

PEEWEE

Live Demonstration

Group 2:
Eugenia, Ziqin, Chengyao, Guang, Aaron, Hamka

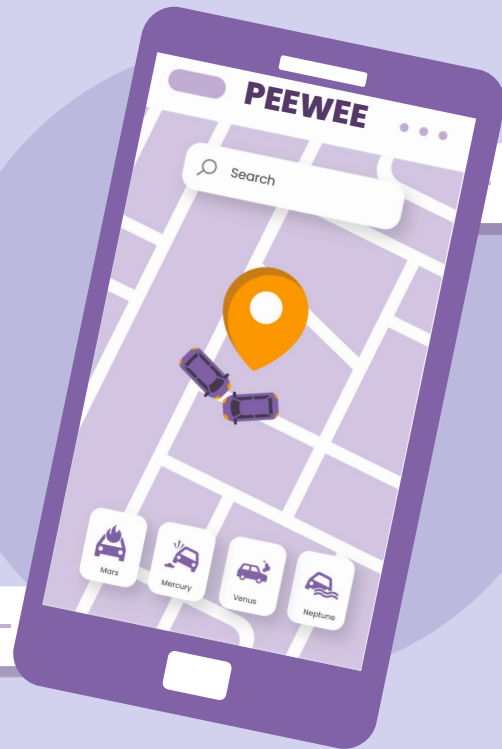




Table of contents

01

Introduction

02

Live Demo

03

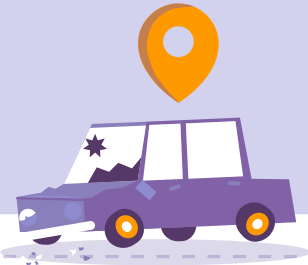
**SE Design
Practices**

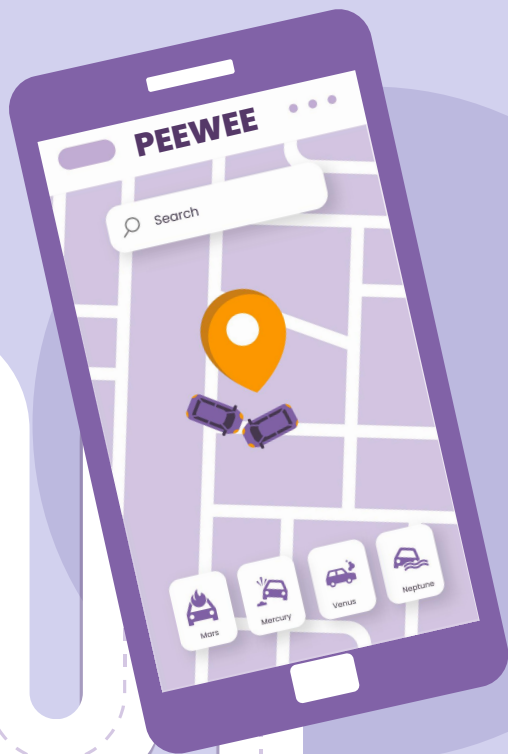
04

Traceability

05

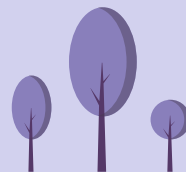
Conclusion





01

Introduction



Brief Overview

Problem Statement

According to a comprehensive study by TomTom in 2022:

- 56 out of 156 hours attributed to congestion
- 227 SGD out of 1118 SGD of private transport costs linked directly to traffic congestion.
- Peak hours increase average travel time for 10 kilometers to 23 minutes, from the 16-minute average recorded in 2022.



Target Audience

- Workers, students, and professionals reliant on efficient transportation during rush hours and traffic congestion.
- Commuters valuing convenience and seeking reliable tools to streamline their daily routines.

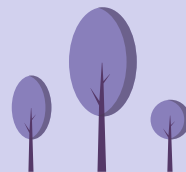
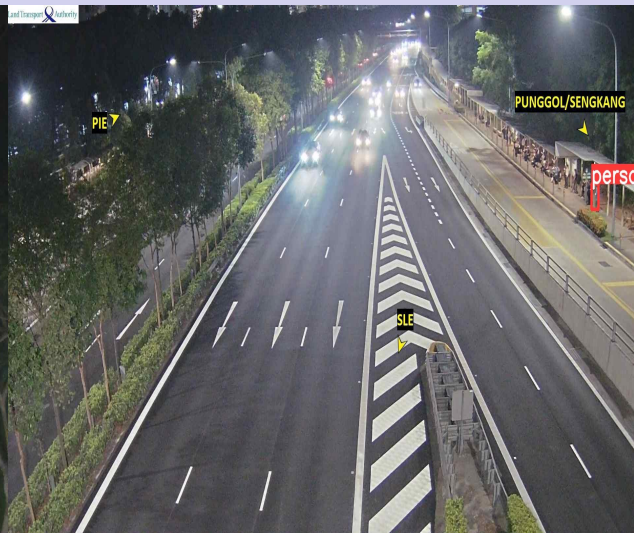
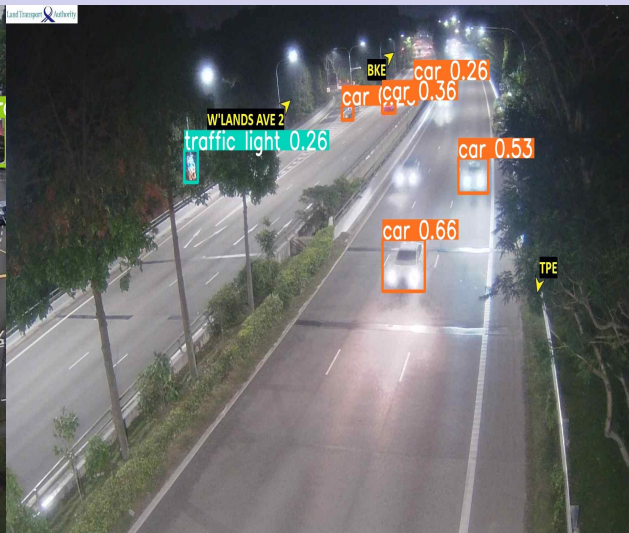
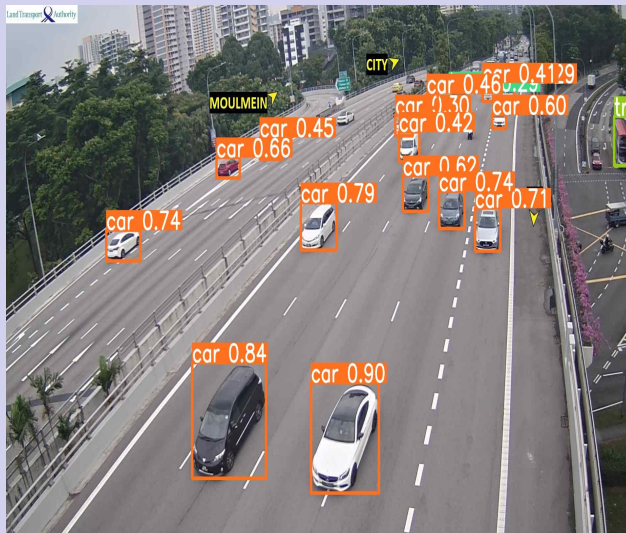


Overview of PEEWEE

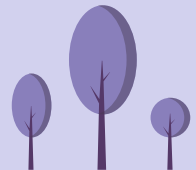
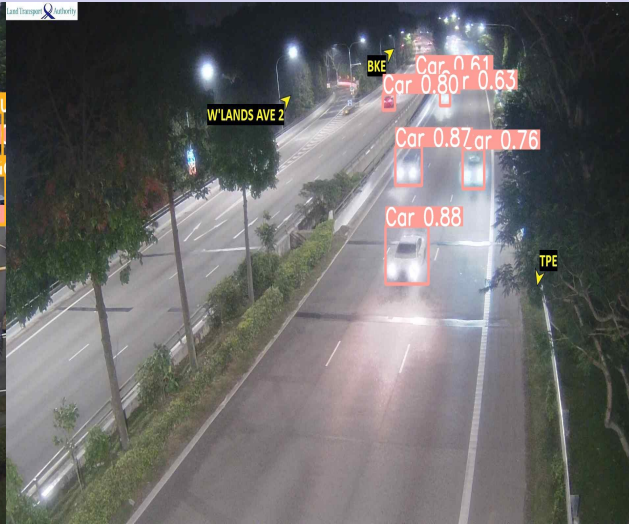
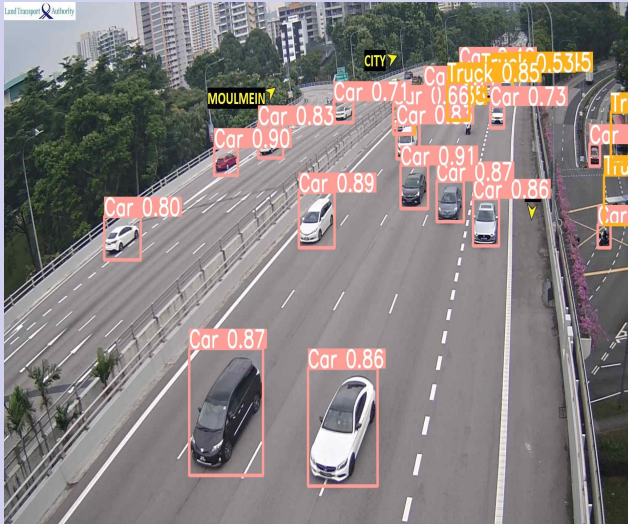
- Utilize Land Transport Authority (LTA) API to generate the links to latest images from traffic cameras all around Singapore
- Improve driver experiences with efficient navigation and timely information.



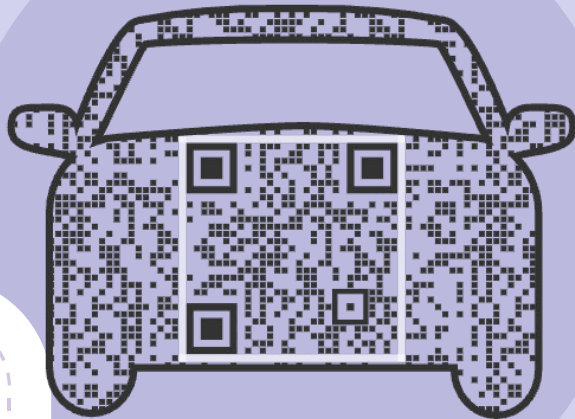
Untrained Data using YoloV8 Model



Trained Data with our PEEWEE model

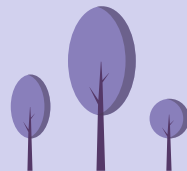


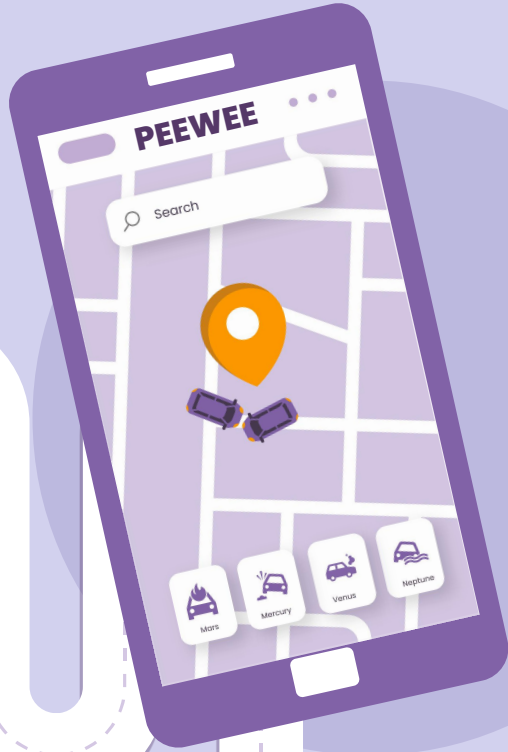
PEEWEE Demo



02

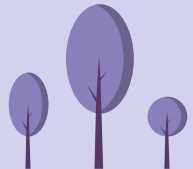
Live Demo



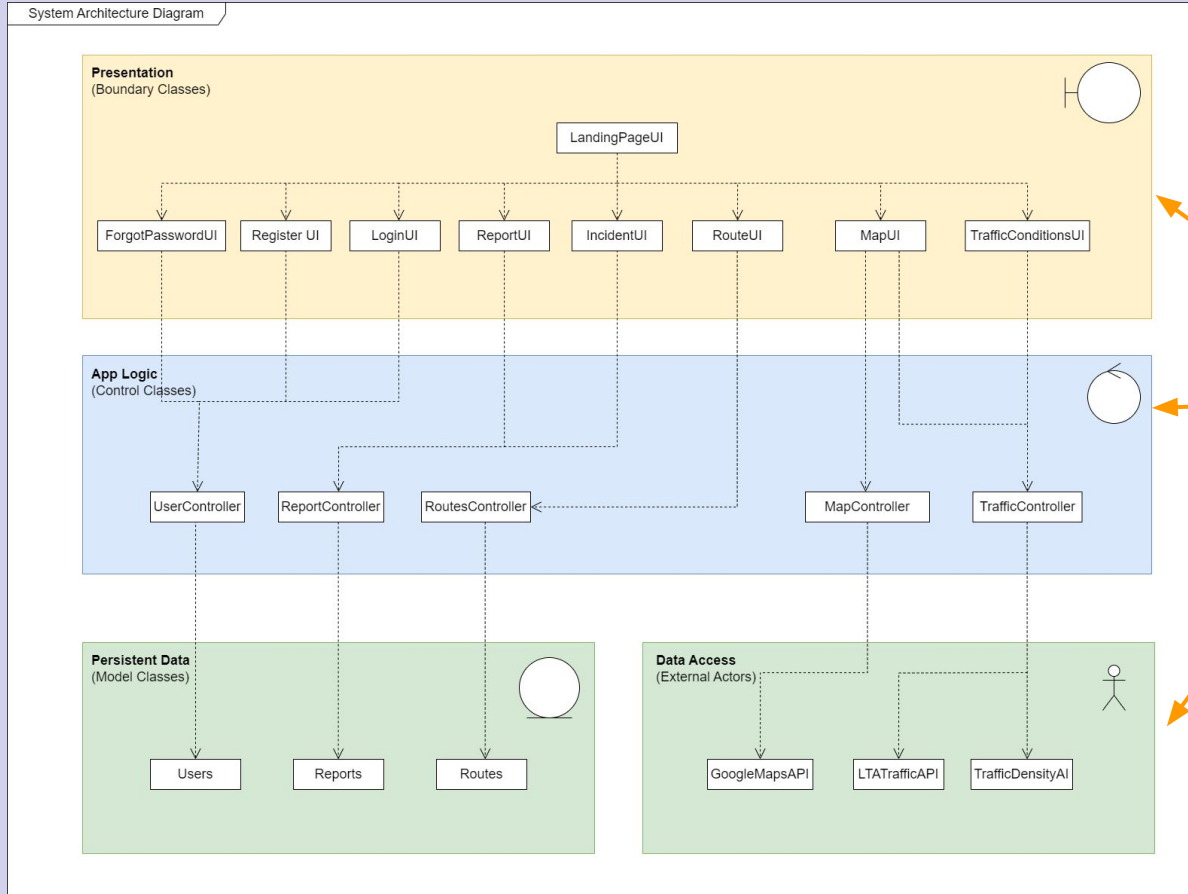


03

SE Design Practices



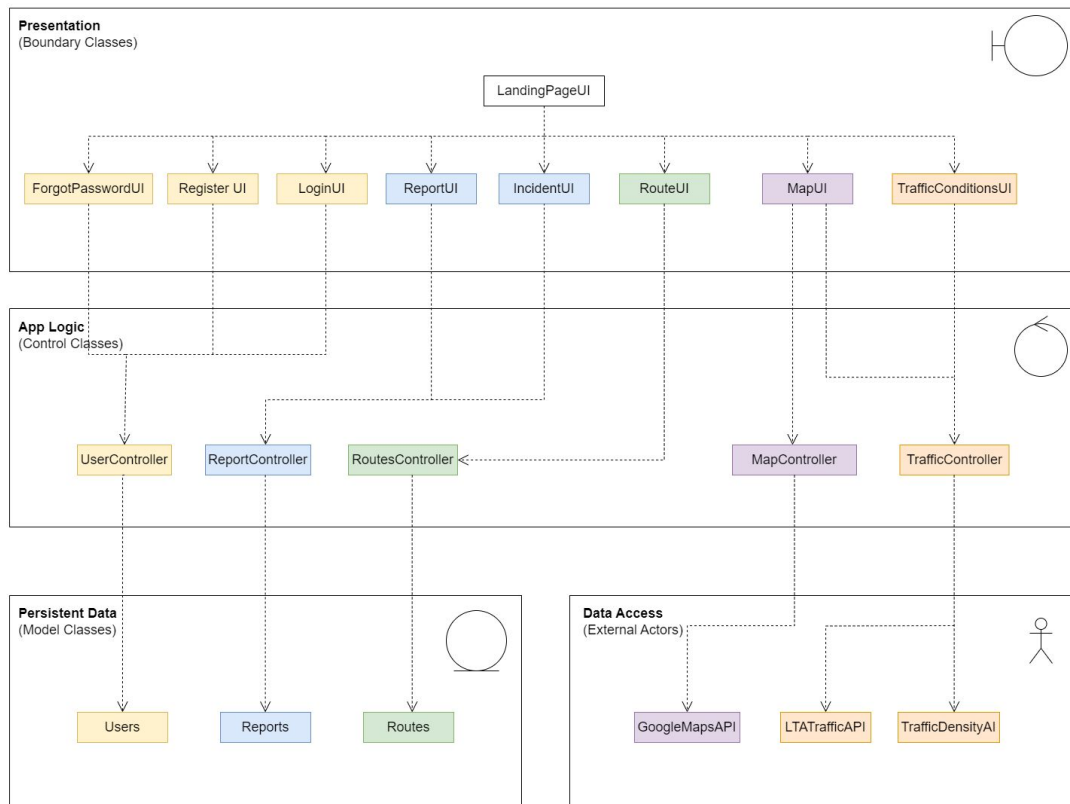
System Architecture



Group modules into presentation, app logic and persistent data layer

System Architecture

System Architecture Diagram



Single Responsibility Principle

- Clear segregation of responsibilities
- Ensures modules are dependent only on relevant modules

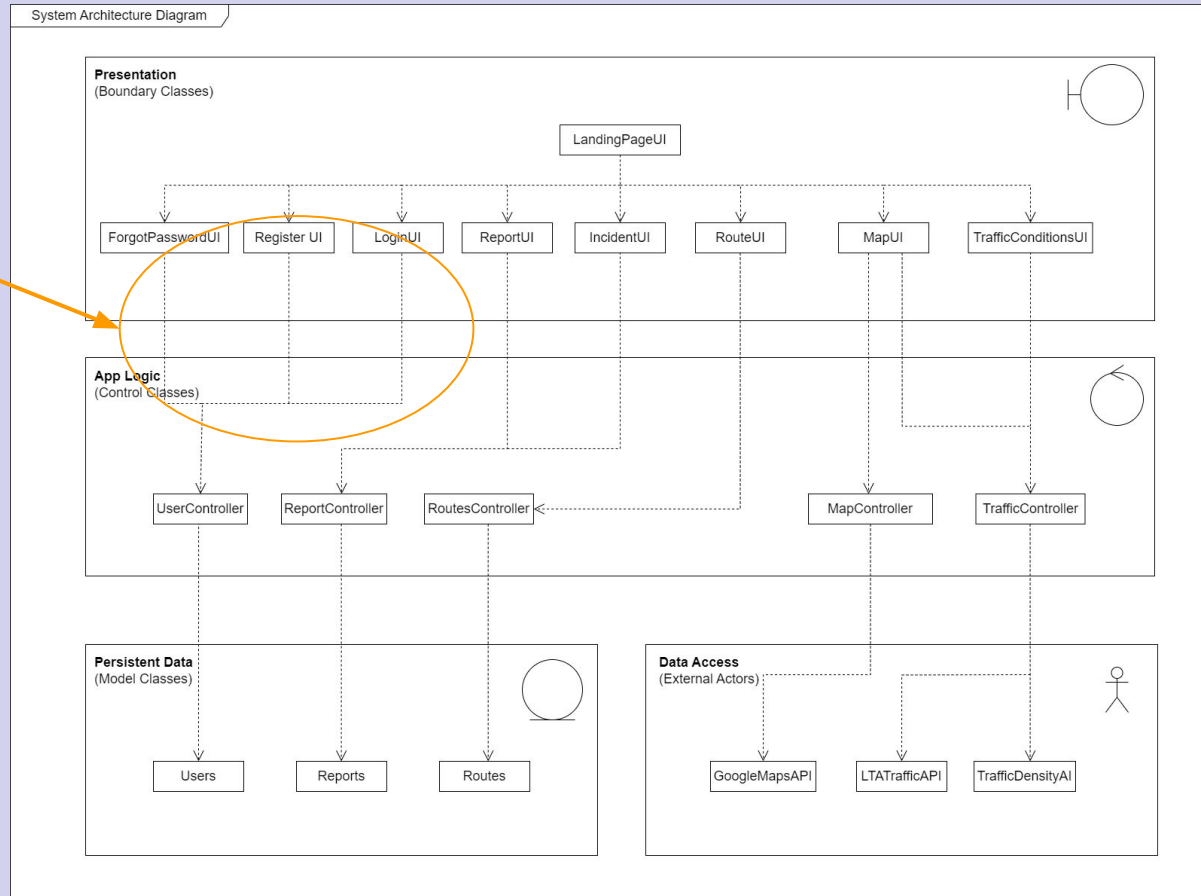
System Architecture

Loose Coupling

- Dependencies on a module kept to a max of 3

Maintainability

- Modules can be easily modified



Components

```
✓ components
  > ForgetPassword
  TS ActionAreaCard.tsx
  TS AppFrame.tsx
  TS AppFrameIndex.tsx
  TS CameraAPI.tsx
  TS ComponentsIndex.tsx
  TS FilterCamera.tsx
  TS FilterIncidents.tsx
  TS HeatmapDrawer.tsx
  TS IncidentListItem.tsx
  # map.css
  TS Map.tsx
  TS Navbar.tsx
  TS PhotoContainer.tsx
  TS Photos.tsx
  TS PrivateRoute.tsx
  TS SearchFilter.tsx
  TS SpecificTrafficChart.tsx
  TS StickyFooter.tsx
  TS TableIndex.tsx
  TS Title.tsx
  TS TrafficChart.tsx
```

Common UI elements used by multiple pages are organised as components



1. Promotes Code Reuse

```
app > src > pages > TS Dashboard.tsx > Dashboard
435 |         sx={{ mt: 3 }}
436 |       >
437 |       Go to map page
438 |     </Link>
439 |   </Stack>
440 |   <MapComponent
441 |     location={{
442 |       lng: 103.7992246,
443 |       lat: 1.3687004,
444 |       address: "Singapore",
445 |     }}
446 |     incidents={incidents}
447 |     zoomLevel={11}
448 |     cameras={cameras}
449 |     showHeatmap={true}
450 |     showCameras={false}
451 |     showAccidents={incidentFilters.includes("accident")}
452 |     showRoadClosures={incidentFilters.includes("roadClosure")}
453 |     showRoadWorks={incidentFilters.includes("roadWork")}
454 |   />
455 | </Paper>
456 | </Grid>
```

```
app > src > pages > TS Map.tsx > Map
380 |   <CssBaseline />
381 |   <AppFrame pageName="Map" maxWidth={false} disableGutters sx={{}}>
382 |     <Container sx={{ height: "90vh" }} maxWidth={false} disableGutters>
383 |       <MapComponent
384 |         location={{
385 |           lng: 103.7992246,
386 |           lat: 1.3687004,
387 |           address: "Singapore",
388 |         }}
389 |         zoomLevel={12}
390 |         cameras={cameras}
391 |         incidents={incidents}
392 |         directionsResponse={directionsResponse}
393 |         showHeatmap={trafficFilters.includes("heatmap")}
394 |         showCameras={trafficFilters.includes("camera")}
395 |         showAccidents={incidentFilters.includes("accident")}
396 |         showRoadClosures={incidentFilters.includes("roadClosure")}
397 |         showRoadWorks={incidentFilters.includes("roadWork")}
398 |       />
399 |     </Container>
400 |   </AppFrame>
```

```
app > src > pages > TS ViewRoute.tsx > ViewRoute
218 |   <ToggleButton value="show-all">Show all</ToggleButton>
219 |   <ToggleButton value="hide-all">Hide all</ToggleButton>
220 | </ToggleButtonGroup>
221 | </Container>
222 | <Container sx={{ height: "90vh" }}>
223 |   <MapComponent
224 |     location={{
225 |       lng: 103.7992246,
226 |       lat: 1.3687004,
227 |       address: "Singapore",
228 |     }}
229 |     zoomLevel={12}
230 |     cameras={cameras}
231 |     incidents={incidents}
232 |     directionsResponse={directionsResponse}
233 |     showHeatmap={trafficFilters.includes("heatmap")}
234 |     showCameras={trafficFilters.includes("camera")}
235 |     showAccidents={incidentFilters.includes("accident")}
236 |     showRoadClosures={incidentFilters.includes("roadClosure")}
237 |     showRoadWorks={incidentFilters.includes("roadWork")}
238 |   />
239 | </Container>
```

2. Maintainability

Changes to the Map component

```
app > src > pages > TS Dashboard.tsx > Dashboard
435     sx={{ mt: 3 }}
436   >
437   Go to map page
438   </Link>
439   </React>
440   <MapComponent
441     location={{
442       lng: 103.7992246,
443       lat: 1.3687004,
444       address: "Singapore",
445     }}
446     incidents={incidents}
447     zoomLevel={11}
448     cameras={cameras}
449     showHeatmap={true}
450     showCameras={false}
451     showAccidents={incidentFilters.includes("accident")}
452     showRoadClosures={incidentFilters.includes("roadClosure")}
453     showRoadWorks={incidentFilters.includes("roadWork")}
454   />
455   </Paper>
456 </Grid>
```

```
app > src > pages > TS Map.tsx > Map
380 <CssBaseline />
381 <AppFrame pageName="Map" maxWidth={false} disableGutters sx={{}}>
382   <Container sx={{ height: "90vh" }} maxWidth={false} disableGutters>
383     <MapComponent
384       location={{
385         lng: 103.7992246,
386         lat: 1.3687004,
387         address: "Singapore",
388       }}
389       zoomLevel={12}
390       cameras={cameras}
391       incidents={incidents}
392       directionsResponse={directionsResponse}
393       showHeatmap={trafficFilters.includes("heatmap")}
394       showCameras={trafficFilters.includes("camera")}
395       showAccidents={incidentFilters.includes("accident")}
396       showRoadClosures={incidentFilters.includes("roadClosure")}
397       showRoadWorks={incidentFilters.includes("roadWork")}
398     >
```

```
app > src > pages > TS ViewRoute.tsx > ViewRoute
218 <ToggleButton value="show-all">Show all</ToggleButton>
219 <ToggleButton value="hide-all">Hide all</ToggleButton>
220 </ToggleButtonGroup>
221 </Container>
222 <Container sx={{ height: "90vh" }}>
223   <MapComponent
224     location={{
225       lng: 103.7992246,
226       lat: 1.3687004,
227       address: "Singapore",
228     }}
229     zoomLevel={12}
230     cameras={cameras}
231     incidents={incidents}
232     directionsResponse={directionsResponse}
233     showHeatmap={trafficFilters.includes("heatmap")}
234     showCameras={trafficFilters.includes("camera")}
235     showAccidents={incidentFilters.includes("accident")}
236     showRoadClosures={incidentFilters.includes("roadClosure")}
237     showRoadWorks={incidentFilters.includes("roadWork")}
238   />
```

Changes reflected in all pages that use the component

3. Increases Code Readability

```
app > src > components > TS AppFrame.tsx > AppFrame
244
245   <Drawer
246     anchor="left"
247     variant="temporary"
248     ModalProps={{
249       keepMounted: true,
250     }}
251     onClose={toggleDrawer}
252     open={open}
253   >
254     {drawer}
255   </Drawer>
256 ) : (
257   <StyledDrawer variant="permanent" open={open}>
258     {drawer}
259   </StyledDrawer>
260 )
261 <Box
262   component="main"
263   sx={{
264     backgroundColor: (theme) =>
265       theme.palette.mode === "light"
266         ? theme.palette.grey[100]
267         : theme.palette.grey[900],
268     flexGrow: 1,
269     height: "100vh",
270     overflow: "auto",
271   }}
272 >
273   <Toolbar />
274   <Container maxWidth={maxWidth} sx={sx} {...props}>
275     {children}
276   </Container>
277 </Box>
278 </Box>
279
280 }
```

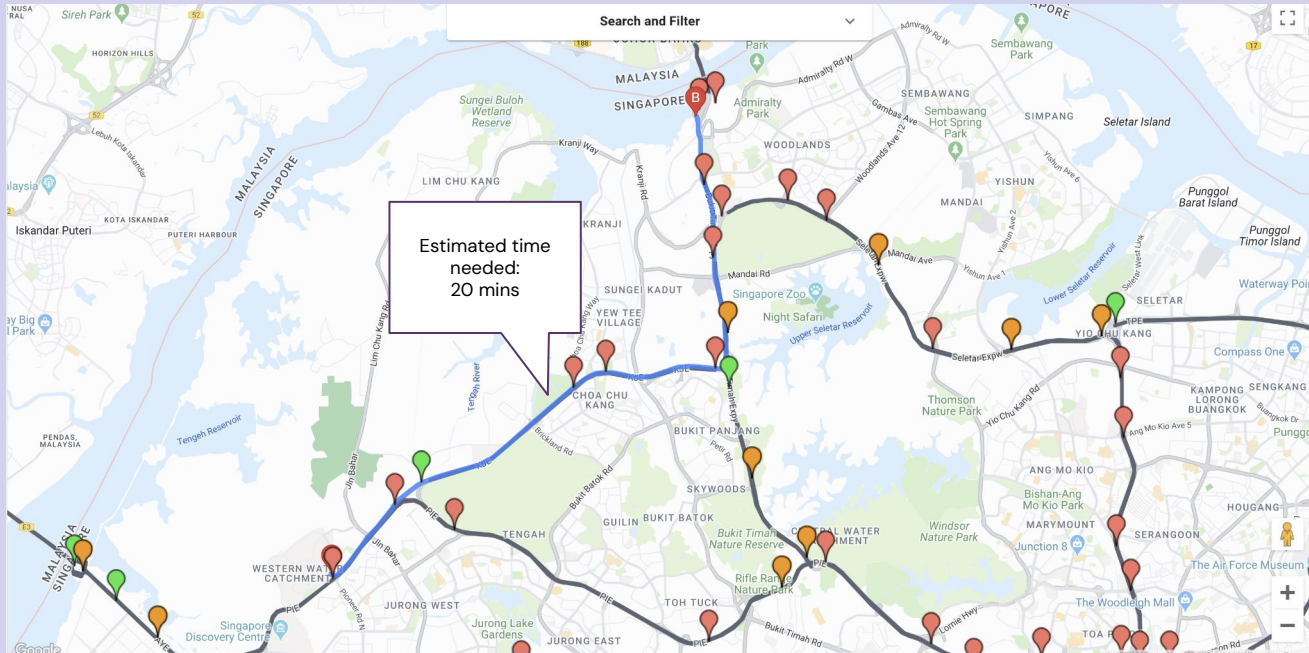
Implementation can be simply represented as `<Component />` instead of a whole sequence of codes

```
app > src > pages > TS Dashboard.tsx > Dashboard
401
402   return (
403     <ThemeProvider theme={defaultTheme}>
404       <CssBaseline />
405       <AppFrame pageName="Dashboard">
406         <Grid container spacing={3}>
```



Extensibility for future upgrades

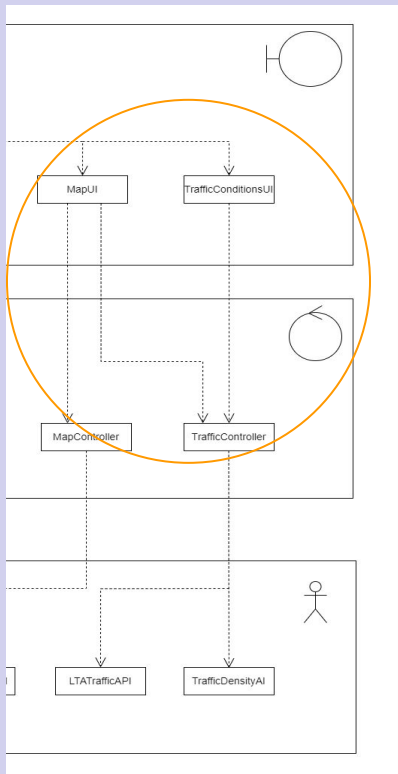
Example Extension : Compute and display time needed to travel along searched routes based on current traffic conditions

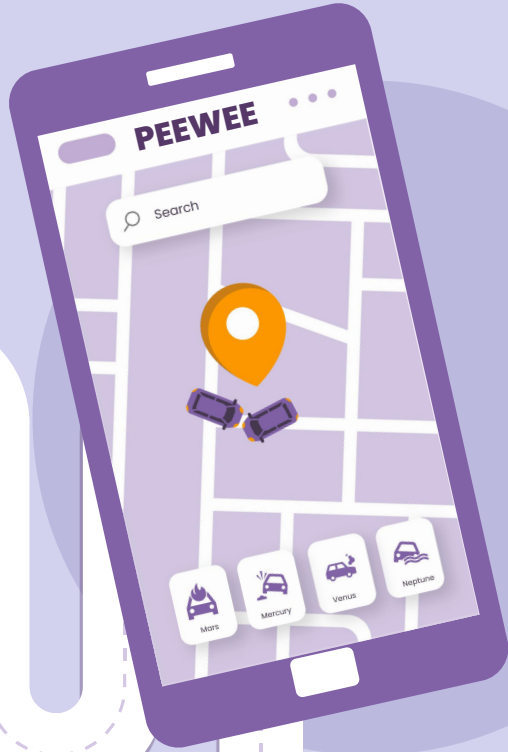


Extensibility for future upgrades

How our system design allows for extension with minimal changes:

- Loose Coupling
 - Modifications to TrafficController for time computation result in potential modifications to **only MapUI and TrafficConditions UI**
 - **Modules not dependent** on TrafficController are left **unchanged**
- Use of Components
 - New UI component for time display can be **easily imported and incorporated** to existing Map component



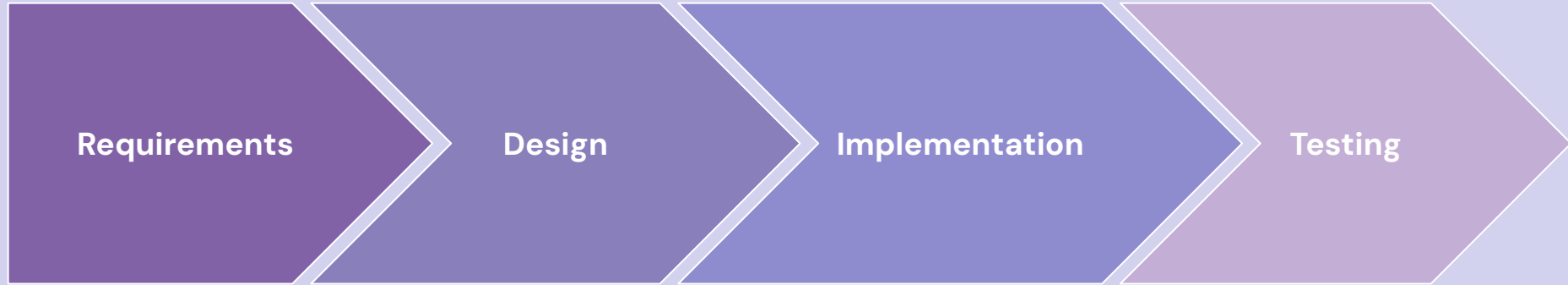


04

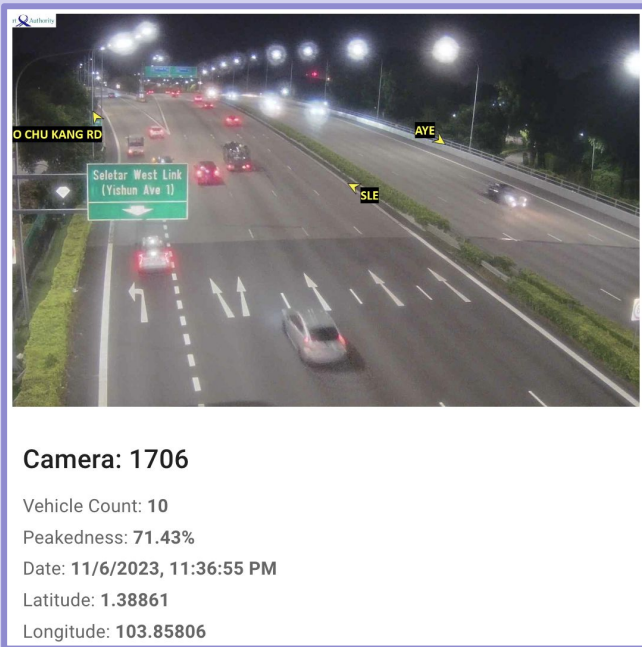
Traceability



Product Development



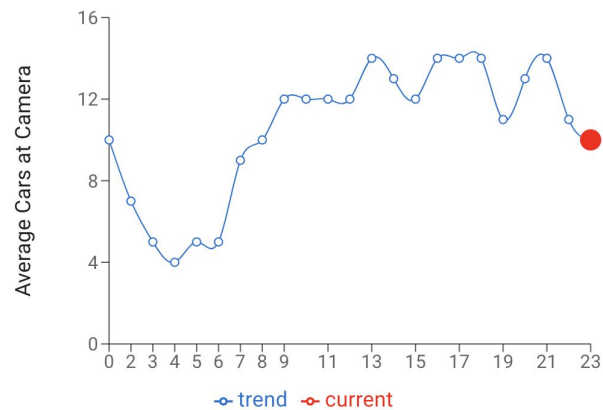
Traffic Trends



Camera View

Road Conditions at Camera Today

Live Traffic Conditions: **Moderate**



Trends

Functional Requirements



API

View all real-time images retrieved from LTA's Traffic Images API



Cameras

Search and view images from a user-specified camera



Trends

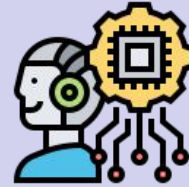
View hourly and daily traffic trends all camera locations

Non-Functional Requirements



Performance

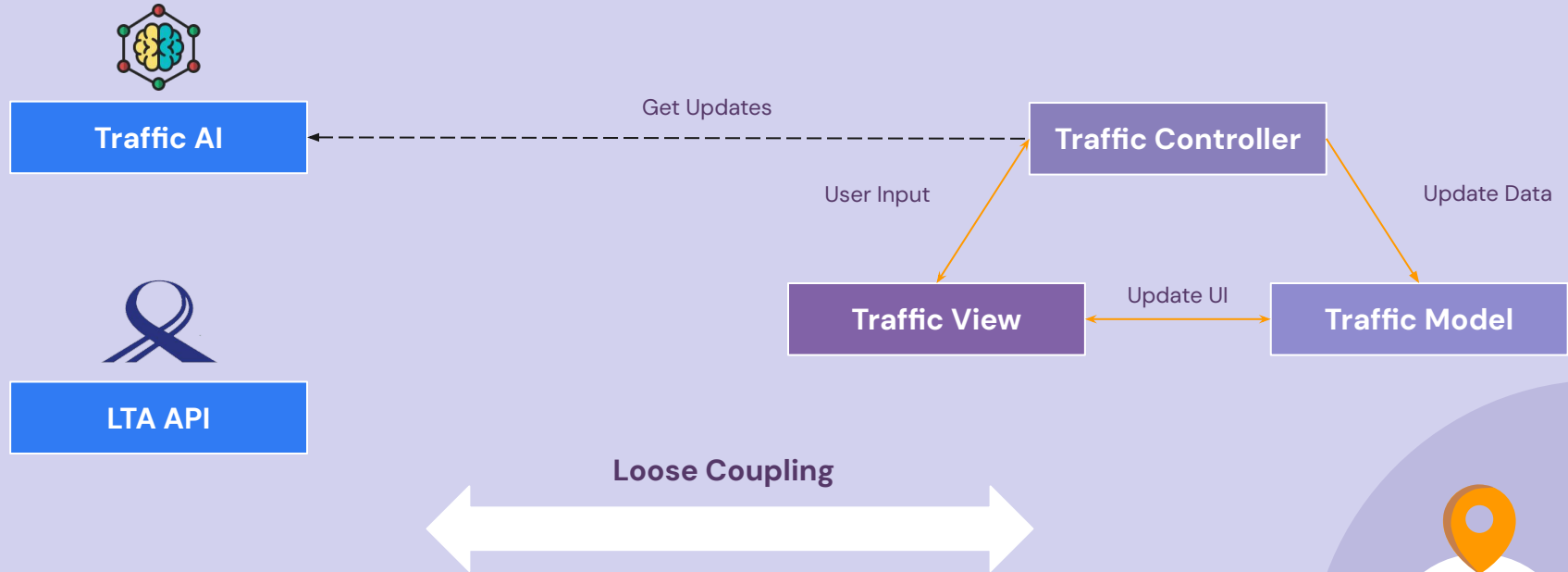
System must load traffic trends in < 5 seconds



Accuracy

System must detect vehicles in images with an accuracy of $> 70\%$

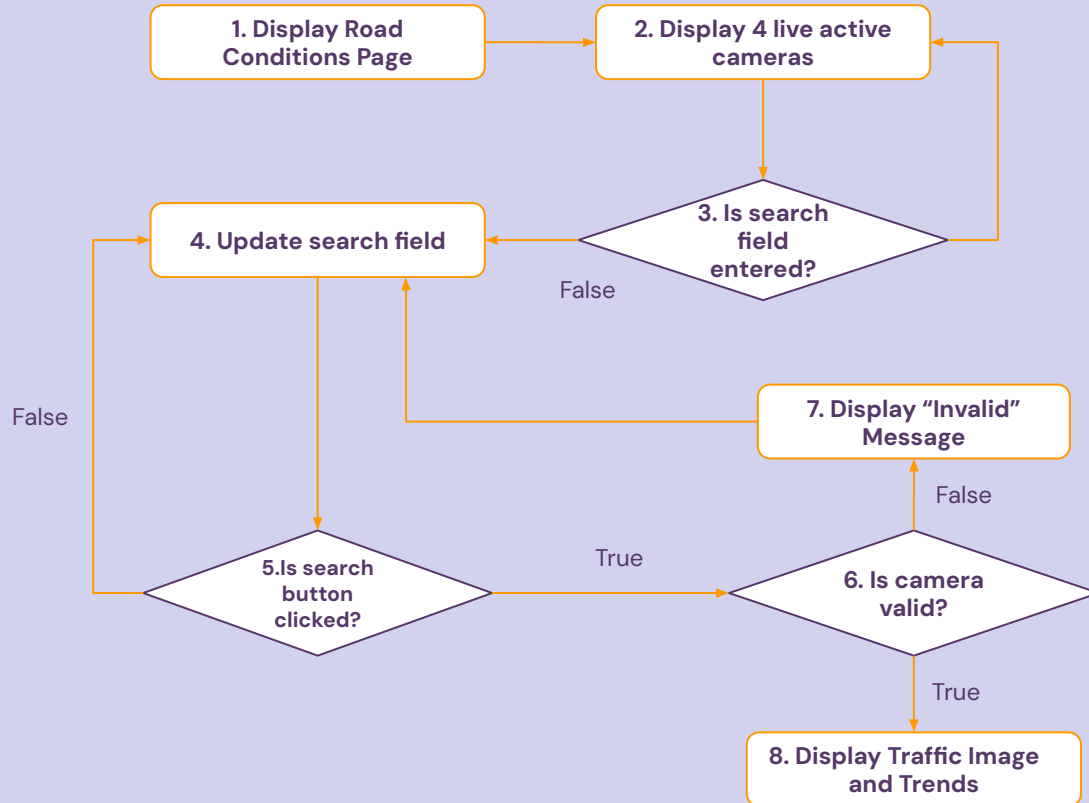
Design and Implementation

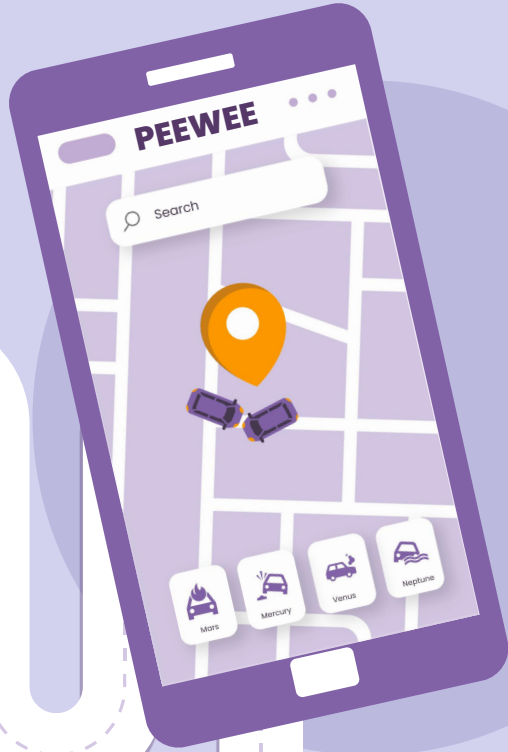


Design Principles



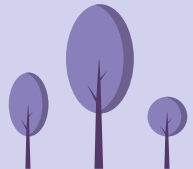
Testing





05

Conclusion



Improvements



Feedback

Allow users to provide feedback on our design.

Live Alerts

Show user live notifications for their frequent routes.

Collaboration

Work with the industry to bring more data to our application.



THANK YOU

