Scripts Page

High-level

This is the web page where staff can run the different Python scripts without needing the command line. Each script is shown as a card with its name and a Run button. When you press the button, the script runs on the server and the results appear directly below it.

Technical

- The page is built with HTML and Bootstrap for styling.
- It lists all scripts from the server's scripts dictionary (provided by Flask/Jinja).
- Each script card has a Run button that submits a form back to the server with the script key and selected language.
- Results from the script run are displayed underneath the card. Messages are styled depending on the icon in the output:
    - ❌ → error (red box)
    - ⚠️ → warning (yellow box)
    - ✅ / ✔ → success (green box)
    - 👶 , 👥 , 📦 , 🔍 → info (blue box)
    - others → generic info (gray/blue box)

Python Scripts

Configuration

All scripts load their settings through a shared config loader.

- Main settings are kept in system_config.json (with a default file available if needed).
- The Bitrix webhook URL is always read from an environment variable (B24_WEBHOOK_URL) so it is never stored on disk.
- On platforms like Render, the config file can be stored in a persistent location defined by CONFIG_DIR.

Child_deduplication.py

High-level
Looks for families (beneficiaries) that might be duplicated because they have the same set of children (same child names and dates of birth). Flags them so they can be checked.

Technical

- Reads setup info from system_config.json.
- Gets all children from Bitrix, groups them by their parent (beneficiary).
- Makes a “signature” for each family based on children’s names + DoB.
- If two families share the same signature, marks them as possible duplicates.
- Skips those still in the Registration stage.
- Updates others with:
    - duplicate flag (unique / possible duplicate)
    - names of the other matching families.

Deduplicate_Payments.py

High-level
Checks all payment records and deletes extra copies where the same person and project appear more than once.

Technical

- Gets all payments from Bitrix.
- Checks if the same National ID + Project Type appears more than once.
- Keeps the first one, deletes the others.
- Prints results to the screen.

Duplicate_Check.py

High-level
Checks people in the Registration stage to see if they are already registered. Uses either their National ID or their full name. Marks them as “Duplicate” or “Unique” with a reason.

Technical

- Loads all beneficiaries in Registration.
- If a person has no duplicate flag yet, compares them with others.
- Looks for same National ID or same normalized name.
- Updates the record with:
    - Duplicate flag (Duplicate / Unique)
    - Reason text (e.g., “Duplicate National ID”).

Eligibility_Check.py

High-level
Checks which beneficiaries qualify for certain projects. If they meet the project rules, marks them as eligible and adds the project name. If not, clears their eligibility.

Technical

- Gets projects in the target stage.
- Gets all beneficiaries in Verified or Eligible stages.
- Compares fields between project and beneficiary (for example, both must have “Yes” in certain fields).
- Updates each beneficiary with:
    - Eligible flag (Yes/No)
    - Program count and names
    - Moves back to Verified if they shouldn’t be Eligible anymore.

resetting_beneficiaries.py

High-level

Resets beneficiaries who finished a project so they can join new ones later. Saves the old project name in their history, removes the project link, and moves them back to Verified stage.

Technical

- Gets beneficiaries in the Completed stage.
- Finds their linked project and saves its name into a “previous projects” field.
- Clears the project link field.
- Changes their stage back to Verified.

Notes

- All scripts handle Bitrix’s paging (fetching results in batches).
- Running scripts again won’t cause problems — they just set the same values again.