CMPG 321 GROUP ASSIGNMENT

Project Phase 2

Adriaan Pienaar (39399575)

Armin Pretorius (34739572)

Francois Botha (34507965)

Michael van Niekerk (29580080)

Louis Willemse (38887657)

Group Assignment submitted for CMPG 321 for the degree *Bachelor of Science* in *Information Technology* at the North-West University

Lecturer: Mr Jaco Pretorius

Table of Contents:

1. Project Team

Project Manager – Project Leader

FRANCOIS BOTHA
Project Manager

Turku, Finland
+27 65 826 9049
f.botha@outlook.com
34507965

Business Analyst

LOUIS WILLEMSE
Business Analyst

Cape Town, South Africa
+27 78 912 1412
fl.willemse@gmail.com
38887657

Software Analyst

ADRIAAN PIENAAR
Software Analyst

Cape Town, South Africa
+27 79 021 4757
japienaar7@gmail.com
39399575

Database Engineer

MICHAEL VAN NIEKERK
Database Designer

Richards Bay, South Africa
+27 83 588 1678
michael.vn98.16@gmail.com
29580080

Software Developer

ARMIN PRETORIUS
Software Developer

Cape Town, South Africa
+27 79 646 8928
arminp1139@gmail.com
34739572

2. System Requirements Analysis

2.1. Introduction

Group 5 (the company) has been approached by North-West University (the client) to facilitate the analysis, design, build, and implementation of a Smart Traffic Management Database System tailored for the NWU Campuses. The project aims to enhance urban mobility, optimize pedestrian, and bicycle commuting, and foster a sustainable and livable environment within and around the campuses.

This Request for Proposal (RFP) outlines the project's objectives, scope, deliverables, and submission requirements for interested vendors.

2.2. Objective

The objective is to provide a Smart Traffic Management Database System customized for NWU campuses and the surrounding areas concerning traffic. It will aim to improve the flow of traffic and reduce congestion as far as possible around the campuses, to ensure pedestrian safety is considered and upheld, and to provide information on how to improve the current methods of transportation.

The goal is to improve traffic and promote sustainable transportation options available to students and other commuters in and around the campus while maintaining a small carbon footprint. Additionally, the system will facilitate data-driven decision-making for stakeholders to improve the campus infrastructure and transportation planning initiatives.

2.3. Scope

The scope of the project has been defined and summarized to the following points:

1. The development of a functional database that is scalable and allows various types of data to be stored.
2. A system that allows real-time monitoring and analysis of data from various data sources.
3. An infrastructure that creates an environment for the collection of data from various data sources.
4. A system that promotes sustainable, efficient, and effective transportation modes that include the use of walking, cycling, public transportation, car-pooling, and route suggestions.
5. A system for reporting and management of incidents and traffic and other transportation-related alerts.
6. Methods for the integration of campus infrastructure and other external systems to supplement decision-making and data generation.

2.4. Assumptions

Assumptions can and have been made to ensure the smooth transition to and implementation of the system. Assumptions included in the overall project:

- Traffic flow sensors are included which will detect the number of cars and pedestrians and how their volume at various times will be recorded in the database, thus the overall flow of traffic.
- Solar or other 'green' energy sources for sensors for always-on connectivity and to ensure the seamless input of data from sensors.
- The database system will make use of an external global map system to plot the various streets and locations of objects as noted in the database.

2.5. Operations

Data Collection & Integration:

- Implement a Traffic Management Database System to collect real-time data from various sources including IoT devices, Google Earth, traffic management authorities, NWU campus stakeholders, and the NWU community.
- Data on streets, intersections, traffic objects, obstructions, traffic infrastructure, pedestrians, incidents, and specific locations will be collected.

Real-time Updates:

- The system relies on real-time updates to the Traffic Management Database System for the most current information. The data will be collected from various sources and therefore the sources should rely on sustainable power sources that ensure consistent streams of input data and rely on uptime through load shedding and other forms of power interruptions.

Data Analytics & Optimization:

- Apply analytics techniques to the collected data to identify traffic trends, predict congestion, and optimize traffic flow.
- Tools will be provided to optimize routes based on factors like distance, safety, and environmental impact. This will assist in promoting efficient and eco-friendly modes of transportation.
- The use of historical data and analytics will provide forecasting of traffic patterns and predict congestion based on previous situations, assisting with traffic flow optimization, and planning alternative routes.
- Predict traffic surges during events and adjust traffic management strategies to accommodate increased traffic volumes.

Smart City Capabilities:

- Leverage real-time and predictive analysis, spatial visualization technologies, and data-driven insights to empower traffic management authorities.

Bicycle Lane Optimization:

- Provide insights into how bicycle lanes can be implemented at a larger scale to ease the congestion by motor vehicles and other factors such as obstructions and incident-prone areas.

Public Transportation:

- Analyse high-usage routes with many commuters and provide insights for better public transportation possibilities by promoting and assisting in the creation of strategies to incentivize the use of public transport.

Urban Mobility and Community Engagement:

- Optimize pedestrian and bicycle commuting to enhance urban mobility within and around campuses to assist in driving more sustainable and healthier options for commuting.
- Commuter engagement will be used to involve the community in traffic management efforts and make smarter decisions about traffic control. User-submitted feedback will enhance the traffic management system by identifying problems and highlighting focus areas.

Standards and Reports:

- Adhere to established standards and guidelines for data collection, storage, and analysis, promoting innovation in traffic management. The technological infrastructure needs to be robust and provide for high throughput and processing of data.
- Users will be provided with reports on traffic flow analysis, congestion patterns, incident summaries, and user feedback while considering ethical concerns and other ethical factors.
- Visualization will be used to present routes in a user-friendly and easily understandable manner.

Data Privacy:

- The system must adhere to strict data privacy and security measures to protect personal information, limiting access to authorized personnel only.

Sustainability and Safety:

- Promote eco-friendly commuting options and support the strategic intent of a sustainable city to align with the intent of the WEF.
- Enhance road safety through clear signage, markings, and intelligent pedestrian signals that adapt to pedestrian volumes.

2.6. Business Rules

1. Street Information: The database will be populated with information on streets, which will form the foundation for the traffic management database system. Information on the name and other location information will be recorded about the street.
2. Street Sections: Street segments will also be recorded to ensure that specific parts of streets are recorded, should they be separated or different. This will allow for geospatial accuracy. Each segment length and direction will be recorded for additional stored information.
3. Intersections: The intersections of crossing streets will be recorded with the exact geolocation of the intersections. Both streets that are connected to the intersection will be recorded. Intersections will also be recorded for the various street segments, should there be an intersection involved. This will give more accuracy to traffic data. Intersections might also have pedestrian crossings.
4. Lanes: Additionally, the various lanes in the streets will be recorded and linked to the street segments to provide more granular information regarding the streets and the smaller sections. This will improve the accuracy of all other recorded data.
5. Road Conditions: The conditions of roads will be recorded with information on the type of condition, the details regarding the condition, and the date range that it is relevant.
6. Road Objects: Road objects will be stored to add more information about the streets. It will contain information on the type of objects and where they are located.
7. Road Events: Road events are recorded and will have an impact on the traffic reports that the system will provide. It is important to record all information regarding the events, including the date and time of the events as it will impact traffic.
8. Speed Limits: The speed limit will be recorded for the various streets, and it should be considered that a street may have different speed limits in different sections of the street. The time of day might also influence the speed limits applied.
9. Public Transport: Public transport is relevant to the database and the public transport stops will be recorded. It is important to note that the public transport stops might not be located on public transport lanes.
10. Traffic & Pedestrian Data: Traffic and pedestrian data will be recorded in the system and will provide context to the usage of streets at various specific sections, as well as the busy intersections. The data and time intervals of the recorded data need to be included for both vehicles and pedestrians. Traffic data will be based on sections of the street, while pedestrian data will be recorded on pedestrian walkways, which might be located only in specific areas of the streets.

3. Logical Schema

3.1. Entities & Attributes

Entity	Attributes	Data Type
LOCATION	loc_id	INT
	loc_name	VARCHAR(50)
	loc_latitude	NUMBER(8,6)
	loc_longitude	NUMBER(8,6)
	loc_city	VARCHAR(50)
	loc_province	VARCHAR(50)
	loc_country	VARCHAR(50)
	loc_postal	VARCHAR(5)
SATELLITE_VIEW	satellite_view_id	INT
	loc_id	INT
	satellite_view_url	VARCHAR(100)
STREET	street_id	INT
	loc_id	INT
	street_name	VARCHAR(50)
STREET_SEGMENT	segment_id	INT
	street_id	INT
	segment_length	NUMBER(8,3)
	segment_direction	NUMBER(8,3)
	segment_road_type	VARCHAR(50)
	segment_start_loc_id	INT
	segment_end_loc_id	INT
ROAD_EVENT	event_id	INT
	segment_id	INT
	event_type	VARCHAR(100)
	event_start_datetime	DATE
	event_end_datetime	DATE
PUBLIC_TRANS_STOP	ptstop_id	INT
	segment_id	INT
	ptstop_name	VARCHAR(50)
	ptstop_type	VARCHAR(50)
	ptstop_loc_id	INT
SPEED_LIMIT	slimit_id	INT
	segment_id	INT
	slimit_value	INT
	slimit_start_time	DATE
	slimit_end_time	DATE
ROAD_OBJECT	object_id	INT
	segment_id	INT
	object_type	VARCHAR(50)
	object_description	VARCHAR(100)
	object_loc_id	INT
CONGESTION_LEVEL	congestion_id	INT
	congestion_type	VARCHAR(50)
	congestion_description	VARCHAR(100)
LIVE_TRAFFIC_DATA	tdata_id	INT
	segment_id	INT
	congestion_id	INT
	tdata_datetime	DATE
	tdata_average_speed	NUMBER(8,3)
	tdata_traffic_volume	INT
	tdata_live_travel_time	NUMBER(8,3)

TRAFFIC_DATA_PREDICTOR	predictor_id	INT
	segment_id	INT
	predictor_day_of_week	INT
	predictor_hour_of_day	INT
	predictor_travel_time	NUMBER(8,3)
	predictor_speed	NUMBER(8,3)
TRAFFIC_CONDITION	traf_cond_id	INT
	segment_id	INT
	traf_cond_type	VARCHAR(50)
	traf_cond_description	VARCHAR(100)
	traf_cond_start_datetime	DATE
	traf_cond_end_datetime	DATE
STREET_VIEW	street_view_id	INT
	segment_id	INT
	street_view_url	VARCHAR(100)
ROAD_CONDITION	road_cond_id	INT
	segment_id	INT
	road_cond_type	VARCHAR(50)
	road_cond_description	VARCHAR(100)
	road_cond_start_datetime	DATE
	road_cond_end_datetime	DATE
POINT_OF_INTEREST	poi_id	INT
	street_id	INT
	poi_name	VARCHAR(50)
	poi_description	VARCHAR(100)
	poi_category	VARCHAR(50)
	poi_loc_id	INT
INTERSECTION	intersection_id	INT
	first_street_id	INT
	second_street_id	INT
	intersection_loc_id	INT
INTERSECTING_SEGMENTS	segment_id	INT
	intersection_start_id	INT
	intersection_end_id	INT
PEDESTRIAN_WALKWAY	walkway_id	INT
	segment_id	INT
	walkway_type	VARCHAR(50)
	walkway_width	NUMBER(8,3)
	walkway_accessibility	VARCHAR(50)
	walkway_surface	VARCHAR(50)
PEDESTRIAN_CROSSING	crossing_id	INT
	intersection_id	INT
	crossing_type	VARCHAR(50)
	crossing_lights	VARCHAR(50)
	crossing_signal	VARCHAR(50)
PEDESTRIAN_DATA	pdata_id	INT
	walkway_id	INT
	pdata_datetime	DATE
	pdata_average_speed	NUMBER(8,3)
	pdata_pedestrian_volume	INT
	pdata_congestion_level	INT

3.2. Relationships

Entity 1	Entity 2	Relationship Description
LOCATION	SATELLITE_VIEW	Locations have satellite views, and may have more than one view.
LOCATION	STREET	Locations have one or many streets.
LOCATION	STREET_SEGMENT	Street segments have start and end locations to provide exact starts and ends.
LOCATION	PUBLIC_TRANS_STOP	Locations may have one or more public transport stops.
LOCATION	ROAD_OBJECT	Locations may have one or more road objects located in them.
LOCATION	POINT_OF_INTEREST	Locations may have one or more points of interest(s).
LOCATION	INTERSECTION	Locations may contain intersections.
STREET	STREET_SEGMENT	Streets have one or more street segments, and segments have a direction.
STREET	POINT_OF_INTEREST	Points of interest are located in streets.
STREET	INTERSECTION	Streets may have intersections.
STREET_SEGMENT	ROAD_EVENT	Street segments may have multiple road events.
STREET_SEGMENT	PUBLIC_TRANS_STOP	Street segments may contain public transport stops.
STREET_SEGMENT	SPEED_LIMIT	Street segments have speed limits, but also different speed limits at different times of the day.
STREET_SEGMENT	ROAD_OBJECT	Street segments contain road objects that create obstructions or add information to segments.
STREET_SEGMENT	LIVE_TRAFFIC_DATA	Specific street segments are used for recording traffic data to ensure accuracy.
STREET_SEGMENT	TRAFFIC_DATA_PREDICTOR	Traffic can be predicted on segments of streets based on historical data.
STREET_SEGMENT	TRAFFIC_CONDITION	Street segments (parts of a street) will have different traffic conditions.
STREET_SEGMENT	STREET_VIEW	Street segments have street views to give various views of the same street.
STREET_SEGMENT	ROAD_CONDITION	Street segments have different road conditions, as conditions may vary on the same street.
STREET_SEGMENT	INTERSECTING_SEGMENTS	Street segments may intersect with other street segments.
STREET_SEGMENT	PEDESTRIAN_WALKWAY	Street segments may have pedestrian walkways and different accessibility options.
LIVE_TRAFFIC_DATA	CONGESTION_LEVEL	Traffic data have different levels of congestion at different times of the day.
INTERSECTION	INTERSECTING_SEGMENTS	Intersections connect street segments.
INTERSECTION	PEDESTRIAN_CROSSING	Intersections may have pedestrian crossings and crossings have different signs and lights.
PEDESTRIAN_WALKWAY	PEDESTRIAN_DATA	Pedestrian data is recorded on different pedestrian walkways.

3.3. Logical Design Model

STREET_SEGMENT (segment_id (PK), *street_id (FK)*, *segment_start_loc_id (FK)*, *segment_end_loc_id (FK)*, segment_length, segment_direction, segment_road_type)

STREET (street_id (PK), *loc_id (FK)*, street_name)

STREET_VIEW (street_view_id (PK), *segment_id (FK)*, street_view_url)

INTERSECTION (intersection_id (PK), *first_street_id (FK1)*, *second_street_id (FK2)*, *intersection_loc_id (FK3)*)

INTERSECTION_SEGMENTS (segment_id (PK, FK1), intersection_start_id (PK, FK2), intersection_end_id (PK, FK3))

PEDESTRIAN_CROSSING (crossing_id (PK), *intersection_id (FK)*, crossing_type, crossing_lights, crossing_signal)

LOCATION (loc_id (PK), loc_name, loc_latitude, loc_longitude, loc_city, loc_province, loc_country, loc_postal)

SATELLITE_VIEW (satellite_view_id (PK), *loc_id (FK)*, satellite_view_url)

POINT_OF_INTEREST (poi_id (PK), *loc_id (FK1)*, *street_id (FK2)*, poi_name, poi_description, poi_category)

PUBLIC_TRANS_STOP (ptstop_id (PK), *loc_id (FK)*, *segment_id (FK)*, ptstop_name)

PEDESTRIAN_WALKWAY (walkway_id (PK), *segment_id (FK)*, walkway_type, walkway_width, walkway_accessibility, walkway_surface)

PEDESTRIAN_DATA (pdata_id (PK), *walkway_id (FK)*, pdata_datetime, pdata_pedestrian_volume, pdata_congestion_level, pdata_travel_time)

SPEED_LIMIT (slimit_id (PK), *segment_id (FK)*, slimit_value, slimit_start_datetime, slimit_end_datetime)

ROAD_EVENT (event_id (PK), *segment_id (FK)*, event_type, event_description, event_start_datetime, event_end_datetime)

ROAD_OBJECT (object_id (PK), *loc_id (FK)*, *segment_id (FK)*, object_type, object_description)

LIVE_TRAFFIC_DATA (tdata_id (PK), *segment_id (FK)*, *congestion_id (FK)*, tdata_datetime, tdata_average_speed, tdata_traffic_volume, tdata_live_travel_time)

CONGESTION_LEVEL (congestion_id (PK), congestion_type, congestion_description)

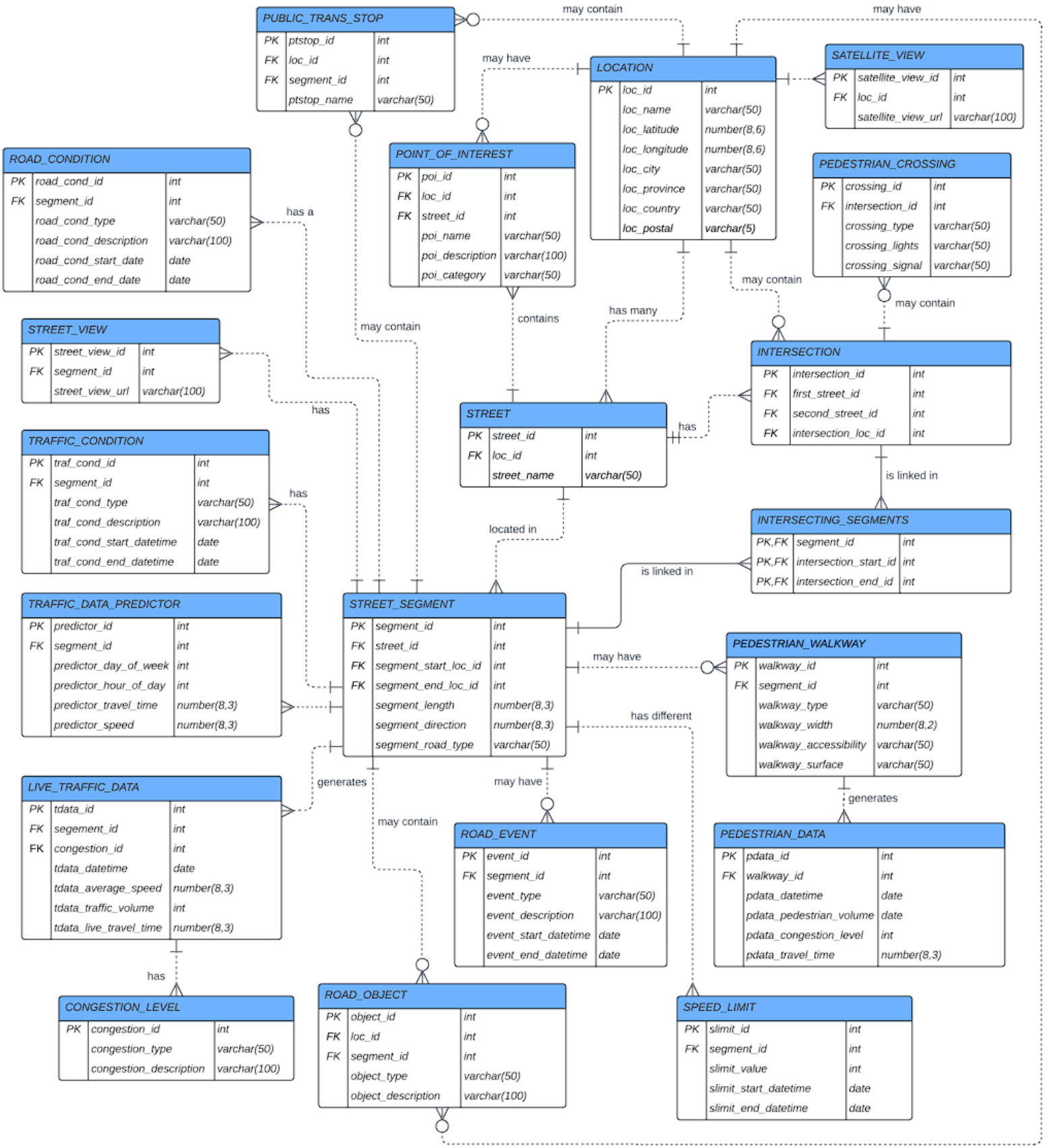
TRAFFIC_DATA_PREDICTOR (predictor_id (PK), *segment_id (FK)*, predictor_day_of_week, predictor_hour_of_day, predictor_travel_time, predictor_speed)

TRAFFIC_CONDITION (traf_cond_id (PK), *segment_id (FK)*, traf_cond_type, traf_cond_description, traf_cond_start_datetime, traf_cond_end_datetime)

ROAD_CONDITION (road_cond_id (PK), *segment_id (FK)*, road_cond_type, road_cond_description, road_cond_start_date, road_cond_end_date)

3.4. Entity Relationship Diagram

Integrated Traffic Management Database System ERD



4. SQL

4.1. Oracle SQL Code

```
CREATE SEQUENCE seq_Location
MINVALUE 1
START WITH 1
INCREMENT BY 1
NOCACHE
NOCYCLE;

CREATE TABLE LOCATION(
    loc_id INT NOT NULL CONSTRAINT Location_PK PRIMARY KEY,
    loc_name VARCHAR(50),
    loc_latitude NUMBER(8,6) NOT NULL,
    loc_longitude NUMBER(8,6) NOT NULL,
    loc_city VARCHAR(50) NOT NULL,
    loc_province VARCHAR(50) NOT NULL,
    loc_country VARCHAR(50) NOT NULL,
    loc_postal VARCHAR(5) NOT NULL
);

CREATE SEQUENCE seq_SatelliteView
MINVALUE 1
START WITH 1
INCREMENT BY 1
NOCACHE
NOCYCLE;

CREATE TABLE SATELLITE_VIEW(
    satellite_view_id INT NOT NULL CONSTRAINT SatelliteView_PK PRIMARY KEY,
    loc_id INT NOT NULL CONSTRAINT SatelliteView_Location_FK REFERENCES LOCATION(loc_id),
    satellite_view_url VARCHAR(100) NOT NULL
);

CREATE SEQUENCE seq_Street
MINVALUE 1
START WITH 1
INCREMENT BY 1
NOCACHE
NOCYCLE;

CREATE TABLE STREET(
    street_id INT NOT NULL CONSTRAINT Street_PK PRIMARY KEY,
    loc_id INT NOT NULL CONSTRAINT Street_Location_FK REFERENCES LOCATION(loc_id),
    street_name VARCHAR(50) NOT NULL
);

CREATE SEQUENCE seq_Segment
MINVALUE 1
START WITH 1
INCREMENT BY 1
NOCACHE
NOCYCLE;

CREATE TABLE STREET_SEGMENT(
    segment_id INT NOT NULL CONSTRAINT Segment_PK PRIMARY KEY,
    street_id INT NOT NULL CONSTRAINT Segment_Street_FK REFERENCES STREET(street_id),
    segment_length NUMBER(8,3) NOT NULL,
    segment_direction NUMBER(8,3) NOT NULL,
    segment_road_type VARCHAR(50) NOT NULL,
    segment_start_loc_id INT NOT NULL CONSTRAINT Segment_StartLoc_FK REFERENCES LOCATION(loc_id),
    segment_end_loc_id INT NOT NULL CONSTRAINT Segment_EndLoc_FK REFERENCES LOCATION(loc_id)
);

CREATE SEQUENCE seq_Event
MINVALUE 1
START WITH 1
INCREMENT BY 1
NOCACHE
NOCYCLE;

CREATE TABLE ROAD_EVENT(
    event_id INT NOT NULL CONSTRAINT Event_PK PRIMARY KEY,
    segment_id INT NOT NULL CONSTRAINT Event_Segment_FK REFERENCES STREET_SEGMENT(segment_id),
    event_type VARCHAR(50) NOT NULL,
    event_description VARCHAR(100) NOT NULL,
    event_start_datetime DATE NOT NULL,
    event_end_datetime DATE NOT NULL
);
```

```
77
78  CREATE SEQUENCE seq_PTStop
79  MINVALUE 1
80  START WITH 1
81  INCREMENT BY 1
82  NOCACHE
83  NOCYCLE;
84
85  CREATE TABLE PUBLIC_TRANS_STOP(
86      ptstop_id INT NOT NULL CONSTRAINT PTStop_PK PRIMARY KEY,
87      segment_id INT NOT NULL CONSTRAINT PTStop_Segment_FK REFERENCES STREET_SEGMENT(segment_id),
88      ptstop_name VARCHAR(50),
89      ptstop_type VARCHAR(50) NOT NULL,
90      ptstop_loc_id INT NOT NULL CONSTRAINT PTStop_Loc_FK REFERENCES LOCATION(loc_id)
91  );
92
93  CREATE SEQUENCE seq_SpeedLimit
94  MINVALUE 1
95  START WITH 1
96  INCREMENT BY 1
97  NOCACHE
98  NOCYCLE;
99
100 CREATE TABLE SPEED_LIMIT(
101     slimit_id INT NOT NULL CONSTRAINT SLimit_PK PRIMARY KEY,
102     segment_id INT NOT NULL CONSTRAINT SpeedLimit_Segment_FK REFERENCES STREET_SEGMENT(segment_id),
103     slimit_value INT NOT NULL,
104     slimit_start_time DATE,
105     slimit_end_time DATE
106 );
107
108 CREATE SEQUENCE seq_RoadObject
109 MINVALUE 1
110 START WITH 1
111 INCREMENT BY 1
112 NOCACHE
113 NOCYCLE;
114
115 CREATE TABLE ROAD_OBJECT(
116     object_id INT NOT NULL CONSTRAINT Object_PK PRIMARY KEY,
117     segment_id INT NOT NULL CONSTRAINT Object_Segment_FK REFERENCES STREET_SEGMENT(segment_id),
118     object_type VARCHAR(50) NOT NULL,
119     object_description VARCHAR(100) NOT NULL,
120     object_loc_id INT NOT NULL CONSTRAINT Object_Loc_FK REFERENCES LOCATION(loc_id)
121 );
122
123 CREATE SEQUENCE seq_Congestion
124 MINVALUE 1
125 START WITH 1
126 INCREMENT BY 1
127 NOCACHE
128 NOCYCLE;
129
130 CREATE TABLE CONGESTION_LEVEL(
131     congestion_id INT NOT NULL CONSTRAINT Congestion_PK PRIMARY KEY,
132     congestion_type VARCHAR(50) NOT NULL,
133     congestion_description VARCHAR(100) NOT NULL
134 );
135
136 CREATE SEQUENCE seq_TrafficData
137 MINVALUE 1
138 START WITH 1
139 INCREMENT BY 1
140 NOCACHE
141 NOCYCLE;
142
143 CREATE TABLE LIVE_TRAFFIC_DATA(
144     tdata_id INT NOT NULL CONSTRAINT TData_PK PRIMARY KEY,
145     segment_id INT NOT NULL CONSTRAINT TData_Segment_FK REFERENCES STREET_SEGMENT(segment_id),
146     congestion_id INT NOT NULL CONSTRAINT TData_Congestion_FK REFERENCES CONGESTION_LEVEL(congestion_id),
147     tdata_datetime DATE NOT NULL,
148     tdata_average_speed NUMBER(8,3) NOT NULL,
149     tdata_traffic_volume INT NOT NULL,
150     tdata_live_travel_time NUMBER(8,3) NOT NULL
151 );
```

```
152
153   CREATE SEQUENCE seq_TrafficPred
154   MINVALUE 1
155   START WITH 1
156   INCREMENT BY 1
157   NOCACHE
158   NOCYCLE;
159
160   CREATE TABLE TRAFFIC_DATA_PREDICTOR(
161       predictor_id INT NOT NULL CONSTRAINT TDataPred_PK PRIMARY KEY,
162       segment_id INT NOT NULL CONSTRAINT TDataPred_Segment_FK REFERENCES STREET_SEGMENT(segment_id),
163       predictor_day_of_week INT NOT NULL,
164       predictor_hour_of_day INT NOT NULL,
165       predictor_travel_time NUMBER(8,3) NOT NULL,
166       predictor_speed NUMBER(8,3) NOT NULL
167   );
168
169   CREATE SEQUENCE seq_TrafficCond
170   MINVALUE 1
171   START WITH 1
172   INCREMENT BY 1
173   NOCACHE
174   NOCYCLE;
175
176   CREATE TABLE TRAFFIC_CONDITION(
177       traf_cond_id INT NOT NULL CONSTRAINT TrafCond_PK PRIMARY KEY,
178       segment_id INT NOT NULL CONSTRAINT TrafCond_Segment_FK REFERENCES STREET_SEGMENT(segment_id),
179       traf_cond_type VARCHAR(50) NOT NULL,
180       traf_cond_description VARCHAR(100) NOT NULL,
181       traf_cond_start_datetime DATE NOT NULL,
182       traf_cond_end_datetime DATE
183   );
184
185   CREATE SEQUENCE seq_StreetView
186   MINVALUE 1
187   START WITH 1
188   INCREMENT BY 1
189   NOCACHE
190   NOCYCLE;
191
192   CREATE TABLE STREET_VIEW(
193       street_view_id INT NOT NULL CONSTRAINT StreetView_PK PRIMARY KEY,
194       segment_id INT NOT NULL CONSTRAINT StreetView_Segment_FK REFERENCES STREET_SEGMENT(segment_id),
195       street_view_url VARCHAR(100) NOT NULL
196   );
197
198   CREATE SEQUENCE seq_RoadCond
199   MINVALUE 1
200   START WITH 1
201   INCREMENT BY 1
202   NOCACHE
203   NOCYCLE;
204
205   CREATE TABLE ROAD_CONDITION(
206       road_cond_id INT NOT NULL CONSTRAINT RoadCond_PK PRIMARY KEY,
207       segment_id INT NOT NULL CONSTRAINT RoadCond_Segment_FK REFERENCES STREET_SEGMENT(segment_id),
208       road_cond_type VARCHAR(50) NOT NULL,
209       road_cond_description VARCHAR(100) NOT NULL,
210       road_cond_start_datetime DATE NOT NULL,
211       road_cond_end_datetime DATE
212   );
213
214   CREATE SEQUENCE seq_POI
215   MINVALUE 1
216   START WITH 1
217   INCREMENT BY 1
218   NOCACHE
219   NOCYCLE;
220
221   CREATE TABLE POINT_OF_INTEREST(
222       poi_id INT NOT NULL CONSTRAINT POI_PK PRIMARY KEY,
223       street_id INT NOT NULL CONSTRAINT POI_Street_FK REFERENCES STREET(street_id),
224       poi_name VARCHAR(50) NOT NULL,
225       poi_description VARCHAR(100) NOT NULL,
226       poi_category VARCHAR(50) NOT NULL,
227       poi_loc_id INT NOT NULL CONSTRAINT POI_Loc_FK REFERENCES LOCATION(loc_id)
228   );
229
```

```
CREATE SEQUENCE seq_Intersection
MINVALUE 1
START WITH 1
INCREMENT BY 1
NOCACHE
NOCYCLE;

CREATE TABLE INTERSECTION(
    intersection_id INT NOT NULL CONSTRAINT Intersection_PK PRIMARY KEY,
    first_street_id INT NOT NULL CONSTRAINT Intersection_FirstStreet_FK REFERENCES STREET(street_id),
    second_street_id INT NOT NULL CONSTRAINT Intersection_SecondStreet_FK REFERENCES STREET(street_id),
    intersection_loc_id INT NOT NULL CONSTRAINT Intersection_Loc_FK REFERENCES LOCATION(loc_id)
);

CREATE TABLE INTERSECTING_SEGMENTS(
    segment_id INT NOT NULL CONSTRAINT Intersecting_Segment_FK REFERENCES STREET_SEGMENT(segment_id),
    intersection_start_id INT NOT NULL CONSTRAINT Intersecting_Start_FK REFERENCES INTERSECTION(intersection_id),
    intersection_end_id INT NOT NULL CONSTRAINT Intersecting_End_FK REFERENCES INTERSECTION(intersection_id),
    CONSTRAINT Intersecting_PK PRIMARY KEY (segment_id, intersection_start_id, intersection_end_id)
);

CREATE SEQUENCE seq_PedestrianWalkway
MINVALUE 1
START WITH 1
INCREMENT BY 1
NOCACHE
NOCYCLE;

CREATE TABLE PEDESTRIAN_WALKWAY(
    walkway_id INT NOT NULL CONSTRAINT Walkway_PK PRIMARY KEY,
    segment_id INT NOT NULL CONSTRAINT Walkway_Segment_FK REFERENCES STREET_SEGMENT(segment_id),
    walkway_type VARCHAR(50) NOT NULL,
    walkway_width NUMBER(8,3) NOT NULL,
    walkway_accessibility VARCHAR(50) NOT NULL,
    walkway_surface VARCHAR(50) NOT NULL
);

CREATE SEQUENCE seq_PedestrianCrossing
MINVALUE 1
START WITH 1
INCREMENT BY 1
NOCACHE
NOCYCLE;

CREATE TABLE PEDESTRIAN_CROSSING(
    crossing_id INT NOT NULL CONSTRAINT Crossing_PK PRIMARY KEY,
    intersection_id INT NOT NULL CONSTRAINT PedCross_Intersection_FK REFERENCES INTERSECTION(intersection_id),
    crossing_type VARCHAR(50) NOT NULL,
    crossing_lights VARCHAR(50),
    crossing_signal VARCHAR(50)
);

CREATE SEQUENCE seq_PedestrianData
MINVALUE 1
START WITH 1
INCREMENT BY 1
NOCACHE
NOCYCLE;

CREATE TABLE PEDESTRIAN_DATA(
    pdata_id INT NOT NULL CONSTRAINT PData_PK PRIMARY KEY,
    walkway_id INT NOT NULL CONSTRAINT PData_Walkway_FK REFERENCES PEDESTRIAN_WALKWAY(walkway_id),
    pdata_datetime DATE NOT NULL,
    pdata_average_speed NUMBER(8,3) NOT NULL,
    pdata_pedestrian_volume INT NOT NULL,
    pdata_congestion_level INT NOT NULL
);
```

4.2. Database Schema