

# Дизајн и архитектура на софтвер

## Лабораториска вежба – Технички прототип

### Технички прототип: Crypto Exchange Analyzer

#### Технологии:

- Backend: Python, Flask
- Frontend: HTML, CSS, JavaScript
- Податоци: CSV датотеки (локално складирани)

#### 1. Архитектура на апликацијата

Апликацијата користи Pipe and Filter архитектура со следните компоненти:

#### Филтри (Filters):

- SymbolFilter** - Го филтрира списокот на криптовалути
- DateCheckFilter** - Проверува датуми на постоечки податоци
- DataFillFilter** - Пополнува ги недостасувачките податоци

#### Утилити(Utils):

- CSVManager** – Ракување со CSV датотеки
- PerformanceTimer** – Мерење на перформанси

#### 2. Податочно складирање

```
# File Storage Configuration
DATA_DIR = "data"
SYMBOLS_DIR = os.path.join(DATA_DIR, "symbols")
HISTORICAL_DIR = os.path.join(DATA_DIR, "historical")
METRICS_DIR = os.path.join(DATA_DIR, "metrics")
```

#### 3. Backend дел (Flask)

Главна апликација (web\_prototype.py):

```
# Flask Web Application
app = Flask(__name__)
processor = CryptoExchangeProcessor()
```

```
# Global variable to track if data has been loaded
data_loaded = False
```

Модел на подароци (Data Models):

```
# Симболи на криптовалути
{
    'id': 'bitcoin',
    'symbol': 'BTC',
    'name': 'Bitcoin',
    'market_cap_rank': 1,
    'current_price': 91417.00
}
```

```
# Историски податоци
{
    'date': '2024-01-15',
    'open': 90000.00,
    'high': 92000.00,
    'low': 89000.00,
    'close': 91417.00,
    'volume': 28475920184
}
```

```
# Метрики
{
    'date': '2024-01-15',
    'price': 91417.00,
    'volume_24h': 71459168696,
    'market_cap': 1824018810920
}
```

#### 4. Frontend дел (HTML/JavaScript)

```
<!DOCTYPE html>
<html>
<head>
    <title>Crypto Exchange Analyzer</title>
</head>
<body>
    <div class="search-section">
        <input type="text" id="searchInput" placeholder="Внесете криптовалута...">
        <button onclick="searchCrypto()">SEARCH</button>
    </div>
    <div id="resultsContainer"></div>
</body>
</html>
```

### **JavaScript функционалност:**

```
// Пребарување на криптовалути
async function searchCrypto() {
    const searchTerm = document.getElementById('searchInput').value;
    const response = await fetch('/search', {
        method: 'POST',
        body: 'search_term=' + encodeURIComponent(searchTerm)
    });
    const data = await response.json();
    displayResults(data.results);
}

// Приказ на резултати
function displayResults(results) {
    const container = document.getElementById('resultsContainer');
    let html = '<div class="results-grid">';

    results.forEach(crypto => {
        html += `
            <div class="crypto-card" onclick="showCryptoDetails('${crypto.id}')">
                <div class="crypto-symbol">${crypto.symbol}</div>
                <div class="crypto-name">${crypto.name}</div>
                <div class="crypto-price">${crypto.current_price}</div>
            </div>
        `;
    });
    container.innerHTML = html;
}

// Навигација до детали
function showCryptoDetails(cryptoid) {
    window.location.href = '/crypto/' + cryptoid;
}
```

### **5. Детална страница (templates/crypto\_details.html)**

```
<!DOCTYPE html>
<html>
<head>
    <title>Детали за криптовалута</title>
</head>
<body>
    <a href="/" class="back-button">← Назад кон пребарување</a>

    <div class="crypto-header">
```

```
<h1 id="cryptoName">Bitcoin (BTC)</h1>
<div id="cryptoPrice">$91,417.00</div>
</div>

<div class="data-section">
    <h3>Историски податоци за цени</h3>
    <table id="historicalTable">
        <thead>
            <tr>
                <th>Датум</th>
                <th>Отворена</th>
                <th>Висока</th>
                <th>Ниска</th>
                <th>Затворена</th>
            </tr>
        </thead>
        <tbody id="historicalData"></tbody>
    </table>
</div>
</body>
</html>
```

## 6. Стартување на апликацијата

# Инсталација на зависности  
pip install flask pandas requests

# Стартување на серверот  
python web\_prototype.py

# Пристап до апликацијата  
# Отворете: <http://localhost:5000>

## 7. Функционалности

Пребарување и навигација:

- Пребарување на криптовалути по симбол или име
- Приказ на основни информации
- Детални страници со историски податоци
- Навигација помеѓу страници

Податоци:

- 1000+ активни криптовалути
- 10+ години историски податоци
- Дневни метрики од меѓународни разменувачи
- Реално време ажурирање на податоци

## **8. Демонстрација на прототипот**

За демонстрација на техничкиот прототип, следете ги овие чекори:

1. **Стартувајте ја апликацијата** (напишете `python web_prototype.py` во terminal-от и кликнете на линкот кој што ќе ви биде даден: ќе добиете ваква порака (Ready to search! Open <http://localhost:5000> in your browser)
2. **Пребарувајте криптовалута** (на пр. "BTC" или "Bitcoin"...)
3. **Приказ на резултати** во форма на картички
4. **Клик на криптовалута** за навигација до детална страница
5. **Преглед на историски податоци** и метрики

## **9. Потврда на концептот**

Овој технички прототип докажува дека:

- Избраните технологии се погодни за имплементација
- Може да се поврзат сите компоненти на апликацијата
- Може да се пристапи до податоци од базата (CSV датотеки)
- Може да се имплементира навигација помеѓу екрани
- Може да се прикажат податоци за крипто разменувачи