

Car Mileage/Maintenance Log System

Software Architecture



Emil Bsaibes
Aniruddh Bhatt
Andrew Cary
Yinzhe Lu

Problem Domain

Each day, millions of **automobile owners** drive on roads and highways in vehicles that are in existing need of service. This daily recurring situation puts auto owners and passengers at risk for vehicle breakdown, traffic accidents, costly repairs, and other safety issues.

Most automobile owners attend to vehicle maintenance only when fueling or when the dashboard service light turns on -- indicating a malfunction has already occurred. Furthermore, drivers usually wait several days before taking it to the service shop due to scheduling and coordination issues.

The System

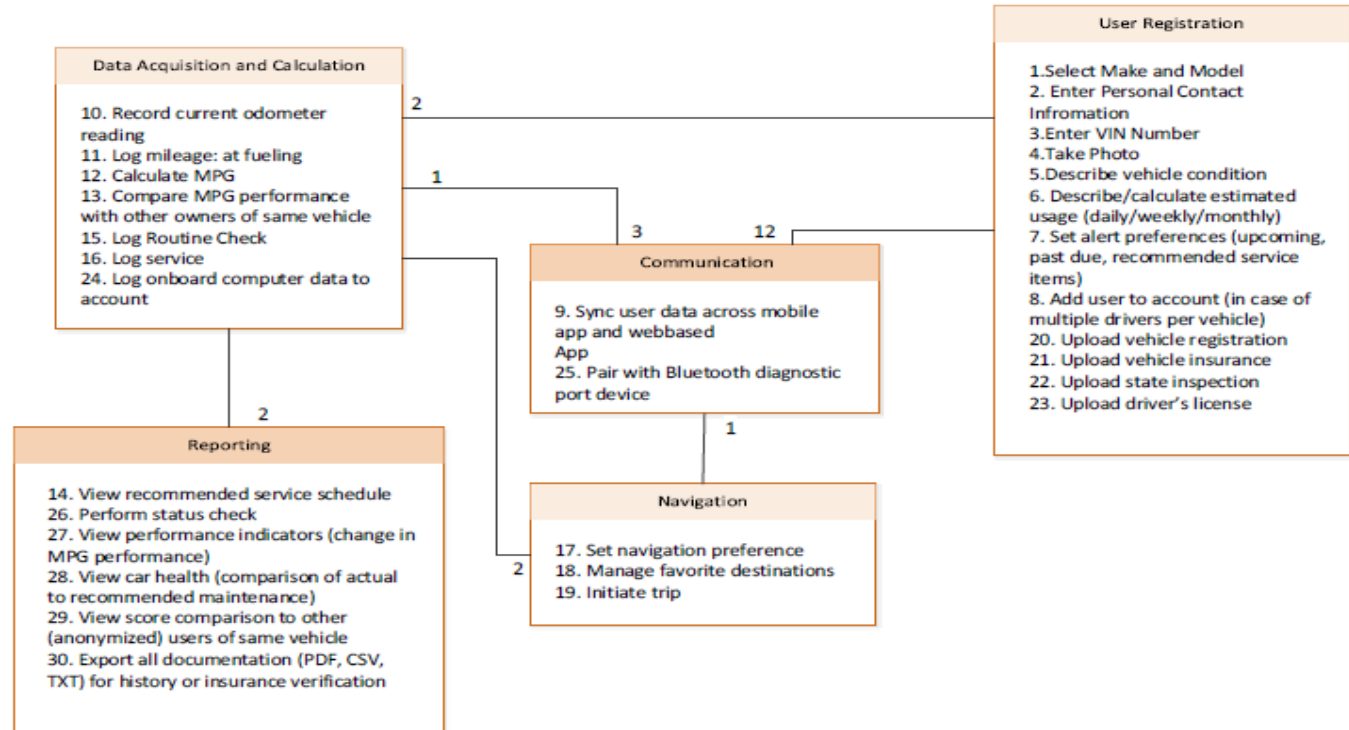
The Car Mileage/Maintenance Log System helps drivers take better care of their vehicles with tools and incentives to **monitor vehicle usage** and **maintain optimal operating condition** through regular service.

Drivers can:

- Track fuel usage (manually and via OBD-II port)
- View manufacturer's recommended service schedule
- Log maintenance checks and services
- Set and receive alerts for service items
- View performance indicators
- Compare car "health" with other drivers

Business Blueprint

Graphic:



Complexity Measure

Component Complexity				
Components	Functions	Data Elements	Input/output Across all Functions	Complexity
User Registration	12	12	14	38
Data Handling	7	6	10	23
Communication	2	1	17	20
Navigation	3	2	3	8
Reporting	6	3	2	11

BB: Evaluation

Motivation: Software systems tend to grow in size and complexity. Meeting the changing stakeholder requirements and having traceability is of highest priority.

Rationale: Based on stakeholder priorities and blueprint complexity “Grouping based on Similar Capabilities” and Grouping by Architectural Style” were adopted. The complexity may have increased for an individual component however the cohesion increased as well.

Quality: The functional and data requirements are easily traceable to the Business component elements. The stakeholder requirements wrt timing and interface are addressed with the “Grouping based on Similar Capabilities” and Grouping by Architectural Style” approach.

Solution Blueprint

Graphic:

Set of Functions Each Module (Solution) is Capable of Providing	
Navigation Module	17,18,19
Log Module	10,11,12,15,16,24,14,13
Sync Module	24,25
Network Module	9
Registration Module	1,2,3,4,5,6,7,8,20,21,22,23
Overall Status Module	27,28,29,30,26,14,13

SB: Evaluation

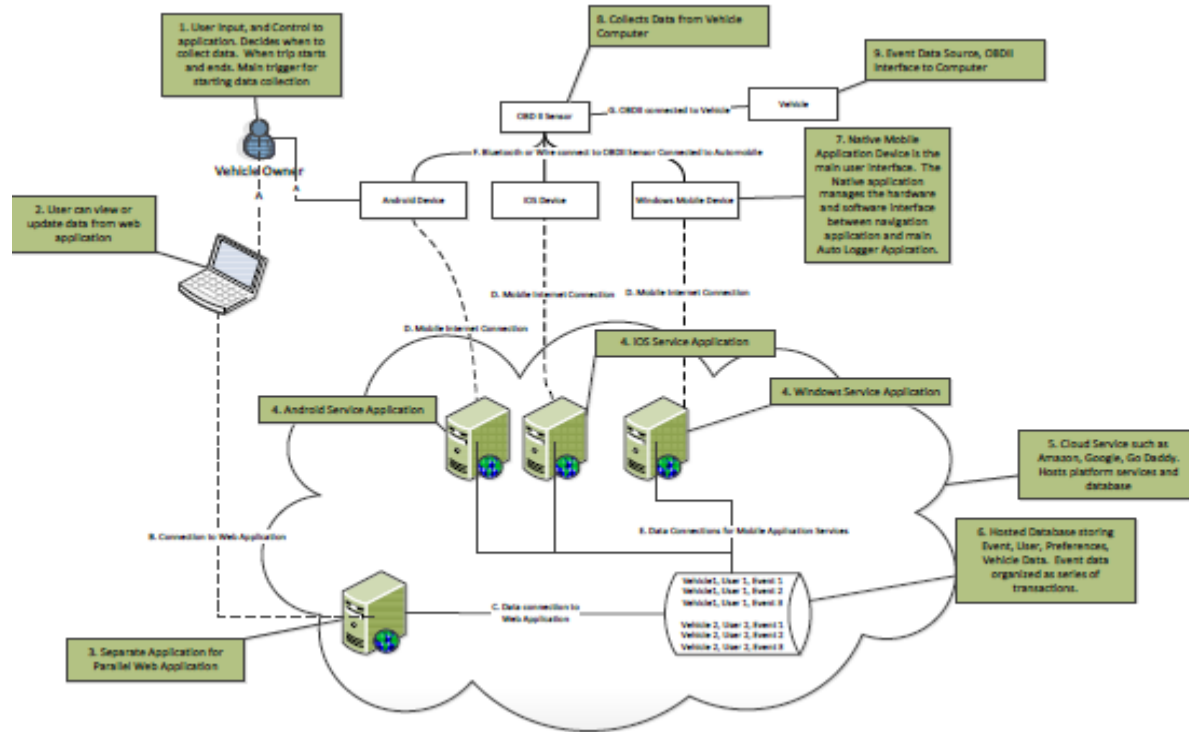
Motivation: Requirements evolve at different rates, thus architectures evolve at different rates. To scope the range of functionality and data requirements required by the customer.

Rationale: Installation constraints change more often than Non-functional requirements which change more often than business requirements. To see how well the functions and features fit in the Business Blueprint structure.

Quality: “Grouping based on similar capabilities” approach ensures that the solution blueprint configurations would map seamlessly onto the Business Blueprint. Thus proving high degree of functional and I/O compliance in terms of scoping the business requirements. The solution components are tend to form Client-Server based pattern in which technologies could provide maximum requirement compliance.

Deployment Blueprint

Graphic:



DB: Evaluation

Motivation: Instantiate Business Blueprint by describing implementation details to actualize the stakeholder requirements.

Rationale: Implementing site-specific deployments with installation constraints. The Client-Server architecture has maximum compliance to the the prioritized functional, I/O and non-functional requirements of the stakeholder.

Quality: The compliance metric provides boundary error details.

CompFuncBoundary Error			
Solutions	Degree to which t does not satisfy the best fit component	Degree to which t exceeds the best fit component	Total Error for (t)
Navigation Module	$1-3/3=0$	0	0
Log Module	$1-7/7=0$	1/6	0.167
Sync Module	$1-1/2=0.5$	1/7	$\frac{1}{2}+1/7=0.643$
Network Module	$1-1/2=0.5$	0	0.5
Registration Module	$1-12/12=0$	0	0
Overall Status Module	$1-6/6=0$	1/7	0.143

Thanks!

