

Дизајн и Архитектура на Софтвер

Домашна Работа 1

1. Формирање на тимот

Име и Презиме	Индекс
Џејда Хусеини	221181
Дарио Спиров	221276
Бојан Ристевски	221251

2. Краток Опис

- Нашиот проект има за цел да развие цевковод за обработка на податоци за анализа на берзанските податоци поврзани со Македонската берза. Користејќи го стилот на архитектурата на цевки и филтри, ќе го автоматизираме извлекувањето, трансформацијата и вчитувањето (ETL) на дневните податоци за акциите за сите котирали издавачи, фокусирајќи се на последните десет години. Овој систем ќе обезбеди податоците да се ажурирани, конзистентно форматирани и подготвени за понатамошна анализа. На оваа основа ќе се надградат идните задачи, проширувајќи ја апликацијата за поддршка на различни анализи и прашања за македонските акции.

3. Дефинирање функционални и нефункционални барања

- Функционални барања:

Овие барања специфицираат што системот мора да направи за да ја исполни својата цел. Бидејќи овој проект вклучува прибирање, трансформирање и складирање на податоци за акции, да ги разгледаме сите функции одделно.

1. Прибирање на податоци

Барање: Системот треба да прибере листа на кодови на издавачи од веб-страницата на Македонската Берза.

Опис: Системот треба да користи веб-скрапинг за да ги собере сите кодови на издавачи наведени на берзата. Треба да ги исклучи сите кодови што се однесуваат на обврзници или содржат броеви.

Причина: Преземената листа е потребна за да се идентификуваат издавачите

чи податоци треба да се соберат, обезбедувајќи дека сите релевантни податоци за акции се вклучени.

2. Проверка на база на податоци и ажурирање на податоци

Барање: Системот треба да ја провери базата на податоци за најновиот датум на зачувани податоци за секој издавач.

Опис: За секој издавач, доколку нема податоци, системот треба да прибере дневни податоци од најмалку последните 10 години. Ако има податоци, треба да ги преземе само недостасувачките податоци по последниот зачуван датум.

Причина: Ова овозможува базата на податоци да содржи ажурирани и комплетни податоци за секој издавач.

3. Трансформација и форматирање на податоци

Барање: Системот треба да ги форматира и трансформира податоците пред нивно складирање.

Опис: Датумите треба да следат конзистентен формат (на пример, `YYYY-MM-DD`), а цените треба да користат запирки за илјадници и точки за децимали (на пример, `21,600.00`).

Причина: Конзистентното форматирање овозможува лесно пребарување и интегритет на податоците во базата.

4. Складирање на податоци

Барање: Системот треба да ги складира форматираните податоци во структурирана база на податоци.

Опис: Податоците треба да се внесат во базата на податоци со уникатно ограничување на комбинацијата од кодот на издавачот и датумот, за да се избегнат дупликати.

Причина: Правилната структура на складирање обезбедува дека податоците се организирани и дупликатите се избегнати.

5. Пристап на корисници

Барање: Корисниците треба да можат да пребаруваат и да ги добијат историските и тековните податоци за акции.

Опис: Системот треба да овозможи корисниците да ги гледаат податоците за акции за специфични издавачи, филтрирајќи по временски опсег доколку е потребно.

Причина: Ова им овозможува на корисниците да вршат анализи или да гледаат податоци без директен пристап до базата на податоци.

- Нефункциски барања:

Нефункциските барања ги дефинираат квалитетни атрибути што системот треба да ги исполни, како перформанси, скалабилност и одржливост.

1. Перформанси

Барање: Системот треба да ги презема и ажурира податоците за акции во минимален временски рок.

Опис: Времето за прибирање и обработка на податоците не треба да надминува неколку минути за дневни ажурирања, со минимални доцнења.

Причина: Брзото ажурирање осигурува дека корисниците имаат пристап до најновите податоци на време.

2. Сигурност

Барање: Системот треба да обезбеди конзистентност на податоците во сите записи.

Опис: Операциите за прибирање податоци мора да вклучуваат ракување со грешки, повторување на операциите и логирање на неуспесите.

Причина: Конзистентноста е клучна за да се спречи губење или дуплирање на податоците, одржувајќи го интегритетот на базата на податоци.

3. Скалабилност

Барање: Системот треба да се проширува за да поддржува зголемен број на издавачи и историски записи за податоци.

Опис: Базата на податоци и архитектурата за обработка треба да бидат дизајнирани за да поддржуваат зголемена количина на податоци како што се додаваат нови издавачи со тек на време.

Причина: Ова осигурува дека апликацијата може да се прошири по потреба без значително влошување на перформансите.

4. Одржливост

Барање: Кодот на системот треба да биде модуларен и добро документиран.

Опис: Секој чекор за обработка на податоци треба да биде инкапсулиран во посебна функција, а кодната база треба да биде документирана за полесно ажурирање.

Причина: Ова овозможува идните развивачи лесно да го ажурираат или дебагираат кодот.

5. Употребливост

Барање: Системот треба да овозможи пријателски интерфејс или метод за пристап на податоците.

Опис: Јасни инструкции, пораки за грешка и организиран изглед треба да ги водат корисниците при прибирање на податоци.

Причина: Добрата употребливост поддржува лесен пристап за корисниците без обемни технички знаења.

- Кориснички сценарија и персони:

Персона 1: Ана - Финансиски аналитичар

Цел: Да ги добие и анализира најновите податоци за акции на македонските компании.

Сценарио: Ана треба да пристапи до најновите податоци за акции за секој издавач и да ги спореди трендовите со текот на времето. Таа пристапува до системот за да ги гледа дневните податоци за цени, форматирани за лесна анализа во алатки како Excel.

Персона 2: Марко - Податочен научник

Цел: Да ги користи историските податоци за да изгради предвидувачки модели за акции.

Сценарио: Марко го прашува системот за податоците за акции од последните 10 години, што ги користи за да го обучи својот модел за машинско учење. Конзистентниот формат на податоците му овозможува брзо да ги импортира во својот модел.