File created: 18.08.2021., 13:26 CET by Fatima Velic

Last modified: 19.10.2021., 14:58 CET by Fatima Velic

Dürr Systems AG/ Operations

ORC calculation

**The Requirements Document**

**Abstract**

This website provides calculation mechanism which, based on user input, simulates behaviour of the Organic Rankine Cycle (ORC) and suggests machines based on the results of the calculation. In this cycle, liquid medium is fed to a feed pump, where it gets optionally preheated, then evaporated and relaxed again in a turbine. This is followed by the condensation of the medium and the return to the feed pump.

**Document Revision History**

The project was initially created by Dr Josip Kutnjak (Dürr Systems AG) with the aim to simulate the behaviour of real-world ORC machine. Project was initially created as a group of scripts but was later integrated into fully operable website with user management and machine suggestion mechanisms.

Version 0 (initial version) – program is operable and executable inside an IDE editor and as a console application. Contents: routines for necessary calculations and hardcoded variable inputs. Calculation results exported as excel file and displayed in the console.

Version 1 (August 12th, 2021.) – performance optimisation in all modules, calculations execute <2 s with fixed inputs.

Version 1.0.1 (August 17th, 2021.) – added user inputs for mass flow, temperature and pressure variables with exception handling for non-numeric symbols, negative numbers and upper and lower limits. Included exception handling for results file and folder creation. Added feature to graph the results as a T-s diagram.

Version 1.1.1 (August 30th, 2021) – fixed circuit bugs, added bounding conditions and T-Q diagrams.

Version 2.0. (September 15th, 2021) – setup for web application completed. Basic model, templates and views created.

Version 2.1. (September 20th, 2021) – calculation modules integrated. Authorisation and authentication setup correctly, User registration forms added.

Version 2.1.1. (September 27th, 2021) – added custom validators, miscellaneous messages, searching and filtering options.

Version 2.1.2. (October 19th,2021) – incorporated solutions suggestions based on calculation, added more custom validators, introduced contact form, statistical graphs and personal style customisation

**Glossary**

|  |  |
| --- | --- |
| Python | Widespread OOP used to carry out all calculations. |
| Django | Web framework with Python backend engine |
| NIST REFPROP | Program which provides database with the most accurate thermophysical property models for a variety of industrially important fluids and fluid mixtures, including accepted standards. |
| REFPROP DLL | High level API intended to allow interaction between Python code and NIST REFPROP program and database |

**Introduction**

The Organic Rankine Cycle (ORC) is named for its use of an organic, high molecular mass fluid with a liquid-vapor phase change, or boiling point, occurring at a lower temperature than the water-steam phase change. The fluid allows Rankine cycle heat recovery from lower temperature sources such as biomass combustion, industrial waste heat, geothermal heat, solar ponds etc. The low-temperature heat is converted into useful work, that can itself be converted into electricity.

The aim of this code is to simulate ORC functionality such that it could be used as a backend operability for a website. When it comes to the website, it is to be developed such that users can input variables for various parameters of the cycle in order to get accurate calculations on the amount of electricity produced and suggestion of possible machine purchase.

**System requirements specification**

Project is intended to be a dynamic website with user friendly frontend and functional yet fast backend, system requirements for it are minimal including a working internet explorer programme of choice and a working network connection. Website is also intended to have user management system for which database is required in order to store user information and calculation parameters.

Database will hold separate tables for storing user information, calculation input parameters and results. Information will be retrieved by comparing and matching keys between the tables.

Technology intended to be used: MySQL (database), Django API framework (backend and frontend), Python (external scripts and programming logic), bootstrap.js (template formatting and design), CSS (custom design and visual personalisation).

**Requirements list**

**Must have**

- Registration and log in (high priority)

- User input fields for parameters such as mass flow, temperature and pressure (high priority)

- Visual representation of the results (various types of diagrams and graphs) (medium priority)

**Should have**

- Different language options (English/German) (medium priority)

- At request, results can be stored to a file (excel/text/pdf) and exported. (medium priority)

**Project tasks**

|  |  |
| --- | --- |
| Task | Description |
| Input | User can input parameters based on which calculations will be made |
| Registration and log in | User must logged in, in order to perform calculation and save results for future references and suggestions. |
| Validation | Validation will be implemented through username user uses for registration |
| Database | Database stores user logs, preferences and calculation results for future references. |
| Authentication and authorisation | User management is executed as a role-based system. |
| Visualisation | User is presented with visual representation of the cycle in form of T-S and T-Q diagrams. |

**Functional requirements**

1. **Registration –** Registration is performed by administrators. Due to nature of the machines and ORC itself, users cannot register on their own. They must contact the company and resolve legal terms before give access to calculation simulation. After the legal aspect is resolved, clients given access credentials under which website administrators registered them into database.
2. **Login –** once logged in, users will be able to perform the calculation after which calculation results as well as the input parameters will be stored in database. Users will be presented with a visual representation of cycle in forms of two different kinds of diagrams and will be offered a possibility to get results exported as PDF file. Users will be also offered a suggestion on which machine to purchase based on their calculation data inputed. .
3. **Calculation** – Calculation can be performed only by logged in users. Once program receives required data, calculation will be performed in the backend and the results will be returned together with the visual representation and a suggestion on possible machine based on calculation. Calculation data will be stored in order to be used for future marketing purposes and observing market behaviour.

**Non-functional requirements**

|  |  |  |
| --- | --- | --- |
| **Requirement** | **Definition** | **More Details** |
| NFR1: Hardware interface | Hardware requirement for the server and client | Operating system: website requires working operating system and running server. It is not OS dependent thus it can operate cross-platform. |
| NFR2: Software interface | Software requirement for the system | Database: user log information, calculation parameters and results, calculation history |
| NFR3: Communication Interface | User needs connection to the internet to access the store via browser | Stable internet connection |
| NFR4: Security | The user must be ensured that his data will not be misused | Passwords are stored as hashes thus cannot be seen nor accessed by administrator |
| NFR5: Availability | User can visit site at any time | Site must be always available |

**Primary actors and user cases**

|  |  |
| --- | --- |
| **Primary actors** | **Use cases** |
| User | * Log in * Perform calculation * Store input parameters and calculation results in database * View machine suggestion based on calculation * Export calculation details in a file |
| Administrator | * Filter calculations * Add user * Remove user * Observe overall statistics |

**Use cases**

**ID:** FR1

**Name:** Registration

**Priority:** High (must have)

**Pre-condition:** browser, internet connection, access password for Administrator account

**Post-condition:** Account is created

**Flow:**

* Administrator user enters “Register new Clients” or “Register new Administrators”
* Administrator enters necessary information and clicks ‘Register’ button
* Final step of registration: administrator is redirected to *All users* page with output message about successful registration

**What could go wrong:**

* Administrator enters email address which is already in use. Administrator will be informed and prompted to use different address to create an account or will be provided with instructions to navigate to login screen and log in with corresponding email and password.
* Administrator enters passwords which does not match. In this case, administrator will be prompted with message to re-enter matching passwords. When validation is completed, message about successful registration will pop up.

**ID:** FR2

**Name:** Log in

**Priority:** high (must have)

**Pre-condition**: user is already registered

**Post-condition:** user will be able to perform calculation, view the results and the suggestion, save the results for future references.

**Flow**:

* User enters homepage link
* User clicks ‘Login’ link and log in page is displayed
* User enters username and password and clicks ‘Login’ button
* Website recognises username and password either as Administrator or Clients and enables functionality for corresponding role.

**What could go wrong:**

* User enters invalid username or password. Error message pops up indicating either one of login information is wrong and user is asked to enter again. Once entered correct, user can proceed using website according to the role its account is authorised with.
* User selects ‘forget username or password’ option. User is asked to enter email address linked to the account it’s trying to access. Recovery link is sent to the email which through which user will be able to access its account and offered to enter new password. After user completes all steps, it will be able to proceed using page according to role its account has.
* User enters non-existing username or password. Error message pops up indicating the error and list contact information of department where valid credentials could be obtained.

**ID:** FR3.1

**Name:** Calculation

**Priority:** High (must have)

**Pre-condition**: User must be logged in in order to complete the action.

**Post-condition:** Calculation will initiate, and user/viewer will be presented with results

**Flow:**

* User enters index link
* User inputs requested values respectively and initiates calculation
* Calculation is completed and user is presented with calculation details including input parameters and results, the T-s and T-Q diagrams of the cycle, machine suggestion based on the details provided and the option to export results into a PDF file.

**What could go wrong:**

* User enters values which are negative, non-numeric or exceed the limits of the machine. User is displayed with short explanation message about error and prompted to input again. User inputs correct values and proceeds to calculation. Calculation is completed successfully, and user is presented with results.
* User enters values which will cause changes in calculations. Since parameters depend on each other, combining certain values will cause state of the medium to change (from liquid to vapor etc.). User is displayed with the results and message about the change in the state that occurred.

**ID:** FR3.3

**Name:** Information storage and export in file.

**Priority:** Medium (should have)

**Pre-condition**: User is logged in as Client

**Post-condition:** User’s calculation details will be saved in a file and exported.

**Flow:**

* User enters index link and inputs required parameters
* User clicks on Calculate and calculation is performed in background
* User is presented with calculation details, diagrams, machine suggestion and an option to export it to PDF file.
* User requests file by clicking the button
* User is presented with the message that the file has been downloaded.

**What could go wrong:**

* File fails to be created on user’s request due to raised exception or incorrect parsing. User is prompted with a message that something went wrong and to try again. On the second go, file is created successfully, and user is presented with message that file is created successfully and ready to be downloaded.

**Timeline**

Diagram

Description automatically generated

**System evaluation**

Evolution started with functionality demonstration to mentor and supervisor. Based on feedback provided, slight adjustments and modifications were made in design. Our web app will be adapted to the continuing growth (the size of system implementing the changes grows according to the lifestyle changes of the business) and also the software system will support more features if that deems as necessary due to future developments.

**References**

1. Lemmon, E.W., Bell, I.H., Huber, M.L., McLinden, M.O. NIST Standard Reference Database 23: Reference Fluid Thermodynamic and Transport Properties-REFPROP, Version 10.0, National Institute of Standards and Technology, REFPROP DLL documentation, Gaithersburg, 2018.