X*Waze* App
Database Design Document (DDD)

Version 1.0

Prepared by: Carreon, Alessandra Celine B
Panganiban, Dorothy Joy Y.

Revision History

Date	Version	Description	Author

Table of Contents

1 Introduction

The section introduces the Database Design Document (DDD) for XWaze App to its readers.

1.1 Document Objectives

This DDD for the Fake Waze App has the following objectives:

- Describe the design of a MySQL database, which is a collection of related data stored in a manner that can be accessed by users or computer programs via a database management system (DBMS).
- Outline the database schema, including tables, relationships, and constraints, to support the app's functionality.
- Serve as the basis for implementing the database, providing visibility into the design for software support and future enhancements.

1.2 Intended Audiences

This DDD is intended for the following audiences:

- Fake Waze App developers, including:
    - Architects, whose overall system architecture must align with this database design.
    - Designers, who must ensure the data model meets application requirements.
    - Programmers, who must implement the database structure and queries.
    - Testers, who must validate the database implementation against application requirements.

1.3 References

This DDD refers to the following references:

- https://believed-bongo-319.notion.site/CTINFMGL-Project-Specifications-19296450aad180aea9ebf27987415f4c

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 Data dictionary

2.1.1 Data dictionary for Element: WazeAccounts

Name	Data Type	Constrain	Description
account_id (PK)	INTEGER	AUTO_INCREMENT	Account ID to identify the user.
email	VARCHAR (50)	UNIQUE, NOT NULL	Email of the User.
username	VARCHAR (50)	UNIQUE, NOT NULL	Chosen username of the user.
password	VARCHAR (50)	NOT NULL	The password of the user.
birthdate	DATE	NOT NULL	User's birthdate.
first_name	VARCHAR (50)	NOT NULL	User's first name/given name..
last_name	VARCHAR (50)	NOT NULL	User's last name/family name.

2.1.2 Data dictionary for Element: WazeRoutes

Name	Data Type	Constrain	Description
route_id (PK)	VARCHAR (25)		Route ID to identify user's saved route.
account_id (FK)	VARCHAR (50)		Account ID to identify the user that is connected to routes.
route_startpoint	VARCHAR (200)		Starting location of the user.
route_endpoint	VARCHAR (200)		End location of the user.

2.1.3 Data dictionary for Element: WazeAltRoutes

Name	Data Type	Constrain	Description
alt_route_id (PK)	VARCHAR (10)		Alternative route ID of the user.
route_id (FK)	VARCHAR (25)		Route ID to identify the user's saved route.
alt_routes	VARCHAR (150)		Alternative routes that the user can choose.
stop_overloc	VARCHAR (200)		Stop over location of the user.

2.1.4 Data dictionary for Element: WazeTravelTime

Name	Data Type	Constrain	Description
traveltime_id (PK)	VARCHAR (10)		Travel time ID to identify user's travel time.
route_id (FK)	VARCHAR (25)		Links to the user's saved route.
est_time	TIME		Estimated time to complete the travel route.

2.1.5 Data dictionary for Element: WazePlannedDrives

Name	Data Type	Constrain	Description
planneddrive_id (PK)	VARCHAR(10)		Unique identifier for a planned drive.
account_id (FK)	VARCHAR (50)		Links to the user who created the planned drive.
route_id (foreign key)	VARCHAR(25)		Links to the saved route for the planned drive.
calendar	DATE		Date of the planned drive.

planned_time	TIME		The time the user plans to start the drive.
pinned_loc	VARCHAR (200)		User's pinned location (e.g., a saved stop over or destination)

2.1.6 Data dictionary for Element: locations

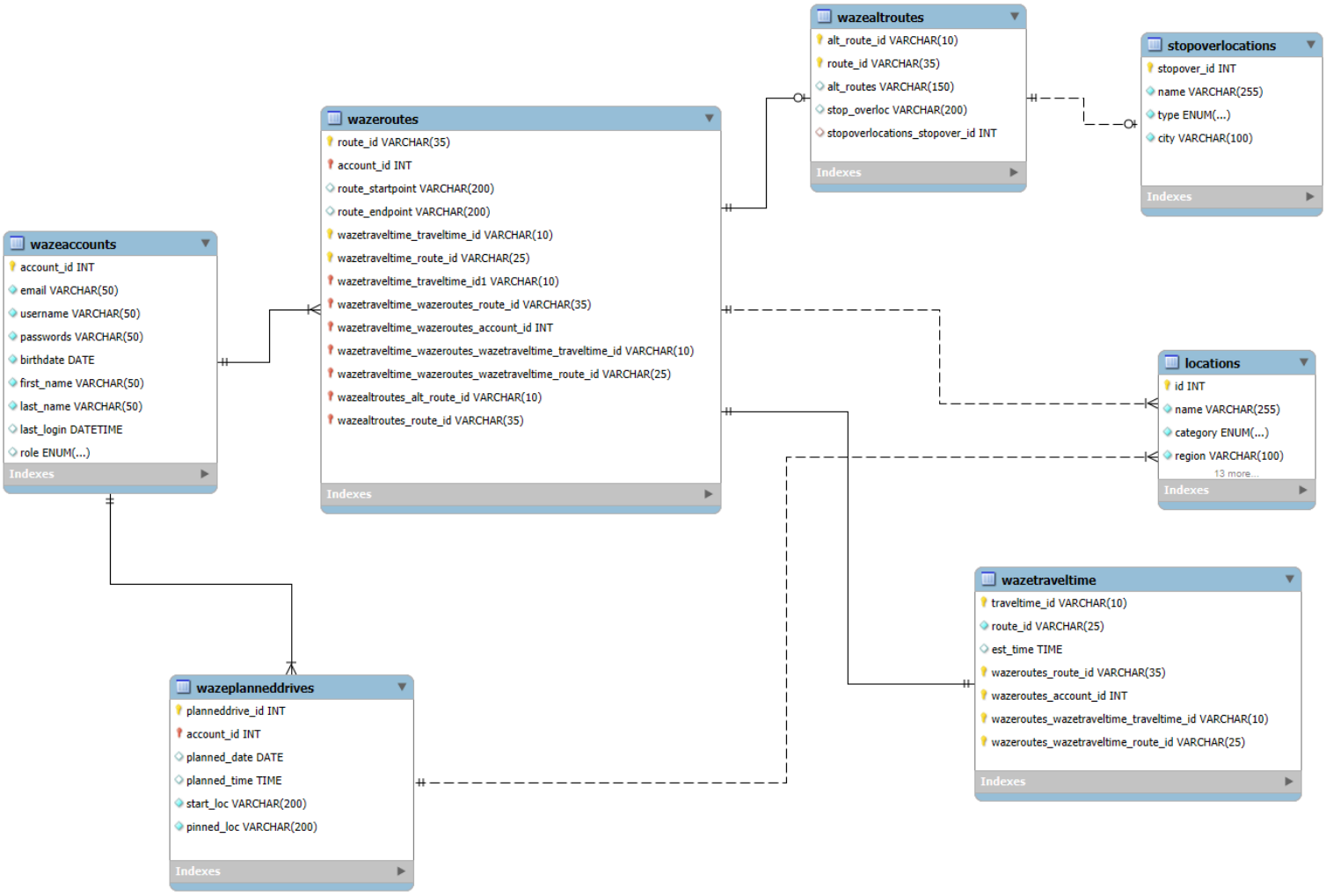
Name	Data Type	Constrain	Description
id (PK)	INTEGER (10)	AUTO_INCREMEN, NOT NULL	Unique identifier for each record.
name	VARCHAR (255)	NOT NULL	Name of the location or entity.
category	ENUM('City', 'Transport Hub', 'Expressway Exit'	NOT NULL	Defines the type of entity (e.g., city, transport hub, or expressway exit).
region	VARCHAR(100)	NOT NULL	Specifies the region where the entity is located.

2.1.7 Data dictionary for Element: StopOverlocations

Name	Data Type	Constrain	Description
stopover_id (PK)	INTEGER (10)	AUTO_INCREMENT	Unique identifier for each stopover location.
name	VARCHAR (255)	NOT NULL	Name of the stopover location.
type	ENUM('Mall', 'Fast Food', 'Gas Station', 'Rest Area')	NOT NULL	Category/type of the stopover location.
city	VARCHAR(100)	NOT NULL	City where the stopover is located.

2.1 MySQL database design (Relational database)

2.2.1 Entity Relationship Diagram



2.2.2 Description

This ERD (Entity Relationship Diagram) represents the **XWaze system**, which manages information for users, administrators, locations, routes, alternate routes, travel time, planned drives, and their corresponding unique IDs. The **admin session** can **view and modify** (e.g., create, update and delete) the user's data even if the user is not signed in.

The **user session** can create their account and change their information (e.g., first name, last name, birthdate, email, username, password) under the account settings menu item. They can **view and modify** their table to create, update and delete their route(s) or plan their drive/s.

2.2.3 Purpose of Tables

2.1.1.1 Purpose of WazeAccounts Table

The **WazeAccounts** table is mainly used to store all the user account information for the app. It keeps track of individual users so the system can personalize their experience, like showing their saved routes, planned drives, or any preferences they might have. Basically, whenever a user plans a drive or saves a route, this table helps connect that data back to the correct person. Without this, the app wouldn't know which user owns which routes or drives.

2.1.1.2 Purpose of WazeRoutes Table

The **WazeRoutes** table holds the main details about the routes users create or save. It includes the starting point and endpoint of each route and connects back to the user who saved it through the account_id. This makes it easier for the app to show each user their own set of routes. The table plays a big role in the whole navigation system because it's where all the key information about user-planned routes is stored.

2.1.1.3 Purpose of WazeAltRoutes Table

The **WazeAltRoutes** table stores information about alternative routes related to a main route. Sometimes, there are different ways to get from point A to point B, and this table keeps those options. It also includes any stopover locations along the alternative routes. Since it's linked to the main routes through the route_id, it ensures that all the alternative paths are correctly associated with the original route a user planned.

2.1.1.4 Purpose of WazeTravelTime Table

The **WazeTravelTime** table keeps track of the estimated travel time for each route. This includes details like how long it will take to complete a route based on current or average traffic conditions. It's connected to the WazeRoutes table, so the estimated times are always tied to the correct route. This table is important because travel time is a big factor in route planning, and users rely on it to pick the fastest or most convenient path.

2.1.1.5 Purpose of WazePlannedDrives Table

The **WazePlannedDrives** table stores details about drives users plan for the future. It includes the date and time the user intends to start the drive, along with the starting location and the pinned destination. This table helps users organize their trips and ensures the app can remind them or provide updates before they start driving. Since it also references the WazeAccounts table, it keeps everything personalized for each user.

2.1.1.6 Purpose of locations Table

The **WazeLocations** table is where all the possible locations that users can select in the app are stored. These could be cities, transport hubs, or expressway exits. This table helps maintain a consistent list of locations so users don't accidentally select or type in invalid places. It makes location management easier and ensures the dropdowns in the app are always up to date.

2.1.1.7 Purpose of StopoverLocations

This table stores information about any stopovers users want to add along their routes. Stopovers are points where users might want to pause during their trip, like gas stations, restaurants, or scenic spots. By keeping

these locations in a separate table, the app can better manage and display them as part of the overall route without mixing them up with main start or end locations.

2.2.4 Relations

From Table	To Table	Relation
WazeAccounts (role ENUM **Admin**)	WazeAccounts (role ENUM **User**)	The WazeAccounts (role ENUM **Admin**) can view and manage (e.g. create, update, and delete) the WazeAccounts (role ENUM **User**)
WazeAccounts (role ENUM **User**)	WazeAccounts (role ENUM **Admin**)	The WazeAccounts (role ENUM **User**) is managed by **one admin**
WazeAccounts (role ENUM **Admin**)	WazeRoute (role ENUM **User**)	The WazeAccounts (role ENUM **Admin**) can view and manage **multiple users'** WazeRoute including adding, updating, and deleting **users' WazeRoutes**.
WazeAccounts (role ENUM **Admin**)	WazeAltRoutes (role ENUM **User**)	The WazeAccounts (role ENUM **Admin**) can view **users' WazeAltRoutes**.
WazeAccounts (role ENUM **Admin**)	WazeTravelTime (role ENUM **User**)	The WazeAccounts (role ENUM **Admin**) can view **users' WazeTravelTime**.
WazeAccounts (role ENUM **Admin**)	WazePlannedDrives (role ENUM **User**)	The WazeAccounts (role ENUM **Admin**) can view and manage multiple **users' WazePlannedDrives** including adding, updating, and deleting **users' WazePlannedDrives.**
WazeRoute (role ENUM **User**)	WazeAccounts (role ENUM **Admin**)	Each WazeRoute (role ENUM **User**) can be viewed and managed (e.g., insert, update and delete) by **one admin**
WazeAltRoutes (role ENUM **User**)	WazeAccounts (role ENUM **Admin**)	Each WazeAltRoutes (role ENUM **User**) can be viewed by **one admin**
WazeTravelTime (role ENUM **User**)	WazeAccounts (role ENUM **Admin**)	Each WazeTravelTime (role ENUM **User**) can be viewed by **one admin**

WazePlannedDrives (role ENUM **User**)	WazeAccounts (role ENUM **Admin**)	Each WazePlannedDrives(role ENUM **User**) can be viewed and managed (e.g., insert, update and delete) by **one admin**
WazeAccounts (role ENUM **User**)	WazeRoutes (role ENUM **User**)	The WazeAccounts (role ENUM **User**) can view, create, update, and delete **their own WazeRoute**
WazeAccounts (role ENUM **User**)	WazeAltRoutes (role ENUM **User**)	The WazeAccounts (role ENUM **User**) can view **their own WazeAltRoute**
WazeAccounts (role ENUM **User**)	WazeTravelTime (role ENUM **User**)	The WazeAccounts (role ENUM **User**) can view **their own WazeTravelTime**
WazeAccounts (role ENUM **User**)	WazePlannedDrives (role ENUM **User**)	The WazeAccounts (role ENUM **User**) can view, create, update, and delete **their own WazePlannedDrives**
WazeRoutes (role ENUM **User**)	WazeAccounts (role ENUM **User**)	Each WazeRoutes (role ENUM **User**) can be viewed and managed (e.g., insert, update and delete) **in their own user account**
WazeAltRoutes (role ENUM **User**)	WazeAccounts (role ENUM **User**)	Each WazeAltRoute (role ENUM **User**) can be viewed **in their own user account**
WazeTravelTime (role ENUM **User**)	WazeAccounts (role ENUM **User**)	Each WazeTravelTime (role ENUM **User**) can be viewed **in their own user account**
WazePlannedDrives (role ENUM **User**)	WazeAccounts (role ENUM **User**)	Each WazePlannedDrives (role ENUM **User**) can be viewed and managed (e.g., insert, update and delete) **in their own user account**
WazeRoutes (role ENUM **Admin**)	Locations	The WazeRoutes (role ENUM **Admin**) connects and access to **multiple Locations**
Locations	WazeRoutes (role ENUM **Admin**)	The Locations can provide one location to WazeRoutes (role ENUM **Admin**)

WazeRoutes (role ENUM **User**)	Locations	The WazeRoutes (role ENUM **User**) connects to **multiple Locations in their own user account**
Locations	WazeRoutes (role ENUM **User**)	The Locations can provide one location to WazeRoutes (role ENUM **Users**)
WazeAltRoutes (role ENUM **Admin**)	StopOverLocations	The WazeAltRoutes (role ENUM **Admin**) can have **one StopOverLocation**
StopOverLocations	WazeRoutes (role ENUM **Admin**)	The StopOverLocations can provide **one stop over location to** WazeRoutes (role ENUM **Admin**)
WazeAltRoutes (role ENUM **User**)	StopOverLocations	The WazeAltRoutes (role ENUM **User**) can have **one StopOverLocation**
StopOverLocations	WazeRoutes (role ENUM **User**)	The StopOverLocations can provide **one stop over location to** WazeRoutes (role ENUM **User**)
WazePlannedDrives (role ENUM **Admin**)	Locations	The WazePlannedDrives (role ENUM **Admin**) can choose **many** Locations
Locations	WazePlannedDrives (role ENUM **Admin**)	The Locations can show many **locations** to WazePlannedDrives (role ENUM **Admin**)
WazePlannedDrives (role ENUM **Users**)	Locations	The WazePlannedDrives (role ENUM **Users**) can choose **many** Locations
Locations	WazePlannedDrives (role ENUM **Users**)	The Locations can show many **locations** to WazePlannedDrives (role ENUM **Users**)