**Київський національний університет імені Тараса Шевченка**

**факультет комп’ютерних наук та технологій**

**кафедра теорії та технології програмування**

**КУРСОВА РОБОТА**

**на тему:**

**“Автоматизація бізнес-процесу**

**«Облік кадрів компанії»”**

**студента 5 курсу**

**Шуляка Нікіти Віталійовича**

**Науковий керівник:**

**доцент, кандидат технічних наук**

**Ткаченко Олексій Миколайович**

**Київ – 2022**

ЗМІСТ

[ВСТУП 3](#_Toc104112135)

[ТЕОРЕТИЧНА ЧАСТИНА 4](#_Toc104112136)

[**1.1.** **Що таке RPA?** 4](#_Toc104112137)

[**1.2.** **Сфера використання технології RPA** 5](#_Toc104112138)

[**1.3.** **Приклади готових рішень** 6](#_Toc104112139)

[ПРАКТИЧНА ЧАСТИНА 8](#_Toc104112140)

[**2.1.** **Обґрунтування вибору програмного середовища розробки** 8](#_Toc104112141)

[**2.2.** **Розробка програмного продукту** 9](#_Toc104112142)

[**2.2.1.** **Постановка задачі** 9](#_Toc104112143)

[**2.2.2.** **Аналіз задач проекту. Вимоги до системи** 10](#_Toc104112144)

[**2.2.3.** **Алгоритм роботи бізнес-процесу** 11](#_Toc104112145)

[ПОСІБНИКИ ПРОГРАМІСТА ТА КОРИСТУВАЧА 18](#_Toc104112146)

[**3.1.** **Посібник програміста** 18](#_Toc104112147)

[**3.2.** **Посібник користувача** 19](#_Toc104112148)

[ВИСНОВКИ 20](#_Toc104112149)

[СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 21](#_Toc104112150)

**ВСТУП**

Актуальність теми курсової роботи. На сьогоднішній день основною і невід’ємною частиною діяльності будь-якого підприємства є застосування інформаційних технологій та інструментів для оцінки і визначення структури бізнес-процесів. Схожі розробки на даний момент є важливим етапом для більшості українських підприємств, тому дана тема не втратить своєї актуальності в найближчі декілька десятиліть.

**Мета** курсової роботи полягає у практичному ознайомлені з досить новою технологією RPA та безпосередньої автоматизації бізнес-процесу “Облік кадрів”.

Для досягнення бажаної цілі вирішуються наступні задачі:

* аналіз предметної області;
* огляд існуючих програмних продуктів;
* об’єктно-орієнтоване моделювання предметної області;
* вибір середовища розробки;
* програмна реалізація.

В теоретичній частині курсової роботи буде наведена більш детальна інформація стосовно понять та визначень у технологій RPA. Будуть наведені новітні готові програмні продукти, які широко застосовуються у сучасних підприємствах України та усього світу для автоматизації бізнес-процесів.

У практичній частині здійснено аналіз вимог, вибір моделі життєвого циклу, середовища програмування, а також наведені посібники програміста, який описує основні характеристики проекту автоматизації, та посібник користувача, що характеризує роботу з готовим програмним продуктом.

**ТЕОРЕТИЧНА ЧАСТИНА**

* 1. **Що таке RPA?**

У даній частині будуть коротко наведені переваги роботизації робочих процесів та критерії застосовності роботів у роботі компаній.

Автоматизація рутинних операцій знижує собівартість корпоративних бізнес-процесів, виключає помилки при обробці інформації, скорочує цикл її обробки, а також гарантує дотримання корпоративних та галузевих стандартів.

Незважаючи на досить високий рівень автоматизації, обʼєм ручних операцій користувачів як і раніше залишається на досить високому рівні.

Характерними прикладами таких операцій є:

* повторюваний ввід одних і тих самих даних у декілька автоматизованих інформаційних систем,
* ручний переніс результуючих даних із однієї системи в іншу у якості вихідних для наступної обробки
* пошук даних у декількох системах із цілю їх збору, консолідації та підготовки корпоративної звітності.

Крім того, справа не обмежується корпоративними системами. Користувачі зачасту повинні здійснювати пошук інформації на сторонніх сайтах, наприклад, з метою отримання оголошень про заплановані тендерні процедури, аналізу цін конкурентів, оновлення статусу виконання замовлень партнерами, пошуку кандидатів на роботу та в інших аналогічних випадках.

Найсумніше є те, що такою роботою зайняті найчастіше дуже кваліфіковані працівники. І це зовсім не разові завдання, а цілком регулярні, що є частиною робочого процесу. Причому, у ряді галузей подібні рутинні завдання можуть займати до 50% робочого часу співробітника.

Для діяльності багатьох організацій характерні наступні особливості:

* великий обсяг повторюваної ручної роботи;
* великі обʼєми даних;
* розʼєднанні інформаційні системи;
* використання висококваліфікованих співробітників на задачах, які не потребують професійних знань
* наявність «пікових» періодів у обробці інформації.

Robot Process Automation (RPA) – це спеціалізовані програмні засоби, які імітують роботу користувача, взаємодіючи із різними автоматизованими інформаційними системами через стандартний інтерфейс користувача. Завдяки цьому відпадає необхідність у трудомісткій та витратній програмній інтеграції і забезпечується спільна праця декількох корпоративних систем. В тому числі і тих, до яких відсутній будь-який інший доступ, окрім як через інтерфейс користувача, та/або принципово неможливе використання програмної інтеграції за допомогою API (Application Programming Interface).

Найкращими кандидатами на автоматизацію є ті робочі процеси, для котрих притаманні наступні риси:

* використання структурованих початкових даних;
* наявність формальних правил та чіткої послідовності виконання операцій;
* повторюваність та висока частотність;
* відсутність необхідності приймання рішень людиною на основі інформації, яка лежить поза контекстом виконуваного процесу;
* високі вимоги до якості результату;
* відстеження виконаних операцій та виявлення звʼязку результату із вхідними даними.

Прикладами задач, які успішно виконуються роботами, є:

* внесення інформації із типових документів будь-яких форматів до облікових систем;
* перенесення інформації із однієї системи у іншу;
* пошук інформації на зовнішніх сайтах / внутрішніх ресурсах із цілю збору та підготовки інформації для подальшого аналізу, та багато інших.
  1. **Сфера використання технології RPA**

Технології RPA не надають перевагу певній галузі застосування. Вони із самого початку розроблялись для крос-галузевого застосування, орієнтовані на вдосконалення процесів організації.

Приклади успішних проектів по впровадженню програмних роботів існують майже у будь-якій сфері діяльності, однак, найбільше поширення вони отримують там, де традиційно високий рівень ручної праці при обробці важливої інформації. Це, наприклад, так сфери діяльності, як:

* банківська сфера;
* страхування;
* логістика;
* органи державного управління;
* проведення аудиту і деякі інші.

За допомогою технології RPA можна ефективно вирішувати більшість бізнес-задач у наступних галузях:

* будь-який вид звітності (бухгалтерський, кадровий, фінансовий, фінансово-господарський);
* автоматизація обробки актів звірки інформації;
* пошук кандидатів на роботу;
* клієнтська робота;
* торгівля;
* аналіз інформації про потенційного клієнта у відкритих джерелах;
* збір даних із різних систем для подальшого їх аналізу та обробки, та інші.

Бізнес-ефекти використання даної технології є досить наглядними. Для RPA взагалі характерна прозорість і простота розрахунку економічного ефекту. На основі всесвітнього досвіду можна відокремити наступні економічні на бізнес-ефекти впровадження технології:

* скорочення собівартості робочих процесів (25-50%);
* умовне звільнення співробітників у результаті впровадження одного робота;
* приріст продуктивності співробітників (25-50%);
* виключення помилок та впливу людського фактору.

Однак, головним ефектом від впровадження технології є приріст задоволеності співробітників своєю роботою, оскільки зникає необхідність у виконанні рутинних операцій, а отже зʼявляється час для виконання задач, які потребують професійних навичок.

Дана технологія є багатофункціональною.

Роботи можуть імітувати дії людини, використовуючи стандартні засоби роботи користувачів: зчитувати інформацію з екранів, виконувати ввід даних, керувати роботою програм, виконувати окремі транзакції з використанням корпоративних програм. Організовувати спільну роботу декількох корпоративних та/або зовнішніх систем, як повʼязані логікою одного процесу, але, в силу своєї архітектури, не мають технологічних програмних інтерфейсів. Автоматизовувати обробку запитів, пошук та збір даних, надсилати відповіді та підтвердження, взаємодіяти із зовнішніми системами, ініціювати запити до контрагентів. Розпізнавати дані, які надаються через інтерфейс користувача корпоративної програми та передавати ці дані на обробку.

* 1. **Приклади готових рішень**

У наш час існує багато готових рішень, які знайшли своє використання по усьому світу. Одними із таких є всесвітньо відомі рішення:

* Blue Prism;
* UiPath;
* Automation Anywhere.

**Blue Prism** побудовано на базі Microsoft .NET Framework. Він автоматизує будь-яку програму та підтримує будь-яку платформу (мейнфрейм, Windows, WPF, Java, веб тощо), представлену різними способами (емулятор терміналу, thick-клієнт, thin-клієнт, веб-браузер, Citrix та веб-сервіси). Він був розроблений для моделі розгортання в кількох середовищах (розробка, тестування, стейдж та продакшн) з фізичним і логічним контролем доступу. Програмне забезпечення Blue Prism RPA включає централізований інтерфейс керування і модель розподілу змін процесу, що забезпечує високий рівень видимості та контролю. Додатковий контроль надається бізнесу через централізовану модель для розробки та повторного використання процесів. Blue Prism записує кожний вхід в систему, зміну дії керування, а також рішення та дії, що вживаються роботами для визначення статистики та оперативної аналітики в реальному часі. Програмне забезпечення підтримує нормативні контексти, такі як PCI-DSS, HIPAA і SOX, з великою кількістю засобів контролю, які забезпечують необхідну безпеку та керування. Усе кодування процесу автоматизовано всередині, що дозволяє навіть нетехнічним користувачам автоматизувати процес, перетягуючи компоненти в інтерфейс.

**UiPath** розробляє програмне забезпечення для автоматизації повторюваних цифрових завдань, які зазвичай виконують люди. Технологія поєднує в собі імітацію того, як люди читають екрани комп’ютера (комп’ютерний зір AI), разом з API, і надає користувачам доступ до попередньо створених компонентів автоматизації, які можна комбінувати для автоматизації рутинних процесів. Його попередні продукти спрощували завдання, виконувані за допомогою іншого програмного забезпечення для бізнесу, такого як системи CRM або ERP, у внутрішніх і бек-офісних сферах, таких як бухгалтерський облік, документообіг кадрових ресурсів та обробка запитів. Нові програми програмного забезпечення компанії включають координацію з системами штучного інтелекту для спрощення повторюваних завдань фронт-офіс, включаючи керування клієнтами. Вбудовані в UiPath механізми OCR від Google, Microsoft і ABBYY можуть зчитувати інформацію з екрана або відсканованого документа.

Основним продуктом UiPath є платформа автоматизації UiPath. Платформа поєднує в собі сімейство продуктів візуального інтегрованого середовища розробки з низьким кодом (IDE) під назвою Studio для створення процесів, з клієнтськими агентами, які називаються Роботами, які виконують ці процеси. Процеси розгортаються, контролюються й керуються віддалено за допомогою центрального інструменту керування, який називається Orchestrator.

**Automation Anywhere (AA)** спеціалізується в області роботизації бізнес-процесів і пропонує платформу, на основі якої велика кількість організацій створює інтелектуальних цифрових співробітників всесвітнього рівня.

AA має на меті масштабованість і розвʼязок складних проблем для великих компаній. Хмарна веб-архітектура і єдина платформа допомагають збільшити швидкість налаштування.

Розвинені когнітивні функції та вбудований штучний інтелект краще сприймає неструктуровані дані.

**ПРАКТИЧНА ЧАСТИНА**

У даній частині наведено опис та програмну реалізацію бізнес-процесу «Обліку кадрів». Також будуть описані подальші кроки для покращення використання даного процесу.

* 1. **Обґрунтування вибору програмного середовища розробки**

Програмна реалізація була написана мовою C# (.Net Framework 4.7.2) у середовищі розробки Visual Studio 2022. Microsoft Visual Studio – серія продуктів фірми Майкрософт, які містять інтегроване середовище розробки програмного забезпечення та низку інших інструментальних засобів. Ці продукти дають змогу розробляти як консольні програми, так і програми з графічним інтерфейсом, включно з підтримкою технології Windows Forms, а також вебсайти, везастосунки, вебслужби як у рідному, так і в керованому кодах для всіх платформ, що підтримуються Microsoft Windows, Windows Mobile, Windows Phone, Windows CE , .NET Framework, .NET Compact Framework та Microsoft Solverlight.

Вибір цієї мови програмування базувався на тому, що це є безкоштовна платформа для розробки, із зручним інтерфейсом користувача та попередньо вже існував практичний досвід написання програм на даній платформі.

Усі необхідні пакети та бібліотеки встановлюються через NuGet Packages. Пакет NuGet є окремим ZIP-файлом з розширенням .nupkg, що містить скомпільований код та інші файли, пов'язані з цим кодом. А також описовий маніфест, що включає такі відомості, як номер версії пакету. Розробники, які мають код, до якого потрібно надати спільний доступ, створюють пакети та публікують їх на закритих або відкритих вузлах. Споживачі отримують ці пакети з відповідних вузлів, додають їх до своїх проектів, а потім викликають функції пакета в коді свого проекту. При цьому NuGet сам обробляє всі проміжні дані.

* 1. **Розробка програмного продукту**

Розробка кожного програмного продукту включає в себе наступні задачі:

* аналіз задач проекту;
* створення архітектури ПЗ: визначення того, які процеси будуть реалізовані; які алгоритми будуть використовуватись; які технології будуть найкращим чином відповідати поставленим задачам;
* розробка алгоритмів та тестування розробником.
  + 1. **Постановка задачі**

Існує таблиця TimeTracking компанії, у якій описана інформація про кожного робітника (його ПІП, позиція, спеціалізація, команда) та витрати часу на різні проекти. Протягом місяця кожен робітник указує скільки годин він витратив на виконання проектів. Також, у випадку взяття робітником лікарняного, відпустки чи вихідного дня, йому необхідно вказати це у таблицю. Якщо робітник не виконував ніяких задач по проекту, існує поле із записом “Company Name”, де працівник вказує скільки часу він сидів без задач. Робочий день складає 8 годин.

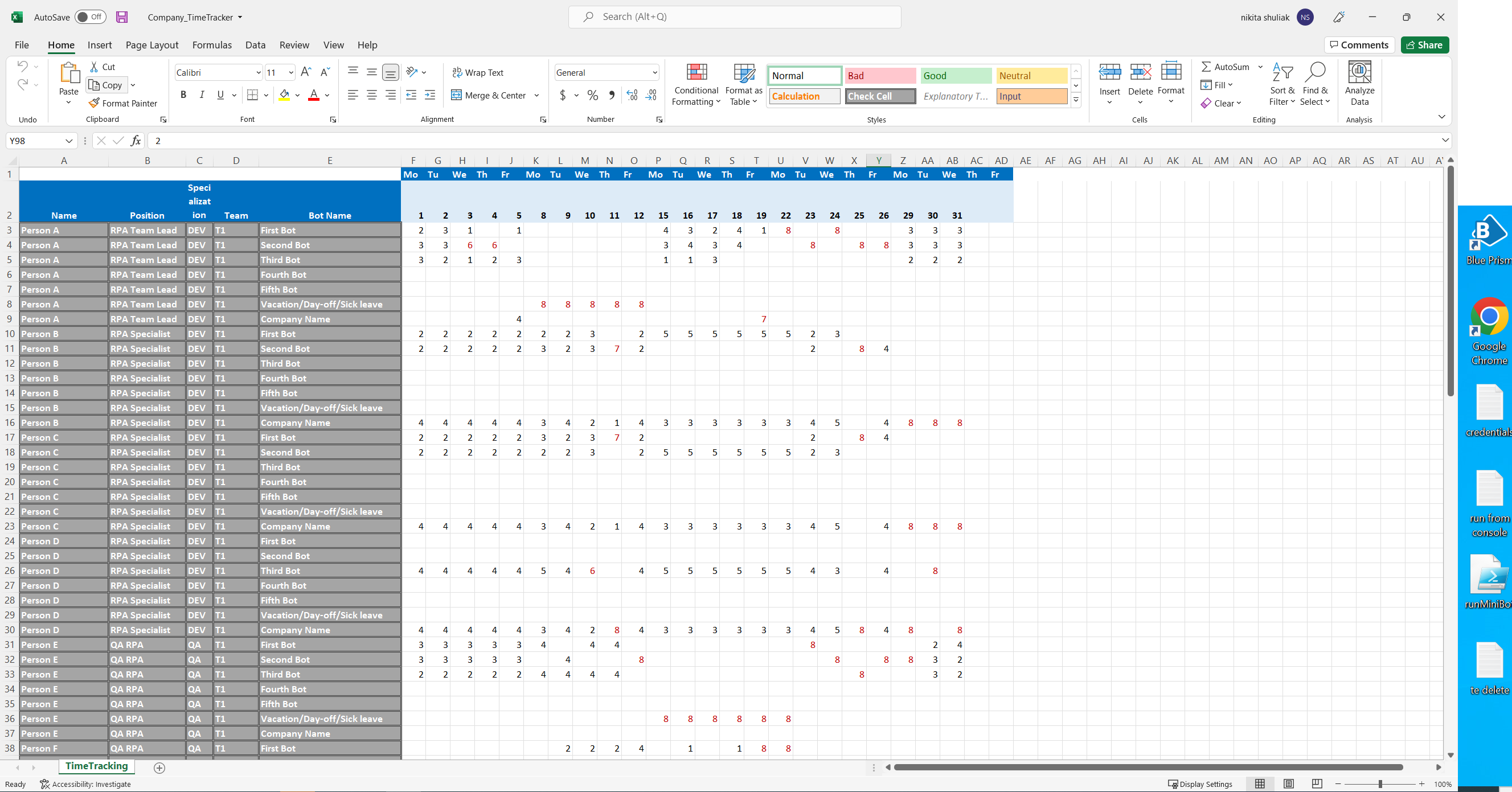


Рис. 2.1. Приклад документу TimeTracking

Після закінчення місяця дана таблиця надсилається на поштову скриньку бота із певною темою (тему листа задає користувач). Коли бот отримав лист, користувач ініціює процес обробки та аналізу ботом вхідного файлу.

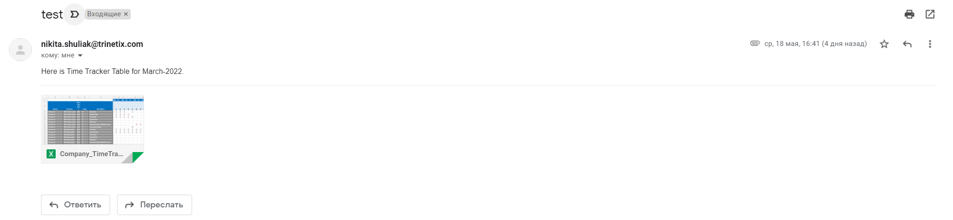


Рис. 2.2. Приклад формату вхідного листа

Після закінчення процесу, бот надсилає лист користувачу (пошту, на яку бот відправлятиме результат, вказує користувач) із інформацією про 3-ох найбільш та найменш продуктивних робітників компанії, та детальну інформацію про кожного з них.

У випадку будь-яких помилок при виконанні процесу, бот надсилає лист користувачу із деталями помилок і можливими причинами їх виникнення.

* + 1. **Аналіз задач проекту. Вимоги до системи**

Система складається із двох частин – програмний робот та система Microsoft Outlook.

Інтерфейс користувача у даній версії продукту відсутній. Запуск програмного бота здійснюється за допомогою ініціації .exe файлу програмного рішення.

Існують наступні вимоги до системи:

* У системі налаштовані і встановлені системи Microsoft Outlook, Microsoft Excel;
* Надані усі необхідні доступи до вищеназваних систем.

Бізнес-процес «Обліку кадрів» не потребує використання бази даних, оскільки інформація про робітників компанії є конфіденційною і у рамках безпеки дана інформація не зберігається у системі.

Для розробки системи було обрано водоспадну чи каскадну модель життєвого циклу через її чітку дисципліну. Вибір аргументований тим, що на кожному кроці розробки існує чіткий та описаний шлях дій (документація), а також тим, що дана модель виключає ризики недооцінки часу виконання проекту. Головною перевагою даної моделі є також її простота у визначені затрат. Основним недоліком моделі є те, що перехід від однієї фази до іншої потребує повну коректність результату попередньої фази.

Дана модель не підходить для реалізації крупних проектів, але досить гарно себе демонструє при створені невеликих систем, таких як роботизована автоматизація системи. Усі вимоги до бізнес-процесу повинні бути попередньо створені, аргументовані, зафіксовані та не підлягають майбутнім змінам. Якщо у майбутньому необхідно змінити деякі кроки процесу – це потребуватиме аналізу ризиків та, можливо, створення нового допоміжного бізнес- процесу.

* + 1. **Алгоритм роботи бізнес-процесу**

Структура рішення складається із окремих класів, які відповідають за той чи інший логічний процес.

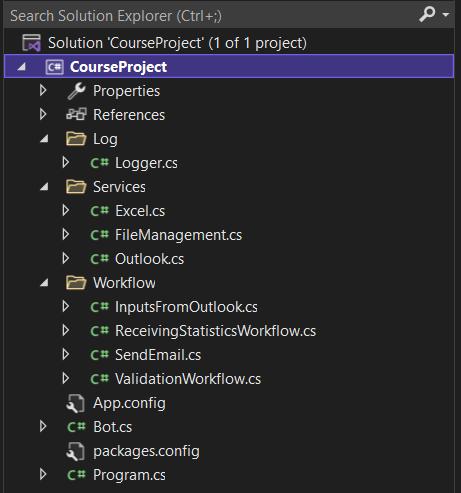


Рис. 2.3. Структура рішення бізнес-процесу

Алгоритм роботи бізнес-процесу виглядає наступним чином:

Бот розпочинає свою роботу після ініційованого запуску зі сторони клієнта .exe файлу програми. Попередньо у файлі input.json клієнт задає інформацію: логування кроків бота; шлях до збереження логів; імʼя боту; булевський параметр, який відповідає за необхідність логування кроків бота; тема вхідного листа; пошти, на яку бот надсилатиме лист із результатом; булевський параметр, чи необхідно надсилати лист.

****

Рис. 2.4. Приклад input.json файлу

Далі йде виконання головної частини програми.

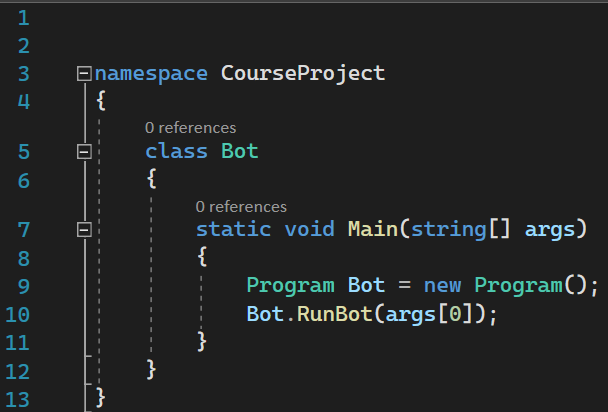


Рис. 2.5. Функція main() тіла програми

Функція main() приймає масив args[] як аргумент, у якому прописано шлях до файлу input.json. Потім у тілі функції створюється екземпляр класу Program і викликається метод RunBot(). Власе у методі RunBot() і описана вся виконувана логіка.

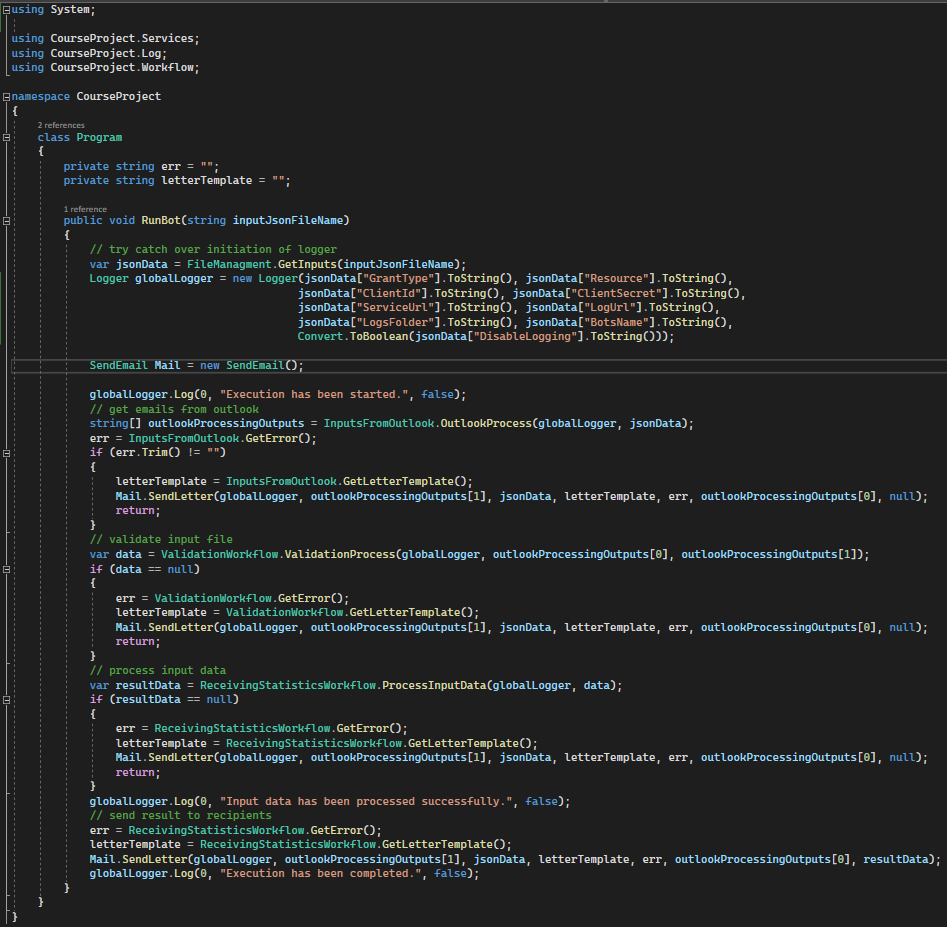


Рис. 2.6. Клас Program та метод RunBot()

Спочатку ми зчитуємо дані із файлу input.json у змінну jsonData. Наступним кроком ми створюємо обʼєкт Logger, який дозволяє нам зберігати та відслідковувати хід програми на окремих фазах процесу. Далі створюється обʼєкт класу SendMail, який відповідає за надсилання листів.

Наступна фаза – зчитування листів із поштової скриньки бота.

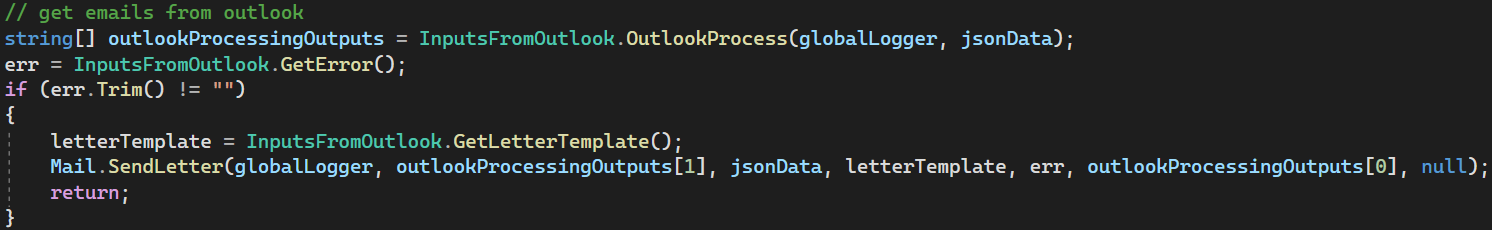


Рис. 2.7. Фаза зчитування листів із поштової скриньки бота

Метод OutlookProcess() перевіряє чи є лист на пошті бота. Якщо листів немає, бот сповістить про помилку. Також бот валідує лист, а саме:

* здійснюється перевірка чи існує доступ до ресурсу Outlook;
* здійснюється перевірка чи має лист в собі необхідний файл потрібного формату (формат файлу .xlsx);
* якщо надіслано більше 1-го файлу, бот сповістить про помилку.

Після вдалої валідації, бот скачує файл та зберігає у памʼяті шлях до файлу та імʼя файлу. Якщо виникне помилка при завантажені файлу, бот зупинить виконання процесу і сповістить про помилку завантаження.

Наступний етап валідує завантажений файл та зчитує дані із нього.

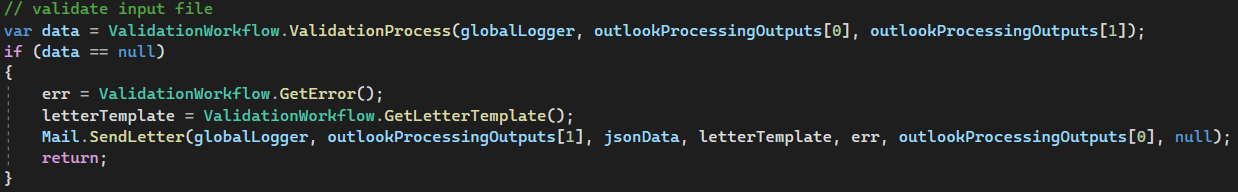


Рис. 2.8. Фаза валідації завантаженого файлу та зчитування даних із нього

При валідації бот перевіряє наступне:

* перевірка на пошкодження файлу;
* перевірка на наявність даних;
* перевірка на доступ до зчитування даних;
* перевірка на наявність паролю у файлі.

Якщо будь-яка із цих помилок виникає, бот зупиняє процес і звітує про наявність помилки.

Одним із ключових методів даної фази є зчитування даних із файлу. Це є досить громісткий процес, який потребує великої кількості операцій та перевірок. Як можна бачити з Рис. 2.9, результатом цієї обробки є досить складна структура даних, яка містить в собі кортежі із списків, словників, стрічок та інших кортежів. Це все зроблено заради зручності майбутньої обробки цієї інформації.

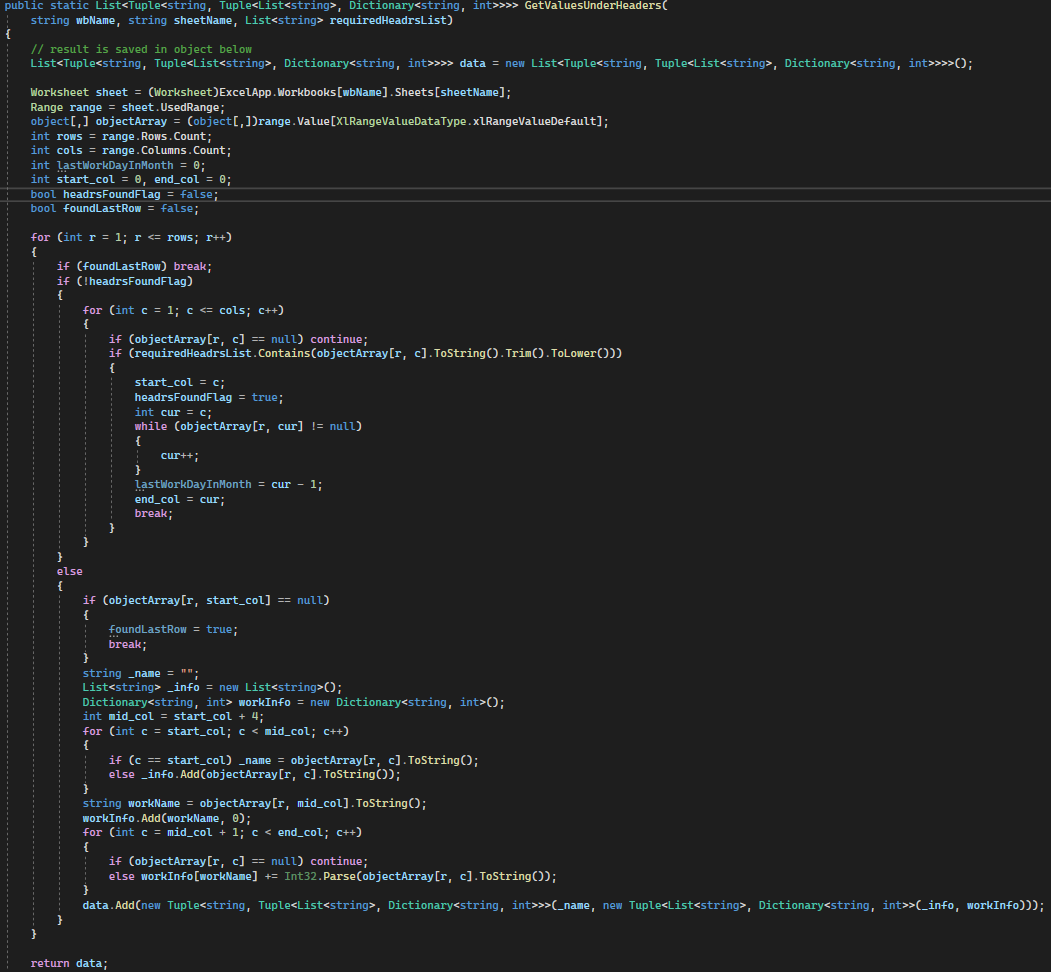


Рис. 2.9. Зчитування даних із файлу

Наступна фаза – отримання результату.

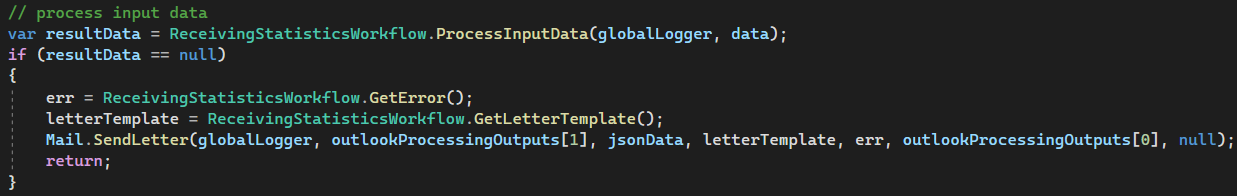


Рис. 2.10. Фаза отримання результату

Даний етап є також складним у тому розумінні, що результат має ще складнішу структуру, ніж на попередньому етапі. Метод GetStatisticksFromInputData() повертає список кортежів, який містить в собі усю проаналізовану інформацію про кожного із робітників.

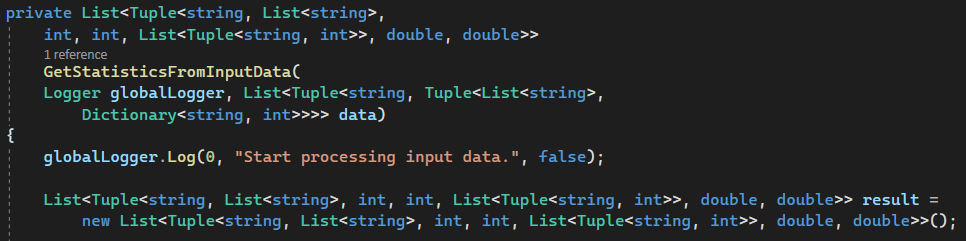


Рис. 2.11. Структура результату та метод GetStatisticksFromInputData()

Метод GetStatisticksFromInputData() повертає список кортежів, який містить в собі усю проаналізовану інформацію про кожного із робітників.

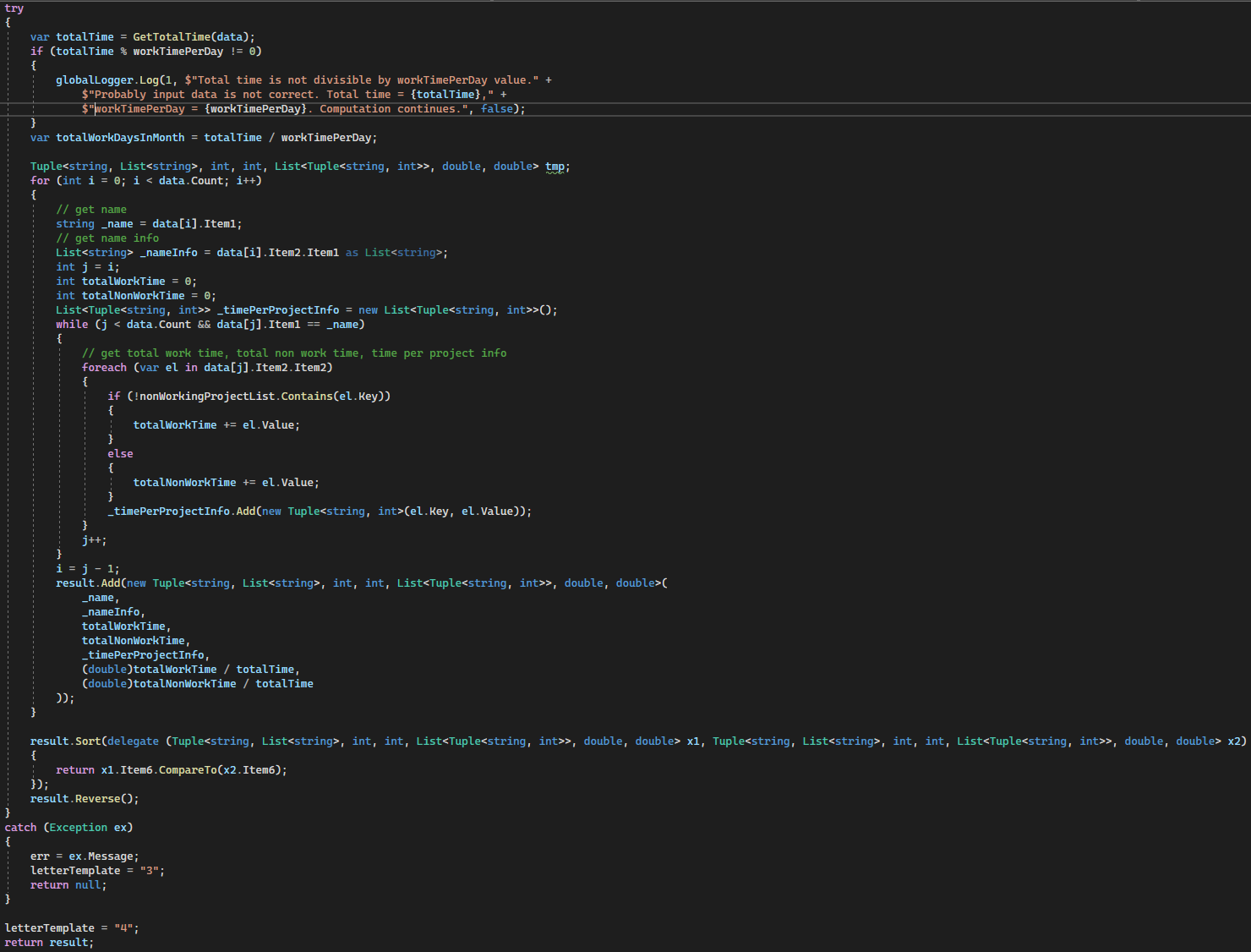


Рис. 2.12. Алгоритм обробки та отримання результату

Завершення програми полягає у надсиланні листа на пошту користувача. Цю логіку реалізовано у окремому класі SendEmail.cs.

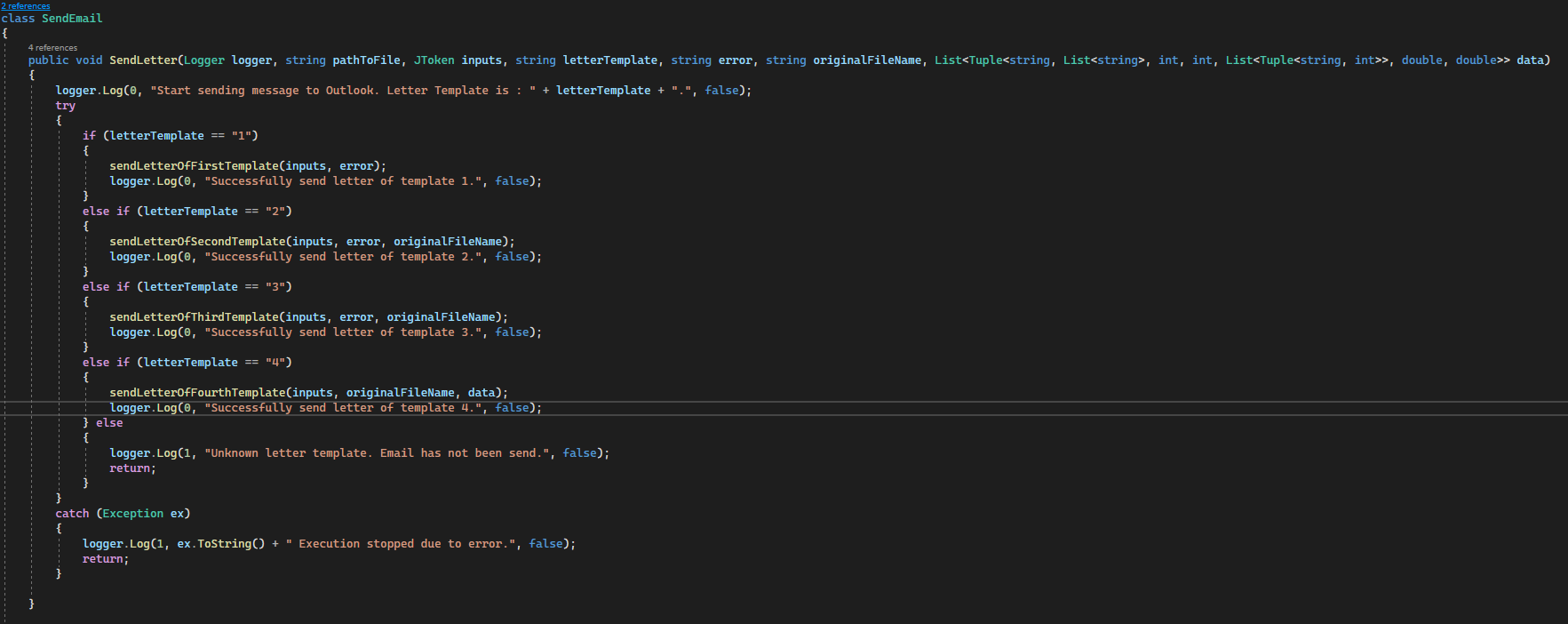


Рис. 2.13. Реалізація класу SendEmail.cs та методу SendLetter()

В залежності від шаблону, який отримує метод SendLetter(), ініціюється відправка відповідного листа. Наведемо шаблони усіх листів процесу.

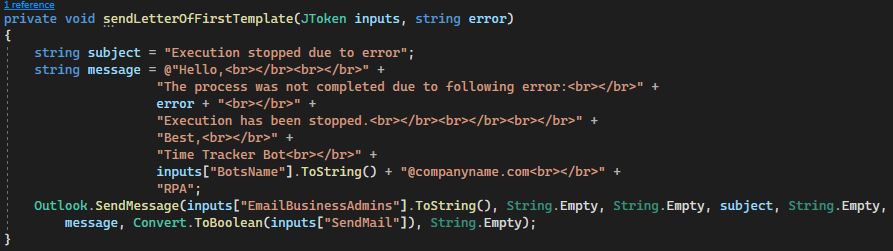


Рис. 2.14. Перший шаблон листа

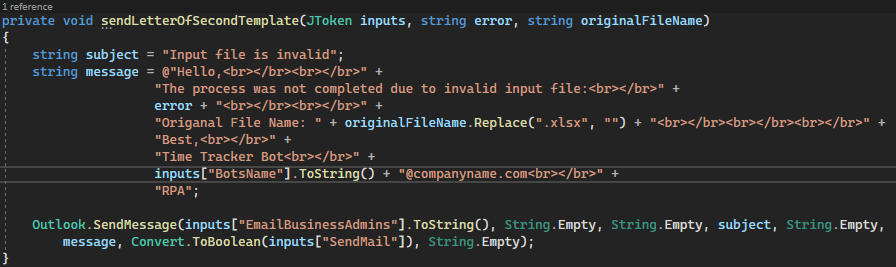


Рис. 2.15. Другий шаблон листа

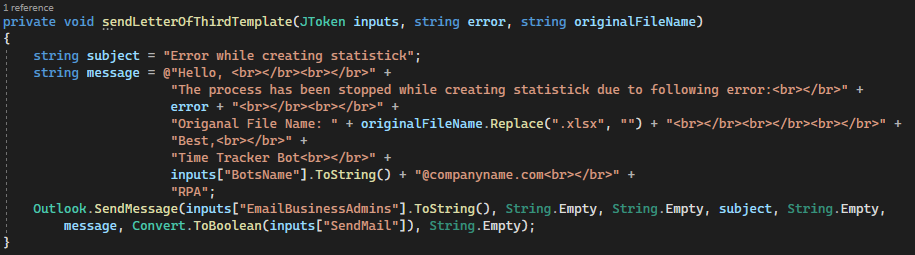


Рис. 2.16. Третій шаблон листа

Шаблон листа №4 є найскладнішим по своїй структурі. Цей лист надсилається лише у випадку успішного виконання процесу і отримання усіх необхідних даних. У листі наведено детальну інформацію (імʼя, позиція, спеціалізація, команда) про кожного із 6 робітників (3 найбільш продуктивних та 3 найменш продуктивних), їх ККД та кількість годин, яку вони витратили на кожен із проектів протягом місяця та загальну кількість годин.

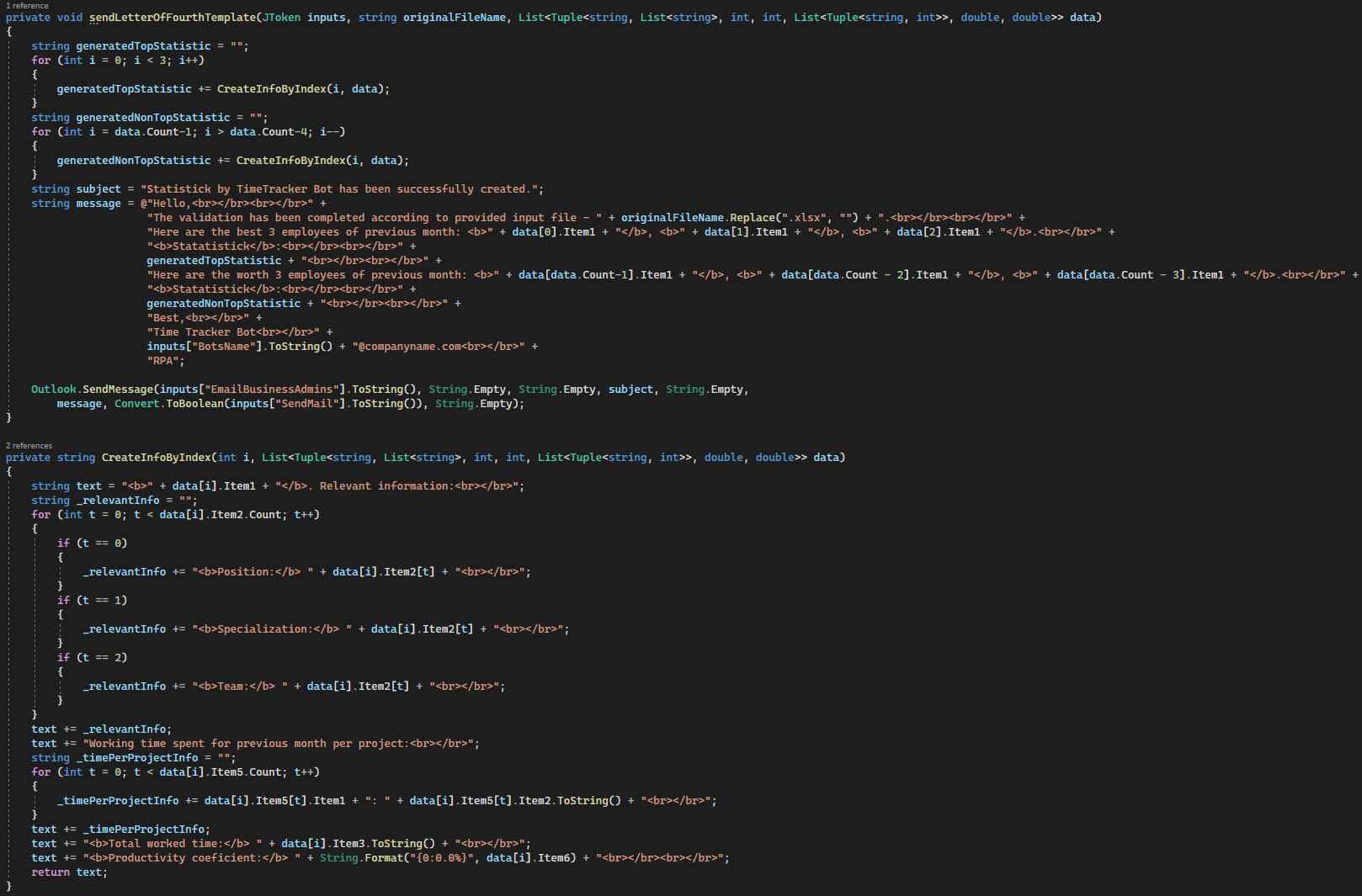


Рис. 2.17. Четвертий шаблон листа

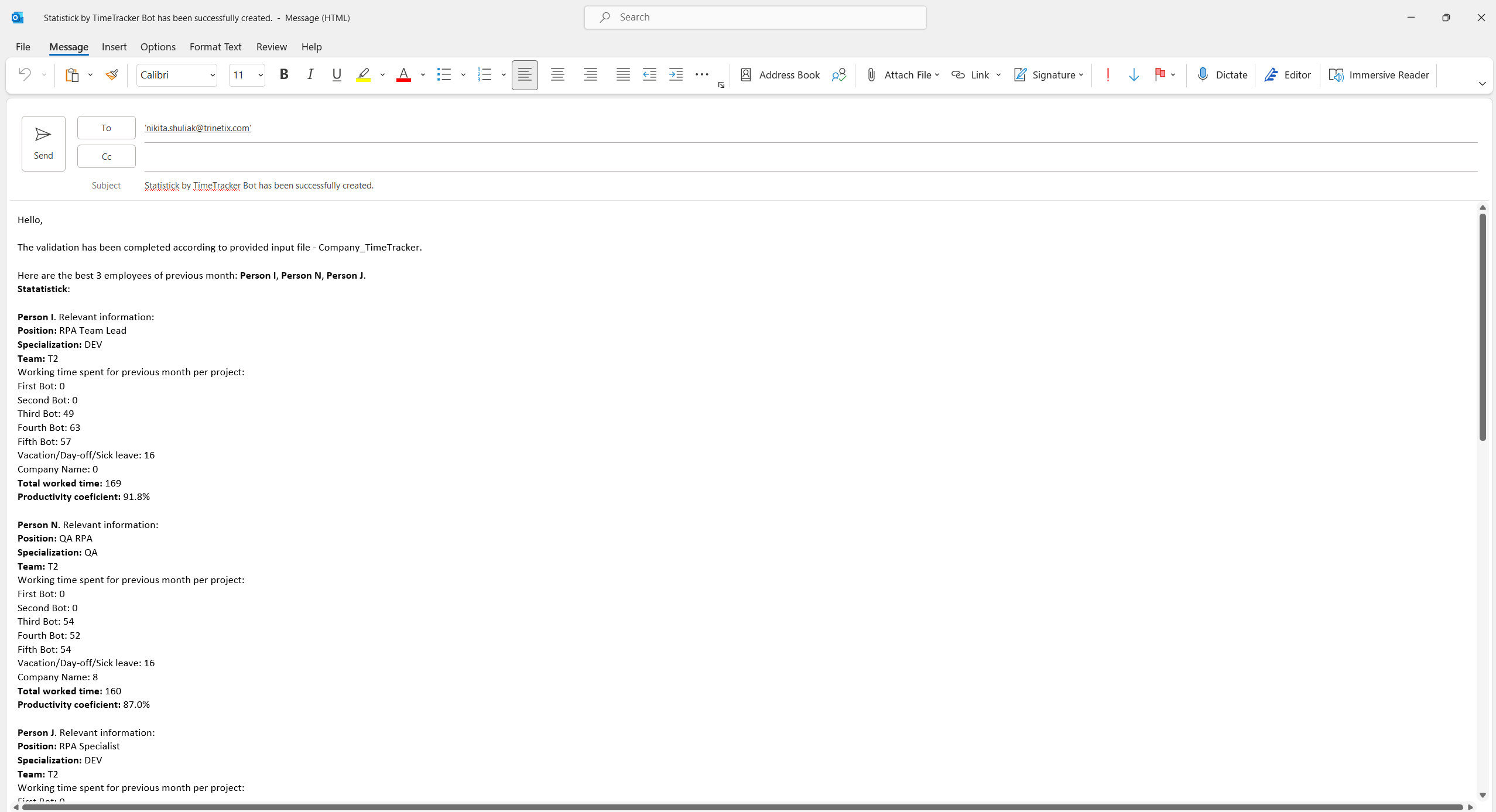


Рис. 2.18. Приклад успішного виконання та отримання листа

**ПОСІБНИКИ ПРОГРАМІСТА ТА КОРИСТУВАЧА**

* 1. **Посібник програміста**

Призначення та умови використання програми:

Програмний продукт призначений для ознайомлення фахівців компанії, але так само придатний для ознайомлення і навчання нових робітників.

Область застосування програмного продукту є звітність по обліку кадрів. Дане ПЗ є корисним для фахівців даної області із навичками програмування на мові C# і розумінням платформи .NET Framework.

Технічні вимоги, необхідні для роботи рішення: ПЗ повинно функціонувати, як мінімум, на програмному забезпеченні, що задовольняє наступним характеристикам:

* Windows Server 2019 (та новіші);
* Windows 7 (x64 – x86);
* Windows 8 (x64 – x86);
* Windows 10 (x64 – x86).

Архітектури, що підтримуються:

* 32-бітні (x86);
* 64-бітні (x64).

Кількість вільного простору на диску – 30 Мб.

Запуск програми ініціюється через .exe файл програми.

Вхідні дані – шлях до файлу налаштування (input.json).

Вихідні файли відсутні (надсилається лист на пошту користувача).

Майбутні покращення – ідеальним варіантом покращення цього рішення було б створення зручного UI web-інтерфейсу для користувача, із можливістю заповнення даних файлу input.json на веб-сторінці і ініціації початку обробки через натискання клавіші “start”.

* 1. **Посібник користувача**

Для встановлення програмного забезпечення, компʼютер користувача повинен мати наступні характеристики:

* Windows Server 2019 (та новіші);
* Windows 7 (x64 – x86);
* Windows 8 (x64 – x86);
* Windows 10 (x64 – x86).

Архітектури, що підтримуються:

* 32-бітні (x86);
* 64-бітні (x64).

Кількість вільного простору на диску – 30 Мб.

Процес запуску досить простий – користувач заповнює дані у input.json файлі, да запускає програму через .exe файл.

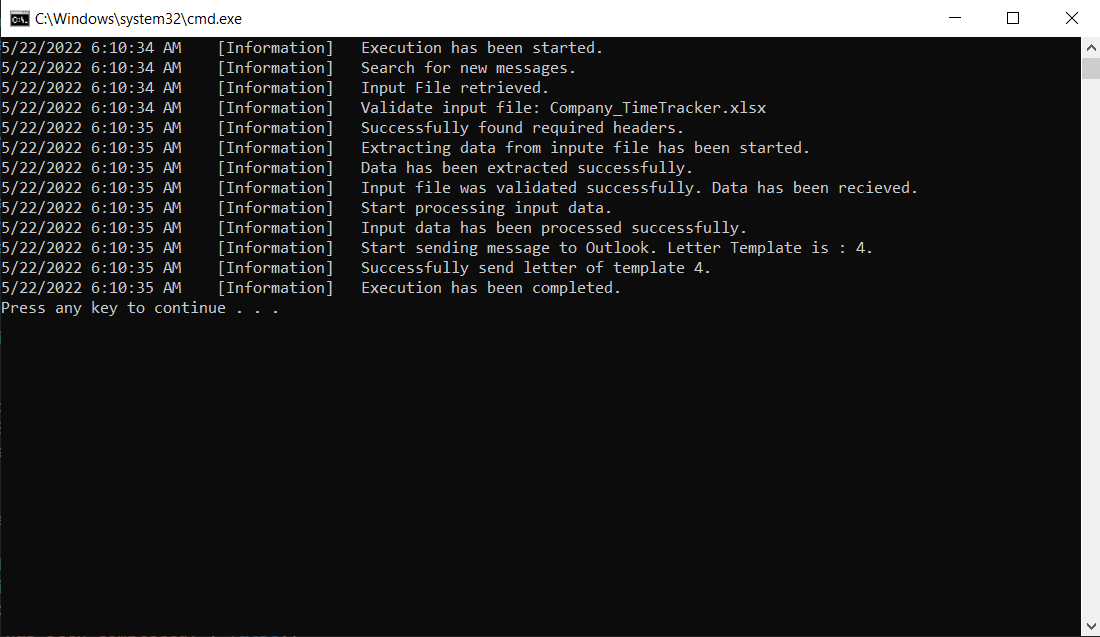


Рис. 3.1. Приклад виконання програми

**ВИСНОВКИ**

По результатам роботи можна зробити наступні висновки:

На сьогоднішній день основною і невід’ємною частиною діяльності будь-якого підприємства є застосування інформаційних технологій та інструментів для оцінки і визначення структури бізнес-процесів. Були наведені підходи до автоматизації бізнес-процесів, існуючі інструменти та методи, а також область використання програмних рішень та складність їх впровадження.

Мета курсової роботи полягала у практичному ознайомлені з досить новою технологією RPA та безпосередньої автоматизації бізнес-процесу “Обліку кадрів”.

В результаті виконаної роботи можна зробити наступні висновки о реалізації цілі проекту:

* Чітко визначена структура і підхід до реалізації автоматизації бізнес-процесу
* Обрана каскадна модель життєвого циклу
* Наведено алгоритм реалізації програми із прикладами його виконання

**СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ**

1. <https://en.wikipedia.org/wiki/Blue_Prism#Blue_Prism_technology>
2. <https://en.wikipedia.org/wiki/UiPath>
3. <https://www.tadviser.ru/index.php/%D0%9A%D0%BE%D0%BC%D0%BF%D0%B0%D0%BD%D0%B8%D1%8F:Automation_Anywhere>
4. <https://yur-gazeta.com/publications/practice/informaciyne-pravo-telekomunikaciyi/robotic-process-automation-yuristam-nichogo-boyatisya.html>
5. <https://www.tadviser.ru/index.php/%D0%A1%D1%82%D0%B0%D1%82%D1%8C%D1%8F:RPA_(Robotic_process_automation,_%D0%A0%D0%BE%D0%B1%D0%BE%D1%82%D0%B8%D0%B7%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%BD%D0%B0%D1%8F_%D0%B0%D0%B2%D1%82%D0%BE%D0%BC%D0%B0%D1%82%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F_%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81%D0%BE%D0%B2)>
6. https://bpm-systems.ru/what-is-rpa/