UNIVERSITI TEKNOLOGI MALAYSIA

SECJ3553 ARTIFICIAL INTELLIGENCE

SEMESTER 1 2024/2025

ASSIGNMENT 1

(ROUTESMART - AI-DRIVEN ROAD NAVIGATION APPLICATION)

SECTION: 02

LECTURER: ASSOCIATE PROF. DR. MOHD ADHAM ISA

GROUP NAME: LogiCode

NO.	NAME	MATRIC NO.
1.	KOH LI HUI	A22EC0059
2.	KOH SU XUAN	A22EC0060
3.	MAISARAH BINTI RIZAL (LEADER)	A22EC0192
4.	TANG YAN QING	A22EC0109

TABLE OF CONTENT

1.0 Knowledge Representation

1.1. Determine Route Adjustment for Congestion

IF traffic_congestion = TRUE AND auto_adjust_route = ON THEN route_status = SET adjusted

Explanation:

When the system detects traffic congestion and the auto-adjustment feature is enabled, it will automatically modify the current route to avoid the congested area. This ensures that users are guided along a less congested path for a smoother travel experience.

First Order Logic(FOL):

T(x): Traffic condition x is congested

A(y): Auto-adjust feature y is ON

R(z): Route z status is SET adjusted

$\forall x, y, z\ ((T(x) \wedge A(y)) \rightarrow R(z))$

1.2. Update Destination Route

IF user_input_received = TRUE AND (input_type = voice AND hands_free_mode = ON) THEN navigation_update = SET by_voice.

IF user_input_received = TRUE AND (input_type = typing AND hands_free_mode = OFF) THEN navigation_update = SET by_typing.

IF user_input_received = TRUE AND (input_type = voice AND hands_free_mode = OFF) THEN navigation_update = SET by_voice with caution warning.

Explanation:

This rule ensures the system updates navigation based on user input while considering safety. If typing input is received and hands-free mode is off, the navigation is updated manually. For voice input with hands-free mode enabled, the system updates navigation safely without manual intervention. However, if voice input is received while hands-free mode is off, the navigation still updates but includes a caution warning to alert the user of potential safety risks due to manual involvement. This approach balances flexibility with safety across different input methods.

First Order Logic(FOL):

U(x): User input x is received.

T(y): Input type y is typing.

V(z): Input type z is voice.

H(w): Hands-free mode w is ON.

A(v): Navigation update by typing.

B(v): Navigation update by voice.

C(v): Navigation update by voice with caution warning.

$\forall x,z,w,v((U(x) \wedge V(z) \wedge H(w)) \rightarrow B(v))$

$\forall x,y,w,v((U(x) \wedge T(y) \wedge \neg H(w)) \rightarrow A(v))$

$\forall x,z,w,v((U(x) \wedge V(z) \wedge \neg H(w)) \rightarrow C(v))$

1.3. Recommended Departure Time

IF user_input_received = TRUE AND destination_set = TRUE AND arrival_time_specified = TRUE THEN travel_time_estimation = START

IF travel_time_estimation = DONE AND traffic_condition = CLEAR THEN departure_time = (arrival_time - estimated_travel_time)

IF travel_time_estimation = DONE AND traffic_condition = CONGESTED THEN departure_time = (arrival_time - (estimated_travel_time + traffic_delay))

IF updated_traffic_data = RECEIVED AND departure_time_set = TRUE AND new_traffic_delay_detected = TRUE THEN departure_time = RECALCULATE

IF departure_time = CALCULATED AND departure_time < current_time THEN notify_user = LEAVE_NOW

Explanation:

When the user enters the destination they want to go to and the time they want to arrive, the system will recommend the best route to avoid traffic congestion. The system suggests the optimal departure time based on current data by subtracting the travel time from the desired arrival time. If there is a traffic jam, the system will update the route to ensure the user can still arrive at the specified time.

First Order Logic(FOL):

U(x): User input x is received.

D(x): Destination x is set.

A(x,t) : Desired arrival time t for user x is specified.

E(t): Estimated travel time is t.

C(c): Traffic condition c (either CLEAR or CONGESTED).

T(d): Traffic delay is d.

L(x,d): Departure time d is calculated for user x.

R(x,d): Recalculate departure time d for user x.

N(x): Notify user x to leave immediately.

$\forall x,t(U(x) \wedge D(x) \wedge A(x,t) \rightarrow E(t))$

$\forall x,t,e(E(e) \wedge C(CLEAR) \rightarrow L(x,t-e))$

$\forall x,t,e,d(E(e) \wedge C(CONGESTED) \wedge T(d) \rightarrow L(x,t-(e+d)))$

$\forall x,d(T(d) \wedge L(x,d) \wedge new\_traffic\_data(x) \rightarrow R(x,d))$

$\forall x,d,c(L(x,d) \wedge d<current\_time(c) \rightarrow N(x))$

1.4. Include Nearby Amenities in Route

IF user requests nearby amenities = TRUE AND amenities_set = TRUE THEN route adjustment = INCLUDE amenity in route.

Explanation:

When the user specifies a need for any nearby amenity (e.g., gas station, restaurant, restroom, charging station), the system checks if amenities are available (indicated by amenities_set = TRUE). If available, the system automatically adjusts the route to include the nearest amenity. This design provides flexibility, enabling the user to add any type of stop they may need without being limited to specific types. The route will be seamlessly integrated with the selected amenity, then resume the original plan after the stop.

First Order Logic(FOL):
P(x): user x requests nearby amenities
Q(a): amenities a are set (available)
R(x,a): include amenity a in route for user x

$\forall x,a(P(x) \wedge Q(a) \rightarrow R(x,a))$
$\forall x(\text{request\_amenity}(x) \wedge \text{amenities\_set}(a) \rightarrow \text{include\_POI}(x,a))$

1.5. Respond to Traffic Incidents

IF traffic incident detected on route = TRUE AND user is on affected route = TRUE THEN notify user = ALERT AND suggest route = ALTERNATIVE.

Explanation:
This feature allows the system to proactively monitor and respond to unexpected traffic incidents (e.g., accidents, road closures) on the user's current route. If an incident occurs and the user is on the affected route, the system issues an alert notification and suggests an alternative route to avoid the delay. This real-time incident response aims to provide a smoother and more efficient navigation experience for the user, minimizing travel disruptions.

First Order Logic(FOL):
P(r): traffic incident detected on route r
Q(x): user x is on route r
R(x,r′): system suggests alternative route r′ for user x
N(x): system notifies user x with an alert

$\forall x,r,r'(P(r) \wedge Q(x) \rightarrow (N(x) \wedge R(x,r')))$
$\forall x,r\ (\text{traffic\_incident}(r) \wedge \text{on\_route}(x,r) \rightarrow (\text{notify}(x,\text{alert}) \wedge \text{suggest\_route}(x,\text{alternative})))$

1.6. Enable Emergency Route Priority

IF emergency_route_request = TRUE THEN prioritize_route = TRUE AND notify_users = TRUE

Explanation:

This feature helps ensure emergency vehicles can arrive at their destination as fast as possible. When an emergency route request is detected, the system sets prioritize_route to TRUE, thus marking the route as an emergency path. At the same time, notify_users will be set to TRUE to send alerts to nearby non-emergency users, suggesting they take alternative routes. This helps clear the way for emergency vehicles and improves response times.

First Order Logic(FOL):

E(x): Emergency route request for vehicle x is detected

M(r): Route r is marked as an emergency path

N(u): Notify non-emergency user u to suggest an alternative route

$\forall x,r,u(E(x)\rightarrow(M(r)\land N(u)))$