Development notes

Content

[1 Repository 1](#_Toc57031701)

[2 Web application directory structure 1](#_Toc57031702)

[3 URL structure 1](#_Toc57031703)

[4 Database 2](#_Toc57031704)

[5 Class diagram 2](#_Toc57031705)

[6 Process logistic scheme 3](#_Toc57031706)

[7 Possible extensions 3](#_Toc57031707)

# Repository

All project data is stored in the github repository: <https://github.com/bubilem/elis>.

# Web application directory structure

The directory structure is designed with respect to the object-oriented application and the support of the MVC architecture.

* **app/elis** (elis source code directory)
  + controller (controller classes, main logic)
  + component (component classes for presenters, stand-alone units)
  + presenter (controller classes, preparing the pages, inserting data to templates)
  + model (ORM, database model classes)
  + template (html templates)
  + utils (support classes)
* **code-lists** (code lists for roles, event types, package states, countries…)
* **dict** (dictionary files for localizing the application to other languages)
* **files** (other files, reports…)
* **logs** (system log files for monitoring)
* **www** (directory with web access)
  + css (generated minified cascade style sheets)
  + img (images, favicons…)
  + js (java script files)
  + scss (sass source files)

# URL structure

The web application supports COOL format URL addresses.

|  |  |  |
| --- | --- | --- |
| URL part | meaning | note |
| https:// | protocol |  |
| [www.domain.eu](http://www.domain.eu) | domain address | domain TBA |
| /presenter | main launch presenter |  |
| /parameter | first parameter | optional |
| /parameter | second parameter | optional |
| /parameter | third parameter | optional |
| ?a=b&c=d | GET data | optional |

## Examples

* www.domain.eu
* www.domain.eu/login
* www.domain.eu/error/404
* www.domain.eu/home

## URL in administration

|  |  |  |
| --- | --- | --- |
| URL part | meaning | note |
| https:// | protocol |  |
| [www.domain.eu](http://www.domain.eu) | domain address | domain TBA |
| /adm-vehicle | administration presenter | adm-dashboard  adm-user  adm-vehicle  adm-place  adm-package  adm-route  adm-event |
| /edit | presenter method | new-form  new  edit-form  edit  delete-quiestion  delete  table  … |
| /4 | first parameter | optional  record id |

# Database

We use MySQL DBMS, SQL language.

For E-R Scheme see external files.

# Class diagram

See external files (core, utils).

# Process logistic scheme

See external file.

# Possible extensions

* end customers
* wares in packages
* vehicle logs
* other language dictionaries