

Дизајн и архитектура на софтвер

Домашна работа 1

МАКЕДОНСКА БЕРЗА

Илија Бунчески 221094

Филип Иваноски 221038

Александар Јованчов 221070

Содржина

1. Опис на проектот.....	3
2. Функциски барања.....	4
3. Нефункционални барања.....	6
4. Кориснички сценарија.....	7
4.1. Сценарио 1: Прибирање на почетни податоци.....	7
4.2. Сценарио 2: Инкрементално ажурирање на податоци.....	7
5. Персони.....	8
5.1. Персона: Филип, финансиски аналитичар.....	8
5.2. Персона: Александар, студент по економија.....	8
5.3. Персона: Илија, програмер.....	8
6. Use - case дијаграми.....	9

1. Опис на проектот

Овој проект се однесува на изработка на апликација за следење, анализа и тргување со акции на Македонската берза. Основната цел на проектот е да се развие веб апликација која ќе им овозможи на инвеститорите, брокерите и општата јавност лесен пристап до информации за пазарот на Македонската берза.

Оваа апликација е дизајнирана како backend алатка за автоматизирана обработка на точни и ажурирани дневни податоци од изминатите 10 години. Таа користи различни модуларни софтверски архитектури кои овозможуваат преземање и трансформација на податоците од веб-страницата на Македонската берза. Податоците ќе бидат зачувани во чист и структуриран формат, со посебен акцент на ефикасноста и точноста.

Апликацијата ќе поддржува автоматско преземање на листата на издавачи од веб-страницата на Македонската берза, проверка и ажурирање на податоците кои недостасуваат, како и нивна конверзија во формат погоден за складирање за лесна манипулација и понатамошна анализа. Крајниот корисник ќе биде аналитичар или истражувач кој се занимава со финансии и има потреба од точни и ажурирани податоци. Апликацијата ќе обезбеди основа за натамошни анализи како што се трендови, прогнозирање и финансиски извештаи.

2. Функциски барања

- 2.1. Апликацијата треба да биде достапна како веб-базирана платформа, преку веб- прелистувачи, без потреба од дополнителна инсталација на корисничките уреди
- 2.2. Корисниците треба да можат да пристапуваат до апликацијата преку стандардни веб-протоколи од различни уреди, вклучувајќи десктоп, лаптоп и мобилни уреди
- 2.3. Апликацијата треба да опслужува два типа на корисници: администратор и општ корисник на апликацијата
- 2.4. Апликацијата треба автоматски да ги извлекува најновите податоци за сите издавачи котирани на веб-страницата на Македонската берза.
- 2.5. Апликацијата треба да ги извлекува податоците за најмалку последните 10 години на дневно ниво за секој издавач
- 2.6. Апликацијата треба да ги исклучи неважечките издавачи, како што се обврзници или кодови со броеви
- 2.7. Апликацијата треба да ги складира преземените податоци во добро организиран формат
- 2.8. Апликацијата треба да извршува ажурирање кога ќе станат достапни нови податоци
- 2.9. Апликацијата треба да може да ги споредува најновите податоци во својата база на податоци со податоците од веб-страницата на Македонската берза
- 2.10. Апликацијата треба да презема податоци што недостасуваат од веб страната на Македонската берза
- 2.11. Апликацијата треба да прави трансформација на извлечените податоци за датумите во соодветен формат „ден.месец.година“
- 2.12. Апликацијата треба да прави трансформација на извлечените податоци за нумеричките податоци во соодветен формат каде што како разделувач на децималите се користи запирка (,), додека за илјадниците се користи

точка (.)

- 2.13. Апликацијата треба да направи анализа за тоа дали одредена акција треба да се купи
- 2.14. Апликацијата треба да го следи и прикажува времето потребно за целосно ажурирање на базата на податоци со најновите достапни податоци за берзата
- 2.15. Апликацијата треба да овозможи корисникот да пребарува податоци по издавач, временски период или цена
- 2.16. Апликацијата треба да овозможи извезување на обработените податоци во различни формати (CSV , JSON) за надворешна употреба
- 2.17. Апликацијата треба да овозможи генерирање визуелизации за издавачите базирани на историските податоци
- 2.18. Апликацијата треба да извести кога се достапни нови податоци или кога процесот на ажурирање е завршен
- 2.19. Апликацијата треба да ги евидентира времето и датумот на последното направено ажурирање на податоците
- 2.20. Апликацијата треба да може да работи со податоци на македонски јазик

3. Нефункциски барања

- 3.1. Апликацијата треба ефикасно да обработува ажурирања за голем број на податоци без значително влошување на перформансите.
- 3.2. Апликацијата треба да биде лесна за одржување и ажурирање, со јасна структура на код и функции на модуларен филтер за да се поедностават сите потребни промени
- 3.3. Апликацијата треба да нуди кориснички интерфејс кој треба да биде интуитивен и лесен за користење
- 3.4. Апликацијата треба да обезбеди сигурност дека сите податоци се соодветно складирани и заштитени од неовластен пристап
- 3.5. Апликацијата треба да биде достапна 99% од времето на годишно ниво.
- 3.6. Апликацијата треба да биде инсталирана и конфигурирана на два или повеќе сервер компјутери кои ќе работат истовремено и чија цел ќе биде намалување на времето во кое не се нуди услуга
- 3.7. Во случај на неочекуван пад, рестартирањето на еден сервер компјутер треба да трае пократко од 5 минути.
- 3.8. Апликацијата треба да биде веродостојна без да има било каков случај на конфликт во податоци
- 3.9. Апликацијата треба да биде веродостојна без да се појавуваат грешки при извршување на работните задачи за кои е дизајнирана
- 3.10. Апликацијата треба да чува резервна копија од сите податоци за во случај на проблем истите да не бидат загубени
- 3.11. Апликацијата треба да подржува неограничен број на истовремени пребарувања од страна на корисниците
- 3.12. Апликацијата треба да биде лесна за раководење и одржување од страна на корисниците

4. Кориснички сценарија

4.1. Сценарио 1: Прибирање на почетни податоци

- **Корисник:** Аналитичар на податоци.
- **Цел:** Да се пополни базата на податоци со историски податоци за акции од Македонската берза, со цел нивна понатамошна анализа.
- **Чекори:** Аналитичарот ја извршува апликацијата, која автоматски извлекува податоци од веб-страницата на Македонската берза, ги обработува, ги форматира и ги складира во базата на податоци. Откако ќе заврши почетното вчитување, апликацијата обезбедува резиме на потрошеното време и комплетноста на податоците.
- **Исход:** Аналитичарот сега има целосно пополнета и форматирана историска база на податоци подготвена за анализа.

4.2. Сценарио 2: Инкрементално ажурирање на податоци

- **Корисник:** Аналитичар на податоци.
- **Цел:** Да се ажурира базата на податоци само со најновите податоци кои недостасуваат во однос на последното ажурирање.
- **Чекори:** Аналитичарот ја иницира функцијата за ажурирање. Апликацијата го споредува последниот запис во базата на податоци за секој издавач и презема само нови записи од берзата. Новите податоци се форматираат и се додаваат во базата на податоци, одржувајќи конзистентност во форматот.
- **Исход:** Апликацијата само додава нови податоци, намалувајќи го непотребното време за преземање и обработка.

5. Персони

5.1. Персона: Филип, финансиски аналитичар

- **Позадина:** Филип работи во инвестициска фирма и често ги анализира податоците за акциите за да му помогне на неговиот тим при донесувањето одлуки за инвестирање.
- **Потреби:** Извор на податоци кој постојано се ажурира, така што тој не мора секој ден рачно да проверува дали се случило промена.
- **Цели:** Да ја рационализира неговта анализа со користење на податоци што се веќе обработени и форматирани, заштедувајќи време на претходна обработка.

5.2. Персона: Александар, студент по економија

- **Позадина:** Александар е студент по економија кој работи на проект по предметот Бизнис аналитика и за којшто треба да направи компаративна анализа на Македонската и Американската берза.
- **Потреби:** Да собере податоци за неговиот проект, за оној дел поврзан со економските трендови во Македонија
- **Цели:** Александар ја користи апликацијата за да преземе податоци за одреден издавач и временски период. Податоците ги користи за понатамошна анализа.

5.3. Персона: Илија, програмер

- **Позадина:** Илија има задача да обезбеди апликацијата да работи непречено и да ги реши проблемите со обработката на податоците.
- **Потреби:** Јасен, модуларен код што му овозможува да ги идентификува и решава грешките во протокот на податоци или да ја прилагоди апликацијата на промените на страницата.
- **Цели:** Да се воспостави ефикасна и скалабилна архитектура што ги исполнува барањата за перформанси, минимизирајќи го времето на извршување и обезбедувајќи сигурно и стабилно функционирање на апликацијата.

6. Use - case дијаграми

