

**НАЦИОНАЛНА ОЛИМПИАДА ПО ИНФОРМАЦИОННИ ТЕХНОЛОГИИ**  
**2024 – 2025 УЧЕБНА ГОДИНА**

Регистрационен № 205



**Factory Management Service**

- **ТЕМА: Служба за управление на предприятия**

Направление: Софтуерни приложения

- **АВТОРИ:**

Back-end developer

*Калоян Калинов Иванов*, ЕГН: 0842180506, адрес: жк. Меден Рудник 81 вх.3 ет.5 ап.9, телефон: 087 688 5563, Имейл: [KKIvanov22@codingburgas.bg](mailto:KKIvanov22@codingburgas.bg), X клас от ПГКПИ – гр. Бургас.

Front-end developer

*Симеон Стилианов Стефанов*, ЕГН: 0850280442, адрес: гр. Свети Влас, ул. „Св. св. Кирил и Методий“, № 46, телефон: 087 702 1717, Имейл: [KKIvanov22@codingburgas.bg](mailto:KKIvanov22@codingburgas.bg), X клас от ПГКПИ – гр. Бургас

- **РЪКОВОДИТЕЛ:**

Красимир Ватев, [kvatev@codingburgas.bg](mailto:kvatev@codingburgas.bg) , учител по теоретично обучение в ПГКПИ – гр. Бургас.

- **РЕЗЮМЕ**

- **Цели**

Целта на проекта ни е да създадем приложение, което осигурява достъпност за всеки при управлението на техните работници, екипи и финанси. С подробни визуализации и интерактивни функции, то предоставя на потребителите ефективно работно изживяване.

- **Основни етапи в реализирането на проекта:**

- **Рефлексия по темата**

- През първите дни на нашия проект екипът се събра и обсъди идеи, базирани на нашата тема.
- **Произход на името**
- Името произлиза от идеята за финансовото приложение, което обсъждахме да създадем. Използваме добре познатото съкращение FMS (Factory Management Service).
- **Дизайн**
- Дизайнът е разработен с принос от целия екип. ChatGPT генерира изображенията, а ние заедно обсъдихме кои да използваме..
- **Развитие**
- Разработката на FMS е тримесечен проект с ежеседмични етапи в GitHub, за да се поддържа постоянен напредък. Калоян Иванов работи по бек-енда, докато Симеон Стефанов се занимава с фронт-енда. Чрез работа в екип, съсредоточена около приложението *FMS*, създадохме надеждно приложение.

- **Ниво на сложност:**

Проектът **FMS (Factory Management System)** представлява цялостно решение за управление на финанси, което включва множество аспекти на софтуерната разработка, анализ на данни и потребителско изживяване.

## 1. Техническа сложност

- **Бек-енд разработка** – Използваме **Python** за обработка на данни, управление на бази данни и автоматизация.
- **Фронт-енд разработка** –Интерфейсът изградихме с **JavaScript, TypeScript, HTML и CSS**, осигурявайки интерактивност и удобство за потребителите.
- **База данни** – Съхраняване и обработка на финансови данни, изискващо ефективни структури за бързо извличане и сигурност.
- **Интеграция с GitHub** – Управление на кода и сътрудничество в екип чрез Git.

## 2. Функционална сложност

- **Проследяване на приходи и разходи** – Автоматизирани процеси за записване и категоризиране на финансови транзакции.
- **Визуализации** – Генериране на диаграми и графики за представяне на финансова информация по достъпен начин.

- **Реално време и автоматизация** – Автоматично актуализиране на данни и известия за финансовото състояние.
- **Сигурност** – Защита на чувствителна информация чрез криптиране и други мерки за сигурност.

### 3. Логистична сложност

- **Работа в екип** – Разпределяне на задачите между разработчиците (бек-енд и фронт-енд).
- **Планиране и изпълнение** – Разделяне на разработката на седмични етапи чрез GitHub, за да се гарантира постоянен напредък.
- **Презентация и документация** – Използване на **Microsoft Word** и **PowerPoint** за представяне на проекта.

- **Структура на проекта**

Проектът е съставен от няколко компонента, които заедно формират една интегрирана и функционална система.

**Бек-енд разработка** – Кодът е написан на **Python**

**Фронт-енд разработка** – Интерфейсът е разписан с **JavaScript, TypeScript, HTML** и **CSS**, осигурявайки интерактивност и удобство за потребителите.

**База данни** – Съхраняване и обработка на финансови данни, изискващо ефективни структури за бързо извличане и сигурност.

- **Реализация, обосновка за използвани технологични средства, алгоритми, литература, програмни приложения и др.**

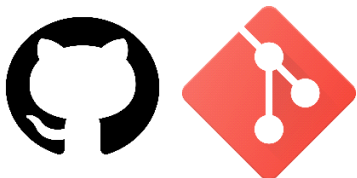
Софтуерът, който използваме за програмиране на нашия проект, е Visual Studio Code.



За създаването на нашия дизайн основната програма е Figma.



Нашите инструменти за съвместна работа са GitHub и Git.



Програмните езици, които използвахме, са Python, JavaScript, TypeScript, HTML и CSS.



Софтуерите, използвани за представянето на проекта, са Microsoft Word и Microsoft PowerPoint.



- **Описание на приложението**

**Инструкция за инсталиране:**

Разархивирайте папката на проекта (205).

**Стартирането на приложението:**

**1. Изтегляне на необходимите файлове**

За да стартирате приложението, първо трябва да изтеглите необходимите файлове.

## 2. Инсталиране на зависимости

### 2.1 Инсталиране на Python

- Изтеглете последната стабилна версия на Python от [официалния HYPERLINK](https://www.python.org/downloads/) ["https://www.python.org/downloads/"](https://www.python.org/downloads/) [HYPERLINK](https://www.python.org/downloads/) ["https://www.python.org/downloads/"](https://www.python.org/downloads/) сайт.
- По време на инсталацията се уверете, че сте отметнали **Add Python to PATH**.

### 2.2 Инсталиране на Node.js и TypeScript

- Изтеглете и инсталирайте [Node.js](#), което включва npm (Node Package Manager).
- След инсталацията отворете **Command Prompt (cmd)** или **PowerShell** и изпълнете командата:

```
npm install -g typescript
```

## 3. Стартиране на приложението

### 3.1 Създаване и активиране на виртуална среда в Python

- Отворете **Command Prompt (cmd)** или **PowerShell** и навигирайте до папката с приложението.
- Създайте виртуална среда с командата:

```
python -m venv venv
```

- Активирайте виртуалната среда:

```
venv\Scripts\activate
```

- Ако получите грешка при разрешенията, изпълнете:

```
Set-ExecutionPolicy -ExecutionPolicy Restricted -Scope CurrentUser
```

### 3.2 Инсталиране на Python зависимости

След активиране на виртуалната среда, изпълнете:

```
pip install -r requirements.txt
```

### 3.3 Инсталиране на Electron

```
npm install electron
```

### 3.4 Стартиране на приложението

```
python app.py
```

### 3.5 Изключване на виртуалната среда след приключване

```
deactivate
```

## 4. Допълнителна информация

Ако срещнете проблеми:

- Уверете се, че всички зависимости са инсталирани правилно.
  - Проверете дали използвате актуални версии на **Python** и **Node.js**.
  - Ако получите грешки при инсталацията на зависимости, опитайте да стартирате терминала с администраторски права.
- 
- **Бъдеща разработка**

Проекта е в стадии на разработка и ще бъдат добавени AI модел за помощ на потребителя да разпределя работниците в компания, ще се добави статус на работниците, спецификация на специалността на работниците и нива на стаж.

### Описание на работните директории:

- **Заклучение**

FMS (Factory Management System) е мощно решение, създадено да опрости финансовото проследяване, бюджетирането и отчетността както за индивидуални потребители, така и за бизнеси. С информация в реално време и автоматизация, то повишава точността и ефективността, като минимизира човешките грешки.

Неговият удобен за потребителя интерфейс гарантира достъпност за всички, а силните мерки за сигурност защитават чувствителните финансови данни. FMS е мащабируем, което го прави подходящо както за лични, така и за корпоративни финансови нужди.

Бъдещите актуализации ще включват AI-базирана аналитика и подобрена автоматизация за по-добър финансов контрол. С FMS потребителите могат да оптимизират ресурсите си, да вземат информирани решения и да постигнат дългосрочна финансова стабилност с увереност и лекота.