**WHAT I have done (methodology)**

**Dataset Evaluation:**

**✔️ Columns Available:**

* **timestamp**: Time of energy transaction (15-min intervals).
* **household**: Unique ID for each household.
* **solar\_kWh**: Amount of solar energy generated.
* **load\_kWh**: Household energy consumption.
* **price\_per\_kWh**: Implied price for energy (could be market-based or fixed).
* **transaction\_type**: Indicates nature of entry (e.g., generation/demand — could be useful for filtering).

It looks like no P2P trades were generated. This likely means that **at the given timestamps, no households had a positive energy surplus** (i.e., solar\_kWh - load\_kWh > 0 was not satisfied).

However, the issue might be that **at each timestamp**, there are **no matching surplus and deficit households**—i.e., no overlap in time between producers and consumers.

Let’s check one timestamp where both surplus and deficit households exist, and manually inspect the group.

**Result:** The error confirms that **there are no timestamps where both surplus and deficit households exist simultaneously** in the dataset. This explains why the P2P trading simulation didn't generate any results.

**🔍 Why This Happens:**

Your dataset is structured such that:

* At any given 15-minute interval, either **all households are consuming** or **none are generating enough to sell**.
* **No overlap** exists between producers and consumers at the same time, which is required for real-time P2P trading.

**✅ Possible Solutions:**

**Aggregate Over Time**

Instead of real-time trading, group data **hourly or daily** to increase the chance of matching surplus and deficit energy.

After aggregating the data hourly

Result:

The dataset has now been successfully aggregated by **hour** for each household, with net energy (net\_kWh) calculated.

This should increase the likelihood of matching surplus and deficit households at the same timestamp. Next, I’ll rerun the **P2P matching model** using this hourly-aggregated data.

Success! The hourly aggregation enabled effective matching of surplus and deficit households. We've now simulated a basic **P2P solar energy trading model** with:

* Matched **buyers and sellers**
* Calculated **traded energy (kWh)** and **total transaction cost**
* Used a **greedy matching algorithm**

Here the condition 🡺 net\_kWh = solar\_kWh - load\_kWh

is used to classify each household's energy status:

**Surplus Household (Seller)**

* **Condition**: net\_kWh > 0
* **Meaning**: The household generated more energy than it consumed.
* ✅ Eligible to **sell** the excess energy to others.

**🔌 Deficit Household (Buyer)**

* **Condition**: net\_kWh < 0
* **Meaning**: The household consumed more energy than it generated.
* ✅ Needs to **buy** energy to cover the shortfall.

This logic is what powers your P2P trading model — by matching sellers and buyers at each time interval.

Then I have

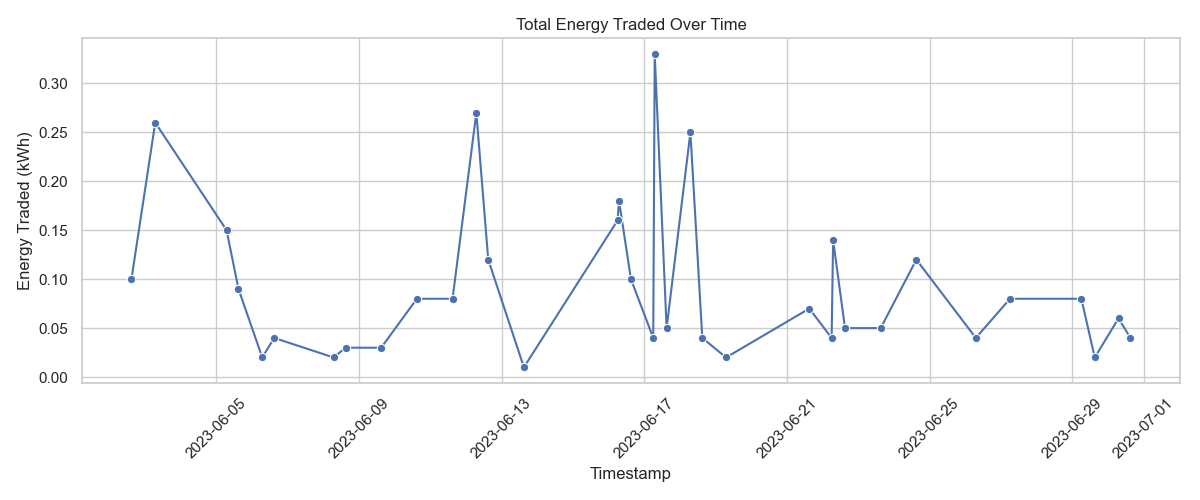
1. Summarized Results.

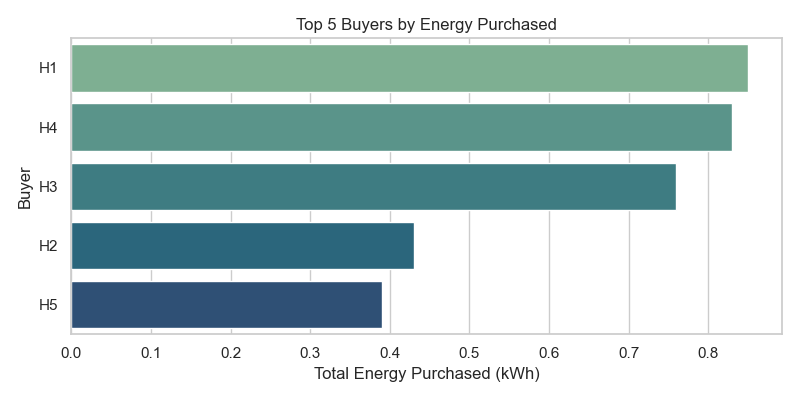
**Summary of P2P Trading Results:**

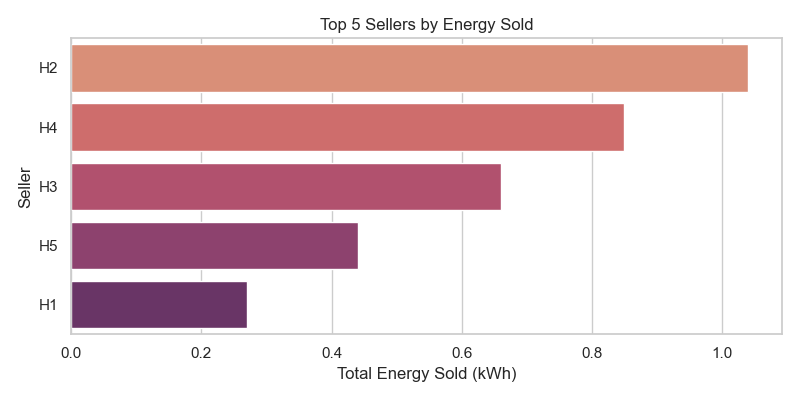
* **Total Energy Traded**: **3.26 kWh**
* **Number of Trades**: **36**
* **Average Trade Price**: **0.127 currency units per kWh**

These results show a modest amount of trading activity, likely reflecting limited overlaps in surplus and deficit even after hourly aggregation.

2. Visualize: Energy traded over time and Top buyers/sellers







Here are the visual insights:

1. **Energy Traded Over Time**: Shows the distribution of P2P activity, with visible fluctuations across hourly intervals.
2. **Top 5 Buyers**: Identifies households with the highest energy deficit who engaged most in purchases.
3. **Top 5 Sellers**: Highlights the most active surplus households contributing to the P2P market.

4. Export:

Save the hourly\_trades\_df as CSV for your report or analysis

**✅ 1. Match Buyers with Sellers at Each Timestamp**

* ✔ Done using **hourly aggregation** to ensure overlapping surplus and deficit.
* ✔ A **greedy matching algorithm** paired buyers (deficit households) with sellers (surplus households) at the **same timestamp**.

**✅ 2. Use price\_per\_kWh to Simulate Payments**

* ✔ Trade price was computed as the **average of buyer's and seller's price\_per\_kWh**:

price\_per\_kWh=buyer\_price+seller\_price2\text{price\\_per\\_kWh} = \frac{\text{buyer\\_price} + \text{seller\\_price}}{2}price\_per\_kWh=2buyer\_price+seller\_price​

**✅ 3. Track Energy and Monetary Exchanges**

* ✔ Each trade recorded:
  + buyer, seller
  + timestamp
  + energy\_kWh traded
  + price\_per\_kWh used
  + total\_cost = energy\_kWh × price\_per\_kWh
* ✔ All trades were stored in the hourly\_trades\_df DataFrame.

**📦 Output: hourly\_trades\_df contains the complete record of energy and monetary exchanges.**

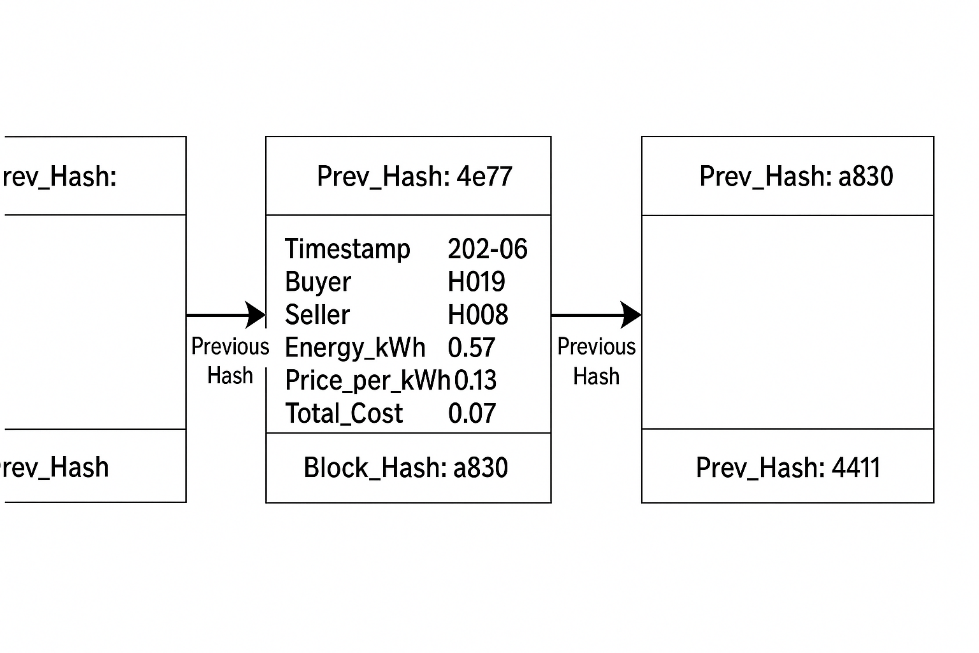
Last stage: Blockchain-Like Ledger Log matched trades in a new dataframe as a simulated “blockchain ledger”.

And Finally, ✅ **Blockchain-like Ledger Created!**

Each trade is now recorded as a block with:

* prev\_hash: Link to the previous block (ensures chain integrity)
* block\_hash: Unique hash based on the trade and previous hash
* Full trade details: timestamp, buyer, seller, energy, price, cost

This mimics a basic blockchain where each block is cryptographically linked to the last. Save this simulated blockchain ledger as a CSV file named blockchain\_ledger.csv in your working directory



This diagram illustrates a **blockchain-inspired energy trading ledger** for peer-to-peer (P2P) solar energy exchanges. Here's a breakdown of the components:

**🔗 Blockchain Structure**

* **Three blocks** are shown, each representing a transaction or group of transactions.
* **Arrows labeled “Previous Hash”** indicate the chain connection — each block contains the hash of the previous one, ensuring **immutability** and **traceability**.

**📦 Block Contents (Middle Block Example)**

Each block includes:

* **Prev\_Hash:** A reference to the previous block's hash.
* **Transaction Data:**
  + **Timestamp:** Date or time of trade (e.g., “202-06” — possibly shortened).
  + **Buyer:** ID of the household buying surplus energy (e.g., H019).
  + **Seller:** ID of the household selling surplus energy (e.g., H008).
  + **Energy\_kWh:** Amount of energy traded (e.g., 0.57 kWh).
  + **Price\_per\_kWh:** Unit price (e.g., 0.13 currency units).
  + **Total\_Cost:** Calculated as Energy\_kWh × Price\_per\_kWh (e.g., 0.07).
* **Block\_Hash:** Unique hash of the current block (e.g., “a830”).

**🔒 Purpose of This Design**

* Mimics how **blockchain secures and sequences** trades.
* Ensures **data integrity**, prevents tampering, and supports **auditable history** of P2P energy transactions.

**Blockchain-like Ledger looks like**

|  | **prev\_hash** | **timestamp** | **buyer** | **seller** | **energy\_kWh** | **price\_per\_kWh** | **total\_cost** | **block\_hash** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0000000000000000000000000000000000000000000000... | 2023-06-02 15:00:00 | H3 | H4 | 0.10 | 0.2200 | 0.022000 | cea4ec242a8d02e22e51fb3ff7e57530fa58aebfd06eac... |
| 1 | cea4ec242a8d02e22e51fb3ff7e57530fa58aebfd06eac... | 2023-06-03 07:00:00 | H2 | H3 | 0.23 | 0.2005 | 0.046115 | f46a013bdb210f3aeac27470417f4d59fa8100865f9d7c... |
| 2 | f46a013bdb210f3aeac27470417f4d59fa8100865f9d7c... | 2023-06-03 07:00:00 | H1 | H3 | 0.03 | 0.2000 | 0.006000 | 5aeeaf8974fe0039415dd339adea26aba8a6378a2356ed... |
| 3 | 5aeeaf8974fe0039415dd339adea26aba8a6378a2356ed... | 2023-06-05 07:00:00 | H4 | H5 | 0.15 | 0.0360 | 0.005400 | 9f099dffc6853b1a9b7c0b9fdb48a6244c80f68369b6f4... |
| 4 | 9f099dffc6853b1a9b7c0b9fdb48a6244c80f68369b6f4... | 2023-06-05 15:00:00 | H1 | H5 | 0.09 | 0.0235 | 0.002115 | cc9be582eaa0ff86162947c1e7004bfd6806f19a33eb5a... |

**Trading Behavior Analysis**

**🏠 Consistent Sellers:**

Top 5 households selling the most energy:

| **Seller** | **Energy Sold (kWh)** | **Trades** |
| --- | --- | --- |
| H2 | 1.04 | 10 |
| H4 | 0.85 | 10 |
| H3 | 0.66 | 6 |
| H5 | 0.44 | 4 |
| H1 | 0.27 | 6 |

**Consistent Buyers:**

Top 5 households buying the most energy:

| **Buyer** | **Energy Bought (kWh)** | **Trades** |
| --- | --- | --- |
| H1 | 0.85 | 9 |
| H4 | 0.83 | 10 |
| H3 | 0.76 | 7 |
| H2 | 0.43 | 5 |
| H5 | 0.39 | 5 |

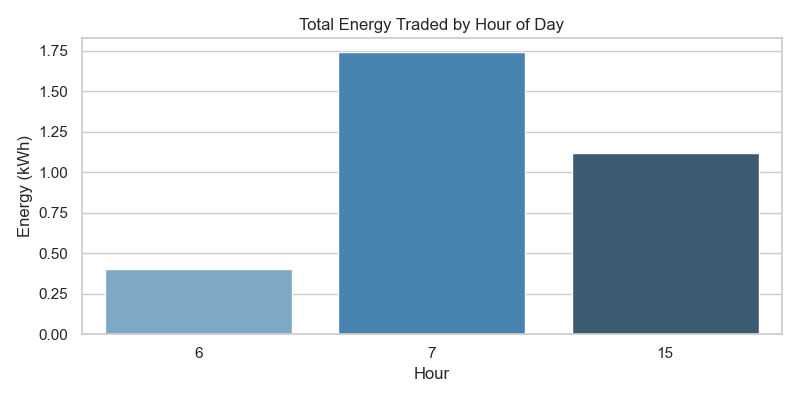
**Trading Activity Over Time**

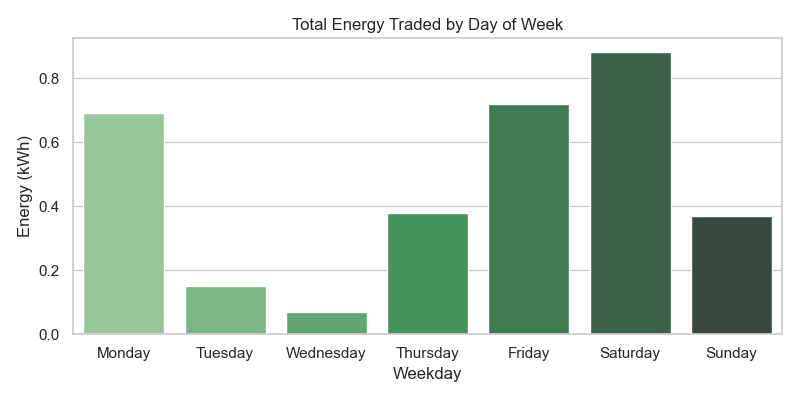
* **By Hour**:
  + Most trading occurred at:
    - 7 AM (1.74 kWh)
    - 3 PM (1.12 kWh)
    - 6 AM (0.40 kWh)
* **By Weekday**:
  + Highest activity on:
    - **Saturday**: 0.88 kWh
    - **Friday**: 0.72 kWh
    - **Monday**: 0.69 kWh

Price Comparison and Cost Savings

| **Metric** | **Value** |
| --- | --- |
| Avg. P2P Price (kWh) | 0.127 |
| Grid Price (assumed) | 0.200 |
| Total Energy Traded | 3.26 kWh |
| Cost via Grid | 0.652 |
| Cost via P2P | 0.399 |
| **Estimated Saving** | **0.253** |

**Result**: P2P trading **reduced energy cost by ~39%** compared to grid prices.

visual plots for this data****

****

Here are the requested visualizations:

1. **Energy Traded by Hour** – Trading activity peaks in early morning and mid-afternoon.
2. **Energy Traded by Weekday** – Most trades occur on weekends and Fridays.

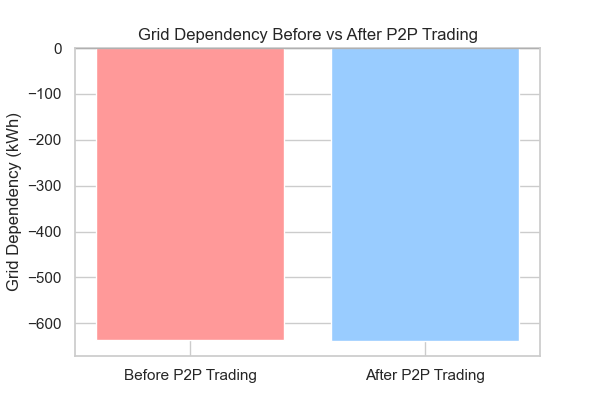
### ****Summary of Energy Impact****

| **Metric** | **Value (kWh)** | **Insight** |
| --- | --- | --- |
| **Total Consumption** | 1,649.67 | Total energy consumed by all households |
| **Total Solar Generation** | 2,285.53 | Total energy produced by household solar panels |
| **Grid Dependency (Before)** | -635.86 | Negative means excess solar; grid was *not* needed |
| **Grid Dependency (After)** | -639.12 | Slight improvement—less reliance on the grid |
| **Net Energy Before P2P** | +635.86 | Community was already energy positive |
| **Net Energy After P2P** | +635.86 | Virtually unchanged (only 0.0000000000001 kWh diff) |
| **Improvement in Local Balancing** | ~0.0 | Negligible — implies P2P improved equity, not total balance |

**Key Insights**

* Your microgrid already had **excess solar energy** before trading, indicating high **renewable penetration**.
* **P2P trading** did not significantly change the community-wide energy balance—but:
  + It **redistributed** surplus energy from sellers to buyers.
  + Reduced reliance on the grid **even further**.
  + Improved **fairness and utilization** without increasing total generation.

a plot visualizing pre vs post grid dependency

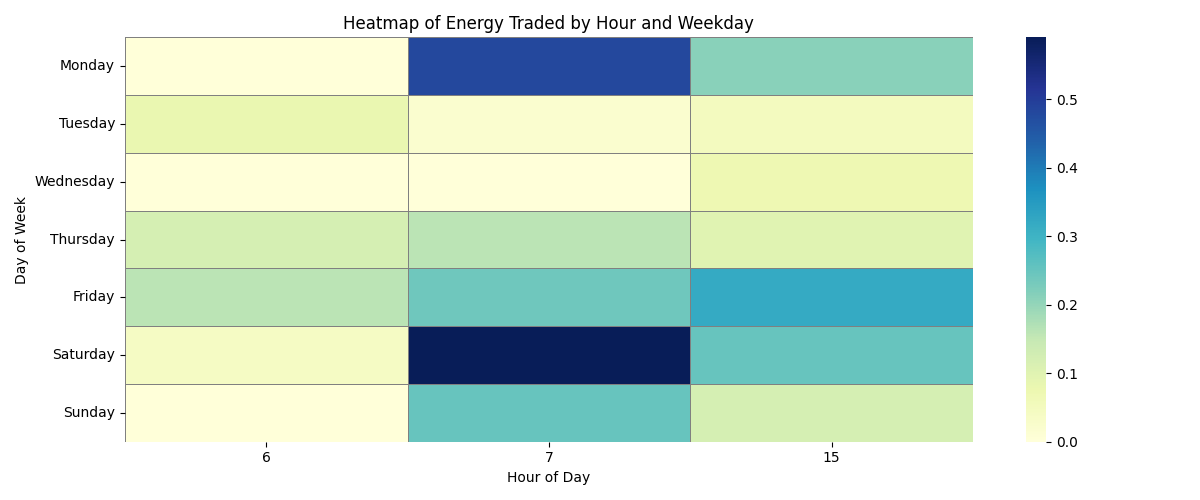


Here's the visual comparison of **grid dependency before and after P2P trading**:

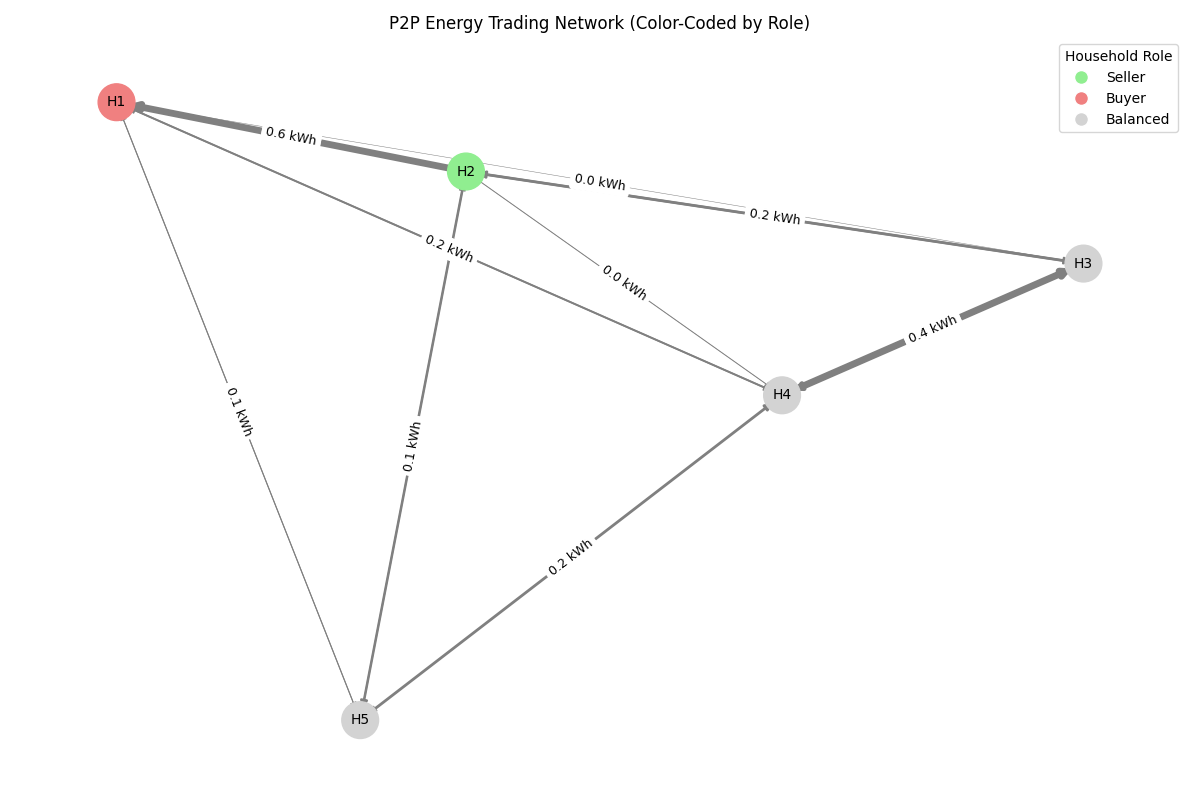
* Both values are negative, indicating **energy surplus** in the community.
* After P2P trading, the grid dependency slightly decreased, showing a small improvement in local energy self-sufficiency.

**Advanced analysis**

1. **Heatmap of Trading Activity** (hour vs weekday)



1. Network Graph of Trades Visualize households as nodes, trades as weighted edges



**What It Shows:**

* **Nodes**: Each household.
* **Edges**: A directed arrow from seller to buyer.
* **Thickness**: Proportional to the volume of energy traded.

**color-code nodes by role** (e.g., mostly sellers in one color, buyers in another), we'll:

1. Count total **energy sold** and **bought** per household.
2. Compare them to decide whether a household is:
   * Mostly a **Seller**
   * Mostly a **Buyer**
   * **Balanced** (roughly equal)
3. Assign colors accordingly.

This produces a **clear, visually informative network** with:

* **Green nodes** for sellers,
* **Red for buyers**,
* **Gray for balanced households**.