Malaysia | M |
| 941130-07-5153 | Icewallow | Kam | Jalan Setiabudi 10 | Malang | Jawa Timur | 65111 | Indonesia | F |
| 610522-10-5378 | Ligma | Bllas | No. 12 Jalan Bersatu (13/4) | Petaling Jaya | Selangor | 46200 | Brunei | F |
| 123400-56-9876 | Ching | Chong | Jiu Wei Cun 81hao | Sha Tian | Guang Dong | 523000 | China | F |
| 581235-61-2456 | Dhason | Padnakunar | Lot 3706 Jln Tengah | Bandar Seri Begawan | Brunei and Muara | 8411 | Brunei | M |
| 615493-59-7984 | Sivaguru | Kael | No. 3 Jln Sek Keb | Bukit Payong | Terengganu | 21400 | Malaysia | M |
| 923497-36-5951 | Tanvir | Subarnamiyn | 20-1 A Jln Pjs 3/36 Taman Sri Manja | Petaling Jaya | Selangor | 58200 | Malaysia | M |
| 164973-59-1576 | Usman | Mailis | No. 10 Kampung | Bota | Perak | 32600 | Malaysia | M |
| 985632-41-6932 | Upin | Ipin | 11-33 Jalan Ibrahim | Batu Pahat | Johor | 83000 | Malaysia | M |

**FLIGHT\_BOOKING\_T (fb\_book\_num\_pk**, **fli\_num\_pk**, **cust\_id\_pk**, **bo\_id\_pk**, fb\_book\_date, fb\_class, fb\_price, fb\_status, fb\_paid\_amount, fb\_outstanding**)**

**PRIMARY KEY: (fb\_book\_num\_pk**, **fli\_num\_pk**, **cust\_id,\_pk**, **bo\_id\_pk)**

**FOREIGN KEY:** cust\_id\_pk REFERENCES CUSTOMER\_T (cust\_id\_pk), fli\_num\_pk REFERENCES FLIGHT\_T (fli\_num\_pk), bo\_id\_pk REFERENCES BOOKING\_OFFICE\_T (bo\_id\_pk)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Name of Table: | FLIGHT\_BOOKING\_T | | | | | | | | |
| Primary Key: | (fb\_book\_num\_pk,fli\_num\_pk, cust\_id\_pk, bo\_id\_pk) | | | | | | | | |
| Foreign Key: | cust\_id\_pk REFERENCES CUSTOMER\_T (cust\_id\_pk), fli\_num\_pk REFERENCES FLIGHT\_T (fli\_num\_pk), bo\_id\_pk REFERENCES BOOKING\_OFFICE\_T (bo\_id\_pk) | | | | | | | | |
| **fb\_book\_num\_pk** | **fli\_num\_pk** | **cust\_id\_pk** | **bo\_id\_pk** | fb\_book\_date | fb\_class | fb\_price | fb\_status | fb\_paid\_amount | fb\_outstanding |
| 0456327110 | 1023 | 550106-12-5821 | PKL7373 | 2021-04-17 | B | 600 | S | 100 | 400 |
| 6902341890 | 1134 | 930906-14-7043 | PKL7373 | 2020-08-09 | E | 220 | B | 200 | 20 |
| 9736440028 | 1454 | 941130-07-5153 | NSN6969 | 2020-08-16 | E | 230 | C | NULL | NULL |
| 8969912362 | 1256 | 610522-10-5378 | PKL7373 | 2020-12-21 | E | 170 | B | 170 | NULL |
| 6124953478 | 1324 | 123400-56-9876 | SBH7171 | 2020-12-14 | B | 1000 | B | 900 | 100 |
| 8453125549 | 1242 | 581235-61-2456 | SRW7272 | 2021-02-06 | B | 575 | B | 550 | 25 |
| 1324875613 | 1432 | 615493-59-7984 | PHG7070 | 2021-01-09 | E | 235 | C | NULL | NULL |
| 4950351204 | 2157 | 923497-36-5951 | PKL7373 | 2021-04-12 | E | 220 | S | NULL | NULL |
| 0326592175 | 1953 | 164973-59-1576 | PRK6868 | 2020-10-13 | E | 215 | S | NULL | NULL |
| 1203562047 | 1214 | 985632-41-6932 | NSN6969 | 2020-12-19 | B | 240 | C | NULL | NULL |

**CUSTOMER\_EMAIL\_T (cust\_id\_pk, ce\_id\_pk,** ce\_email\_address**)**

**PRIMARY KEY: (cust\_id\_pk, ce\_id\_pk)**

**FOREIGN KEY:** cust\_id\_pk REFERENCES CUSTOMER\_T(cust\_id\_pk)

|  |  |  |
| --- | --- | --- |
| Name of Table: | CUSTOMER\_EMAIL\_T | |
| Primary Key: | (cust\_id\_pk, ce\_id\_pk) | |
| Foreign Key: | cust\_id\_pk REFERENCES CUSTOMER\_T(cust\_id\_pk) | |
| **cust\_id\_pk** | **ce\_id\_pk** | ce\_email\_address |
| 550106-12-5821 | 1 | [digbeck@hotmail.com](mailto:digbeck@hotmail.com) |
| 930906-14-7043 | 2 | [bendover69@yahoo.com](mailto:bendover69@yahoo.com) |
| 941130-07-5153 | 3 | [wallowicekam123@gmail.com](mailto:wallowicekam123@gmail.com) |
| 610522-10-5378 | 4 | [ligmaballs@hotmail.com](mailto:ligmaballs@hotmail.com) |
| 123400-56-9876 | 5 | [chingchongchang@yahoo.com](mailto:chingchongchang@yahoo.com) |
| 581235-61-2456 | 6 | [dhason@apu.edu.my](mailto:dhason@apu.edu.my) |
| 615493-59-7984 | 7 | [tanvir@apu.edu.my](mailto:tanvir@apu.edu.my) |
| 923497-36-5951 | 8 | [sivaguru123@yahoo.com](mailto:sivaguru123@yahoo.com) |
| 164973-59-1576 | 9 | [usmanthegreat@yahoo.com](mailto:usmanthegreat@yahoo.com) |
| 985632-41-6932 | 10 | [xXDemonLordXx@yahoo.com](mailto:xXDemonLordXx@yahoo.com) |

**CUST\_HPHONE\_NUM\_T (cust\_id\_pk, ch\_id\_pk,** ch\_hphone\_num**)**

**PRIMARY KEY : (cust\_id\_pk, ch\_id\_pk)**

**FOREIGN KEY:** cust\_id\_pk REFERENCES CUSTOMER\_T(cust\_id\_pk)

|  |  |  |
| --- | --- | --- |
| Name of Table: | **CUST\_HPHONE\_NUM\_T** | |
| Primary Key: | (cust\_id\_pk, ch\_id\_pk) | |
| Foreign Key: | cust\_id\_pk REFERENCES CUSTOMER\_T(cust\_id\_pk) | |
| **cust\_id\_pk** | **ch\_id\_pk** | ch\_hphone\_num |
| 550106-12-5821 | 1 | 6084901516 |
| 930906-14-7043 | 2 | 60249241562 |
| 941130-07-5153 | 3 | 62498456852 |
| 610522-10-5378 | 4 | 67180200120 |
| 123400-56-9876 | 5 | 8156562001 |
| 581235-61-2456 | 6 | 67199998877 |
| 615493-59-7984 | 7 | 60164624389 |
| 923497-36-5951 | 8 | 608214944201 |
| 164973-59-1576 | 9 | 60121249895 |
| 985632-41-6932 | 10 | 60333321644 |

**ATTENDANT\_SCHEDULE\_T (fa\_id\_pk, fli\_num\_pk)**

**PRIMARY KEY: (fa\_id\_pk, fli\_num\_pk)**

**FOREIGN KEY:** fa\_id\_pk REFERENCES FLIGHT\_ATTENDANT\_T (fa\_id\_pk), fli\_num\_pk REFERENCES FLIGHT\_T (fli\_num\_pk)

|  |  |
| --- | --- |
| Name of Table: | **ATTENDANT\_SCHEDULE\_T** |
| Primary Key: | (fa\_id\_pk, fli\_num\_pk) |
| Foreign Key: | fa\_id\_pk REFERENCES FLIGHT\_ATTENDANT\_T (fa\_id\_pk), fli\_num\_pk REFERENCES FLIGHT\_T (fli\_num\_pk) |
| **fa\_id\_pk** | **fli\_num\_pk** |
| 1-235 | 1023 |
| 0-564 | 1134 |
| 1-333 | 1454 |
| 0-565 | 1256 |
| 9-325 | 1324 |
| 4-562 | 1242 |
| 9-936 | 2157 |
| 2-252 | 1214 |
| 6-693 | 1432 |
| 6-969 | 1953 |

**PILOT\_SCHEDULE\_T (pil\_id\_pk, fli\_num\_pk)**

**PRIMARY KEY: (pil\_id\_pk, fli\_num\_pk)**

**FOREIGN KEY:** pil\_id\_pk REFERENCES PILOT\_T (pil\_id\_pk), fli\_num\_pk REFERENCES FLIGHT\_T (fli\_num\_pk)

|  |  |
| --- | --- |
| Name of Table: | **PILOT\_SCHEDULE\_T** |
| Primary Key: | (pil\_id\_pk, fli\_num\_pk) |
| Foreign Key: | pil\_id\_pk REFERENCES PILOT\_T (pil\_id\_pk), fli\_num\_pk REFERENCES FLIGHT\_T (fli\_num\_pk) |
| **pil\_id\_pk** | **fli\_num\_pk** |
| 10-3765 | 1023 |
| 05-4057 | 1134 |
| 11-1019 | 1454 |
| 07-0870 | 1256 |
| 01-6969 | 1324 |
| 02-7777 | 1242 |
| 12-4231 | 2157 |
| 51-3241 | 1214 |
| 23-6523 | 1432 |
| 21-6924 | 1953 |

**CUST\_BOOKING\_PLACE\_T (cust\_id\_pk, bo\_id\_pk)**

**PRIMARY KEY : (cust\_id\_pk, bo\_id\_pk)**

**FOREIGN KEY:** cust\_id\_pk REFERENCES CUSTOMER\_T(cust\_id\_pk), bo\_id\_pk REFERENCES BOOKING\_OFFICE\_T(bo\_id\_pk)

|  |  |
| --- | --- |
| Name of Table: | **CUST\_BOOKING\_PLACE\_T** |
| Primary Key: | (cust\_id\_pk, bo\_id\_pk) |
| Foreign Key: | cust\_id\_pk REFERENCES CUSTOMER\_T(cust\_id\_pk), bo\_id\_pk REFERENCES BOOKING\_OFFICE\_T(bo\_id\_pk) |
| **cust\_id\_pk** | **bo\_id\_pk** |
| 550106-12-5821 | PKL7373 |
| 930906-14-7043 | PKL7373 |
| 941130-07-5153 | NSN6969 |
| 610522-10-5378 | PKL7373 |
| 123400-56-9876 | SBH7171 |
| 581235-61-2456 | SRW7272 |
| 615493-59-7984 | PHG7070 |
| 923497-36-5951 | PKL7373 |
| 164973-59-1576 | PRK6868 |
| 985632-41-6932 | NSN6969 |

# ENTITY RELATIONSHIP DIAGRAM

## CONCEPUTAL DATABASE DESIGN (ERD-ENTITY RELATIONSHIP DIAGRAM)

### STEPS INVOLVED

STEP 1: Read the scenario given

STEP 2: Identify all entities found in the scenario

Entities:

1. AIRLINE\_T
2. BOOKING\_OFFICE\_T
3. FLIGHT\_T
4. PILOT\_T
5. FLIGHT\_ATTENDANT\_T
6. CUSTOMER\_T
7. FLIGHT\_BOOKING\_T

STEP 3: Identify all attributes in the scenario

Attributes:

1. AIRLINE\_T : air\_flight\_num, air\_company
2. BOOKING\_OFFICE\_T : bo\_id, bo\_state, bo\_address, bo\_city, bo\_street, bo\_postcode, bo\_phone\_num
3. FLIGHT\_T : fli\_num, fli\_business\_seat, fli\_economy\_seat, fli\_capacity, fli\_departure\_date, fli\_departure\_time, fli\_arrival\_date, fli\_arrival\_time,
4. PILOT\_T : pil\_id, pil\_staff\_fname, pil\_staff\_lname, pil\_age, pil\_experience, pil\_position, pil\_flying\_hours, pil\_salary, pil\_gender
5. FLIGHT\_ATTENDANT\_T : fa\_id, fa\_fname, fa\_lname, fa\_position, fa\_salary, fa\_phone\_num, fa\_street, fa\_city, fa\_state, fa\_postcode, fa\_country, fa\_gender
6. CUSTOMER\_T : cust\_id, cust\_fname, cust\_lname, cust\_email, cust\_street, cust\_city, cust\_state, cust\_postcode, cust\_country, cust\_hphone\_num, cust\_gender
7. FLIGHT\_BOOKING\_T : fb\_book\_num, fb\_book\_date, fb\_class, fb\_price, fb\_status, fb\_paid\_amount, fb\_outstanding

STEP 4: Identify Primary Keys (PKs)

Primary Keys:

1. AIRLINE\_T : **air\_flight\_num\_pk**, air\_company
2. BOOKING\_OFFICE\_T : **bo\_id\_pk**, bo\_state, bo\_address, bo\_city, bo\_street, bo\_postcode, bo\_phone\_num
3. FLIGHT\_T : **fli\_num\_pk**, fli\_business\_seat, fli\_economy\_seat, fli\_capacity, fli\_departure\_date, fli\_departure\_time, fli\_arrival\_date, fli\_arrival\_time,
4. PILOT\_T : **pil\_id\_pk**, pil\_staff\_fname, pil\_staff\_lname, pil\_age, pil\_experience, pil\_position, pil\_flying\_hours, pil\_salary, pil\_gender
5. FLIGHT\_ATTENDANT\_T : **fa\_id\_pk**, fa\_fname, fa\_lname, fa\_position, fa\_salary, fa\_phone\_num, fa\_street, fa\_city, fa\_state, fa\_postcode, fa\_country, fa\_gender
6. CUSTOMER\_T : **cust\_id\_pk**, cust\_fname, cust\_lname, cust\_email, cust\_street, cust\_city, cust\_state, cust\_postcode, cust\_country, cust\_hphone\_num, cust\_gender
7. FLIGHT\_BOOKING\_T : **fb\_book\_num\_pk**, fb\_book\_date, fb\_class, fb\_price, fb\_status, fb\_paid\_amount, fb\_outstanding

STEP 5: Identify the relationships between entities

1. Each flight travel must have two pilot. Each pilot may fly one or more flights. FLIGHT\_T **> | many-to-many** PILOT\_T **| <**
2. Each flight travel must have at least one and up to 3 flight attendants.

FLIGHT\_T **> |** **many-to-many** FLIGHT\_ATTENDANT\_T **| <**

1. Each customer has zero or more house phone numbers.

CUSTOMER\_T **| one-to-many** CUST\_HPHONE\_NUM\_T **| <**

1. Each customer has zero or more email addresses. Each email address is used only by one customer.

CUSTOMER\_T **| one-to-many** CUST\_EMAIL\_T **| <**

1. Each customer may book one or more flights.

CUSTOMER\_T **> | many-to-many** FLIGHT\_T **| <**

STEP 6: Draw an ERD

|  |  |
| --- | --- |
| Graphical representation | Meaning |
|  | Entity |
|  | Weak Entity |
|  | Attribute |
|  | Multivalued attribute |
|  | Derived attribute |
|  | Relationship |
|  | Identifying relationship |
|  | One-to-one relationship |
|  | One-to-many relationship |
|  | Many-to-many relationship |

Table 5.1 Entitiy Diagram Symbols

Diagram

Description automatically generated

Figure 5.1 Entity Relationship Diagram

### ERD

## LOGICAL DATABASE DESIGN (EERD-ENHANCED ENTITY RELATIONSHIP DIAGRAM)

### STEPS INVOLVED

**STEP 1:** Remove each multi-valued attribute found in ERD, then transform it into a new entity.

Multi-valued attribute: cust\_email, cust\_hphone\_num

New entity: CUST\_EMAIL\_T, CUST\_HPHONE\_NUM\_T

**STEP 2:** Remove every many-to-many relationship found in the ERD, then transform it into a new entity.

Many-to-many relationship:

1. FLIGHT\_T **> | many-to-many** PILOT\_T **| <**
2. FLIGHT\_T **> |** **many-to-many** FLIGHT\_ATTENDANT\_T **| <**
3. CUSTOMER\_T **> | many-to-many** FLIGHT\_T **| <**

New entity:

1. PILOT\_SCHEDULE\_T
2. ATTENDANT\_SCHEDULE\_T
3. FLIGHT\_BOOKING\_T

**STEP 3:** Develop database schema.

### EERD

Figure 5.2 Full Enhanced Entity Relationship Diagram

Diagram

Description automatically generated

Diagram

Description automatically generated

Figure 5.3 Enlarged Enhanced Entity Relationship Diagram Part 1

Diagram, schematic

Description automatically generated

Figure 5.4 Enlarged Enhanced Entity Relationship Diagram Part 2

# CONCLUSIONS

In doing this assignment, the team decides to divide the work into parts and distribute it as equal as possible among the members. The assignment is carried out with the utilisation of various Microsoft Office products, such as Microsoft Word (Documentation), Microsoft Excel (Database Schema), Microsoft Visio (ERD and EERD), and Microsoft Teams (Meeting and Coordination).

The process of doing the assignment goes on smoothly with proper communication between members and coordination from the group leader to members. Additionally, the team’s lecturer and advisor, Mr. Dhason Padmakumar, also gives excellent explanations and assistance, which contributes significantly to the outcome of the tasks.

When doing the assignment with the guidance of Mr. Dhason, the team learned a lot of professional skills and knowledge that are essential for constructing industry-standard databases. Skills such as drawing ERD & EERD, normalising unorganised databases, creating database schema, and identifying the relations between entities have deepened the team’s knowledge of database design.

With the skills in hand, the team could identify and construct a detailed database according to the scenario given. The assignment scenario, Malaysian Airlines Reservation System, revolves around building a detailed and centralised air-reservation system on the current booking system. The scenario is tricky and challenging but also clear and descriptive, allowing the team to analyse and identify the appropriate system requirements. Additionally, the scenario also takes on real-world problems as the basis of the storyline, which evokes the feeling of solving real and existing problems in the industry world.

Nevertheless, all the team members have done their part and completed the tasks well. The team has successfully built and constructed a centralised air-reservation system that goes in accordance with the Malaysian Airlines Reservation System scenario. To sum it all up, the team felt that this assignment is a wonderful experience that has developed much of their understanding on databases. The team also felt delighted at the results and knowledge gained. Moreover, there are no unhealthy conflicts involved in doing the tasks, and the team’s collaboration skills has also improved considerably.

# WORKLOAD DISTRIBUTION

|  |  |  |
| --- | --- | --- |
| TASK ID | TASK NAME | CONTRIBUTERS |
| 1.0 | Introduction | Mr. KEVIN MATTHEW ADYAN (TP058466) |
| 2.0 | Functionality of the current system | Mr. SELVAN NICHOLAS (TP058084) |
| 3.0 | Database and Database Management System |  |
| 3.1 | Disadvantages of File-Based System | Mr. MARCELL AGUNG WAHYUDI (TP058650) |
| 3.2 | Advantages of Databases and DBMS, functions of DBMS | Mr. MARCELL AGUNG WAHYUDI (TP058650) |
| 3.3 | Justifications for the Database Management System selected | Mr. KEVIN MATTHEW ADYAN (TP058466) |
| 4.0 | Business Rules & Normalisation |  |
| 4.1 | Business Rules | Mr. WATARU SHINZATO (TP061217) |
| 4.2 | Normalisation | Mr. WATARU SHINZATO (TP061217) |
| 5.0 | Entity Relationship Diagram |  |
| 5.1 | Conceputal Database Design (Erd-Entity Relationship Diagram) |  |
| 5.1.1 | Steps Involved | Mr. KEVIN MATTHEW ADYAN (TP058466)    Mr. WATARU SHINZATO (TP061217) |
| 5.1.2 | ERD | Mr. KEVIN MATTHEW ADYAN (TP058466)    Mr. MARCELL AGUNG WAHYUDI (TP058650) |
| 5.2 | LOGICAL DATABASE DESIGN (EERD-ENHANCED ENTITY RELATIONSHIP DIAGRAM) |  |
| 5.2.1 | STEPS INVOLVED | Mr. SELVAN NICHOLAS (TP058084) |
| 5.2.2 | EERD | Mr. KEVIN MATTHEW ADYAN (TP058466)    Mr. SELVAN NICHOLAS (TP058084) |
| 6.0 | Conclusions | Mr. SELVAN NICHOLAS (TP058084) |
| 8.0 | References | Mr. MARCELL AGUNG WAHYUDI (TP058650) |

Figure 7.1 Workload Distribution

# REFERENCES

Bédard, Y. & Henriques, W., 2002. Modern Information Technologies in Environmental Health Surveillance: An Overview and Analysis. *Canadian Journal of Public Health,* 93(1), pp. 29-33.

C# Curator, 2019. *What Are Object-Oriented Databases And Their Advantages.* [Online]   
Available at: https://www.c-sharpcorner.com/article/what-are-object-oriented-databases-and-their-advantages2/#:~:text=An%20object%2Doriented%20database%20(OODBMS,object%2Doriented%20database%20management%20systems.  
[Accessed 16 Apr 2021].

Caseau, Y., 1991. An Object-Oriented Deductive Language. *Annals of Mathematics and Artificial Intelligence,* 3(2-4), pp. 211-258.

Castro, K., 2018. *Advantages of Database Management System.* [Online]   
Available at: https://www.tutorialspoint.com/Advantages-of-Database-Management-System  
[Accessed 16 Apr 2021].

Computing Students, n.d. *Databases and File Based Systems.* [Online]   
Available at: http://www.computingstudents.com/notes/database\_systems/database\_file\_based\_systems.php#:~:text=A%20file%20based%20system%20is,can%20be%20used%20or%20transported.  
[Accessed 16 Apr 2021].

DICODING INTERN, 2020. *Apa itu Database? Contoh Produk dan Fungsinya.* [Online]   
Available at: https://www.dicoding.com/blog/apa-itu-database  
[Accessed 16 Apr 2021].

Erkec, E., 2020. *Overview of Microsoft SQL Server Management Studio (SSMS).* [Online]   
Available at: https://www.sqlshack.com/overview-of-microsoft-sql-server-management-studio-ssms/  
[Accessed 24 April 2021].

GeeksforGeeks, 2020. *Difference between Open Source Database and Commercial Database.* [Online]   
Available at: https://www.geeksforgeeks.org/difference-between-open-source-database-and-commercial-database/  
[Accessed 28 May 2021].

Ghanayem, M., 2019. *What is SQL Server Management Studio (SSMS)?.* [Online]   
Available at: https://docs.microsoft.com/en-us/sql/ssms/sql-server-management-studio-ssms?view=sql-server-ver15  
[Accessed 24 April 2021].

Guru99, 2019. *What is DBMS? Application,Types,Example,Advantages.* [Online]   
Available at: https://www.guru99.com/what-is-dbms.html  
[Accessed 16 Apr 2021].

JavaTPoint, 2020. *Types of Databases.* [Online]   
Available at: https://www.javatpoint.com/types-of-databases  
[Accessed 28 May 2021].

Nethi, D., 2019. *SQL Server Management Studio (SSMS) 18.0 released for general availability.* [Online]   
Available at: https://cloudblogs.microsoft.com/sqlserver/2019/04/24/sql-server-management-studio-ssms-18-0-released-for-general-availability/  
[Accessed 24 April 2021].

Panwar, A., 2020. *Types of Database Management Systems.* [Online]   
Available at: https://www.c-sharpcorner.com/UploadFile/65fc13/types-of-database-management-systems/  
[Accessed 8 May 2021].

Rackspace Technology, n.d. *What Are Cloud Databases?.* [Online]   
Available at: https://www.rackspace.com/library/what-are-cloud-databases  
[Accessed 16 Apr 2021].

Schaefer, L., n.d. *What is NoSQL?.* [Online]   
Available at: https://www.mongodb.com/nosql-explained  
[Accessed 16 Apr 2021].

Talmage, R. & Delaney, K., 2005. *SQL Server 2005 Management Tools.* [Online]   
Available at: https://www.itprotoday.com/sql-server/sql-server-2005-management-tools  
[Accessed 1 June 2021].

Thakur, S., 2015. *Disadvantages of File Processing System.* [Online]   
Available at: https://whatisdbms.com/disadvantages-of-file-processing-system/#:~:text=Poor%20data20security%20is%20the,data%20up%20to%20a%20level.  
[Accessed 20 April 2021].

THIRU, n.d. *Functions of Database Management System (DBMS).* [Online]   
Available at: http://www.myreadingroom.co.in/notes-and-studymaterial/65-dbms/465-functions-of-dbms.html  
[Accessed 16 Apr 2021].

Tutorials Point, 2019. *MS SQL Server - Create Database.* [Online]   
Available at: https://www.tutorialspoint.com/ms\_sql\_server/ms\_sql\_server\_create\_database.htm  
[Accessed 16 April 2021].

Tutorialspoint, 2020. *Types of databases.* [Online]   
Available at: https://www.tutorialspoint.com/Types-of-databases  
[Accessed 28 May 2021].