



HoliDes



Holistic Human Factors **D**esign of
Adaptive Cooperative Human-
Machine Systems

HoliDes Platform Builder functional analysis

Project Number:	332933
Classification:	Confidential
Work Package(s):	WP1
Milestone:	M5
Document Version:	V5.0
Issue Date:	September 2016
Document Timescale:	Project Start Date: October 1, 2013
Start of the Document:	Month 16
Final version due:	Month 23
Deliverable Overview:	Main document: D1.7-HF-RTP Version 2.0 inc. Methodology and Requirements Analysis update. <Confidential>
Compiled by:	Nacho González – ATOS
Authors:	Nacho González – ATOS, Jordi Fonoll - ATOS
Reviewers:	Morten Larsen - AWI, Fabio Tango - CRF
Technical Approval:	Jens Gärtner, Airbus Group Innovations
Issue Authorisation:	Sebastian Feuerstack, OFF

© All rights reserved by HoliDes consortium

This document is supplied by the specific HoliDes work package quoted above on the express condition that it is treated as confidential to those specifically mentioned on the distribution list. No use may be made thereof other than expressly authorised by the HoliDes Project Board.

	<p>HoliDes</p> <p>Holistic Human Factors Design of Adaptive Cooperative Human- Machine Systems</p>	
---	--	---

RECORD OF REVISION		
Date (DD.MM.JJ)	Status Description	Author
15.01.2015	First version of the document	Nacho González - ATOS
16.01.2015	Added no functional requirements	Nacho González - ATOS
21.01.2015	Completed system architecture overview	Nacho González - ATOS
17.03.2015	Added user interface diagrams and descriptions	Nacho González - ATOS
19.03.2015	Review and changes to input mockups	Linda Onnasch - HFC
27.03.2015	Added section Data Model	Jordi Fonoll - ATOS
10.06.2015	PB app version 1.0	Jordi Fonoll - ATOS
09.11.2015	PB new design	Jordi Fonoll - ATOS
09.11.2015	PB app version 1.1	Jordi Fonoll - ATOS
16.12.2015	Added tool chain to My Projects. PB app version 2.0	Jordi Fonoll - ATOS
04.04.2016	Added connectivity functionality. PB app version 3.0	Jordi Fonoll - ATOS
09.05.2016	Added regulations functionality. PB app version 4.0	Jordi Fonoll - ATOS
16.08.2016	Added land page, new menu, navigation bar in the PB app version 5.0	Jordi Fonoll - ATOS





	<p>HoliDes</p> <p>Holistic Human Factors Design of Adaptive Cooperative Human- Machine Systems</p>	

Table of Contents

1	Introduction	6
1.1	Purpose.....	6
1.1.1	What is the system described actually intended for?	6
1.1.2	What are the different components of the PB?.....	7
1.1.3	What functionalities will be implemented?.....	8
1.1.4	Who will use the different components and what is their knowledge?.....	9
1.2	Terminology.....	10
1.3	Application description	10
1.4	Client tier	12
1.5	Business logic tier.....	12
1.6	Database tier	13
2	System architecture overview	14
2.1	Controller layer	15
2.2	Model layer.....	15
2.3	View layer	15
2.4	Technological relation.....	15
3	Graphical user interfaces.....	17
3.1	Application description	17
3.2	GUI - main	17
3.3	GUI - Land page.....	18
3.4	GUI – Search MTTs.....	21
3.5	GUI – MTTs database.....	22
3.6	GUI - HF-RTP instance	23
3.7	GUI - My projects	24
3.8	GUI – Statistics	25
3.9	GUI – Connectivity.....	26
3.10	GUI – Adaptation framework	27
4	DataBase model.....	28
4.1	Entity/Relationship Diagram.....	28

List of Figures

Figure 1 PB conceptual design	10
Figure 2 Layered architecture	12
Figure 3 MVC architectural pattern	14
Figure 4 Main screen	17
Figure 5 Land page	18
Figure 6 Menu bar.....	19
Figure 7 PB navigation bar help menu	20
Figure 8 Input page	21
Figure 9 List of MTTs	22
Figure 10 HF-RTP instance	23
Figure 11 Project details	25
Figure 12 Statistics screen	25
Figure 13 MTTs connectivity	26
Figure 14 Adaptation framework	27
Figure 15 Entity/Relationship model	29

	<p>HoliDes</p> <p>Holistic Human Factors Design of Adaptive Cooperative Human- Machine Systems</p>	
---	--	---

1 Introduction

This chapter remains the purpose of the document and list the parties involved.

1.1 Purpose



The purpose of this document is to guarantee that all the parties involved agree on the answers to the following questions regarding the HoliDes PB.

1.1.1 What is the system described actually intended for?

The system described actually is intended for HF experts who want to find MTT for their purposes.

For this reason a web page called: PB app, is created to give to the HF experts the tools necessities to get a list of MTTs(methods technics and tools) in a friendly way(web application accessible from different devices).

The PB app design and structure are defined in the document:
"HoliDes Platform Builder requirements"

	<p>HoliDes</p> <p>Holistic Human Factors Design of Adaptive Cooperative Human- Machine Systems</p>	
---	---	---

1.1.2 What are the different components of the PB?

- The web app: PB app www.holidays.eu/PlatformBuilder.
- The database server: "Platform Builder" located in the MySQL server: <http://arisrv32.es.atos.net/phpmyadmin/>, only the database administrator has rights to manage the database.
- The documentation. Divided in 4 documents:
 - Analysis document:
 - Describes the description of the project.
 - The functionalities.
 - The GUI used in the PB.
 - Requirements document:



Including:

 - Software engineering: the architecture layers involved and the software components used.
 - Requirements: functional and no functional.
 - Uses cases.
 - Diagrams: sequence diagrams, data flow diagrams and state-transition diagrams.
 - Validation report document:

It's a test document checking that all the requirements described in the requirements document are working fine. For each validation is attached an image to check that a test is done for each requirement specified.
 - Hand book document.

It's a user tutorial. Explains how it's working the PB app, including:

 - The URL of the PB app.
 - A couple of users/passwords to access to the PB.
 - An explanation for each PB app screen, including images.

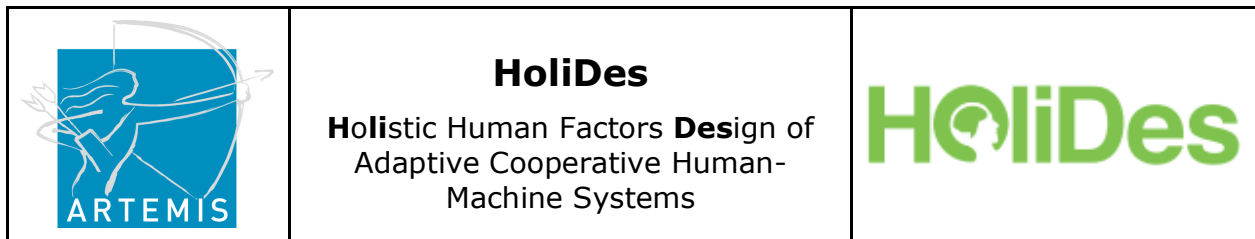
	<p>HoliDes</p> <p>Holistic Human Factors Design of Adaptive Cooperative Human- Machine Systems</p>	
---	--	---

1.1.3 What functionalities will be implemented?

The PB is helping the final user to find the appropriate MTTs for their purposes, so, the PB has been created as a solution for improving the configuration and instantiation capabilities of the HoliDes HF-RTP.

Then the functionalities to help the HF experts are:

- Instantiation of a HF-RTP.
- MTTs management (inserting, deleting and retrieving MTTs in the PB database), in order to help the HF experts to know which are the indicate MTTs for their purpose.
- Possibility to save the search done in the PB app as a project.
- Get statistics of the number of MTTs used by the user logged in the PB app.
- Possibility to print or export the instantiation of a HF-RTP to a common format files (.xls, .csv).
- Connectivity between MTTs in order to give a vision in a graphical environment.



1.1.4 Who will use the different components and what is their knowledge?

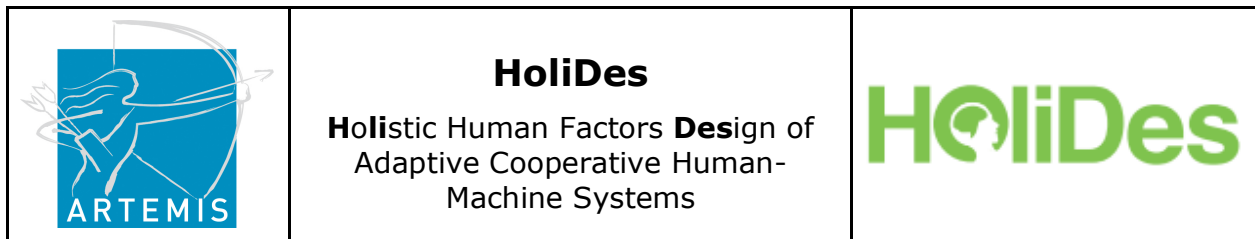
The HF experts are the users that have rights to access to the PB app.

For each user has provided a different user/pwd to control the access to the PB app.

A user role administrator has created to manage all the PB database entities with rights to do "CRUD" operations: create, read, update and delete, for example: create a new table, view the relations between, delete users, insert a new "HF issue", etc.

They should be used to work with HF knowledge. The descriptions of the MTTs are related with HF terminology.

It's important to remark that the HF experts have a handbook tutorial in case those HF experts don't know how it's working the PB app. The handbook tutorial is included in the PB app, clicking on the menu bar, option handbook.



1.2 Terminology

Specific terminology used:

HF-RTP: human-factors reference technology platform.

UI: user interface.

BL: business logic.

DB: data base.

LTE: long term evolution.

MVC: model-view-controller.

MTT: methods technic and tools.

PB: Platform Builder.

GUI: graphical user interface.

HF: human factor.

PB app: Platform Builder application.

J2EE: java platform enterprise edition.

1.3 Application description

The HoliDes PB is a solution for improving the configuration and instantiation capabilities of the HoliDes HF-RTP. Based on a HF problem description, the PB aims to set up an instantiation of the HF-RTP for a specific domain project.

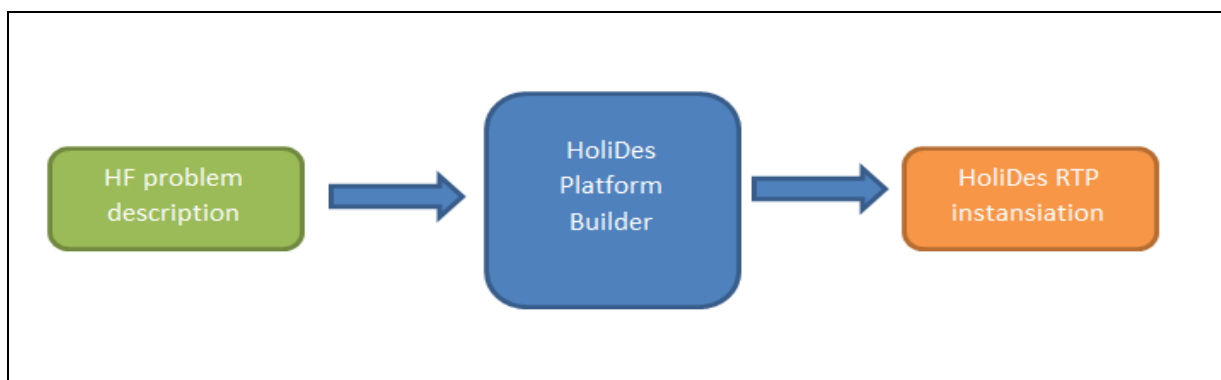
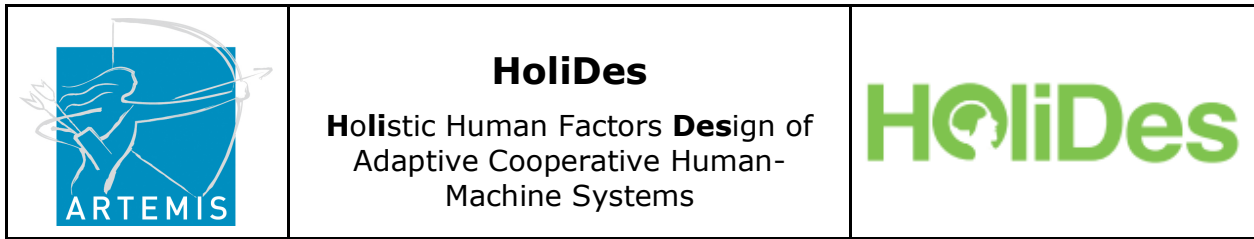


Figure 1 PB conceptual design



For the accomplishment of the objectives and goals of the PB, a web application will be developed using a three-tier architecture. With this architectural approach, all three tiers are separated onto different computers. The UI runs on the client (what the user is working with).

The BL is running on a separate server, called the business logic tier, middle tier, or service tier. Finally the DB is running on its own database server.

In the client-server solution the client was handling the UI and the BL that makes the client “thick”. A thick client means that it requires heavy traffic with the server, thus making it difficult to use over slower network connections like Internet and Wireless (4G, LTE, or Wi-Fi).

By introducing the middle tier, the client is only handling presentation logic (UI). This means that only little communication is needed between the client and the middle tier (BL) making the client “thin” or “thinner”. An example of a thin client is an Internet browser that allows you to see and provide information fast and almost with no delay.

As more users access the system a three-tier solution is more scalable than the other solution because you can add as many middle tiers (running on each own server) as needed to ensure good performance (N-tier or multiple-tier).

Security is also the best in the three-tier architecture because the middle tier protects the database tier.

There is one major drawback to the N-tier architecture and that is that the additional tiers increase the complexity and cost of the installation.

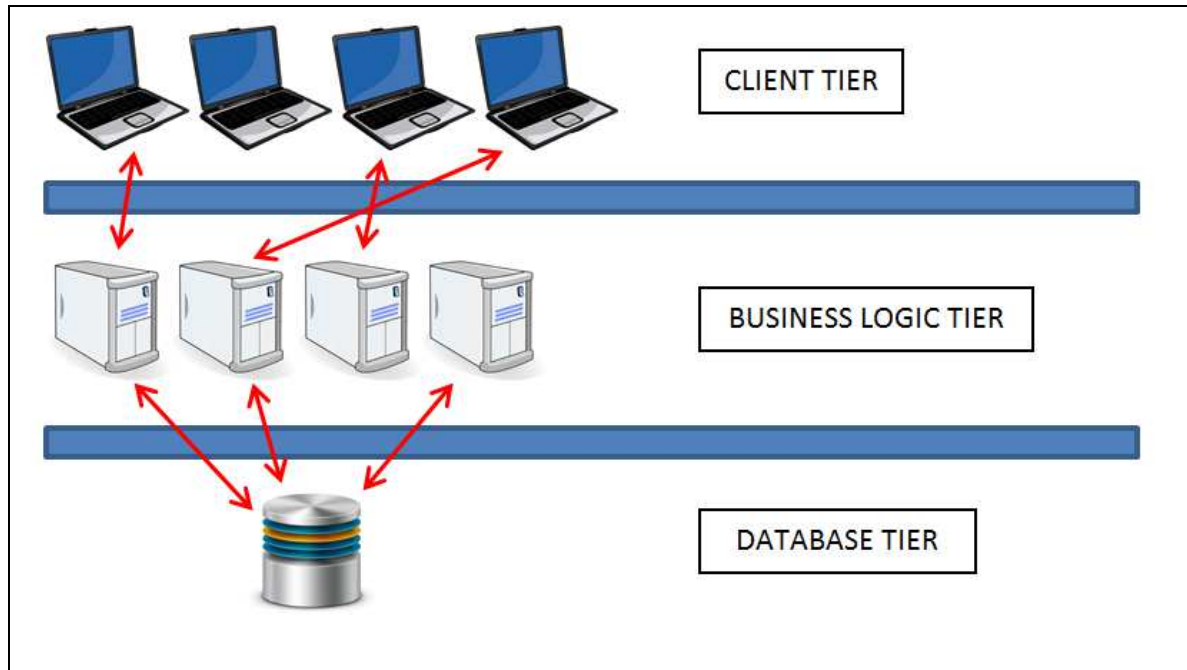




Figure 2 Layered architecture

1.4 Client tier

This is the topmost level of the PB tiered architecture. The client tier displays information related to such services as the whole RTP instantiation process information. It communicates with other tiers by which it puts out the results to the browser/client layer and all other tiers in the network. (In simple terms it is a tier which users can access directly such as a web page, or an operating systems GUI)

1.5 Business logic tier

This tier encodes the real-world business rules that determine how data can be created, displayed, stored and changed. The processing of the different requests done by the final user will be done in this tier, using dedicated web services, which at the same time, will use the resources provided by the database layer.

	<p style="text-align: center;">HoliDes Holistic Human Factors Design of Adaptive Cooperative Human- Machine Systems</p>	
---	--	---

1.6 Database tier

In this tier the information is stored and retrieved from the database. This layer keeps data neutral and independent from application servers or business logic. Giving data on its own tier also improves scalability and performance.

2 System architecture overview

The final mission of the PB is helping the final user to find the appropriate MTTs for their purposes (e.g. fulfilment of domain specific regulations, dealing with HF problems, designing of a new system... so a system able to provide advanced user interfaces management is needed. For this reason, the software architectural pattern that will be used in the design and development phases will be Model-View-Controller (MVC).

This pattern divides the web application into three interconnected parts, so as to separate internal representations of information from the ways that information is presented or accepted from the user.

The model captures the behaviour of the application in terms of its problem domain, independent of the user interface. The model directly manages the data, logic and rules of the application. A view can be any output representation of information, such as a chart or a diagram; multiple views of the same information are possible, such as a bar chart for management and a tabular view for accountants. The third part, the controller, accepts input and converts it to commands for the model or view.

Using Java technology, the architecture can be represented in the next figure:

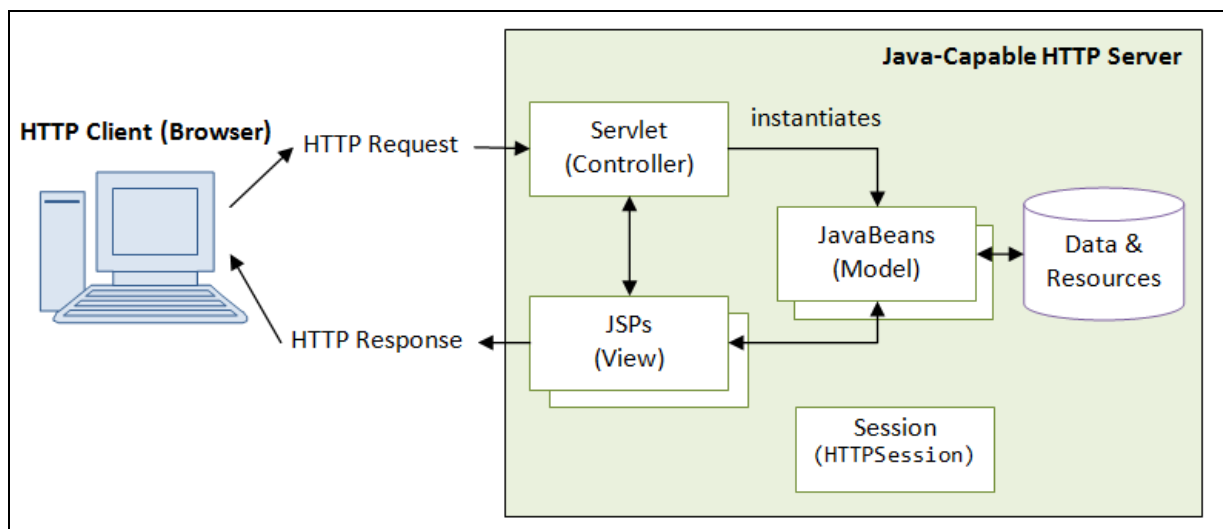
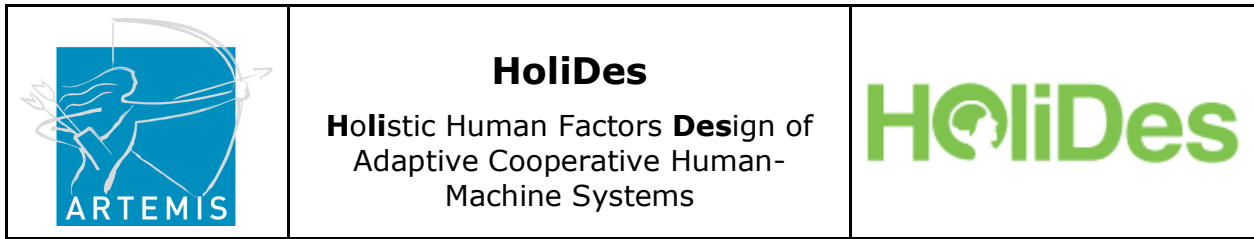


Figure 3 MVC architectural pattern



2.1 Controller layer

The controller layer sends commands to the model to update the model's state.

The servlets in the PB are developed in java and using J2EE architecture.

2.2 Model layer

The model layer stores data that is retrieved according to commands from the controller and displayed in the view.

The model layer uses object beans class to manage all entity in the J2EE.

2.3 View layer



The view layer generates an output presentation to the user based on changes in the model.

The view layer uses: html, jQuery, datatable jQuery to give a professional view based on responsive design (CSS bootstrap) adaptable automatically for different devices(laptops, pc, smartphones, tablets) and navigators(Chrome, Firefox, Safari, Internet explorer)

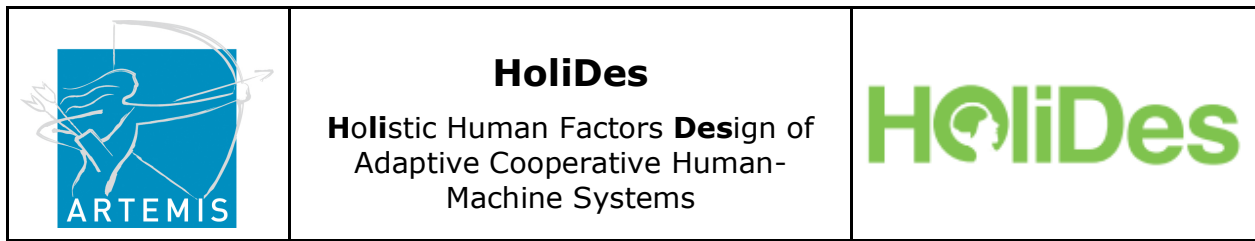
2.4 Technological relation

List of the relation between the software used and the layers involved.

Layer	Technological constraints
Controller layer	Java JRE v.1.7, bean objects
Model layer	Java JRE v.1.7, MySQL Server v.5.5

	<p style="text-align: center;">HoliDes</p> <p style="text-align: center;">Holistic Human Factors Design of Adaptive Cooperative Human- Machine Systems</p>	
---	---	---

<p>Client layer</p>	<p>JavaScript jQuery v1.11.1 Bootstrap CSS v3.3.5 Google chart tools DataTable plugin for jQuery v1.10.7</p>
----------------------------	--



3 Graphical user interfaces

3.1 Application description

The PB app will have several screens in order to facilitate the user's experience and ease the navigation between them.

There is a main screen where the user can select if he wants to explore the HF ontology or create a customized HF-RTP instantiation. Then, the process goes through different ways, so we will describe each screen individually and finally we'll detail the navigation flow.

3.2 GUI - main

GUI to give to the user the possibility to access to the PB app options, using the user/pwd provided.



Log in

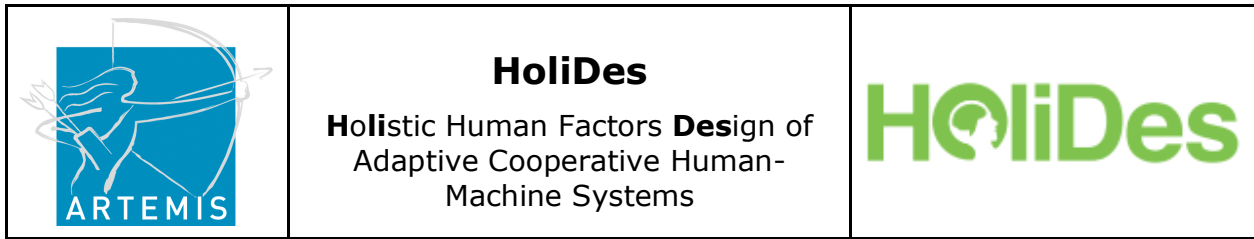
Username:

Password:

[Log in to the Platform Builder](#)

[Sign up](#) | [Forgot Password?](#)

Figure 4 Main screen



3.3 GUI - Land page

This screen provides 3 possible options, redirect to:
 "Create an HF-RTP Instance"
 "Information & How to"
 "MTT Database"

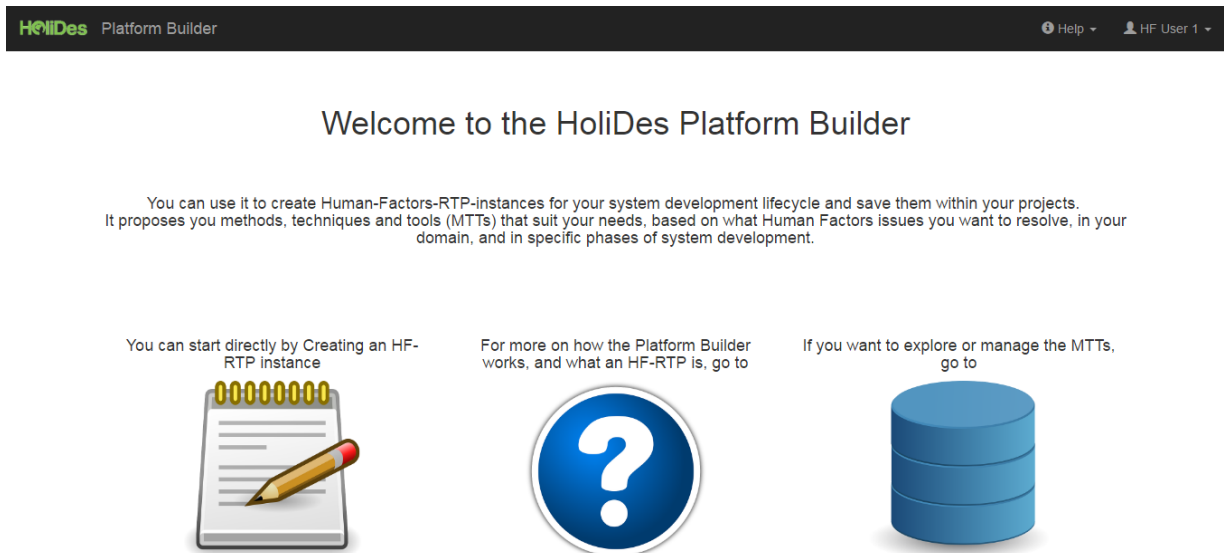


Figure 5 Land page

The option "Information & How to" redirects to the PB video tutorial channel. Selection one of the options "Create an HF-RTP Instance" or "MTT Database", a menu bar will appear in the left size:

PB app GUI provides a menu bar to navigate easily between the different options.

View of the menu bar:

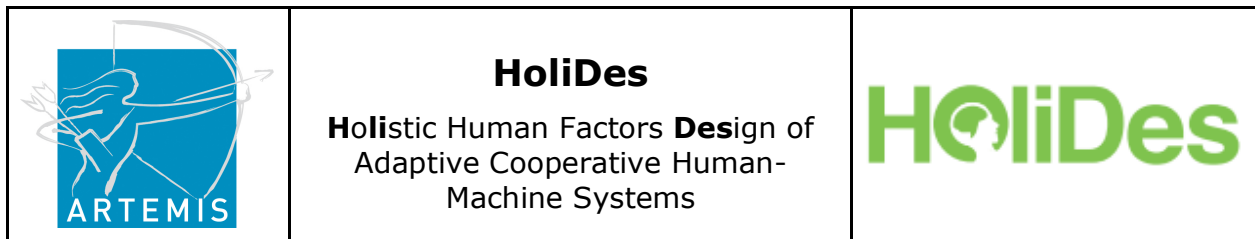


Figure 6 Menu bar

The users here have several options:

- Application selection
 - Create an Instance HF-RTP: the user will go through a process where he will get a customized HF-RTP with a set of methods, techniques and tools for solving his particular HF problematic.
 - Get my projects: clicking here the user can get his projects saved before.
 - MTTs database: The user can insert a new MTT into the PB, also, can get all the MTTs inserted in the PB.
 - Statistics: provides (in graphic mode) a number of MTTs used by user, selecting a domain previously.
 - MTTs connectivity: Show the MTTs selected in a graphic. Using a V-Model picture the MTTs will be included in one of these categorizations: analysis, conceptualization, design, system implementation, evaluation, certification and deployment.
 -

Documentation placed in the help menu dropdown list:

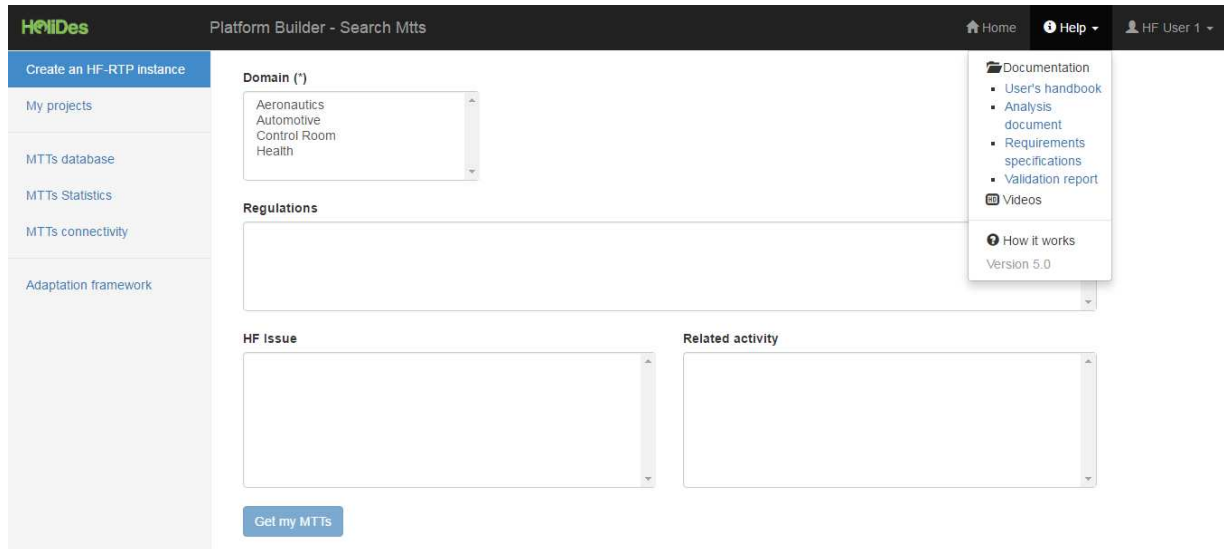
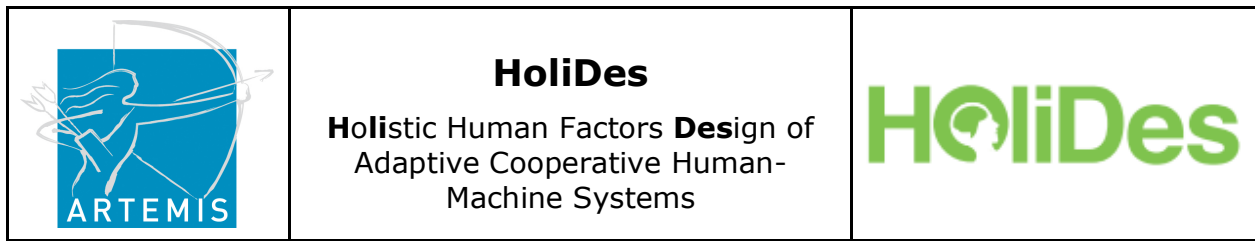


Figure 7 PB navigation bar help menu

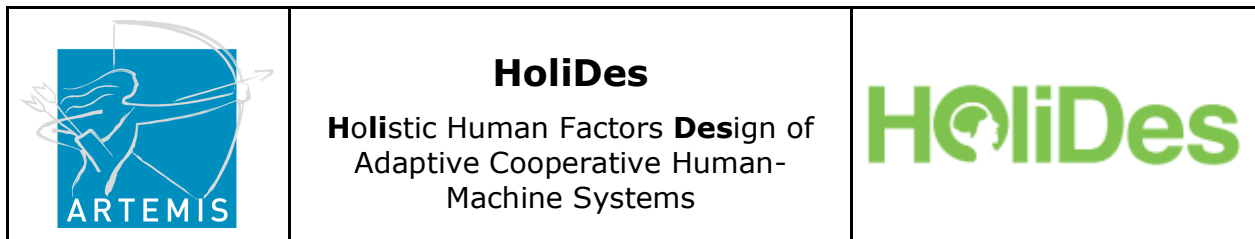
- Documentation: here the user will have access to different documentation, reports and training material. This documents will be:
 - User's handbook.
 - Analysis document.
 - Requirements specification.
 - Validation report.
- Video tutorials:
 - A channel video has been created to include the PB video tutorials.

Click in the link below to access to the video channel:

https://www.youtube.com/playlist?list=PLzBvpl_o7OobL6VCo7ur-sm9BREddw4jP3

also in the ajaxplorer:

/4-WorkPackages/WP01/WorkingDocuments/Platform Builder/Video tutorials



3.4 GUI – Search MTTs

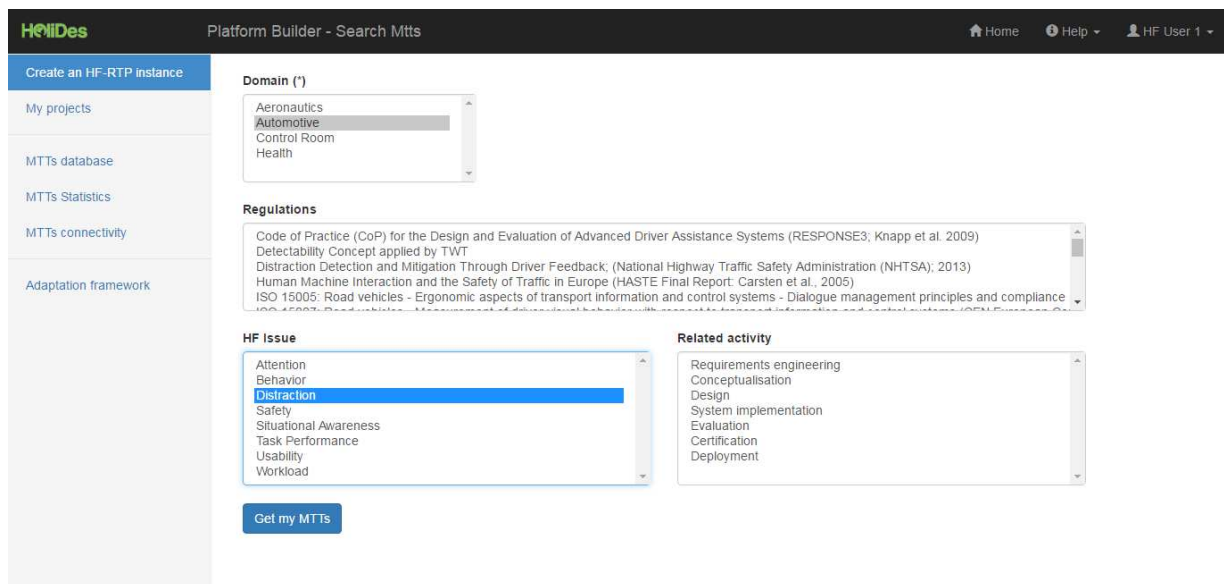
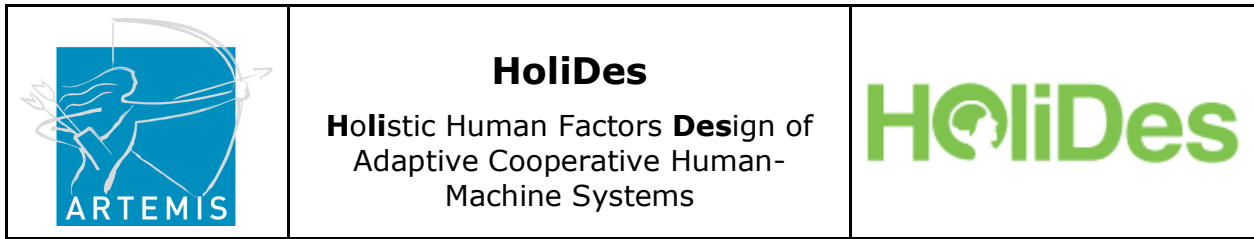


Figure 8 Input page

In this screen of the PB, the user will have access to different functionalities:

- Select the domain for his particular HF problematic.
- Select (if needed) one HF issue for his particular HF problematic
- Select (if needed) one Regulation for his particular HF problematic.
- Select a related activity.
- Get a list of MTTs as first step for his HF-RTP instantiation.



3.6 GUI - HF-RTP instance

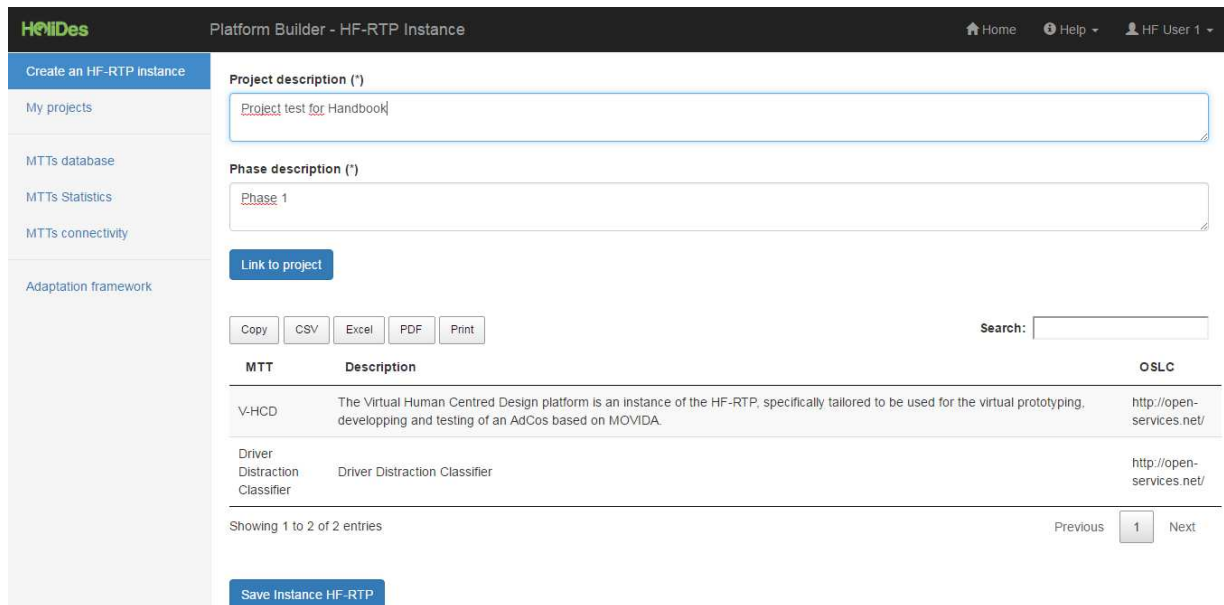
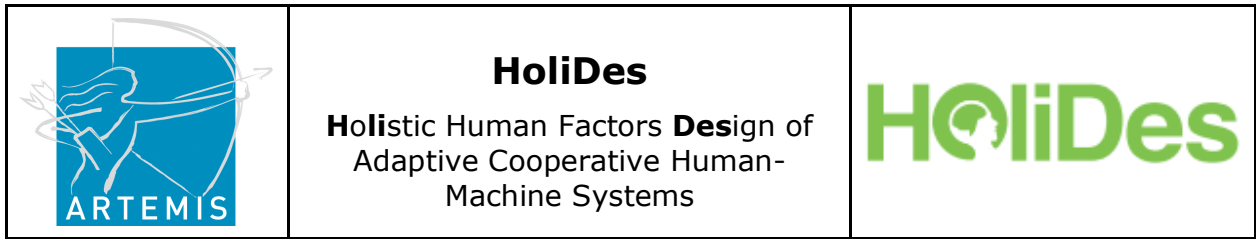


Figure 10 HF-RTP instance

This screen will show the final result of the HF-RTP instantiation process. There are information related with data entered by the user, and the list of MTTs resulting.



3.7 GUI - My projects

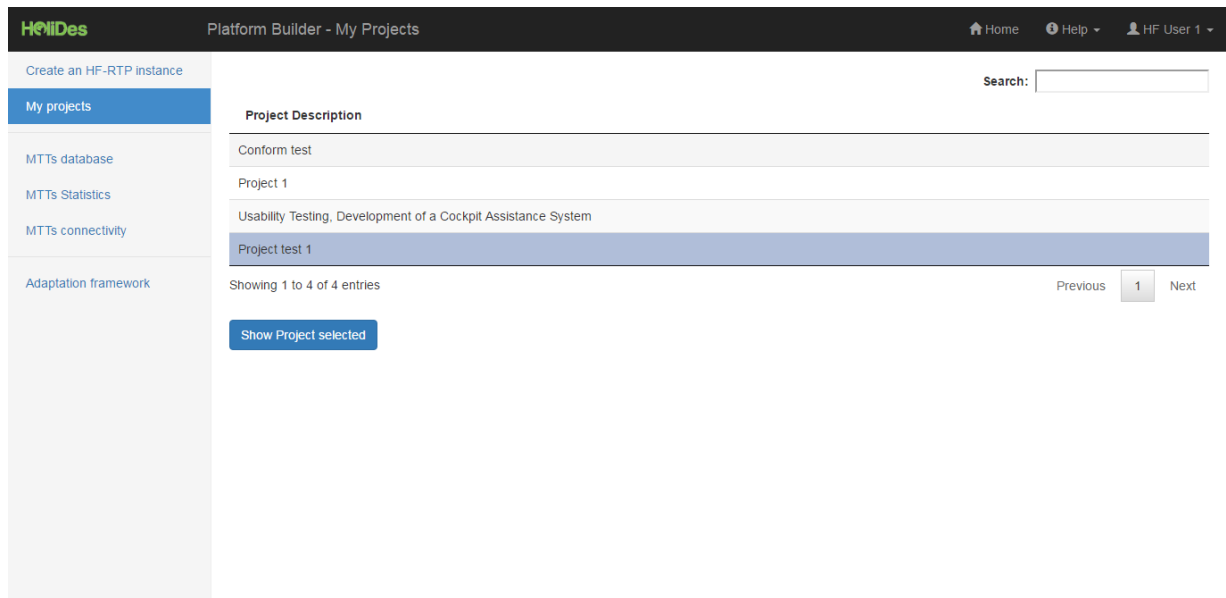
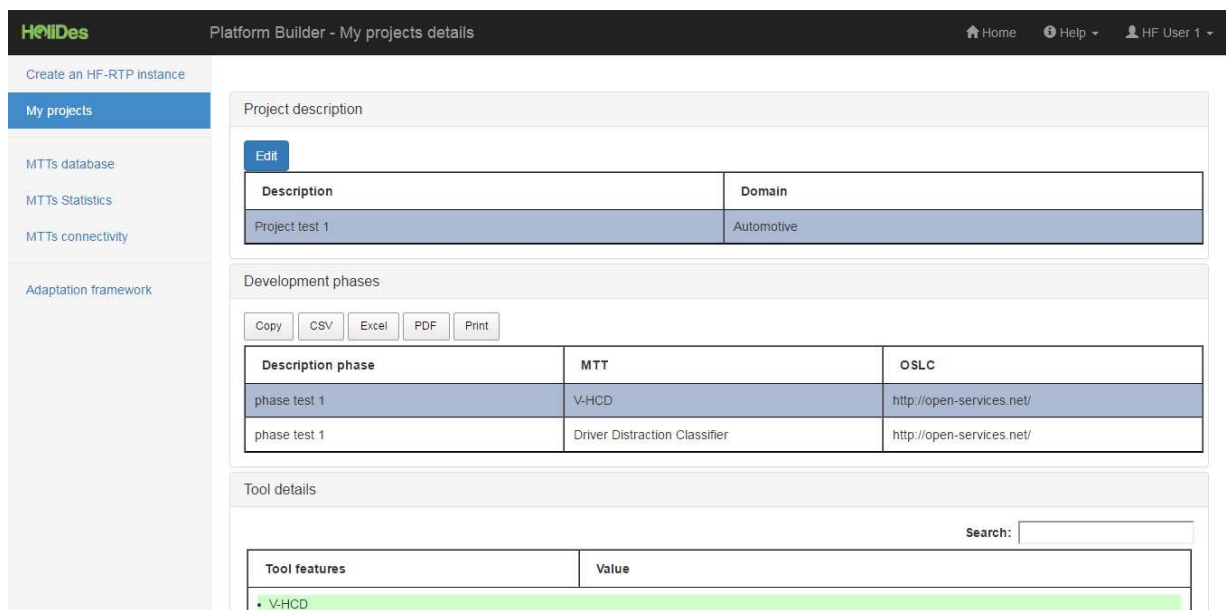


Figure 8 Get my projects

This screen will shows the list of Projects saved by User. Clicking on a project you can get the project details.



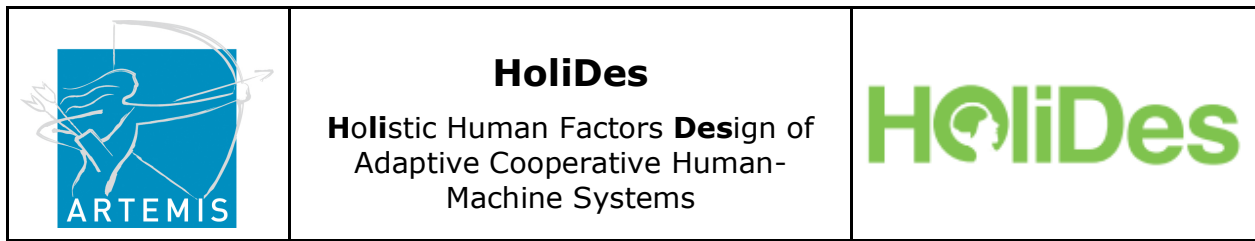


Figure 11 Project details

3.8 GUI – Statistics

Get the number of the MTTs used by user and divided by domain. The result is shown in a graphical environment.

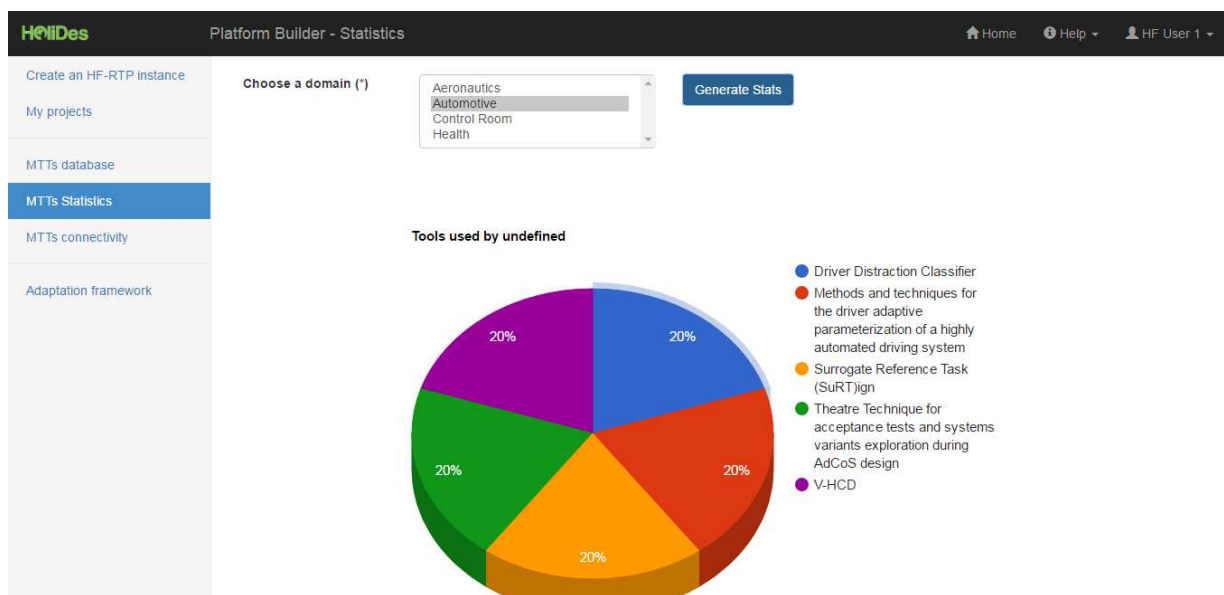
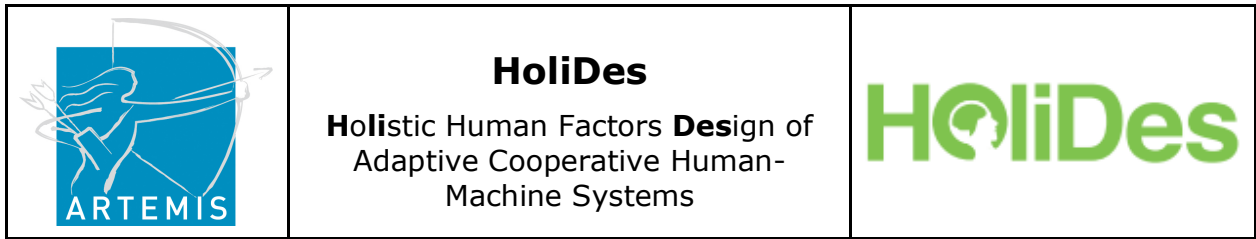


Figure 12 Statistics screen

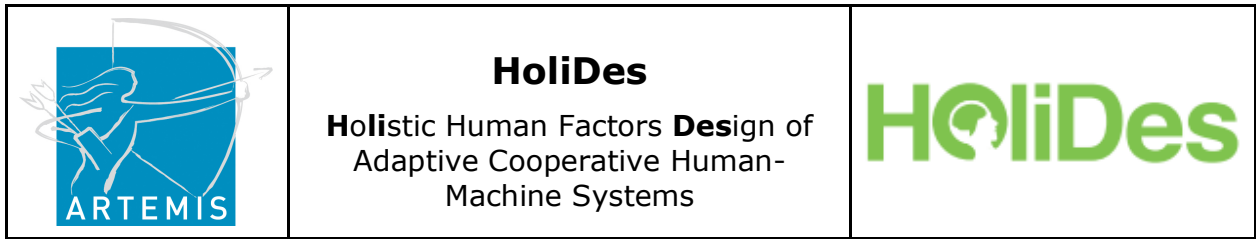


3.9 GUI – Connectivity

Get the MTTs filtered by domain, HF issue or related activity. The result is shown in a graphical environment.

The screenshot displays the 'Platform Builder - MTT connectivity' interface. At the top, there are navigation links for Home, Help, and the user profile (HF User 1). Below the header, there are three filter menus: 'Domains' with 'Automotive' selected, 'HF Issues' with 'Behavior' and 'Distraction' selected, and 'Related activity' with 'Design' selected. The main area features a large diagram showing the connectivity of MTTs across various stages: Analysis, Conceptualization, Design, Evaluation, System implementation, Certification, and Deployment. Each stage contains several MTTs represented by colored boxes (e.g., V-HCD, Means-end analy, HFFiler, BAD MoB, MOVIDA, RTMaps, Empirical analy, Driver Distract, Driver Intentio). To the right of the diagram is a 'List of MTTs' panel with a 'Select all MTT's' button and a list of MTTs with checkboxes: BAD MoB, Driver Distraction Classifier, Driver Intention Recognition, Empirical analysis, HFFiler, Means-end analysis, MOVIDA, RTMaps, and V-HCD.

Figure 13 MTTs connectivity



HoliDes

Holistic Human Factors Design of Adaptive Cooperative Human-Machine Systems

3.10 GUI – Adaptation framework

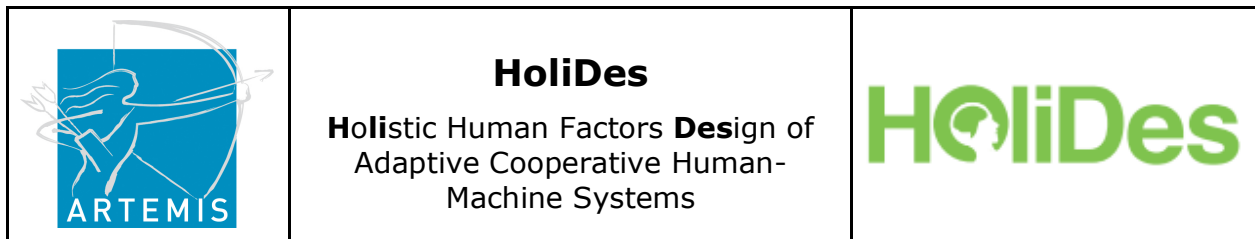
Provides an interface to create your own adCos.

The screenshot shows the "Platform Builder - Adaptation framework" interface. It includes a sidebar with navigation options like "Create an HF-RTP Instance", "My projects", "MTTs database", "MTTs Statistics", "MTTs connectivity", and "Adaptation framework". The main content area explains how to build an AdCoS Requirement kernel using the Adaptation Framework. It defines Cognitive Loop primitives as single simplified entities that allow an agent (human or machine) to interact on an object (a user interface, a task, a task distribution or a process) that are involved in the original process. It provides examples of loops: Patient Positioning, DVA, Workload Balancing, and Lane Change. A table lists the components of these loops:

Loop	Agent	Object	Process
M-LI	display advisor	gantry display	patient positioning
H-P	practitioner	patient positioning	patient positioning

Below the table, there are buttons for "Add Loop", "Remove Loop", and "Reset All". A section titled "Cognitive Loop used in your AdCoS:" shows a diagram of a loop with 'M' (Machine) and 'H' (Human) agents interacting with 'UI' (User Interface) and 'P' (Process) objects. The "AdCoS Requirements..." section lists requirements for MACHINE (display advisor), USER INTERFACE (gantry display on PROCESS patient positioning), HUMAN (practitioner), and PROCESS (patient positioning) components.

Figure 14 Adaptation framework



4 DataBase model

4.1 Entity/Relationship Diagram

The Entity relationship diagram of the Holidés PB (figure below):
Contains the entities and connections between them, including the tables, fields, primary keys and the foreign keys.

The entities used in the PB database model are:

Provider: Data from HOLIDES MTTs providers. Is the owner of MTT. This entity contains all data interesting and relevant from providers.

Regulations: Regulations allow the user filter for HF. It's linked with Domain entity with identifier domain field.

HF Issue: Descriptions that can help to filter and detail more to find an accurate MTT. It's connected with domain entity.

User: User with access to PB. Including in this entity aim data as: password, user identifier and user personal data. All data is saved in the database and it's protected too.

MTT: Contains each tool included in Holidés project. Detailing for each their partner owner or provider.

ToolDetails: Contains for each MTT the details including advantages and disadvantages for each Tool.

Related Activity: This entity contains data as: analysis, design ,evaluation. It's a field describing the activity purpose of the MTT. It's another filter to select the accurate MTT tool.

Domain: is the stage of development in which the problem of HF is framed. Contains: aeronautics, Health, automotive and control rooms domains as a part of each pilot in the HOLIDES project.

RTP: PB results including list of categorized tools. Including information about: domain, HF issues and related activity.



HoliDes

Holistic Human Factors **Design** of Adaptive Cooperative Human-Machine Systems



Project: List of Searches saved in database by HF Users.

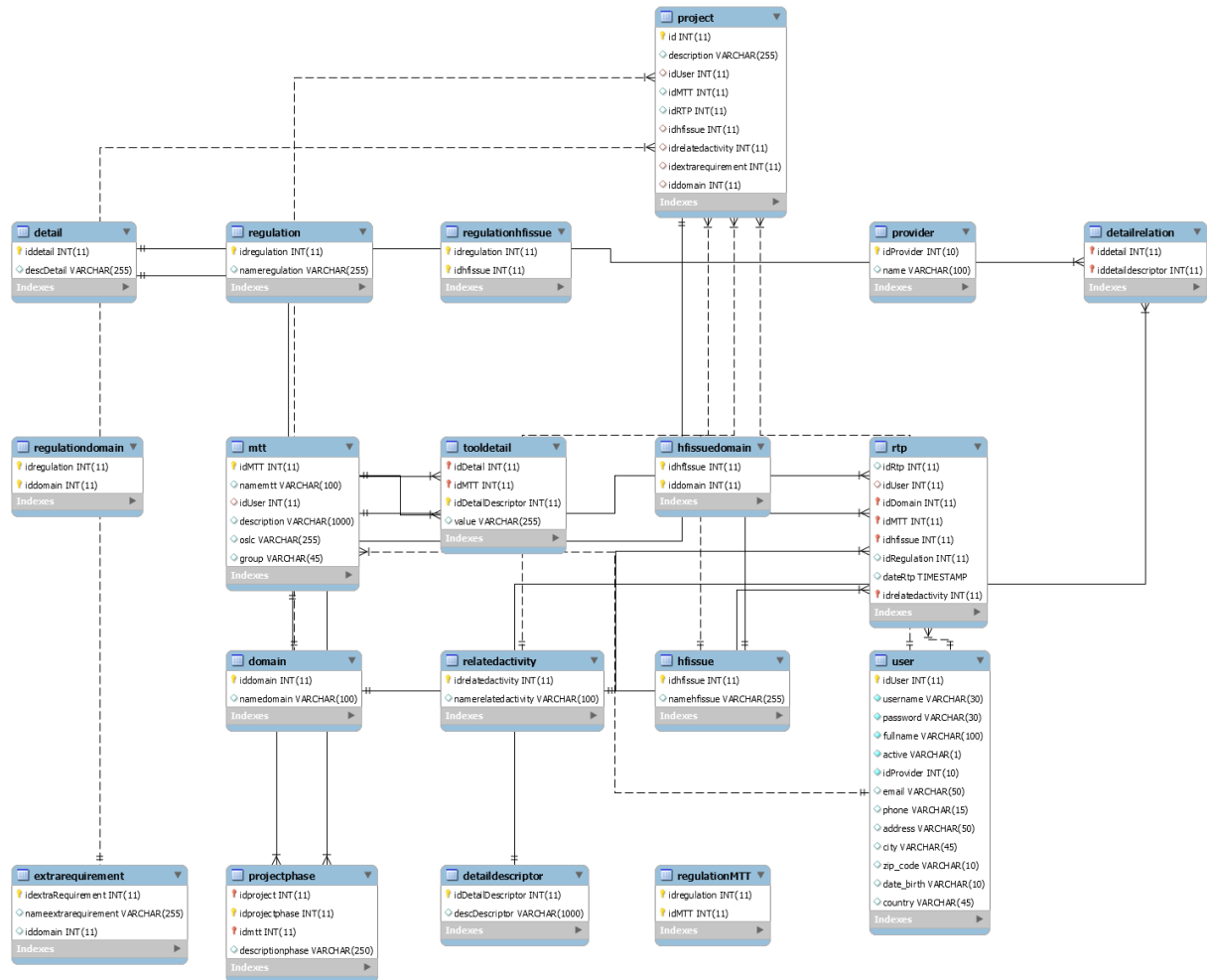


Figure 15 Entity/Relationship model