# Group 3: Product Backlog

Explanation

The columns are based on the scrum workshop example.

- Prioritization
    - **Must have:** Critical feature for the project
    - **Should have:** Important but not critical
    - **Could have:** Nice thing to have
    - **Won't have:** Things agreed upon to be excluded
- Estimation (Story Points)
    - **N/A:** Not assigned
    - **1-2 SP:** Small complexity
    - **3-5 SP:** Medium complexity
    - **8-13 SP:** High complexity
    - **13 or more SP:** Ultra complexity
- Connextra template
    - "As a [type of user], I want [some goal] so that [some reason]."
- ID Naming Scheme
    - Asset Manager (AM)
    - Source Data Manager (SDM)
    - Result Data Manager (RDM)
    - Optimizer (OPT)
    - Data Visualization (DV)
    - API
- **Scenario 1:** Using a gas boiler and an oil boiler to meet heat demand cost-effectively, prioritizing the cheaper gas boiler.
- **Scenario 2:** Adds a gas motor and an electric boiler to optimize heat and electricity production based on varying electricity prices.

| ID | User Story | Priority | Estimate | Confirmation |
|---|---|---|---|---|
| AM-1 | As a dev, I want to initialize the AM so that we can easily and statically manage system info. | Must have | 3 | By getting the static system info from the AM successfully. |
| SDM-1 | As a dev, I want to set up the SDM so that it can manage dynamic system data e.g., heat demand and prices for Optimization. | Must have | 5 | By dynamically updating and accessing demand and prices. |
| RDM-1 | As a dev, I want to configure the RDM so that it can store and retrieve the optimization results to analyze performance. | Must have | 3 | By having the ability to save and recall optimization results. |
| OPT-1 | As a dev, I want to create a simple OPT algorithm for the **first scenario** so that we can efficiently schedule heat production with the least expenses. | Must have | 8 | By Displaying a cost-effective heat production schedule in the **first scenario**. |
| OPT-2 | As a dev, I want to improve the OPT algorithm for the **second scenario** so that electricity production for cost and profit OPT is included. | Should have | 13 | By Displaying cost and profit OPT for electricity production in the **second scenario**. |
| DV-1 | As a user, I want a simple DV so that I can see the results and analyze them. | Could have | 8 | By having the ability to visually analyze the results. |
| AM-2 | As a dev, I want to ensure boilers are easy to configure, so that I can simulate their operation and impact. | Must have | 2 | By showing off easy configuration and simulation of the boiler. |
| SDM-2 | As a dev, I want to use real heat demand and price data, so that it can reflect actual conditions. | Must have | 5 | By using real-world heat demand and price data. |
| API-1 | As a dev, I want **APIs** for module communication so that data flow and the system run smooth. | Could have | 5 | By having data exchange with the APIs. |
| API-2 | As a dev, I want to get **real-time** prices so that we can have a relevant and accurate optimization. | Could have | 5 | By showing the use of real-time prices in the OPT process. |
| OPT-3 | As a user, I want to compare the different configs in the **second scenario** so that I can make informed decisions about our setup. | Could have | 8 | By giving options to compare configs and their outcomes. |
| OPT-4 | As a dev, we **won't** implement **machine learning** so that we cam predict heat demand and prices due to our level and timeframe. | Won't have | N/A | By **not** having machine learning implemented in the scope. |

# Definition of Done (DoD)

| Done | Definition |
| --- | --- |
| | Code is complete and follows the team standards (Industry standards). |
| | Code has been Refactored. |
| | Code has undergone peer review by at least one other member. |
| | We have met the acceptance criteria. |
| | Unit tests are written and passed. |
| | Documentation is up-to-date and reflects changes. |
| | Product owner has reviewed feature or user story and approves. |
| | No high severity/noticeable bugs. |
| | All code and documentation are available in the repository. |