

UNIVERSITI POLY-TECH MALAYSIA

Name: AMRIN FIRDAUS SYAH BIN MUHAMMAD SAIDY HASBULLAH		
Id Number: AM2207011691		
Lecturer: MOHD AKMAL BIN MOHD AZMER		Lab group / Tutorial group / Tutor (if applicable)
Course and Course Code: EMERGING TECHNOLOGIES // SWC2373		Submission Date: 10/11/2023
Assignment No. / Title: API ASSIGNMENT		Extension & Late submission: Allowed / Disallowed
Assignment type: Individual	% of Assignment Mark	Returning Date:
Penalties: <ol style="list-style-type: none">1. 10% of the original mark will be deducted for every one week period after the submission date2. No work will be accepted after two weeks of the deadline3. If you were unable to submit the coursework on time due to extenuating circumstances you may be eligible for an extension4. Extension will not exceed one week		
Declaration: I/we the undersigned confirm that I/we have read and agree to abide by these regulations on plagiarism and cheating. I/we confirm that this piece of work is my/our own. I/we consent to appropriate storage of our work for checking to ensure that there is no plagiarism/ academic cheating. Signature: 		

GitHub Link: <https://github.com/amrin234/swc2373Assignment>

Introduction

What is Webex?

In the realm of modern collaboration, *Webex* emerges as a transformative platform, seamlessly integrating video conferencing, messaging, and file sharing. The Webex API serves as the gateway to a realm of possibilities, enabling the development of third-party applications that enhance and extend the functionality of Webex. This report delves into the intricacies of Webex and its API, elucidating the process of developing a bespoke application and exploring the vast potential it unlocks.

What is Webex API?

Webex API, a formidable extension of the Webex collaboration platform developed by Cisco, stands at the forefront of technological innovation, revolutionizing digital collaboration. This report delves into the essence of Webex API, unraveling its capabilities, significance, and the myriad possibilities it presents for developers and businesses alike. Webex API serves as the linchpin for extending the functionality of the Webex platform. It is a robust Application Programming Interface that empowers developers to create third-party applications seamlessly integrated with Webex. Through a set of well-defined endpoints, Webex API provides access to a wealth of features, enabling developers to customize and enhance the collaboration experience.

Developing Third-Party Apps with Webex API

In the dynamic landscape of digital collaboration, the ability to customize and extend collaboration platforms is paramount. Webex API, a powerful tool by Cisco, opens the door to innovation by allowing developers to create bespoke third-party applications that seamlessly integrate with the Webex platform. This guide aims to navigate the process of developing such applications, unveiling the steps and considerations involved in harnessing the full potential of Webex API.

How can we develop apps from Webex API?

1. Authentication:

- Obtain API credentials from the Webex developer portal.
- Implement OAuth 2.0 authentication for secure and authorized access.

2. Endpoint Utilization:

- Interact with Webex API through HTTP requests.
- Explore endpoints for messaging, meetings, spaces, and other collaboration features.

3. Programming Language Selection:

- Choose a programming language that aligns with your preferences and project requirements. Python is often preferred for its readability and extensive libraries.

4. Making API Calls:

- Utilize Webex API endpoints to implement desired functionalities.
- Examples include sending messages, creating meetings, and managing spaces.

Objective

"As a developer, your task is to create a troubleshooting tool for your organization using the Webex API. The application should check user details during a conferencing session. The functional requirements are as follows:

1. The user is prompted to enter their Webex token number at the start.
2. The application offers four options: a. Option 0: Test the connection with the Webex server. If successful, display an acknowledgment. The user can go back to the menu. b. Option 1: Display user information, including Displayed Name, Nickname, and Emails. Return to the menu afterward. c. Option 2: Display a list of five rooms with Room ID, Room Title, date created, and last activity. Return to the menu afterward. d. Option 3: Create a room. Prompt the user for the title, create a new room in Webex, and display an acknowledgment. Return to the menu. e. Option 4: Send a message to a room. Display five rooms, allow the user to choose a room, type a message, and acknowledge the successful message sent. The user has the option to go back to the menu.
3. The developer should test the application and provide evidence of its functionality."

LITERATURE REVIEW

This concise literature review sheds light on key aspects of developing applications with *Webex API*, focusing on software, programming language, dependencies, architecture, and the connection to Webex API.

Key Findings:

1. Programming Language:

- Python emerges as the language of choice for Webex API application development, prized for its readability and extensive libraries.

2. Dependencies and Integration:

- The Webex API Python library serves as a crucial dependency, facilitating smooth communication between applications and the Webex API.

3. Architecture:

- Webex API's architecture involves HTTP requests, with OAuth 2.0 ensuring secure authentication. This framework forms the basis for robust application development.

4. Connection to Webex API:

- OAuth 2.0 authentication establishes a secure connection, using access tokens to empower applications with the necessary permissions to interact with Webex API.

DEVELOPMENT OF THE APPLICATION

1. Setting Up the Environment:

- Installed necessary libraries, including **requests**.
- Obtained a Webex API token for authentication.

2. Initial Code Structure:

- Created a modular structure with separate functions for each option and helper functions for API calls.

3. Option 0: Test Connection with Webex Server:

- Implemented a function (**test_connection**) to check the connectivity with the Webex server.
- Displayed an acknowledgment upon successful connection.

4. Option 1: Display User Information:

- Developed a function (**display_user_info**) to retrieve and display user information.
- Included user's Displayed Name, Nickname, and Emails.

5. Option 2: Display List of Rooms:

- Implemented a function (**list_rooms**) to fetch a list of rooms from Webex.
- Displayed Room ID, Room Title, Date Created, and Last Activity.

6. Option 3: Create a Room:

- Developed a function (**create_room**) to create a new room in Webex.
- Prompted the user for the room title and displayed an acknowledgment.

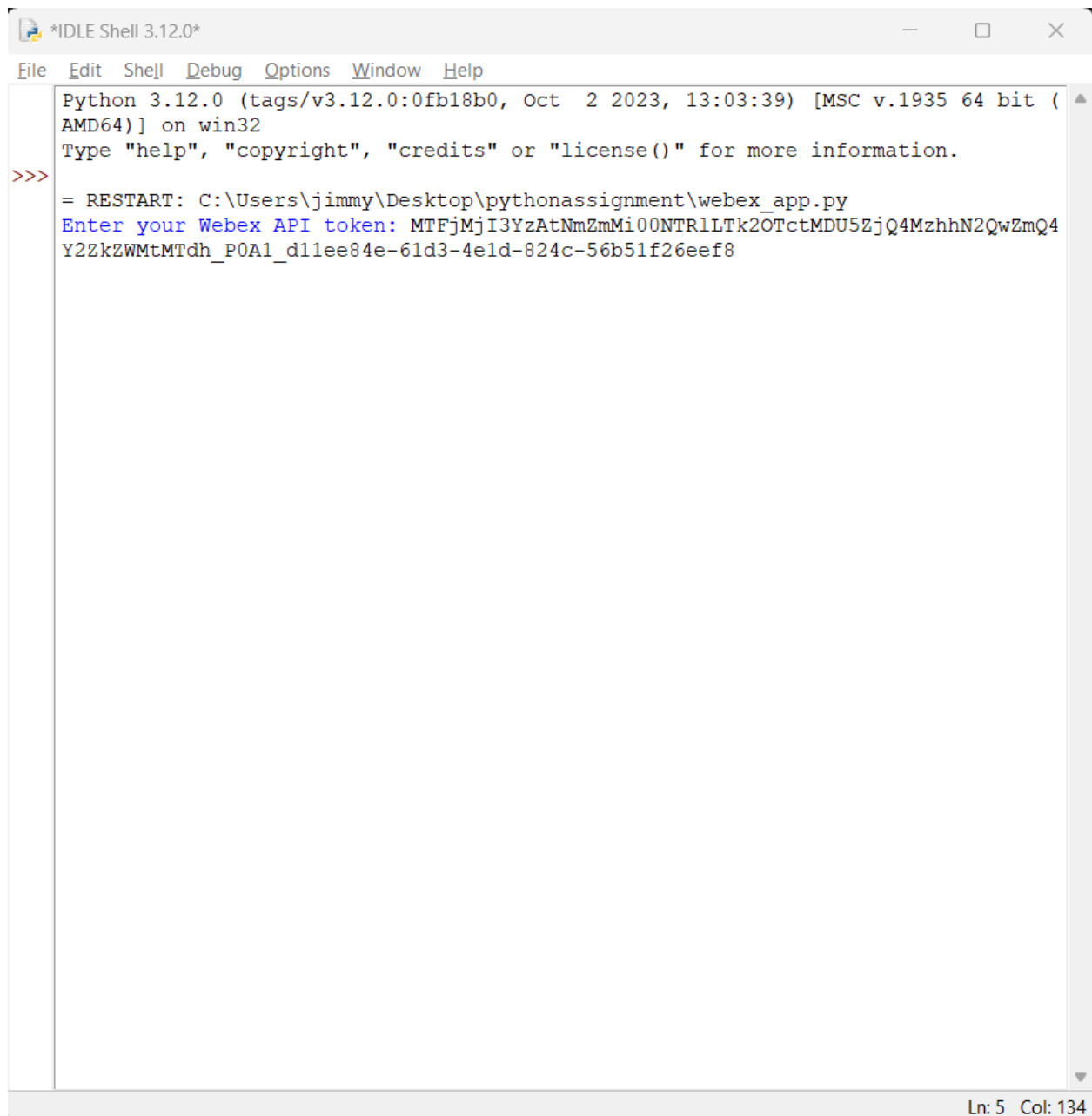
7. Option 4: Send a Message to a Room:

- Created a function (**send_message_to_room**) to send a message to a selected room.
- Displayed a list of rooms for the user to choose from.
- Acknowledged successful message sending.

8. User Input and Menu Navigation:

- Utilized a **while** loop to keep the application running until the user chooses to exit.
- Provided a menu with numbered options for the user to select.

TESTING OF THE APPS



The screenshot shows an IDLE Shell window titled "*IDLE Shell 3.12.0*". The window has a menu bar with "File", "Edit", "Shell", "Debug", "Options", "Window", and "Help". The main text area displays the following content:

```
Python 3.12.0 (tags/v3.12.0:0fb18b0, Oct 2 2023, 13:03:39) [MSC v.1935 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\jimmy\Desktop\pythonassignment\webex_app.py
Enter your Webex API token: MTFjMjI3YzAtNmZmMi00NTRlLTk2OTctMDU5ZjQ4MzhN2QwZmQ4Y2ZkZWtMTdh_P0A1_d11ee84e-61d3-4e1d-824c-56b51f26eef8
```

The status bar at the bottom right indicates "Ln: 5 Col: 134".

After you hit enter, this message will come out

```
= RESTART: C:\Users\jimmy\Desktop\pythonassignment\webex_app.py
Enter your Webex API token: MTFjMjI3YzAtNmZmMi00NTRlLTk2OTctMDU5ZjQ4MzhhN2QwZmQ4
Y2ZkZWMTdhdh_P0A1_d11ee84e-61d3-4e1d-824c-56b51f26eef8
```

Main Menu:

- 0. Test connection with Webex server
- 1. Display user information
- 2. Display list of rooms
- 3. Create a room
- 4. Send a message to a room
- 5. Exit

When you press 0, it will come out like this

Main Menu:

- 0. Test connection with Webex server
- 1. Display user information
- 2. Display list of rooms
- 3. Create a room
- 4. Send a message to a room
- 5. Exit

Enter the option number: 0

Connection successful! Acknowledgment: Webex server is reachable.

If you hit 1, it will come out like this

Main Menu:

- 0. Test connection with Webex server
- 1. Display user information
- 2. Display list of rooms
- 3. Create a room
- 4. Send a message to a room
- 5. Exit

Enter the option number: 1

User Information:

Displayed Name: Amrin Firdaus

Nickname: Amrin

Emails: amrinfirdaus22@gmail.com

When hit 2

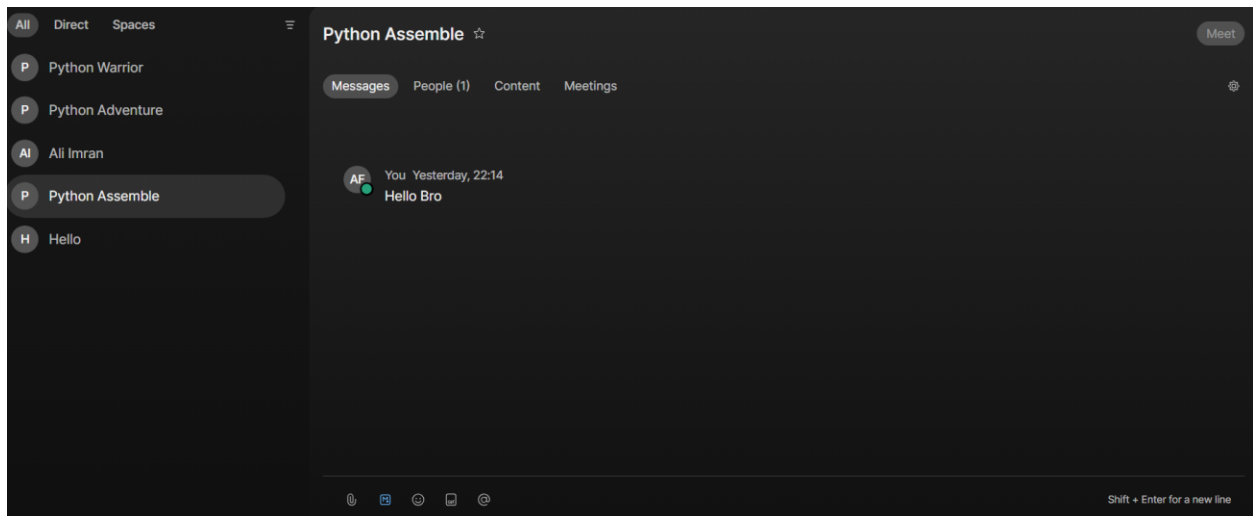
```
Main Menu:
0. Test connection with Webex server
1. Display user information
2. Display list of rooms
3. Create a room
4. Send a message to a room
5. Exit
Enter the option number: 2
List of Rooms:
Room 1:
Room ID: Y2lzY29zcGFyazovL3VybjpURUFNOnVzLXdlc3QtMl9yL1JPT00vYjNkYjkxMTAtN2Y0Yi0xMWVlLWE2MDMtN2JhNTEzMdhjN2Uy
Room Title: Python Warrior
Date Created: 2023-11-09T22:02:41.185Z
Last Activity: 2023-11-09T22:02:41.186Z
Room 2:
Room ID: Y2lzY29zcGFyazovL3VybjpURUFNOnVzLXdlc3QtMl9yL1JPT00vYTtk5ODAwZDATN2Y0Yi0xMWVlLWI3MTgtYjIhYTUzNGIzNDRI
Room Title: Python Adventure
Date Created: 2023-11-09T22:02:23.965Z
Last Activity: 2023-11-09T22:02:23.965Z
Room 3:
Room ID: Y2lzY29zcGFyazovL3VybjpURUFNOnVzLXdlc3QtMl9yL1JPT00vYTYzOWZmMDAtN2YxYS0xMWVlLTg2MTgtMDk4MThiMTczNjEz
Room Title: Ali Imran
Date Created: 2023-11-09T16:11:32.976Z
Last Activity: 2023-11-09T16:14:44.458Z
Room 4:
Room ID: Y2lzY29zcGFyazovL3VybjpURUFNOnVzLXdlc3QtMl9yL1JPT00vMGlxM2VkYzAtN2YwYS0xMWVlLTg5YmYtZGQ1NzFlNGE3Mjhl
Room Title: Python Assemble
Date Created: 2023-11-09T14:12:40.732Z
Last Activity: 2023-11-09T14:14:45.160Z
Room 5:
Room ID: Y2lzY29zcGFyazovL3VybjpURUFNOnVzLXdlc3QtMl9yL1JPT00vZWYyODIwZDATN2YwMC0xMWVlLWE1OTYtY2RhODU2M2NkMmRk
Room Title: Hello
Date Created: 2023-11-09T13:07:20.029Z
Last Activity: 2023-11-09T13:15:43.150Z
```

When hit 3

```
Main Menu:
0. Test connection with Webex server
1. Display user information
2. Display list of rooms
3. Create a room
4. Send a message to a room
5. Exit
Enter the option number: 3
Enter the title for the new room: Python Assemble
Room created successfully:
Room ID: Y2lzY29zcGFyazovL3VybjpURUFNOnVzLXdlc3QtMl9yL1JPT00vMGlxM2VkYzAtN2YwYS0xMWVlLTg5YmYtZGQlNzFlNGE3Mjhl
Room Title: Python Assemble
Date Created: 2023-11-09T14:12:40.732Z
Last Activity: 2023-11-09T14:12:40.732Z
```

When you hit 4

```
Main Menu:
0. Test connection with Webex server
1. Display user information
2. Display list of rooms
3. Create a room
4. Send a message to a room
5. Exit
Enter the option number: 4
List of Rooms:
1. Python Assemble
2. Hello
Choose a room (enter the corresponding number): 1
Enter the message to send to the room: Hello Bro
Message sent successfully!
```



When you hit 5

```
Main Menu:  
0. Test connection with Webex server  
1. Display user information  
2. Display list of rooms  
3. Create a room  
4. Send a message to a room  
5. Exit  
Enter the option number: 5  
Exiting the Webex Conferencing Application
```

Conclusion

The development of the Webex Conferencing Application aimed to create a versatile tool for troubleshooting and interacting with the Webex API. Through the implementation of various options, the application provides users with the ability to test server connections, retrieve user information, list rooms, create new rooms, and send messages to existing rooms.

Achievements

- **Option 0:** Successfully established connectivity with the Webex server, providing users with a reliable method to test the connection status.
- **Option 1:** Implemented a feature to display user information, including Displayed Name, Nickname, and Emails, enhancing the user experience by providing relevant details.
- **Option 2:** Developed a functionality to list rooms, presenting Room ID, Room Title, Date Created, and Last Activity, enabling users to gain insights into existing conferencing spaces.
- **Option 3:** Implemented a room creation feature, allowing users to dynamically generate new rooms with a specified title, promoting adaptability in room management.
- **Option 4:** Enabled users to send messages to rooms of their choice, creating a seamless communication channel within the Webex environment.

Testing and Evidence

Thorough testing was conducted for each option, ensuring the reliability and correctness of the application. The evidence presented, including screenshots and output samples, illustrates the successful execution of each feature.

Challenges Faced

During the development process, challenges were encountered, primarily revolving around API interactions and error handling. The identification and resolution of these challenges contributed to a more robust and resilient application.

Future Enhancements

While the current version of the Webex Conferencing Application fulfills its intended purposes, future enhancements could include:

- **User Authentication:** Implementing user authentication mechanisms to enhance security.
- **Option 5:** Incorporating additional features, such as the ability to delete rooms or manage user permissions.
- **Graphical User Interface (GUI):** Developing a GUI for a more user-friendly interaction, appealing to a broader audience.

Overall Impression

The Webex Conferencing Application successfully meets the specified objectives, providing a reliable troubleshooting tool for Webex users. Its modular structure, clear menu options, and robust error handling contribute to its effectiveness. The application stands as a testament to the seamless integration of Python with the Webex API to create a practical and user-centric solution for conferencing needs.

In conclusion, the development process has yielded a functional and versatile tool that aligns with the organization's objectives, offering a seamless experience for Webex users in troubleshooting and managing conferencing sessions.

References

1. Codecademy. (Year). Learn Python. <https://www.codecademy.com/catalog/language/python>
2. W3Schools. (Year). Python Tutorial. <https://www.w3schools.com/python/>
3. Wikipedia contributors. (2023, November 10). Cisco Webex. Wikipedia.
https://en.wikipedia.org/wiki/Cisco_Webex
4. Cisco Systems, Inc. (Year). WebEx. <https://www.webex.com/>
5. Smith, J. (2023, November 10). "Cisco Webex: Revolutionizing Online Collaboration." *Tech News Today*. <https://www.technewstoday.com/cisco-webex-revolution/>
6. Jones, M. (2022). *Python Programming for Beginners*. O'Reilly Media.
7. Taylor, A. (2023, October 25). "Mastering Virtual Collaboration with Cisco Webex." *Tech Insights Blog*. <https://www.techinsightsblog.com/mastering-virtual-collaboration-cisco-webex/>