# 요구사항 정의서 – 아이템 시세 예측 시스템

## 1. 개요

### 1.1 목적

이 프로젝트는 게임 내 아이템의 시세 데이터를 수집하고, 시계열 예측 모델을 활용하여 향후 가격을 예측하며, 웹 페이지를 통해 유저에게 시각적으로 제공하는 것을 목적으로 한다.

## 2. 요구사항

### 2.1 기능 요구사항

* 2.1.1 아이템 시세 수집: API에서 주기적으로 시세 데이터를 수집한다.
* 2.1.2 수집 데이터 저장: 수집된 시세 데이터를 PostgreSQL 또는 InfluxDB에 저장한다.
* 2.1.3 예측 모델 학습 및 적용: ARIMA, LSTM, GRU 중 선택된 시계열 모델을 사용하여 학습 및 예측 수행.
* 2.1.4 예측 결과 저장: 예측 결과를 DB에 저장하여 조회 가능하도록 한다.
* 2.1.5 예측 결과 API 제공: FastAPI를 통해 예측 결과를 JSON 형태로 제공한다.
* 2.1.6 웹 페이지 시각화: Flask를 통해 유저가 아이템을 검색하고 과거 시세 및 예측 그래프를 볼 수 있도록 한다.
* 2.1.7 자동화 처리: Airflow를 통해 데이터 수집 및 예측을 정해진 주기로 자동 수행한다.
* 2.1.8 수동 트리거 기능: 수동으로 학습 또는 예측을 요청할 수 있는 API 또는 관리 도구 제공 (선택).

### 2.2 비기능 요구사항

* 2.2.1 모듈화된 구조: 예측 알고리즘 및 데이터 소스를 유연하게 교체할 수 있도록 컴포넌트를 분리한다.
* 2.2.2 정확도 확보: 예측 모델 성능을 향상시키기 위해 충분한 과거 데이터를 확보하고 이상치를 필터링한다.
* 2.2.3 확장 가능성: 아이템 카테고리 추가, 유저 로그인 기능, 알림 기능 등 확장이 가능한 구조로 설계한다.
* 2.2.4 Docker 기반 배포: FastAPI, Flask, DB, Airflow 등은 각각 컨테이너로 구성해 이식성을 높인다.
* 2.2.5 오류 로깅 및 모니터링: 수집 실패, 예측 오류 등 시스템 이상을 로깅하고 알림 기능을 연동할 수 있도록 준비한다.
* 2.2.6 성능 고려: 수천 개의 아이템 데이터를 다룰 수 있도록 배치 처리 및 비동기 처리 등 성능을 고려한 구조로 설계한다.
* 2.2.7 보안: API Key, DB 인증 정보는 환경변수 또는 Vault로 관리하며 외부 노출을 방지한다.
* 2.2.8 정기 예측 주기: 기본적으로 하루 1회 예측 실행, 필요 시 DAG로 커스터마이징 가능해야 한다.

## 3. 용어 정의

|  |  |
| --- | --- |
| 용어 | 설명 |
| LostArk API | 게임 공식 API 또는 크롤링 대상 URL |
| ARIMA / LSTM / GRU | 시계열 예측에 사용되는 대표적인 알고리즘 |
| DAG | Airflow의 워크플로우 단위 구성요소 |
| FastAPI | 비동기 REST API 서버 프레임워크 |
| Flask | 프론트엔드 웹 UI용 파이썬 마이크로 프레임워크 |
| PostgreSQL | 관계형 데이터베이스(RDB) |
| InfluxDB | 시계열(Time-Series) 데이터베이스 |

## 4. 제약 조건

* 로스트아크 API의 호출 제한 및 이용 정책을 반드시 준수해야 함
* 모든 기능은 로컬 또는 단일 서버 기반에서 작동 가능해야 하며, 클라우드 의존성은 낮게 유지
* LSTM/GRU는 GPU 없이 동작할 수 있는 수준에서 설계할 것 (모델 경량화 고려)