Deliverable #2

SE 3A04: Software Design II – Large System Design

Tutorial Number: T03
Group Number: G8
Group Members:

- Hashim Bukhtiar
- Jaden Moore
- James Ariache
- Olivia Reich
- Omar Abdelhamid

IMPORTANT NOTES

- Please document any non-standard notations that you may have used
  - *Rule of Thumb*: if you feel there is any doubt surrounding the meaning of your notations, document them
- Some diagrams may be difficult to fit into one page
  - Ensure that the text is readable when printed, or when viewed at 100% on a regular laptop-sized screen.
  - If you need to break a diagram onto multiple pages, please adopt a system of doing so and thoroughly explain how it can be reconnected from one page to the next; if you are unsure about this, please ask about it
- Please submit the latest version of Deliverable 1 with Deliverable 2
  - Indicate any changes you made.
- If you do NOT have a Division of Labour sheet, your deliverable will NOT be marked

1 Introduction

1.1 Purpose

This document provides a high-level overview of the RideRecon car identification system architecture, including high-level design considerations of the system and design consideration for various subsystems. This document is intended for internal RideRecon stakeholders, including but not limited to, project managers, developers, domain experts, and RideRecon team members/investors.

Note that RideRecon Deliverable 1 should be read before Deliverable 2, and technical knowledge may be beneficial in better understanding this document's contents.

1.2 System Description

The RideRecon system is designed to identify vehicles based on both text and image inputs, leveraging multiple expert modules (such as a reverse image search engine, a text-based LLM, and trained ML models). To integrate these diverse experts effectively, RideRecon follows a blackboard architecture. In this approach, all relevant data—user inputs, partial identifications, and expert findings—are posted to a central "blackboard." Each expert module reads from this shared repository, processes the data according to its specialization, and writes back its results. This iterative process continues until a consensus or a conflict resolution mechanism determines the final identification outcome.

A blackboard architecture is well-suited to RideRecon because it supports concurrent data processing among the different expert modules, each of which may require varying amounts of time or resources. It also simplifies the system's ability to integrate new expert modules in the future—such as additional AI models or external APIs—by giving them the same shared data repository to read from and write to. This design ensures that updates or new insights posted by one expert can immediately inform the decision-making of others, resulting in a flexible, extensible platform for complex vehicle identification tasks.

1.3 Overview

Describe what the rest of the document contains and explain how the document is organised (e.g. "In Section 2 we discuss...in Section 3...").

2 Analysis Class Diagram

This section should provide an analysis class diagram for your application.

3 Architectural Design

This section should provide an overview of the overall architectural design of your application. Your overall architecture should show the division of the system into subsystems with high cohesion and low coupling.

3.1 System Architecture

- Identify and explain the overall architecture of your system
- Be sure to clearly state the name of the architecture you used (this is the name of the architectural pattern, not the name of your system)
- Provide the reasoning and justification of the choice of architecture
- Provide a structural architecture diagram showing the relationship among the subsystems (if appropriate)
- List any design alternatives you considered, but eliminated (and explain why you eliminated them)

3.2 Subsystems

The RideRecon system is built on a hybrid architecture that primarily leverages a blackboard approach to integrate multiple expert modules while incorporating repository principles for robust data management. At its core, the system uses a blackboard architecture where diverse expert modules—such as the Vertex AI Image Model, Reverse Image Search, a specialized text-based LLM, and an internally trained ML model—collaborate by posting their insights to a shared central platform. This collaborative process enables iterative refinement and conflict resolution, ensuring that the final vehicle identification is both accurate and reliable.

Complementing this dynamic processing core, the Input Management Subsystem captures and preprocesses user inputs, ensuring that both text and image data are formatted consistently before being submitted to the central blackboard. Meanwhile, the Data Management Subsystem employs a repository architecture to securely store both the user inputs and the outcomes of the identification process. This persistent storage supports future retraining, auditing, and continuous improvement of the models. Additionally, the Account Management Subsystem manages user authentication and profile data, enabling personalized interactions and maintaining a history of user engagements. Together, these interconnected subsystems form a robust and scalable platform tailored to address the complex challenges of vehicle identification in dynamic, real-world scenarios.

4 Class Responsibility Collaboration (CRC) Cards

This section should contain all of your CRC cards.

- Provide a CRC Card for each identified class
- Please use the format outlined in tutorial, i.e.,

Class Name:	
Responsibility:	**Collaborators:**

A Division of Labour

Include a Division of Labour sheet which indicates the contributions of each team member. This sheet must be signed by all team members.

Hashim Bukhtiar	Jaden Moore	James Ariache	Olivia Reich	Omar Abdelhamid
1.1, 1.2, 3.2 Compiled Final Doc	X Cards in Section 4	Section 2 James Ariache	Section 3.1	Y Cards in Section 4 Section 1.3 Omar Hassan

Table 1: Division of Labour