



RoMusic Database Design Document (DDD)

Version <version number>

Prepared by: Paterio S. Castro III

Table of Contents

1 INTRODUCTION.....	1
1.1 DOCUMENT OBJECTIVES.....	1
1.2 INTENDED AUDIENCES.....	1
1.3 REFERENCES.....	1
2 DETAILED DATABASE DESIGN.....	2
2.1 DYNAMODB DESIGN (NOSQL DATABASE).....	2
2.1.1 <i>Object Diagram</i>	3
2.1.2 <i>Data dictionary</i>	4
2.1.2.1 Data dictionary for Element: User.....	4
2.1.2.2 Data dictionary for Element: Password.....	4
2.1.2.3 Data dictionary for Element: File.....	4
2.1.2.4 Data dictionary for Element: Area.....	5
2.2 SQLITE DATABASE DESIGN (RELATIONAL DATABASE).....	6
2.2.1 <i>Conceptual diagram</i>	6
2.2.2 <i>Description</i>	6
2.2.3 <i>Purpose of Tables</i>	6
2.2.3.1 Purpose of Area Table.....	6
2.2.3.2 Purpose of File Table.....	7
2.2.3.3 Purpose of User Table.....	7
2.2.4 <i>Relations</i>	7
3 REFERENCES.....	7
4 APPENDIX 1 – XML SCHEMA.....	8

1 Introduction

The section introduces the Database Design Document (DDD) \

Document Objectives

This DDD for the LocAdoc software has the following objectives: 3

- Describe the design of a DynamoDB and SQLite database, that is, a collection of related data stored in one or more computerized files in a manner that can be accessed by users or computer programs via a database management system (DBMS). It can also describe the software units used to access or manipulate the data.
- To serve as the basis for implementing the database. It provides the acquirer visibility into the design and provides information needed for software support.

1.1 Intended Audiences

This DDD is intended for the following audiences:

- Technical reviewers, Supervisor and UOW staff who must evaluate the quality of this document.
- LocAdoc developers including:
 - Architects, whose overall architecture must meet the requirements specified in this document.
 - Designers, whose design must meet the requirements specified in this document.
 - Programmers, whose software must implement the requirements specified in this document.
 - Testers, whose test cases must validate the requirements specified in this document.

1.2 References

This DDD refers to the following references:

- Apple Music
- Spotify

2 Detailed Database Design

This section describes the actual design of different databases at varying levels of abstraction. A subsection for each of conceptual, internal, logical and physical levels.

2.1.1 Data dictionary

2.1.1.1 Data dictionary for Element: <Table Name>

Name	Data Type	Constrain	Description
Username	string	Min :1, Max:1	Username
Name	String		Name of the user
Password	Integer	Min :1, Max:1	The password of the user
AdminArea	Integer		User has the freedom to set the admin area.

2.1.1.2 Data dictionary for Element: <Table Name>

Name	Data Type	Constrain	Description
Password	String		Hashed Password

2.1.1.3 Data dictionary for Element: <Table Name>

Name	Data Type	Constrain	Description
FileID (primary key)	Integer		
CurrentFileName	string	Min :1, Max:1	A new name for the file assigned by the application
Password	Integer		Password that was used to encrypt the file (password ID)

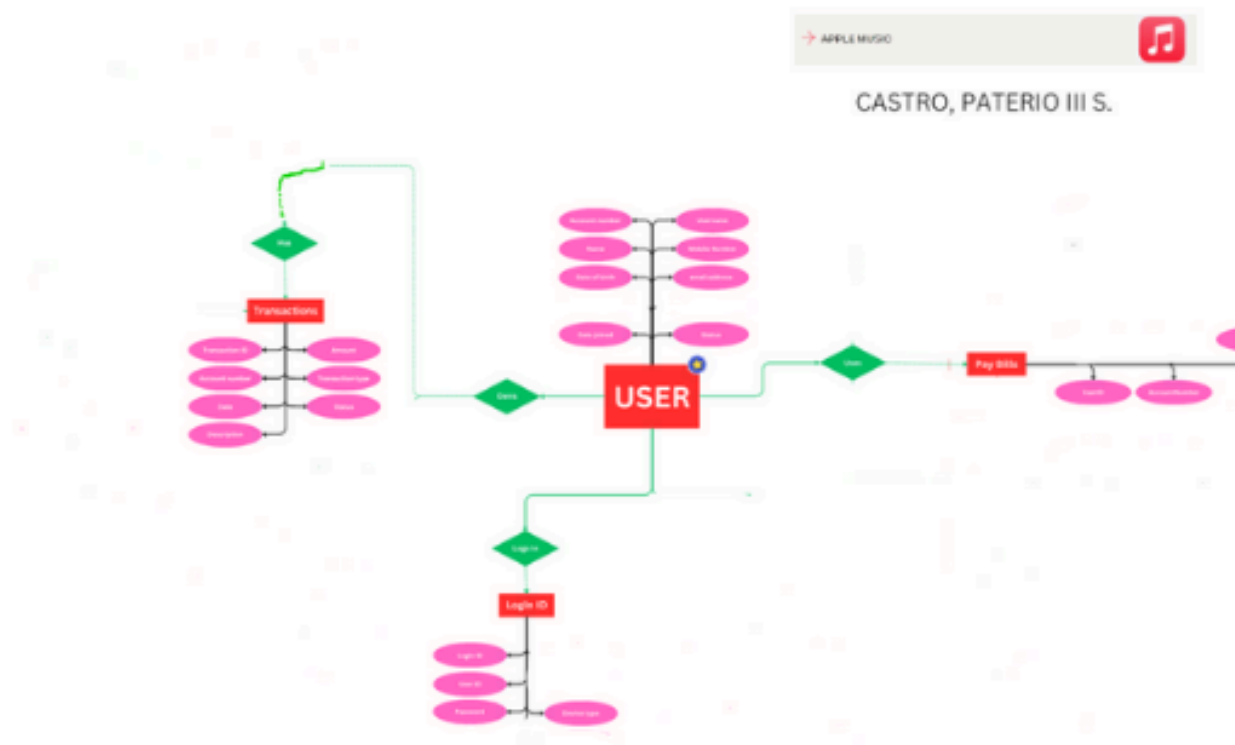
Area	Integer		The area where the file has been grouped in (Area ID)
------	---------	--	---

2.1.1.4 Data dictionary for Element: <Table Name>

Name	Data Type	Constrain	Description
Area ID (primary key)	string	Min :1, Max:1	ID to identify the area
AreaName	string		Stores the name of the area given by the user.
Description	string		Stores the description created by the user for the area.
Longitude	Decimal		The Longitude of the first file that was created in this area
Latitude	Decimal		The Latitude of the first file that was created in this area.
Radius	Decimal		The radius around the point where the first file was created.

2.2 MySQL database design (Relational database)

2.2.1 Conceptual diagram



2.2.2 Description

The SQL database's conceptual model is shown in this diagram. Once the user's data has been imported, this database will be established. Details about the user who is presently logged in will be included in the user session. In addition to having a password, the user will have access to an admin section where he can modify his account, including changing his password (which is optional). Every user will have at least one file. Only the Area and file information will be stored in the local database. After being encrypted with the user's password, all data aside from the main and foreign keys will be kept. Upon request from the user, the database contents will be decrypted.

2.2.3 Purpose of Tables

2.2.3.1 Purpose of <Table Name> Table

This table stores the information regarding the Area a file was created. The radius is the area around that point where the files grouped in that area can be accessed. The longitudinal and latitudinal value is used to encrypt the file in that area.

2.2.3.2 Purpose of <Table Name> Table

This table stores all the information regarding a file used by the user. When the file is imported into the application, a new file name is generated and it is mapped with the actual table. The longitude and the latitude of the location where the file was created is also stored. The file will also will have a password which was used to encrypt the file.

2.2.3.3 Purpose of <Table Name> Table

The user stores the details of the user currently logged into the system. The user will have a password and an admin area where he can make changes to the account. The primary key of the table will be the email ID. This table will only one record as the database only belongs to one user.

2.2.4 Relations

From Table	To Table	Relation
User	Area	A user may set an admin area.
File	Password	A file will be encrypted using a password.
User	Password	A user has a password.

3 References

[1] Apple Music

[2] Spotify

