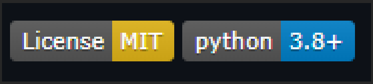
MCAP - Profile-Activity Matching System 📊



A Python package designed for optimal profile-to-activity assignment based on skills matching, developed by Abdel YEZZA (Ph.D). This solution employs advanced algorithms to maximize the correspondence between required activity competencies and available profile skills.

It is designed to provide a comprehensive solution for profile-activity matching, offering multiple models, flexible scaling options, complete web application, streamlit web interface, a robust logging system, a detailed validation of input data, and customizable processing of MCAP functions (sum, mean, Euclidean and any custom function).

This project is built on the two following non-technical articles:

> 1. UN NOUVEAU MODELE POUR AFFECTER LES PROFILS ADEQUATS - by Abdel YEZZA (Ph.D) - 2024
> 2. UNE NOUVELLE FAÇON D'AFFECTATION DES PROFILS AUX ACTIVITES - by Abdel YEZZA (Ph.D) - 2022

🎯 Key Features

> - **Skills Matrix Analysis**: Process and analyze competency-activity (MCA) and competency-profile (MCP) matrices
> - **Multiple Model Support**: Five different matching models available (model1 through model5 or any custom function)
> - **Flexible Scaling**: Support for different scale types (0-1, free)
> - **Web Interface**: Built-in web application using FastAPI and Streamlit
> - **Detailed Logging**: Comprehensive logging system for tracking operations
> - **Data Validation**: Robust input validation and error handling
> - **Customizable Processing**: Support for different MCAP functions (sum, mean, euclidean and any custom function)

Installation

1. Clone the repository:

```
git clone [repository-url]
cd profiles_assignment
```

2. Create and activate a virtual environment (recommended):

```
# install venv on Linux/MacOS:
python -m venv venv
source venv/bin/activate
# to activate on Windows:
venv\Scripts\activate
Install dependencies:
pip install -r requirements.txt
```

Dependencies

- streamlit >= 1.24.0
- pandas >= 1.5.0
- scikit-learn
- matplotlib
- fastapi >= 0.104.0
- uvicorn >= 0.24.0
- python-dotenv >= 1.0.0
- python-multipart >= 0.0.6
- sqlalchemy >= 2.0.23

Usage

Command Line Interface

Basic usage:

```
python main.py
```

Advanced usage with custom parameters:

```
python main.py --mca path/to/mca.csv --mcp path/to/mcp.csv --model
model_name --scale scale_type --mcap mcap_function
```

Example:

```
python main.py --mca .\data\input\mca.csv --mcp .\data\input\mcp.csv --
model model5 --scale 0-1 --mcap sqrt
```

Command Line Arguments (Console case)

- --mca: Path to the MCA (Matrix Competency-Activity) file
- --mcp: Path to the MCP (Matrix Competency-Profile) file
- --model: Model selection (model1, model2, model3, model4, model5 or your own model function)
- --scale: Scale type (0-1 or free)
- --mcap: MCAP function type (sum, mean, sqrt or a custom MCAP function)

Input File Formats

MCA (Competency-Activity Matrix)

```
Activity,Comp1,Comp2,Comp3
Activity1,0.8,0.6,0.7
Activity2,0.5,0.9,0.4
```

MCP (Competency-Profile Matrix)

```
Profile,Comp1,Comp2,Comp3
Profile1,0.7,0.8,0.6
Profile2,0.9,0.5,0.8
```

Streamlit demo application

streamlit.cmd run .\src\streamlit\app.py

You should have three menu items:

1. Start page
2. Test application
3. Interactive application

Web application

1. Backend

```
# go to backend folder
cd web/backend

# Install dépendancies if any
pip install -r requirements.txt

# Run the serveur with uvicorn
uvicorn main:app --reload --log-level debug
```

2. Frontend

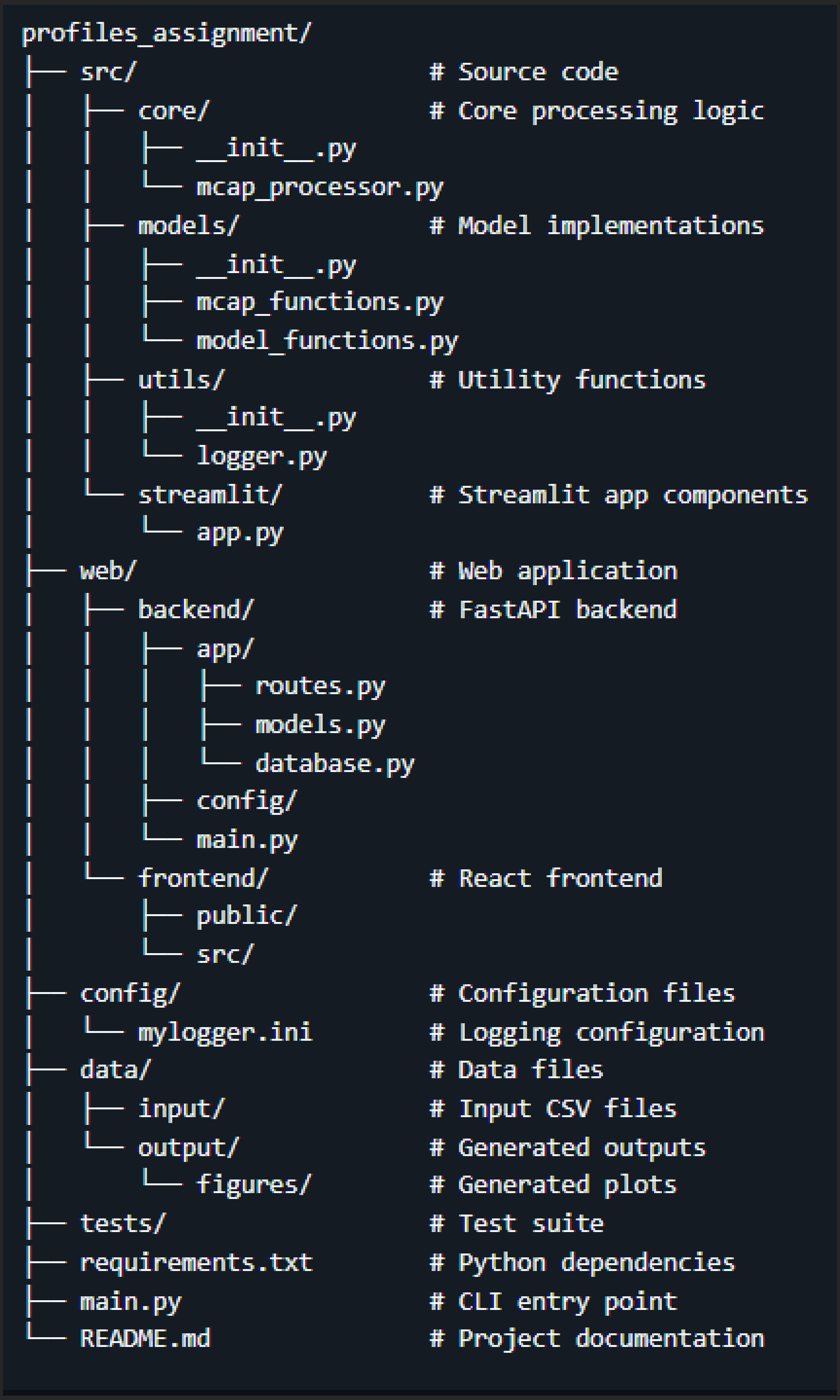
```
# go to frontend folder
cd web/frontend

# Install dépendancies if any
npm install

# Run the dev server
npm start
```

You should get a message like:

You can now view mcap-frontend in the browser.

```
  Local:            http://localhost:3001
  On Your Network:  http://192.168.1.19:3001
```

Project Structure



```
profiles_assignment/
├── src/                    # Source code
│   ├── core/               # Core processing logic
│   │   ├── __init__.py
│   │   └── mcap_processor.py
│   ├── models/             # Model implementations
│   │   ├── __init__.py
│   │   ├── mcap_functions.py
│   │   └── model_functions.py
│   ├── utils/              # Utility functions
│   │   ├── __init__.py
│   │   └── logger.py
│   └── streamlit/          # Streamlit app components
│       └── app.py
├── web/                    # Web application
│   ├── backend/            # FastAPI backend
│   │   ├── app/
│   │   │   ├── routes.py
│   │   │   ├── models.py
│   │   │   └── database.py
│   │   ├── config/
│   │   └── main.py
│   └── frontend/           # React frontend
│       ├── public/
│       └── src/
├── config/                 # Configuration files
│   └── mylogger.ini        # Logging configuration
├── data/                   # Data files
│   ├── input/              # Input CSV files
│   └── output/             # Generated outputs
│       └── figures/        # Generated plots
├── tests/                  # Test suite
├── requirements.txt        # Python dependencies
├── main.py                 # CLI entry point
└── README.md               # Project documentation
```