Software Architecture Document

TRIP SPLITTER/MERGER AND ENRICHER

CROSSYN

Date	**:**	**12/01/2022**
Version	**:**	**1.0**
State	**:**	**Released**
Author	**:**	**Dobri Trifonov, Joran van de Moosdijk, Robert Enuta, Stanislav Petkov, Tobias Halomoan**

Version history

Version	Date	Author(s)	Changes	State
0.1	09/11/2021	Dobri Trifonov, Joran van de Moosdijk, Robert Enuta, Stanislav Petkov, Tobias Halomoan	Initial draft of architecture document	Draft
1.0	12/01/2022	Dobri Trifonov, Stanislav Petkov	Updating C3 & C4 Models, Database diagram, API documentation and Introduction chapter	Released

Distribution

Version	Date	Receivers
0.1	10/11/2021	Gupta, Roopali R.
1.0	12/01/2022	Gupta, Roopali R., Bram van Herwijnen

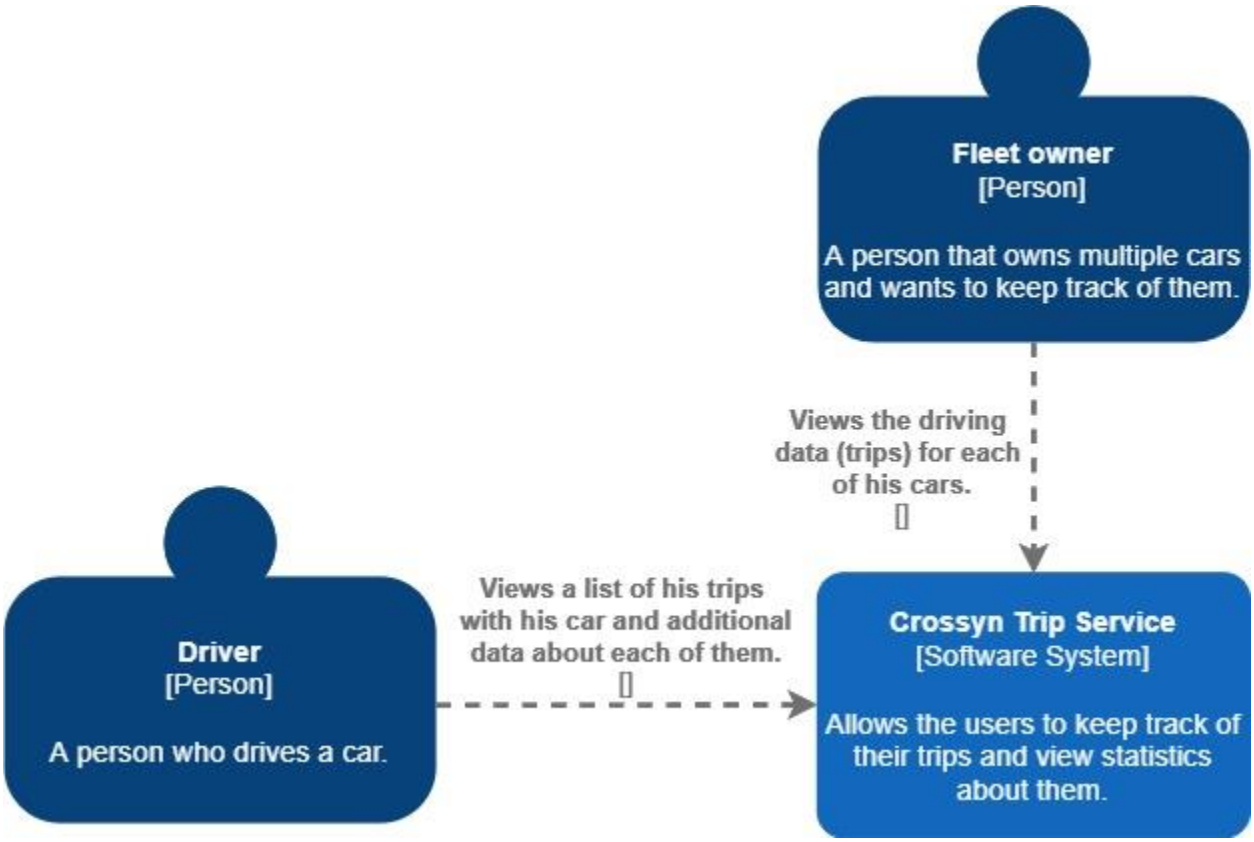
Contents

1. **Introduction** 4
2. **System Context (C1)** 5
3. **Containers and tech choices (C2)** 6
4. **Components (C3)** 7
5. **Class diagrams and sequence diagrams** 8
6. **Database Design** 9
7. **Interfaces or API documentation** 10

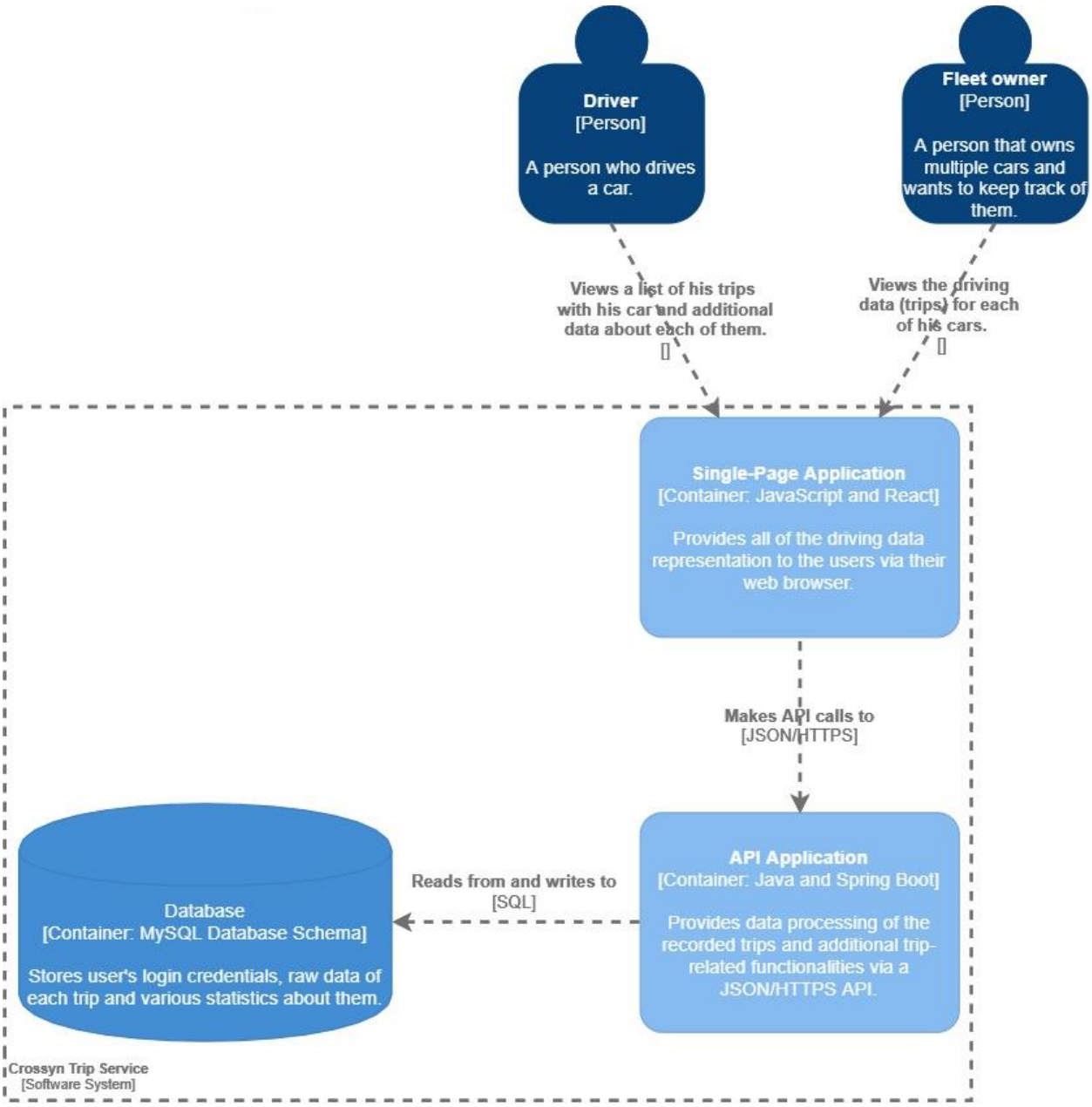
1. Introduction

The purpose of this document is to showcase the architecture of the software system that our team has developed for the company “Crossyn”, to guarantee a SOLID architectural design and provide an overview on framework choices and how the system is going to operate.
This application is going to have two types of users: drivers and fleet owners, where the drivers will be able to keep track of the trips they have done with their vehicle and the fleet owners will be able to manage and view various enriched data about their vehicles generated by this software system.

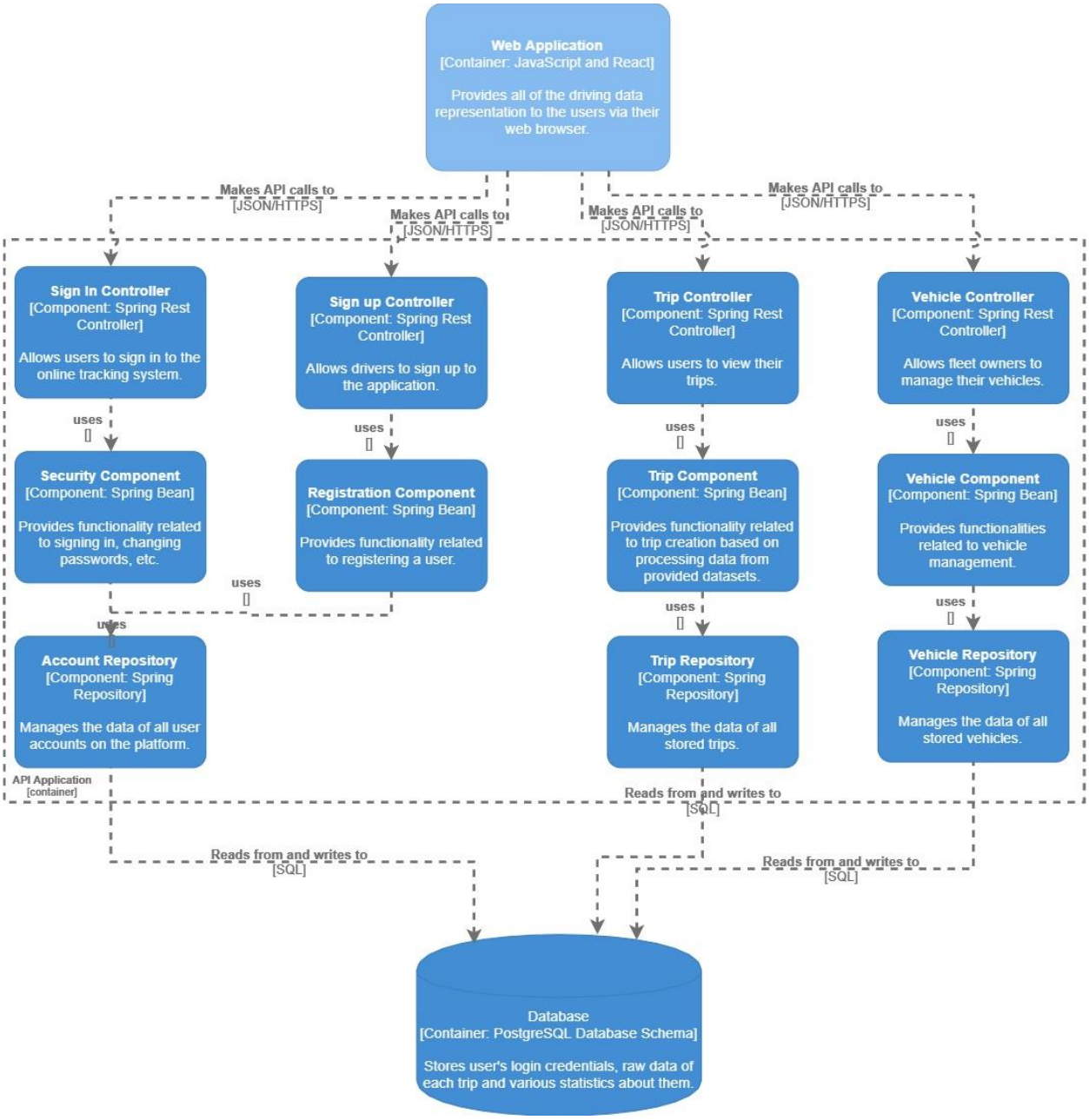
2. System Context (C1)



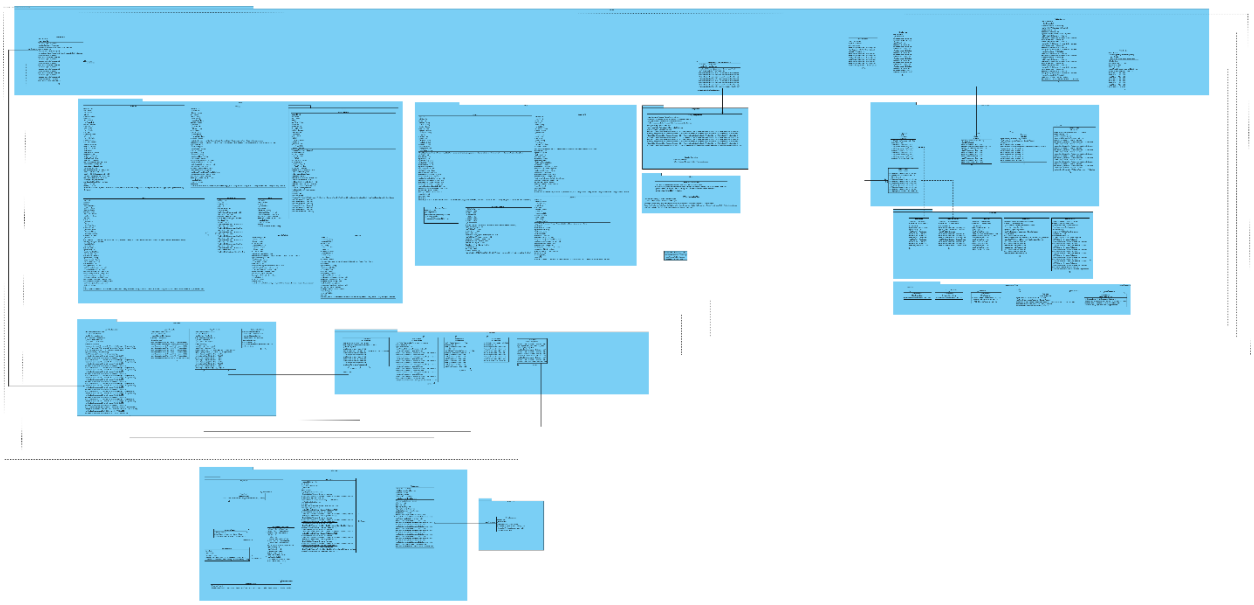
3. Containers and tech choices (C2)



4. Components (C3)

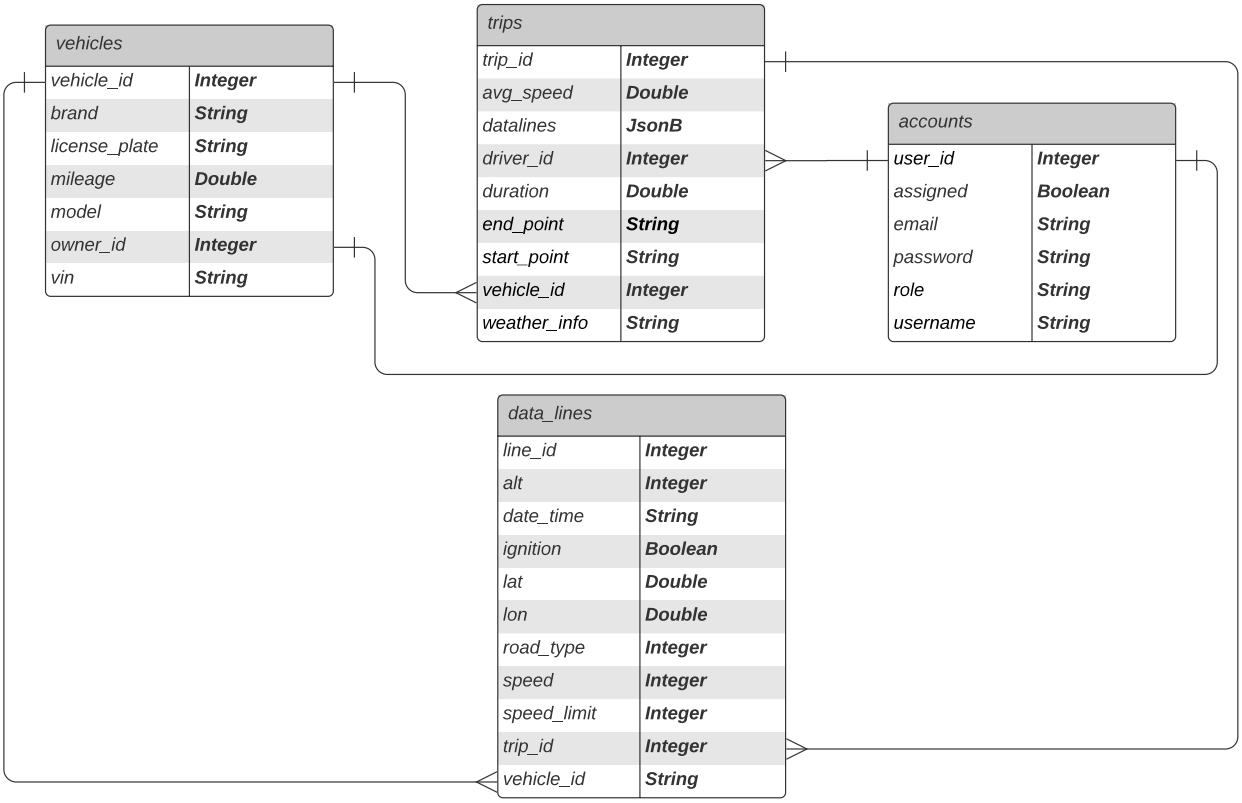


5. Class diagram



** Raw file included in the documentation directory (For better quality)*

6. Database Design



7. Interfaces or API documentation

URL	resource	operation	result
/trips	Trips	GET	read a list with all trips
/trips/driver	Trips	GET	read a list with all trips of the current user
/trips/1	Trips	GET	read a list with all trips for vehicle with id 1
/trips	Trips	POST	create a list of trips based on provided dataset
/users/register	Accounts	POST	create a new user account
/login	Accounts	POST	provides access token based on correct user credentials
/vehicles	Vehicles	GET	read a list with all available vehicles
/vehicles/2	Vehicles	GET	read a vehicle with id 2
/vehicles/owner	Vehicles	GET	read a list with all vehicles owned by the current user
/vehicles/1/drivers	Accounts	GET	read a list with all current drivers of vehicle with id 1
/vehicles/assignDriver?licensePlate=212-32-54	Vehicles	POST	Assign the current user to vehicle with license plate number: 212-32-54
/vehicles	Vehicles	POST	create a new vehicle from provided form information