



1

2

GENIVI Alliance

3GENIVI Document CS00066

4POIService

5Component Specification

6Accepted Version 2.0.0

725-01-2017

8Sponsored by:
9GENIVI Alliance

10true

11Accepted

12**Abstract:**

13This document provides the Component Specification for the POIService

14**Keywords:**

15POI, Search, Navigation

2

1This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International
2License. Copyright © 2017, trueBMW Car IT GmbH, Continental Automotive GmbH, PCA Peugeot Citroën, XS
3Embedded GmbH, TomTom International B.V., Alpine Electronics R&D Europe GmbH, AISIN AW CO.LTD.

4

5All rights reserved.

6The information within this document is the property of the true and its use and disclosure are restricted. Elements of
7GENIVI Alliance specifications may be subject to third party intellectual property rights, including without limitation, pat-
8ent, copyright or trademark rights (and such third parties may or may not be members of GENIVI Alliance). GENIVI Alli-
9ance true not responsible and shall not be held responsible in any manner for identifying, failing to identify, or for securing
10proper access to or use of, any or all such third party intellectual property rights.true

11“”“”

12

13

14

15

16

17GENIVI and the GENIVI Logo are trademarks of GENIVI Alliance in the U.S. and/or other countries. Other company,
18brand and product names referred to in this document may be trademarks that are claimed as the property of their respective
19owners.

20trueThis work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

21<http://creativecommons.org/licenses/by-sa/4.0>

22

23

24The above notice and this paragraph must be included on all copies of this document that are made.

25

26

27

28

GENIVI Alliance
2400 Camino Ramon, Suite 375
San Ramon, CA 94583, USA

1. Revision History

The following table shows the revision history for this document.

Document Revision History

| Version | Date | Author | Change |
|---------|-------------|-------------------------------------|--|
| 0.1 | 24 Oct 2012 | Marco Residori (XS Embedded) | Created empty template |
| 0.2 | 19 Dec 2012 | Marco Residori (XS Embedded) | Added architecture diagram. Added sequence Diagrams. Updated APIs, Added Cedric Ranson's use case descriptions. |
| 1.0 | 19 Dec 2012 | Marco Residori (XS Embedded) | Added architecture diagram. Corrected typos. |
| 1.1 | 3 Feb 2017 | Philippe COLLIOT (PSA GROUPE) | Refine the document for Franca based API |

2

1

Table of contents

2

| | | |
|-----|--|----|
| 31 | Introduction..... | 5 |
| 4 | 1.1 System Overview..... | 5 |
| 5 | 1.2 Component Overview..... | 5 |
| 62 | Terminology..... | 6 |
| 73 | Architecture..... | 7 |
| 8 | 3.1 Interfaces..... | 7 |
| 9 | 3.2 Interaction with other Components..... | 8 |
| 104 | API..... | 9 |
| 11 | 4.1 Git Repository..... | 9 |
| 12 | 4.2 Use Cases..... | 9 |
| | 4.2.1 POI Search..... | 9 |
| | 4.2.1.1 The HMI displays the list of POI categories..... | 9 |
| | 4.2.1.2 The user searches for a restaurants around his/her position..... | 11 |
| | 4.2.1.3 The user is running out of fuel and therefore creates a proximity alert around his/her position to receive a sound alert in case a new fuel station is detected..... | 13 |
| | 4.2.2 POI Content Access..... | 14 |
| | 4.2.2.1 The user wants to get a list of car parks with free places around the current position sorted by price..... | 14 |
| | 4.2.2.2 The user wants to display on the map the location of all his/her Facebook friends around the vehicle position (the profile picture is used to mark a friend's position). . | 16 |
| | 4.2.2.3 The user wants to navigate to his/her doctor (the one listed in his/her phone contact book) if the office is open..... | 17 |
| | 4.2.2.4 The user wants to watch a movie. The user requests a list of movies projected in nearby cinemas with description, cinema name and price, all sorted by start time..... | 18 |
| | 4.2.2.5 The user creates a proximity alert along its route to receive a sound in case of dangerous area..... | 19 |
| 13 | 4.3 Interfaces..... | 21 |
| 14 | | |

11 Introduction

2This document describes the API of the POIService component.

31.1 System Overview

4The GENIVI Software Platform is a platform consisting of standardized middleware, application layer interfaces and frame-
5works defined or adopted by the GENIVI Alliance.

61.2 Component Overview

7The POIService is a software component of the above mentioned GENIVI Software Platform that
8offers POI search to client applications.

12 Terminology

| <i>Term</i> | <i>Description</i> |
|-------------|-----------------------|
| POI | Point of Interest |
| CAM | Content Access Module |

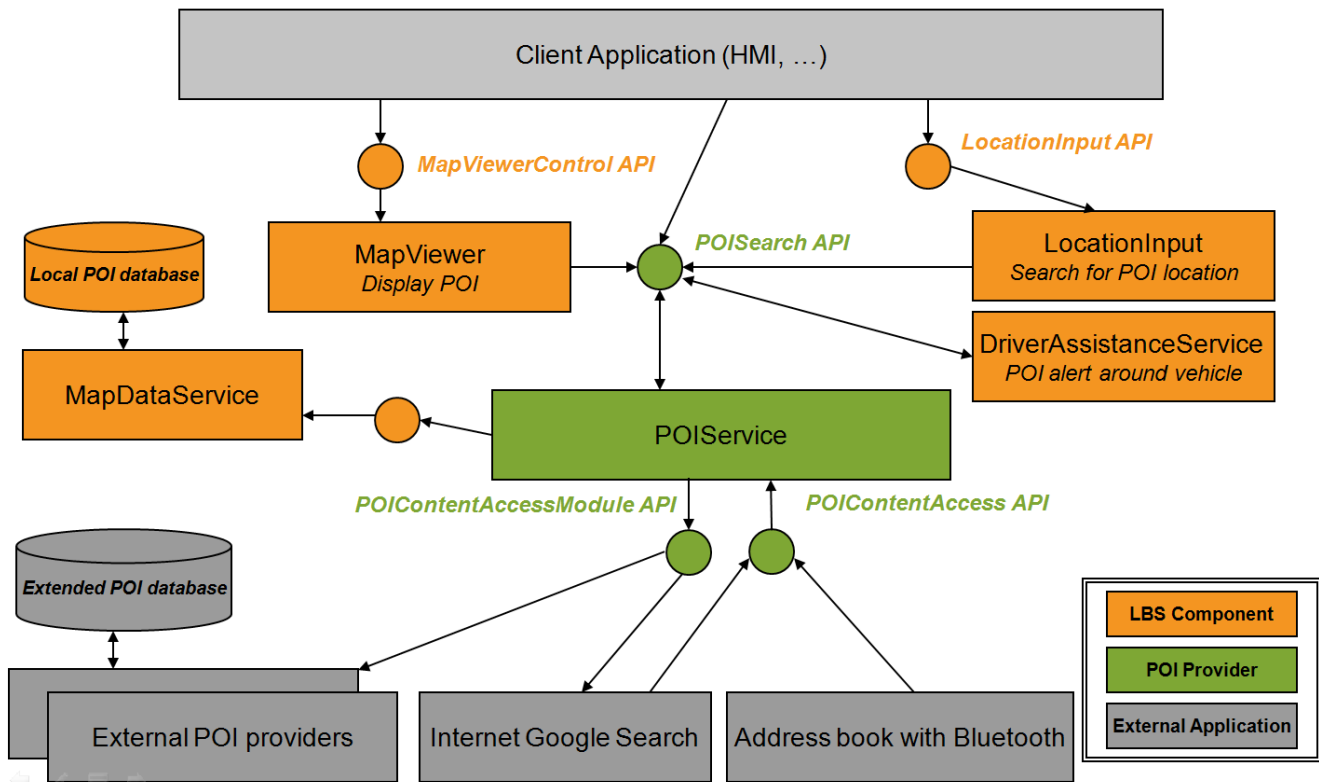
13 Architecture

23.1 Interfaces

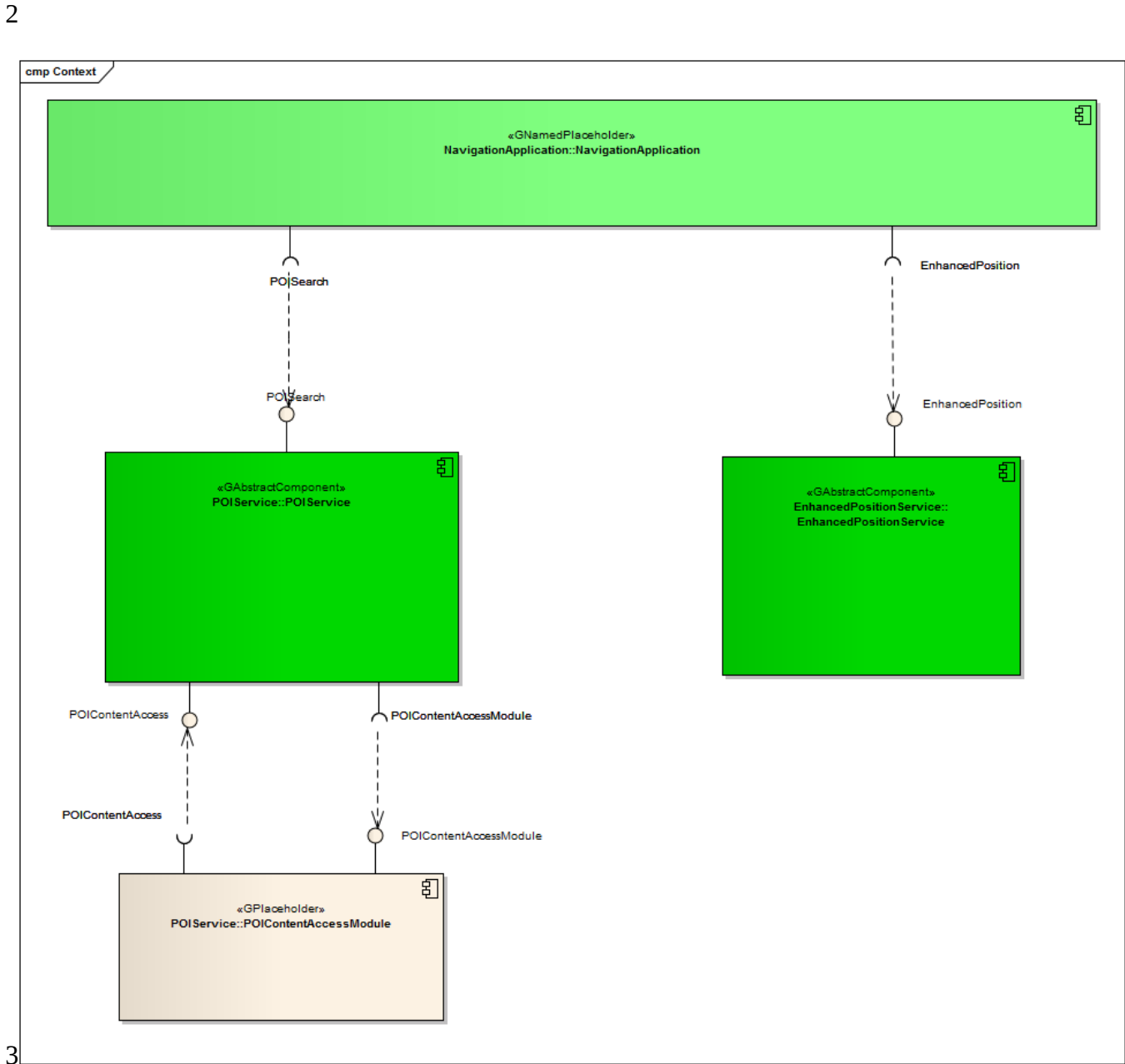
- POISearch

- POIContentAccess

- POIContentAccessModule



2
13.2 Interaction with other Components



14 API

24.1 *Git Repository*

3The POIService interfaces can be found in the GENIVI Git repository at:

4<https://github.com/GENIVI/navigation/tree/master/api/franca/navigation/poiservice>

54.2 *Use Cases*

64.2.1 POI Search

7

84.2.1.1 **The HMI displays the list of POI categories**

10**Description**

12The HMI retrieves the root category from the component (should be ALL_CATEGORIES).

13The HMI retrieves the children categories of the root category.

14For each child category, the HMI retrieves its children:

15If there are no children, the HMI retrieves the category details from the component and displays its name as a searchable category.

16It can also display its visual icon.

17For each child found, the HMI retrieves the children again and then displays the category name if there are no children.

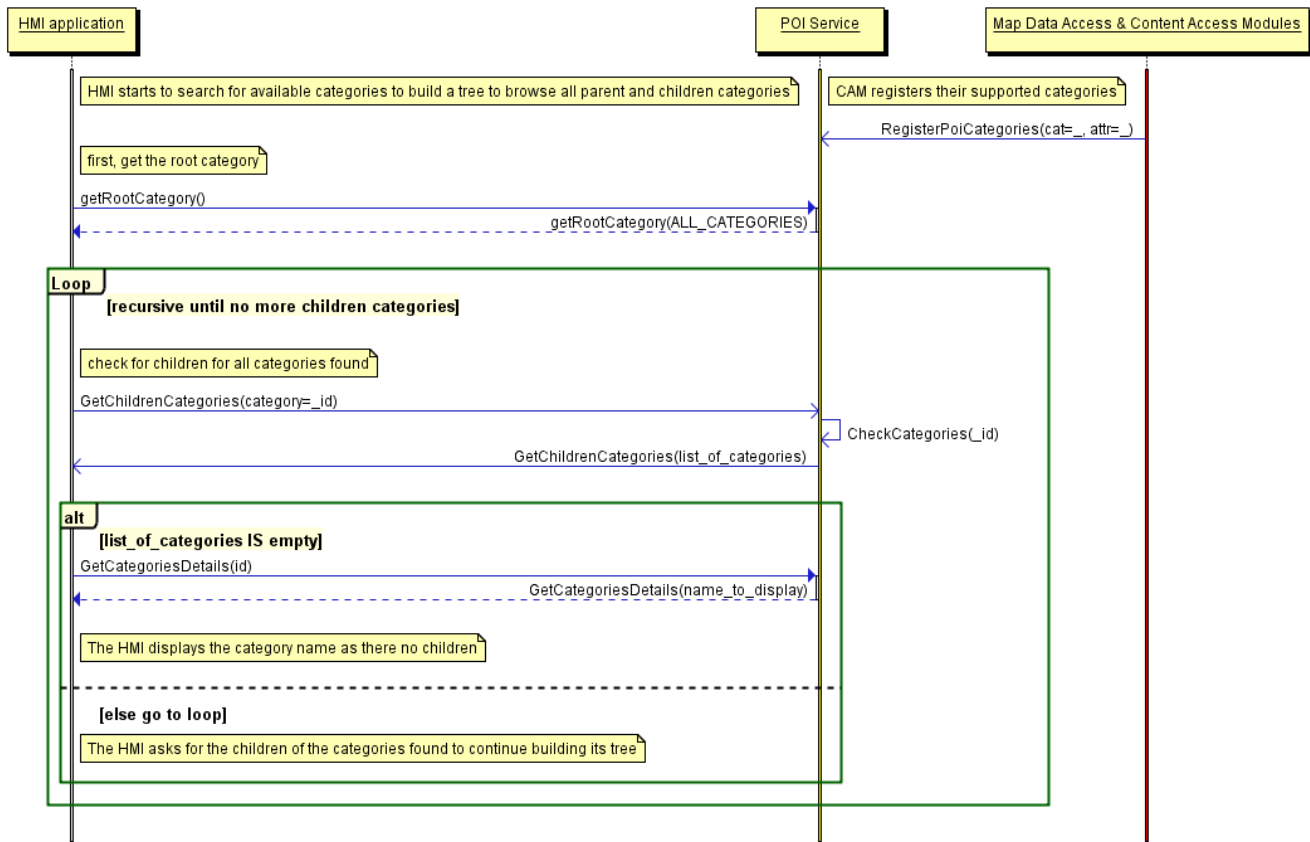
20**Error Management**

21Root category is not available.

22

24**Sequence diagram**

25



2

1

2.4.2.1.2 **The user searches for a restaurants around his/her position**

3

4**Description**

5

6The HMI retrieves the list of POI categories from the **search POI** API.

7The HMI retrieves the vehicle position from the **enhanced position** API.

8The HMI selects restaurant as a category and searches for POI around the vehicle position. It does not provide a search radius.

9The POIService component defines a default radius for the category *restaurant*.

10The POIService component searches for restaurant around the vehicle position within the local database.

11The POIService component provides the list of restaurants sorted by distance. It also provides basic details.

12The HMI browses the list and ask for detailed information for one of the POI.

13The POIService component gets detailed information from the local database.

14The HMI decides to drive to this restaurant.

15The POIService component provides the location of the restaurant.

16

17**Error Management**

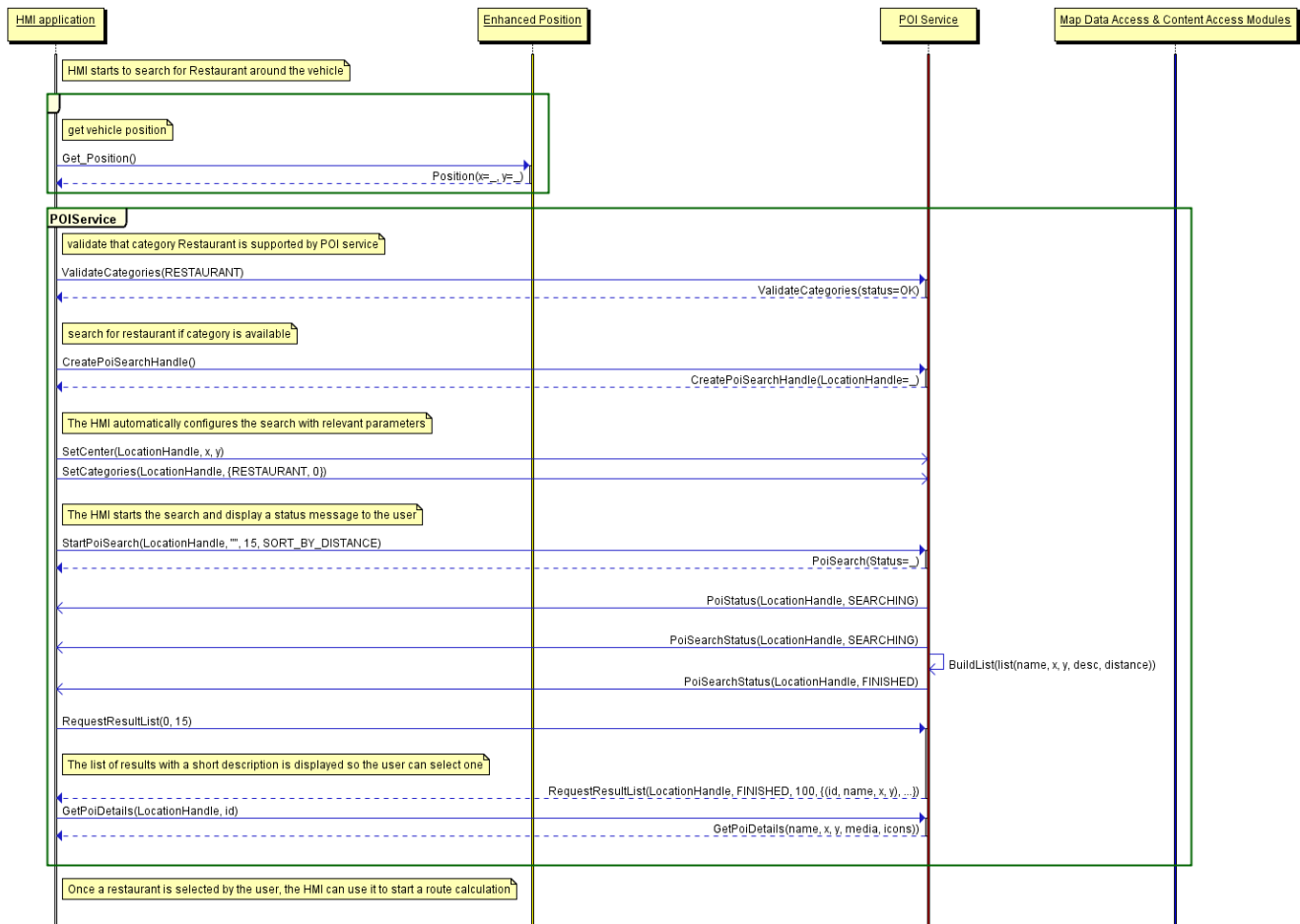
18If the local database access is down, the component returns an error.

20

21**Sequence diagram**

22

23



1.4.2.1.3 **The user is running out of fuel and therefore creates a proximity alert around his/her position to receive a sound alert in case a new fuel station is detected**

5 **Preconditions**

6 The fuel station is a category that is supported by the local database.

8 **Description**

9 The HMI configures a new POI proximity alert for fuel station around the vehicle.

10 The HMI retrieves the vehicle position and the vehicle speed from the **enhanced position** API.

11 The POIService component defines the search radius of 500 meters.

12 The POIService component gets the list of fuel stations around the current position from the local database using **map data access** API.

13 A new list of POI is sent to the HMI with all the fuel station that match the alert configuration.

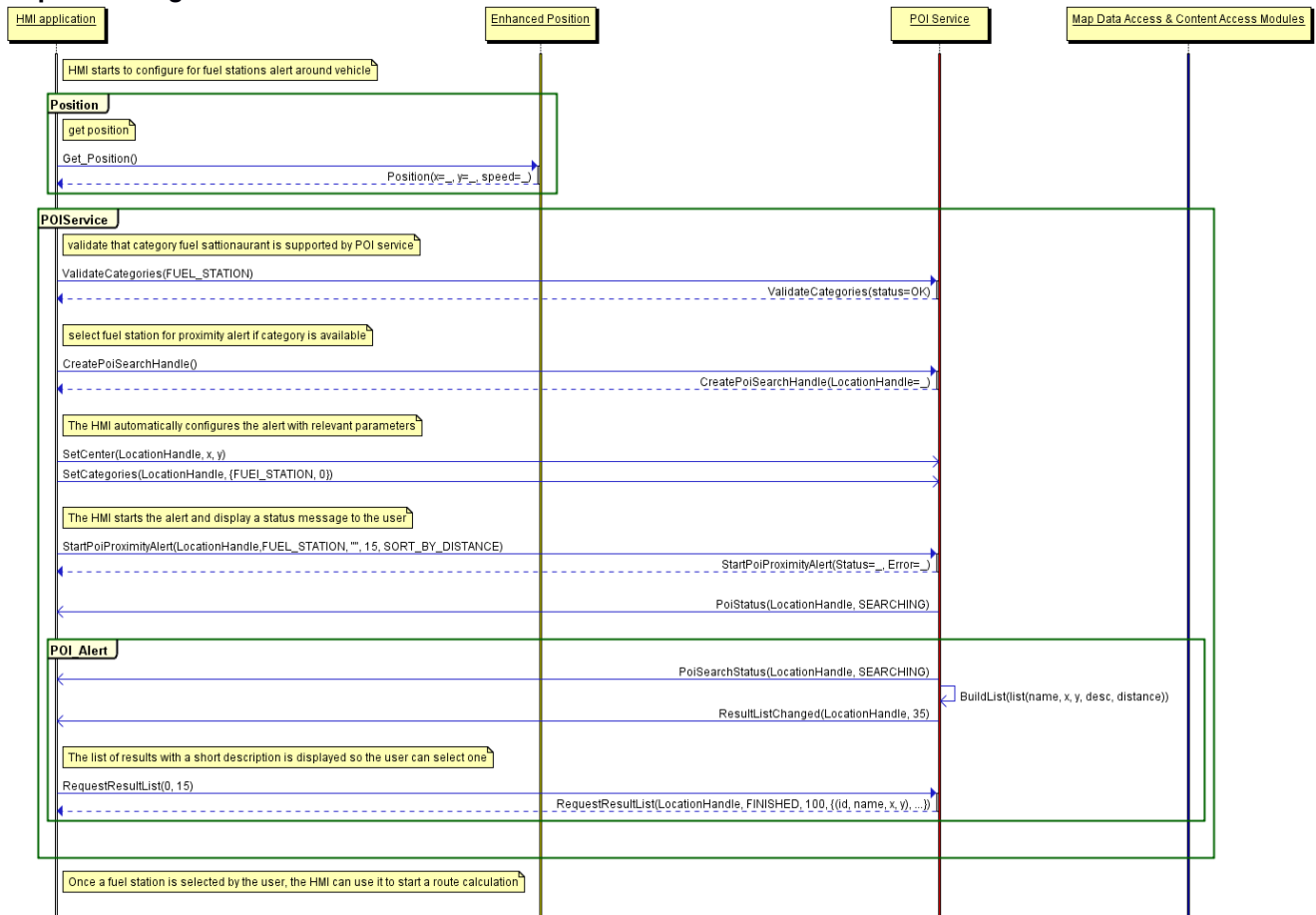
14 The HMI plays a sound when a new (non-empty) list of POI is received.

15 As soon as new fuel stations are available or current fuel stations are out the search radius, a new list is provided.

17 **Error Management**

18 If the local database access is down, the component returns an error.

21 **Sequence diagram**



14.2.2 POI Content Access

2

3_{4.2.2.1} **The user wants to get a list of car parks with free places around the current position sorted by price**

6**Preconditions**

7A plug-in was already installed in the POI Service component to access an identified online service that provides car park information (price and places left) from the name and coordinates of a car park.

10**Description**

11The HMI retrieves the vehicle position from the **enhanced position** API.

12The HMI retrieves the list of car parks with free places sorted by price around the position from the **search POI** API.

13The POIService component get the list of car parks around the current position from the local database using **map data access** API.

14For each car park name and location found, the POI Service uses **POI content access** API to get free places available and the park price from the plug-in (described below).

16The POIService component stores the result (all POI found with dynamic content) into the persistent storage to manage the 'connection-down' error. This content is deleted when it gets too old.

18The POIService component updates each car park attribute with available places and sorts the list by price.

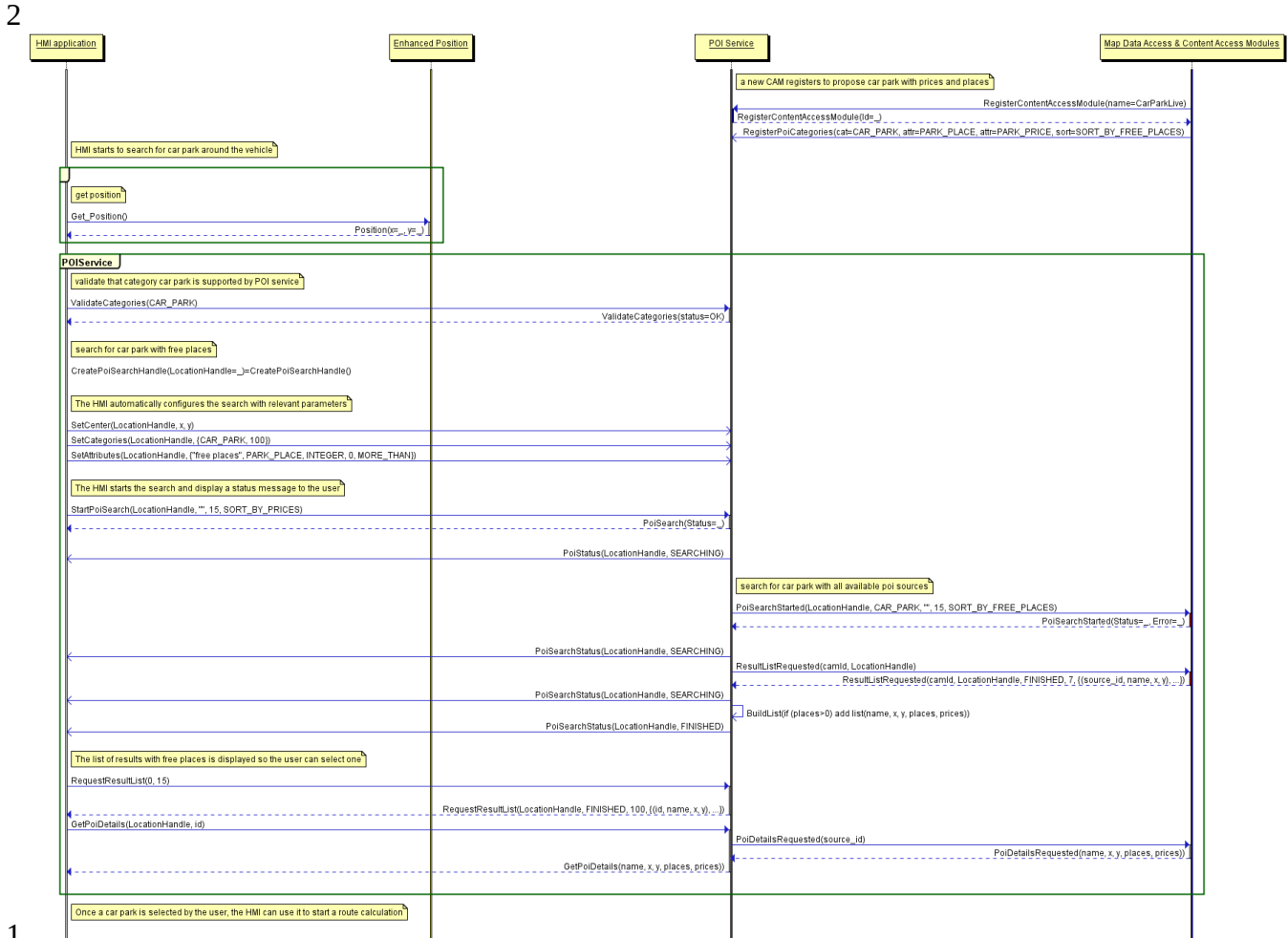
20**Error Management**

21If the local database access is down, the component returns an error.

22If the network connection to the online provider is down, the content access module tries to retrieve latest content from its persistent storage (if any). If this content is too old (depends on CAM policy), it returns an error.

24

25**Sequence diagram**



2

1_{4.2.2.2} **The user wants to display on the map the location of all his/her Facebook**
2 **friends around the vehicle position (the profile picture is used to mark a**
3 **friend's position)**

4

5Preconditions

6A plug-in was already installed in the POI Service component to access Facebook services and extract the profile picture and the current address from a list of friends.

8This plug-in also registered a new POI category called **Facebook friends** and sub-category of predefined **User contact**.

9The MapViewer API provides a way to create a custom location on the map with coordinates, name and icons.

11Description

12The HMI retrieves the vehicle position from the **enhanced position** API.

13The HMI retrieves the list of Facebook friends from the **search POI** API around the vehicle position.

14POI Service components gets the list of Facebook friends (with photo, name and coordinates) around the current position from the Facebook plug-in **POI content access** API.

16The POI Service component stores the result into the persistent storage to manage the 'connection-down' error.

17This content is deleted when it gets too old.

18For each friend, the POI Service uses **MapViewer** API to update the map with icons.

19

20Error Management

21If the network connection to the online provider is down, the content access module tries to retrieve latest content from its persistent storage (if any). If this content is too old (1 hour), it returns an error.

23
24

2

1.4.2.2.3 **The user wants to navigate to his/her doctor (the one listed in his/her phone contact book) if the office is open**

4 **Preconditions**

5 A plug-in was already installed in the POI Service component to access a doctor's web site. The plug-in extracts the opening times and the doctor's office address.

6 A plug-in was already installed in the POI Service component to access the user's smartphone. The plug-in extracts a contact list from a group of names (friend, family, doctor, ...).

7 This plug-in created a new POI category called **My Phone Groups** with sub-categories like **friend** or **doctor**.

11 **Description**

12 The HMI retrieves the doctor's office coordinates and opening times from the user's phone POI group doctor with the **search POI** API.

13 The POIService component gets the doctor's office opening time and coordinates from the smartphone contacts plug-in using the **POI content access** API.

14 The HMI validates that the opening times match current time and start a route calculation.

15 The POIService component gets the doctor's office coordinates from previously stored data.

16 The POIService uses **NavigationCore** API to start a route calculation from vehicle position to doctor's office.

19 **Error Management**

20 If the network connection to the online provider is down, the content access module returns an error.

21

2

1.4.2.2.4 **The user wants to watch a movie. The user requests a list of movies
 2 projected in nearby cinemas with description, cinema name and price, all
 3 sorted by start time**

4Preconditions

5A plug-in was already installed in the POI Service component to access an identified online service that provides cinema information (list of
 6movies, opening times, prices and movies information like review and trailer link) from the name and coordinates of a cinema.

8Description

9The HMI retrieves the vehicle position from the **enhanced position** API.

10The HMI retrieves the list of cinema with movies that starts in less than 1 hour around the current position from the **search POI** API. The list
 11is sorted by start time.

12The POIService component gets the list of cinemas around the current position from the local database using the **map data access** API.

13For each cinema name and location found, the POI Service uses **POI content access** API to get list of movies available with start time,
 14movie information and price from the plug-in (described below).

15The HMI display the list of movies sorted by start time (the sooner the first) with a quick description, a review, the price and the cinema
 16name.

17The rear seat HMI downloads a video from movie trailer link and displays it.

18Depending upon the selected film, the POIServicecomponent gets the cinema coordinates from previously stored data.

19The POIService uses **NavigationCore** API to start a route calculation from vehicle position to the cinema.

21Error Management

22If the local database access is down, the component returns an error

23If the network connection to the online provider is down, the content access module tries to retrieve latest content from its persistent storage

24(if any). If this content is too old (20 minutes), it returns an error.

25

2

1.4.2.2.5 **The user creates a proximity alert along its route to receive a sound in case of dangerous area**

3 **Preconditions**

4 A plug-in was already installed in the POI Service component to access an identified online service that provides dangerous area content (road conditions, weather, traffic jam, men at work) around a location.

5 The dangerous area is a category that is supported by the local database.

6

7 **Description**

8 The HMI configures a new POI proximity alert for dangerous events in front of the vehicle.

9 The HMI retrieves the vehicle position and speed from the **enhanced position** API.

10 The POI component defines the search radius in front of vehicle position depending on the category and the vehicle speed.

11 The POIService component gets the list of dangerous areas around the current position from the local database using the **map data access** API.

12 The POIService component gets the list of dangerous events around the position from content access module according to its defined radius and criteria.

13 The POIService component updates local events with dynamic contents if necessary.

14 A new list of POI is sent to the HMI with all the events (from local and from CAM) that match the alert configuration.

15 The HMI plays a sound when a new (non-empty) list of POI is received.

16

17 **Error Management**

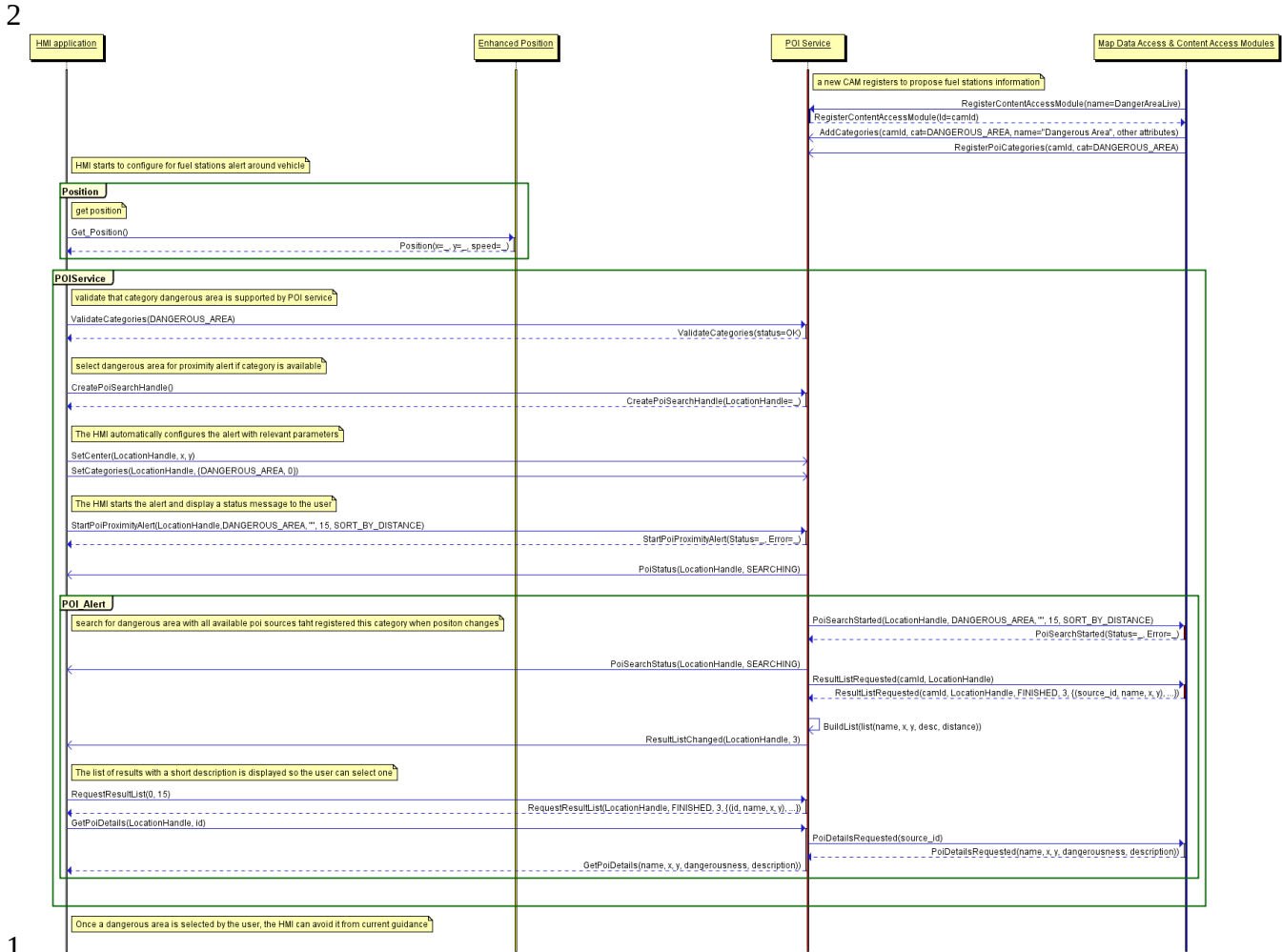
18 If the local database access is down, the component returns an error.

19 If the network connection to the online provider is down, the content access module tries to retrieve latest content from its persistent storage

20 (if any). If this content is too old (more than 2 minutes), it returns an error.

21 **Sequence diagram**

22



14.3 Interfaces

The following pages describe the interfaces of the POIService API.