# Технічне завдання — Price Complex Processor (PriceComPro)

**Призначення системи:  
PriceComPro — це веб-орієнтований програмний комплекс на Python для збору, уніфікації, обробки та експорту прайс-листів постачальників.**

Нижче — пропрацьований, деталізований ТЗ, яке можна відразу брати в роботу. Я позначив **MVP** (мінімально життєздатний продукт) та **Phase 2+** (після релізу) для пріоритизації.

**1) Цілі та результат**

**Мета:** дати користувачу веб‑інструмент для:

* отримання прайсів з різних джерел (файли/FTP/HTTP/парсинг);
* уніфікації даних у тимчасову таблицю **«Товари постачальників»** з можливістю додавати власні поля та правила обробки;
* напів/автоматичного оновлення **«Наших товарів»**;
* створення кастомних експортів з фільтрами/умовами;
* автономної роботи за розкладом.

**Ключовий ідентифікатор товарів:** артикул (з підтримкою прив’язки «Артикул постачальника ↔ Внутрішній артикул/SKU»).

**2) Технологічний стек**

* **Backend (API):** Python 3.11+, **FastAPI** (MVP)
* **ORM/БД:** **PostgreSQL** 14+ з SQLAlchemy 2.x, Alembic міграції (MVP)
* **Черги/планування:** **Celery** + Redis; періодика — **APScheduler** або Celery beat (MVP)
* **Парсинг:** pandas, openpyxl, lxml/BeautifulSoup, requests, ftplib (MVP)
* **Фронтенд:** React + TypeScript + shadcn/ui (або простіше — FastAPI + Jinja2 у MVP, React — Phase 2)
* **Файли:** локальне сховище + опційно S3‑сумісне (Phase 2)
* **Авторизація:** OAuth2 (JWT), ролі (admin, manager, viewer) (MVP)
* **Контейнеризація:** Docker + docker‑compose (MVP)
* **Логи/моніторинг:** structlog + Prometheus + Grafana (Phase 2)

**3) Модель даних (схема БД)**

**3.1. Вхідні прайси (вхідні сутності)**

* **suppliers** (Постачальники) — *MVP*
  + id (PK), name, code, contacts(jsonb), ftp/http creds (зашифровано), active
* **price\_lists** (Прайс‑листи) — *MVP*
  + id (PK), supplier\_id (FK), name, source\_type(enum: local, ftp, http, parser)
  + source\_config(jsonb): шлях/URL, креденшали, парсер/селектора
  + format(enum: csv, xlsx, xml, json, html), charset, delimiter, sheet, header\_row, skip\_rows, etc.
  + mapping(jsonb): відповідність «колонка джерела → уніфіковане поле»
  + schedule(cron), last\_run\_at, last\_status, active
* **supplier\_products** (Товари постачальників, тимчасова уніфікована) — *MVP*
  + id (PK)
  + supplier\_id, price\_list\_id, import\_batch\_id
  + supplier\_sku (артикул постачальника) [unique per supplier?], name, description
  + brand\_raw, category\_raw
  + price\_raw, currency\_raw, qty\_raw, uom\_raw
  + normalized fields: brand\_id(FK), category\_id(FK), currency\_id(FK)
  + price\_converted, qty (int/decimal), availability\_status
  + custom\_fields jsonb (для користувацьких полів/формул)
  + hashes (row\_hash для дедуплікації), updated\_at
* **import\_batches** — історія операцій завантаження/обробки (MVP)
  + id, price\_list\_id, started\_at, finished\_at, status, stats(jsonb), log\_path

**3.2. Локальний каталог**

* **currencies** — *MVP*
  + id, code (UAH, PLN, USD…), rate\_to\_base, updated\_at, manual\_override(bool)
* **manufacturers** — *MVP*
  + id, name, aliases (jsonb)
* **categories** — *MVP*
  + id, name, parent\_id, path, external\_mappings(jsonb)
* **products** («Наші товари») — *MVP*
  + id (PK), sku (унікальний внутрішній), supplier\_link(jsonb: list of supplier\_sku), name, description\_html
  + brand\_id, category\_id
  + price, currency\_id, stock, uom
  + attributes jsonb (для гнучких фіч), images jsonb
  + status(enum: draft, active, archived)
  + audit: created\_at/by, updated\_at/by
* **field\_mappings** (повторно використовувані мапінги колонок) — Phase 2
* **export\_profiles** — налаштування експортів (MVP)
  + id, name, target\_format(csv/xlsx/xml/json), filters(jsonb), field\_map(jsonb), schedule(cron), last\_run\_at, active

Примітка: **jsonb** застосовуємо для гнучких користувацьких полів та мапінгів; критичні для фільтрів/індексації поля зберігаємо окремими стовпцями.

**4) Джерела отримання прайсів**

**Підтримувані в MVP:**

* Локальне завантаження файлу через UI (csv/xlsx/xml/json)
* FTP(S): host, port, path, пасивний режим, маска файлу
* HTTP(S): пряме посилання / авторизація, basic/bearer
* Парсинг HTML: URL + CSS/XPath‑селектори, таблиці, атрибути

**Опції нормалізації при імпорті:**

* Кодування/роздільник/лист (Excel)/рядок заголовку
* Трим/очистка HTML/перетворення одиниць
* Мапінг колон → уніфіковані поля
* Валюта за замовч., курс (з таблиці валют)
* Політики артикулу: trim/upper/replace (набір правил)

**Логи й діагностика:** прев’ю перших N рядків, підсумкові метрики (рядків, пропущених, оновлених, нових), архів сирого файлу.

**5) Уніфікація, кастомні поля та правила**

**5.1. Мапінги (UI)**

* Майстер мапінгу: підхоплення заголовків колонок, призначення уніфікованим полям.
* Збереження мапінгу в **price\_lists.mapping** для повторного використання.

**5.2. Користувацькі поля**

* Додавання полів у **supplier\_products.custom\_fields** через UI (типи: text, number, boolean, enum, formula).
* Заповнення з: константи / мапа з колонки / формула.

**5.3. Правила обробки цін/даних (rules engine)**

* **Типи правил (MVP):**
  + множення на курс валюти (currencies.rate\_to\_base або вибрана)
  + множення/додавання фіксованого коефіцієнта/надбавки
  + округлення (до 0.01/0.1/1/5/9/99 закінчень)
  + умови IF (price < X → coeff A; інакше coeff B)
  + мапа значень (бренд/категорія → коефіцієнт)
* Правила застосовуються в послідовності (pipeline) до **supplier\_products** в межах **import\_batch**.
* Протоколювання правил у batch‑логи для відтворюваності.

**6) Поведінка таблиці «Товари постачальників»**

* **Повне оновлення (агрегація «Усі прайси»):** перед завантаженням **supplier\_products** очищається (опція «очистити перед імпортом» = True). *MVP*
* **Часткове оновлення (окремий прайс):** без очистки; товари додаються або оновлюються за ключем (supplier\_id, supplier\_sku); зберігається останній стан і batch\_id. *MVP*
* **Дедуплікація:** контроль дубльованих рядків через row\_hash та/або артикул. *MVP*
* **Валідація:** обов’язкові: supplier\_sku, name, ціна/валюта (або правила підстановки). Помилки — в окрему «помилки імпорту». *MVP*

**7) Редагування в UI**

* **Показ таблиць:** пагінація, пошук, фільтри по постачальнику/категорії/бренду/ціні/наявності. *MVP*
* **Редагування рядка:** інлайн або модалка; валідація; запис змін з аудитом. *MVP*
* **Масові операції:** вибір рядків → застосувати правило/значення/коефіцієнт. *Phase 2*

**8) Оновлення «Наших товарів»**

* **Мапінг відповідностей:**
  + автоматичний лінк «supplier\_sku → products.sku» за таблицею відповідностей (products.supplier\_link),
  + або «прив’язати вручну». *MVP*
* **Стратегії синхронізації (налаштовувані):**
  + тільки ціна/склад,
  + ціна/склад/назва/опис (з прапорцями «не переписувати, якщо ручна правка»),
  + створення нових продуктів, якщо немає відповідності. *MVP*
* **Конфлікти:** показ різниці, вибір політики (брати з постачальника / зберегти локальне). *MVP*
* **Історія змін (audit trail):** хто/коли/що змінив. *MVP*

**9) Експорти**

* **Export Profile:** назва, формат (csv/xlsx/xml/json), **field\_map** (які поля і як називати), **filters** (SQL‑подібні або builder у UI), сортування, ліміт. *MVP*
* **Мультипрофільність:** можна створити кілька профілів під різні маркетплейси/сайти. *MVP*
* **Розклад запуску:** cron у профілі (черга Celery). *MVP*
* **Прев’ю & завантаження:** перегляд перших N рядків, кнопка «Завантажити». *MVP*

**10) Автономний режим**

* **Job chain:**
  1. отримати прайси (за розкладом),
  2. уніфікація + правила,
  3. оновити «Товари постачальників»,
  4. синхронізувати «Наші товари»,
  5. виконати експорт(и).  
     Усе — через Celery таски з ретраями та нотифікаціями. *MVP*
* **Нотифікації:** email/Telegram/Webhook про статуси batch/import/export. *Phase 2*

**11) Безпека та права**

* **Ролі:** admin (усе), manager (імпорт/редагування/експорт), viewer (перегляд). *MVP*
* **Зберігання секретів:** креденшали FTP/HTTP — шифрування (наприклад, Fernet з майстер‑ключем із env). *MVP*
* **Логування доступів/дій:** IP, user, операції над даними. *Phase 2*

**12) Продуктивність**

* Індекси: (supplier\_id, supplier\_sku), sku, brand\_id, category\_id, price, updated\_at. *MVP*
* Чанкування імпорту (наприклад, 5–20 тис. рядків у пачці). *MVP*
* Потокова обробка великих CSV (iterator). *MVP*

**13) Міграції та резервування**

* **Alembic** для схем. *MVP*
* Бекапи БД (щоденно) + архів batch‑файлів. *MVP*
* Версіонування експорт‑профілів та мапінгів. *Phase 2*

**14) Інтерфейс (MVP макети екранів)**

1. **Дашборд:** останні імпорти/експорти, статуси, кнопки «Запустити зараз».
2. **Постачальники:** список/форма; перевірка підключення FTP/HTTP.
3. **Прайс‑листи:** джерело, формат, мапінг, розклад; «Тестовий імпорт».
4. **Мапінг колонок:** таблиця колонок із прев’ю; збереження мапінгу.
5. **Правила:** конструктор правил (черга операцій).
6. **Товари постачальників:** грід, фільтри, редагування, масові дії (Phase 2).
7. **Наші товари:** грід, прив’язки, синхронізація, історія змін.
8. **Експорти:** профілі, прев’ю, завантаження, розклад.
9. **Журнали/батчі:** історія із деталями, помилки, завантаження логів/сирих файлів.
10. **Валюти:** список, курс базової валюти, ручні правки.

**15) Валюти й курси**

* Базова валюта (налаштування системи).
* Курсова таблиця; оновлення вручну (MVP) + підключення до API (Phase 2).
* У правилах — вибір курсу: системний/постачальника/ручний.

**16) Валідації та помилки**

* **Під час імпорту:** порожній артикул, нечислова ціна, некоректна валюта → рядок у «помилки імпорту» з описом; основний потік не зупиняємо. *MVP*
* **При синхронізації:** конфліктні поля → статус «потребує підтвердження» + UI для вирішення. *MVP*

**17) Тестування та приймальні критерії**

**Юніт‑тести (MVP):** парсери (csv/xlsx/xml/html), правила, мапінг, дедуп, конвеєр імпорту.  
**Інтеграційні:** ланцюжок від джерела до «Товари постачальників», далі — в «Наші товари», далі — експорт.  
**Критерії приймання (MVP):**

1. Можна створити постачальника, додати прайс‑лист (csv/xlsx), задати мапінг, виконати імпорт.
2. Дані з’являються в «Товари постачальників», застосовані правила цін.
3. Редагування рядка зберігається.
4. Синхронізація оновлює «Наші товари» за обраною стратегією.
5. Створюється експорт‑файл за профілем і завантажується.
6. Працює планувальник (запуск за розкладом) і логуються батчі.

**18) MVP обсяг (узагальнено)**

* БД + міграції.
* CRUD для постачальників/прайс‑листів/валют/брендів/категорій/«Наших товарів».
* Імпорт із локального файлу/FTP/HTTP + простий HTML‑парсер таблиці.
* Мапінг колонок + базові правила перетворення цін.
* Поведінка «очистити/не очищати» для «Товари постачальників».
* Редагування рядків у «Товари постачальників» і «Наших товарах».
* Синхронізація з вибірковими полями.
* Експорт‑профілі (csv/xlsx/json) з фільтрами.
* Планувальник + історія батчів + прості логи.
* Ролі/логін, збереження секретів.

**19) Phase 2+ (на розвиток)**

* Просунутий конструктор правил (формули, вирази, шаблони цін за сегментами).
* Масові операції, імпорт/експорт шаблонів мапінгу.
* Інтеграції: API курсу валют, marketplace коннектори, S3‑сховище.
* Версіонування продуктів та відкат.
* Нотифікації (email/Telegram/Webhooks).
* Аналітика: відхилення цін, націнки по брендам/категоріям, динаміка.
* Мульти‑валютні прайси з автоматичним визначенням валюти.

**20) Нюанси та припущення**

* Артикул постачальника може містити пробіли/реєстр → нормалізація (trim/upper, конфігурується).
* Внутрішній **sku** відрізняється від **supplier\_sku**; можлива багато‑до‑багатьох прив’язка (один наш товар — декілька артикулів у різних постачальників).
* Категорії/бренди можуть нормалізуватись через довідники та таблиці відповідностей/аліасів.

**21) API (скорочено)**

* POST /auth/login
* GET/POST /suppliers, GET/POST /price-lists, POST /price-lists/{id}/test-import, POST /price-lists/{id}/run
* GET/PUT /supplier-products, GET/PUT /products
* GET/POST /export-profiles, POST /export-profiles/{id}/run, GET /exports/{id}/download
* GET/POST /rules (якщо окремо), GET/POST /currencies, POST /scheduler/run-chain

**22) Розгортання**

* Docker Compose: web(api), worker(celery), redis, postgres, nginx.
* .env для секретів; майстер‑ключ шифрування у секретах оточення.

**23) Наступні кроки**

1. Підтвердити структуру БД (вище) та поля уніфікованої таблиці.
2. Затвердити перелік **уніфікованих полів** для supplier\_products (мінімум: supplier\_sku, name, brand\_raw/brand\_id, category\_raw/category\_id, price\_raw/price\_converted, currency\_raw/currency\_id, qty/availability).
3. Узгодити сценарії синхронізації з «Нашими товарами» (які поля міняємо автоматично).
4. Визначити 1–2 типових приклади прайсів для тестового імпорту (CSV/XLSX) та приклад профілю експорту.
5. Старт розробки MVP по частинах: БД → імпорт → правила → «Товари постачальників» → «Наші товари» → експорт → планувальник.