



## Deliverable D 3.4

### Modular open source repository

<b>Project acronym:</b>	GoSAFE RAIL
<b>Starting date:</b>	01/10/2016
<b>Duration (in months):</b>	36
<b>Call (part) identifier:</b>	H2020-S2R-OC-CCA-04-2015
<b>Grant agreement no:</b>	730817
<b>Due date of deliverable:</b>	Month 30
<b>Actual submission date:</b>	15-09-2019
<b>Responsible/Author:</b>	Damian Harasymczuk, Timo Hartmann (CT)
<b>Dissemination level:</b>	PU
<b>Status:</b>	Issued

Reviewed: (yes)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the GoSAFE RAIL grant agreement No 730817.

The content of this document reflects only the author's view - the Joint Undertaking is not responsible for any use that may be made of the information it contains.

<b>Document history</b>		
<i>Revision</i>	<i>Date</i>	<i>Description</i>
1	15-09-2019	First issue
2	10-01-2020	Addressed the comments of the PO: <ul style="list-style-type: none"> <li>- Included information on funding and disclaimer</li> <li>- Added GA number to footer</li> <li>- Discuss the modules developed and the intention behind it in the executive summary</li> <li>- Provide the background of the work</li> <li>- Extended the objective / aim of the document – it is a technical guideline for developers</li> <li>- Included a list of modules in the executive summary</li> <li>- Described the converter module</li> <li>- Added a conclusion</li> <li>- Added a biography / reference list</li> </ul> Added some more detail about REST API interfaces
3	17.01.2020	Added disclaimer on the JU's non-responsibility

<b>Report contributors</b>		
Name	Beneficiary Short Name	Details of contribution
Damian Harasymczuk	CT	Main author of the document
Timo Hartmann	CT	Content and language editing

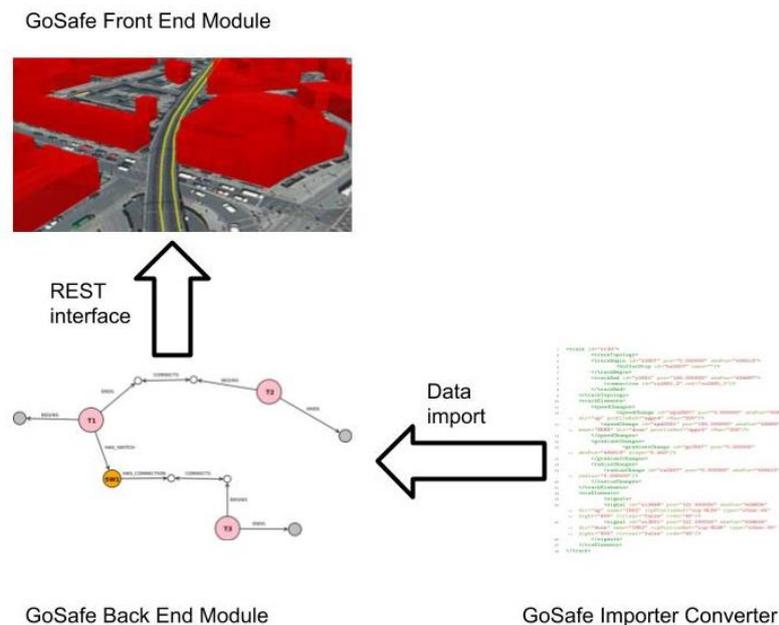


## Table of Contents

1.	Executive Summary .....	4
2.	Background .....	6
3.	Objective/Aim .....	7
4.	Prerequisites .....	8
5.	GoSafe Back End .....	9
6.	GoSafe Front End .....	11
7.	GoSafe Converter.....	12
8.	Docker .....	14
9.	Issues tracking.....	15
10.	Conclusion.....	16
11.	References .....	17

## 1. Executive Summary

To support agencies with implementing the GoSafe decision making procedure into a decision-making tool we provide an Open Source repository of required modules. These modules can serve as a starting point for developing solution that cater to the local context of agencies across Europe, but also form all the components of the implemented GoSafe platform.



**Figure 1: Overview of the GoSafe open source modules**

In detail, we provide

- An information management module that allows for representing different data standards widely supported within a distributed geographical database system. The module provides basic functionality for importing the required data required within the decision support module and persisting them. The module can serve as the data back-end for an implemented decision support system.
- The GoSafe Converter Module: A module that allows to import data from various file formats into the GoSafe information management module. This module allows developers to get started to develop interoperable interfaces between different data sources existing.
- The GoSafe Front End Module: A module to visualize information around railway assets within a state-of-the-art browser-based 3D geographical information system environment. This module can serve as a starting point to develop the user interfaces. This module that allows decision makers to remodel an existing railway network, by adding / removing nodes and links. This railway modelling part of the module is the starting point for implementing systems that allow users to establish different possible scenarios and compare them. Finally, this module allows for



the intuitive analysis of data related to a railway network using advanced machine learning methods. The GoSafe front-end also provides the form the basis for implementing advanced data analytics functionalities.

The connection between the modules is represented in Figure 1.

The Open Source repository contains all the modules creating the GoSafe platform. The source code is public and available for all users. Every module has a free software license – MIT, which has only minimum restrictions and allows the re-use of the modules. Most important, copying, modifying and using the modules for commercial purposes is permitted so other parties will be able to use the software to develop new market ready solutions.

The source code is hosted on Github. Github is the most popular platform that supports software development and can be used for sharing code. It allows allow others to contribute to the further development of the code by proposing new changes or reporting issues. Github accounts to contribute in GoSafe project can be created for free so everybody can contribute to the modules in the future.

All the modules are built upon free and open source software and libraries and can be run internally on any machine without any additional costs. Although the GoSafe developers do not provide any public databases that are necessary to run and work with these modules.



## 2. Background

The present document constitutes the Deliverable D3.4 “Modular Open Source Repository” in the framework of the TD 3.6, task 3.6.2 IT architecture and 3.6.4 Open standard interfaces, and CCA Work Area 3 Safety, Standardisation, Smart Maintenance, Smart Materials & Virtual certification, Sub WA 3.1 Safety, Subtask WA3.3.3 Data selection and formatting (*Shift2Rail Joint Undertaking, Multi-Annual Action Plan, Brussels, November 2015*).



### 3. Objective/Aim

This document has been prepared to provide a technical guide that will allow software developers to start using the developed modules. For each of the modules we will describe how they can be set up and run in a local development environment. This will provide software developers a starting point for the development of their own decision-making platforms.

## 4. Prerequisites

To set up the modules within a local development environment several prerequisites need to be fulfilled:

1. It is required to download and install git from: <https://git-scm.com/downloads> then configure it properly. Git homepage's documentation section provides all necessary information. Downloading GoSafe repositories can be performed by running `git clone` command.
2. The GoSafe platform requires a Neo4j graph database, running locally or on cloud. The database must be connected to the back-end module as described in the following sections. Neo4j can be downloaded from <https://neo4j.com/download>
3. Node JS and NPM are required to run back-end and front-end modules. NPM can be downloaded from <https://docs.npmjs.com/> and Node from <https://nodejs.org/en/download>. The recommended Node version is 8, however it is compatible with other versions.
4. The GoSafe Converter is built in Go programming language that requires Go tools installed. Homepage contains necessary guides: <https://golang.org/doc/install>

All the tools described above are multiplatform, that means it is possible to run every part of the GoSafe platform under three different environments: Windows, GNU/Linux, MacOS. Although we strongly recommend using GNU/Linux for both local and especially for server.

## 5. GoSafe Back End

The GoSafe Back End repository is available at:

<https://github.com/ContecltGMBH/GoSAFE.back>

To download the backend via git, execute a command:

```
git clone https://github.com/ContecltGMBH/GoSAFE.back.git
```

Then navigate to the downloaded folder:

```
cd GoSAFE.back
```

And install all the necessary libraries:

```
npm install
```

Before running the application, the module has to be linked to the installed Neo4j database url that needs to be added to the modules configuration file located at *config/config.development.js* by adding a new line:

```
config.graph_database = "<url>"
```

Where <url> is a valid url to the database that includes credentials and port (for example <http://user:password@localhost:7474/db/data>).

At this point the app is ready to start:

```
npm start
```

The module provides the following representational state transfer (REST) interfaces for interacting with the database:

- @api {GET} /api/v1/tracks
- @apiDescription Get all tracks from the database (all nodes with the 'Track' label). Executes a cypher query.
- @apiName GetTracks
- @apiGroup Neo4j
- @apiSuccess (200) {array} tracks an array of objects (graph nodes), representing the railway network
- @api {POST} /api/v1/elements

- @apiDescription Get all nodes related to the given track. Relation between the track and elements are RailMI relations. Executes a cypher query.
- @apiName GetTrackElements
- @apiGroup Neo4j
- @apiParam {string} id a track id
- @apiSuccess (200) {array} elements an array of objects (graph nodes), representing the given track and related elements
- @api {POST} /api/v1/add/element
- @apiDescription Creates a new node related to the given track. Executes a cypher query.
- @apiName AddElement
- @apiGroup Neo4j
- @apiParam {string} label a label for a new node
- @apiParam {string} relationship relationship between a new node and the given track
- @apiParam {string} track\_id given track id
- @apiParam {string} properties new node properties
- @apiSuccess (200) {boolean} created returns true
- @api {POST} /api/v1/remove/element
- @apiDescription Removes given element from the database. Executes a cypher query.
- @apiName RemoveElement
- @apiGroup Neo4j
- @apiParam {string} label a label for the removed node
- @apiParam {string} id an id for the removed node
- @apiSuccess (200) {boolean} removed returns true
- @api {POST} /api/v1/remove/track
- @apiDescription Removes given track from the database. Executes a cypher query.
- @apiName RemoveTrack
- @apiGroup Neo4j
- @apiParam {string} id an id for the removed node
- @apiSuccess (200) {boolean} removed returns true

The GoSafe front-end module uses these interfaces.



## 6. GoSafe Front End

The GoSafe FrontEnd repository is available at:

<https://github.com/ContechtGMBH/GoSAFE.front>

To download it via git, execute a command:

```
git clone https://github.com/ContechtGMBH/GoSAFE.front.git
```

Navigate to the downloaded folder:

```
cd GoSAFE.front
```

And install all the necessary libraries:

```
npm install
```

At this point the application is ready to start in development mode or be deployed into production into server. Development mode can be started after executing:

```
npm start
```

For production it's strongly recommended to build the static app and host it with Nginx server. Build can be created with:

```
npm run build
```

Result will be created in a new folder "public" which should contain html, javascript and css file.



## 7. GoSafe Converter

GoSafe Converter repository is available at:

[https://github.com/ContechtGMBH/Go-GoSAFE\\_converter](https://github.com/ContechtGMBH/Go-GoSAFE_converter)

To download it via git, execute a command:

```
git clone https://github.com/ContechtGMBH/Go-GoSAFE_converter.git
```

GoSafe Converter must be connected to the same Neo4j database as GoSafe Back End. To do this, user must provide a valid link with credentials and port number in *config/db.go* file.

To run or build the application locally the Go environment must be configured properly and the main folder should be a part of the Go workspace (Go documentation: <https://golang.org/doc>).

To install all the required dependencies, navigate to the root folder:

```
cd $PATH_TO_GO_WORKSPACE/src/Go-GoSAFE_converter
```

And execute command:

```
go get ./..
```

To run the app in development environment:

```
go run main.go
```

For production or any staging server it's recommended to build a binary executable file:

```
go build -o gosafeconverter
```

This command creates a compressed file for the given platform. It can be started by executing (for GNU/Linux):

```
./gosafeconverter
```

The converter provides the following REST API interfaces:

- @api {POST} /api/v1/import/railml
- @apiDescription Converts a RailML file to neo4j graph
- @apiGroup Railml
- @apiName ConvertRailml
- @apiParam {string} line A line name

- @apiParam {string} epsg CRS EPSG number, should match geooCoords CRS
- @apiParam {file} file A valid RailML file that contains the Infrastructure subschema
- @apiSuccess (200) {json} object Response message with the number of extracted tracks
- @api {POST} /api/v1/export/railml@apiDescription Export a subgraph to the railml file
- @apiGroup Railml@apiName ExportRailml
- @apiParam {string} line A line name
- @apiSuccess (200) {XML} RailML A valid RailML document that describes the given line
- @api {POST} /api/v1/import/spatial @apiDescription Converts a spatial file into R-Tree graph
- @apiGroup Spatial
- @apiName ConvertSpatial
- @apiParam {string} name Layer Name
- @apiParam {string} format File format
- @apiParam {string} epsg CRS EPSG number, should match geooCoords CRS
- @apiParam {file} file A spatial file, one of the following formats: (GityGML, Shapefile, GeoJSON)
- @apiSuccess (200) {json} object Response message with the number of extracted features

The functionality of the interfaces is described as code comments.



## 8. Docker

All the applications contain Dockerfiles which allow to run it in an isolated container.  
More information about Docker including install and configuration guide:

<https://docs.docker.com>

To build a docker container run:

```
docker build --network="host" .
```

The docker image provides a complete configuration of all GoSafe modules within an easy to deploy package.



## 9. Issues tracking

The Github platform provides an intuitive and easy to navigate issue tracker where each problem can be reported and developers can get help, find useful information or report an issue.

GoSafe Front End:

<https://github.com/ContechtGMBH/GoSAFE.front/issues>

GoSafe Back End:

<https://github.com/ContechtGMBH/GoSAFE.back/issues>

GoSafe Converter:

<https://github.com/ContechtGMBH/Go-GoSAFE.converter/issues>

Issues can be created regarding to existing bugs, suggestions, and possible improvements.



## 10. Conclusion

This report provides the technical documentation of how to set up the GoSafe Open Source modules. Developers can use the information in this document to get started developing their own solutions by setting up a basic development environment for the modules. The document also describes the Github platform on which the modules are hosted and explains the possibilities to get involved with the continuous development of the modules in the future.



## 11. References

Not applicable.