A Gaming and Social Networking Platform for Evolving Energy Markets' Operation and Educating Virtual Energy Communities

H2020 ICT-731767

First integration and validation activities of SOCIALENERGY system

Deliverable D5.2

Document Information

Scheduled delivery	30.06.2018
Actual delivery	29.06.2018
Version	Final
Deliverable type	Demonstrator (DEM)
Responsible Partner	ICCS

Dissemination Level

Public

Contributors

Prodromos Makris, Dimitrios Vergados, Nikolaos Efthymiopoulos (ICCS)
John Papagiannis, Vassilis Nikolopoulos (INTELEN)
Jevgenijs Danilins, Boris Irmscher, Andrew Pomazanskyi (NRG)
Atanas Georgiev, Katina Pancheva, Krassen Stefanov (SU-NIS)

Internal Reviewers

Vassilis Nikolopoulos (INTELEN), Manos Varvarigos (ICCS)

Acknowledgements

The research leading to these results has received funding from the EC Framework Programme HORIZON2020/2014-2020 under grant agreement n° 731767.

Table of Contents

List of Figures and Tables

List of Figures

List of Tables

Document History

This deliverable includes a summary of the first S/W integration and validation results of the SOCIALENERGY platform. D5.2 is a prototype (DEM) deliverable. This report aims at guiding and supporting the platform's end users towards setting up, installing and configuring the various operation parameters of SOCIALENERGY platform. Thus, it can also serve as a user manual for dissemination/communication purposes towards SOCIALENERGY products and services' commercialization. D5.2 is the first version representing the work progress done during M13-M18 period. The updated version (i.e. D5.3) will be delivered in M27 (March 2019).

Table 1: Document History Summary

Revision Month	File version	Summary of Changes
30/04/2018	v0.1	Draft ToC circulated to the entire consortium by ICCS.
10/05/2018	v0.2	Final ToC version and ICCS contributions about RAT user manual.
06/06/2018	v0.4	First round of contributions from all partners regarding the user manuals per subsystem and the system as a whole.
22/06/2018	v0.8	Second round of contributions from all partners regarding the S/W integration, testing and validation results.
26/06/2018	v0.9	Final version has been prepared from ICCS for internal review.
29/06/2018	v1.0	INTELEN reviewed the deliverable and made the final amendments before the submission in ECAS portal by ICCS coordinator.

Executive Summary

The SOCIALENERGY architecture is *"modular by design"* in order for all subsystems (i.e. GSRN deployed by INTELEN, GAME deployed by NRG, RAT deployed by ICCS and LCMS deployed by SU-NIS) to be potentially exploitable as stand-alone commercial products in the future. The technical APIs for the interaction between the various subsystems have been appropriately designed in a way that any possible combinations of SOCIALENERGY subsystems to be commercially exploitable in the future (e.g. GSRN with RAT as one single product, GAME with RAT as another one, GSRN-GAME as another one, GSRN-RAT-LCMS as another one, etc.). This strategic decision at the design phase provides the flexibility to the consortium to decide how to prioritize its dissemination, communication and further exploitation activities towards commercialization. Of course, the default choice and ultimate objective of the consortium is to fully integrate all 4 subsystems into one single SOCIALENERGY S/W platform in the context of WP5 work. This way, the SOCIALENERGY product and associated services are expected to be competitive enough in order to enter the liberalized ICT/energy market and be sustainable as a product from a business perspective.

D5.2 is a 'DEMO' deliverable aiming at demonstrating the 1st version of S/W integration and validation activities until M18. However, this document is an accompanying report describing the results of all the related work, which has been undertaken during M13-M18 period in the context of Task 5.2. An integrated DEMO of SOCIALENERGY's 'alpha' version is currently available and will be demonstrated during the 2nd review meeting (Athens, 18/9/2018).

The remainder of this report is structured as follows: Section 1 provides explanatory and comprehensive user manual for the setup, configuration and basic experimentation of each main SOCIALENERGY subsystem. In particular, any interested user (e.g. researcher, utility, market stakeholder, developer, etc.) is able to download the respective open-source S/W prototype and follow the step-by-step guidelines towards operating each subsystem in a stand-alone mode. Mock-up as well as indicative historical datasets are also provided in order for the end user to be able to experiment himself/herself with the basic functionalities of each subsystem.

Section 2 provides a comprehensive user manual for the setup and configuration of the SOCIALENERGY system as a whole. More specifically, once the end user has installed and tested all subsystems, then s/he is able to setup the various web services (i.e. technical APIs) in order to start experimenting with the interactions among the 4 main subsystems. All this documentation has also been uploaded and is publicly available in project's GitHub area[1].

In section 3, we test and validate the SOCIALENERGY system with respect to the use case scenarios defined in WP2 work. It should be noted that only mock-up and indicative historical datasets have been utilized at this phase of S/W testing and validation. Real-life energy and behavioural datasets will be integrated in the platform during the upcoming months. The goal is to release a first 'beta' version of SOCIALENERGY S/W platform by M24

[1] https://github.com/socialenergy-project

and then start the real-life pilot testing. As a result, D5.3 is expected to include testing and validation results taking into consideration input data from real end users. The final step will be to gather all pilot-testing results from diverse real-life experiments and report them in D5.4 at the end of project's lifetime.

1. User Manual per subsystem

In this section, a user manual for each SOCIALENERGY subsystem is provided describing the: i) the purpose, ii) installation steps, iii) user registration and experimentation, iii) indicative navigation and visualization screenshots from the 'alpha' version of the S/W platform.

1.1. GSRN (Green Social Response Network)

1.1.1. Purpose

The purpose of the core GSRN platform is to validate and demonstrate a gaming and social network platform for educating energy consumers and virtual energy communities towards evolving EU energy markets' operation. GSRN platform consists of several S/W modules. '*Data Analytics*' module visualizes all RAT-API outputs and provides a visualized KPIs' dashboard to the users in order to allow them to check their overall performance. *'Energy module'* is connected to the MDM-API and RAT-API in order to visualize real energy consumption curves (ECCs) from users' meters and billing information respectively. *'Gaming profile'* module connects directly to the GAME-API and gets all relevant details from the game, regarding each specific user. User gets badges, leader board, performance, stages, points and all available GAME-API inputs. Virtual marketplace component bridges the gap between energy consumers and multiple other market stakeholders related to the energy efficiency sector. Finally, *'Socialties'* module is also working at the backend and is used to get user's social network information, as the user logs in the system. All these sub-systems are modular and accessible through GSRN.

1.1.2. Installation steps

The following steps have been tested on Ubuntu 16.04. Please adapt accordingly for other distributions/OSs.

1. Clone the repository into a directory on the local computer, and enter the directory. If git is not present, it should be installed before continuing to the next step.

git clone https://github.com/socialenergy-project/gsrn.git
cd gsrn/dbm

2. In order to setup GSRN, we must install the appropriate database with all the necessary tables:
Database name: socialEnergy
The tables needed are the following:

- *Actions*
- *AdminUserGroups*
- *DashboardFormat*
- *DashboardFormatPlaces*
- *DashboardFormation*
- *GameUserActions*
- *LcmsCreateAccount*
- *LcmsCreatePlanResuly*

- *MarketplaceProductsViews*
- *MemberUserGroups*
- *ProductNum*
- *Questionnaire*
- *RecomedationsTips*
- *UsersCredentials*
- *chat*
- *logOutAuth*
- *marketPlaceProductTraffic*
- *printNotToGrousAdmin*

3. Create an empty database and name it socialEnergy:
(command on mysql shell: create database socialEnergy)

4. Install to mysql the following sql script:
(command on mysql shell:
mysql -u username –-password=your_password database_name < file.sql)
socialEnergy.sql (Path: gsrn/dbm)

4. Verify that you have created the required database along with the needed tables:
(Command: Use socialEnergy;
or
connect socialEnergy;
show tables;
)

5. Install Software
Tools: php (php -v PHP 5.6) – framework : Codeigniter

a. cd gsrn/gsrn_software

b. Open the following file application / config / database.php
add your server database credentials.

c. Install Memcache extension for php

d. Open the following file application / config / config.php

Find config variable: $config['base_url'] add the path where you have place GSRN in your web server (apache or nginx) the absolute path.
e.g:
http://localhost/gsrn/CodeIgniter/

If you have followed all the steps, the next thing to do is to open your favorite browser and navigate to the installation path.
e.g:
http://localhost/dashboard2/CodeIgniter/index.php/Login
If everything is correct, the login screen will be up.

You can use as credentials:
username: socialenergy
password: 123456

1.1.3. Registration and mock data experimentation

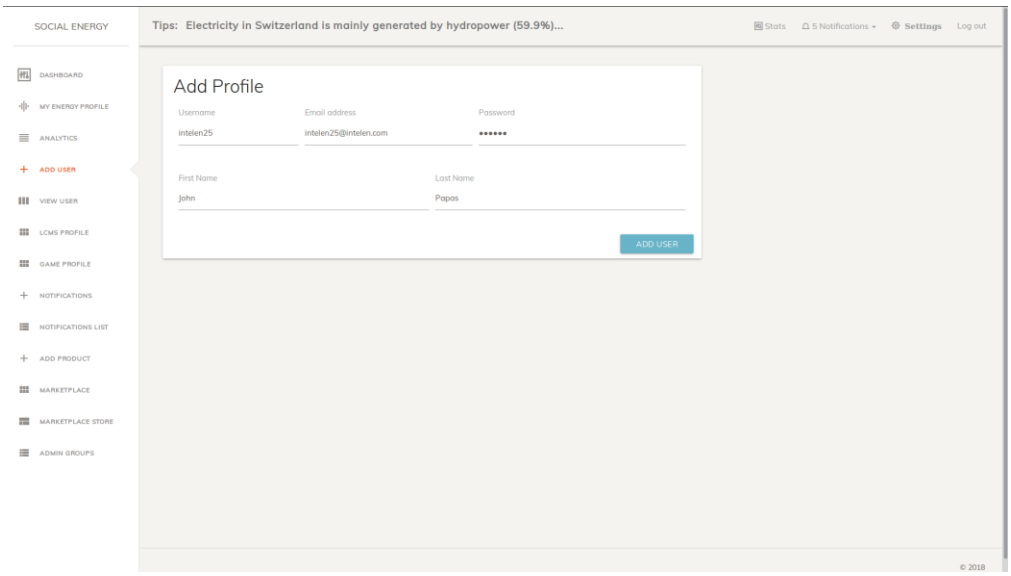
In order to use GSRN platform as a standalone platform or in combination with other components, the database and the software needs to be installed and configured following the installation instructions.

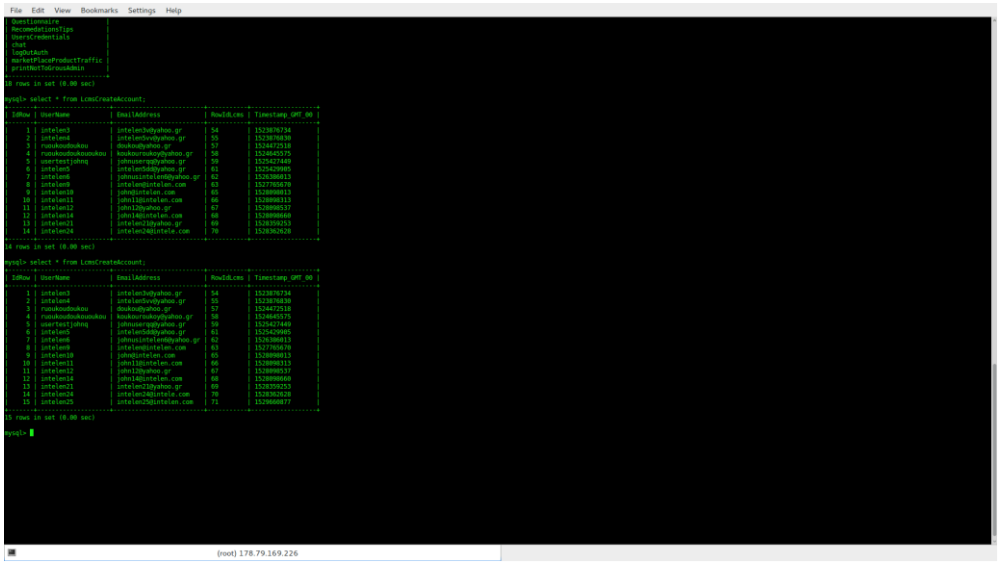
Create User
1. Open your browser and redirect to GSRN, login to the platform

use as credentials:
username: socialenergy
password: 123456

2. Select from the main left menu option 'Add user'
Complete the form and press *ADD USER*.
When this is done, the oauth2 has being updated with this new user and LCMS has created a new account as well.

3. Login to mysql.
Command : mysql -u userName -p
add password
use socialEnergy;
execute this query:
select * from LcmsCreateAccount;





When a new account on LCMS is created, GSRN saves the result from API interaction to be able at any time to confirm the success of the transaction. If you see in the list your username, this means that everything went well (see screenshots above).

Confirm that the new account has being added to oauth2:
command : use my_oauth2_db;
command : show tables;
*command: select * from oauth_users; or if you have many records*
*select * from oauth_users where username='intelen25';*
(you can replace intelen25 with your username)

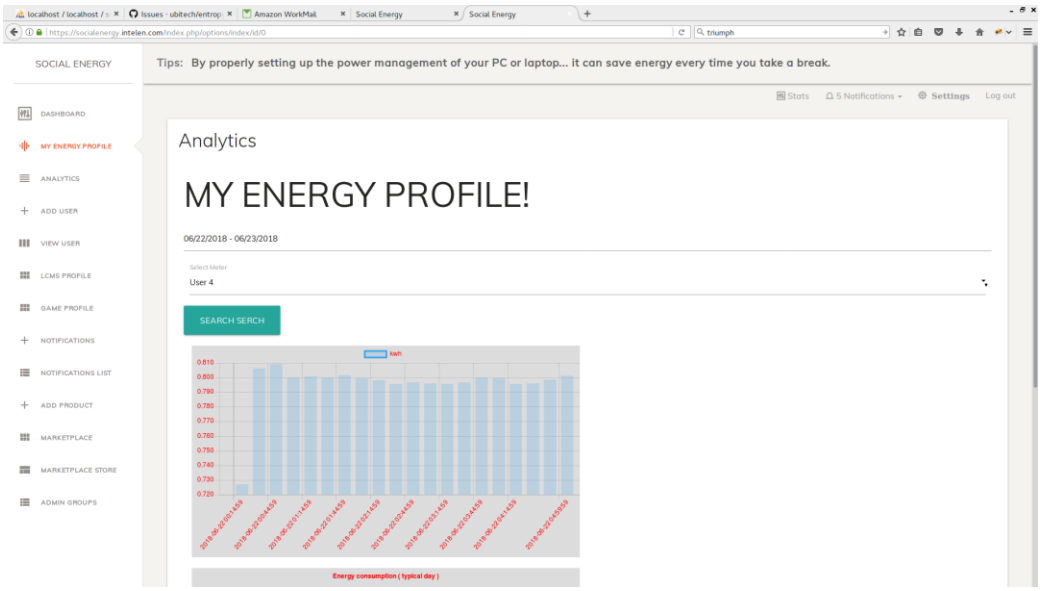
If everything went well, you will see your record there. Having that said, it means oauth2 is updated with the new record and can support user authentication with RAT or LCMS using the corresponding authentication controllers.

1.1.4. Navigation and visualization

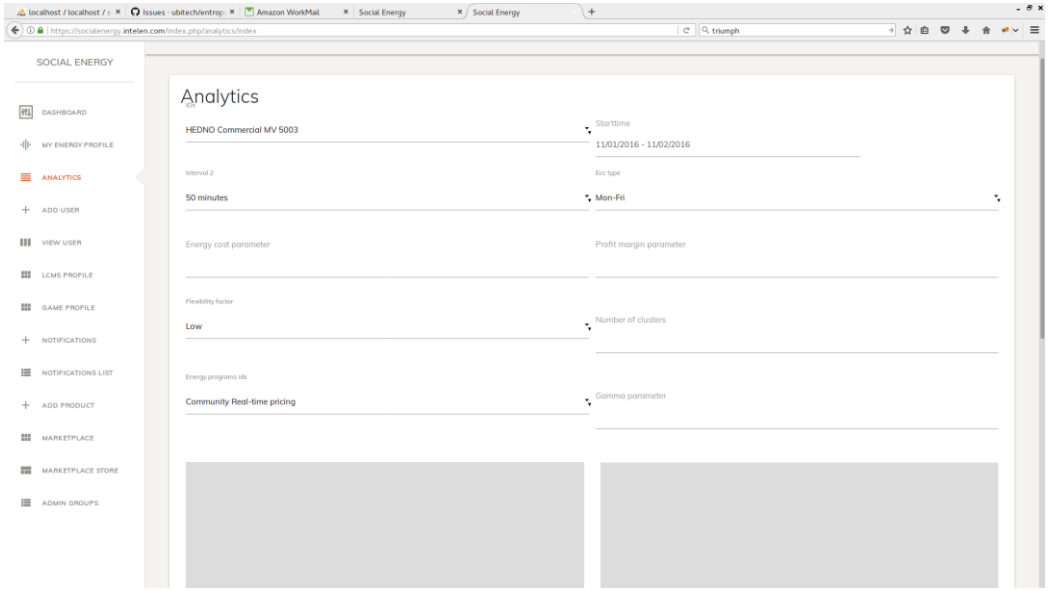
A working instance of GSRN can be found at:
https://socialenergy.intelen.com/index.php/login/

The most important functionalities of the core GSRN platform are the following:

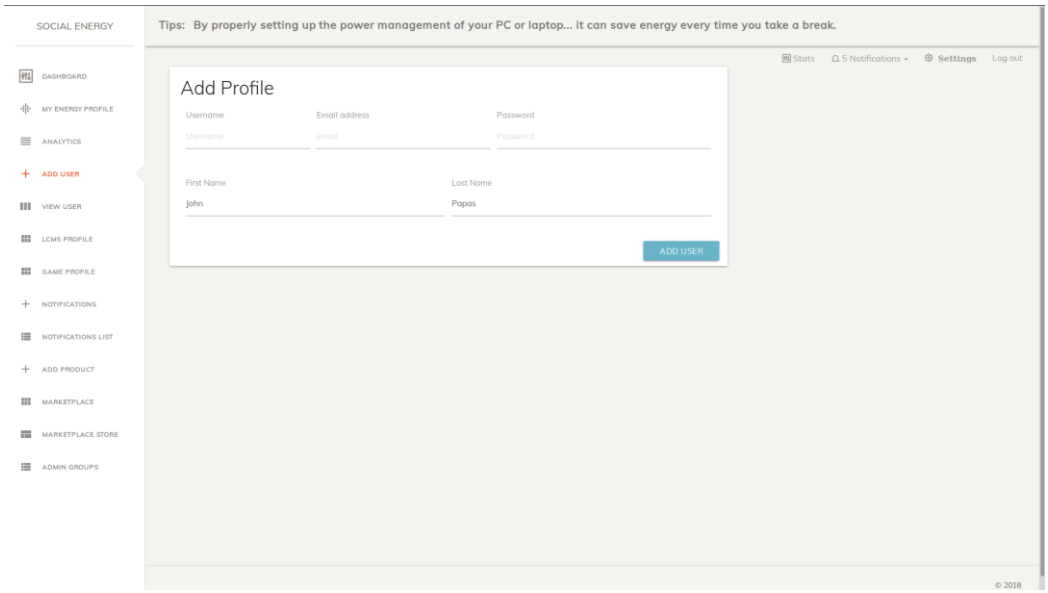
'MY ENERGY PROFILE' tab:
https://socialenergy.intelen.com/index.php/options/index/id/0



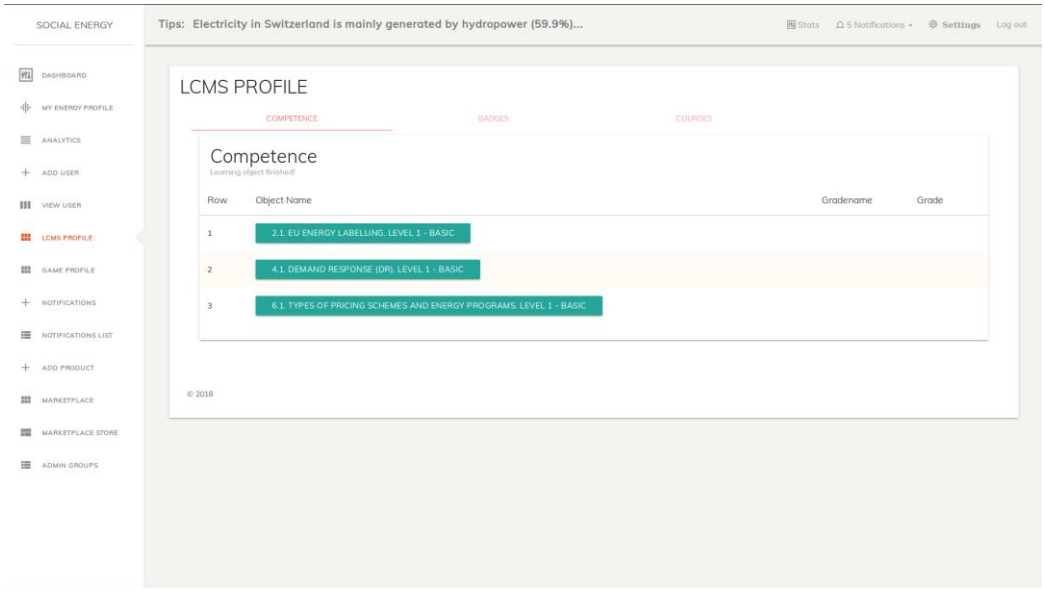
'ANALYTICS' tab:
https://socialenergy.intelen.com/index.php/analytics/index



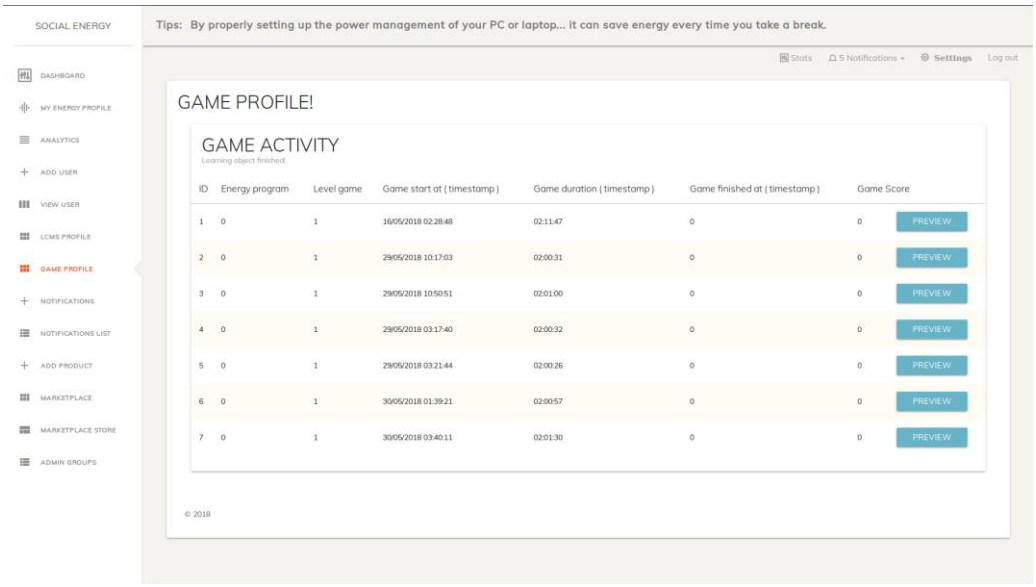
'ADD USER' tab:
https://socialenergy.intelen.com/index.php/adduser/index



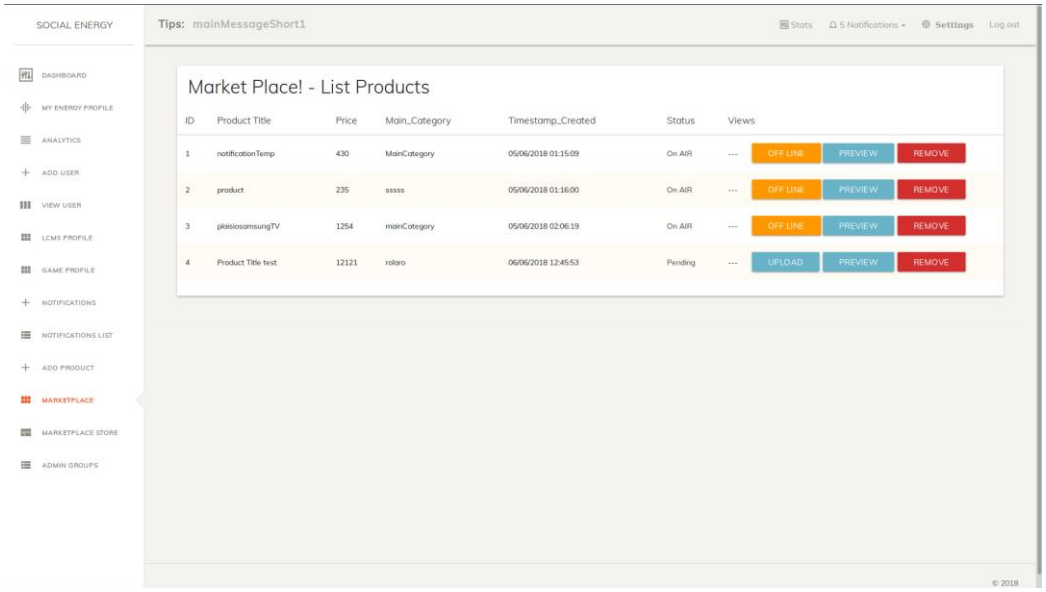
'LCMS PROFILE' tab:
https://socialenergy.intelen.com/index.php/Lcmsprofile/index



'GAME PROFILE' tab:
https://socialenergy.intelen.com/index.php/options/index/id/2



'MARKET PLACE' tab:
https://socialenergy.intelen.com/index.php/marketplace/index



1.2. RAT (Research Algorithms' Toolkit)

1.1.4. Purpose

The purpose of this S/W component is to allow administrators or privileged RAT users (i.e. electric utility users) to evaluate the performance of various energy programs, algorithms for the EC creation and dynamic adaptation as well as common types of data analytics services such as automated profiling, searching and recommendation. Thus, RAT is a business analytics and intelligence tool, which helps the system administrator to automatically analyze various business/strategy 'what-if' scenarios by running parameterized system-level simulations.

The portal is available at https://rat.socialenergy-project.eu.

1.2.2. Installation steps

The following steps have been tested on Ubuntu 16.04 and Ubuntu 18.04. Please adapt accordingly for other distributions/OSs.

1. Clone the repository into a directory on the local computer, and enter the directory. If `git` is not present, it should be installed before continuing to the next step.

   git clone https://github.com/socialenergy-project/rat.git
   cd rat/

2. Install ruby version 2.4.1, using *rbenv*. Installation instructions for *rbenv* may be found here https://github.com/rbenv/rbenv#installation

3. Install postgres, create database user for rat, and setup a password:

   sudo apt install postgresql-common
   sudo apt install libpq-dev

   sudo -u postgres createuser rat -s
   sudo -u postgres psql
   postgres=# \password rat

4. Setup the environment variables for the project. First create a *.env* file, using the provided sample:

   cp -i .env.sample .env

   and then edit the file to provide the appropriate values.

   SECRET_KEY_BASE=f24...
   RAT_DATABASE_PASSWORD=aCZ..

   SMTP_USERNAME=user@gmail.com
   SMTP_PASSWORD=pass

   - The value for *SECRET_KEY_BASE* variable is obtained by executing *rails secret.*
   - The value for *RAT_DATABASE_PASSWORD* variable is the password set for user `rat` in the previous step.
   - The values for *SMTP_USERNAME* and *SMTP_PASSWORD* are used for connecting to gmail to send emails. Different mail servers may be added by editing file *config/initializers/smtp_settings.rb*

5. Install the required gems:

   bundle install

6. Create the project database

   rails db:create
   rails db:migrate

7. Now you can start the server with the command

   rails s

   You can then visit the site by opening a browser at http://localhost:3000/

8. In order to be able to run the algorithms, you need to install the "pricing algorithms" submodule https://bitbucket.org/socialenergy-iccs/crtp_prtp_rtp , in a direct subdirectory of this repository, with the default name. Ensure that the sub-module is installed correctly, by following the instruction in the corresponding README file.

9. Ensure that the tests pass, with:

   rails test

1.2.3. Registration and mock data experimentation

In order to use the RAT platform as a standalone platform, an admin user must be created, and the database needs to be initialized with consumers and other objects. Finally, energy consumption data for the individual energy consumers should be added to the database. Please follow the steps below:

Register admin user

1. Start the server by running command *rails s* from the installation directory
2. Open a browser window at location http://localhost:3000/ (or https://rat.socialenergy.eu)
3. Register a new user using the *Sign up* link (or navigate to http://localhost:3000/users/sign_up – https://rat.socialenergy-project.eu/users/sign_up).
   Set an email and a password and submit the form.
4. To make the user an administrator, navigate to the project directory, and execute:

   rails console

   In the prompt that appears execute the command:

   User.find_by(email: 'YOUR_EMAIL').add_role :admin

Database initialization

1. Decompress the file with consumption data:

bunzip2 --keep db/initdata/DataPoint.csv.bz2

A file named *db/initdata/DataPoint.csv* should be created.

2. Run the script to seed the database with initial data:

   rails db:seed

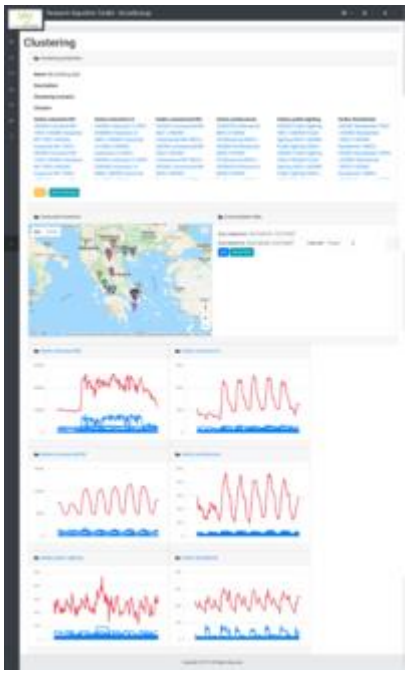
After this command, navigating to https://localhost:3000/, you should be able to see all consumers, with consumption data for the dates from 1/1/2015 to 30/9/2016.

1.2.4. Navigation and visualization

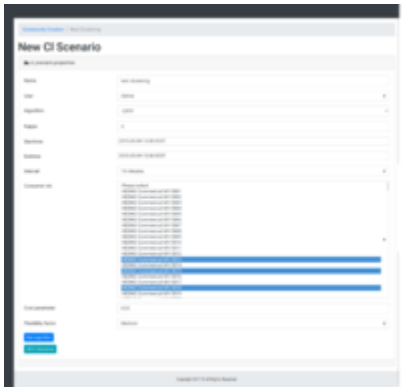
A working demo the module may be found at https://rat.socialenergy-project.eu/, where anyone can register an account.

The most important functionalities of the RAT module are the following:

1. Consumption data visualization. Visit for example the following links:
   - http://localhost:3000/consumers (or https://rat.socialenergy-project.eu/consumers)
   - http://localhost:3000/consumers/5001?start_date=2015-05-02T14%3A57%3A14.333%2B03%3A00&end_date=2015-05-09T14%3A57%3A14.333%2B03%3A00&interval_id=2 (or https://rat.socialenergy-project.eu/consumers/5001?start_date=2015-05-02T14%3A57%3A14.333%2B03%3A00&end_date=2015-05-09T14%3A57%3A14.333%2B03%3A00&interval_id=2)
   - http://localhost:3000/communities/105 (or https://rat.socialenergy-project.eu/communities/105)
   - http://localhost:3000/clusterings/1 (or https://rat.socialenergy-project.eu/clusterings/1)



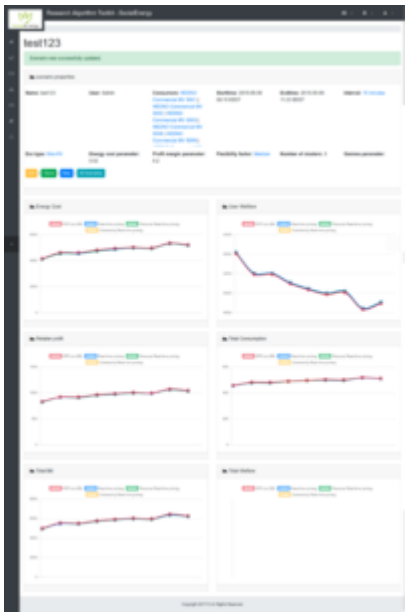
2. Clustering algorithms:
   - To view the list of created clusterings, visit http://localhost:3000/clusterings/1
   - To create a new clustering, by executing a clustering algorithm, visit the link: http://localhost:3000/cl_scenarios/new (or https://rat.socialenergy-project.eu/cl_scenarios/new)
     The dialog looks like this:



   - To view the list of clustering scenarios, in order to view their parameters and/or edit them, you can visit http://localhost:3000/cl_scenarios (or https://rat.socialenergy-project.eu/cl_scenarios)

3. Evaluation of energy programs (i.e. pricing algorithms):
   - To view the list of pricing scenarios, visit: http://localhost:3000/scenarios/ (or https://rat.socialenergy-project.eu/scenarios)
   - To create a new scenario to test the performance of each pricing algorithm under different conditions, visit http://localhost:3000/scenarios/new (or https://rat.socialenergy-project.eu/scenarios/new)
   - After the scenario is created, you may see it in the url http://localhost:3000/scenarios/1 (or https://rat.socialenergy-project.eu/scenarios/621).

   The results are depicted like in the screenshot below:



4. Recommendation engine:
   - To view all the recommendations that have been created visit: http://localhost:3000/recommendations/ (or https://rat.socialenergy-project.eu/recommendations)
   - To create a new recommendation visit: http://localhost:3000/recommendations/new
   - To preview a recommendation visit: http://localhost:3000/recommendations/1 (where the final number is the id of the recommendation)
   - To send the recommendation, click the button `Send`. The users that are associated with the consumers in the recommendation will be notified with a notification like in the screenshot below:

1.3. GAME

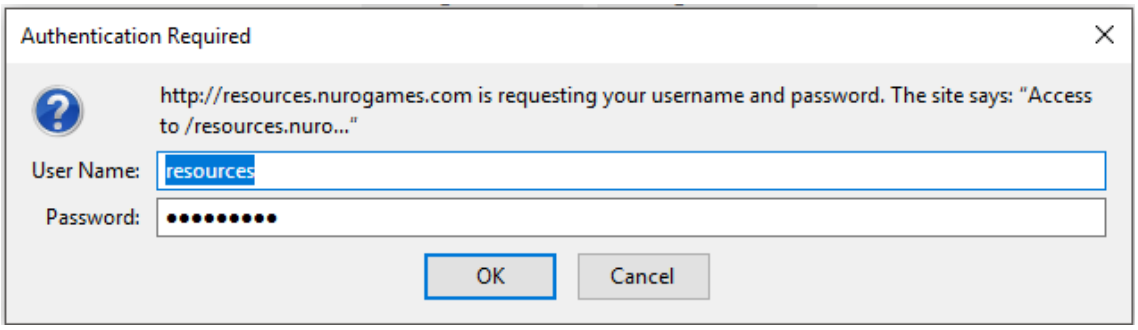
1.1.5. Access & Log In

The SOCIALENERGY Game Demo can be played in actual web browsers (i.e. chrome, firefox).

An internet connection is required. To get access, the following URL has to be typed in:

http://resources.nurogames.com/socialenergy/Demo2018-06-01/

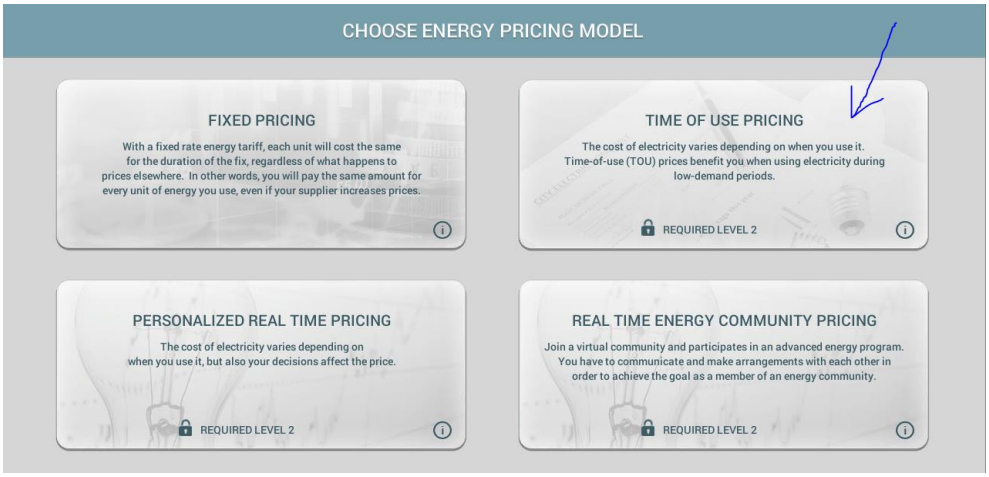
After the URL has been loaded, the browser will pop up a window where the username and a password have to be typed in:



After a successful login the loading screen is shown until all data is loaded.

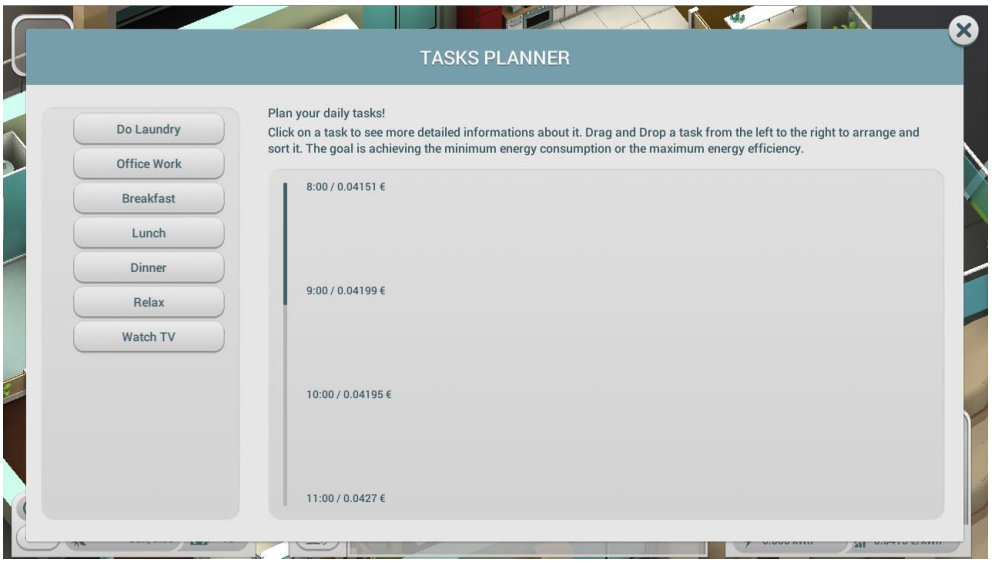
1.1.6. Avatar editor and use case selection menu

After the game has been loaded, the Avatar Editor is shown. For the Demo Version of the game, the Avatar can NOT be customized. To go on with the game, the 'Start' Button has to be clicked with the mouse. After the Avatar Editor has been shown, the Use Case Selection Menu is shown. Here, the different Energy Programs are shown. For the Demo Version of the game, only the Time Of Use (ToU) Pricing Program can be selected.



1.1.7. Tasks planner

After the Use Case Selection Menu has been shown, the Task Planner window is shown. Here, the player can plan and arrange his/her daily 'to dos or else jobs' taking into consideration the Energy Price that changes during the progress of time.



To get more information for each job, each job can be clicked and a pop up window shows more details. In the Demo Version, the content of the Job Information Pop Ups is just placeholder content. The pop up can be hidden again by using the close button.
To arrange the jobs on the Timeline, they can be dragged and dropped from the left to the right by clicking and holding the mouse.

On the Time Table, the Energy Prices are displayed related to the Time, so the player can figure out the best starting times for each job to reach the best relation between energy efficiency and convenience for the avatar. If the player has finished the plan, s/he can use the 'close' button to start the day.

1.1.8. Virtual Home and Avatar

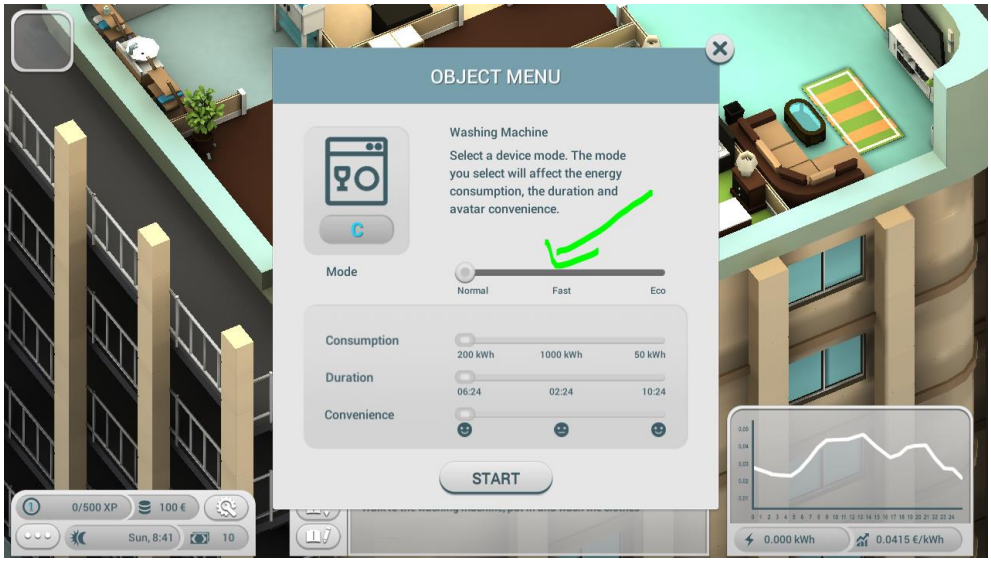
Now, the player has to control the movements of the avatar in the virtual home, by clicking on the target spots. The avatar will automatically move to the target spot. In the Demo Version, the animations of the avatar are not final and polished.

1.1.9. Execute Jobs

To execute a job, the player has to activate a job in the daily job list. In the Demo version, the Job “Do Laundry” is pre-selected (other jobs are NOT available in the Demo Version), so the player can start directly. At the Bottom/Center of the screen, the next required step of the job is displayed.

Now, the player has to execute each job by moving the avatar to the required objects in the virtual home. The target object that the player has to interact with next are marked with an orange exclamation mark. If the player clicks on the object, the avatar will move to it and start the interaction.

Depending on the type of the object, the interaction will start automatically or a pop up where the player can set up the device options is shown. For example, the player can choose one of different modes and see how the avatar convenience, the energy costs and the duration of use are affected.



During the Time of Use Pricing mode, the user has to keep in mind that the energy price changes depending on the day time. Therefore, the price curve is displayed at the bottom right of the heads-up display.

After the player has set up the device mode he can press the start button to start the interaction. After this, a status bar shows the progress of the interaction. The design of the status bar is not final in the demo version.

1.1.10. Finished Jobs

If the player has executed all steps of a job, a "Finished Job" pop up will show the energy consumption, the costs and the convenience that the player has reached. The player is also rewarded with XP and Credits.

The rating and scoring will be calculated and shown at the end of a virtual day. That is not included in the demo version, so if the user presses the "Ok"-Button the Demo Version is finished. To restart the game demo, the "reload"-Button of the browser should be pressed.

1.4. LCMS (Learning Content Management System)

1.4.1. Purpose

LCMS is the subsystem, where the users educate themselves about good practices on energy efficiency. LCMS interacts with GSRN. Thus, the latter can provide recommendation services to the user according to the educational content that is mostly keen on watching next based on her/his current educational profile and experiences in both SOCIALENERGY's real and virtual worlds. The role of the LCMS is important, because it provides the user the opportunity to better comprehend the new concepts in the liberalized smart grid markets and inter-relate the "lessons learned" from the GAME with the real-life conditions in order to be able to efficiently interact with her/his electric utility/retailer company. The LCMS subsystem is built on top of Moodle (https://moodle.org/). The LCMS is available at http://socialenergy.it.fmi.uni-sofia.bg

1.4.2. Installation steps

The following steps have been tested on Ubuntu 16.04. Please adapt accordingly for other distributions/OSs.

1. Install and configure LCMS

Basic Requirements

- You will need a working Apache web server, a database (e.g. MySQL) and have PHP 7 configured.
- Moodle requires a number of PHP extensions. However, Moodle checks early in the installation process and you can fix the problem and re-start the install script if any are missing.

2. Create MySQL Database

Login to the MySQL server as *root* and create a user and database for the Moodle installation

```
mysql -u root -p
mysql> CREATE DATABASE moodle;
mysql> GRANT ALL PRIVILEGES ON moodle.* TO 'moodleuser'@'localhost' IDENTIFIED BY 'PASSWORD';
mysql> FLUSH PRIVILEGES;
```

Notes: Don't forget to replace 'PASSWORD' with an actual strong password. Save this password you use for the Moodle user, since you will need it later in the install.

3. Get Moodle from GitHub repository

Change the current working directory and clone Moodle from the official GitHub repository:

```
cd /var/www/html/
git clone -b MOODLE_34_STABLE git://git.moodle.org/moodle.git
```

Create a directory for the Moodle data:

```
mkdir /var/moodledata
```

Set the correct ownership and permissions:

```
chown -R www-data:www-data /var/www/html/moodle
chown www-data:www-data /var/moodledata
```

4. Configure Apache Web Server

Create Apache virtual host for your domain name with the following content:

```
nano /etc/apache2/sites-available/yourdomain.com.conf

        ServerAdmin admin@yourdomain.com
        DocumentRoot /var/www/html/moodle
        ServerName yourdomain.com
        ServerAlias www.yourdomain.com

        Options Indexes FollowSymLinks MultiViews
        AllowOverride All
        Order allow,deny
        allow from all
```

```
ErrorLog /var/log/httpd/yourdomain.com-error_log
CustomLog /var/log/httpd/yourdomain.com-access_log common
```

Save the file and enable the virtual host:

```
a2ensite yourdomain.com

Enabling site yourdomain.com.
```

To activate the new configuration, you need to run:

```
service apache2 reload
```

Finally, reload the web server as suggested, for the changes to take effect:

```
service apache2 reload
```

5. Follow the on-screen instructions and complete the installation

Now, go to http://yourdomain.com and follow the on-screen instructions to complete the Moodle installation. For more information on how to configure and use Moodle, you can check the official documentation.

6. Enabling and configuring OAuth2 authentication

The OAuth2 authentication plugin enables users to log in LCMS (Moodle) using their GSRN account via button on the LCMS login page. You will need to obtain OAuth 2.0 credentials (client ID and client secret) from GSRN platform.

Create and configure new OAuth 2 service:

1. Go to 'OAuth 2 services' in *Site administration > Server* and click the button "*Create new custom service*".
2. Enter the client ID and client secret, make sure *'Show on login page*' is checked, and require email verification is unchecked, and then save changes.
3. Configure the endpoints for the issuer:
    - authorization_endpoint
    - token_endpoint
    - userinfo_endpoint

7. Installing SOCIALENERGY local plugin

This plugin provides Web Service API that enables the integration between GSRN platform and LCMS.

8. Get SocialEnergy plugin from GitHub repository

Change the current working directory and clone the plugin from the project's official GitHub repository:

```
cd /var/www/html/moodle/local
git clone https://github.com/socialenergy-project/moodle-local_socialenergy.git socialenergy
```

The remaining installation is taken care of by LCMS by clicking on *Site Administration > Notifications*.

9. Associated plugins

There is currently one plugin that requires this integration (for details about more associated plugins, please see the final version of the user manual at the end of SOCIALENERGY S/W integration task):

1. Course dedication

10. Enabling web services

A. Access *Site administration > Advanced features*
B. Check '*Enable web services*' then click '*Save Changes*'

11. Enabling protocols

A. Access *Site administration > Plugins > Web services > Manage protocols*
B. Enable REST protocol

12. Creating a service

A. Access *Site administration > Plugins > Web services > External services*
B. Click Add new custom service
C. Check '*Authorised users only*'
D. Enter a name and check *Enabled*
E. Click the button '*Add service*'

13. Adding functions to the service

The newly created service is currently empty and doesn't do anything. Web service functions need to be added.

A. Click 'Add functions' link
B. Select local_socialenergy_user_create_user, local_socialenergy_competency_create_plan, local_socialenergy_user_get_profile functions and click the 'Add functions' button

14. Authorise specific users

A. Access *Site Administration > Plugins > Web services > External Services*
B. Select *Authorised users* link (the service must have been set as Authorised users only in the *Edit* link)
C. Select appropriate user with administrative permissions and click *Add*

15. Create a token

Token Authentication is a standard form of authentication for web services. The LCMS service identifies requests via a unique token and executes requests based on the permissions for that account.

A. Access *Site Administration > Plugins > Web services > Manage tokens*
B. Click on *Add*
C. Select the created user and service
D. Click on save changes

1.4.3. Registration and mock data experimentation

In order to use the LCMS platform as a standalone tool, the competency framework and course data should be imported. Please login as administrator and complete the steps below:

1. **Importing SOCIALENERGY Competency Framework**
   A. Download the csv file from https://github.com/socialenergy-project/moodle-mock_data/blob/master/SEFR.csv
   B. Access *Site administration > Competencies > Import competency framework*
   C. Select CSV comma delimited
   D. Confirm the column mappings on next screen
   E. Finish the import

2. **Configuring SOCIALENERGY local plugin**
   A. Access *Site Administration > Plugins > Local plugins > Social Energy Custom Services*
   B. Select previously imported competency framework
   C. Fill-in default ILP's name
   D. Fill-in the URL address to which user should be redirected after log out
   E. Click on save changes

3. **Importing courses**
   A. Download courses' archives (.mbz files) from https://github.com/socialenergy-project/moodle-mock_data
   B. Access *Site administration > Front page settings > Restore*
   C. Upload the .mbz file and click Restore
   D. Confirm - Check that everything is as required then click the Continue button
   E. Destination - Choose whether the course should be restored as a new course or into an existing course then click the Continue button
   F. Settings - Select activities, blocks, filters and possibly other items as required then click the Next button
   G. Schema - Select/deselect specific items and amend the course name, short name and start date if necessary then click the Next button
   H. Review - Check that everything is as required, using the Previous button if necessary, then click the 'Perform restore' button
   I. Complete - Click the continue button
   J. Repeat steps above for the other courses

1.4.4. Navigation and visualization

A working demo may be found at https://socialenergy.it.fmi.uni-sofia.bg, where anyone can register an account.

All courses in LCMS include several types of learning activities related to the three main aspects of the competency framework: knowledge, skills, responsibility, and autonomy. These major types of learning assets are:

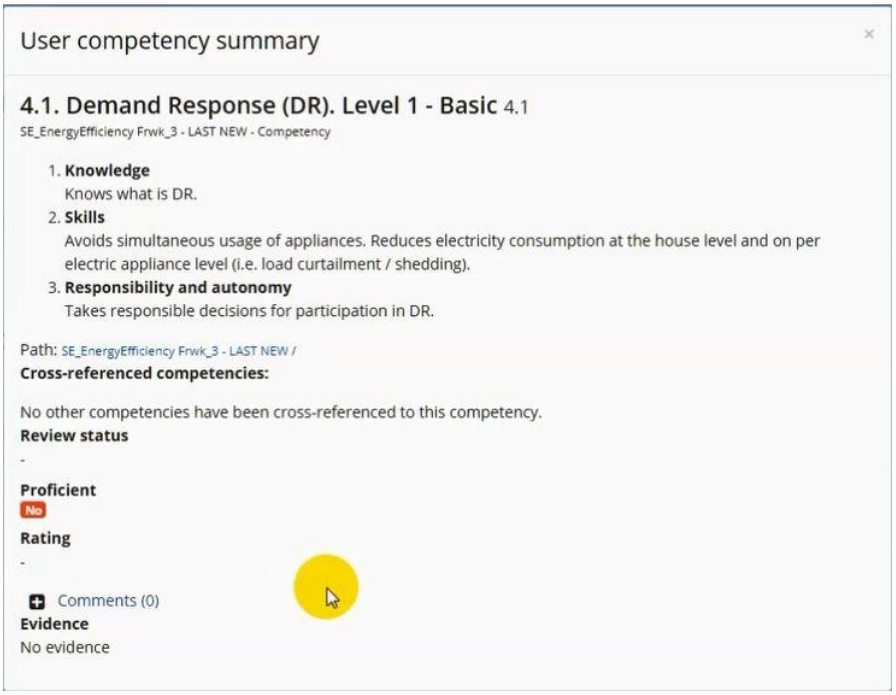
- Readings pdf - Written material intended to be read, that present educational content primarily through text, but also contain appropriate graphics, diagrams, illustrations
- Presentations – summary of reading materials
- Videos - represent a short portion of educational content with examples and information on how to complete a certain task

- Lessons – including practical information
- Quiz – a set of questions and/or problems used as a means of evaluating the abilities, attitudes, skills

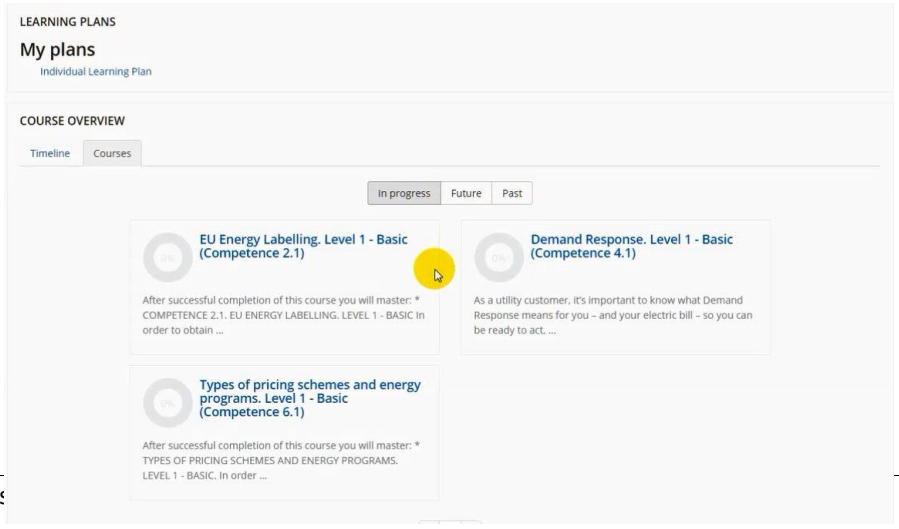
Most important functionalities among all available in LCMS are the following:

Individual Learning Plan Visualization

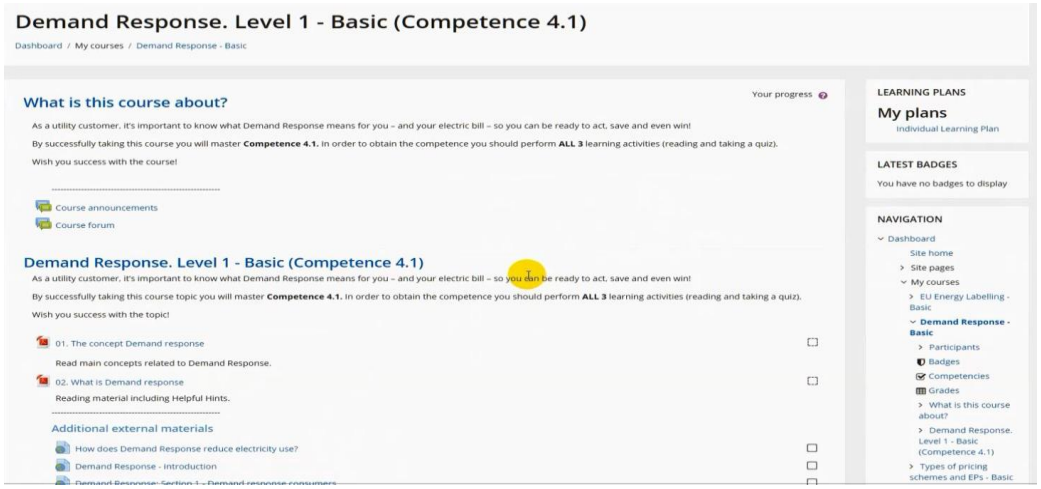
User competence summary



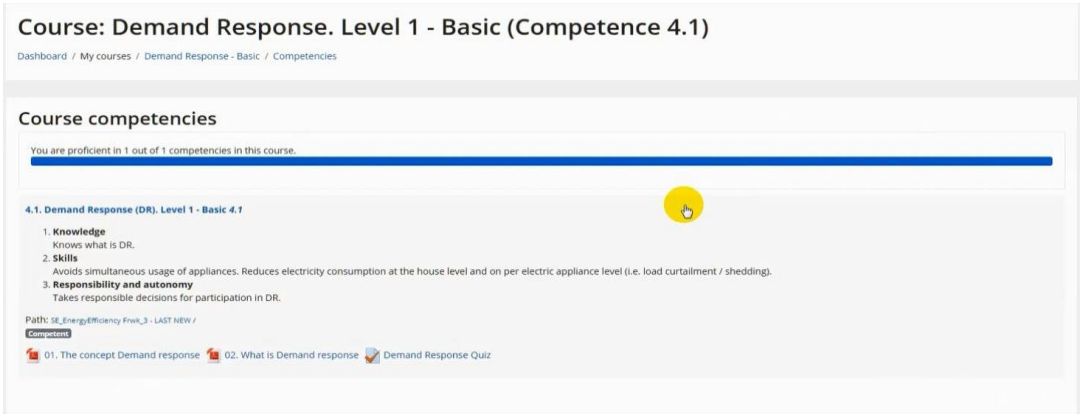
Courses In Progress, Past and Future Overview



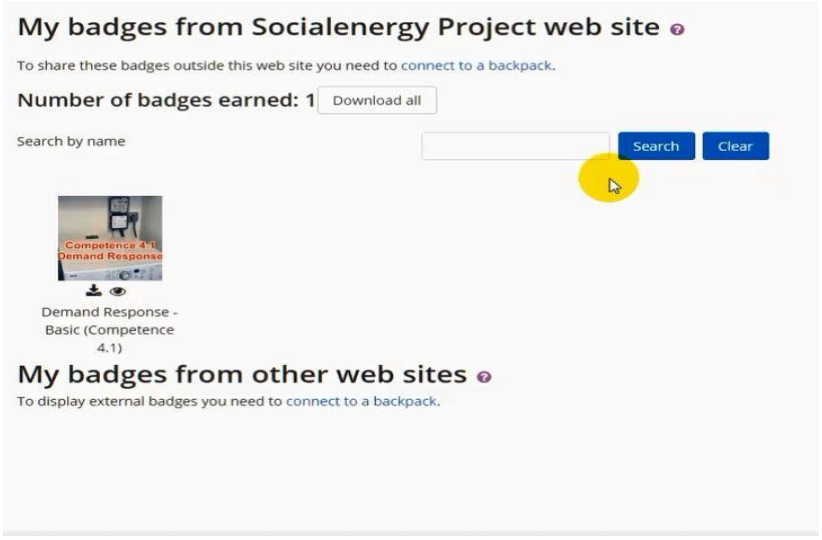
Course overview



Course Proficiency Overview



Awarded upon competence acquirement badges



2 User manual for the SOCIALENERGY platform

This section provides a user manual for setting up the web services (i.e. Application Programming Interface - API) for the interconnection of the 4 subsystems, which have already been described in section 1. After this procedure, the SOCIALENERGY system as a whole will be set up.

2.1. Login in the SOCIALENERGY S/W platform

See below the steps to setup OATH2:

1. Create an empty database, name it my_oauth2_db:
(command on mysql shell: create database my_oauth2_db)

2. Install to mysql the following sql scripts:
my_oauth2_db.sql

3. We need to create an OAUTH2 client:
Add to table oauth_clients values to the following columns:

- *client_id,*
- *client_secret,*
- *redirect_uri*
- *grant_types*
- *scope*
- *user_id*

you want to authenticate with.

e.g:
client_id: intelen
client_secret: secret,
redirect_uri: path of callback controller of client,
grant_types: authorization_code refresh_token password,
scope: openid profile email

3. Create oauth_users, how?

1. Redirect your browser to the path where you have installed the platform,

2. Login to the platform,

3. Select Add User, create a new user,
By doing that you have created a new user to lcms and to OATH2 as well.

4. How to setup oath2 software?

Open gsrn / application / config / config.php

Find config variable: $config['base_url_oauth2'], add the path where you have placed OAUTH2 in your web server (apache or nginx) the path that is accessible by the outside world.

5. Unzip the contents of the folder GSRN_OATH2_SOFTWARE
and place it to your desired path on your web server.

The controllers that you need are the following:

removesession.php = (You need to post username, to terminate session)
authorize_x.php?
token_x.php
userinfo_x.php
usercredClientUser.php

Depending on what functionality you want to expose to a third party, you will provide the appropriate controller. For sure, you have to distribute authorize_x and userinfro_x contoller to your partners.

For example: https://socialauth.intelen.com/removesession.php

2.2. Setup of Web Services

The chosen architecture of GSRN for the web services are micro – services. Since you have downloaded and installed the software, there is no need for extra installation for the services.

The main controller for exporting the service to the outside world is the Webservices controller, you can locate him under the controller folder.

Includes:

- MDM-GSRN service.
- GSRN-RAT service.
- GSRN-GAME service.
- GSRN-LCMS service.
- GSRN-Marketplace service.
- GSRN-GAME Authentication service

3 Initial version of validation activities

This section reports the initial version of SOCIALENERGY system's testing and validation activities with respect to the 2 use cases and 8 use case scenarios, which have been defined in WP2 (see D2.2).

3.1. Use Case no. 1 - SOCIALENERGY's Real World (GSRN platform)

3.1.1. Scenario 1A – Behavioural Demand Response

A behavioral DR approach will be followed in this scenario, but it will be quite restricted to voluntary participation without any direct benefits or incentives for the participants. However, participants were asked to imagine that they will be rewarded after performing a desired behavior. Namely, the reward will be the hypothetical savings derived from reduced charges during a DR event, when scope will be energy increase or reduced charges during non-DR event hours, when the scope will be energy reduction.

The Demand Response Tool and technology has been developed by INTELEN inside GSRN. So far, the administrator and end-user DR tool have been developed and are currently operational, while access will be provided to selected utility users. The DR efficiency will be measured and evaluated using GSRN and RAT algorithms.

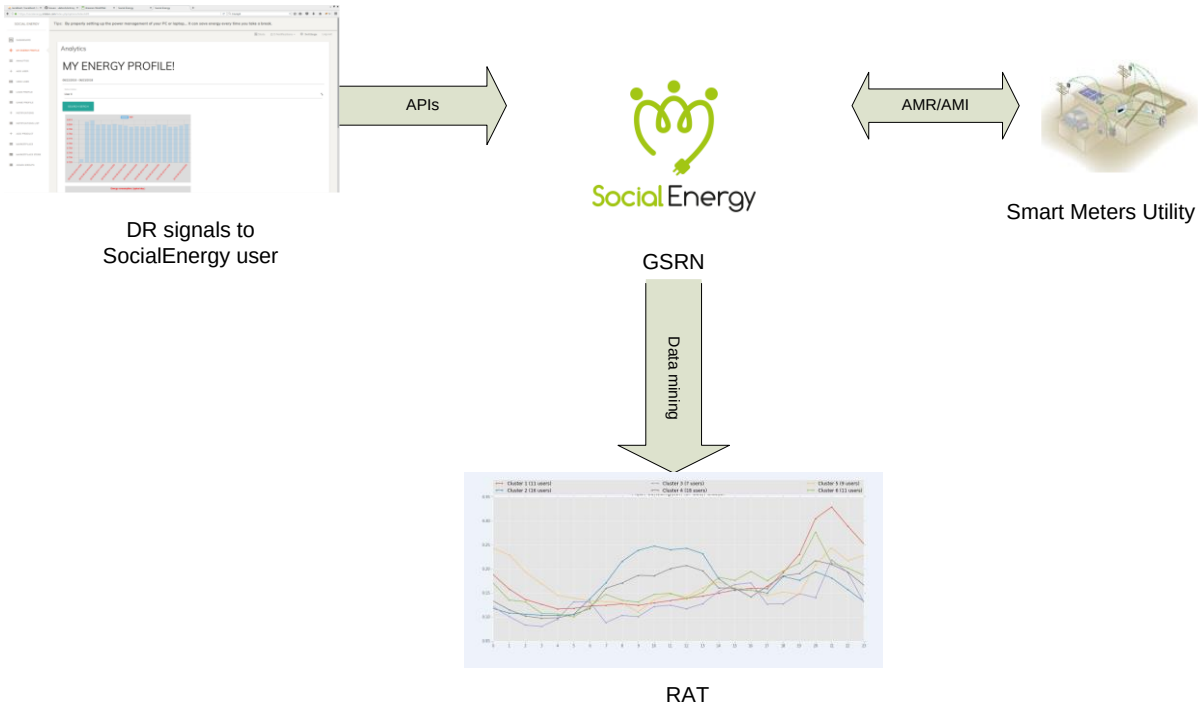


Figure 1: The high-level process of behavioural DR in SOCIALENERGY

Scenario steps:

- The user receives a DR notification on the 'My Energy profile' Dashboard
- The DR message is also visible on the Notification table on GSRN
- The user acknowledges the message

- The user complies with the DR notification and tries to change his/her energy behaviour according to the message details and timing slots.
- RAT analyses the energy consumption curves and justifies or not the change, based on the smart meter readings.
- The verification can be done after the DR event has finished.
- All actions will be performed through the 'My Energy Profile' dashboard.

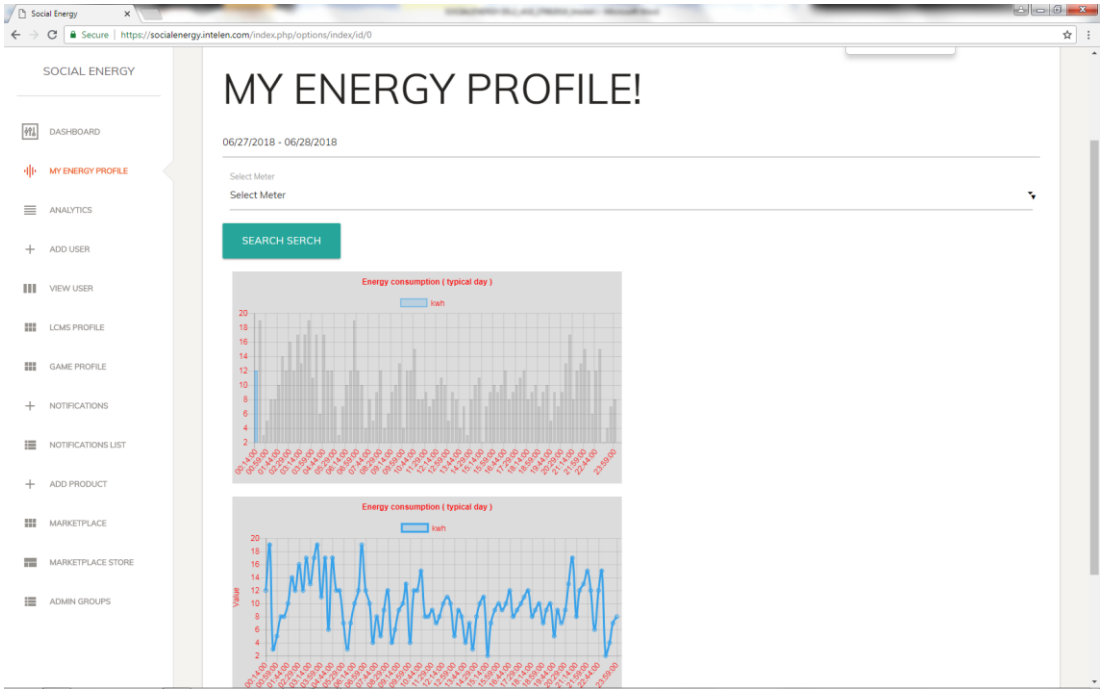


Figure 2: Indicative snapshot from the 'My Energy Profile' dashboard of GSRN

The table below summarizes a potential DR schedule for the GSRN users and the anticipated results. The results and the outcome will be calculated using RAT algorithms and specific KPIs as extensively analyzed in D5.1 (M18).

Table 2: DR Events Summary Example for scheduling the DR events on GSRN

DR Event	Time of Event	Duration	Scope	Status	DR outcome (example)
1	20:00-21:00	1 hour	Reduction	DR was successful	-2.88%
2	19:00-20:00	1 hour	Reduction	DR was not successful	0.37%
3	20:00-22:00	2 hours	Reduction	DR was successful	-3.89%
4	21:00-00:00	3 hours	Reduction	DR was not successful	10.88%
5	12:00-17:00	5 hours	Reduction	DR was not successful	15.45%
6	00:00-02:00	2 hours	Increase	DR was successful	70.47%
7	14:00-18:00	4 hours	Increase	DR was successful	3.37%
8	20:00-22:00	2 hours	Reduction	DR was not successful	7.90%
9	21:00-22:00	1 hour	Reduction	DR was successful	-0.67%
10	19:00-22:00	3 hours	Reduction	DR was not successful	25.14%

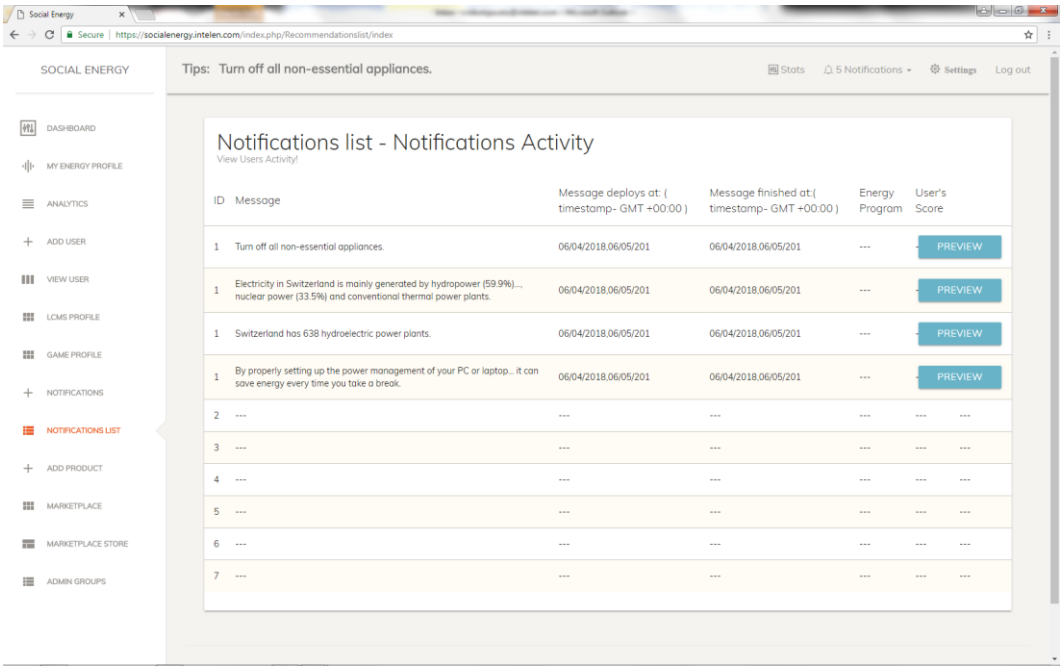


Figure 3: Indicative screenshot for the list of DR notifications viewed by the end user (i.e. energy consumer)

3.1.2. Scenario 1B – Behavioural Demand Response with Gamification

A behavioral DR approach with a points rewarding system will be followed in this scenario, but it will be quite restricted to voluntary participation with direct benefits or incentives for the participants. The users will follow a DR event and upon completion will receive some points, based on the SOCIALENERGY point system (see details in section 3 of D3.1). The points can be later used to be redeemed for prizes and offers at the virtual marketplace.

The table below summarizes a potential DR schedule for the GSRN users and the anticipated results and awarded points; the results and the outcome will be calculated using RAT algorithms and specific KPIs as extensively analyzed in D5.1 (M18).

DR Event	Time of Event	Duration	Scope	Status	Points awarded
1	20:00-21:00	1 hour	Reduction	DR was successful	50
2	19:00-20:00	1 hour	Reduction	DR was not successful	0
3	20:00-22:00	2 hours	Reduction	DR was successful	5
4	21:00-00:00	3 hours	Reduction	DR was not successful	0
5	12:00-17:00	5 hours	Reduction	DR was not successful	0
6	00:00-02:00	2 hours	Increase	DR was successful	50
7	14:00-18:00	4 hours	Increase	DR was successful	50
8	20:00-22:00	2 hours	Reduction	DR was not successful	0
9	21:00-22:00	1 hour	Reduction	DR was successful	100
10	19:00-22:00	3 hours	Reduction	DR was not successful	0

Scenario Steps:

- The user receives a DR notification on the 'My Energy profile' Dashboard.
- The DR message is also visible on the Notification table on GSRN.
- The user acknowledges the message.
- The user complies with the DR notification and tries to change his/her energy behaviour according to the message details and timing slots.
- GSRN rewards the user with some points based on his/her on-line participation.
- RAT analyses the energy consumption curves and justifies (or not) the change, based on the smart meter readings. GSRN awards more points in case RAT indicates savings based on the DR event.
- The verification can be done after the DR event.
- All actions will be performed through the 'My Energy Profile' dashboard.

3.1.3. Scenario 1C – Energy Communities formation and advanced energy programs

In this use case scenario, we test and validate the functionalities of business analytics and intelligence tool of SOCIALENERGY S/W platform. These functionalities are mainly provided by the RAT subsystem, which is actually the backend system of the platform. The most important results from backend are illustrated in the system's frontend (i.e. GSRN) via a user-friendly GUI (cf. 'Analytics' tab). The data exchange between these two subsystems is realized via the GSRN-RAT API. As explained in the respective validation activities' table for RAT in the appendix, historical energy consumption datasets from real users have been used at this phase, while real-time energy data and behavioural data from real users will be integrated in RAT during the upcoming months.

The most important results of the S/W integration work regarding use case scenario 1C was the integration of the dynamic pricing algorithms and the EC creation and dynamic adaptation algorithms inside the RAT. We validated that the research results published in all scientific papers and have been extensively analysed in D3.1 (M15) are the same with the ones presented through the user-friendly GUI in the RAT web tool.

3.1.3.1. Energy programs

Energy program evaluation is currently operational, using historical datasets from ICCS portfolio. A list of scenarios evaluating the energy program performance may be found at https://rat.socialenergy-project.eu/scenarios. We have performed exhaustive testing to validate and verify the correct operation of each dynamic pricing algorithm. As a result, the utility user (i.e. admin) is now able to set a wide range of input parameters (e.g. set of energy consumers, timeframe, energy cost parameter, profit margin parameter, flexibility factor, etc.). Then, the utility user may select the various energy programs (EPs) that s/he wants to compare with respect to several KPIs such as energy cost, aggregated users' welfare (AUW), retailer's profit, total consumption, total electricity bills, etc.

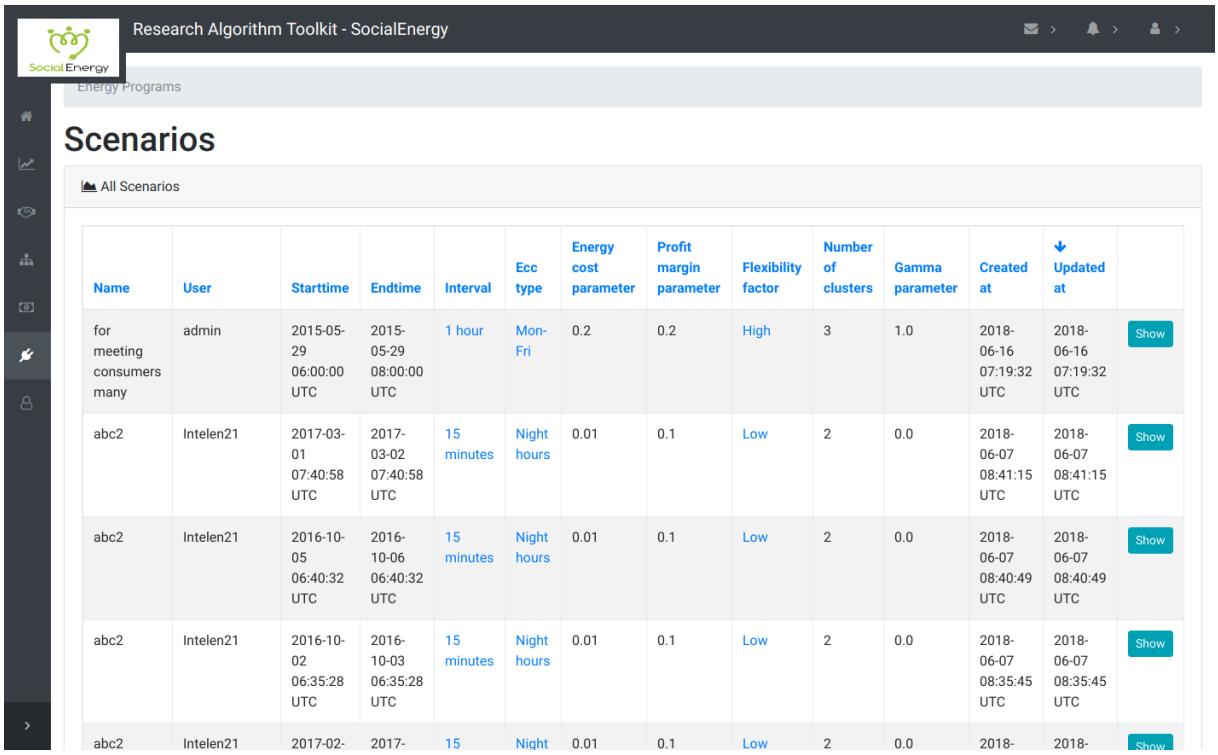


Figure 4: List of scenarios to test and validate the performance of the various energy programs

Results from an indicative scenario are shown in the screenshot below. Four basic energy programs are compared: i) Real Time Pricing (RTP) without demand response (DR), ii) RTP with DR, iii) Personalized RTP (P-RTP), and iv) Community RTP (C-RTP). We can see that the P-RTP and C-RTP programs can achieve significant savings in the system's energy cost, while maintaining a high user welfare. The total energy consumption is also reduced, and the consumers' bills are lower. This provides a strategic advantage to the utility company to be competitive enough in the liberalized retail electricity market, as it can strategically decide how to best exploit the reduction in the total energy cost reduction, either by lowering the clients' electricity bills or by increasing its profits.

3.1.3.2. EC creation (Clustering) algorithms

The main business analytics functionality of SOCIALENERGY platform is mainly realized through the execution of EC creation algorithms. RAT provides an automated and dynamic way through which an electric utility company's portfolio can be efficiently managed. For example, the utility can establish effective communication with its clients by

sending them context-aware reporting and recommendation messages regarding the use of their energy resources. The EC creation algorithms can cluster energy consumers into several groups according to their energy consumption curve/pattern, energy flexibility curve/pattern, their habits, their level of engagement in SOCIALENERGY platform's features, their interest to communicate with other peers in online social networks, their need to purchase more energy-efficient products and services, etc. As a result, the utility user may apply automated advertising, Know-Your-Customer (KYC) and communication campaigns aiming at providing hyper-personalized energy services to its clients. For example, energy consumers who can benefit from a specific energy program can be clustered together and subsequently receive a recommendation message based on which the end user can understand the quantitative and qualitative benefits from switching to a new energy program. Of course, this concept can be further strengthened by the use of the LCMS and GAME subsystems, in which the end user may be informed about further details in order to deeply comprehend the need to change his/her energy behaviour/habits.

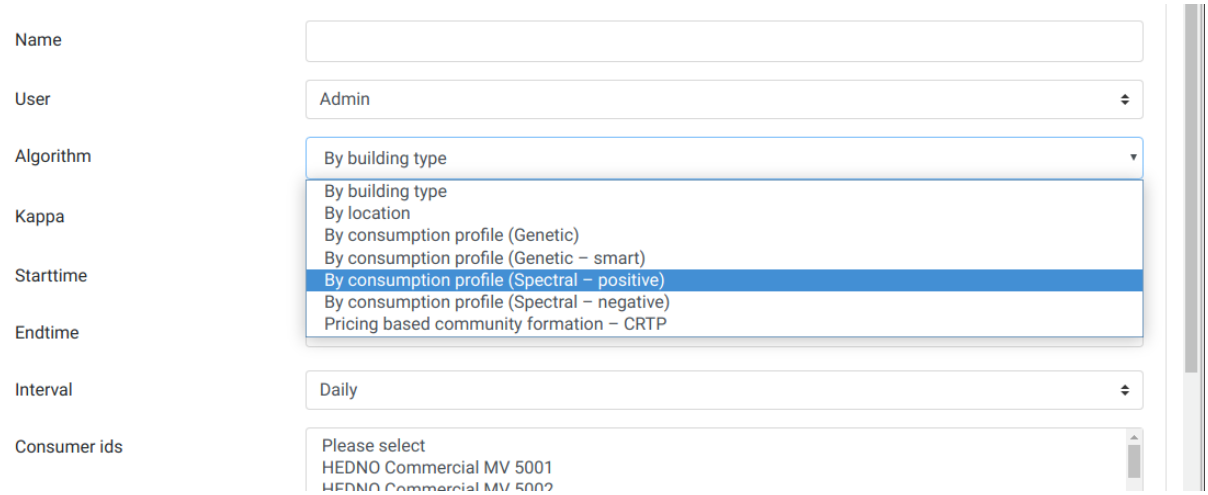


Figure 5: Execute various types of EC creation algorithms

An example run of an indicative EC creation (clustering) algorithm is shown in the figure below. We have created an automatic clustering based on daily energy consumption patterns, using the data of a set of 50 commercial consumers. We can see that three groups are created. The first cluster contains the consumers that show an irregular consumption, whereas the second cluster contains the consumers that have a lower consumption on Sundays, and the third cluster contains the consumers that have a lower consumption on Saturdays and Sundays. As a result, an automatic recommendation message may be created for the third cluster of energy consumers, telling them about the benefits of purchasing an energy program that better fits their energy consumption profile. Subsequently, the end users may find this option quite appealing, as they are usually away from their home on the weekends, so they could get a discount in their electricity bills.

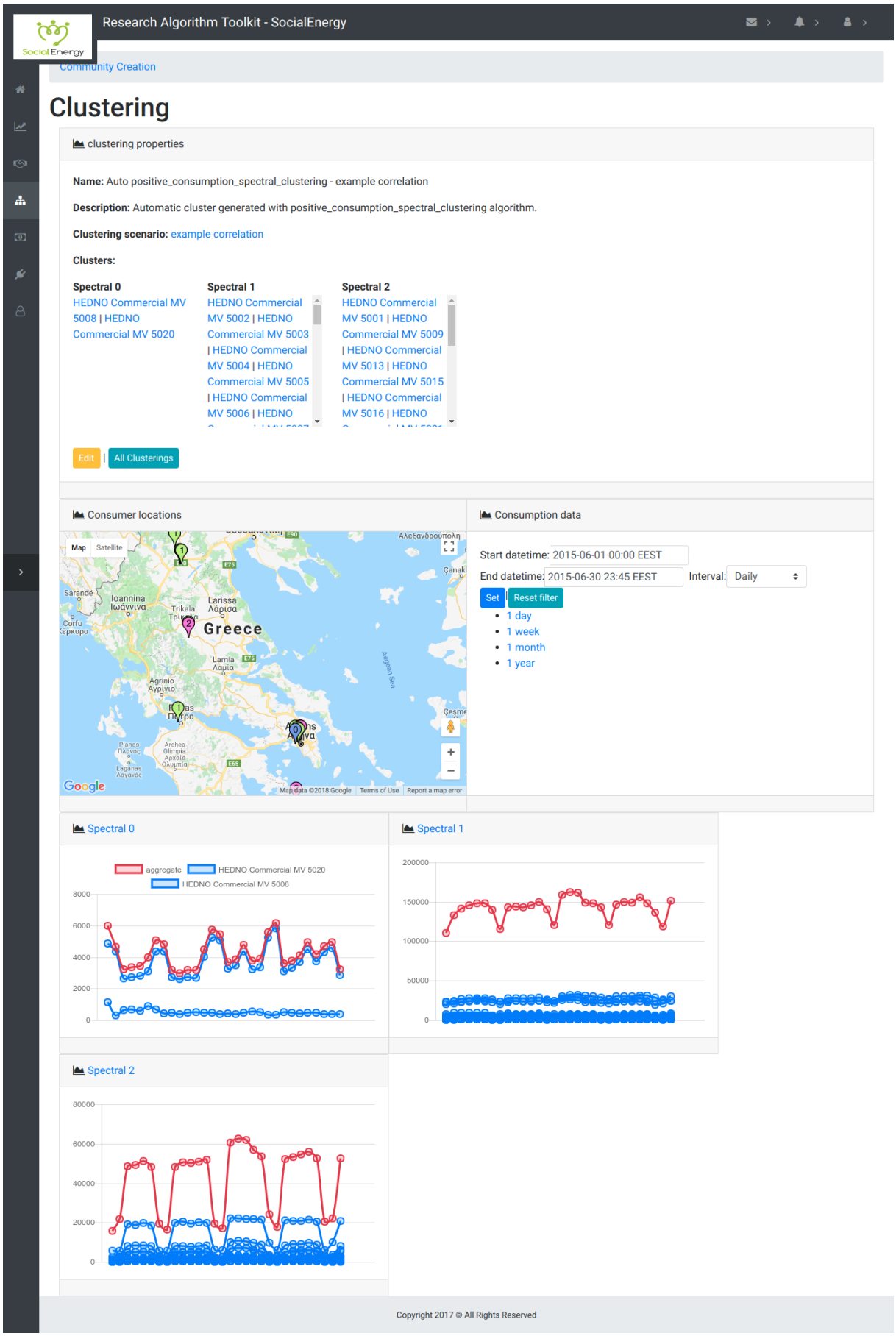


Figure 6: Indicative results from the validation of a spectral clustering algorithm

3.1.4. Scenario 1D – Personalized marketing and virtual marketplace

In this scenario, the users are receiving discounts and offers, through the GSRN Marketplace dashboard. RAT clusters users and creates specific classes of people with common characteristics (i.e. derived from DR assessment, LCMS, Energy analytics, demographics, social KPIs, peer pressure, etc.). Based on these clusters, GSRN sends discounted offers as notifications to users and users are able to go to the Marketplace and select the appropriate electric appliance.

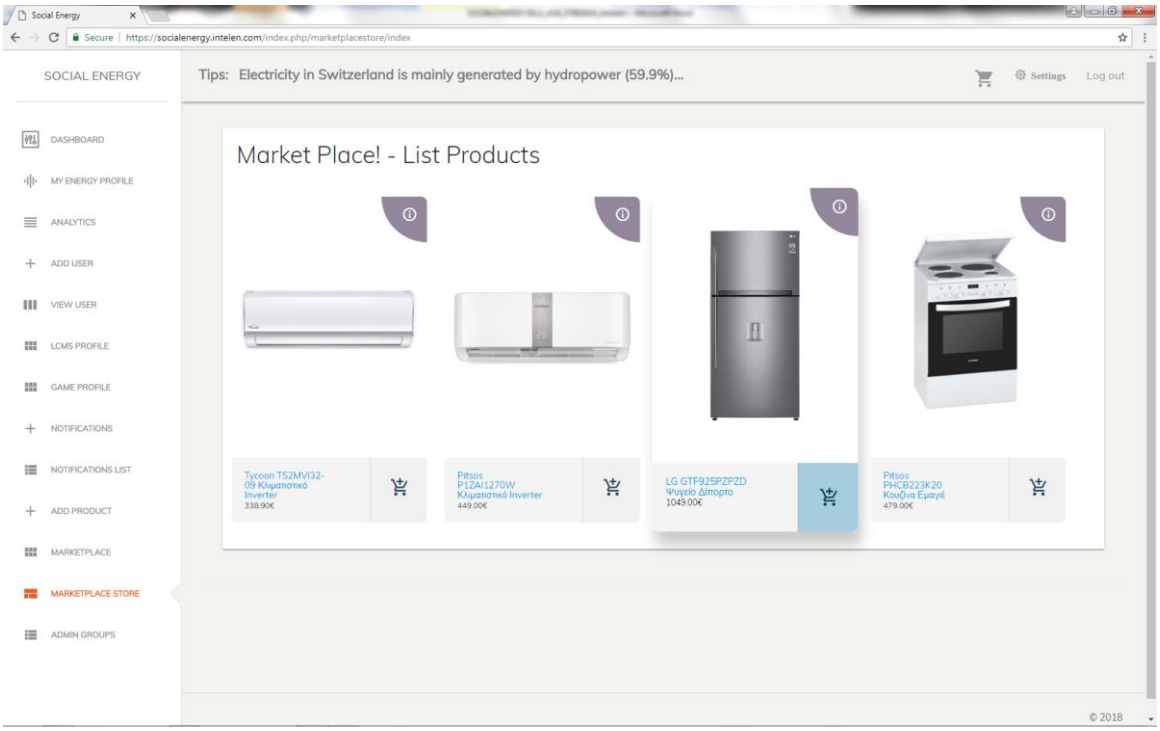


Figure 7: Indicative screenshot of an end user's offers/discounts found in SOCIALENERGY's virtual marketplace

Scenario Steps:

- The user receives a discount/offer notification on GSRN.
- The offer is also visible on the Notification table on GSRN.
- The user acknowledges the message and can click on the offer to go to the 'Marketplace' dashboard.
- The user can read details on the specific appliances (Energy efficiency, costs, etc.).
- The user enters the 'Marketplace' dashboard and reads the discounts and personalised offers.
- RAT analyses the user active on-line participation and other behavioural KPIs.
- The user clicks on the specific appliance (in the offer/discount) and is redirected to the actual e-shop for purchase, by using the unique GSRN discount token (affiliate nets).
- All actions will be performed through the GSRN 'Marketplace' dashboard.

3.2. Use Case no. 2 - SOCIALENERGY's Virtual World (GAME application)

For validation purposes of each separate use case scenario, a similar approach of task fulfilment was followed to ensure that upon completion of the same activities and

completion of the jobs, the player will end up with different game score which is a result of calculation of the satisfaction score and the daily costs score. The satisfaction score depends on the number of jobs available to the player on the certain day, but also if these jobs were fulfilled in a certain timeframe and the convenience from the electronic device modes. The daily costs score are calculated using the daily costs, the maximum daily costs and minimum daily costs. The formulas used for calculation of the scores are the following:

- Maximum Satisfaction Score for each Job = Maximum Satisfaction Score per Day / Total Amount Daily Jobs
- Daily Cost Score = 100 – ((Maximum Costs –Daily User Costs) / (Maximum Costs – Minimum Costs)*100)
- Daily Total Score = Daily Satisfaction Score – Daily Costs Score

The satisfaction score doesn't change from scenario to scenario, but the daily costs score changes as the price calculation is different.

3.2.1. Scenario 2A – Fixed pricing gameplay

Figure 8: Game task planner

To validate the correct implementation of the gaming environment with the use of research algorithms (RAT) and the overall logic of the SOCIALENERGY Game, the Fixed pricing scenario is taken as a baseline as it is the easiest for a regular energy consumer to comprehend. In the fixed pricing scenario, the player will see in the task planner that the price is the same independent from the time.

For validation purposes, all the tasks across the board of the scenarios are selected in the same order and afterwards performed accordingly within the same time frame. The player will see the fixed price in the bottom right corner of the user interface and the graph above the price.

Figure 9: Game environment (HUD)

During the validation of this scenario, the player will perform multiple jobs at different timeframes according to a desired convenience. A job consists of multiple steps, whereas some steps are simple interactions like getting laundry from bedroom shelf that had to be repeated across all the implemented use case scenarios.

To showcase the energy consumption interaction throughout the validation process, interactions with electronic devices were selected. While interacting with the electronic device, the player will have the possibility to choose between three consumption modes. These modes define the energy consumption, but also the convenience for the player.

Figure 10: Game electric device mode selection

The fastest mode has the highest convenience, but also the greatest consumption. The 'Eco' mode has the lowest consumption, but also the most time consuming one. The 'normal' mode is between the two previous modes. Some of the electric devices have only one consumption mode, like an iron that was also used during the validation of the correct implementation throughout the use case scenarios (see figure below).

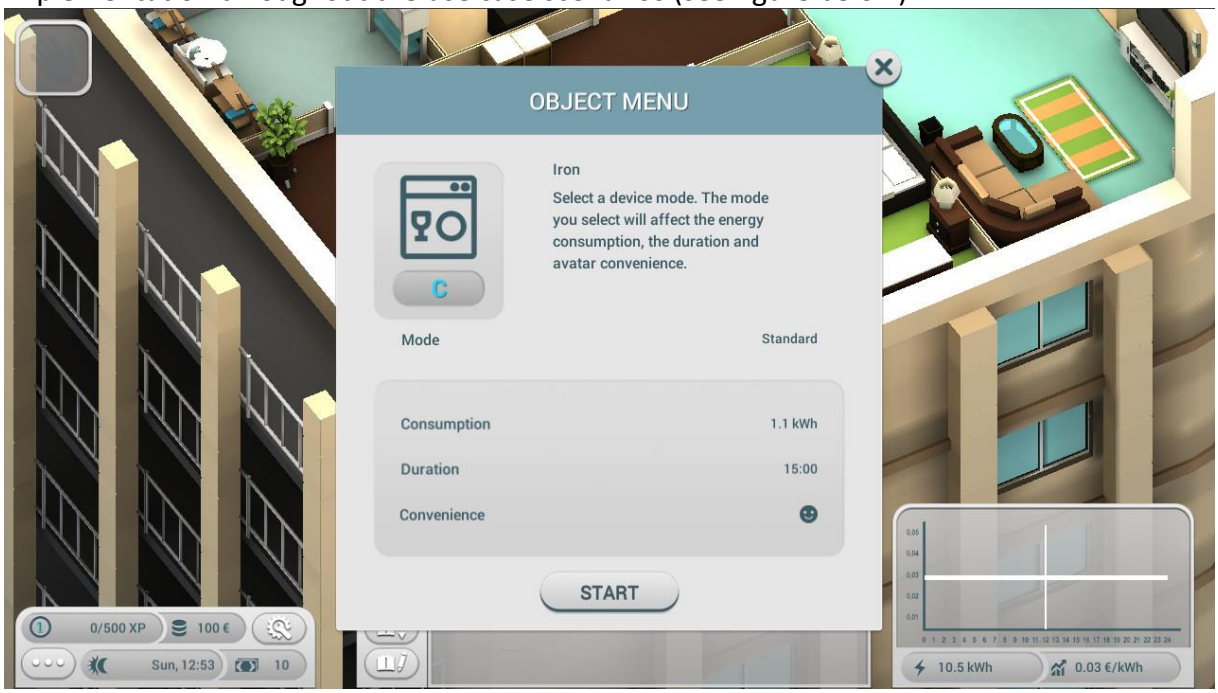


Figure 11: Single mode electric device

Finally, the major point of validating the fixed pricing scenario is presenting that the costs are a constant, which the user neither overspent nor saved (0.32$ of 0.32$) as depicted in the figure below. While doing multiple jobs, the player will see that the price is not changing. The results present the overall energy consumption as well as the convenience

(which is an external factor here that is also dependant on what time the user "had breakfast" for example).

Figure 12: Job results in fixed pricing use case scenario

Here, the player should achieve the best energy efficiency score and possibly improve the convenience. Price is flat and fixed, so the only dynamic parameter is the KWh consumed. The KW consumed is a determinant here as a proof that the same interactions and therefore same jobs, same electric appliances and modes were used in consecutive use cases.

3.2.2. Scenario 2B – Time of Use (ToU) pricing gameplay

In the Time of Use (ToU) scenario, the price is dependent on the timeframe that the various jobs are performed. Like it is in the fixed pricing scenario, the actual price for electricity is shown in the Task Planner and the User Interface. For example, the price is low in the night and reaches a high peak in the morning till mid-day. After the mid-day, the price is slowly decreasing till it reaches a low at about 15 o'clock. In the evening, the price rises till 20 o'clock, after which the price decreases till it's low in the night.

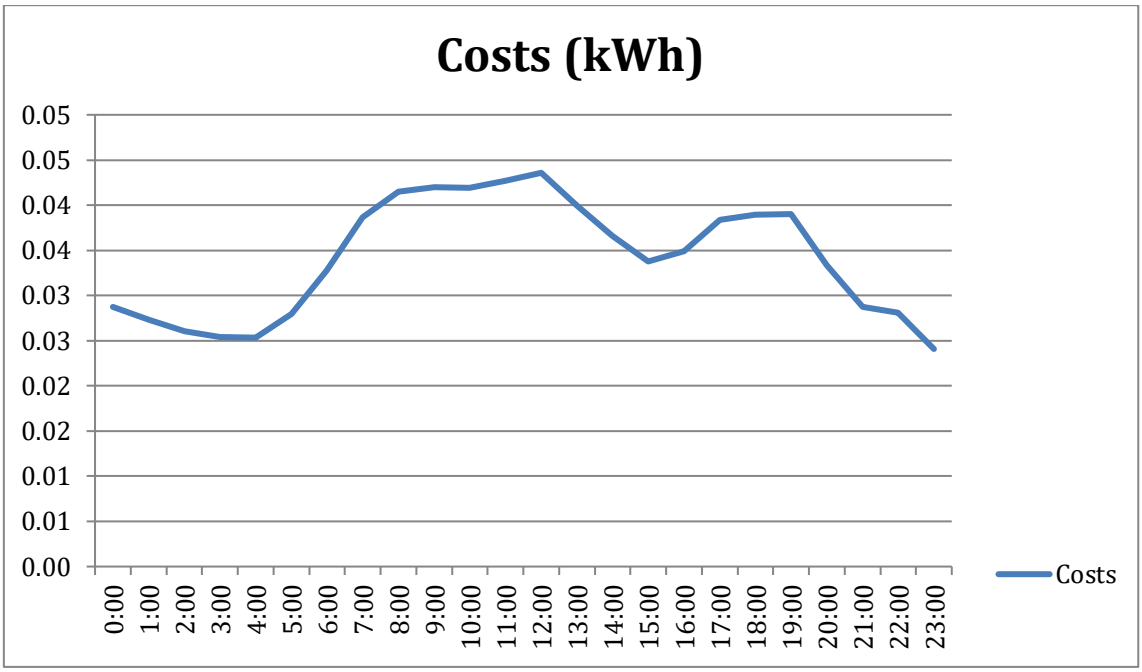


Figure 13: **Time of Use cost chart (taken from real day-ahead markets in Europe)**

The player will get different results depending on the time that the task is performed. As described in the section 3.2.1 above, for validation purposes, the same steps were followed to ensure the validity of the tests. The following two figures indicate the same overall consumption in KW (10.78 KW), but the costs are different depending on the timing that the user performs the actions. For demonstration purposes, two timeframes were used, where the first one exactly reflects the timeframe of the fixed pricing validation (e.g. finishing all the jobs by 13:00) and the second figure shows what would happen, should the player decide to shift some activities to later hours. As a result, we indicate that given the same consumption of energy, different costs arise indicating that the user has reached closer to best gameplay (note: the XP and Credits are not part of this validation, but belong to balancing of the overall gameplay, which is planned to be adjusted when towards the end of the developments). The goal of the player is to achieve the lowest cost by variating the time point the task is performed and variating the energy consumption curve. The convenience of the player should also not be left out of sight.

Figure 14: **Game ToU result by 13:00**

Figure 15: **Game ToU result by 21:00**

3.2.3. Scenario 2C – Personalized Real Time Pricing (P-RTP) gameplay

In the Personalized Real Time Pricing (P-RTP) scenario, the player will get discount according to his/her behaviour in the game, while the base price is changing dependent on time of day. The discount was calculated as follows:

*discount = a_cuts * (price at the moment / maximal price of the day)*

The a_cuts is received from RAT.

For the full price calculation, the following formula was used:

*Price = convenience (mode) – consumption (mode) * (price at the moment / maximal price of the day) + discount*

The player will get discounts by behaving in the most energy efficient way. The behaviour includes performing the task at the best timeframe of the day with the most efficient consumption mode without forgetting the convenience. Again, the same approach for the validation was followed as described in the previous sections. The player will see the difference in the changing price. The following two figures provide indicative results of the changing costs for the user with the same consumption as in previous use cases and for demonstration purposes, Figure 17 shows the change in convenience and again, costs, should the user choose another timeframe for performing the task and choosing another mode of the device (which is reflected by its overall consumption).

Figure 16: Game P-RTP scenario result

Figure 17: Game P-RTP scenario result at 9:00

3.2.4. Scenario 2D – Community Real Time Pricing (C-RTP)

In the Community Real Time Pricing (C-RTP) scenario, the price is not only dependent on the performance of the player, but also on the other members of player's community (i.e. NPCs). The NPC will tell the player, when they perform a task and the player will need to tell the NPC when s/he performs a task. Similar to personalized real-time pricing scenario, the price will change depending on the NPCs' activity. Like in the previous scenarios, the right point in time, consumption mode and player's convenience is important. This use case is currently under development, but the same validation approach as described in the previous use case scenarios will be followed and results will be delivered in D5.3.

3.3. Next steps

The current project's state is that the consortium has successfully accomplished Milestone 5 meaning that the first version of S/W integration and validation activities has taken place. The next step is to release the 'alpha' version of the integrated S/W platform in order to start experimenting and receiving real-life datasets from real users. An integrated DEMO will be presented during the upcoming review meeting (Athens, 18/09/2018). Then, the goal is to release an initial 'beta' version in M24 and start gathering inputs from real end users, who belong to a real electric utility company's portfolio. Subsequently, testing and validation results will be delivered in M27 via D5.3. Finally, in D5.4, all results from real-life pilot tests will be gathered with respect to the 8 use case scenarios of the project.

Appendix

Table 3: Summary of GSRN testing and validation activities

ID	Validation check	Expected outcome	Real outcome
GSRN01	User registration, login, create account to LCMS.	An Admin should be able to create a new user account in GSRN, under the hood: GSRN creates a new user account to LCMS.	Just as expected
GSRN02	User logins with GSRN credentials, to RAT – LCMS using oauth2.	A user with an active GSRN account should be able to login to the RAT – LCMS - GSRN using GSRN credentials.	Just as expected
GSRN03	User account management.	A registered user should be able to update his credentials, edit his own account, etc.	Just as expected
GSRN04	A Registered user should be able to complete the questionnaire.	The first time a user logins, questionnaire must pop-up, after successful completion, questionnaire will be hidden.	Just as expected
GSRN05	Results from questionnaire must be posted to LCMS.	LCMS is using results from questionnaire to create – initialize learning plan of user.	Just as expected
GSRN06	Real-time data visualization of analytics.	Real-time data should be displayed in the 'My Energy Profile', data is coming from MDMS database.	Just as expected
GSRN07	Real-time data visualization of LCMS PROFILE.	Real-time data should be displayed in the 'LCMS PROFILE', data is coming from LCMS service saved to MDMS database.	Just as expected
GSRN08	Real-time data visualization of GAME PROFILE.	Real-time data should be displayed in the 'GAME PROFILE', data is coming from MDMS database, posted from GAME app.	Just as expected
GSRN09	Real-time data visualization of Notification.	Real-time data should be displayed in the 'GAME PROFILE', data is coming from MDMS database, posted from RAT.	Just as expected
GSRN10	Create a new product – support CRUD functionalities.	An admin must be able to create a new product in the marketplace, view, update or delete if s/he wants.	Just as expected
GSRN11	User must be able to add products to his basket or to checkout.	A user should be able to see all the products, to sort them, add them to his basket or even to checkout.	Just as expected
GSRN12	An Admin investigates all transactions of	An admin should be able to edit/update/delete all transactions of	Just as expected

ID	Validation check	Expected outcome	Real outcome
	marketplace.	marketplace.	
GSRN13	User must be able to create communities.	A user can request permission to create group, after success, s/he can search and invite members to his/her group.	Just as expected
GSRN14	An Admin must have CRUD functionalities over communities.	An admin user should be able to edit/update/delete all transactions of communities.	Just as expected
GSRN15	User adds widget to his dashboard.	A user can add/delete/rearrange widgets to his/her dashboard page.	Just as expected
GSRN16	Widget of dashboard showing updates.	Widget of dashboard showing, receiving new records.	Just as expected
GSRN17	MDM-GSRN service.	MDM broadcast data (consumptions – consumers – geolocation) to GSRN.	Just as expected
GSRN18	GSRN-RAT service.	GSRN broadcast data to RAT - user behavioral data from MDM.	Just as expected
GSRN19	GSRN-GAME service.	GSRN authenticate user, saves game actions of user.	Just as expected
GSRN20	GSRN-LCMS service.	GSRN pulls LCMS user actions, saves them to MDMS.	Just as expected
GSRN21	GSRN-Marketplace service.	GSRN broadcast marketplace data. (behavior data – product related data)	Just as expected

Table 4: Summary of RAT testing and validation activities

ID	Validation check	Expected outcome	Real outcome
RAT01	User registration and login	A user should be able to create a user account in RAT and login	Just as expected
RAT02	User login with GSRN credentials	A user with an active GSRN account should be able to login to the RAT using GSRN credentials	Just as expected
RAT03	User account management	A registered user should be able to update his credentials, edit/delete his own account, etc.	Just as expected
RAT04	Register a new energy consumer and synchronize energy data exchange with MDMS	An administrative user should be able to register a new consumer in the RAT database and synchronize with central MDMS database	Just as expected
RAT05	View list of consumers/ consumer details, edit/delete consumer details	A user should be able to view the list consumers together with their details and manage them according to his/her user rights	Just as expected
RAT06	Historical data visualization	Historical data should be displayed in the 'Consumer', 'Community' and 'Clustering'	Just as expected for energy data,

ID	Validation check	Expected outcome	Real outcome
		views in various time granularities and for any given timeframe in the past	behavioral data visualization service is ready for real-life pilots
RAT07	Real-time data visualization	Real-time data should be displayed in the 'Consumer', 'Community' and 'Clustering' views. The graphs should be updated dynamically as new data arrives	Just as expected, need input from more real smart meters (GSRN) to use in real-life pilots
RAT08	Community creation	An admin user should be able to create "communities" (i.e. groups of consumers). A consumer may belong to several communities, as long as they are in different 'clusterings'.	Just as expected
RAT09	View list of communities and details	A user should be able to details about each community such as community members, aggregated consumption, the 'clustering' it belongs to etc.	Just as expected
RAT10	Manage a community	An admin user should be able to edit/update/delete a community, for example to change its name, description, member consumers, etc	Just as expected
RAT11	Create a new 'clustering'	An admin should be able to create a new clustering (i.e. group of communities), based on several input parameters	Just as expected
RAT12	View list of 'clusterings' and details	A user should be able to see a list with all the 'clusterings' in the system and details about each 'clustering'	Just as expected
RAT13	Manage a 'clustering'	An admin user should be able to edit/update/delete clustering details, such as its name, description, and participating communities	Just as expected
RAT14	Algorithmic clustering creation – Consumer type	A user can create a clustering automatically based on the consumer type, location, energy program, etc.	Just as expected, need for more input from GSRN real-life pilots
RAT15	Algorithmic clustering creation – Consumption patterns	A user can create a clustering automatically based on the consumers' consumption patterns	Algorithmic results and visualization just as expected
RAT16	Algorithmic clustering creation – Flexibility patterns	A user can create a clustering automatically based on the consumers' flexibility patters (i.e. similar levels of flexibility over time)	Just as expected, need input from GSRN (real-life pilots w.r.t. DR events)
RAT17	Algorithmic clustering creation – Behavioral data from GSRN activities	A user can create a clustering automatically based on the consumers' GSRN participation (e.g. similar levels of participation)	Just as expected for mock-up data, need input from GSRN during real-life pilots

ID	Validation check	Expected outcome	Real outcome
RAT18	Algorithmic clustering creation – LCMS participation	A user can create a clustering automatically based on the consumers' LCMS participation (i.e. similar levels of participation/ learning achievements/level)	Just as expected for mock-up data, need input from GSRN during real-life pilots
RAT19	Algorithmic clustering creation – GAME participation	A user can create a clustering automatically based on the consumers' GAME participation (i.e. similar levels of gameplay achievements/engagement, etc.)	Just as expected for mock-up data, need input from GSRN during real-life pilots
RAT20	Reporting/ Recommendation service creation	An admin user should be able to create a new recommendation/ reporting service by using the output of the clustering algorithms	Just as expected for manual mode, need to integrate with clustering module for automatic mode
RAT21	View list of recommendations	A user should be able to see/preview the list of all recommendations that have been created and their status	Just as expected
RAT22	Manage recommendations	An admin user should be able to edit/delete/update a recommenda-tion (before it is sent) and then successfully send to the end users	Just as expected, need for more personalized messages in the future
RAT23	Create energy program evaluation scenario	A user should be able to create an energy program evaluation scenario, and assign various parameters to the scenario	Just as expected for historical data, advanced version with real-time datasets
RAT24	Create energy program evaluation scenario through API	A GSRN user can create an energy program scenario through the GSRN platform using a REST API	Just as expected
RAT25	View list of energy program evaluation scenarios	A user should be able to view the list of the energy program evaluation scenarios that have been created and compare the results	Just as expected
RAT26	Manage energy program evaluation scenario	A user should be able to edit/update/delete an evaluation scenario that s/he has created	Just as expected
RAT27	View and compare various energy programs	A user should be able to visualize the results of the evaluation scenario (various KPIs), compare various energy programs in order to select the most beneficial one	Just as expected

Table 5: Summary of GAME testing and validation activities

ID	Validation check	Expected outcome	Real outcome
GAME01	User login with GSRN	A user with an active GSRN account	Just as expected

ID	Validation check	Expected outcome	Real outcome
	credentials	should be able to login to the GAME using GSRN credentials.	
GAME02	GSRN Competence Level	GSRN Competence is received and used to recommend Energy program	
GAME03	Avatar Customization	A user should be able to customize and play the customized avatar.	Just as expected
GAME04	Send Avatar Data	Data of the avatar is sent to GSRN to have a consistent avatar on all platforms	
GAME05	Energy programs	A user should choose energy program according to his current level.	Just as expected
GAME06	Info from LCMS	The user can get more information about the energy program from a link to LCMS.	Just as expected
GAME07	Task planner	The user plans his/her day by dragging and dropping his/her daily tasks into the plan at the hour s/he needs.	Just as expected
GAME08	Avatar control	The avatar can be moved by the user and interaction with objects is possible.	Just as expected
GAME09	Tutorial	The game provides the player with information needed to understand the game concept	Just as expected
GAME10	Jobs	Jobs can be made at the scheduled time or independent from schedule	Just as expected
GAME11	Activity	An activity is a step of a job trigger by interaction with an object. Activities have certain duration and unlock the next step of the job.	Just as expected
GAME12	Device options	Multiple device options are available and differ in energy consumption and convenience.	Just as expected
GAME13	Job Result	After a job is finished, a result window is displayed with user consumption, cost, convenience and reward.	Just as expected
GAME14	Rewarding	After a job is finished, the user is getting a reward.	Just as expected
GAME15	Sending Job Info	Job information is sent to GSRN	Just as expected
GAME16	Fixed price program	In the fixed pricing energy program the prices are not changing.	Just as expected
GAME17	Result of the Day	The results of the day are presented to the user: Amount of Jobs finished, Consumption of the day, costs of the day compared with minimum costs, convenience of the day compared with maximum convenience, daily convenience score, daily cost score.	Just as expected

ID	Validation check	Expected outcome	Real outcome
GAME18	Time of Use pricing program	In the time of use pricing program, the prices depend on the timestamp that the user performs an activity.	Just as expected
GAME19	Personalized real time pricing	The prices depend on the decisions made by user.	Just as expected
GAME20	Real Time Energy Community pricing	The price depends on the decisions of the user and the in-game NPCs.	Not yet implemented, will be implemented till M24
GAME21	Upgrades	The user can purchase upgrades for his/her electronic devices. The upgraded devices replace the former ones. The upgraded devices change the device options.	Just as expected
GAME22	Decorations	The user should be able to buy decoration items.	Just as expected

Table 6: Summary of LCMS testing and validation activities

ID	Validation check	Expected outcome	Real outcome
LCMS01	User registration and login	A user should be able to create a user account in LCMS and login	Just as expected
LCMS02	User login with GSRN credentials	A user with an active GSRN account should be able to login to the LCMS using GSRN credentials	Just as expected
LCMS03	User profile management	Registered users should be able to change their own credentials (available only for locally registered users) and update account's settings	Just as expected
LCMS04	Create competency framework	An administrative user should be able to set up competency frameworks and add competencies to them.	Just as expected
LCMS05	Create learning plan on behalf of GSRN	A user authenticated with GSRN credentials should be automatically assigned with an individual learning plan, which includes all missing competences determined by the results from GSRN questionnaire	Just as expected
LCMS06	Create learning plan	An administrative user should be able to create learning plan templates, add competencies to them and assign learning plans to a cohort of users or to individual selected users.	Just as expected
LCMS07	Create course	An admin user should be able to create course and add learning activities like text reading, take quizzes, forums etc.	Just as expected

ID	Validation check	Expected outcome	Real outcome
LCMS08	Assign competencies to courses and course activities	An admin user should be able to add competencies to courses and course activities, and configure the rules for acquiring proficiency level.	Just as expected
LCMS09	Dashboard	Registered users should be able to view dashboard with their own learning plans, courses (in progress and passed) and the courses' progress.	Just as expected
LCMS10	Follow Learning Plan	According to the learning plan, the LCMS should provide appropriate educational materials to the user. Users should be able to view the degree to which learning plan is fulfilled.	Just as expected
LCMS11	Follow course	The user can view, download learning materials and perform different tasks such as uploading files, fill in quizzes, reply with a text in forums etc.	Just as expected
LCMS12	Obtain badge	When a registered user acquires a new competence, s/he is automatically awarded a badge.	Just as expected
LCMS13	RESTful Competency API	The RESTful Competency API should allow LCMS to communicate with the other subsystems learners' competencies and levels' of proficiency, courses' progress and grades, acquired badges.	Just as expected