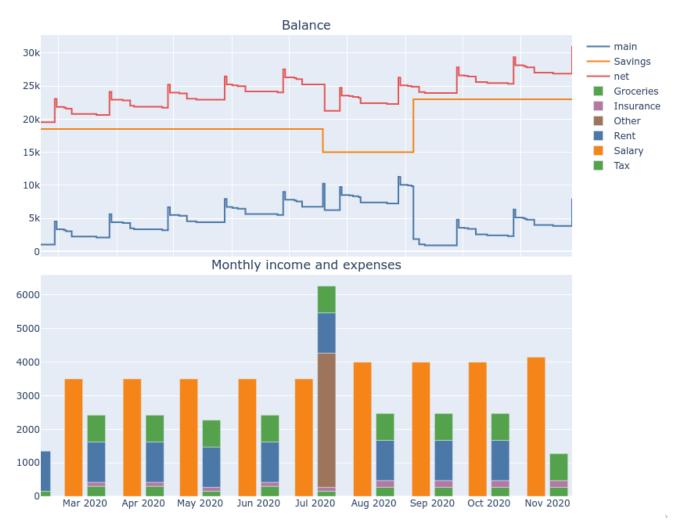# 1 CashPlot

CashPlot helps in finding trends in your financials and shows you where your money is going. It draws a graph of your bank balance and income/expenses over time.

It supports ING Bank export files (English/Dutch CSV).

1.1 Features

- Generates a time/money graph of:
  - Balance of the main and savings accounts
  - Net worth (sum of main + savings accounts)
  - Total period income/expenses grouped by category
- Easy zooming / scrolling / filtering by category

1.2 How to use

Create a Virtualenv directory in `venv/`:
```
virtualenv venv && source venv/bin/activate
```

Install the dependencies:
```
pip install -r requirements.txt
```

List the command-line options:
```
python -m cashplot --help
```

Run CashPlot:
```
python -m cashplot [--config <path/to/config.json>] <path/to/transactions.csv>
```

For example, to generate the graph shown above, run:
```
python -m cashplot --config examples/config.json examples/transactions.csv
```

When you run CashPlot, the graph is displayed in a browser window.
By default, the graph is zoomed in on the last year. You can double-click to zoom out and see all the years for which data is available. You can also filter out a category by clicking on its name, or double-click to filter out all other categories.

1.3 Configuration

Transactions are grouped into categories based on so-called "matching rules". These matching rules can be set in the configuration file (`config.json`). The following options are supported:

`savings_accounts` Array of names of savings accounts
`net_ignore_accounts` Array of names of savings accounts that don't count towards the "net worth" account
`starting_balances` The starting balance of the accounts before the transactions have occurred.
`match_rules` Array of matching rules
Each matching rule is an array of two elements, the category name and the rules table. This table can have the following attributes: `counter_name`, `counter_account`, and `description`. The values of these attributes are Python regexes. For example, `(?i)` can be used to match with case-insensitivity. Transactions are matched against all of the given attributes. It's possible to specify multiple matching rules per category.

The way to categorize transactions from/to a specific savings account is to add a matching rule with the category name set to the account name. CashPlot will treat transactions that match that rule as transactions between the main and savings account.

See the `examples/config.json` file for an example configuration.

1.4 Getting transactions exports

Transactions exports can be downloaded from the ING Bank online portal. When logged in, click on the download icon near the balance of the checking account. It's recommended to select the earliest possible start date, so that the calculated end balance corresponds to the current balance. Select "Comma-separated CSV (English)" and click on "Download".

1.5 Limitations

Take the graph data with a grain of salt; always carefully check before taking conclusions.

Only one main (checking) account is supported. The balance of the savings accounts is calculated by the transactions between the main and savings accounts. If the savings accounts are modified by other sources (for example interest), this won't get taken into account.

Transactions made on the same day can get reordered, which might result in a negative balance on the graph when it did not happen in actuality.

1.6 Development

This program is tested with PyTest and auto-formatted with AutoPEP8 (see `dev-requirements.txt`).

1.7 Feedback/improvements

If you have any issues, questions, or ideas on how this can be improved, feel free to create an issue on GitHub or send me an email (see my GitHub profile).