COMPETITIVENESS AND INNOVATION FRAMEWORK PROGRAMME

Call FP7-ICT-2011-7

Project Acronym:	**Cassandra**
Project Number:	**288429**
Project Type:	**COLLABORATIVE PROJECT: Small or medium scale focused research project (STREP)**
Project Full Title:	**A multivariate platform for assessing the impact of strategic decisions in electrical power systems**

Work programme topic addressed: **ICT-2011.6.2 ICT systems for energy efficiency**

D4.4.1 Cassandra Early Platform

Nature:	**P**
Dissemination Level:	**PU**
Version #:	**1.1**
Delivery Date:	**Initial: 30 April, 2013, Updated version: 1 August, 2013**
Deliverable Leader:	**CERTH**
Author(s):	**Kyriakos C. Chatzidimitriou (CERTH), Antonis Chrysopoulos (CERTH)**
Status:	**Final**
Reviewed on	**22nd July, 2013**
Reviewed by:	**Christos Diou, Andreas Symeonidis (CERTH)**

Abstract

This document contains the manual of the Cassandra early platform prototype. The prototype platform and source code are available online at https://github.com/cassandra-project/platform.

Document History

Version[1]	Issue Date	Stage[2]	Content and changes
#0.1	29th April 2013	Draft	Initial version of the document (CERTH)
#0.5	29th April 2013	Draft	Added training module (CERTH)
#1.0	30th April 2013	Final	Version describing the early platform is ready (CERTH)
#1.1	19th July 2013	Final	List of changes: • Minor updates to all sections • Authentication section • Demographics section • Pricing examples • Training section – major update • New screenshots

List of participants:

Participant no. *	Participant organisation name	Part. short name	Country
1 (Coordinator)	Centre for Research and Technology Hellas/ Informatics and Telematics Institute	CERTH-ITI	Greece
2	Aristotle University of Thessaloniki	AUTH	Greece
3	VaasaETT Global Energy Think Tank	VAASA	Finland
4	Politecnico di Milano	POLIMI	Italy
5	Coventry University Enterprises Ltd	COVUNI	UK
6	Luleå University of Technology	LTU-CDT	Sweden
7	Cleopa GmbH	CLEOPA	Germany
8	Erasmus research Institute of Management	ERASMUS	Netherlands
9	DRAXIS S.A	DRAXIS	Greece

[1] Please use a new number for each new version of the deliverable. Add the date when this version was issued and list the items that have been added or changed. The 'what's new' column will help the reader in identifying the relevant changes. Do not forget to update the version number and date on the front page and the header.
[2] A deliverable can be of these stages: either "draft" or "final". For each stage, several versions of a document can be issued. *Draft:* Work is being done on the contents. *Final*: All chapters have been completed.

Table of contents

Table of Figures

1 Introduction

The aim of the present deliverable is to offer a manual to accompany the early version of the Cassandra platform. The up-to-date electronic version of this document can be found at: https://github.com/cassandra-project/platform/wiki/User-Manual, while the early prototype of the platform is located at: https://github.com/cassandra-project/platform.

The current document is public and is intended to be read, not only by the members of the Consortium, but also by readers that are interested to use the platform in order to model electrical installations and run demand side management scenarios.

The report corresponds to the user manual of the platform (Section 2), with each subsection corresponding to each entity modelled through the platform.

2 *CASSANDRA* Early Platform Manual

This section is the user guide for the current version of the Cassandra platform. The source code of the platform is located at: https://github.com/cassandra-project/platform. A working version of the platform can be found at: https://cassandra.iti.gr:8443/cassandra/app.html.

2.1 Authentication

The platform is user-oriented in the sense that each user has her own workspace consisting of her own projects and a private user library for storing entities for reuse. Before entering the platform the user should be authorized by the system inserting her credentials (username and password). Potential users should contact the system administrator in order to acquire appropriate credentials, since there is no self-registration process.

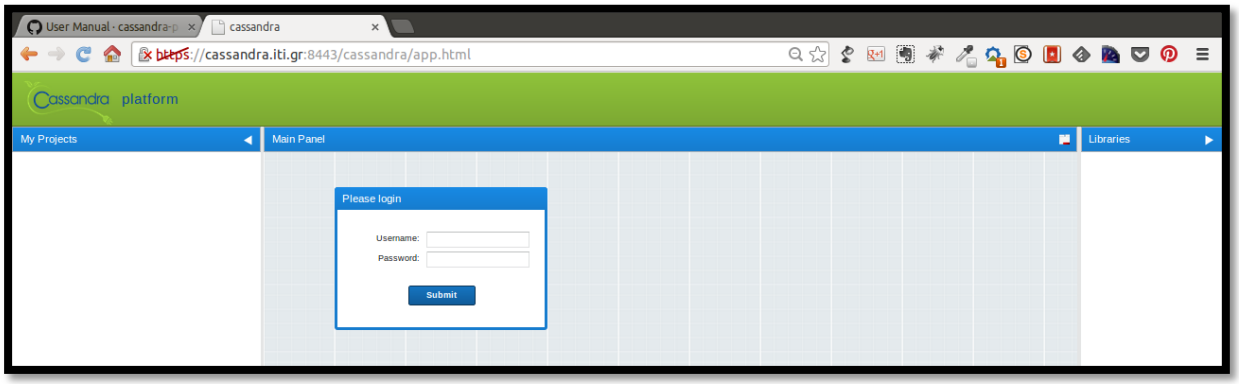

Figure 1: Login screen

2.2 Workspace

The Cassandra platform user interface is separated into three panes:

1. The projects' or workspace pane, at the left of the screen (*My Projects*),
2. the working pane at the middle of the screen (*Main Panel*) and the
3. libraries pane on the right part of the screen (*Libraries*).

The libraries pane includes both the *User library* tree and the *Cassandra library* tree.

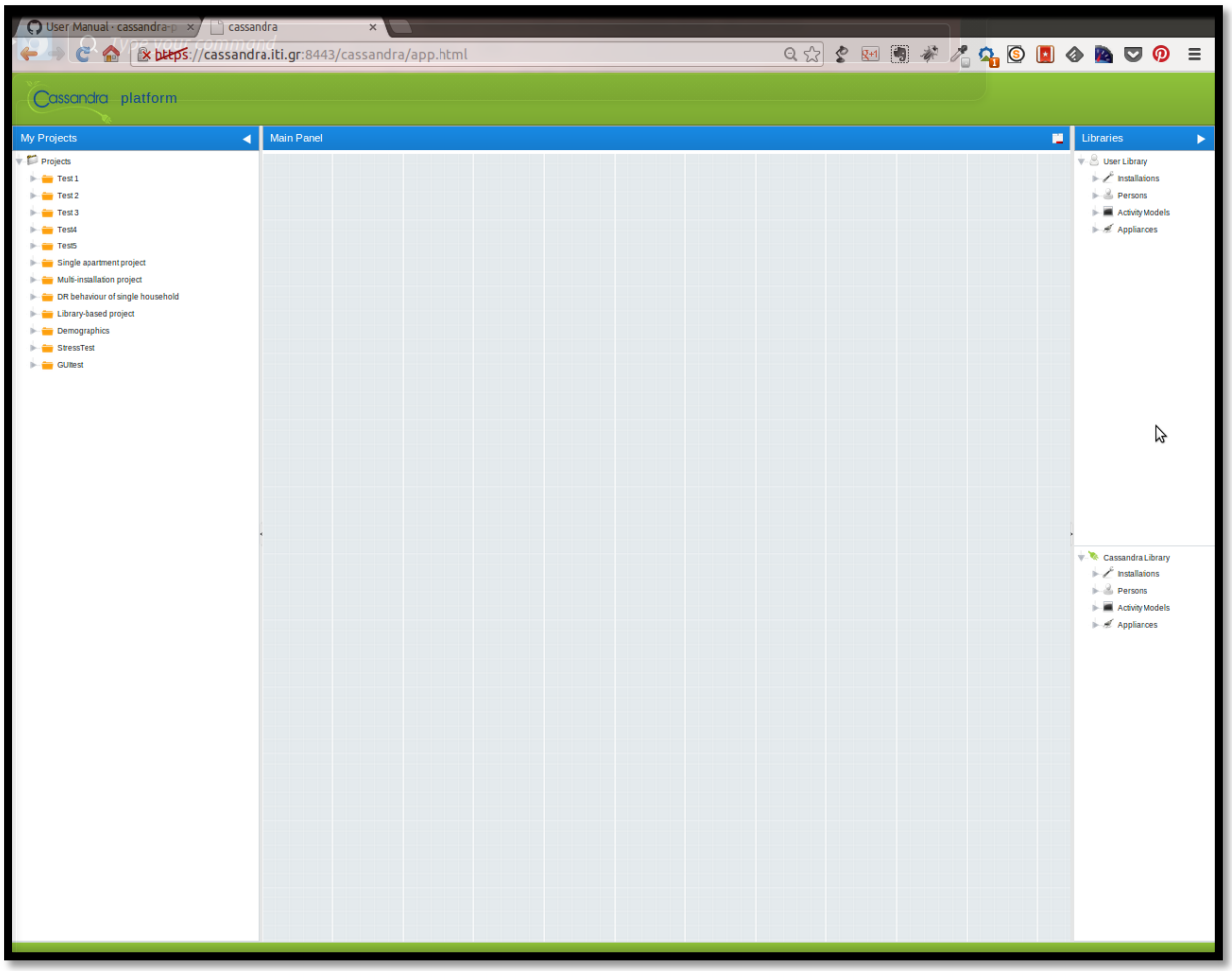

Figure 2: Main screen

The workspace pane holds the projects of the user in a tree structure as follows:

- *Projects* (Level 0)
- *Project* (Level 1)
- *Scenarios* (Level 2)
- *Scenario* (Level 3)
- *Installations* (Level 4)
- *Installation* (Level 5)
- *Persons* (Level 6)
- *Person* (Level 7)
- *Activities* (Level 8)
- *Activity* (Level 9)
- *Activity Models* (Level 10)
- *Activity Model* (Level 11)
- *Appliances* (Level 6)

- *Appliance and Consumption Model* (Level 7)
- *Simulation Parameters* (Grid) (Level 4)
- *Simulation Parameters* (Form) (Level 5)
- *Pricing Schemes* (Level 4)
- *Pricing Scheme* (Level 5)
- *Demographics* (Grid) (Level 4)
- *Demographics* (Form) (Level 5)
- *Runs* (Level 2)
- *Run* (Level 3)

Each tree node represents an entity. The user should click on the links to see instructions about individual entities. Double-clicking a node opens it in a tab on the working pane.

2.3 Libraries

The libraries (User and Cassandra) contain pre-specified entities that can be drag'n'dropped into working scenarios to facilitate scenario building (Figure 3). The supported entities are of type:

- *Installation* (along with its Person and Appliance entities and their children entities),
- *Person* (along with its children entities)
- *Appliance* (along with its Consumption Model)
- *Activity Model* (along with their Distributions)

The *User library* is a read-write library, i.e. a library that the user can store and retrieve entities, while the *Cassandra library* is read-only, meaning that the user can only get entities and not put.

Once a model is placed from a library into a scenario or vice-versa, a copy of the entity is created under the scenario, which can be further customised according to user needs. The user can also store models produced in any scenario from the workspace into the user library for future use. The *Cassandra library* is updated by system administrators only. In general, copies are recursive (deep copies), that is they copy recursively all the entities under the entity being dragged'n'dropped. If the user would like to create a shallow copy, then the Shift key must be kept pressed while dragging and dropping an entity. With respect to deep-copying there is an exception as far as the *Activity Models* are concerned. In general when an entity is dragged'n'dropped that includes Activity Models (i.e. Activity Model, Activity, Person) the appliances listed to participate in the *Activity Model* are not copied and are dropped from the copied Activity Model, since the entity under consideration does not include them. On the other hand if a scenario or an installation is dragged'n'dropped, the Activity Models carries the appliances because they are copied as well.

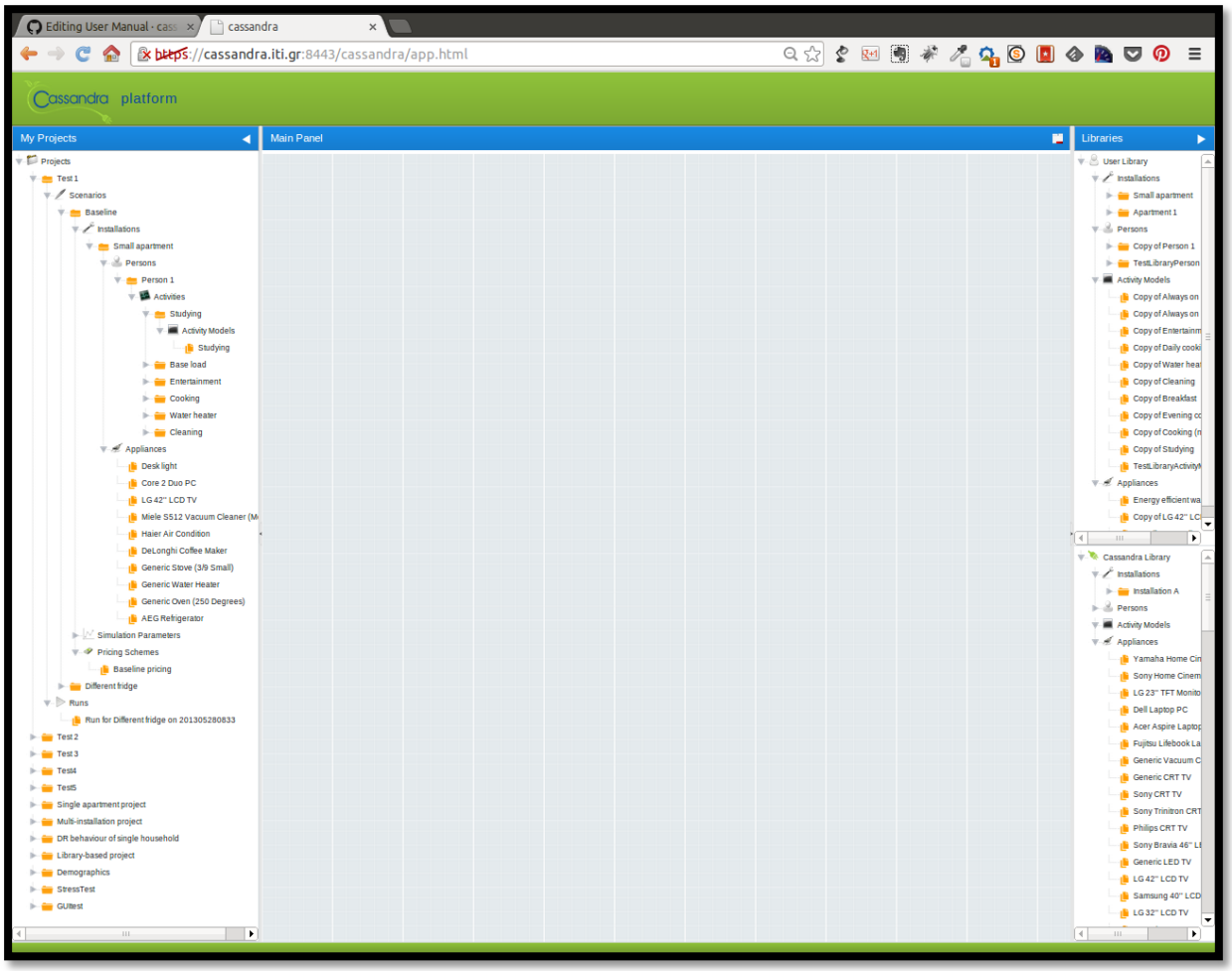

Figure 3: Cassandra and User libraries

2.4 Usage Workflow

The main usage workflow is as follows:

1. The user creates a scenario under a project and setups all the necessary entities and along with their properties: Installations, Persons, Appliances, Activities, Activity Models etc.
2. She specifies the simulation parameters.
3. She runs the scenario.
4. The user double-clicks on the run after it finishes in order to see the results of the simulation. The results open in a new tab on the browser window.
5. The user can run alternative scenarios by changing for example *Appliances*, *Activity Models*, *Pricing Schemes* etc.
6. The user can also compare two runs by selecting two runs (pressing Ctrl while selecting) in the *Runs* grid tab and then by selecting the compare option from the menu.

In future releases of the platform, it will be possible to automatically compute model parameters based on measurements obtained from actual consumer installations. This will allow analysis of consumption, as well as the accurate simulation of a range of 'what if' type scenarios. The Training and Response module will be responsible for this process. An activity model of the process can be found in Figure 4.

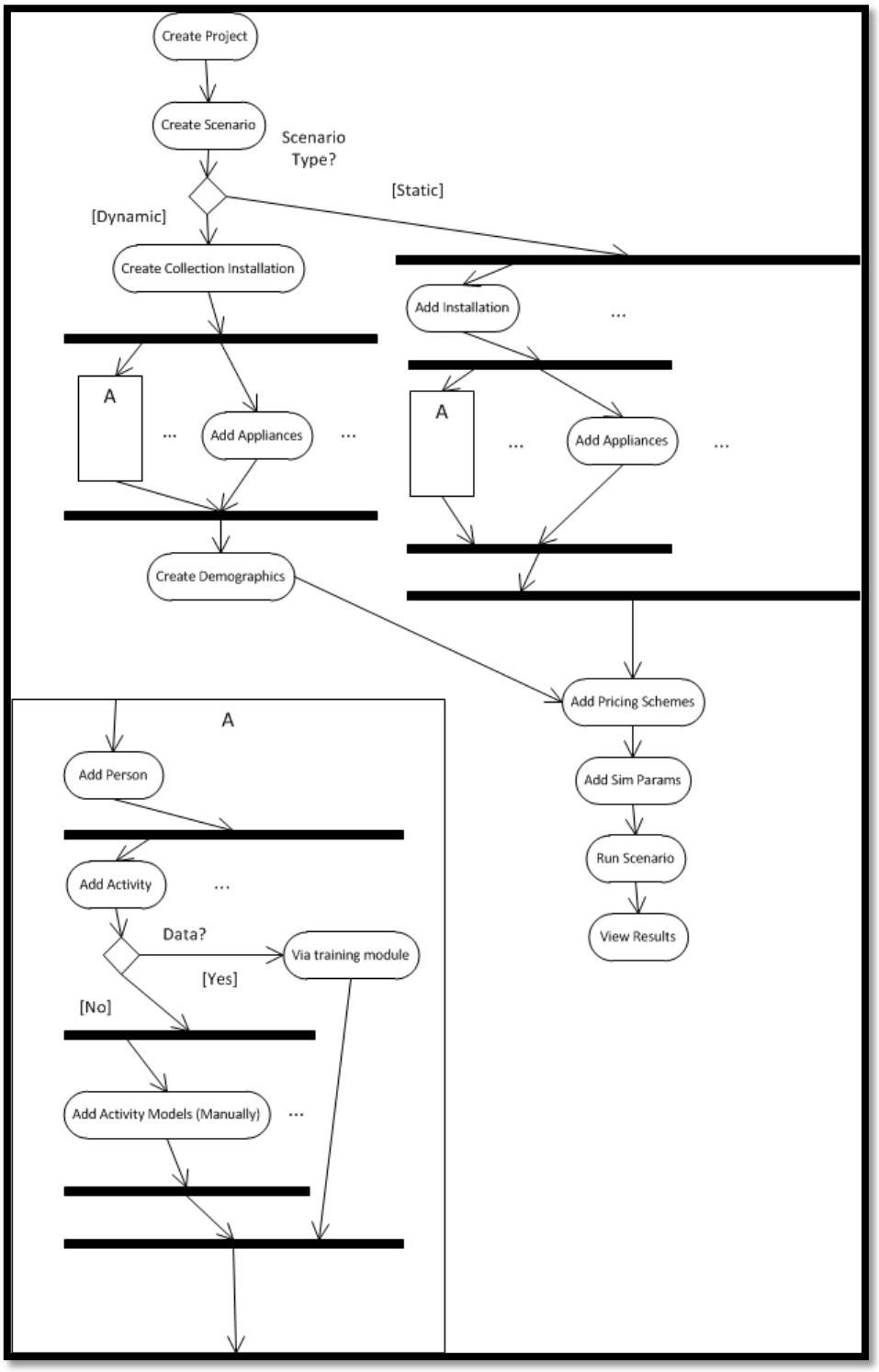

Figure 4 Activity diagram from creating a project to viewing the simulation results

2.5 Projects

The workspace consists of *Projects*. Every project contains scenarios relevant to the project and runs (simulation executions) of those scenarios.

By double-clicking the *Projects* node, the user can see in a grid the projects of the workspace (Figure 5). The available commands are to create a new one and edit or delete an existing one, by first selecting the corresponding row and then pressing either edit or delete. The user can also edit a project by double clicking the corresponding row. A similar grid exists for all other entities in CASSANDRA platform.

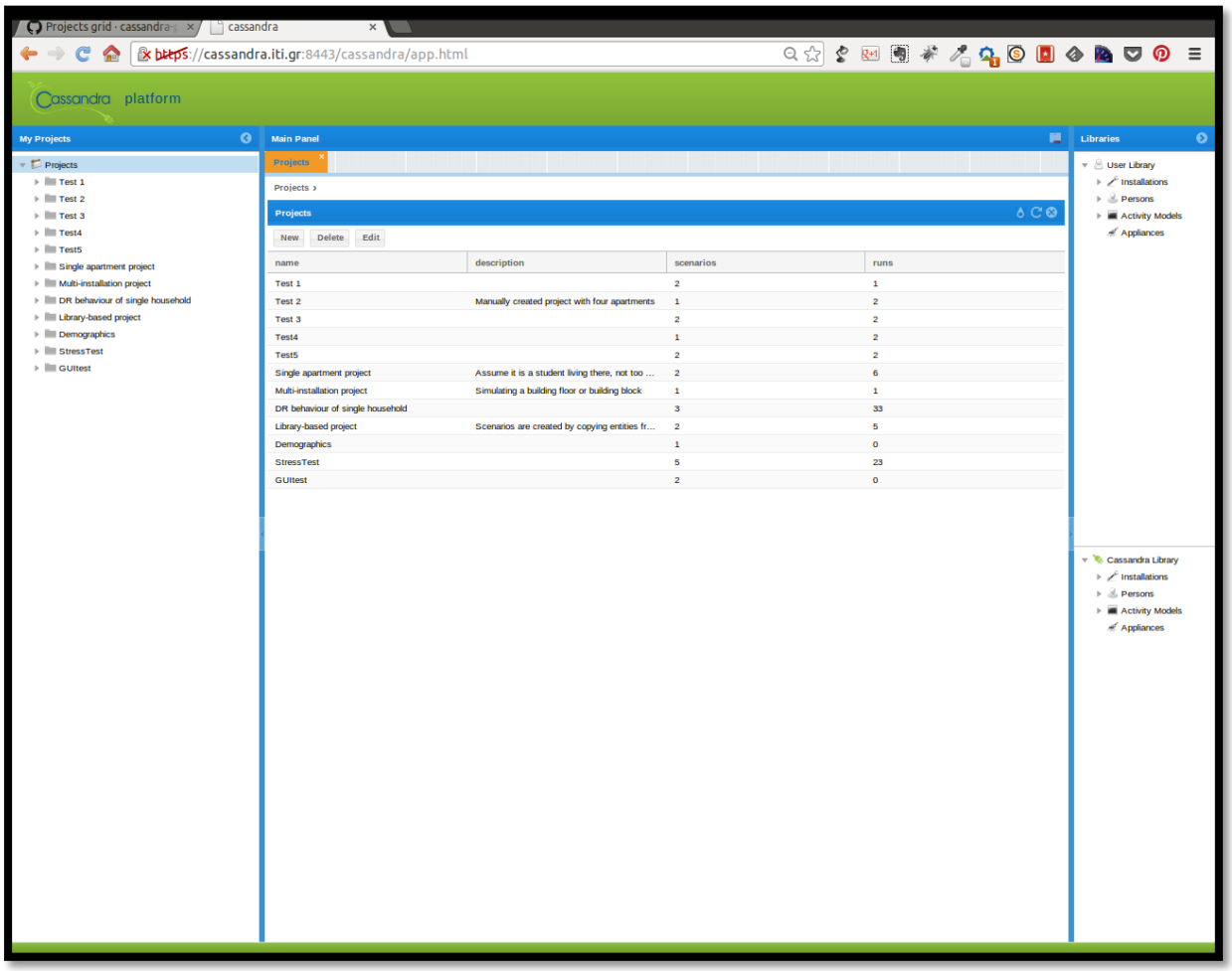

Figure 5: Projects grid

2.6 Scenarios

Scenarios enclose information about electrical loads, consumers and their behavior, as well as simulation parameters. A project can have multiple scenarios, corresponding to different consumer setup, simulation parameters etc.

By double-clicking the *Scenarios* node, the user can see in a grid the scenarios of the project (Figure 6). The available commands are to create a new one and edit or delete an existing one, by first selecting the corresponding row and then pressing either *edit* or *delete*.

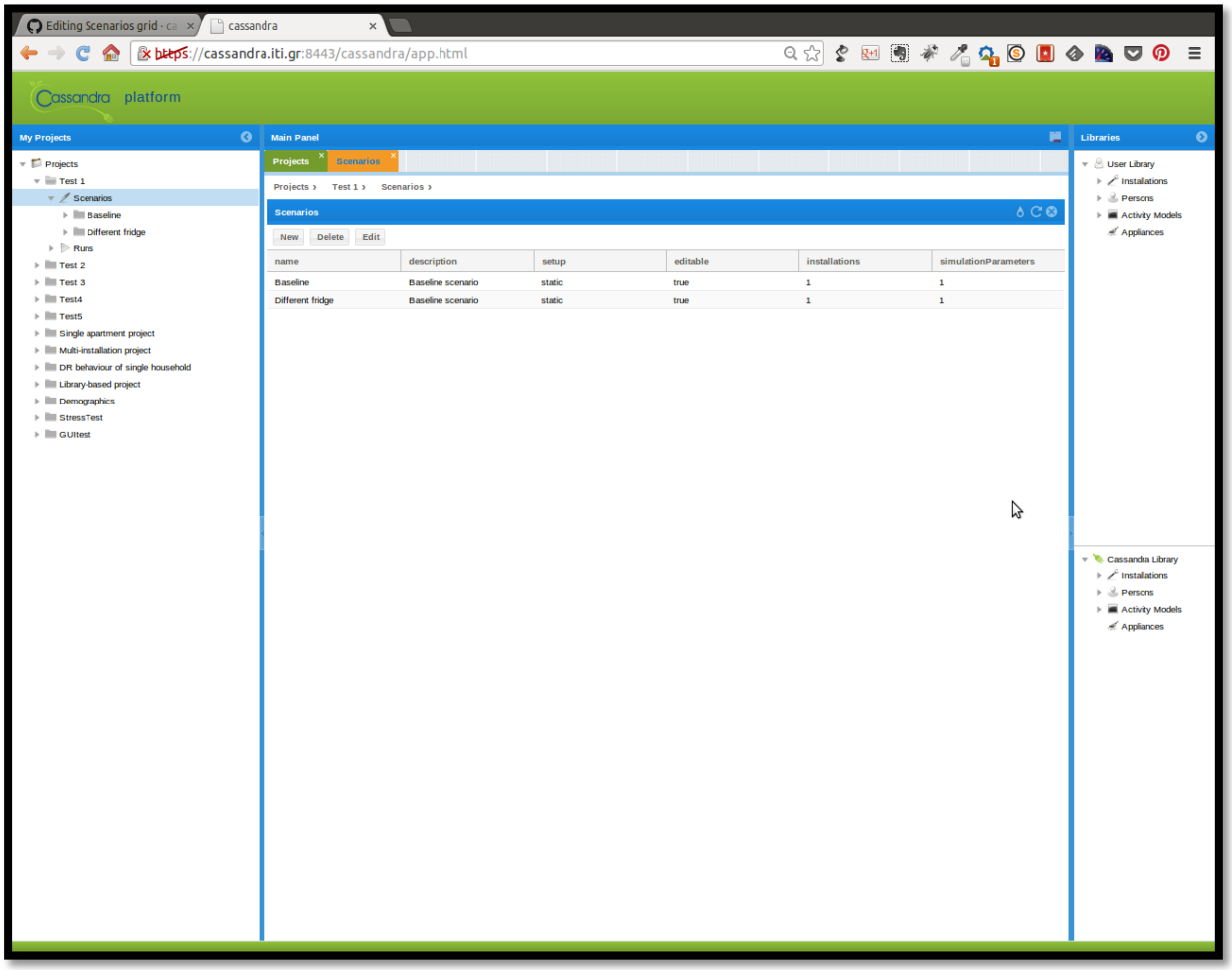

Figure 6: Scenarios grid

2.7 Installations

By double-clicking the *Installations* node, the user can see a grid with the installations of the scenario (Figure 7). Examples of Installations include households, buildings, individual rooms and generally, loads that lie beneath a metering point. *Installations* are containers of *Appliances* that have associated *Consumption Models* and *Persons* that have associated *Activity Models* for *Appliance* usage.

The available commands are to create a new installation and edit or delete an existing one, by first selecting the corresponding row and then pressing either *edit* or *delete*.

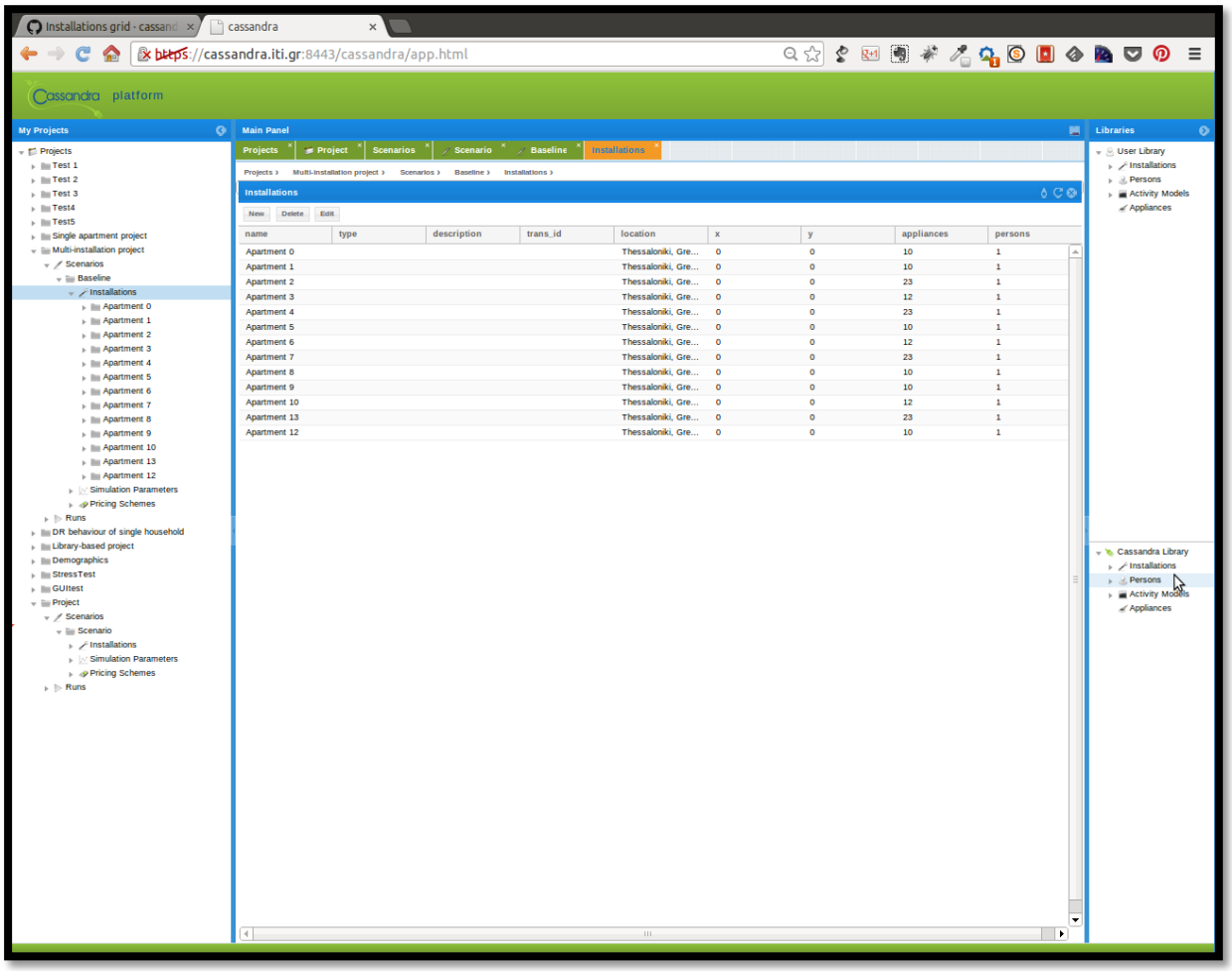

Figure 7: Installations grid

2.8 Persons

By double-clicking the *Persons* node, the user can see a grid with the persons of an installation (Figure 8). The available commands are to create a new one and edit or delete an existing one, by first selecting the corresponding row and then pressing either *edit* or *delete*.

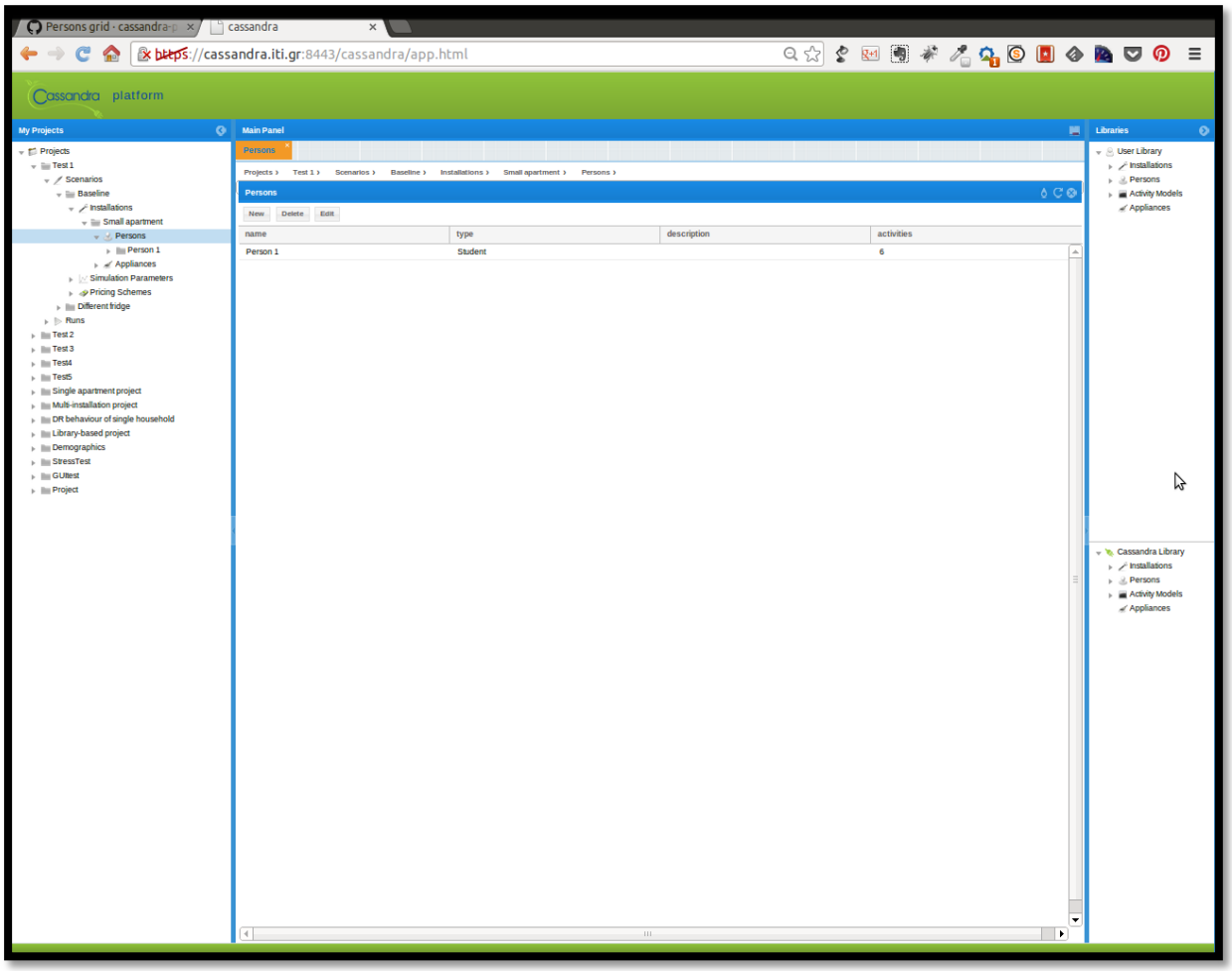

Figure 8: Persons grid

2.9 Activities

A person can have one or more associated *Activities*. These describe how a person uses the *Appliances* in an *Installation*. By double-clicking the *Activities* node, the user can see a grid with the activities of a person (Figure 9). The available commands are to create a new one and edit or delete an existing one, by first selecting the corresponding row and then pressing either *edit* or *delete*.

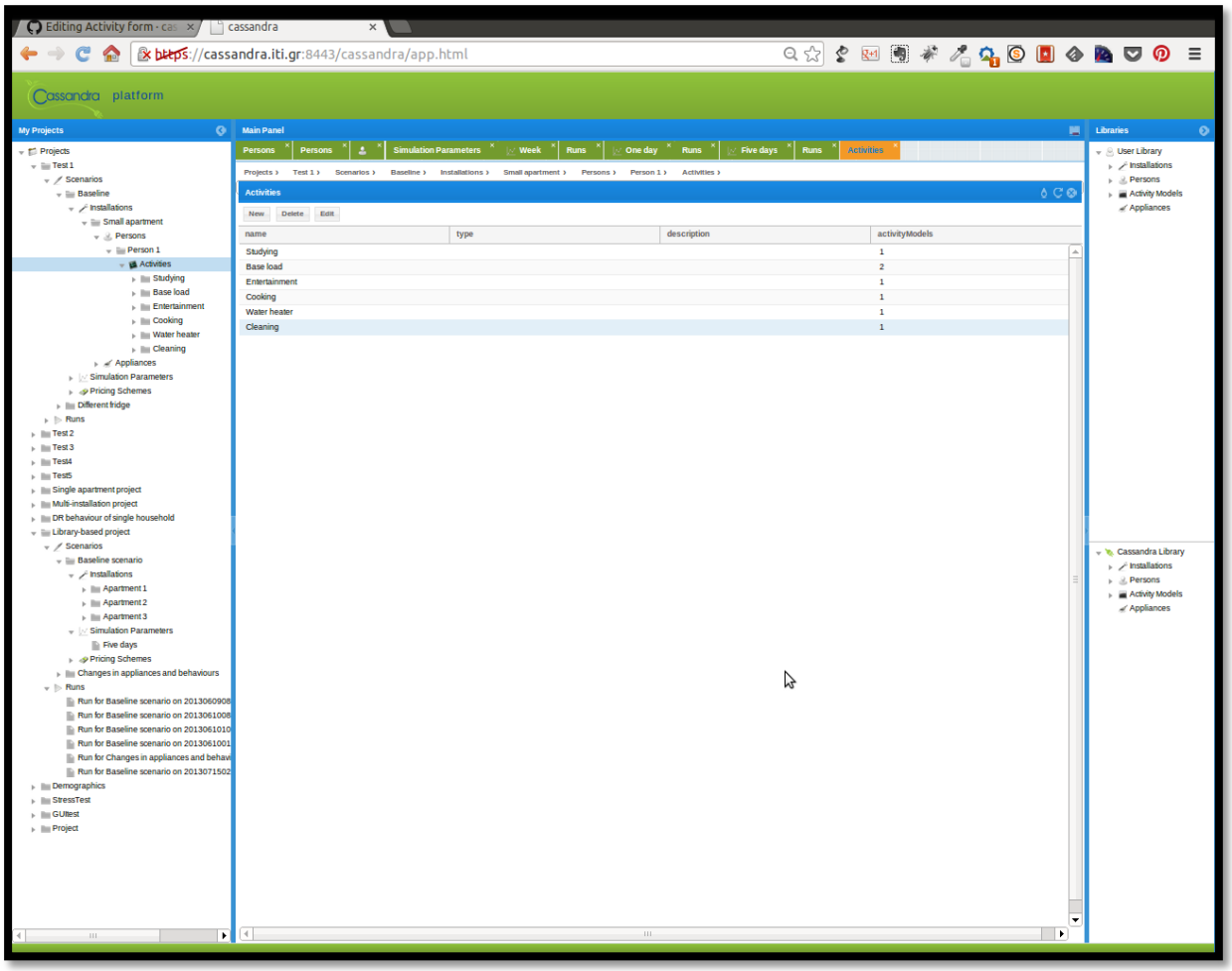

Figure 9: Activities grid

2.10 Activity Models

Activities consist of *Activity Models* that include probability distributions for the start time and duration of use of *Appliances*.

By double-clicking the *Activity Models* node, the user can see a grid with the activity models of a given activity (Figure 10). The available commands are to create a new one and edit or delete an existing one, by first selecting the corresponding line and then pressing either *edit* or *delete*.

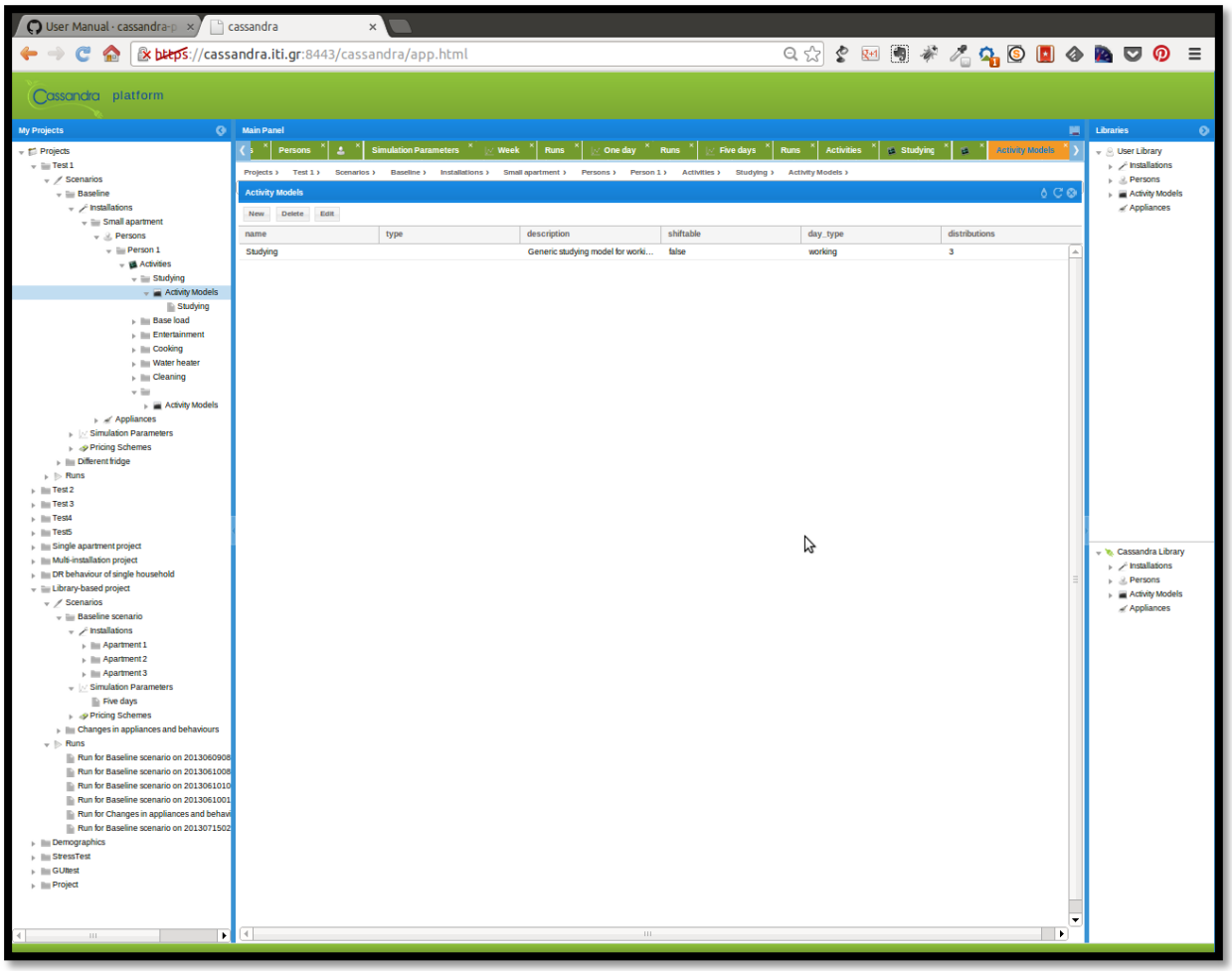

Figure 10: Activity Models grid

2.11 Appliances

By double-clicking the *Appliances* node, the user can see a grid with the appliances of an installation (Figure 11). The available commands are to create a new one and edit or delete an existing one, by first selecting the corresponding row and then pressing either *edit* or *delete*.

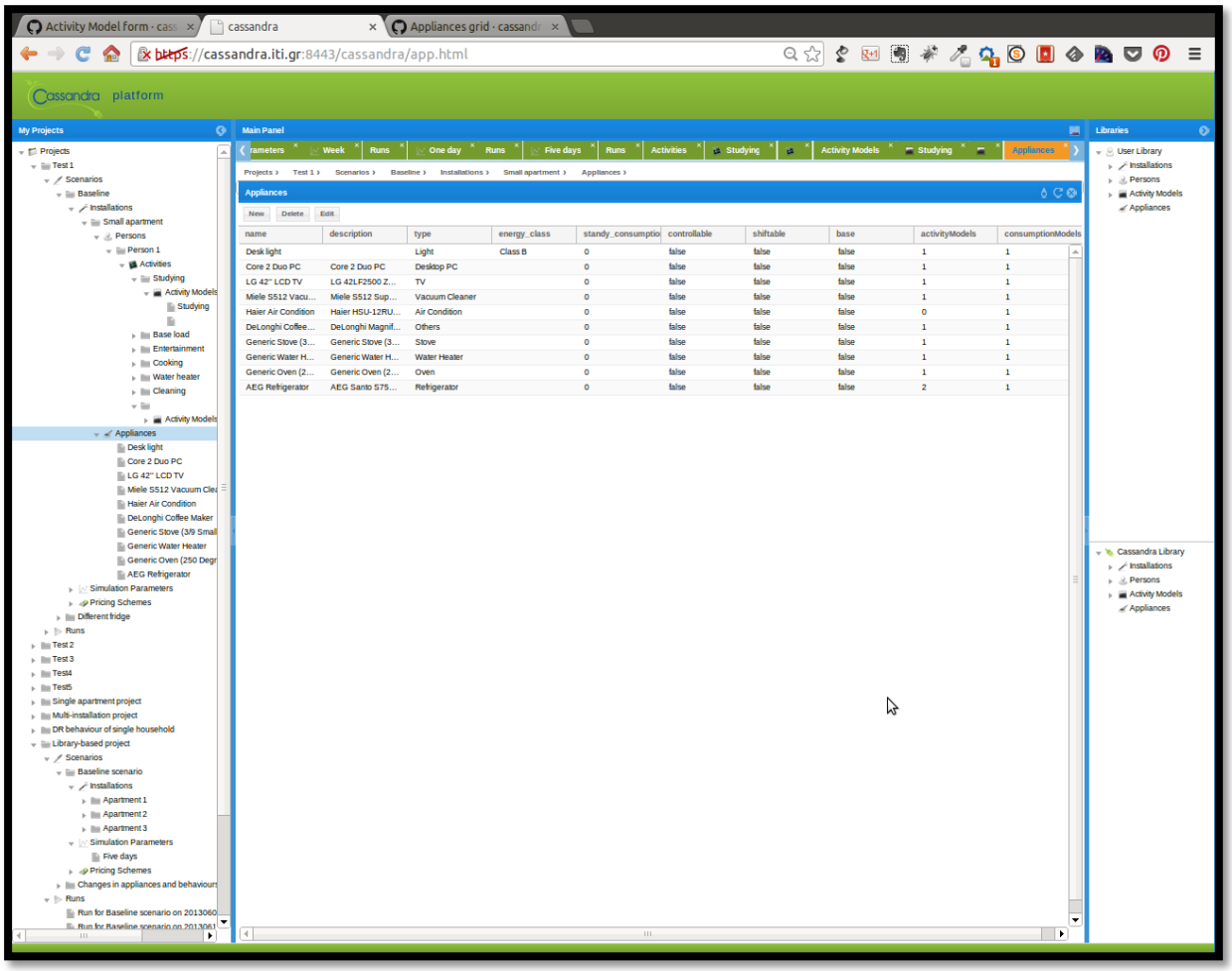

Figure 11: Appliances grid

2.12 Simulation Parameters (Grid)

By double-clicking the *Simulation Parameters* node, the user can see a grid with the simulation parameters corresponding to a scenario (Figure 12). The available commands are to create a new one and edit or delete an existing one, by first selecting the corresponding row and then pressing either *edit* or *delete*.

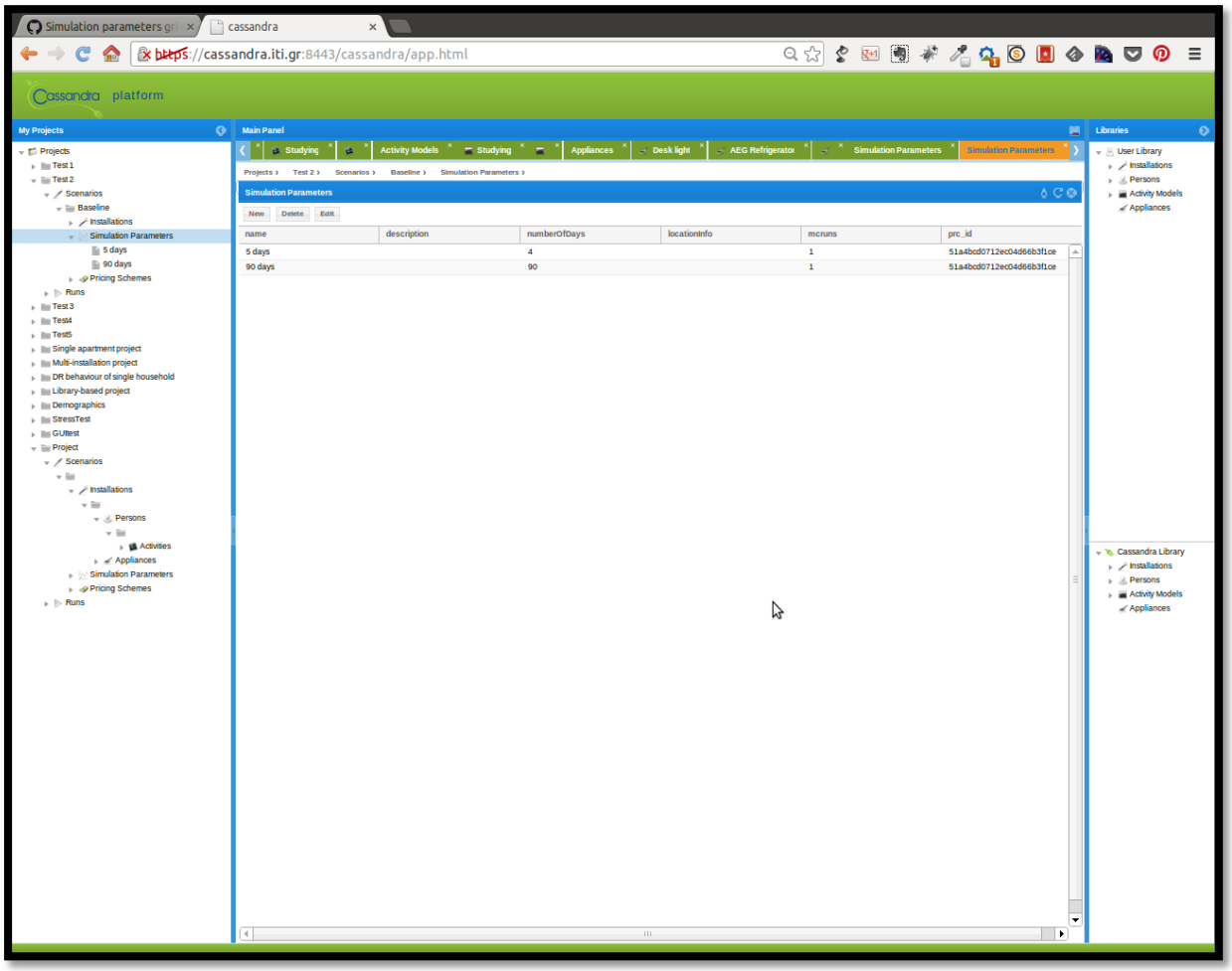

Figure 12: Simulation parameters grid

2.13 Pricing Schemes

By double-clicking the Pricing Schemes node, the user can see a grid with the pricing schemes of a scenario (Figure 13). The available commands are to create a new one and edit or delete an existing one, by first selecting the corresponding row and then pressing either *edit* or *delete*.

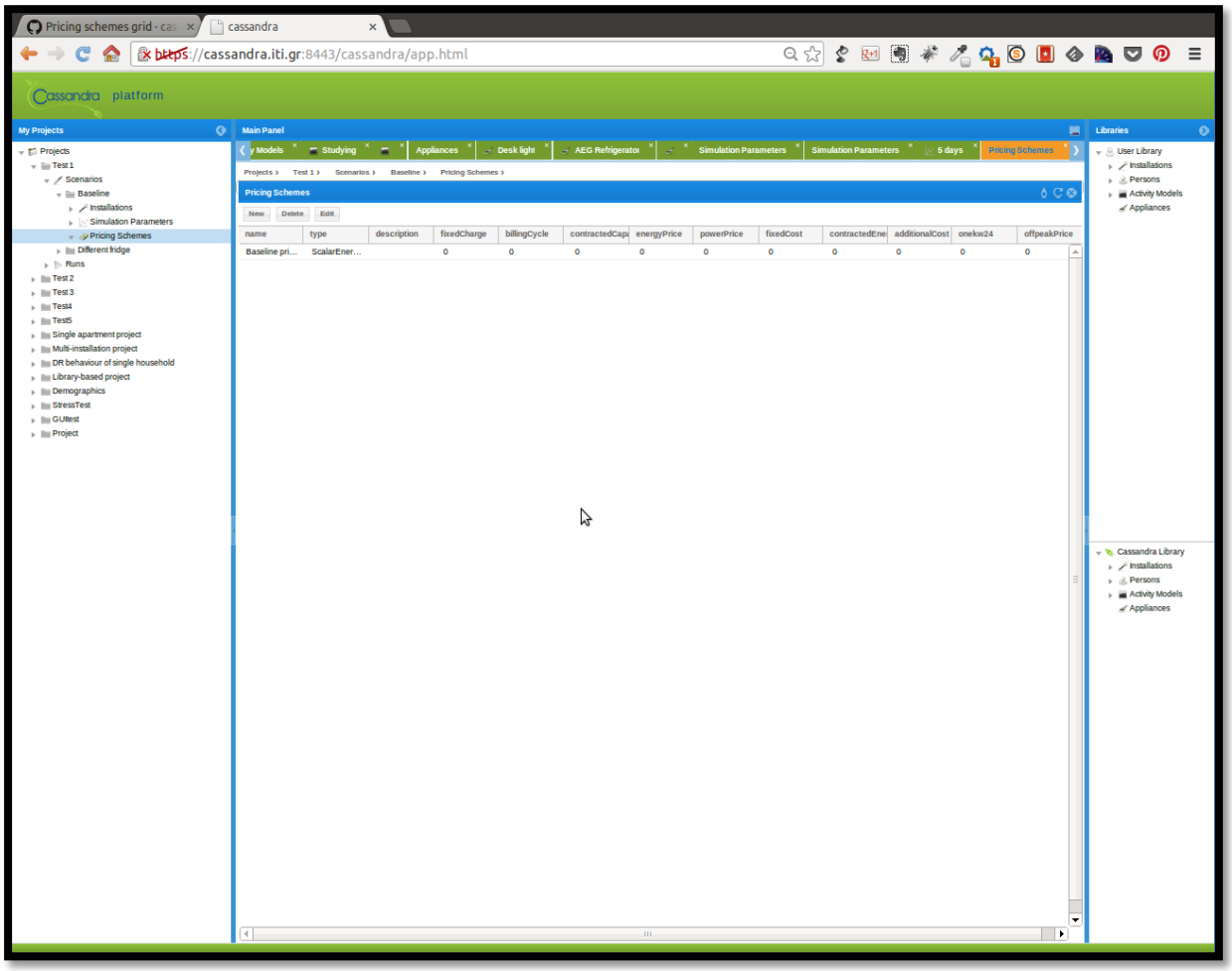

Figure 13: Pricing schemes grid

2.14 Demographics (Grid)

The *Demographics* (node) is included when the scenario is chosen to be a dynamic one. In this case the user does not build the installations one by one, but rather builds a *Collection* installation with a lot of appliances and person types, adding statistics on the distribution of appliances and person types among the generated installations. This way it is easy to construct a variety of installations in the order to 1K, 10K or more installations in an automated manner.

The process is as follows (Figure 14):

1. Declare the scenario as *dynamic*. A *Demographics* node will appear.
2. Create one installation with the name *Collection*.
3. Add in the *Collection* as many persons (with activities and activity models) and appliances (with consumption models) you wish.
4. Create a demographics parameter set.
5. Set the number of entities (Installations for now) you wish to create.

6. Drag and drop persons and appliances from the *Collection* installation to the grid Entities, also setting their probability of participating. The probabilities for person types should sum up to 1, while probabilities for each appliance should be between 0 and 1.

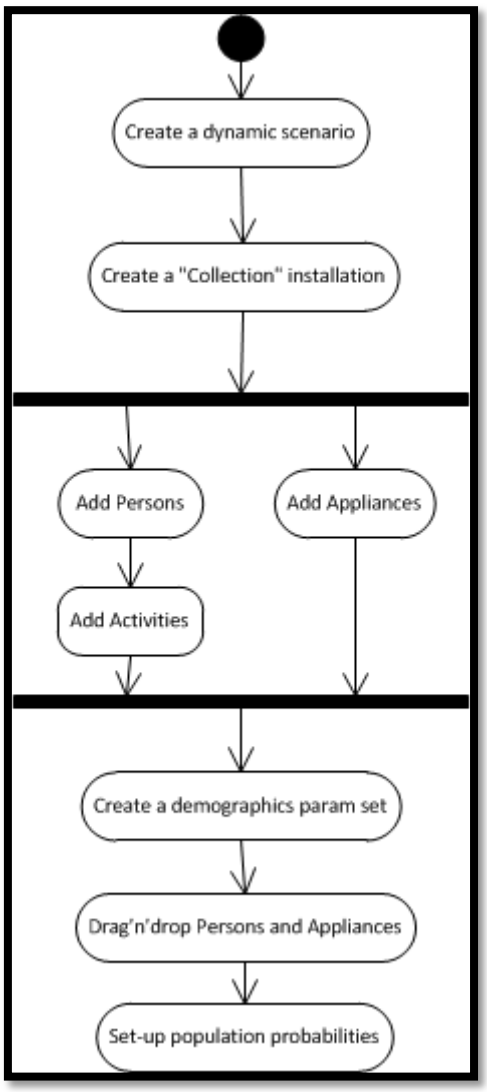

Figure 14 Activity diagram for creating a dynamic scenario

By double-clicking on the *Demographics* node, the user can see a grid with the demographics parameter sets of a dynamic scenario (Figure 15). The available commands are to create a new one and edit or delete an existing one, by first selecting the corresponding row and then pressing either *edit* or *delete*.

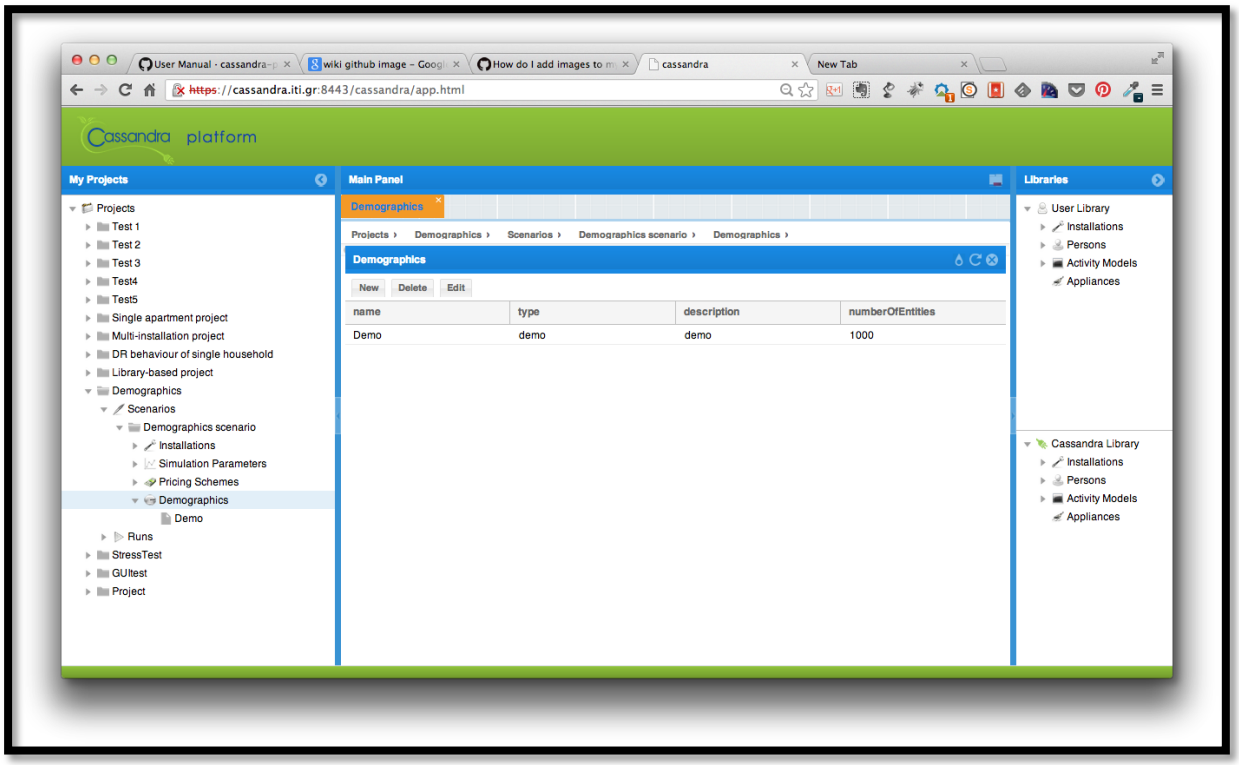

Figure 15: Demographics grid

2.15 Runs

By double-clicking on the *Runs* node, the user gets a grid with the runs of scenarios of the corresponding project (Figure 16). The user can delete a *Run*, watch the progress by pressing the *Refresh* button of incomplete *Runs*, or compare runs by selecting multiple *Runs* (Ctrl+click) and then pressing the *Compare* button (Figure 17).

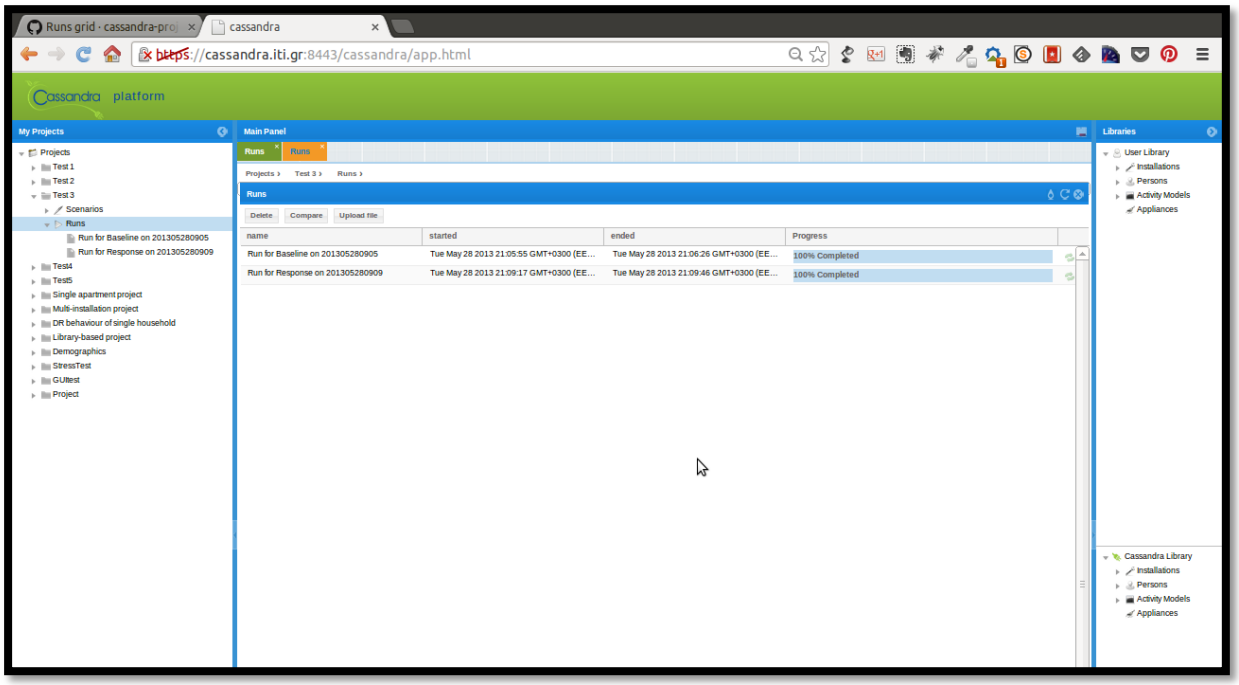

Figure 16: Runs grid

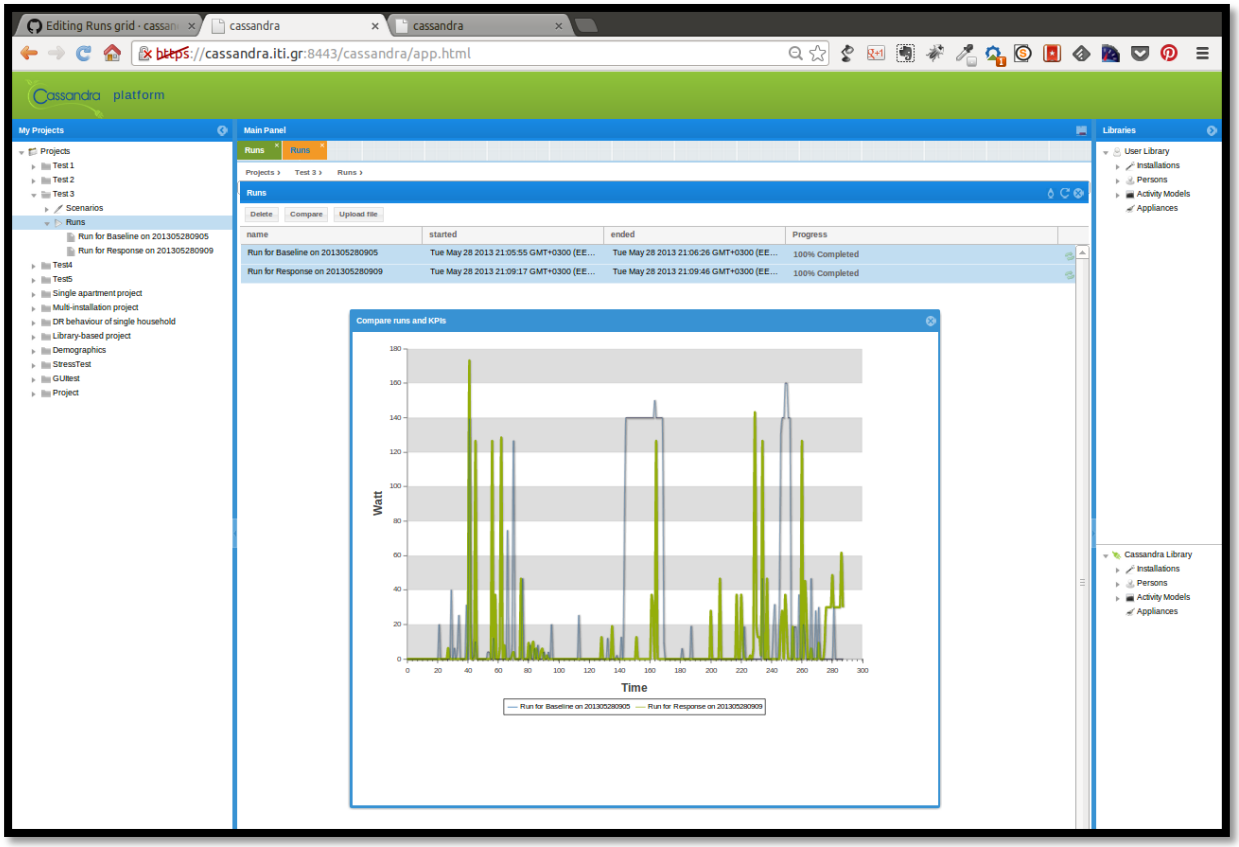

Figure 17: Comparison functionality

2.16 Project

When creating/editing a project (Figure 18), the user enters the following data:

- Name: the name of the project
- Notes: a short description of the project

Upon creating/editing a project the user is informed that at least one scenario should be created under that specific project. The information/warning messages guide the user into creating a complete simulation scenario.

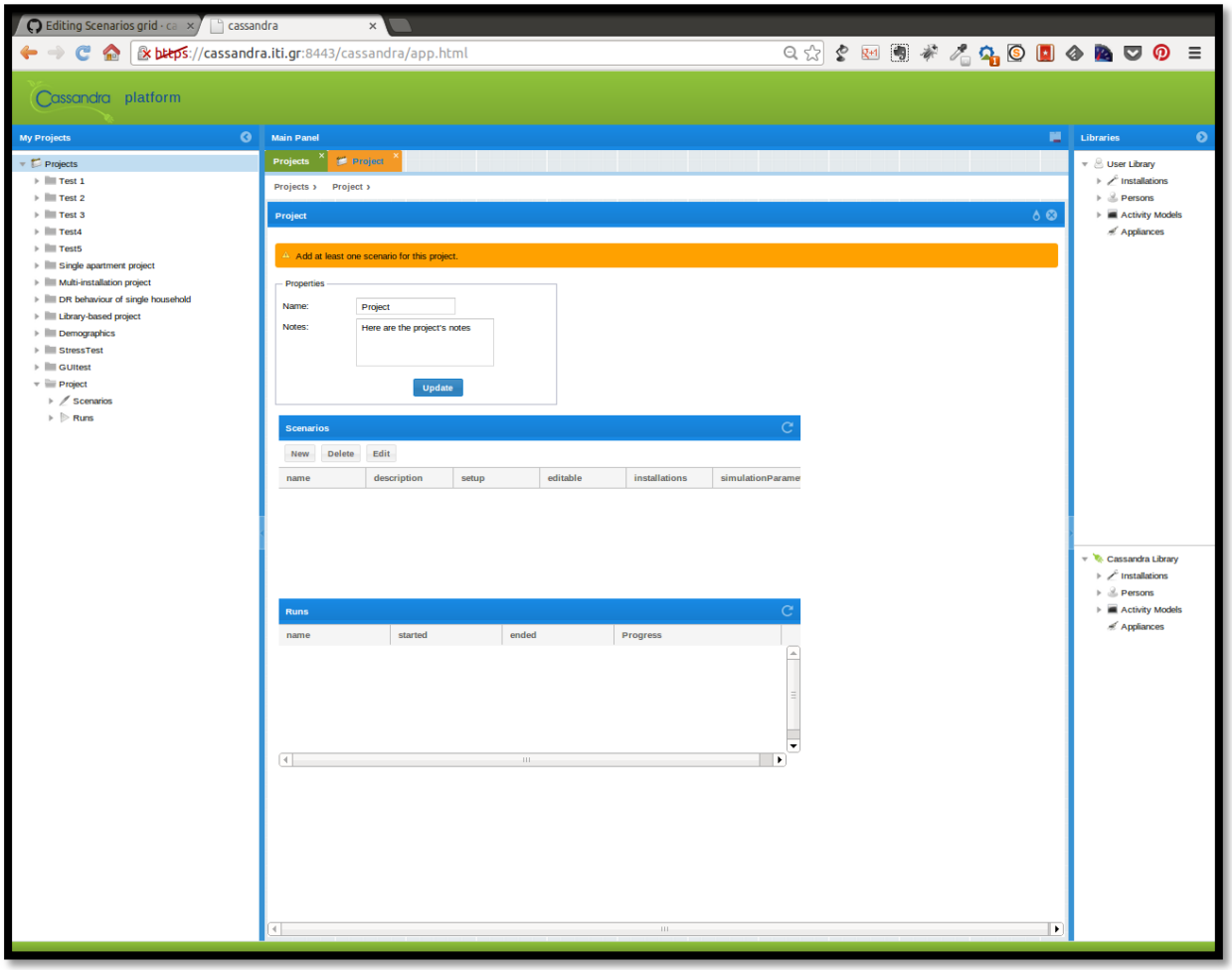

Figure 18: Project form

2.17 Scenario

When creating/editing a scenario (Figure 19), the user enters the following data:

- Name: the name of the scenario
- Notes: a short description of the scenario
- Setup: a selection between *Static* and *Dynamic* scenario setups. In the *Static* case the user builds the scenario step-by-step (probably the best choice for small scenarios, or for larger scenarios with available measurements), while in the *Dynamic* case the user enters demographic data and the platform populates installations based on those demographics.

Each scenario should have at least one complete installation and the necessary simulation parameters set so that it can be simulated.

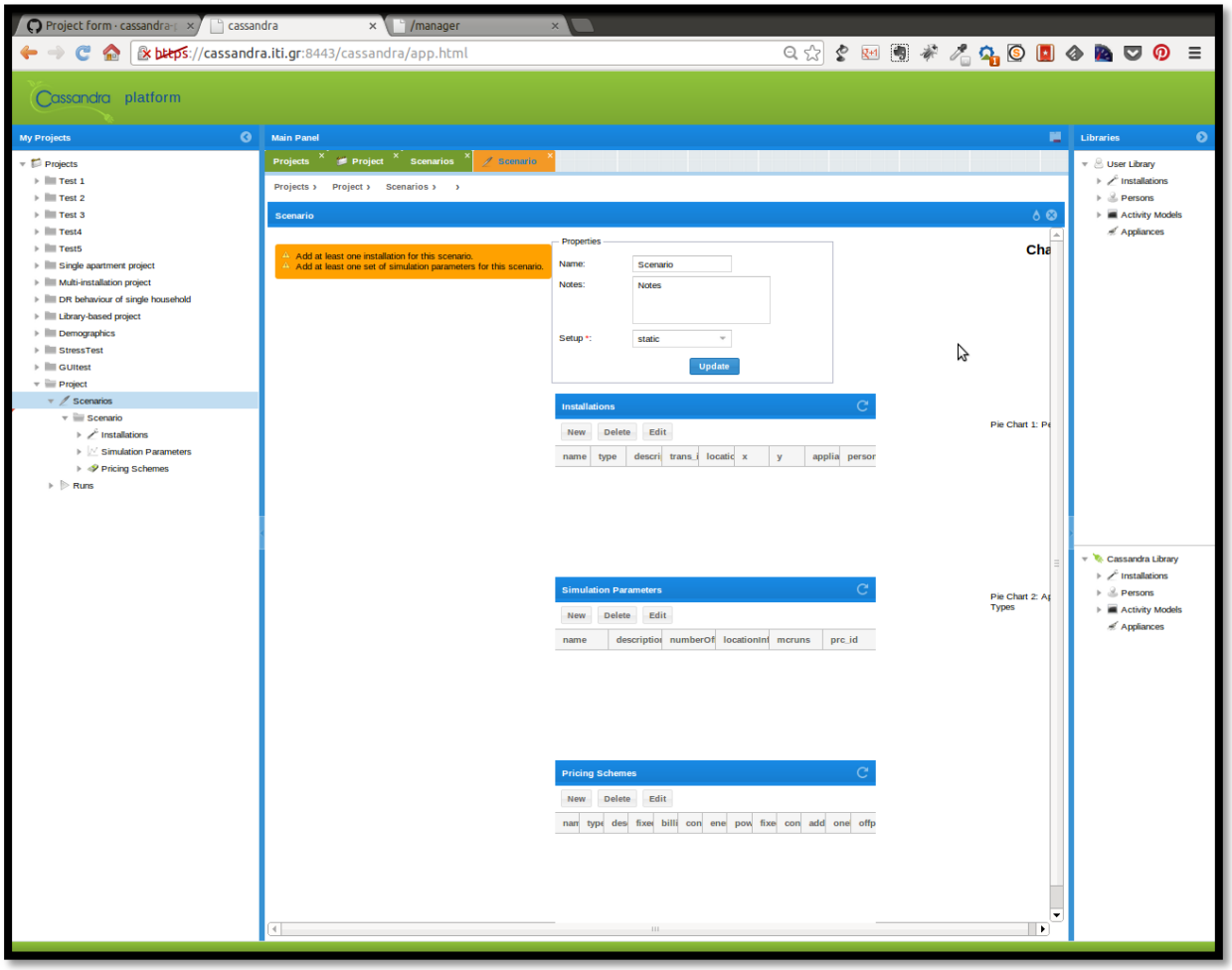

Figure 19: Scenario form

In a complete scenario, the user may observe statistics on person types and appliances comprising the scenario (pie chart diagrams - Figure 20).

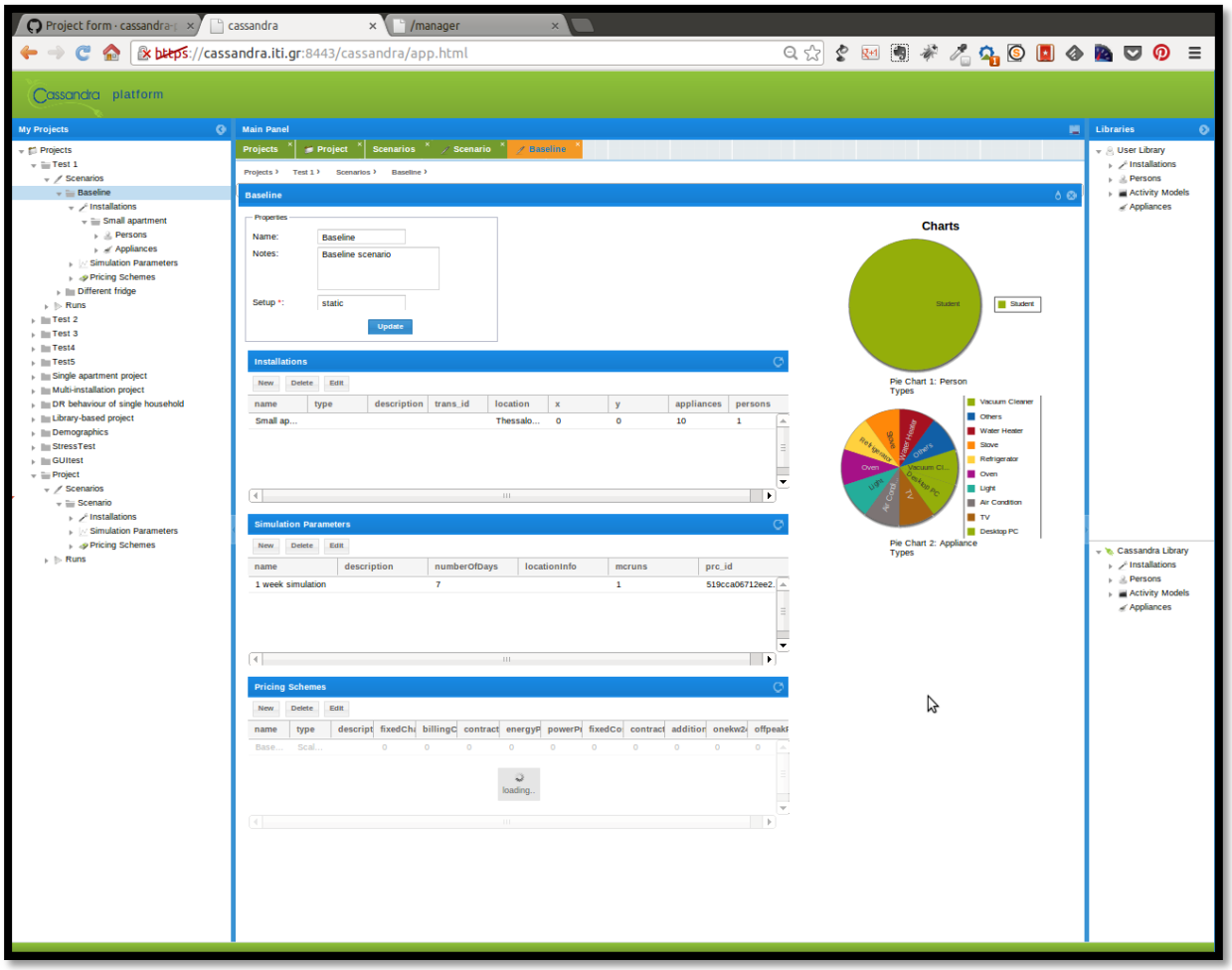

Figure 20: Complete scenario

2.18 Installation

When creating/editing an installation (Figure 21, Figure 22), the user enters the following data:

- Name: the name of the installation
- Type: the type of the installation
- Description: a description of the installation
- Transformer ID: a label indicating the ID of the transformer under which the installation is located.
- Location: the location of the installation
- Lat: the latitude of the installation's geolocation
- Long: the longitude of the installation's geolocation

Each installation should be consisted of at least one complete person and appliance.

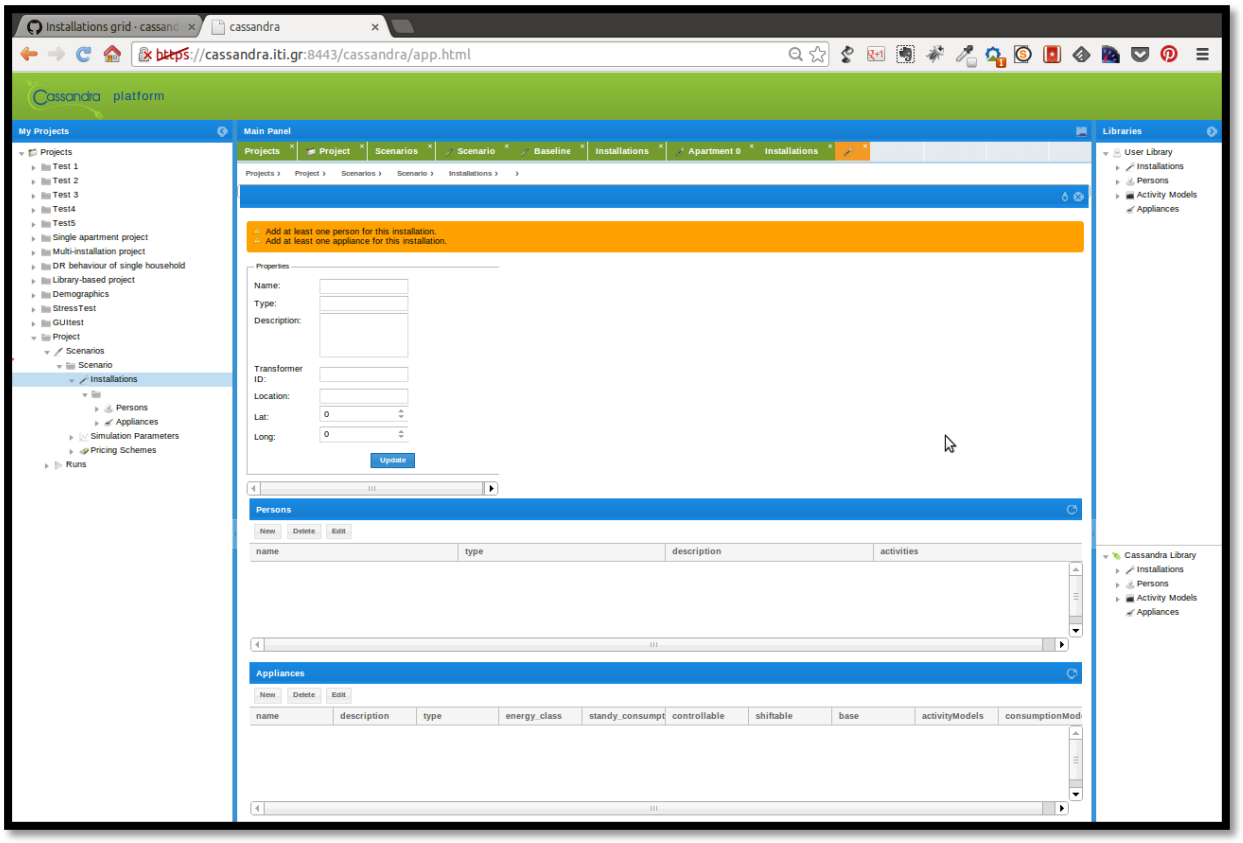

Figure 21: Installation form

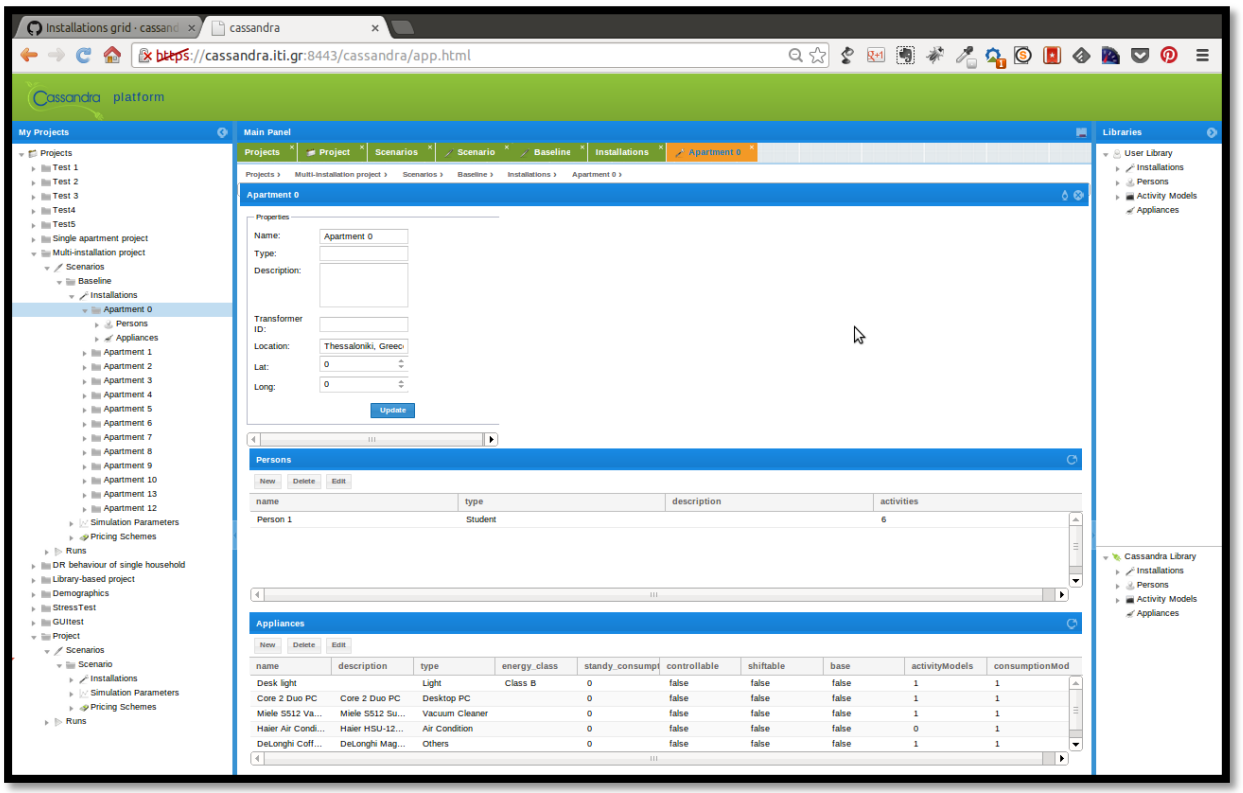

Figure 22: Completed installation form

2.19 Person

When creating/editing a person (Figure 23, Figure 24), the user enters the following data:

- Name: the name of the person
- Type: the type of the person
- Description: a short description of the person

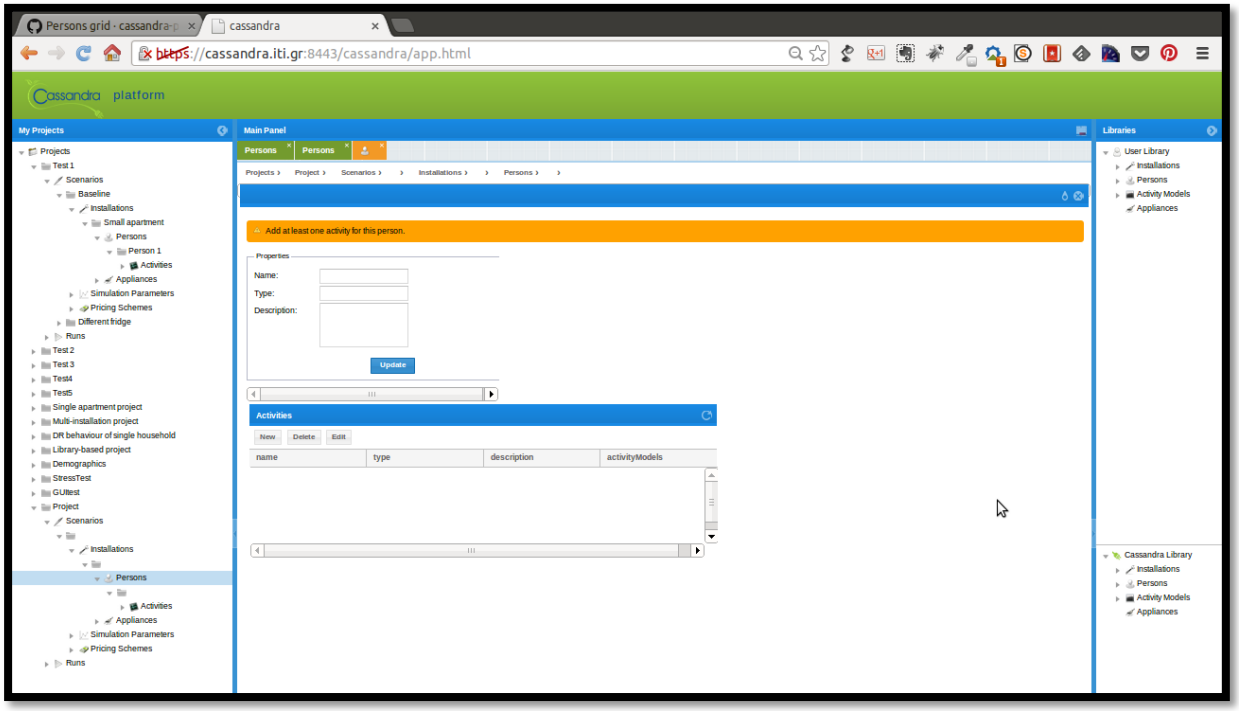

Figure 23: Person form

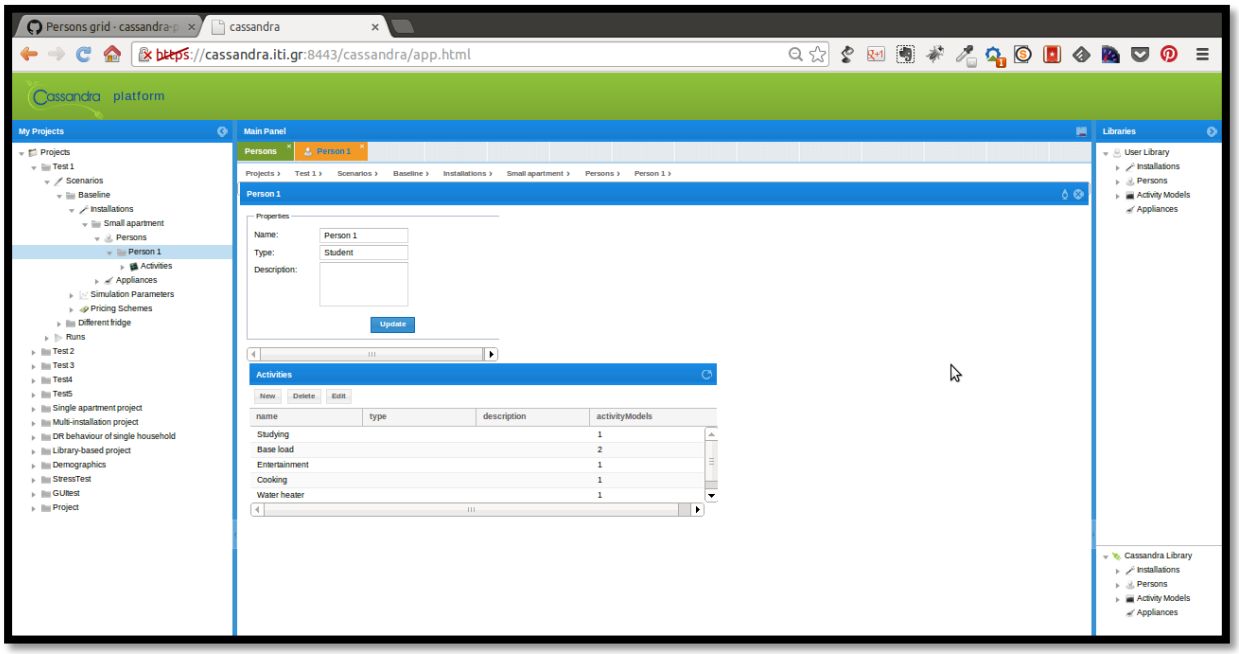

Figure 24: Completed person form

2.20 Activity

When creating/editing an activity (Figure 25, Figure 26), the user enters the following data:

- Name: the name of the activity
- Type: the type of the activity
- Description: a short description of the activity

Each activity should have at least one activity model defining the probabilistic behavior of the person using the appliances participating in the activity.

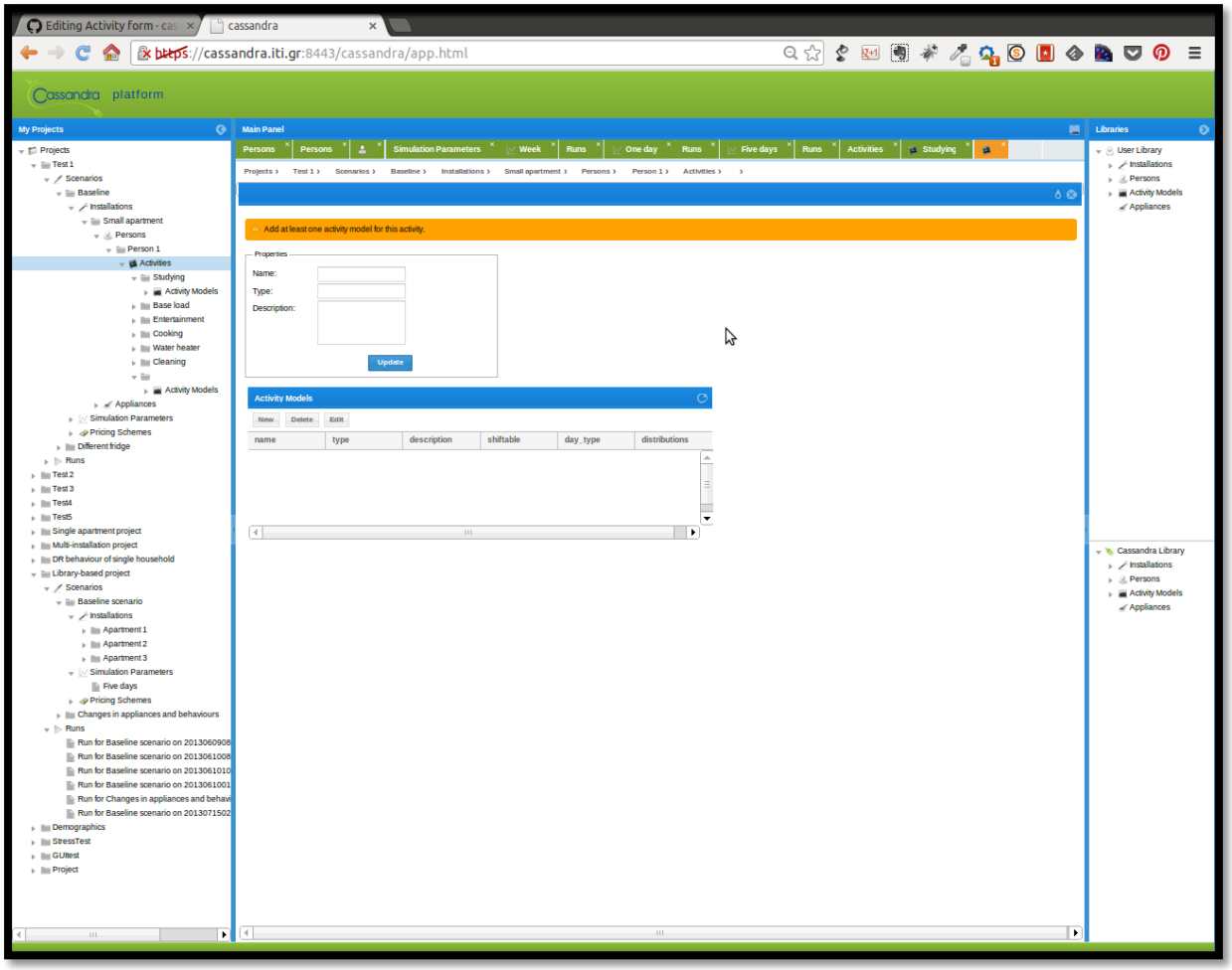

Figure 25: Activity form

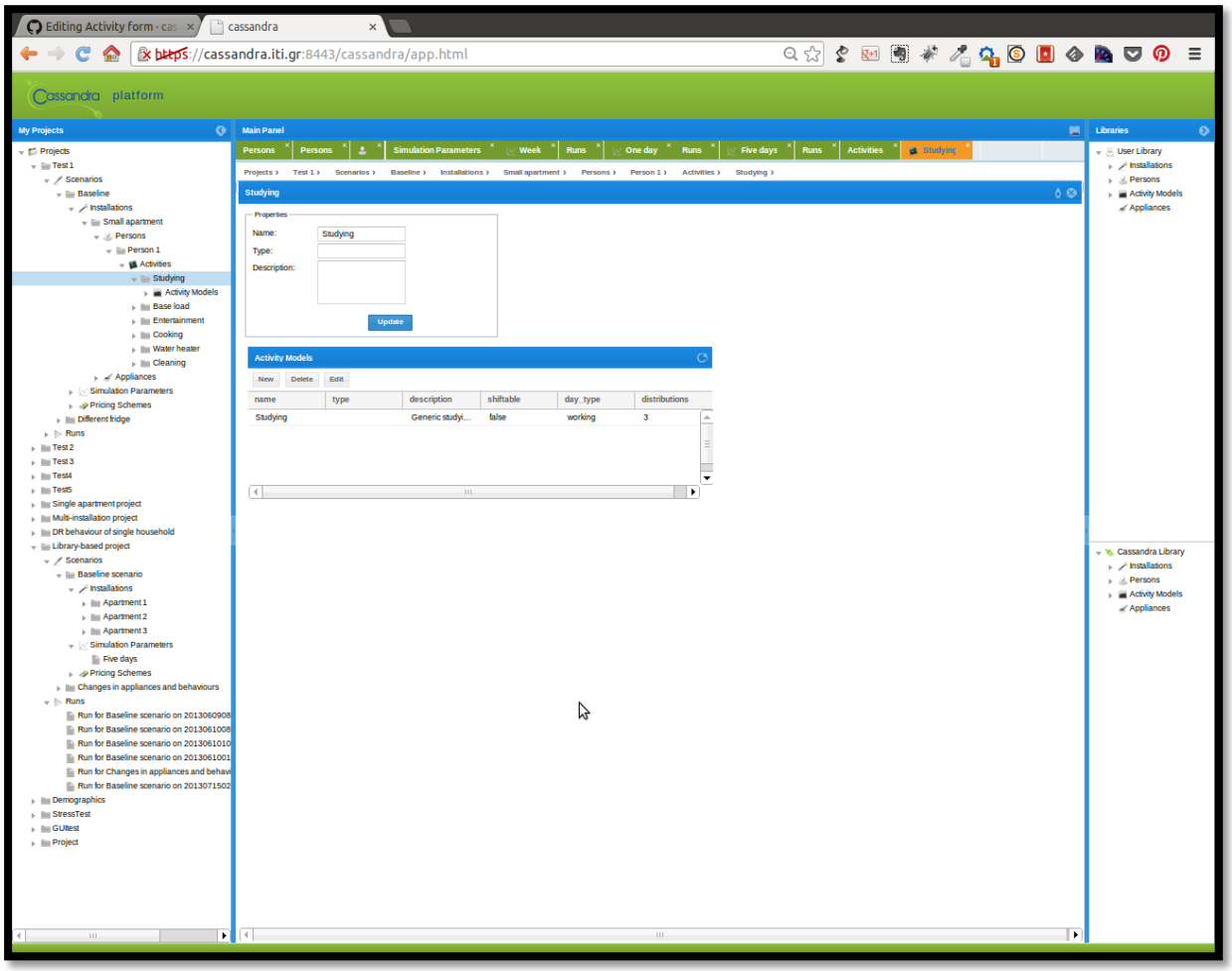

Figure 26: Completed activity form

2.21 Activity Model

When creating/editing an activity model (Figure 27, Figure 28), the user enters the following data:

- Name: the name of the activity model
- Type: the type of the activity model
- Description: a short description of the activity model
- Day type: the day type the activity model corresponds to:
  - any day
  - weekday or weekend
  - abbreviations of specific weekdays, i.e. [Mon, Tue, Sat]
  - specific days formated as 1/12, 31/10 etc.
- Shiftable: whether the activity model can be considered shiftable or not

For every activity model the user can drag-n-drop appliances that participate in it.

To complete the activity model the user must provide the properties and parameters of three distributions that specify an activity model:

- *Duration*,
- *Start-time*, and
- *Number of times per day*,

All the above are probabilistic models of how long the activity takes, at what time in day it starts and how many times per day is executed.

For the first two characteristics the user can select between several types of distributions:

- Uniform Distribution, in the form of an expression with start and end values: [{"start":100,"end":200}]
- Histogram, in the form of values: [1,2,3,4...]
- Normal Distribution, in the form of an expression with mean and std values: [{"mean":45,"std":10}]
- Gaussian Mixture Models, in the form of tuples of w, mean and std: [{"w":1 , "mean":45,"std":10}, {"w":1 , "mean":100,"std":10}]

Under normal system operation, these distributions will be automatically computed via measurements obtained from the actual installation, or by altering an *Activity Model* found in the *Libraries*.

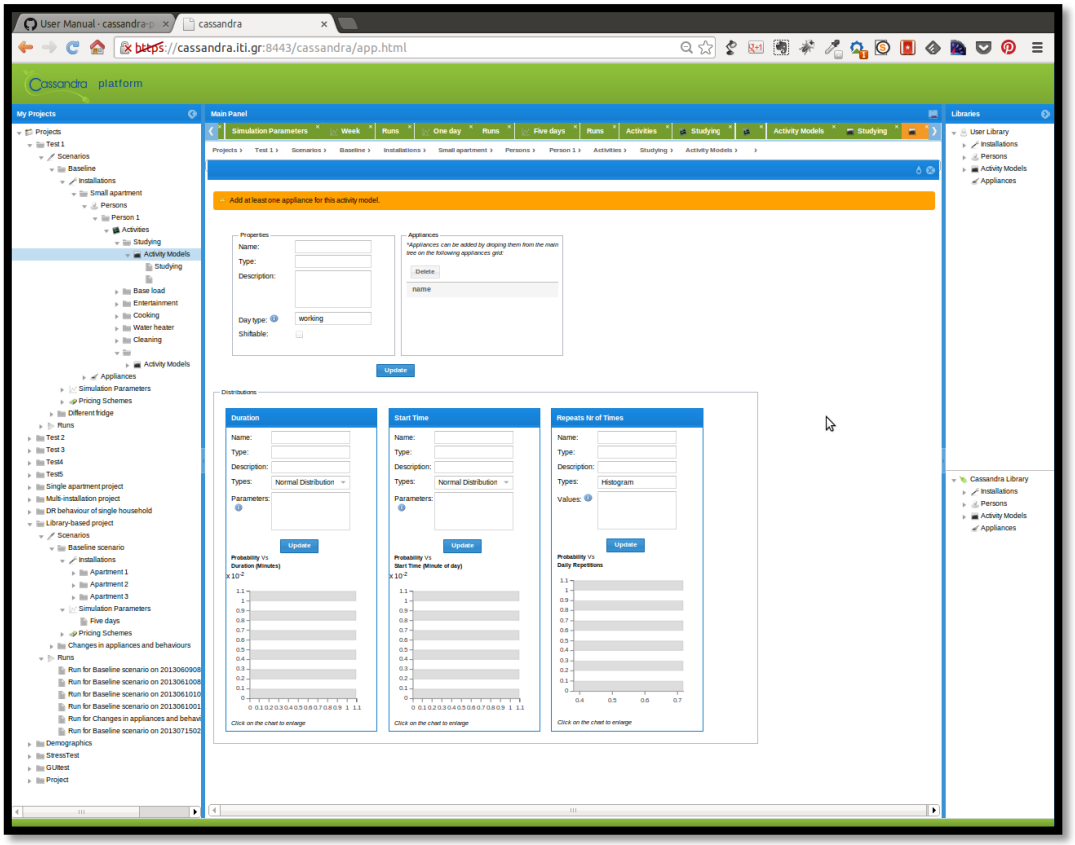

Figure 27: Activity Model form

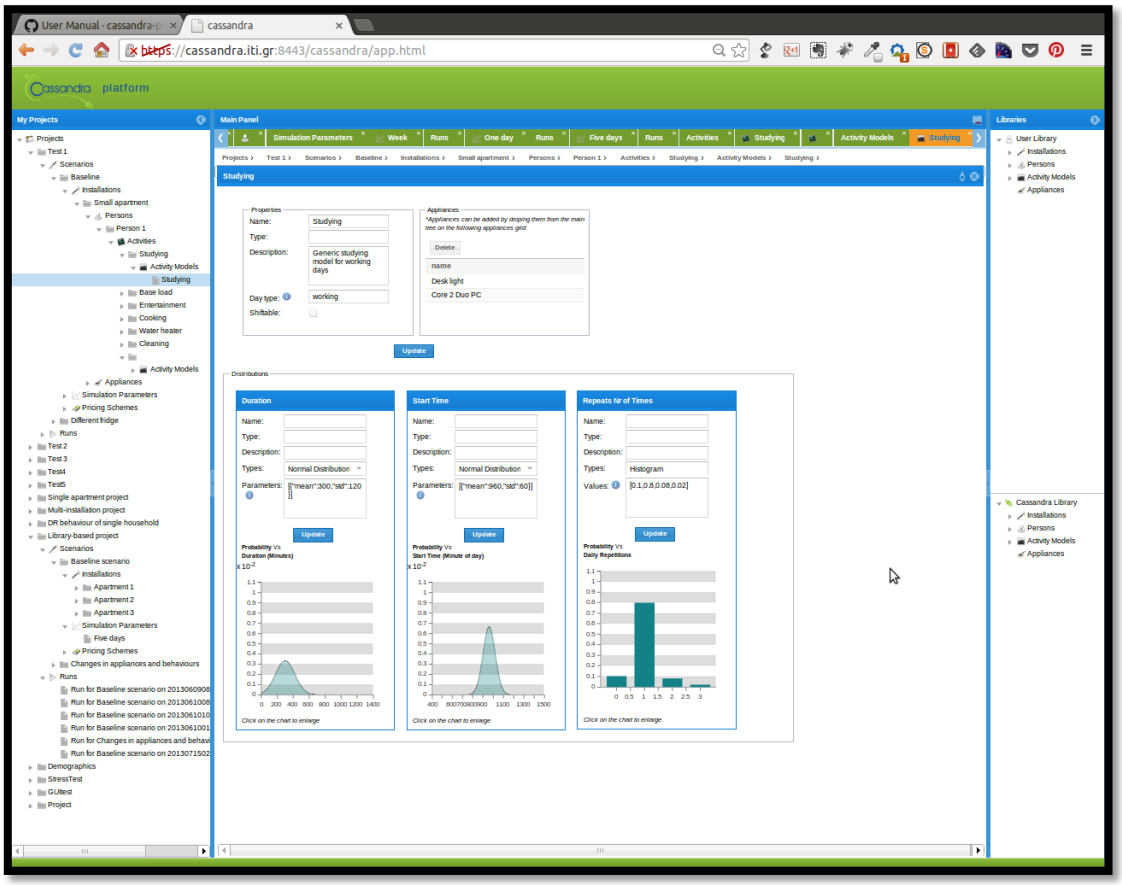

Figure 28: Completed activity model form

2.22 Appliance and Consumption Model

When creating/editing an appliance (Figure 29, Figure 30), the user enters the following data:

- Name: the name of the appliance
- Type: the type of the appliance
- Description: a short description of the appliance
- Energy Class: the energy class of the appliance
- Stand By: the stand-by consumption power
- Base: whether the appliance constitutes a base load
- Shiftable: whether the appliance is shiftable
- Controllable: whether the appliance is controllable

and with respect to the consumption model:

- Name: the name of the consumption model
- Description: a short description of the consumption model
- P-Expression: the expression that provides the active power curve
- Q-Expression: the expression that provides the re-active power curve

P-Expression and *Q-Expression* have the following form:

$$\{m\ \{n_1\ [p_1,d_1,s_1]\ [p_2,d_2,s_2]\},\ \{n_2\ [p_3,d_3,s_3]\},\ ...\}$$

and

$$\{m\ \{n_1\ [q_1,d_1,s_1]\ [q_2,d_2,s_2]\},\ \{n_2\ [q_3,d_3,s_3]\},\ ...\}$$

respectively, with:

- *p*: active power
- *q*: reactive power
- *d*: duration in minutes
- *s*: slope

Tuples p_1 and p_2 will be executed for n_1 timesteps and then p_3 tuple for n_2 timesteps. Those n_1*2+n_2 timesteps (since there will be n_1 steps for p_1/q_1 and n_1 steps for p_2/q_2) will be executed *m* times. Of course there can be n_3, n_4 etc. For loops we can set either *m* or *n* to 0. For example, lamp: {m=0 {n_1=1 [60,1,0]}}, refrigerator:

{"n":0,"params":[{"n":1,"values":[{"p":140,"d":20,"s":0},{"p":117,"d":18,"s":0},{"p":0,"d":73,"s":0}]}]}

Every appliance should have a consumption model.

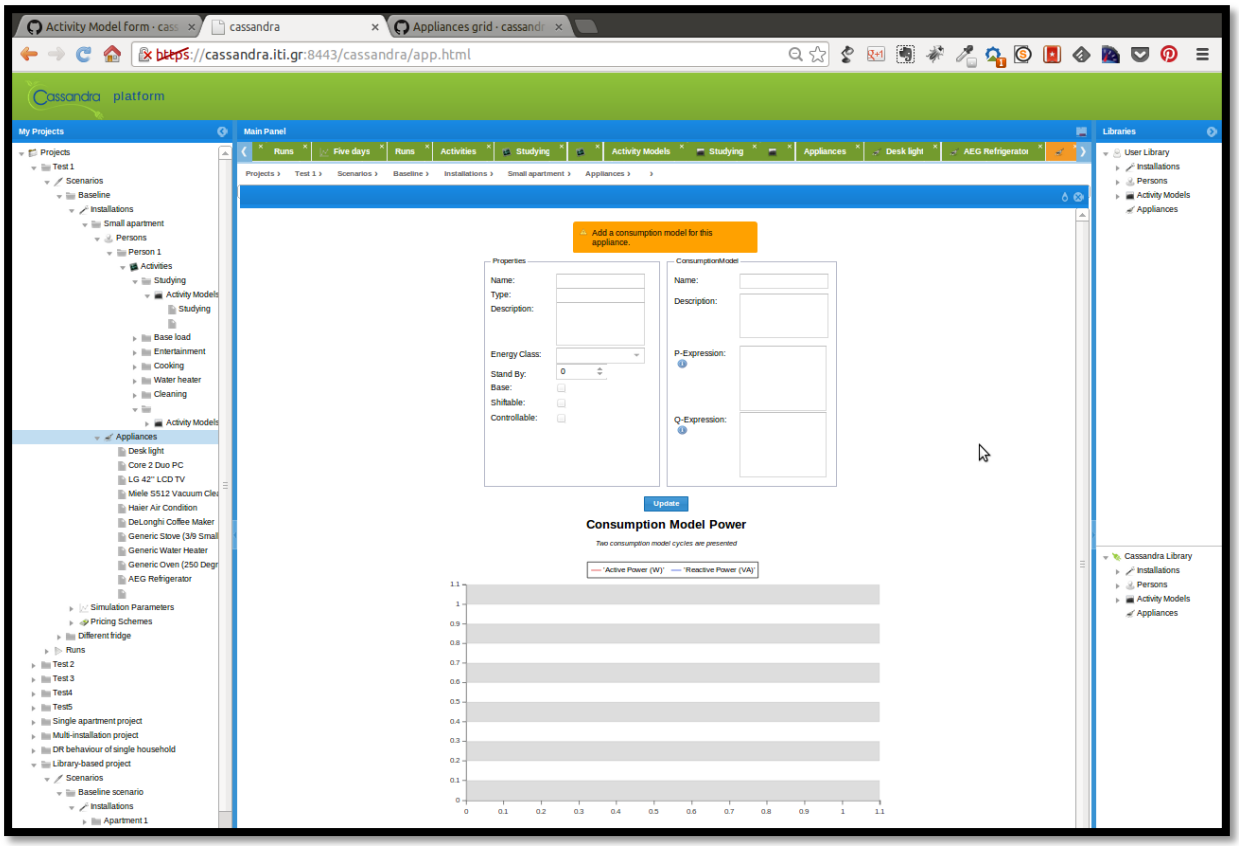

Figure 29: Appliance form

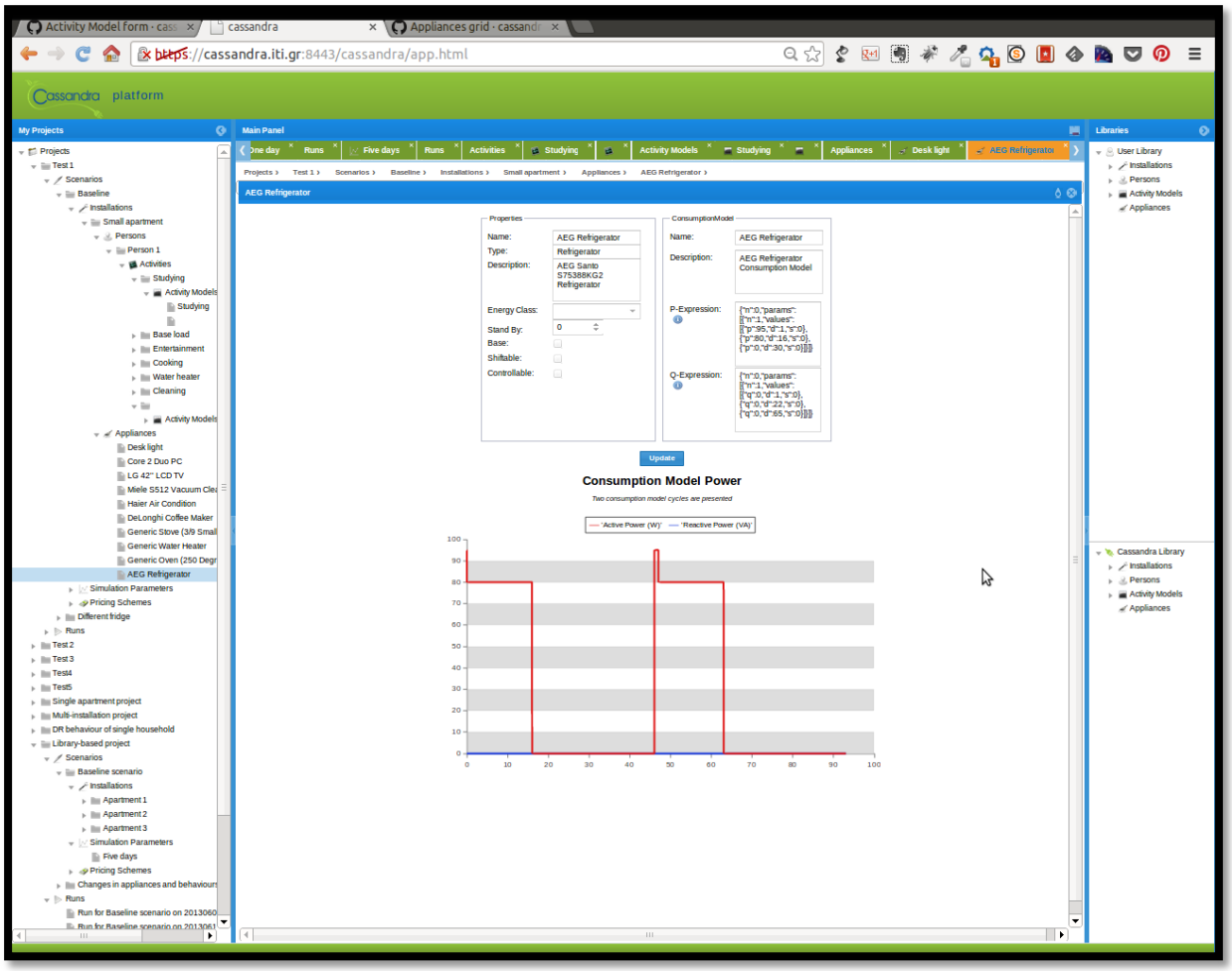

Figure 30: Completed appliance form

2.23 Simulation Parameters (Form)

When creating/editing simulation parameters (Figure 31), the user enters the following data:

- Name: the name of the parameters set
- Location: the location of the simulation
- Monte Carlo Runs: how many times will the simulation run
- Date Started: the starting date of the simulation
- Date Ends: the ending date of the simulation
- Notes: notes on the simulation parameters set
- Pricing Scheme: the pricing scheme under which the energy consumption of the installations will be billed. The pricing scheme is inserted through drag'n'drop from the Pricing Schemes nodes.

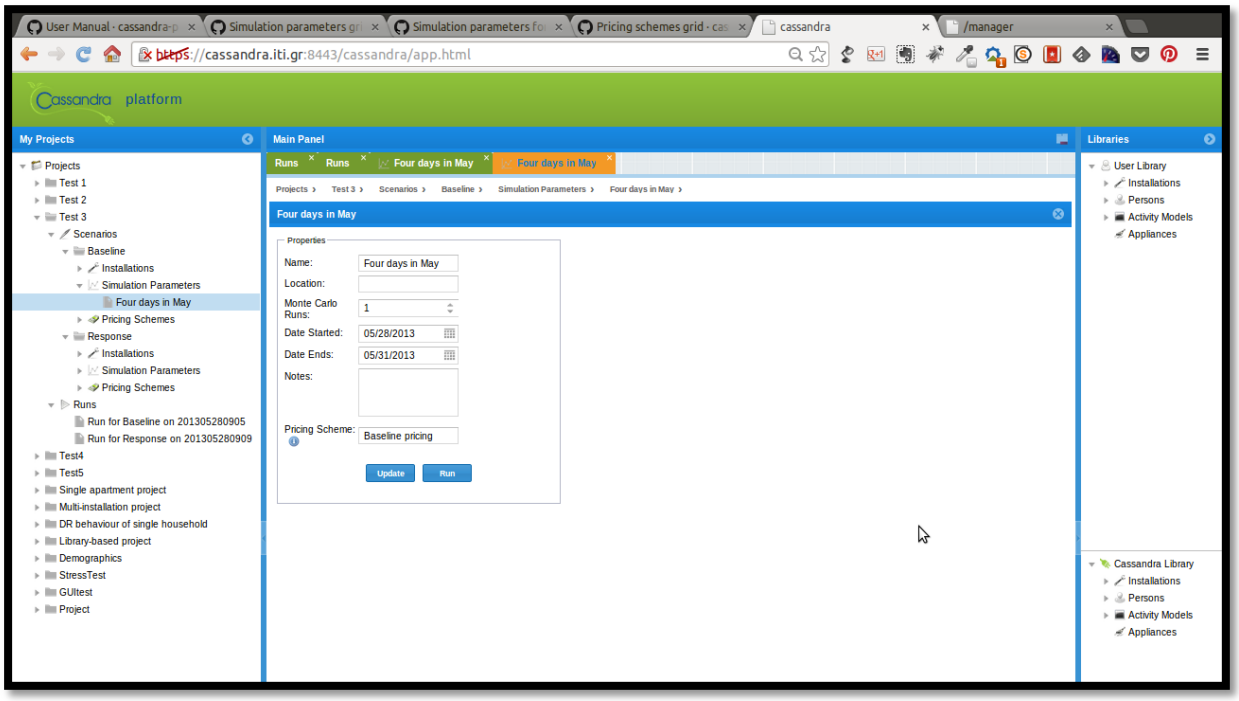

Figure 31: Simulation parameters form

2.24 Pricing Scheme

The CASSANDRA platform at the moment supports five pricing schemes:

- Scalar Energy Pricing
- Scalar Energy Pricing with Off-peak time zones
- Combined Energy and Power Pricing
- All-Inclusive Pricing
- Time-Of-Use Pricing

The schemes are presented in the Sections below along with examples.

2.24.1 Scalar Energy Pricing

The properties that need to be filled by the user are (Figure 32):

- Name: the name of the scheme
- Type: Scalar Energy Pricing
- Description: a short description of the scheme
- Billing Cycle: in days
- Fixed Charge: fixed charge for every billing cycle
- Levels: pairs of price and energy levels

For example:

- Billing cycle: 4 months

- Measured consumption: 1500 kWh
- Fixed charge: 15
- Price Level 1: [500 Kwh,0.06]
- Price Level 2: [400 Kwh,0.07]
- Price Level 3: [400 Kwh,0.08]
- Price Level 4: [0.1]

$$Cost = 0.06 * 500 + 0.07 * 400 + 0.08 * 400 + 0.01 * 200 + 15 = 125$$

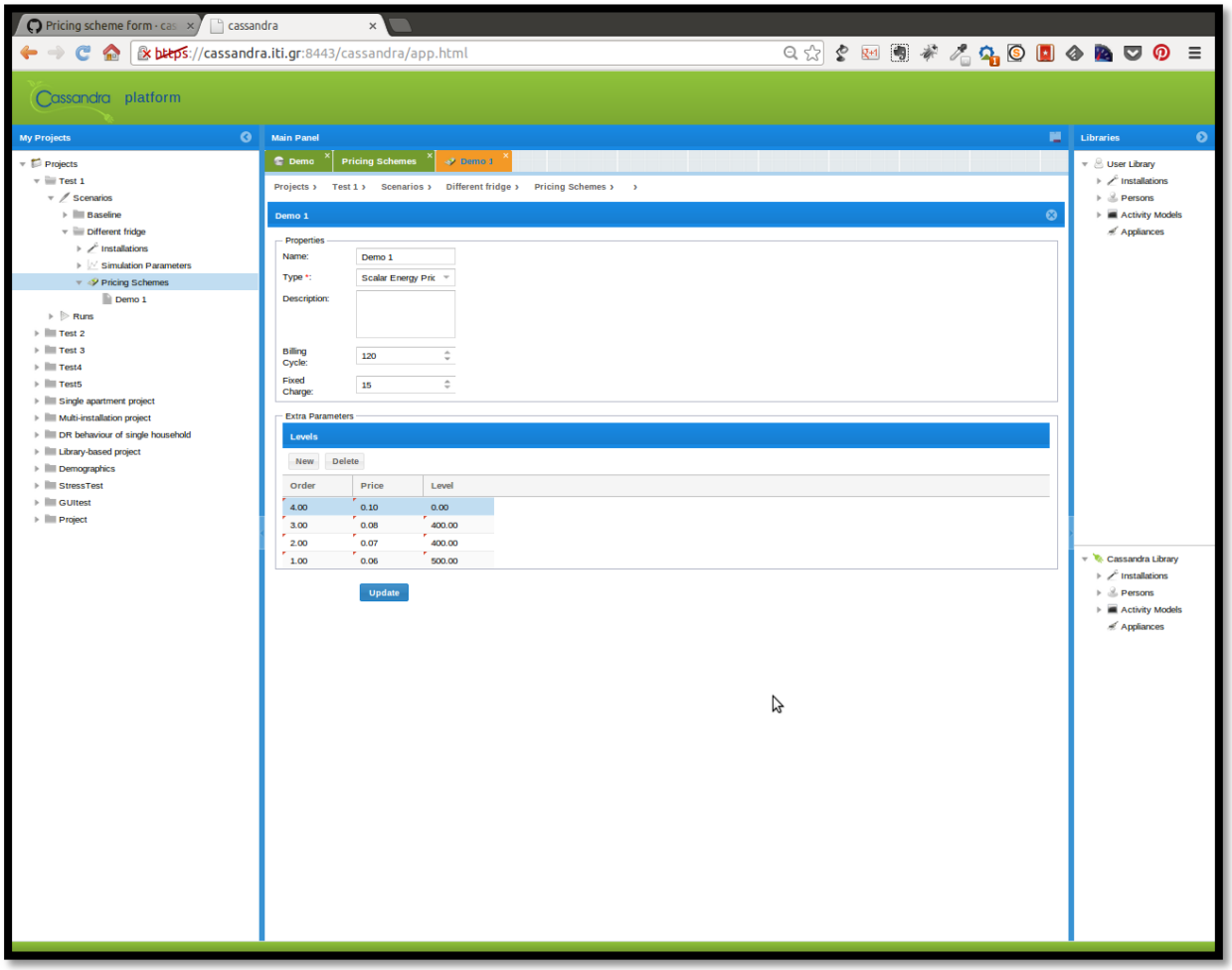

Figure 32: Scalar Energy Pricing form

2.24.2 Scalar Energy Pricing Time Zones

The properties that need to be filled by the user are (Figure 33):

- Name: the name of the scheme
- Type: Scalar Energy Pricing Time Zones
- Description: a short description of the scheme

- Billing Cycle: in days
- Fixed Charge: fixed charge for every billing cycle
- Levels: pairs of price and energy levels
- Offpeak: define the off peak hours of pricing

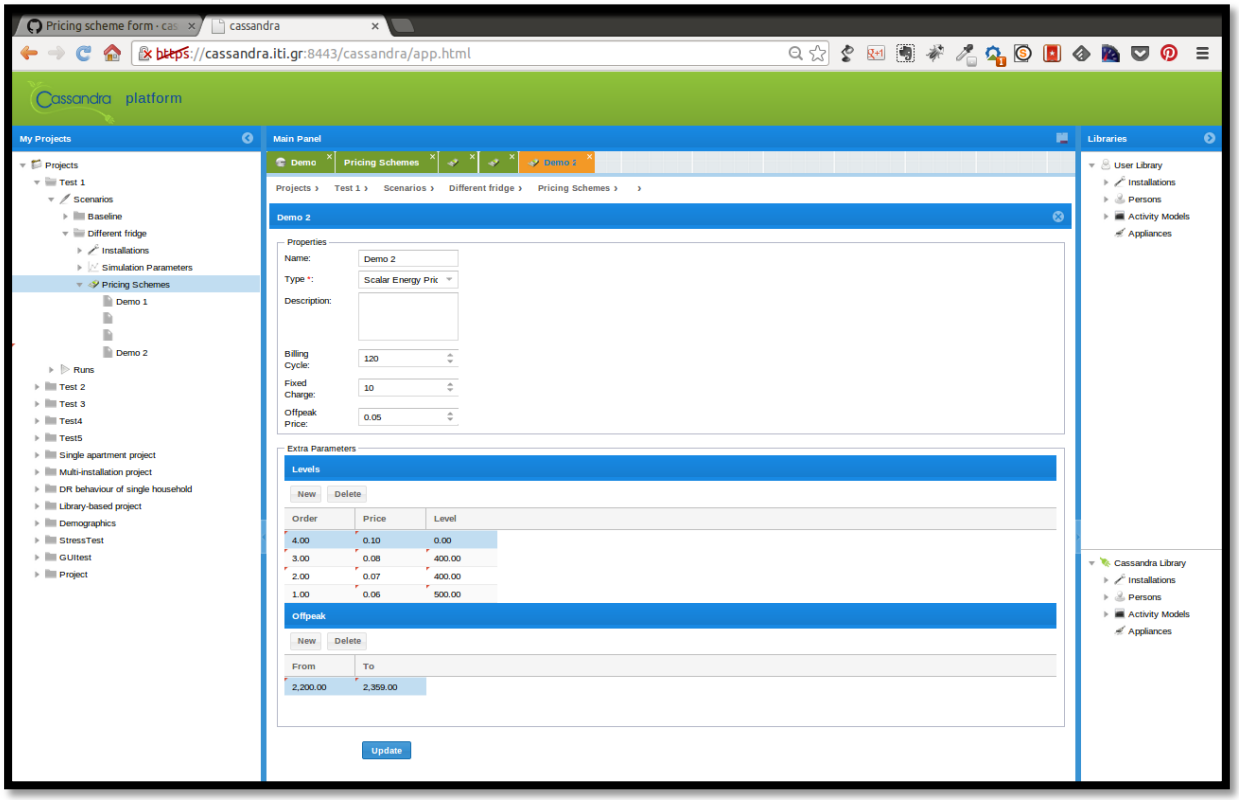

Figure 33: Scalar Energy Pricing Time Zones form

For example:

- Billing cycle: 4 months
- Measured consumption: 1000 kWh in peak and 500 Kwh in off-peak
- Fixed charge: 15
- Price Level 1: [500 Kwh,0.06]
- Price Level 2: [400 Kwh,0.07]
- Price Level 3: [400 Kwh,0.08]
- Price Level 4: [0.1]
- Offpeak price: [0.05]

$$Cost = 0.06 * 500 + 0.07 * 400 + 0.08 * 100 + 0.05 * 500 + 15 = 105$$

2.24.3 Combined Energy and Power Pricing

The properties that need to be filled by the user are (Figure 34):

- Name: the name of the scheme

- Type: Combined Energy and Power Pricing
- Description: a short description of the scheme
- Billing Cycle: in days
- Fixed Charge: fixed charge for every billing cycle
- Contracted Capacity: The contracted power capacity
- Energy Price: The price of energy consumed
- Power Price: The power pricing of the contracted capacity

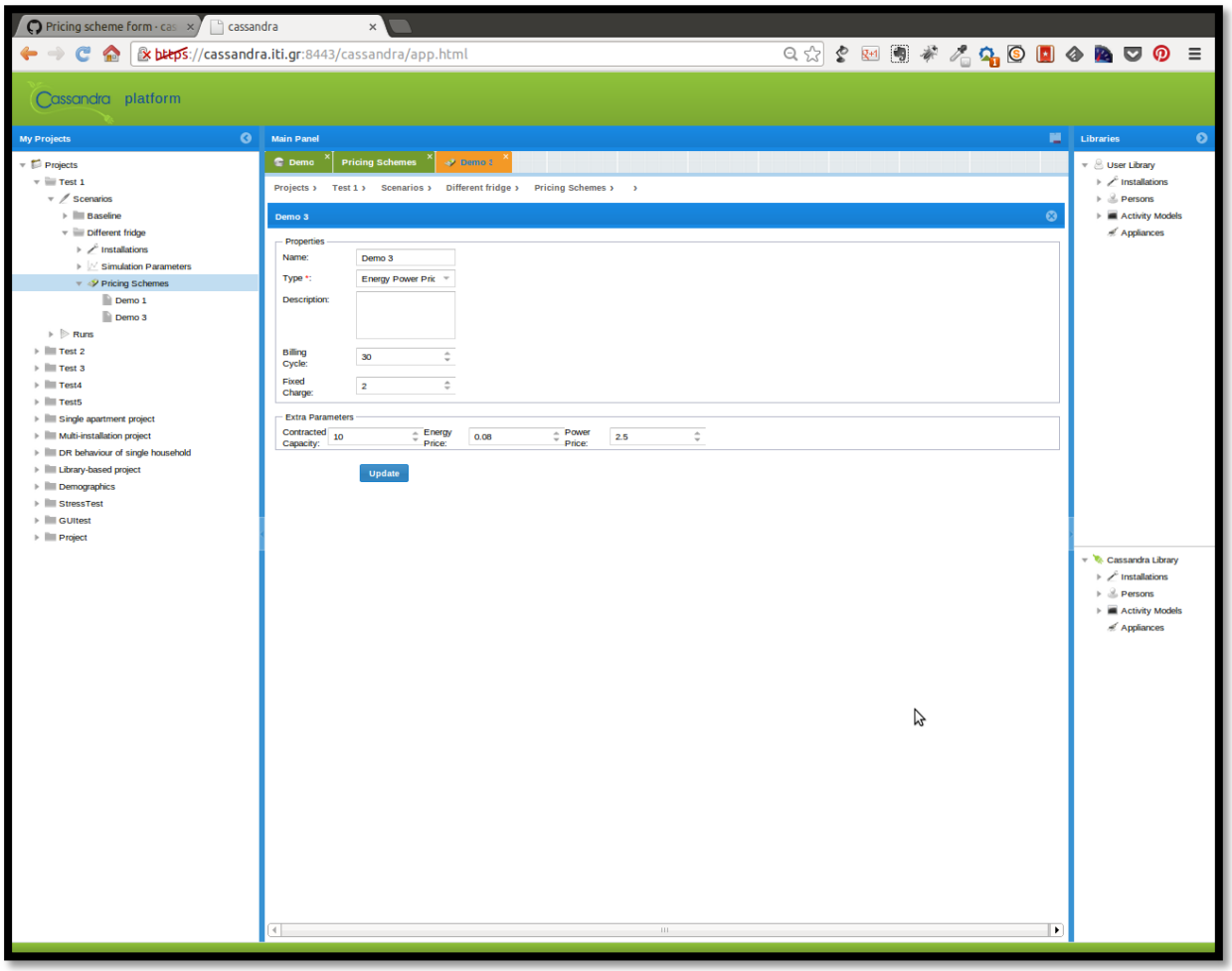

Figure 34: Combined Energy and Pricing form

For example:

- Billing cycle: 1 month
- Measured consumption: 350 kWh with contracted capacity of 10kW
- Fixed charge: 2
- Energy price: 0.08
- Power price: 2.5

$$\text{Cost} = 0.08 * 350 + 2.5 * 10 + 2 = 55$$

2.24.4 All-inclusive pricing

The properties that need to be filled by the user are (Figure 35):

- Name: the name of the scheme
- Type: All-inclusive pricing
- Description: a short description of the scheme
- Billing Cycle: in days
- Fixed Charge: fixed charge for every billing cycle
- Contracted Energy: The contracted energy
- Fixed cost: The price of energy contracted
- Additional cost: The price of additional energy

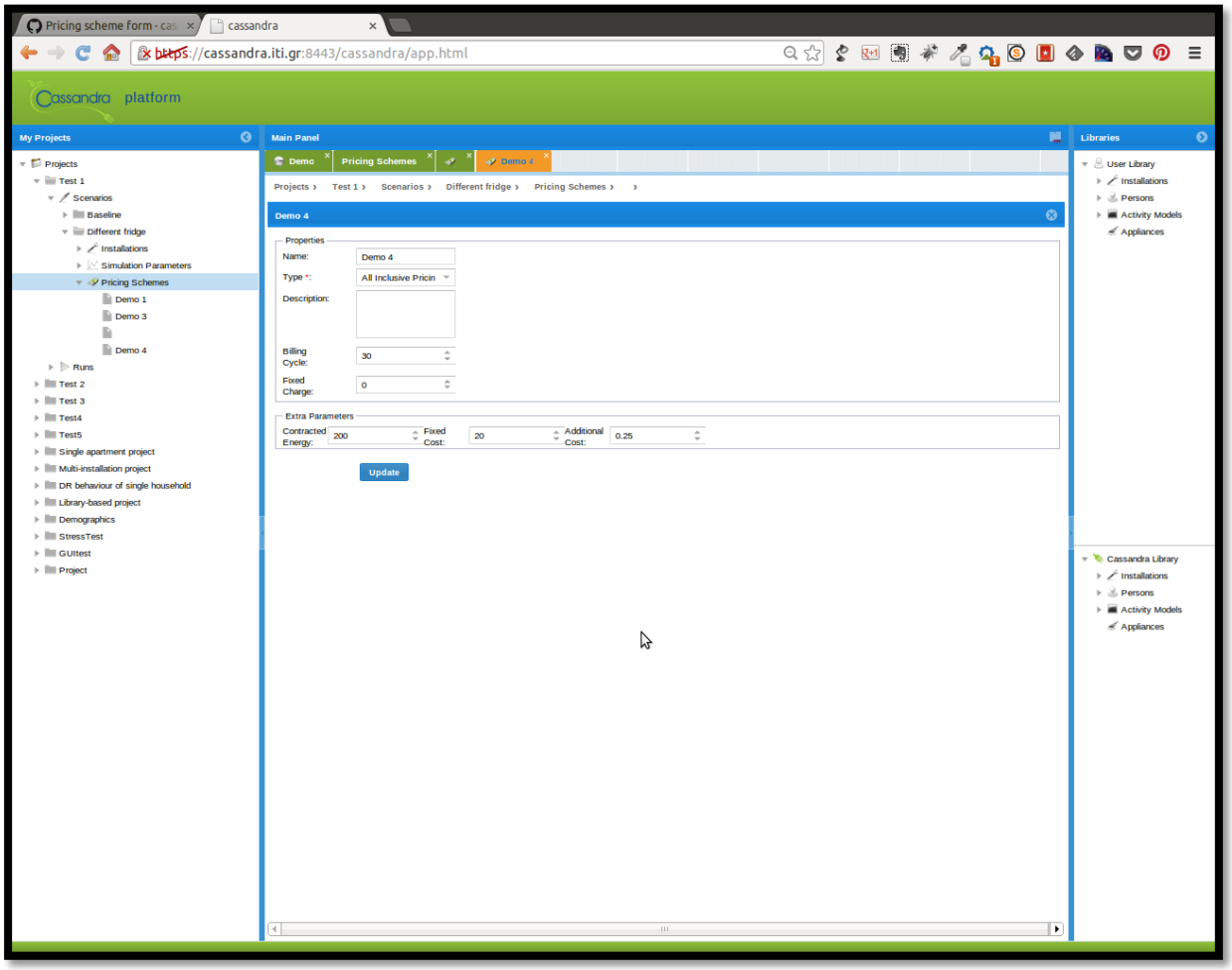

Figure 35: All Inclusive Pricing form

For example:

- Billing cycle: 1 month
- Fixed charge: 0
- Measured consumption: 300 kWh

- Fixed cost for contracted kWh consumption: 20
- Energy price for additional consumption above the contracted: 0.25
- Contracted consumption: 200

$$Cost = 20 + 0.25 * 100 + 0 = 45$$

2.24.5 Time-Of-Use pricing

The properties that need to be filled by the user are (Figure 35):

- Name: the name of the scheme
- Type: TOU pricing
- Description: a short description of the scheme
- Billing Cycle: in days
- Fixed Charge: fixed charge for every billing cycle
- Timezones: price of energy in specific timezones

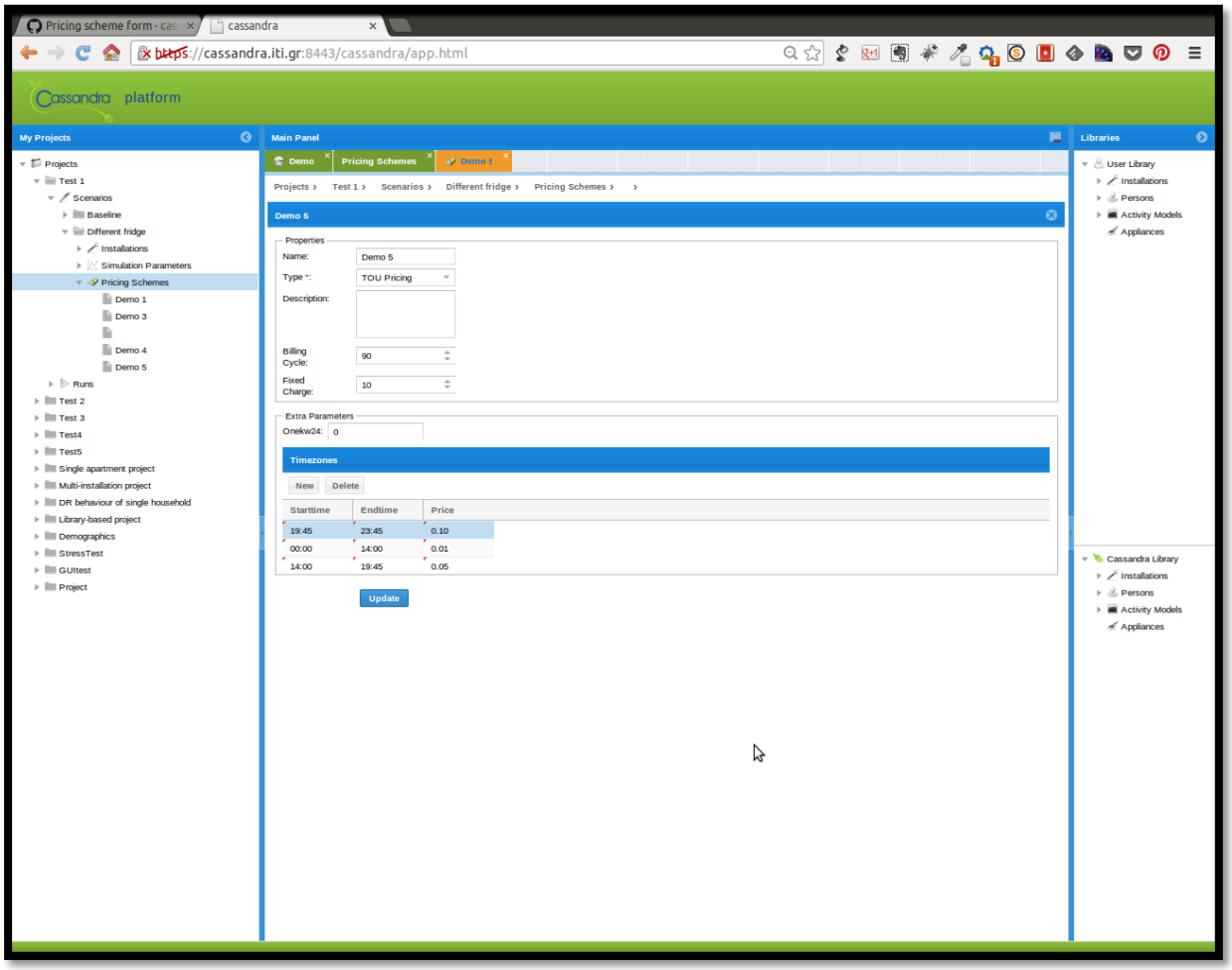

Figure 36: TOU pricing form

2.25 Run

When double-clicking on a *Run*, the platform opens a new tab in the browser where the user can see the results of the simulation (Figure 37). In particular the user is presented with:

- Load curves
- Initial KPIs
  - Max Power
  - Avg Power
  - Energy consumed
  - Cost

The run also includes a snapshot of the scenario entities, the moment it was ordered to run by the user.

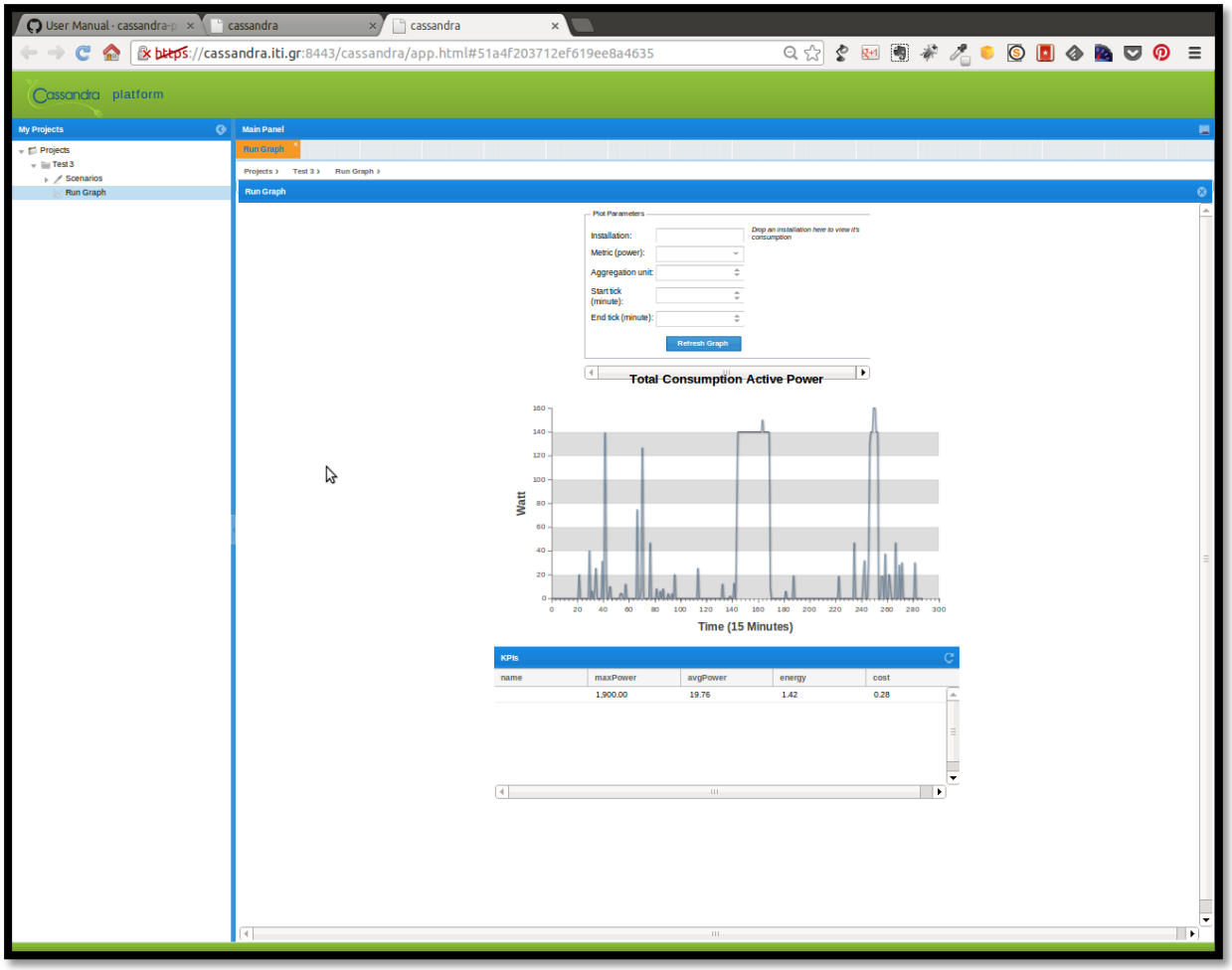

Figure 37: Run window

3 Training Module

This section focuses on the *Cassandra Training Module*. The *Training Module* is a useful add-on to the main *Cassandra* platform, creating *Entity* models (*Installations*, *Appliances*, *Persons* etc.) semi-automatically, using consumption datasets provided by the user. In detail, the *Training Module* can be used for:

- *Importing installation measurements*, for automatic appliances' identification and consumption patterns via *Disaggregation*.
- *Training Activity Behaviour Models* from detected consumption events.
- *Implementing Response Models*, given the basic behaviour model, pricing scheme and the response type as inputs.
- *Exporting Entity Models* to the main platform.

This process can make the use of CASSANDRA much easier for a naïve user, who may not be an expert on Power Systems.

The Training Module user interface is separated into four tabs (as seen in Figure 38):

1. *Import Data*
2. *Training Behaviour (Activity) Models*
3. *Create Response Models*
4. *Export Models*

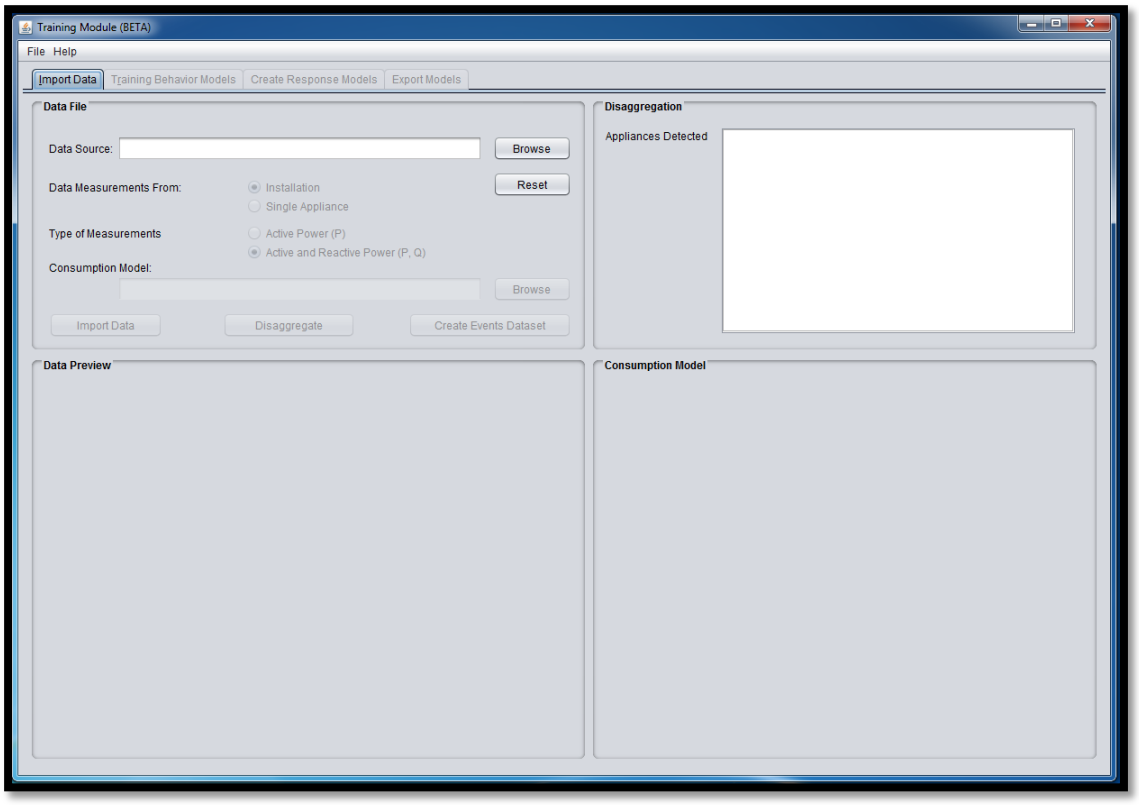

Figure 38: Training Module's Main Screen

The user should note that not all of the tabs are enabled when launching the *Training Module*, in order to guide the user over the procedure, making sure that the correct steps are followed.

3.1 Import Data Tab

Import Data Tab is used in order to import available installation consumptions into the Training Module. It is comprised by 4 different panels, each used for choice selection or data visualization (Figure 39).

- *Data File Panel:* This panel is used for choosing the data files of the available installation measurements, as well as several data attributes, which will help parsing the data set. The dataset can contain either consumption data from a single appliance (if plug-wise measurements are available) or from an installation as a whole.
  It is recommended that the measurements included in the file are both of active and reactive power, in order for the disaggregation to work optimally. In case only active power measurements are available, the resulting appliance identification shall be suboptimal.
  In case of a single appliance, the consumption model of the appliance should be provided, in order to create the consumption event dataset.

- *Disaggregation Panel:* This panel contains the detected appliance(s) from the disaggregation process.

- *Data Preview Panel:* The consumption data contained in the file is visualized in this panel, so that the user has a first overview over the selected data set. The visualization shows the time series of the active and the reactive power, if available.

- *Consumption Model Panel:* In this panel, the consumption model of the selected appliance from the (detected) appliances list is presented here. The visualization shows both the consumption pattern for the active and the reactive power of the model.

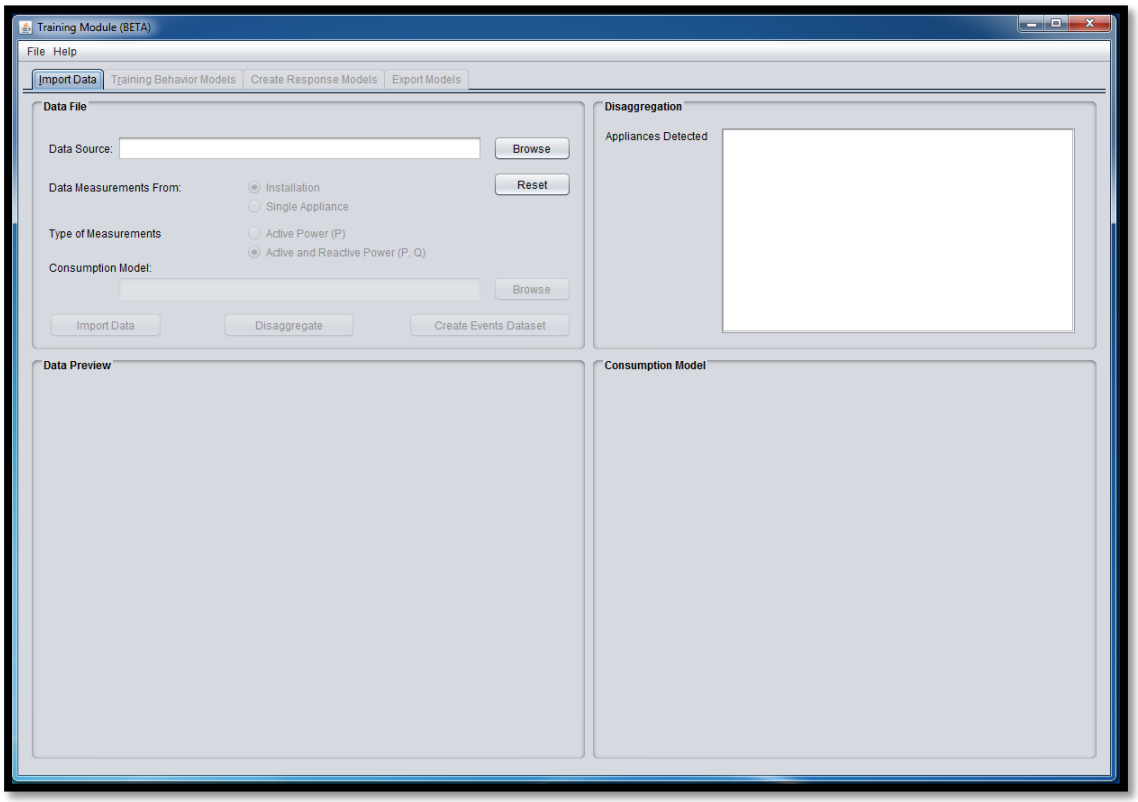

Figure 39: Import Data Tab

3.1.1 Usage workflow

The main usage workflow is as follows:

1. The user browses through her files to choose the data file with the consumption measurements of an installation or a single appliance. The allowed file types for the consumption data sets are .xls (excel files) and .csv (comma separated files). More

details on the structure of the files can be found in the Specifications Document [D3.6].

2. The file selection enables the *Data Parameters* buttons (Figure 40). These buttons allow for the user to provide some more details on the data contained within the chosen file: The source of data measurements (installation or single appliance in case of plug-wises) and the type of measurements (Active and Reactive Power).

   After setting the parameters, the user should press the import button which is responsible for parsing the provided file and visualizing the data. In case of an error in the data file, the parser returns an error message, including the line where the error was found, to help the user fix the erroneous data entry.

   Also, the *Export* tab is enabled, even though the only exportable entities up to now should be the installation and the equivalent person inhabiting it.

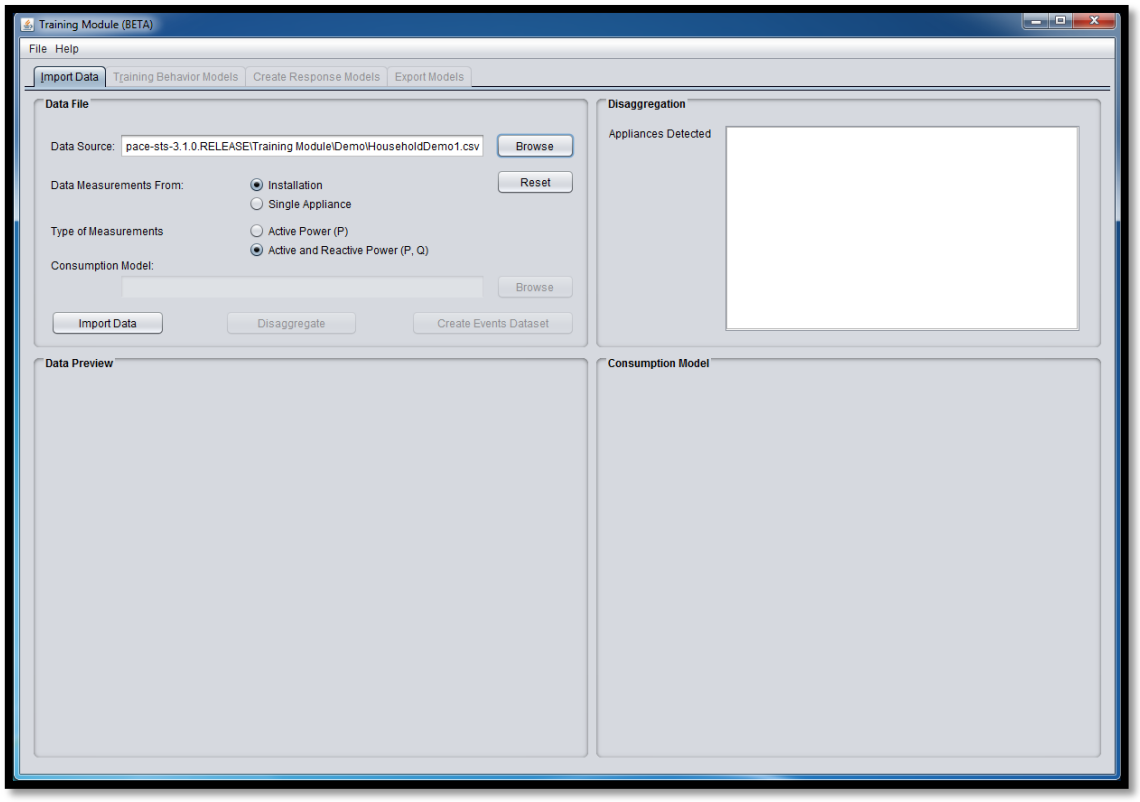

Figure 40: Enabled setting after selecting the data file

3. Given the choice of measurements' source, one of two buttons is enabled:
   - In case of installation, the *Disaggregation* button is enabled (Figure 41).

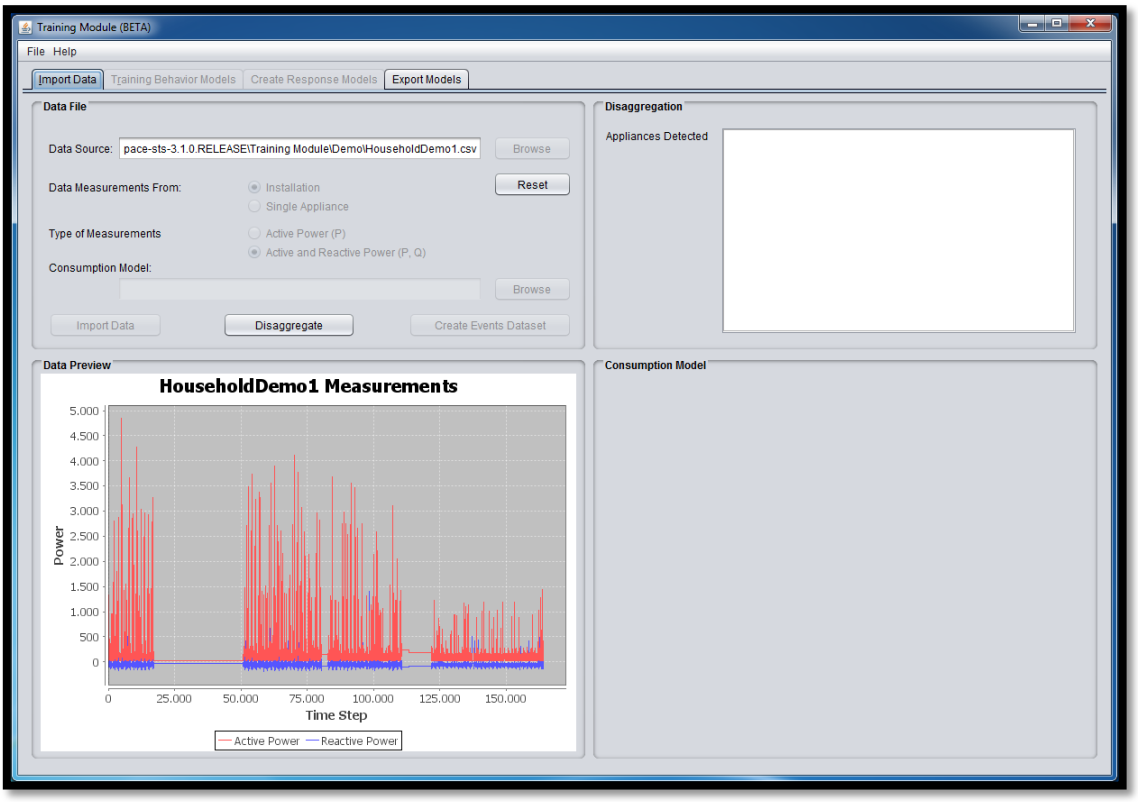

Figure 41 Imported data in case of installation

- In case of a single appliance, the user must provide the file containing the appliance's *Consumption Model*. After that, the *Create Event* button is enabled (Figure 42).

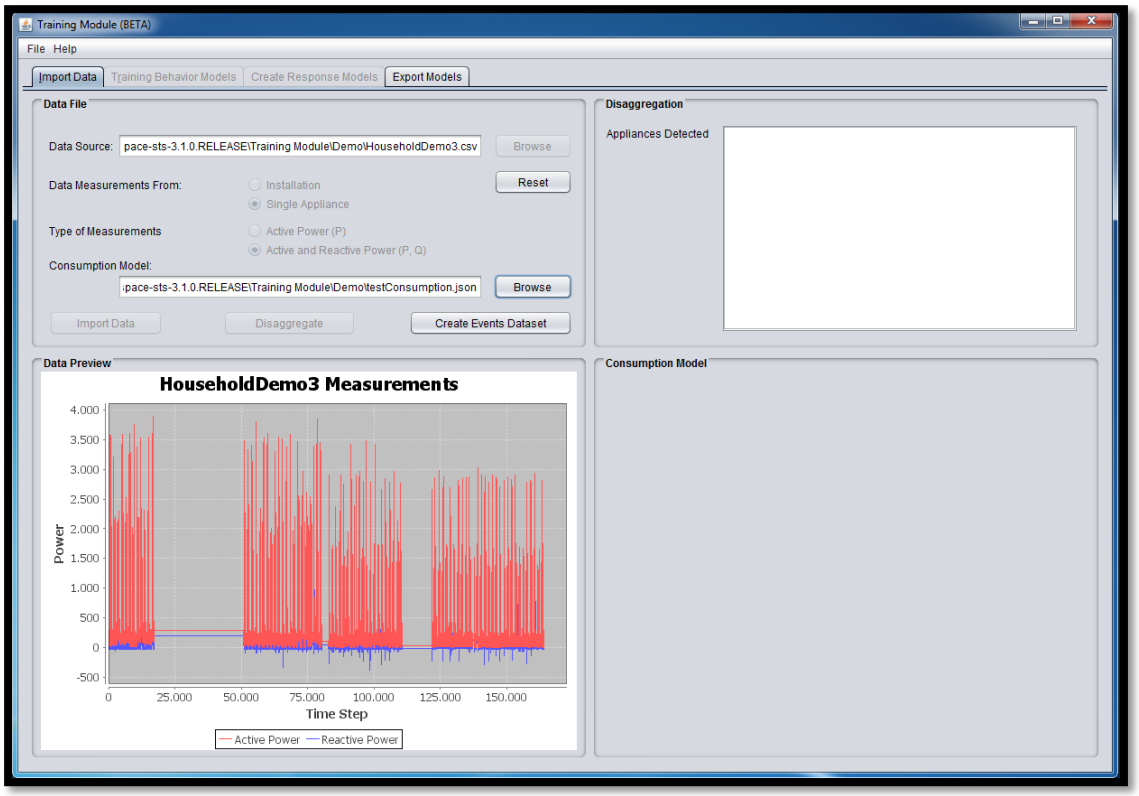

Figure 42 Imported data in case of single appliance

4. After pressing the corresponding button for each case, a list of the *Detected Appliances* is presented on the *Disaggregation* panel. When the user clicks on an appliance, the appliance's *Consumption Model* appears on the lower right panel (Figure 43).

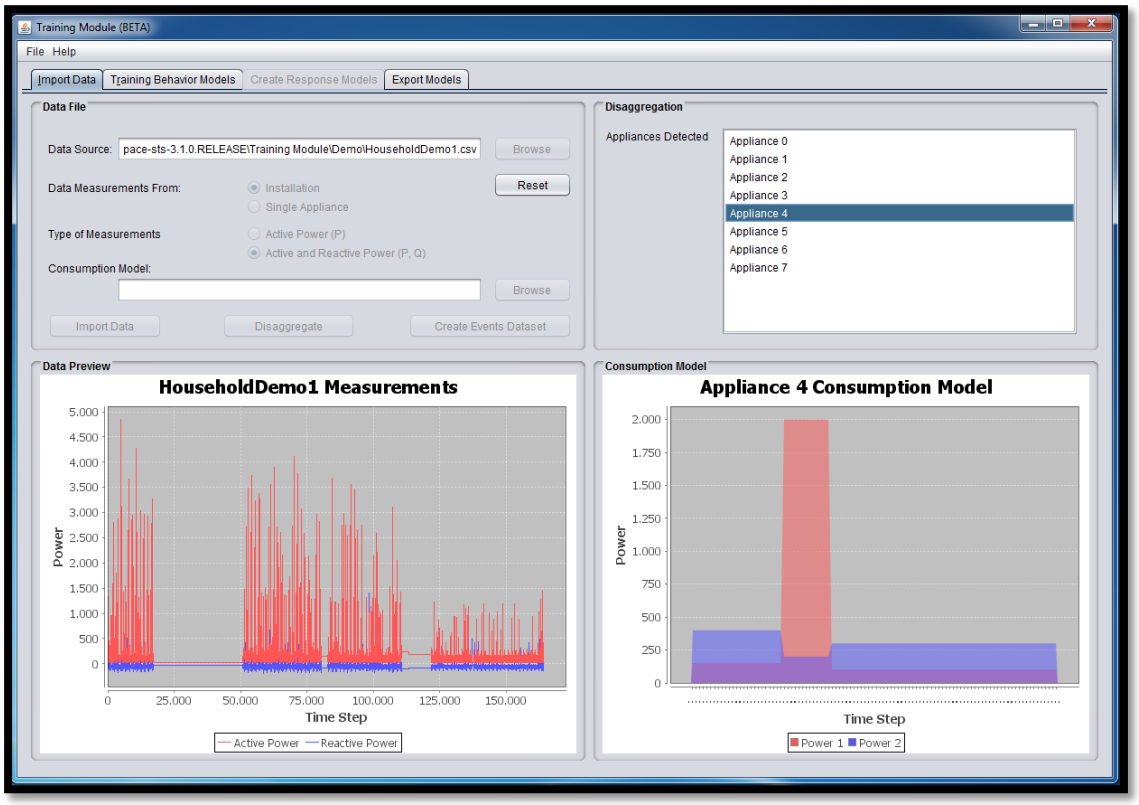

Figure 43: Disaggregation results

After the appliances identification, the data import procedure has been completed. The *Training* tab is activated. More details over this tab can be on the next section.

3.2 Training Behaviour Models Tab

Training Behaviour Models tab is used to create activity models from the events imported or detected for each appliance. This tab is also comprised by 4 different panels, each used for different choice selections or data visualizations (Figure 44).

- *Training Parameters Panel:* This panel is used for choosing the distributions used for each behaviour model attribute at model training. There are 3 alternative choices: *Histogram, Normal Distribution* and *Gaussian Mixture*.
  For the *Times per Day* attribute, Histogram Distribution is the only available choice, since the values are discrete and small in number. The usage of the other two options would produce suboptimal results for this variable.

- *Appliance/Activity Selection Panel:* In this panel, the list of the detected appliances is presented.
  In case of a single appliance, the list contains the same appliance with the appliance on the previous tab. In case of an installation, the activities that were identified as a result

of the disaggregation process are presented. These activities encapsulate all the installation's detected appliances presented in the previous tab.

- *Distribution Preview Panel:* After training behaviour models, this panel presents the distributions produced during the training procedure for the selected appliance/activity. There are 4 buttons, each corresponding to a different activity attribute: *Daily Times, Duration, Start Time* and *Start Time Binned Distribution.*
  The last distribution is produced by aggregating the *Start Time Distribution* in 10-minutes intervals and is used only for better visualization purposes (meaning this distribution is not used in the modelling of the activities/appliances)

- *Consumption Model Panel:* Same as the previous page. In case of an activity, the panel presents the consumption model of the first appliance in the activity's appliance list, in order to give a representative sample of the activity's consumption.

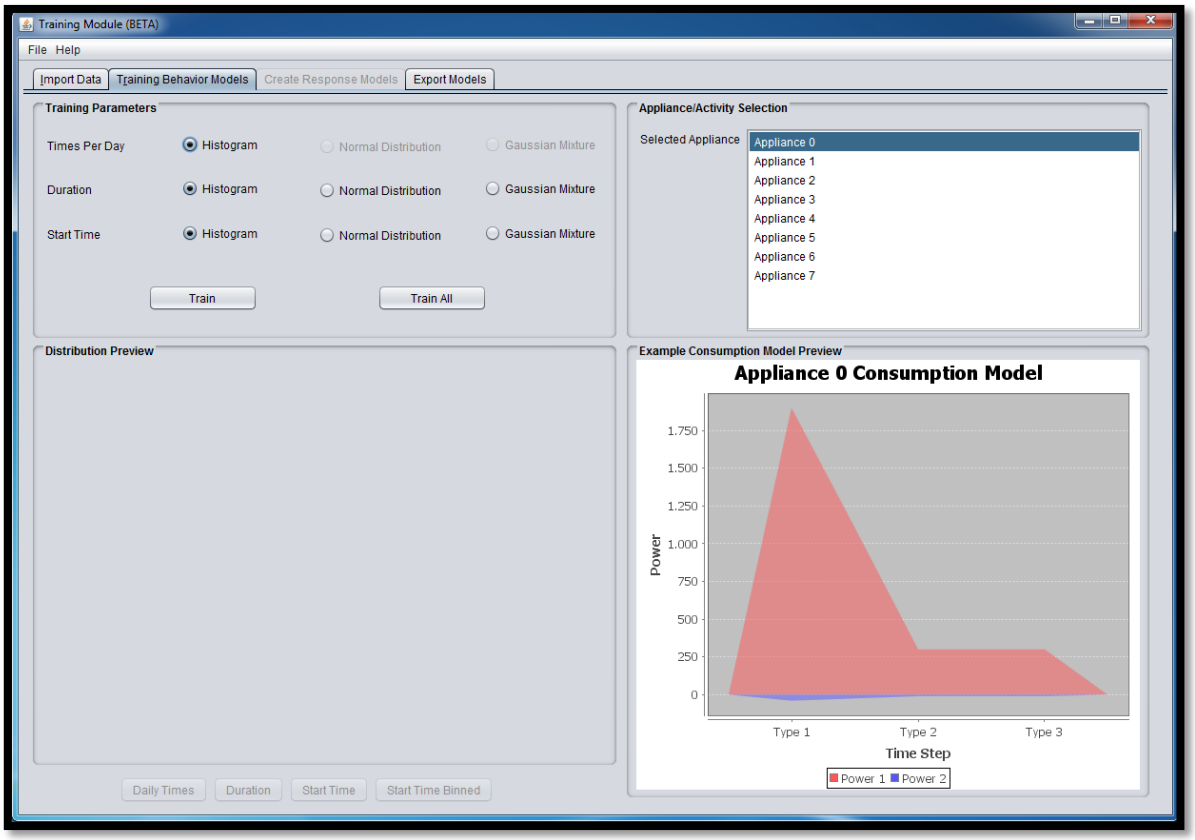

Figure 44: Training Tab

3.2.1 Usage workflow

The main usage workflow is as follows:

1. The user selects an appliance/activity from the list as well as the type of distributions of her preference for the training procedure. Then, she either presses the *Train* or *Train All* button to produce the expected behaviour models for a single appliance/activity or for all the detected appliances/activities.

2. After the completion of the training procedure, the distributions are graphically presented in the *Distribution Preview* panel and the buttons for choosing which distribution should become enabled (Figure 45).

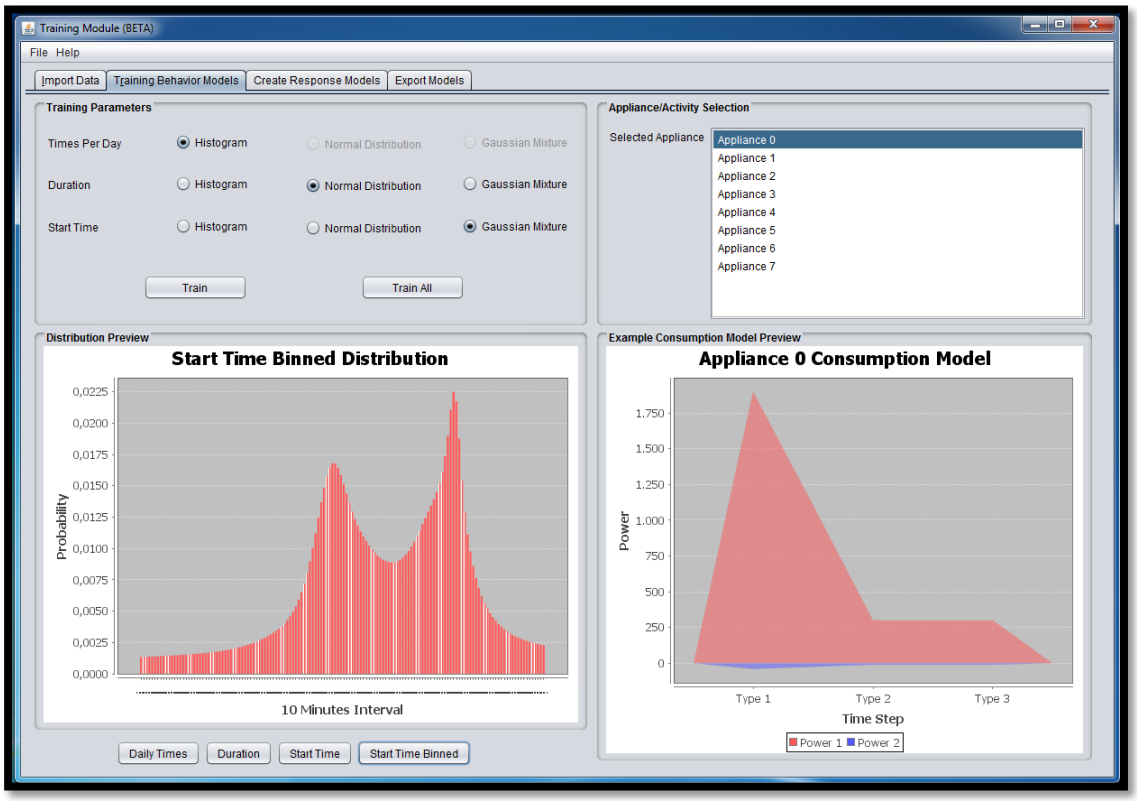

Figure 45: Results of training an example behaviour model

Also, the *Create Response Models* tab is activated. More details on this tab can be found in the next section.

3.3 Create Response Models Tab

Create Response Models tab is responsible for the creation of new behaviour models resulting from providing monetary incentives to the customer in order to change her consumption patterns. In order to do so, the pricing scheme and the response type must be provided.

It comprises 5 different panels, each used for choice selection or data visualization, as before (Figure 46).

- *Response Parameters Panel:* This panel is used for choosing the parameters that matter as far as the response of the customer to certain incentives (monetary or otherwise) is concerned. The user may select the model's sensitivity to money or its environmental awareness, as well as which case scenario the response trainer will use: *Best, Normal* or *Worst Case Scenario.* More details on the response types can be found in the Theoretical Modeling Deliverable [D3.4.2]

- *Behaviour Model Selection Panel:* In this panel, the list of the behaviours that are already produced by the training process (from the previous tab) is presented. The user can choose which one will be her baseline behaviour on which the new pricing scheme will be applied.

- *Pricing Schema Selection Panel:* This panel gives user the capability to set up the base pricing schema, as well as the new pricing policy she wants to test over the behaviour models. There is a parser that checks for the correctness of the schema and provides help in case of an error.

- *Behavioural Change Preview Panel:* This panel is responsible for the visualization of the behavioural change when the user wants to preview the result of the pricing policy to the selected behaviour. The attribute that is affected by the pricing scheme is the *Start Time Distribution*, so the panel presents a comparative chart of the start time distribution with the basic and the new pricing schemes.

- *Pricing Scheme Preview Panel:* This panel presents a comparative chart of the two pricing schemes provided by the user, in order to make sure that the input is the correct one.

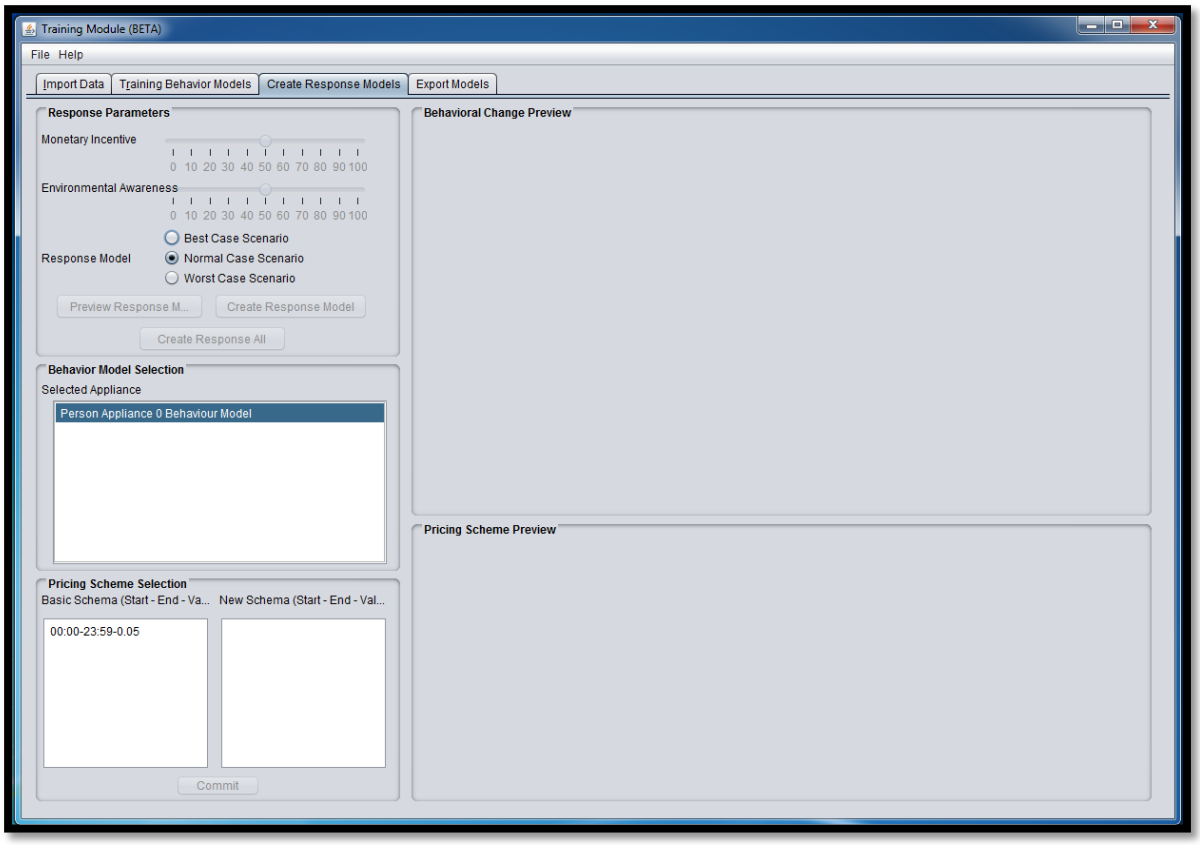

Figure 46 Create Response Models tab

3.3.1 Usage workflow

The main usage workflow is as follows:

1. The user chooses a behaviour model from the list. Next, she types in the pricing schemes (as shown in the Figure 47) as a triplet of starting time – ending time – price (suitable only for TOU Pricing scheme). More details on the specifications of the pricing scheme used in the Training Module can be found at the Specification Deliverable [D3.6].

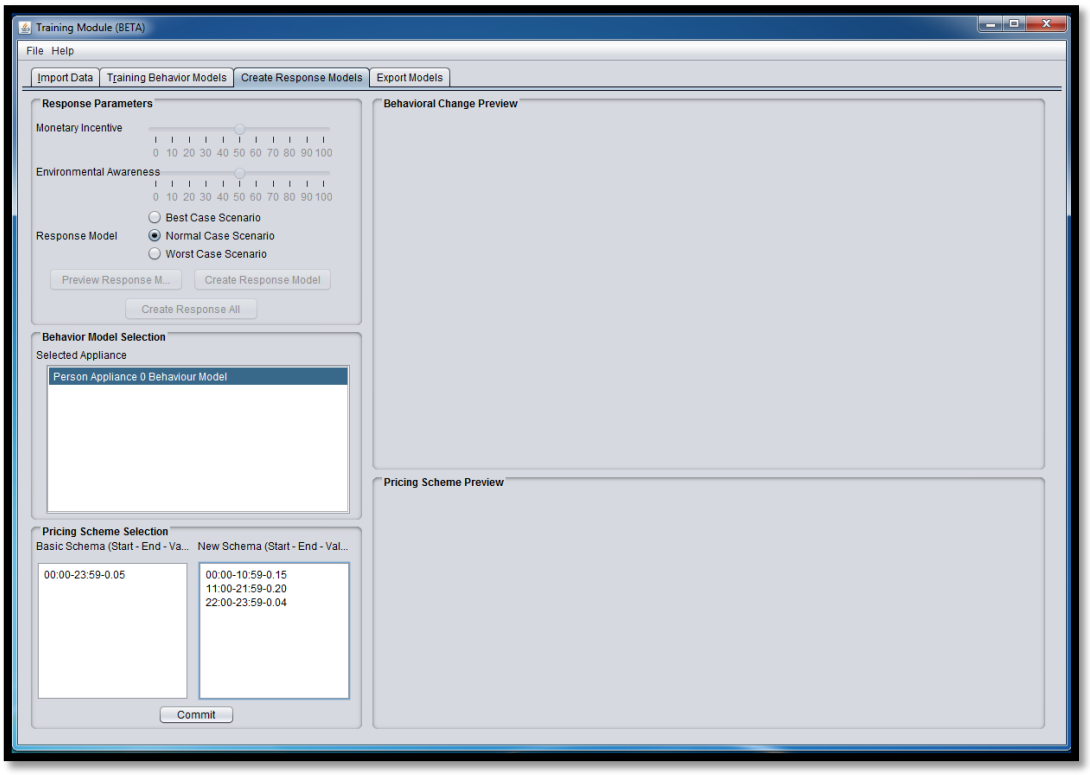

Figure 47 Inserting the pricing scheme

By pressing the *Commit* button the schemes are visualized in the *Pricing Scheme* panel (Figure 48). Also, the *Preview Response* button is enabled.

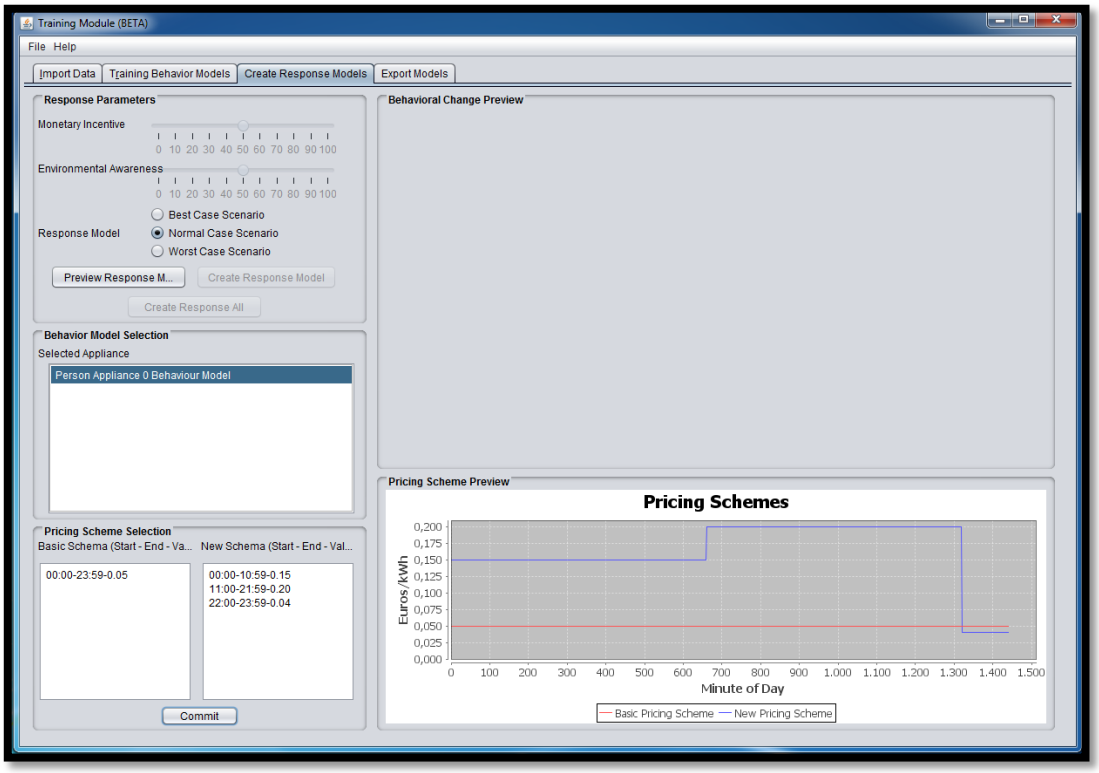

Figure 48 Visualizing pricing scheme and choosing response model

2. The user can now select the parameters for her response model from the *Response Parameters* panel. Then, she can press the *Preview Response* button to see the resulting model's *Start Time* distribution (Figure 49).

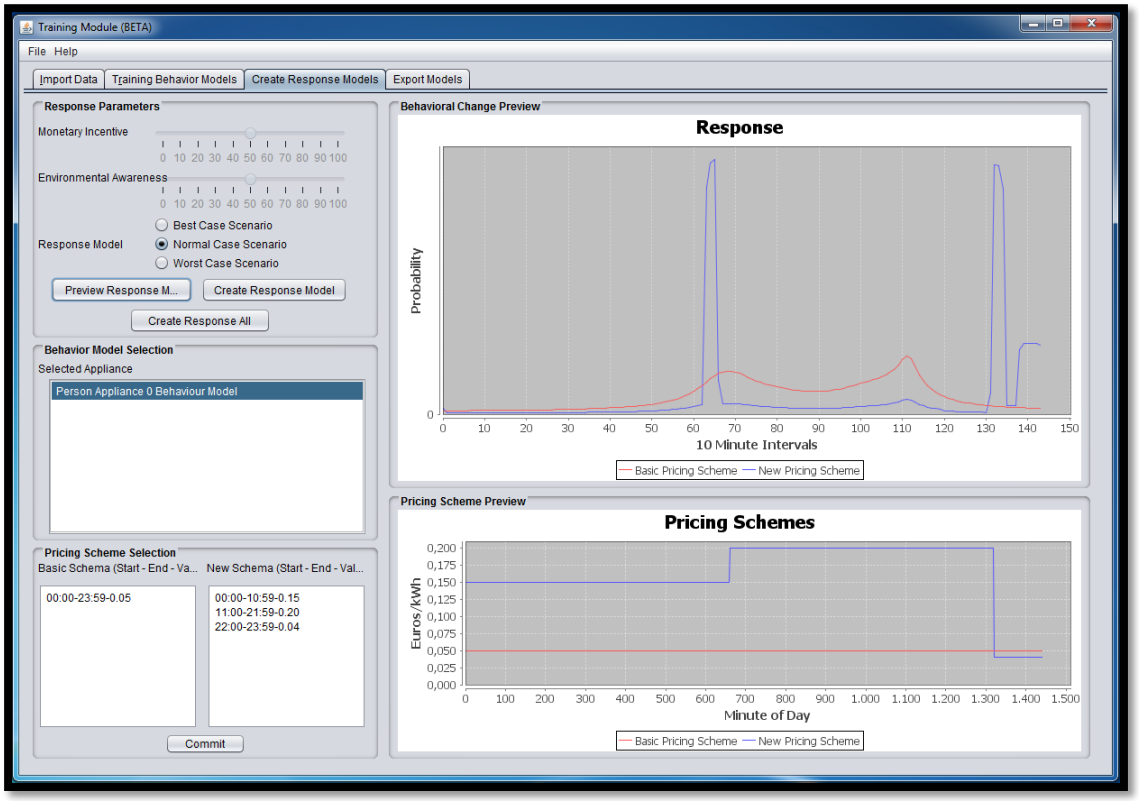

Figure 49 Response Preview

3. After visualizing the response model preview, the user can choose to create the response models for the behaviour selected by pressing the *Create Response Model* button, or she can use the same parameters to create new response models for all the available behaviours on the list, by pressing the *Create Response All* button.

The newly created response models are passed to the Export tab. More details about this tab can be found in the next section.

3.4 Export Models Tab

Export Models tab is responsible for connecting *Training Module* to the main *Cassandra Platform* and adding the created entities (installations, persons, appliances, behaviour (activity) and response) models to the user's library.

It is comprised by 3 different panels, each used for choice selection or data visualization (Figure 50).

- *Model Export Selection Panel:* This panel presents the list of the entities available for export. These can be any kind of entity (Installation, Person, Appliance, Behaviour (Activity) and Response) models.

- *Export Model Preview Panel:* This panel is responsible for visualizing the most appropriate attributes of the selected entity from the list above. The visualization is customized differently for each entity type selected.

- *Connection Properties Panel:* In this panel, the user can input her authentication credentials to the platform and then export the models she had created by selecting them from the entity model list.

  The authentication credentials are the same that are provided to the user for logging in the main Cassandra Platform and they are provided by the administrator of the Cassandra Server.

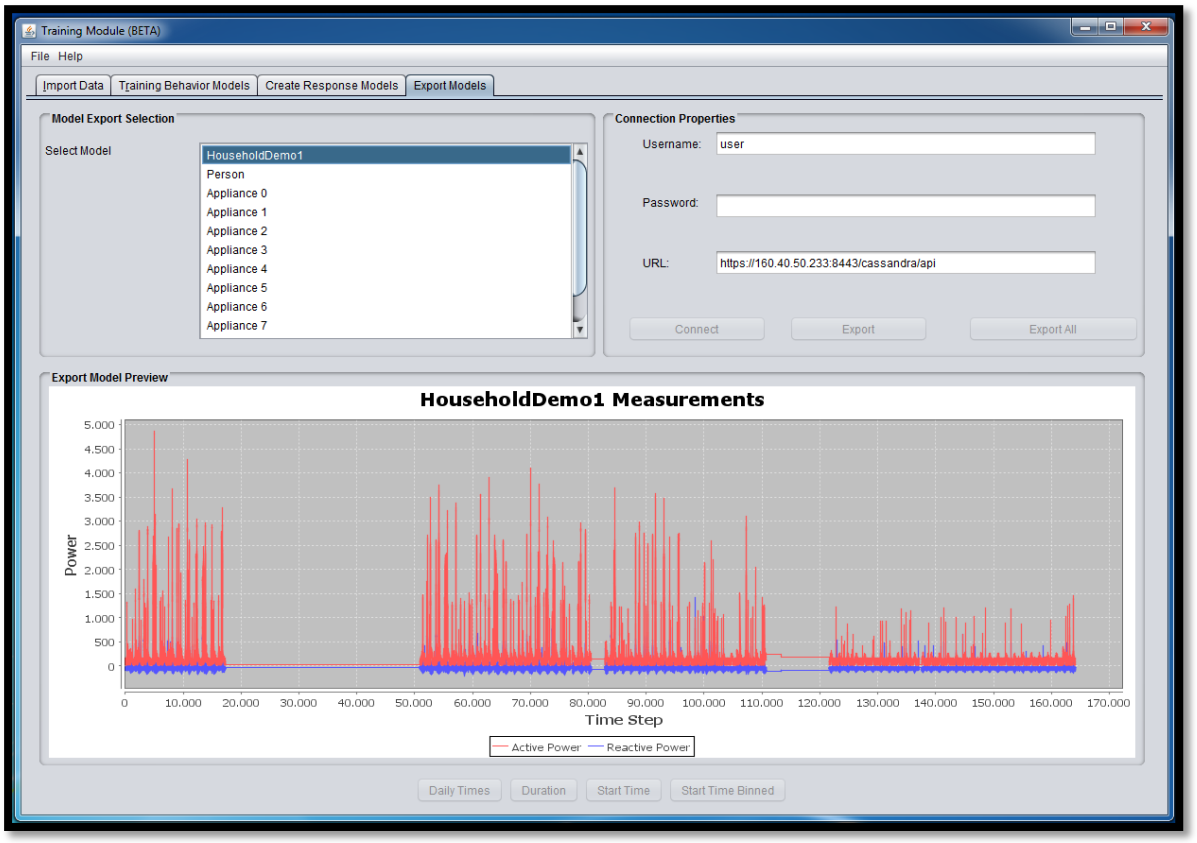

Figure 50 Export Models tab

3.4.1 Usage workflow

The main usage workflow is as follows:

1. The user selects the entity that she wants to export to the main platform. The visualization of the entity's properties is presented differently for each entity type. Some examples can be found in the figures below.

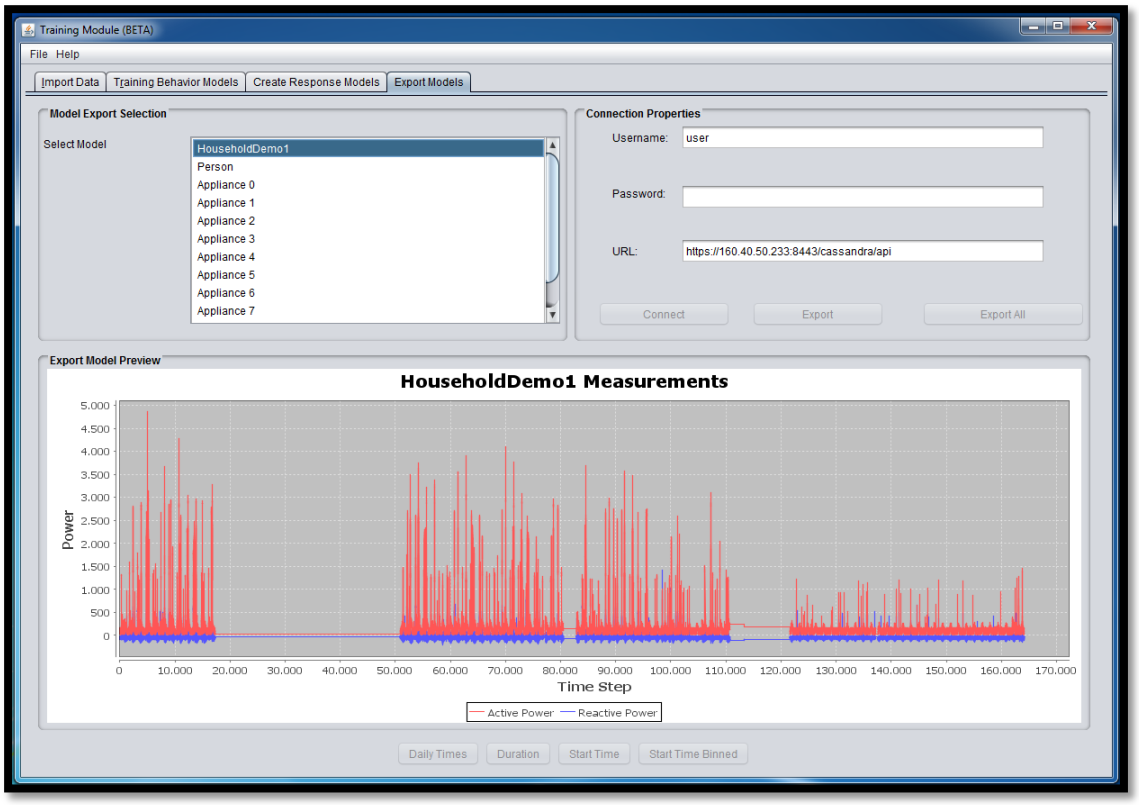

Figure 51 Export Installation models

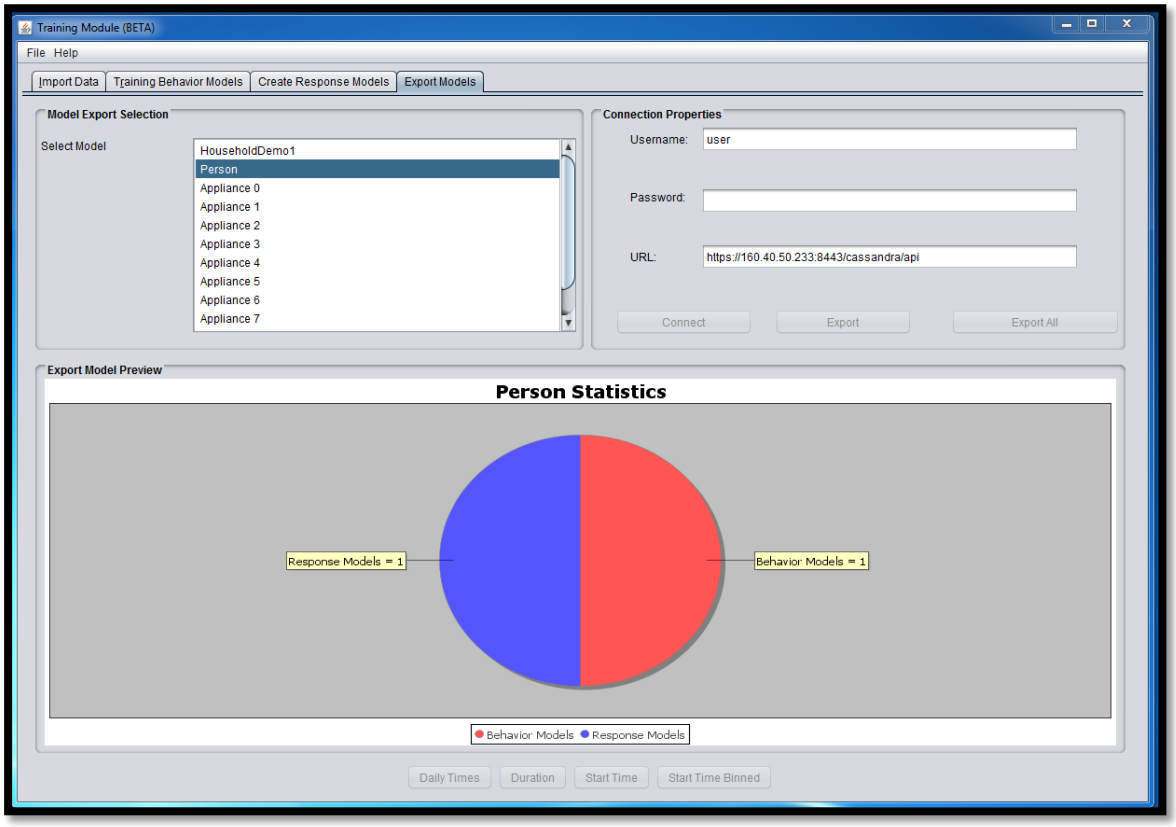

Figure 52 Export Person models

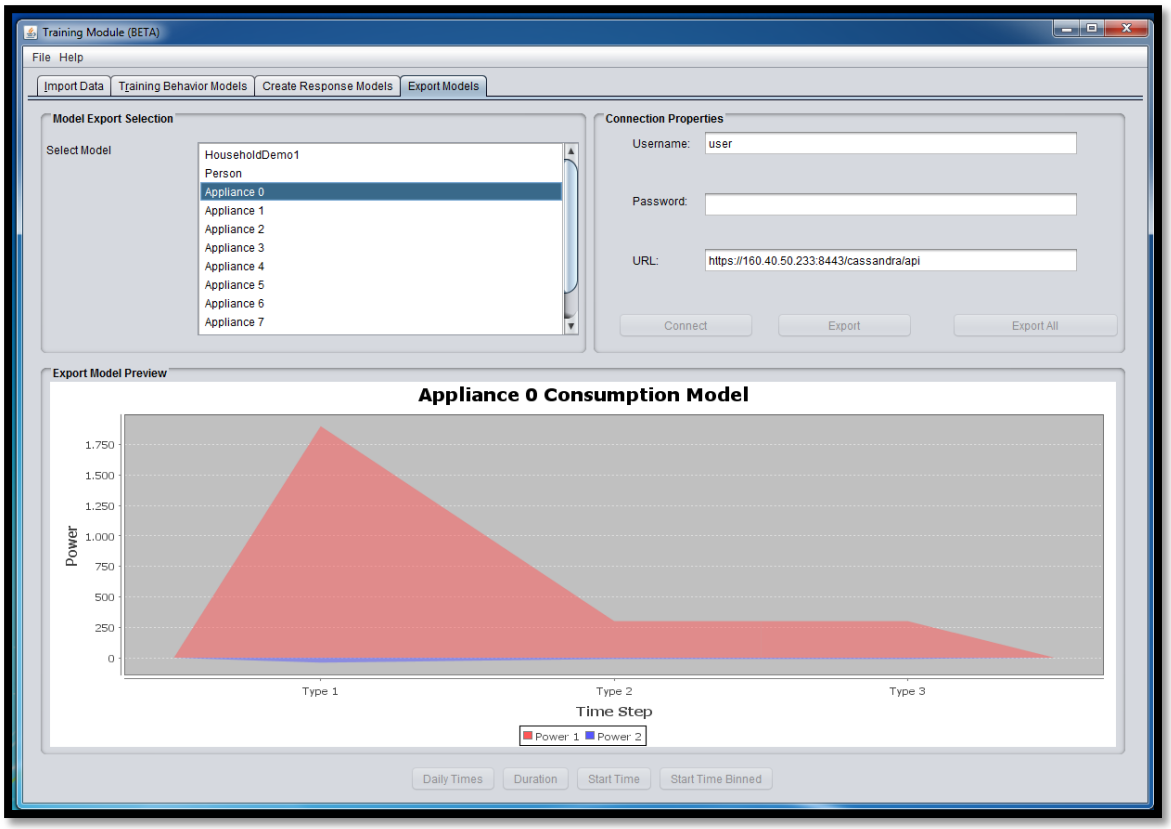

Figure 53 Export Appliance models

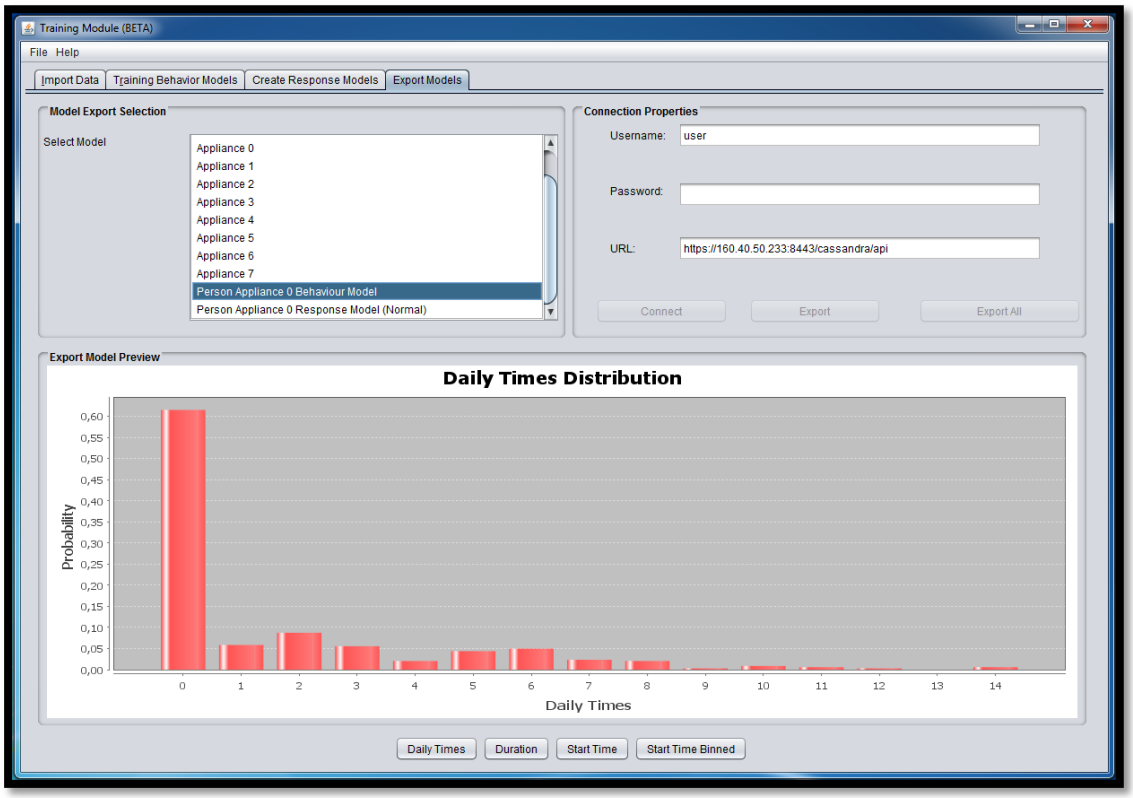

Figure 54 Export Behaviour/Response models

2. The user can type in her credentials and press the *Connect* button.

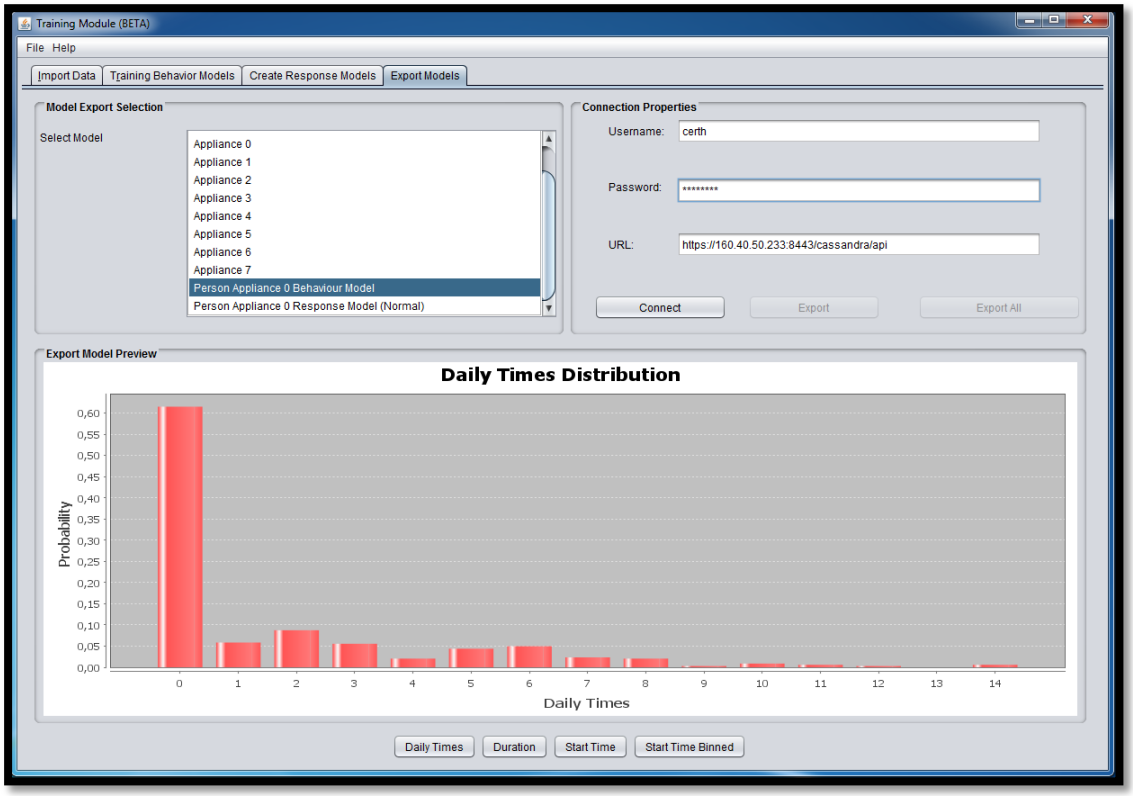

Figure 55 Connect to the Cassandra Server

3. After successfully connecting to the Cassandra Server, the user can choose to either export a single model by pressing the *Export* button, or she can export all the available models as a whole installation, by pressing the *Export All* button.

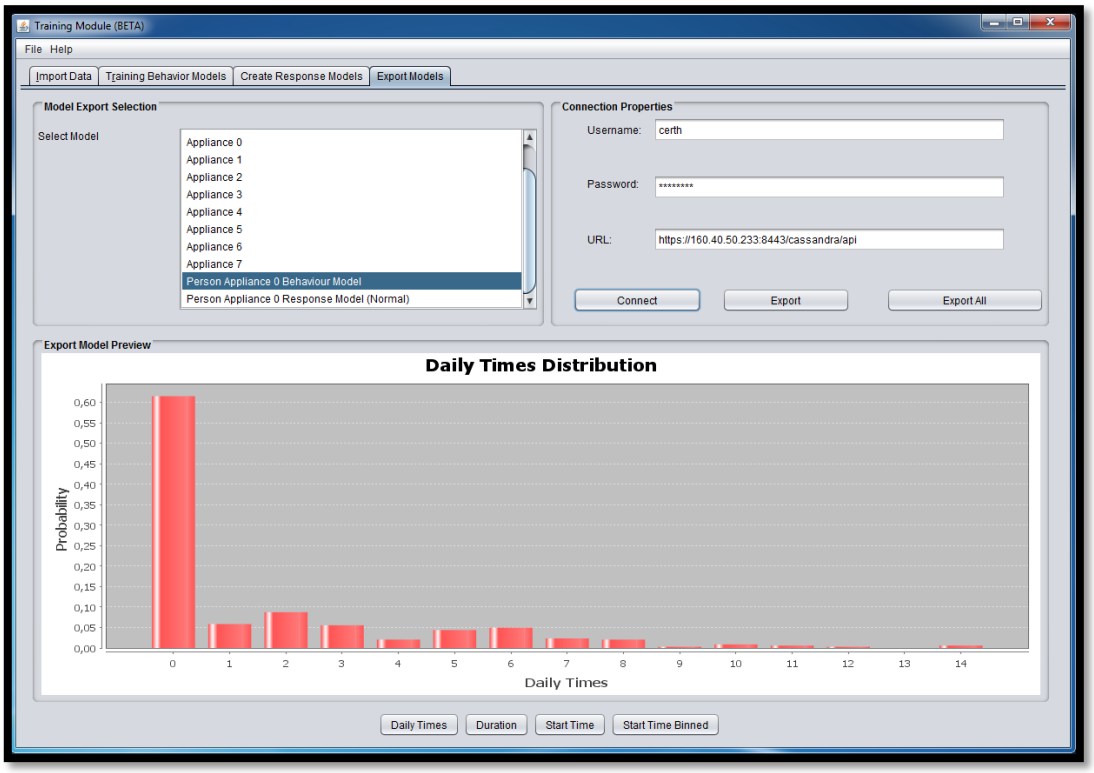

Figure 56 Export (all) models

4 Summary

The aim of the present deliverable is to offer a manual to accompany the early version of the Cassandra platform. The up-to-date electronic version of this document can be found at: https://github.com/cassandra-project/platform/wiki/User-Manual, while the early prototype of the platform is located at: https://github.com/cassandra-project/platform.