**App Aims:**

User can:

* -Store car information for personal use. Where possible this data is sourced from the GOV api.
  + Vehicle make, model, number plate.
  + Keep maintenance records including costs. These costs should be displayed in useful format such as graph and viewable by time period.
  + Keep track of upcoming MOTS, vehicle tax etc with potential for alerts in the future.

Platform can:

**Screens**

*Login*

*Dashboard*

Subheader-vehicle1 (main) – Vehicle 2 – Vehicle 3 – add vehicle (opens modal)

Car Details – edit (pullable data sourced via DVLA API onload)

Car Image - uploadable

Subsections – Maintenance- spend for year + see more maintenance

-MOT, Tax & Admin – shows expiry date for MOT and Tax

-Service Schedule – next service in x weeks/days (large text) – see more

*Maintenance*

Graph charting spend for year. Stretch to 3yr / all time. Maintenance events show on graph with icon (e.g. tyre for tyre spend, spanner for service etc)

Collapsible for all maintenance events in table format. Editable rows/ values.

Add Maintenance Event – opens modal. Possibility to add multiple image files of receipts/ works + add icon

*Service Schedule*

Next Service. Collapsable here for service history in table format.

Set recurrent target service (opens modal.)

**Technologies**

Vue.js, Node (express), PostgreSQL database, Axios for HTTP requests, GraphJs for data display, vuetify for constructs such as collapisbles, jest + playwright for e2e testing, jest + vue-test-utils for component testing, lotty animations.

**Other Requirements**

Mobile First Design

Revealable Sidebar.

**Unknowns**

Best practice for login/ security.

API options for DVLA

Integration with any other services

Hosting

**Progress Tracking**

Trello (Bantam)

Toggl (time)

**Initial Process**

1. Written specification (this document)
2. Screen Wireframes
3. Screen Designs incl. potential logo design/animation selection (Photoshop/Canva)
4. Scaffold Project
5. Build Page by Page starting with Dashboard.