Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Дисциплина «Разработка программных роботов»

Студент: Глухова Д.В.

ФИТ 3 курс 2 группа

Преподаватель: Сазонова Д.В.

# Лабораторная работа № 3. Работа с информацией из изображений, сайтов, электронной почты

**Задание 3-1**. Создать процесс автоматизации в UiPath Studio, который прочитает отсканированное изображение счета и сохранит извлеченные данные в файле **.csv**

**Задание 3-1: Автоматизация чтения счета и сохранение в .csv**

**Создание процесса**:

* + Откройте UiPath Studio и создайте новый проект. Назовите его ImageCsv.

**Рабочая панель**:

* + Перенесите действие Sequence на рабочую панель. Это будет основной контейнер для всех ваших действий.

**Открытие окна "Выполнить"**:

* + Добавьте действие Send Hotkey:
    - Назовите его Открыть окно нажатием клавиш win + r.
    - В свойствах выберите флажок Win и введите r в поле Key. Это инициирует комбинацию клавиш Win + R, открывающую окно "Выполнить".

**Ввод пути к изображению**:

* + Добавьте действие Type Into:
    - В поле Text укажите путь к файлу изображения (например, "C:\Path\To\SampleTest.png").
    - Это действие автоматически введет указанный путь в поле "Выполнить".

**Открытие изображения**:

* + Добавьте еще одно действие Send Hotkey:
    - Назовите его Нажать кнопку OK, чтобы открыть изображение.
    - Введите enter в поле Key, чтобы подтвердить и открыть изображение.

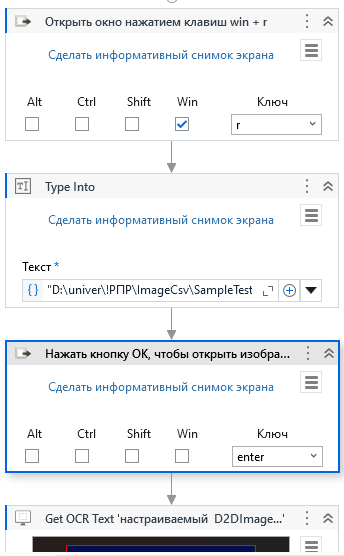
.

Рисунок 1.1 – Использование действий Send Hockey и Type Into

**Извлечение данных (InvoiceTotal)**:

* Добавьте действие Get OCR Text:
  + Удалите UiPath Screen OCR, заменив его на Microsoft OCR или Tesseract OCR.
  + Щелкните по ссылке Указать на экране и выделите область на изображении, содержащую сумму счета.
  + Создайте переменную InvoiceTotal типа String и укажите ее в поле Text, чтобы сохранить извлеченные данные.

**Извлечение данных (EmailAddress)**:

* Повторите шаг 6 для извлечения адреса электронной почты:
  + Добавьте еще одно действие Get OCR Text, выберите нужный OCR и выделите область с адресом электронной почты.
  + Создайте переменную EmailAddress типа String и укажите ее в поле Text.

**Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание**

Рисунок 1.2 – Использование действия Get OCR Text

**Создание таблицы данных**:

* Добавьте действие Build Data Table:
  + Создайте переменную ResultDT типа DataTable и укажите ее в поле DataTable.
  + Нажмите на кнопку DataTable, чтобы открыть редактор таблицы. Создайте два столбца: InvoiceTotal и Email, оба типа String.

**Изображение выглядит как текст, снимок экрана, дисплей, число

Автоматически созданное описание**

Рисунок 1.3 – Задание названий столбцов таблицы

**Добавление строки данных:**

* Добавьте действие Add Data Row:
* В свойстве ArrayRow укажите массив переменных, из которых будут взяты данные для заполнения таблицы: {InvoiceTotal, EmailAddress}.
* Укажите переменную ResultDT в свойстве DataTable, чтобы добавить новую строку в таблицу.

**Сохранение в .csv:**

* Добавьте действие Write CSV:
* Укажите переменную ResultDT в свойстве DataTable.
* В поле FilePath укажите путь и имя файла, в который будет сохранен результат (например, "C:\Path\To\Output.csv").

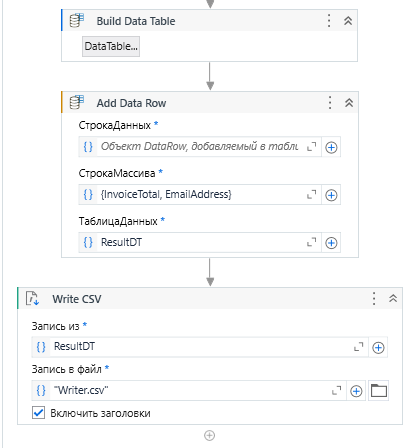


Рисунок 1.4 – Использование действий Build Data Table, Add Data Row, Write CSV

**Запуск процесса:**

Нажмите кнопку запуска, чтобы выполнить процесс автоматизации. Убедитесь, что все действия выполняются корректно и данные из изображения успешно извлекаются и сохраняются в .csv.

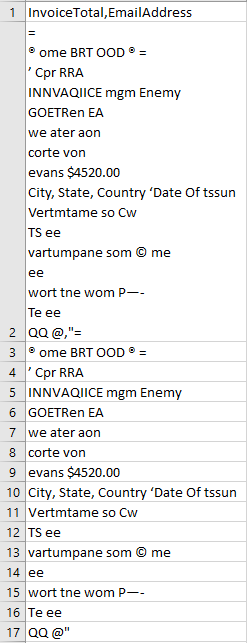


Рисунок 1.5 – Запись в файл Writer.csv

**Задание 3-2**. Создать процесс автоматизации, который будет извлекать информацию о сотрудниках с веб-сайта и сохранять ее в файле Excel.

**Создание процесса**:

* Создайте новый проект и назовите его FromWebToExcel.

**Рабочая панель**:

* Перенесите действие Sequence на рабочую панель.
* Создайте переменные:
  + Number (тип String)
  + EmailToSend (тип String)
  + URL (тип String, значение по умолчанию: "https://www.fakenamegenerator.com")
  + ExtractDataTable (тип DataTable).

Изображение выглядит как текст, Шрифт, снимок экрана, число

Автоматически созданное описание

Рисунок 1.6 – Создание переменных

**Ввод количества сотрудников**:

* Добавьте действие Input Dialog:
  + В Label введите "Количество".
  + В Title введите "Введите количество сотрудников".
  + В Result укажите переменную Number.

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок 1.7 – Использование действий Input Dialog, Build Data Table

**Создание таблицы данных**:

* Добавьте действие Build Data Table:
  + Укажите переменную ExtractDataTable в свойстве DataTable.
  + В редакторе создайте один столбец с именем Full Name (тип String).

**Открытие браузера**:

* Добавьте действие Open Browser:
  + Укажите переменную URL в свойстве Url.

**Переименование блока**:

* Переименуйте блок Do в Do – ‘Генерация фамилий сотрудников и запись их в таблицу'.

**Цикл извлечения данных**:

* Добавьте действие Do While:
  + Создайте переменную Val (тип Int32, значение по умолчанию 0).
  + В Condition укажите Val < CInt(Number).
  + Создайте переменную FullName (тип String).

**Увеличение счетчика**:

* Добавьте действие Assign:
  + Укажите Val в поле To и в Value введите Val + 1.

**Извлечение фамилии**:

* Добавьте действие Get Full Text:
  + Укажите элемент в браузере для извлечения фамилии.
  + В свойстве Text укажите переменную FullName.

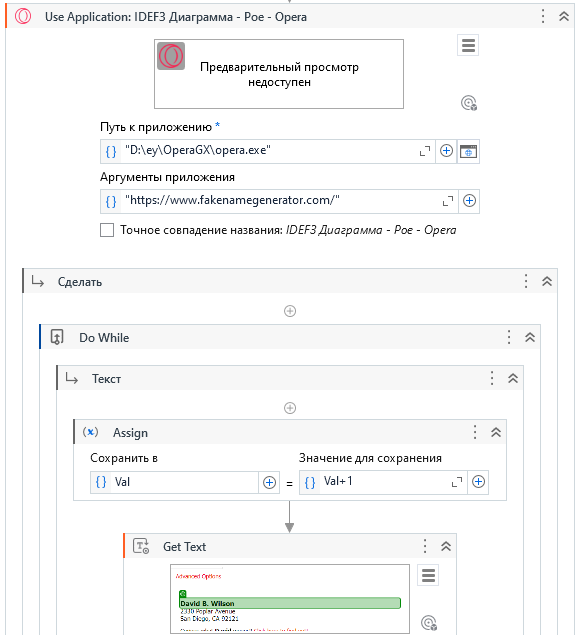
****

Рисунок 1.8 – Использование действий Use Browser, Do, Get Full Text

**Добавление строки данных**:

* Добавьте действие Add Data Row:
  + В ArrayRow укажите {FullName}.
  + Укажите переменную ExtractDataTable в свойстве DataTable.

**Клик для генерации новой фамилии**:

* Добавьте действие Click:
  + Укажите кнопку «Generate» на странице.

**Работа с Excel**:

* Добавьте действие Excel Application Scope:
  + Укажите имя файла Excel (например, "ExtrData.xlsx").

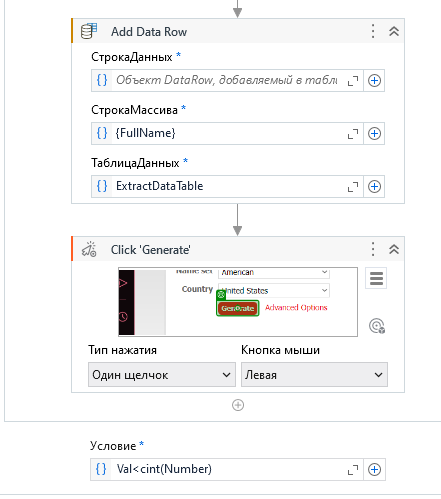


Рисунок 1.9 – Использование действий Add Data Row, Click

**Запись в Excel**:

* Добавьте действие Write Range:
  + В Sheet Name введите "Лист1".
  + В StartingCell укажите "A1".
  + В Value укажите переменную ExtractDataTable.

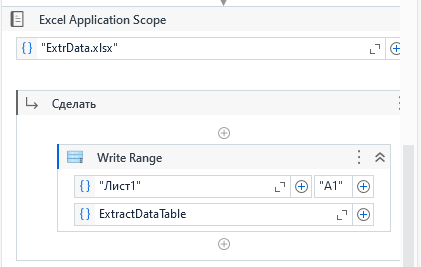


Рисунок 1.10 – Использование действий Excel Application Scope, Write Range и задание свойств

После запуска программы вылазит модальное окно, где нужно указать количество сотрудников.

Далее сгенерированные имена записываются в Excel-файл.



Рисунок 1.12 – Запись пользователь в Excel-файл

**Задание 3-3**. Создать процесс автоматизации для заполнения веб-формы данными, извлеченными из таблицы Excel.

Пусть имеется файл Excel **Firms.xlsx** с информацией о компаниях и заказанных товарах. Создаём простую страницу сайта **Zakaz.html**.

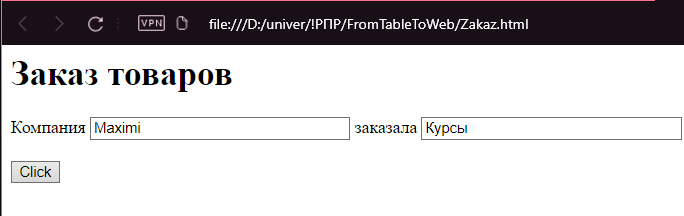


Рисунок 3.17 – Страница Zakaz.html

Открыть UiPath Studio, создать процесс и назвать его **FromTableToWeb**. Поместить в папку **FromTableToWeb** файлы **Zakaz.htm** и **Firms.xlsx**.

2. Перенести действие **Sequence** на рабочую панель.

3. Добавить действие **Open Browser**.

Указать в свойстве **Url** действия **Open Browser** ссылку на форму (**"D:\...\FromTableToWeb\Zakaz.htm"**).

4. Добавить действие **Maximize Window** в последовательность **Do**.

5. Добавить действие **Excel Application Scope**.

В свойство **WorkbookPath** записать имя таблицы Excel (**"Firms.xlsx"**), из которой будет извлекаться информация.

6. Добавить действие **Read Range**.

В свойстве **SheetName** **у**казать имя листа (**"Лист1"**), а в свойстве **Range** указать **""** (двойные кавычки).

Создать переменную **DtTable** типа **DataTable** и указать ее в поле **DataTable.**

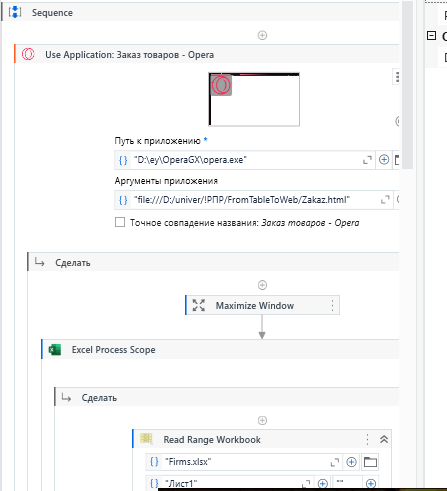


Рисунок 3.18 – Использование действий Use Browser, Maximize Window Excel Process Scope

7. Добавить действие **For Each Row in DataTable.**

Указать переменную **DtTable** в поле **In** (**в**) и **row** в поле **ForEach**.

Создать переменные **Company** и **Goods** типа **GenericValue**.

(для выбора типа: **Тип переменной** => **Выбор типов** => **GenericValue**).

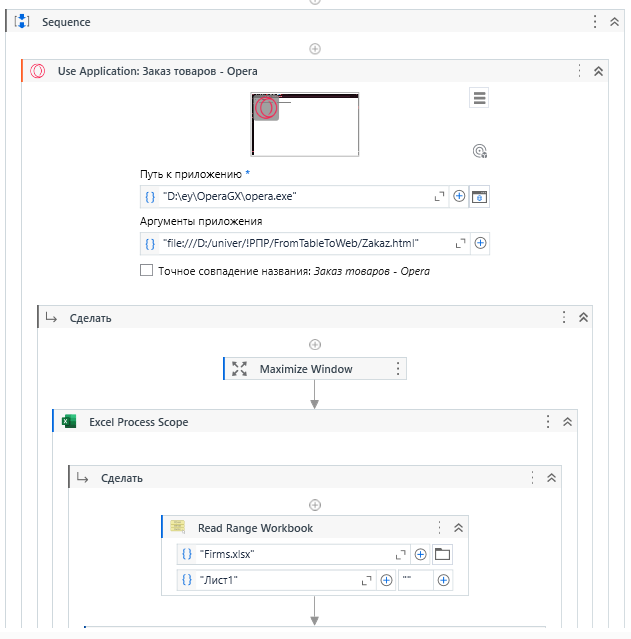


Рисунок 3.19 - Использование действий For Each Row in DataTable, Get Row Item

8. Добавить действие **Get Row Item** в контейнер **Body**.

Указать переменную **Company** в свойстве **Значение**, ввести текст **"Фирмы"** в свойстве **ColumnName**, в поле **Row** ввести **row**.

9. Добавить другое действие **Get Row Item**.

Указать переменную **Goods** в свойстве **Значение**, ввести текст **"Товары"** в свойстве **ColumnName**, в поле **Row** ввести **row**.

10. Добавить действие **Type Into.**

В свойствах указать переменную **Company** в поле **Text**.

Сделав двойной щелчок по файлу **Zakaz.htm** открыть в браузере Internet Explorer страницу с формами. Нажать ссылку **Указать элемент…** и выделить на странице сайта первое текстовое поле.

11. Добавить другое действие **Type Into.**

В свойствах указать переменную **Goods** в поле **Text**.

Нажать ссылку **Указать элемент…** и выделить на странице сайта второе текстовое поле.

12. Добавить действие **Click.**

Нажать ссылку **Указать элемент…** и указать кнопку **Click** на странице в браузере.

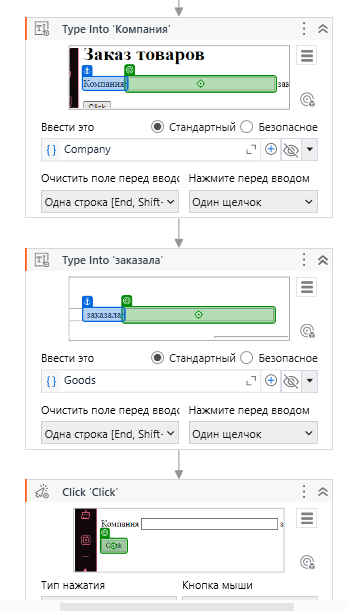


Рисунок 3.20 - Использование действий Type Into, Click

Во время выполнения программы открывается вкладка браузера на которой форма Zakaz.html. Робот начинает вводить значения с файла Firms.xlsx.

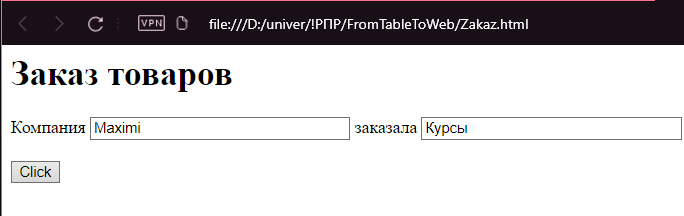


Рисунок 3.21 – Выполнение программы FromTableToWeb

**Задание 3-4**. Создать процесс автоматизации, который извлекает список статей по экономике с сайта Википедии с помощью инструмента **Считывание данных** (**Data Scraping**) и записывает список в таблицу Excel.

**Открытие браузера**:

* Запустите Internet Explorer и перейдите на сайт [Википедия](https://ru.wikipedia.org).

**Поиск статьи**:

* Введите "экономика" в поле поиска и выберите пункт "Поиск страниц, содержащих экономика".

**Создание проекта**:

* Создайте новый процесс в UiPath Studio и назовите его ArticleEconomic.

**Настройка Sequence**:

* Перенесите действие Sequence на рабочую панель.
* Добавьте действие Open Browser и в поле Url укажите URL-адрес страницы с результатами поиска.

**Считывание данных**:

* На ленте нажмите Считывание данных для открытия Мастера извлечения.
* Перейдите на веб-страницу, нажмите Далее, выберите первый элемент, затем последний элемент и снова нажмите Далее.

**Настройка столбцов**:

* В окне "Настроить столбцы" измените имя заголовка на Article Title и нажмите Далее.
* Введите максимальное количество результатов (например, 10) и нажмите Готово.
* Выберите "Нет" в вопросе о нескольких страницах.

**Обновление проекта**:

* Проект обновится, и в последовательности появится действие Считывание данных с автоматически созданной переменной ExtractDataTable. Измените область переменной на Sequence.

**Добавление Excel Application Scope**:

* После блока Считывание данных добавьте действие Excel Application Scope.
* В поле Путь к рабочему журналу укажите имя файла, например, "web\_scraping.xlsx".

**Запись в Excel**:

* В контейнере Do добавьте действие Write Range.
* В поле DataTable укажите переменную ExtractDataTable.

**Запуск процесса**:

* При выполнении программы откроется браузер, произойдет считывание данных, и они будут записаны в файл web\_scraping.xlsx.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 3.25 – Запись данных в Excel-файл

**Задание 3-5**. Создать процесс автоматизации, который получает из диспетчера учетных записей Windows данные пользователя для подключения к почтовому ящику, а затем отправляет письмо на этот же адрес с помощью действия **Send SMTP Mail Message**. Затем процесс автоматизации должен получить последние полученные 5 писем из этого ящика с помощью действия **Get IMAP Mail Messages**, сохранить текст в файле и вложения последнего пришедшего письма.

**Создание проекта**:

* Создайте новый процесс в UiPath Studio и назовите его MailActivities.
* Добавьте изображение imageUiPath.jpg в папку MailActivities.

**Настройка учетных данных**:

* Откройте Диспетчер учетных данных через Пуск/Панель управления.
* Нажмите "Добавить общие учетные данные" и введите название учетной записи, адрес электронной почты и пароль.

**Создание переменных**:

* Перенесите Sequence на рабочую панель.
* Создайте переменные:
  + mailUser (тип String)
  + mailPass (тип SecureString).

**Установка пакета**:

* В панели проекта нажмите "Управление пакетами".
* Найдите и установите пакет UiPath.Credentials.Activities.

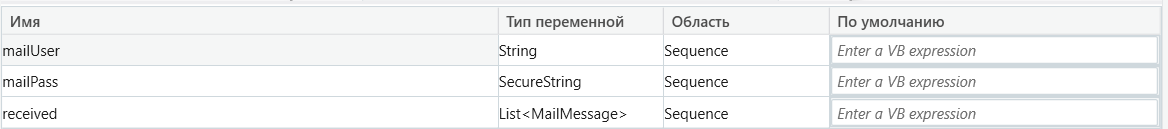


Рисунок 3.27 – Создание переменных

**Получение учетных данных:**

Добавьте действие Get Secure Credential:

В Target укажите имя учетной записи, Password — mailPass, Username — mailUser.

**Отправка почты:**

Добавьте действие Send SMTP Mail Message:

Укажите SMTP сервер и порт (например, "smtp.mail.ru", 465).

В Email укажите mailUser, а в Password — new System.Net.NetworkCredential(mailUser, mailPass).Password.

Введите адрес получателя в To, тему в Subject, текст в Body.

Прикрепите файл imageUiPath.jpg.

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок 3.28 – Использование действий Get Secure Credential, Send SMTP Mail Message

Изображение выглядит как текст, снимок экрана, программное обеспечение, Шрифт

Автоматически созданное описание

Рисунок 3.29 – Прикрепление изображения

* Добавьте действие Get IMAP Mail Messages:
  + Укажите IMAP сервер и порт (например, "imap.mail.ru", 993).
  + Укажите mailUser и new System.Net.NetworkCredential(mailUser, mailPass).Password.
  + Установите OnlyUnreadMessages и Top на 5.
  + В Messages укажите переменную receivedMails.

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок 3.30 – Свойства действия Get IMAP Mail Messages

**Сохранение сообщения**:

* Добавьте действие Save Mail Message:
  + В FilePath укажите "newMessage.eml", в MailMessage — receivedMails(0).

**Сохранение вложений**:

* Добавьте действие Save Attachments:
  + Укажите FolderPath (например, "attachments"), в Message — receivedMails(0).

Изображение выглядит как текст, снимок экрана, число, Шрифт

Автоматически созданное описание

Рисунок 3.31 – Использование действий Get IMAP Mail Messages, Save Mail Message, Save Attachments

Во время выполнения приложения программа подключается и аутифицируется под вашей учётной записью и отправляет указанный текст и прикрепленных файл. Потом это сообщение сохраняется в созданный вами файл “NewMessage.eml”

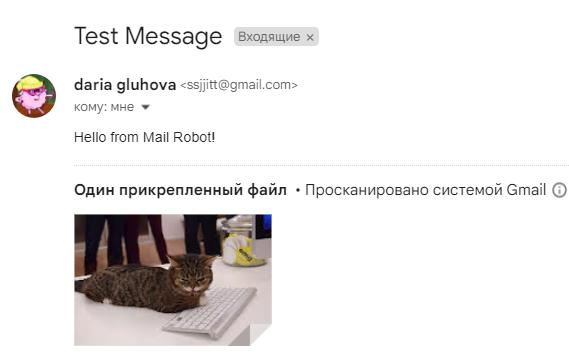


Рисунок 3.33 – Выполнение программы **MailActivities**

**Задание 3-6**. В задании 3-3 дополнить таблицу Excel исходными данными, включив информацию о нескольких товарах и их ценах. Добавить программу для кнопки Click (на языке VBScript или PHP или JScript), в которой вычисляется общая сумма заказанного товара. Отправить по электронной почте полученный результат заказчику.

Выполняем задание 3-3.

**Получение учетных данных:**

Добавьте действие Get Secure Credential:

В Target укажите имя учетной записи, Password — mailPass, Username — mailUser.

**Отправка почты:**

Добавьте действие Send SMTP Mail Message:

Укажите SMTP сервер и порт (например, "smtp.mail.ru", 465).

В Email укажите mailUser, а в Password — new System.Net.NetworkCredential(mailUser, mailPass).Password.

Введите адрес получателя в To, тему в Subject, текст в Body.

Прикрепите файл imageUiPath.jpg.

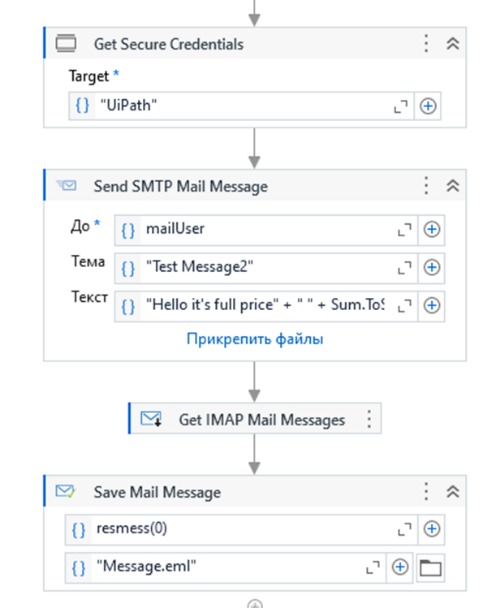


Рисунок 3.28 – Использование действий Get Secure Credential, Send SMTP Mail Message

* Добавьте действие Get IMAP Mail Messages:
  + Укажите IMAP сервер и порт (например, "imap.mail.ru", 993).
  + Укажите mailUser и new System.Net.NetworkCredential(mailUser, mailPass).Password.
  + Установите OnlyUnreadMessages и Top на 5.
  + В Messages укажите переменную receivedMails.

**Сохранение сообщения**:

* Добавьте действие Save Mail Message:
  + В FilePath укажите "newMessage.eml", в MailMessage — receivedMails(0).

**Сохранение вложений**:

* Добавьте действие Save Attachments:
  + Укажите FolderPath (например, "attachments"), в Message — receivedMails(0).