

# LECTURE 11.

# E-MAIL AUTOMATION

---

**Robotic Process Automation**

**[12 December 2023]**

Elective Course, 2023-2024, Fall Semester

Camelia Chisăliță-Crețu, Lecturer PhD

Babeș-Bolyai University

# Acknowledgements

This course is presented to our Faculty with the support of UiPath Romania.



# Contents

- E-mail
  - Need for Automation
- E-mail Automation
  - Details. Key Concepts
  - UiPath Activities
- E-mail Sending
  - The Process
- E-mail Interactions
  - Types
- UiPath Activities
  - Protocols. SMTP
  - POP3. IMAP
  - Microsoft Exchange
  - Microsoft Outlook
  - Generic Activities
- Demo 2A. E-mail Retrieval
- Demo 2B. E-mail Sending with Attachments
- References

# E-mail. Need for Automation

- need for Email Automation related to:
  - work;
  - personal life;
- types of e-mail that can be automated:
  - E-mails received and sent can be considered;
  - complexity vs. benefits of e-mail automation should be considered:
    - **complexity** indicates:
      - the number of steps,
      - application types involved other than e-mail,
      - number of possible scenarios (if/else),
      - input and output types (standard,
      - structured or unstructured);
    - **benefits** are related to:
      - time saving,
      - quality improvement,
      - error reduction.



# E-mail Automation. Details

- nowadays, many business processes are triggered by e-mail or generate e-mail output.
  - E-mail is the oldest and most wide-spread form of digital business communication;
  - many of these processes could easily be automated;
- some estimation results show that an average person spends between 5 and 11 hours a week checking, reading and responding to emails.
  - a research by the Radicati Group, Inc. ([https://www.radicati.com/wp/wp-content/uploads/2021/Email\\_Statistics\\_Report\\_2021-2025\\_Executive\\_Summary.pdf](https://www.radicati.com/wp/wp-content/uploads/2021/Email_Statistics_Report_2021-2025_Executive_Summary.pdf)) shows that 319.6 billion e-mails will be send/received each day by the end of 2021, there is a huge automation opportunity.
- What can we automate?
  - generating and sending automated messages;
  - retrieving messages and extracting data;
  - managing messages;
  - saving attachments;
  - saving messages.



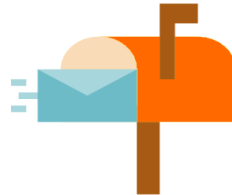
# E-mail Automation. Key Concepts



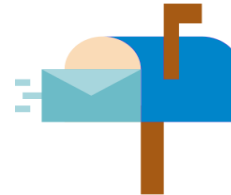
Email Client



SMTP



POP3



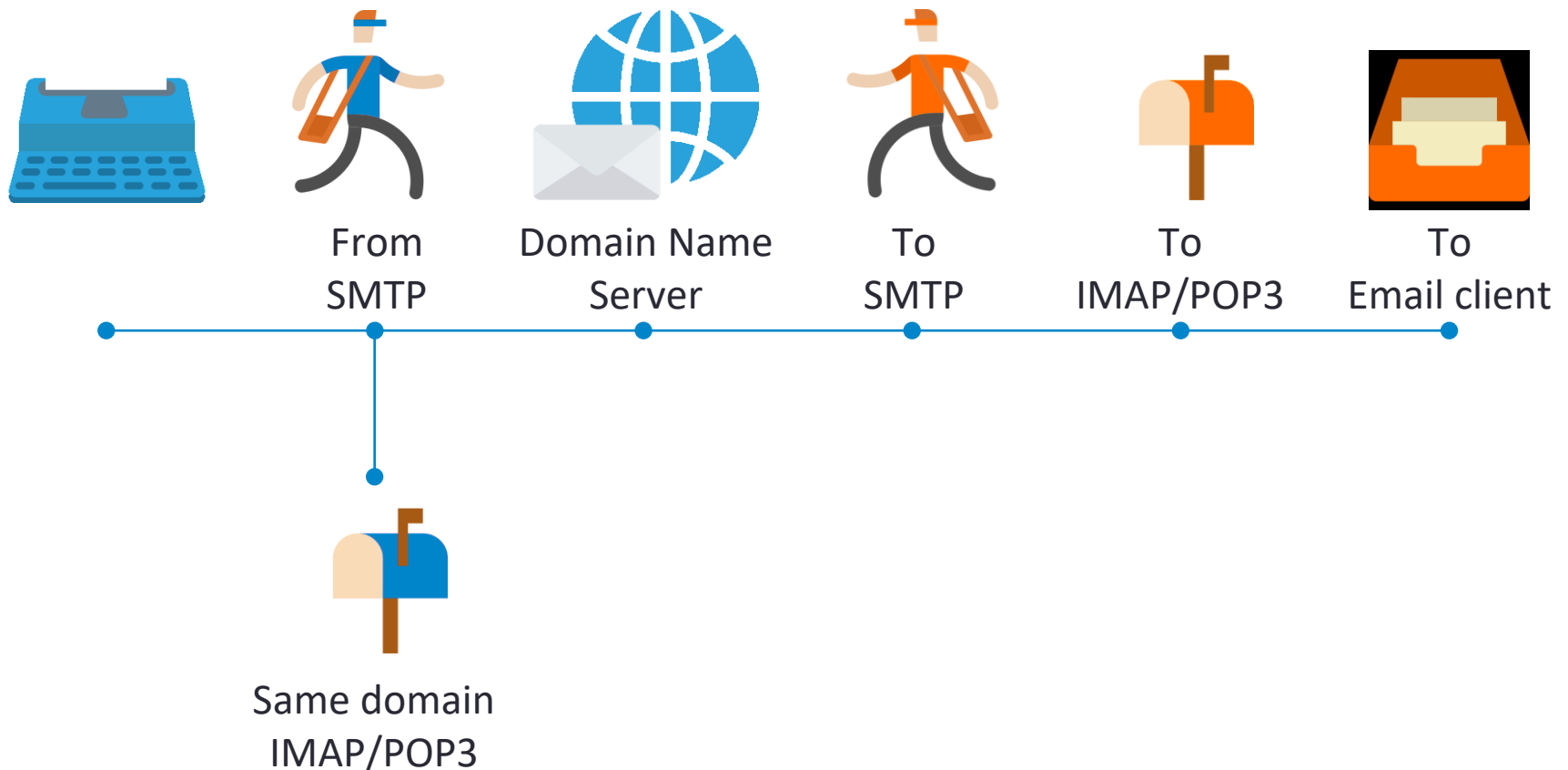
IMAP



Exchange

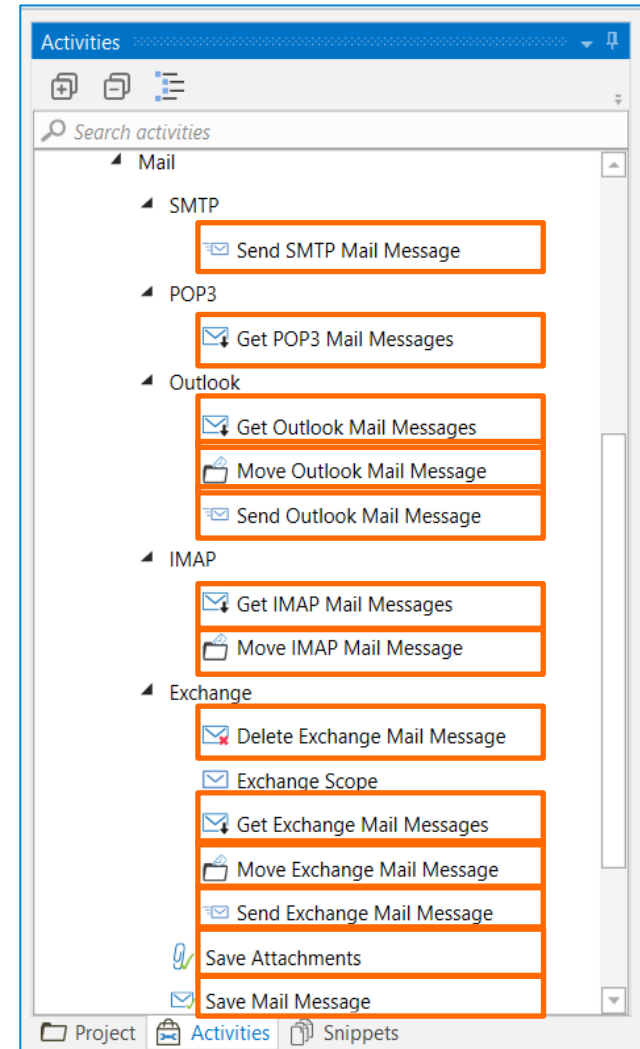


# E-mail Sending. The Process



# E-mail Automation. UiPath Activities

- UiPath Studio includes activities for:
  - **sending**,
  - **retrieving** and
  - **organizing** messages;
- there are several categories of activities:
  1. **generating and sending automated messages:**
    - SMTP, Outlook and Exchange
  2. **retrieving messages and extracting data:**
    - POP3, Outlook, IMAP and Exchange
  3. **managing messages:**
    - Outlook, IMAP, Exchange
  4. **saving attachments and emails:**
    - generic E-mail Activities.





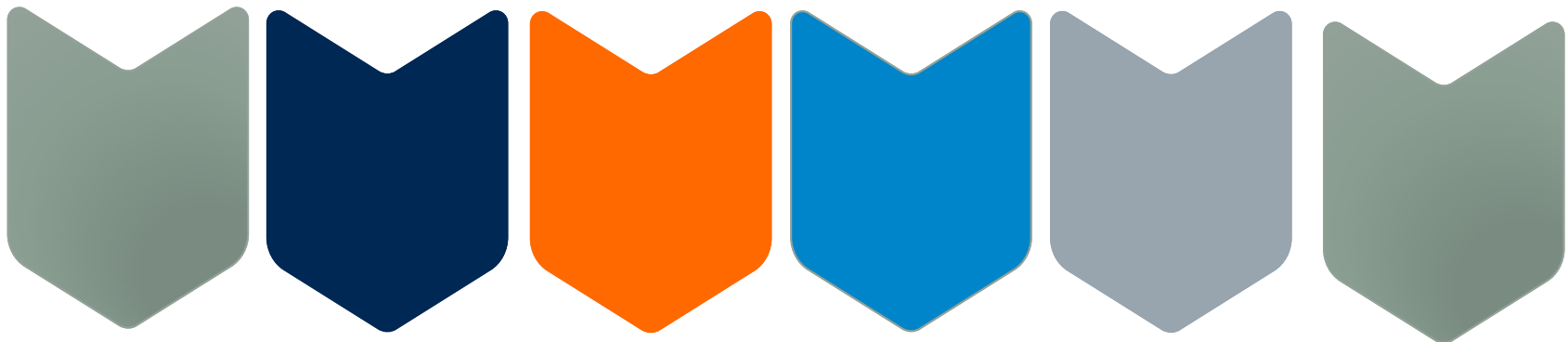
# E-mail Interactions. Types

- **Input** of the process **via an incoming email**
  - the automation gets specific data from incoming e-mails in order to perform a specific task, e.g.:
    - names and/or ID numbers from the **subject** or **body**;
    - input files coming as **attachments** (.xlsx, .pdf) that are downloaded;
    - **counting** the number of e-mails that meet a certain criteria.
- **Output** of the process as Email
  - the automation generates and sends an email according to the set rules, e.g.:
    - *progress reports* – sending an e-mail at a set time based on an Excel file;
    - *exception alerts* - email notification when a process comes across an exception.



# UiPath Activities. Protocols

- depending on the scope of automation, different sets of activities and protocols can be used;
- ***for using email as input***, depending on the client's setup, there are the following options:
  - POP3, IMAP, Outlook or Exchange;
- ***for using email as output***, there are the following options:
  - SMTP, Outlook, or Exchange.



SMTP

POP3

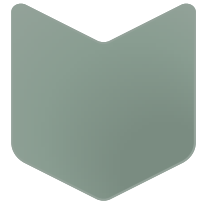
IMAP

Microsoft  
Exchange

Microsoft  
Outlook

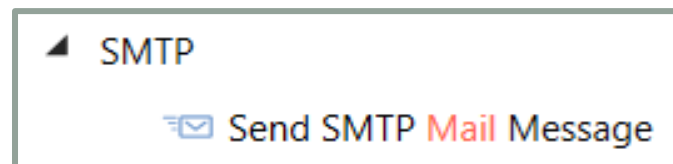
Generic  
Email  
Activities

# UiPath Activities. SMTP



SMTP

- **SMTP** stands for **Simple Mail Transfer Protocol**;
- mail servers and other mail transfer agents use the SMTP protocol **to send mail messages** on **TCP port 25**.
- **Scope:** E-mail as output;
  - E.g.: SMTP can be used with GMail to send emails;
- although proprietary systems (e.g., Microsoft Exchange, IBM Notes) and webmail systems (e.g., Outlook.com, Gmail and Yahoo! Mail) may use their own non-standard protocols internally, all use SMTP when sending or receiving email from **outside their own systems**.
- minimum requirements:
  - **To, Subject, Body, Port, Server, Email, Password.**



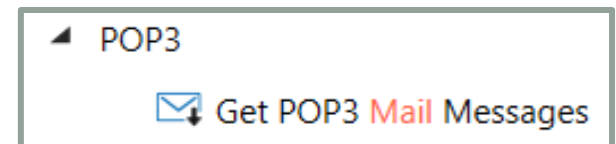
see Demo1B – EmailAutomation

# UiPath Activities. POP3

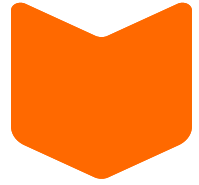


POP3

- **POP3** stands for **Post Office Protocol**;
- it is an old and almost obsolete protocol for reading messages, but most e-mail servers support it.
- a POP3 server listens on **port 110** for service requests.
- **Scope:** Input via e-mail;
  - E.g.: POP3 can be also used to retrieve email from GMail;
- it supports ***download-and-delete*** from a remote mailbox ("maildrop" in POP3 terminology); although most POP3 clients have an option to leave mail on server after download, they generally connect, retrieve all messages, store them on the client system, and delete them from the server;
- minimum requirements:
  - **Port, Server, Email, Password**;
  - **Top** – the number of messages to be retrieved starting from the top of the list.
  - **[Output] Messages** – the name of the **List<System.Net.Mail.MailMessage>** variable used to store the messages.



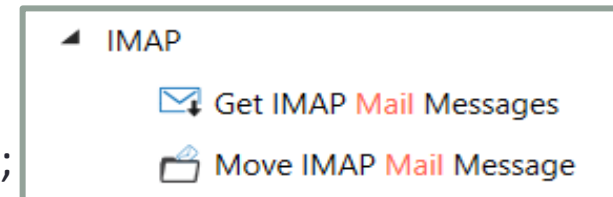
# UiPath Activities. IMAP



IMAP

- **IMAP** stands for **Internet Message Access Protocol**;
- it is only used for receiving messages;
- it offers some useful features to mark messages as read or move them between folders.
- an IMAP server listens on **port 143**; IMAP over SSL (IMAPS) has assigned the **port 993**.
- for retrieving messages, **IMAP** and **POP3** are standard, but proprietary servers also often prefer their own protocols, such as **Exchange ActiveSync**.
- **Scope:** Input via email;
  - E.g.: IMAP can be also used to retrieve email from GMail;
- IMAP was designed with the goal of permitting complete management of an email box by multiple email clients, therefore clients generally leave messages on the server until the user explicitly deletes them; all modern e-mail clients and servers support IMAP along with POP3;
- minimum requirements: **Port, Server, Email, Password**;
  - **Top** – the number of messages to be retrieved starting from the top of the list.
  - **[Output] Messages** – the name of the **List<System.Net.Mail.MailMessage>** variable used to store the messages;  

see **Demo2A – EmailAutomation**
  - the possibility to move the email to different inbox folders.



# UiPath Activities. Microsoft Exchange



Microsoft  
Exchange

- **Microsoft Exchange Server**
  - is a mail and calendaring server developed by Microsoft;
  - it runs exclusively on Windows Server operating systems;
  - primarily uses a proprietary protocol called MAPI to talk to email clients, but subsequently added support for POP3, IMAP, and Exchange ActiveSync (EAS); the standard SMTP protocol is used to communicate to other Internet mail servers;
- **Scope:** sends and receive emails;
- minimum requirements:
  - create and define an **Exchange Scope** activity;
  - include other email activities required in the **Exchange Scope** activity container.

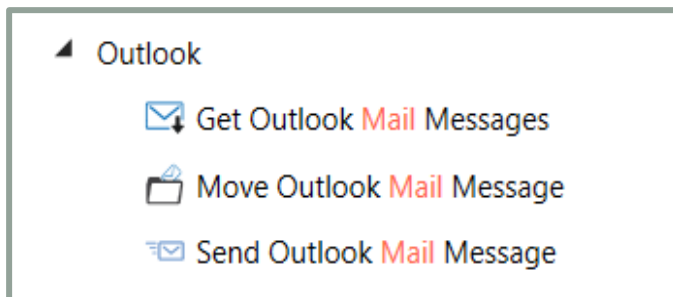


# UiPath Activities. Microsoft Outlook (1)



Microsoft  
Outlook

- **Microsoft Outlook**
  - is a personal information manager developed by Microsoft;
  - this is mainly utilized as **email client** and **manager**, but it also come with calendar, task manager scheduling meeting request and many other facilities.
- **Scope:** Input via email and email as output.
- **MS Outlook** activities work with the API of the desktop application, so the actions already have a context; there is no need to set up servers, users, and other technical details; the activities are designed to use the already existing Outlook accounts;

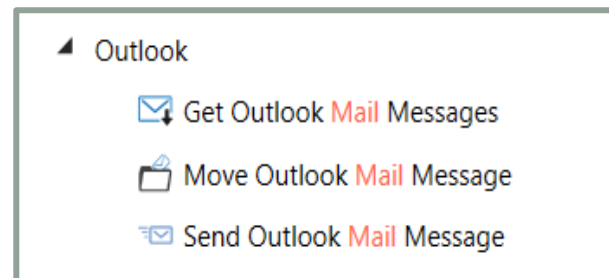


# UiPath Activities. Microsoft Outlook (2)



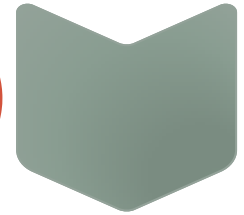
Microsoft  
Outlook

- **Get Outlook Mail Messages** activity
  - **Account**: the email address;
  - **MailFolder**: Generally this folder is "Inbox", but it can be any folder from the Outlook application;
  - **MailMessages**: the output variable which is a list of **MailMessages**;
  - **MarkAsRead**: the messages that are read will be checked as *read* in Outlook;
  - **OnlyUnreadMessages**: to get only the emails that were not read;
  - **Top**: to indicate the number of emails to be extracted.
- **Send Outlook Mail Messages** activity
  - **Body**: the body of the email to be sent;
  - **Subject**: the subject of the email to be sent;
  - **MailMessage**: it can be a **MailMessage** object, that can be forwarded thorough this email without completing the subject and the body;
  - **Account**: the email address; this is considered as the sender address;
  - **CC**: the addresses to include in the 'Cc' field of the e-mail; for more e-mail addresses, comma is used to separate them.



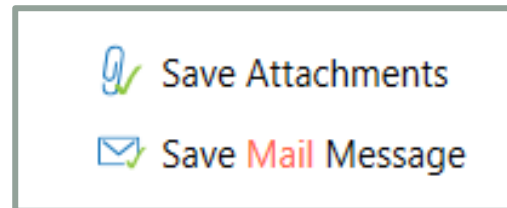


# UiPath Activities. Generic Activities (1)

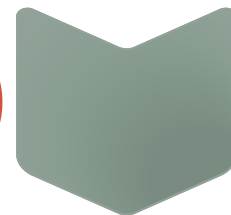


Generic  
Email  
Activities

- Generic activities:
  - **Save Mail Message** activity;
  - **Save Attachments** activity;
- the activities use **System.Net.MailMessage** type in order to save the e-mail as **.eml** and also save the attachments.

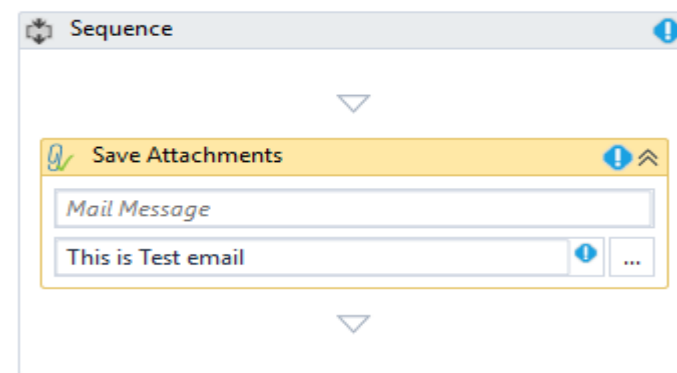
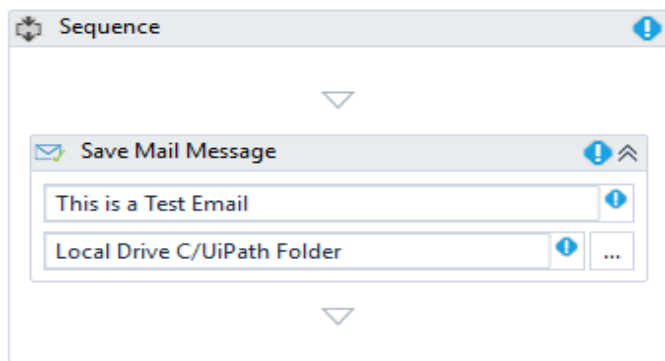


# UiPath Activities. Generic Activities (2)



Generic  
Email  
Activities

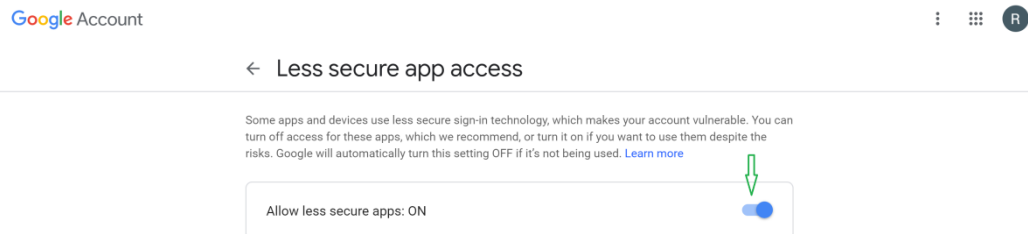
- **Save Mail Message** activity
  - **FilePath**: the path where the **MailMessage** object is to be saved;
  - **MailMessage**: the **MailMessage** object to be saved;
- **Save Attachments** activity
  - **FolderPath**: the path of the folder where the attachments are to be saved;
  - **Message**: the **MailMessage** object whose attachments are to be saved.



see Demo1B – EmailAutomation

# Demo 2A. E-mail Retrieval

- Use **Get IMAP Mail Messages** activity to retrieve e-mails from the **gmail.com** using:
  - IMAP, where the server is **imap.gmail.com**, port **993**;
  - username: **rpaubb**
  - password: **UBBRPA2022\_123**
- **Read** e-mails from “Inbox” where the Subject contains the keyword *errors* and attachments;
- **Save** the attachments into the folder “Attachments/<error type>”.
- Note that in the Google account, menu **Account Setting**, the option **Less secure app access** should be **enabled**.



see Demo1A – EmailAutomation

# Demo 2B. E-mail Sending with Attachments

- Use **Send SMTP Mail Messages** activity:
  - to the e-mail address [vvta@scs.ubbcluj.ro](mailto:vvta@scs.ubbcluj.ro) or [rpaubb@gmail.com](mailto:rpaubb@gmail.com)
    - user: *vvta*, password: *VVTA2020*
  - *to send* formatted e-mails
    - **Read Text File** activity;
    - `String.Format(messageContent, item)`
  - with an attachment file:
    - **Take Screenshot** activity;
    - **Save Image** activity;
  - from the username: *rpaubb* with password: *UBBRPA2022\_123*
  - from the **gmail.com** using:
    - SMTP, where the server is *smtp.gmail.com*, port 465.
- Note that in the Google account, menu **Account Setting**, the option **Less secure app access** should be **enabled**.

see Demo1B – EmailAutomation

# Next lecture

- **Orchestrator - Part 1 & 2**

# References

- UiPath Docs - <https://docs.uipath.com/studio>
  - Email Automation - <https://docs.uipath.com/activities/other/latest/productivity/mail-activities>