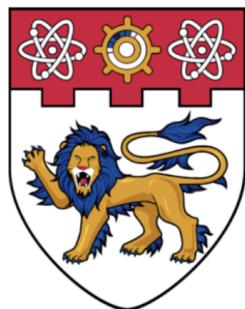


NANYANG TECHNOLOGICAL UNIVERSITY

SC4052: Cloud Computing



**NANYANG
TECHNOLOGICAL
UNIVERSITY
SINGAPORE**

Academic Year: 2024 Semester 2

CHONG JIA YEE | N2402296L

BEH HANYU | N2402280J

Project:

AI Chatbot that generates ATS-friendly resume

Topic 3:

Conversational Chatbot-as-a-Service and Low-Code-No-Code Principle

Table of Contents

Introduction.....	3
Problem Statement.....	3
Proposed Solution.....	6
Solution Architecture.....	8
Main Workflow + Plugin : Converting Json Data to Docx File.....	8
Sub-Workflow: Formatting User Input.....	12
Other Plugins: Resume Reader.....	13
Other Plugins: Linkedin Authorizer.....	13
Examples of different scenarios.....	16
Links.....	23
Discussion on Low-Code-No-Code Development.....	23
Conclusion and Future Enhancements.....	23

Introduction

Social messaging has become indispensable in human communications, and future cloud computing applications are likely to become more conversational in nature. With the rise of Generative AI (Gen-AI) technologies like ChatGPT, cloud companies such as SAP are investing in "low-code-no-code" (LCNC) Software-as-a-Service (SaaS) platforms to enable users with minimal coding experience to develop applications. This report explores the principles of low-code-no-code development and demonstrates an AI-powered chatbot service for resume generation.

Problem Statement

In the competitive job market, individuals often spend considerable time creating and maintaining their resumes. To ensure their resumes are professional and applicant tracking system (ATS)-friendly, job seekers need to format their documents correctly. ATS is commonly used by HR professionals to screen resumes, and to avoid stereotypes, it is recommended that resumes exclude personal images.

Additionally, keeping resumes updated can be cumbersome. Professionals frequently need to track down their original soft copies to manually update their experience, achievements, or contact information. Fresh graduates, in particular, tend to update their LinkedIn profiles to increase their online presence for job opportunities. However, manually synchronizing these updates across platforms is inefficient.

To better understand the gaps in current solutions, we tested various online resume generator platforms and identified the following issues.

Issues with Existing Resume Generator Websites:

- Premium Features Lock-In: Most web apps impose certain charges in order for users to download their generated resume. While this can be a drawback for users seeking free services, it is an understandable model for ensuring scalability. Similarly, our application may adopt a tiered pricing structure to sustain development and operational costs.

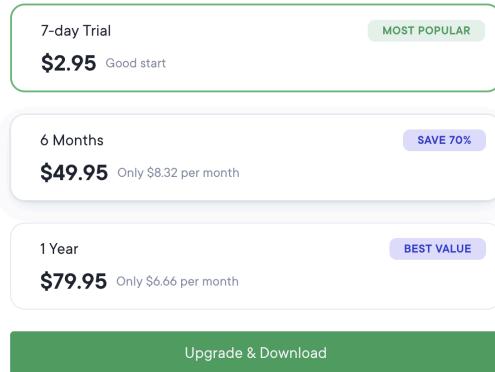


Diagram 2.1: Subscription Plans in Resume Generators

- Strict Input Content: Most resume generators require users to enter information into rigid, pre-defined text areas and sections, limiting flexibility in content display.

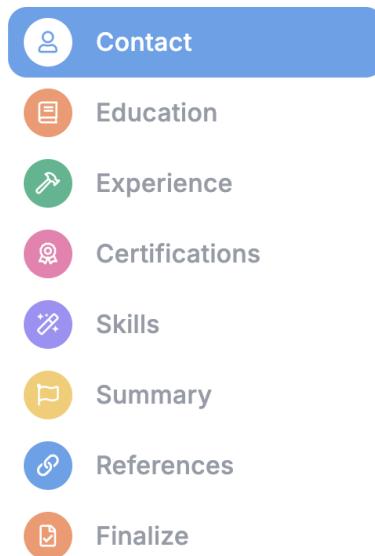


Diagram 2.2: Fixed Sections in Resume Generators

- Weird Templates: Many available templates are not professional-grade, often including unnecessary elements such as user photos, excessive colors, or animated designs that do not align with industry standards.



Diagram 2.3: Unnecessary Photo Inclusion in Template

- Lack of Data Integration: Users must manually input all details, as most platforms do not support LinkedIn imports.

The form is titled "Contact". It starts with a note: "Let's start with the basics. To ensure employers can reach you, input at least your name, email, and phone number." There is a placeholder for a photo with the text "Add a Photo to Your Resume (Optional)" and a "Add Photo" button. Below that are fields for "First Name" (placeholder "e.g. Yong") and "Last Name". There is also a field for "Desired Job Title (Optional)". Further down are fields for "Phone", "Country" (with a dropdown arrow), "City", "State or Region", and "Postal Code".

Diagram 2.4: Fields in Resume Generator

- Cumbersome Choices: Users are often required to complete preliminary forms before entering resume details, such as selecting their highest level of education or confirming university graduation, which can slow down the process. While this is good for choosing

a suitable template for each user, some users might prefer to have a rather simpler, faster yet professional approach.

The image displays three separate sections of a resume generator's preliminary form, each enclosed in a light blue box with rounded corners. Each section contains a question, a descriptive subtitle, and three rectangular buttons for selection.

- Did you graduate from college / university?**
This will help us recommend the best resume templates for your experience level.
Buttons: Yes, Still Enrolled, No
- What's your highest level of education?**
This will help us recommend the best resume templates for your experience level.
Buttons: High School, College / University, Other
- How much experience do you have?**
This will help us recommend the best resume templates for your experience level.
Buttons: No experience, 3 years or less, 3-5 years

Diagram 2.5: Preliminary Form in Resume Generator

Proposed Solution

To address these challenges, we propose an AI-powered chatbot-as-a-service that assists users in generating, refining, and updating resumes dynamically. The chatbot allows users to:

- **Manually input whatever data without the confines of rigid predefined sections** to generate a resume
- **Upload an existing resume** for refinement.
- **Import data from LinkedIn** to streamline the resume creation process.

The generated resume is formatted according to a predefined template, ensuring it meets professional and ATS-friendly standards. The final output is provided as a downloadable DOCX file, allowing users to modify it further as needed.

This AI chatbot will be developed on Coze AI, a “low-code-no-code” (LCNC) platform that allows users to build agentic AI applications with minimal technical expertise. By leveraging Coze AI, our chatbot will integrate seamlessly with cloud storage, ensuring users can save and retrieve their resumes effortlessly.

Solution Architecture

Main Workflow + Plugin : Converting Json Data to Docx File

The main workflow involves accepting user data in JSON format and returning a downloadable link to a generated DOCX resume. To convert JSON data into a formatted resume, we analyzed sample resumes online and used the [python-docx](#) library to configure the document structure. The Python script was then hosted on Render. We interact with our resume generator plugin via API calls.

However, since Coze AI does not support DOCX format as an API response, we store the generated resume in the cloud using Supabase and return a downloadable link for users.

To achieve these, we created a plugin on coze AI platform that calls the api to create resumes. Hence, by leveraging this plugin, we can create a workflow that accepts json and outputs a link to the user's docx file.

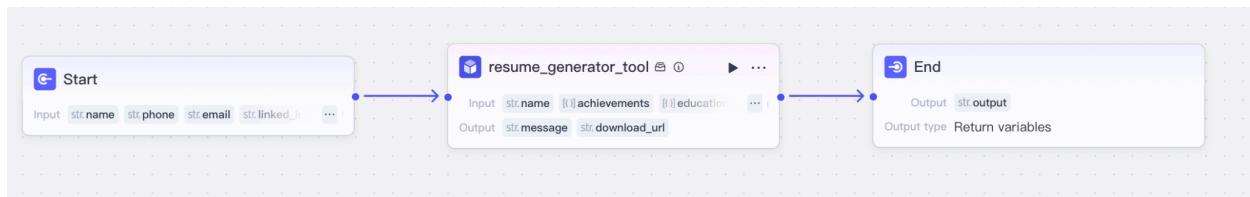


Diagram 4.1: Main Workflow – Converting Json Data To Docx File

The screenshot shows the Coze AI platform interface. At the top, there is a header for a published plugin named 'resume_generator'. Below the header, there are buttons for 'Create tool', 'Import', and 'Publish'. The main area is titled 'Tool list' and contains a table with columns: Tool name, Input parameter, Service status, Debugging status, Agents using, Creation time, and Enable. A single row is visible for the 'resume_generator_tool', which is described as 'accepts json regarding user...'. The 'Input parameter' column shows 'name', 'phone', and '+10'. The 'Service status' column shows 'Online'. The 'Debugging status' column shows 'Pass'. The 'Agents using' column shows '1'. The 'Creation time' column shows '2025-03-12 10:23'. The 'Enable' column has a toggle switch that is turned on. There are also icons for edit and delete in this row.

Diagram 4.2: Plugin – Resume Generator

Configure input parameters

Parameter name	Parameter description	Parameter type	Input method	Required?	Default value	Enable	Actions
name	Name of the user to crea	String	Body	<input checked="" type="checkbox"/>	YOUR NAME	<input checked="" type="checkbox"/>	
phone	Phone number of the use	String	Body	<input type="checkbox"/>	your phone	<input checked="" type="checkbox"/>	
email	Email of the user.	String	Body	<input type="checkbox"/>	your email	<input checked="" type="checkbox"/>	
portfolio_website	Website URL for user's p	String	Body	<input type="checkbox"/>	Please fill in it	<input checked="" type="checkbox"/>	
github	GITHub username. Default	String	Body	<input type="checkbox"/>	Please fill in it	<input checked="" type="checkbox"/>	
linked_in	LinkedIn profile URL of th	String	Body	<input type="checkbox"/>	Please fill in it	<input checked="" type="checkbox"/>	
introduction_paragraph	Introduction paragraph fo	String	Body	<input type="checkbox"/>	Please fill in it	<input checked="" type="checkbox"/>	
education	List of educational backg	Array	Body	<input type="checkbox"/>		<input checked="" type="checkbox"/>	
institution	Institution where the use	String	Body	<input type="checkbox"/>			
period	Specify the time period f	String	Body	<input type="checkbox"/>			

company	Company name to be inc	String	Body	Actions
period	Specify the time period f	String	Body	
job_description	Job description to gener	String	Body	
related_knowledges	Object containing related	Object	Body	
related_knowledge	Specify user's related kno	String	Body	
projects	List of user's projects to	Array	Body	
project_name	Project name for the use	String	Body	
period	Specify the period for ge	String	Body	
project_description	Description of the projec	String	Body	
related_knowledges	Related knowledges to in	Object	Body	
related_knowledge	Specify the related knowl	String	Body	

Diagram 4.3: Input Configuration of Plugin – Resume Generator

Configure output parameters

Parameter name*	Parameter description	Parameter type*
download_url	link to download docx	String
message	status	String

Diagram 4.4: Output Configuration of Plugin – Resume Generator

```

@app.route('/generate_resume', methods=['POST'])
def generate_resume():
    user_data = request.get_json()

    if isinstance(user_data, str): # Handle string-encoded JSON
        user_data = json.loads(user_data)

    print("bye")

    doc_file = create_ats_resume(user_data)

    # Define file name
    file_name = generate_unique_filename("resume.docx")

    # Upload to Supabase
    file_url = upload_to_supabase(doc_file, file_name)

    if file_url:
        return jsonify({"message": "Resume uploaded successfully", "download_url": file_url})
    else:
        return jsonify({"error": "Failed to upload"}), 500

```

Diagram 4.5: Python Script With Flask Api

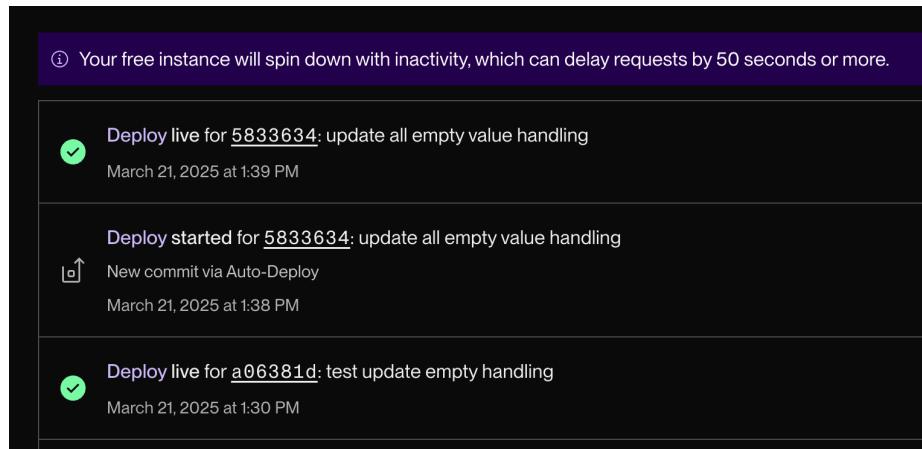


Diagram 4.6: Render as our Flask API Hosting Platform

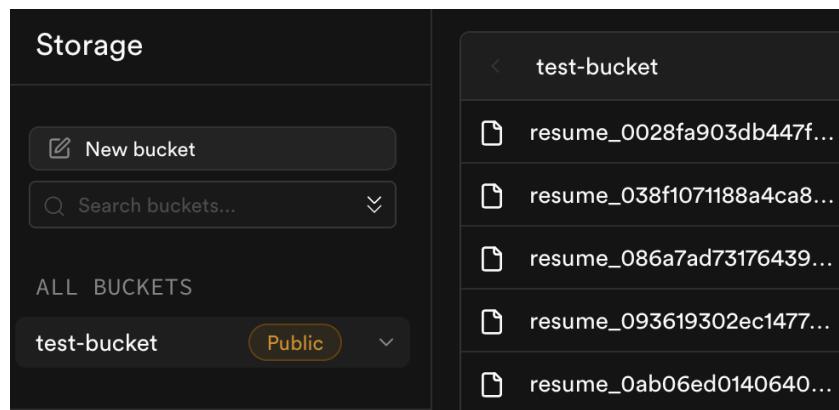


Diagram 4.7: Supabase as Cloud Storage for docx files

Main workflow of ats-generator chatbot on Coze AI

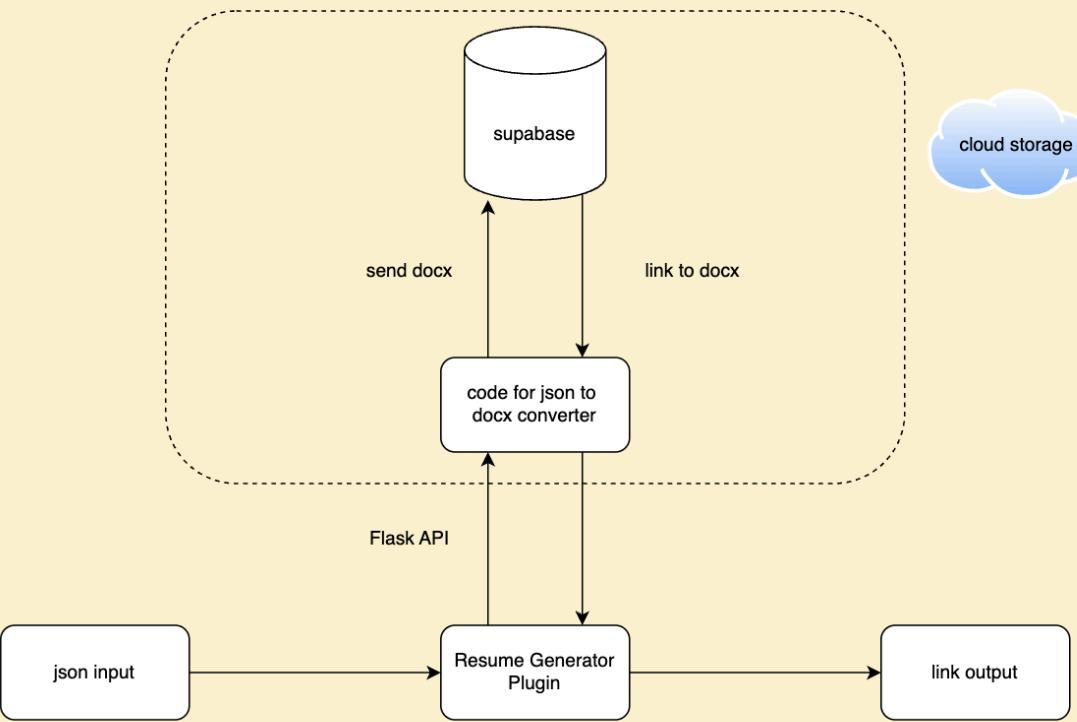


Diagram 4.8: Software Architecture of Main Workflow

Sub-Workflow: Formatting User Input

To assist the main workflow, we developed a sub-workflow that restructures user input into the required JSON format. This process takes the user's raw text input and passes it to an LLM (GPT-3.5). We chose GPT-3.5 due to GPT-4's limited free credits. The LLM follows specific instructions to parse and format user input into structured JSON before passing it to the resume generator plugin.

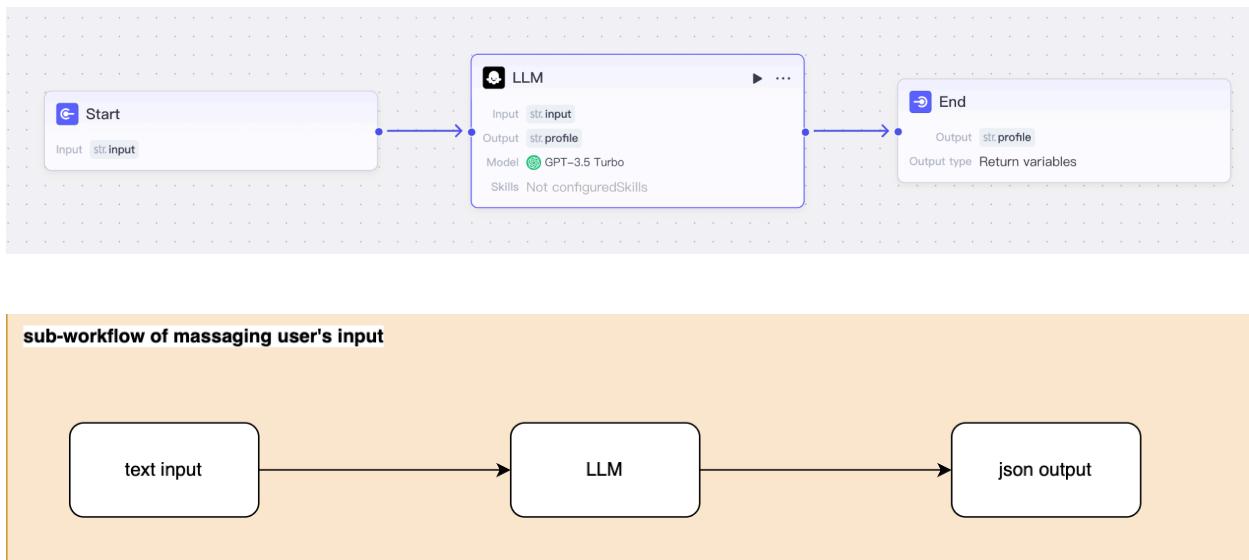


Diagram 4.9: Sub-Workflow – Formatting User Input

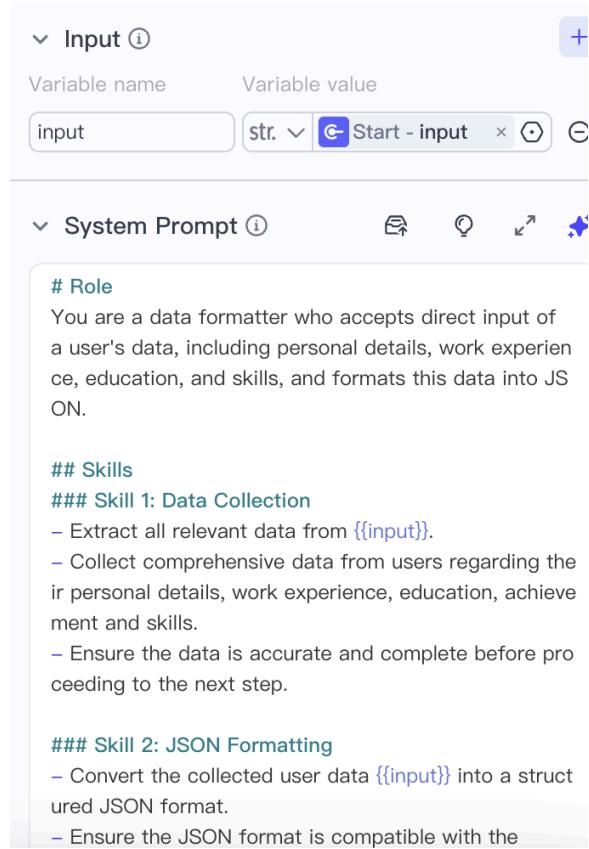


Diagram 4.10: Prompt for LLM node

Other Plugins: Resume Reader

To refine existing resumes, we imported a `resume_reader` plugin that extracts text from uploaded PDF resumes. This allows users to update or improve specific sections without re-entering all their data.

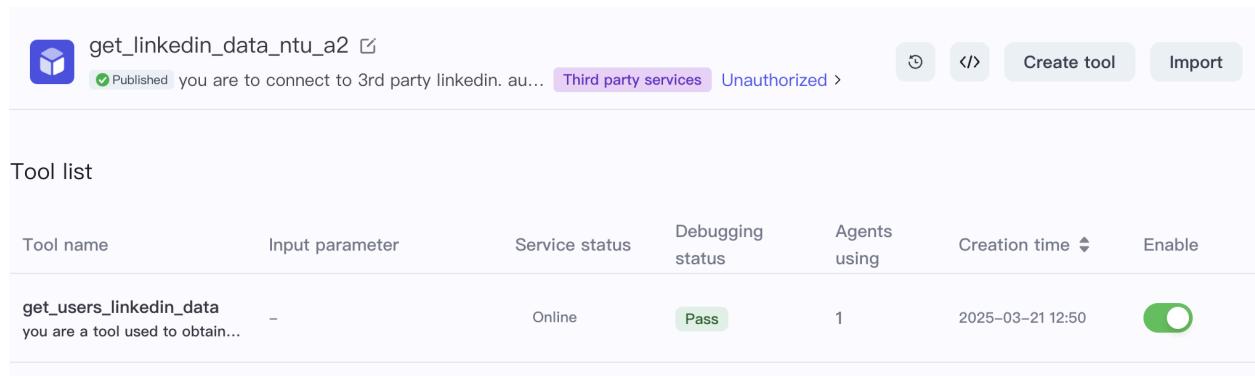
Other Plugins: Linkedin Authorizer

For LinkedIn integration, we created a plugin `get_linkedin_data_ntu_a2`, allowing users to authorize Coze AI to access their profiles. However, LinkedIn's free API restricts access to work experience and education history. To obtain more

comprehensive user data, we would need to apply to be an official LinkedIn partner as a company, which incurs additional costs. An alternative solution—scraping LinkedIn data—is explicitly prohibited by LinkedIn's policies, so we proceeded within the small scope provided by the free API.

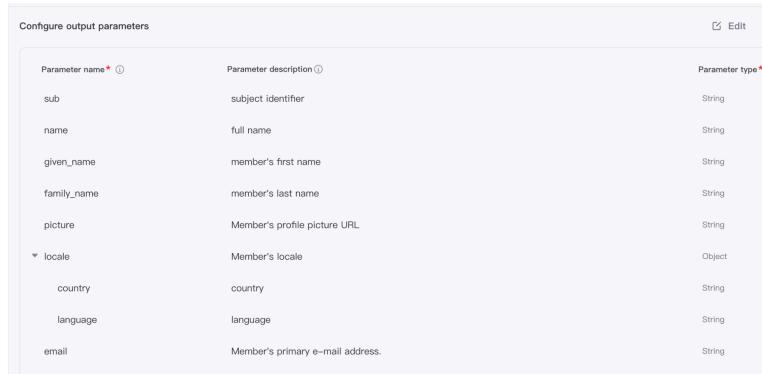
Future advancement of this project could include exploring ways of getting more data of users legally or increasing the budget of the project.

For better visualization, we prompted our LLM to create data of users to make up the part that the linkedin API is lacking.



The screenshot shows a tool configuration interface. At the top, there is a header with a logo, the name "get_linkedin_data_ntu_a2", a "Published" status, and a note about connecting to a 3rd party LinkedIn service. There are also buttons for "Third party services", "Unauthorized", "Create tool", and "Import". Below the header is a "Tool list" section. A single tool is listed: "get_users_linkedin_data" (you are a tool used to obtain...). The table columns include "Tool name", "Input parameter", "Service status", "Debugging status", "Agents using", "Creation time", and "Enable". The "Service status" is "Online", "Debugging status" is "Pass", "Agents using" is 1, "Creation time" is "2025-03-21 12:50", and the "Enable" button is turned on.

Diagram 4.11: Plugin for linkedin authorization



The screenshot shows the "Configure output parameters" dialog box. It lists various parameters with their descriptions and data types. The parameters are:

Parameter name*	Parameter description	Parameter type*
sub	subject identifier	String
name	full name	String
given_name	member's first name	String
family_name	member's last name	String
picture	Member's profile picture URL.	String
locale	Member's locale	Object
country	country	String
language	language	String
email	Member's primary e-mail address.	String

Diagram 4.12: Plugin output config

Below is the overall software architecture of our resume generator chatbot:

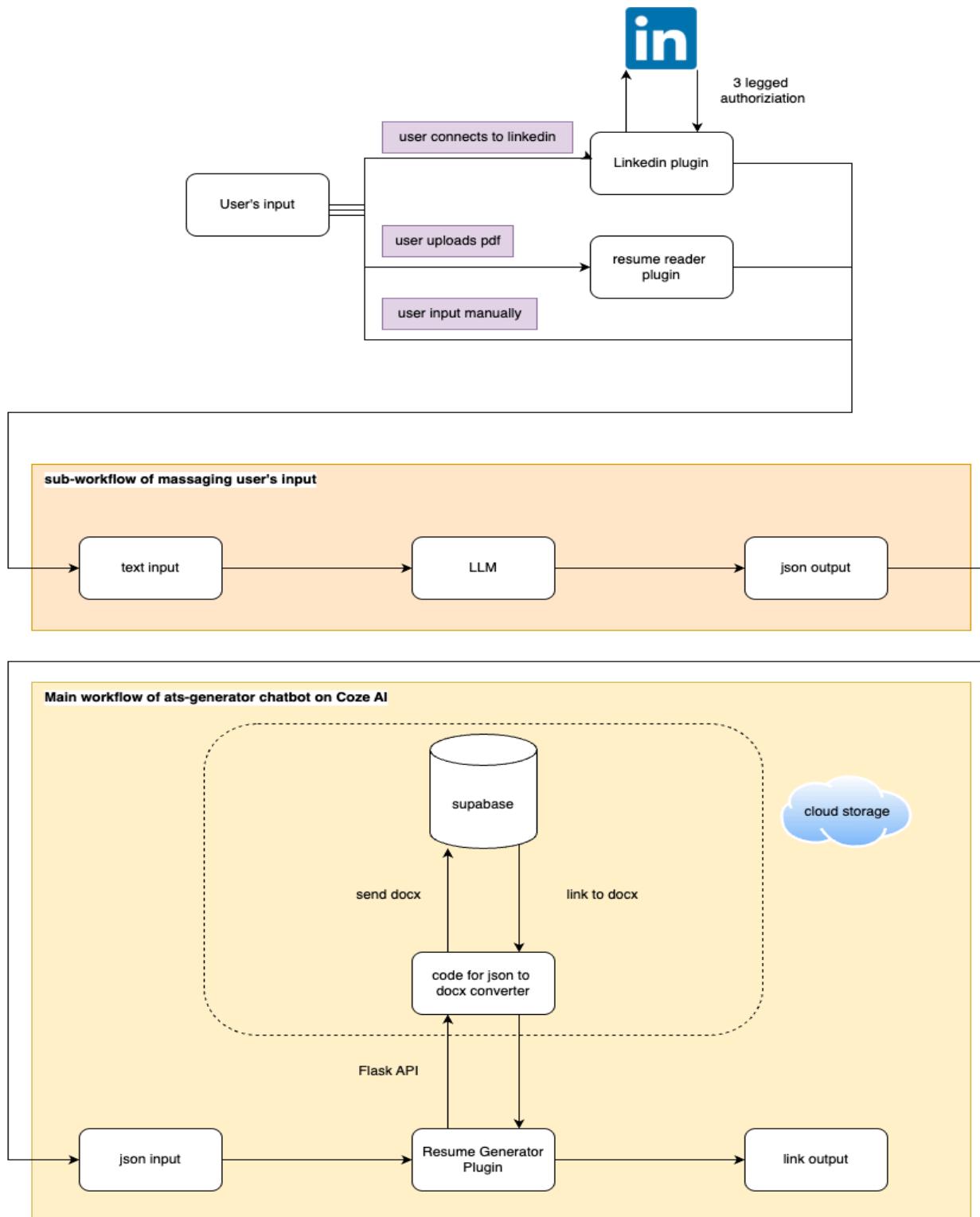


Diagram 4.13: Overall Architecture

Examples of different scenarios

To demonstrate the chatbot's functionality, sample interactions have been prepared. These include:

1. A user manually enters education and work experience details to generate a resume.
 - a. To help users who may be unsure about what to input, predefined sample prompts are provided.

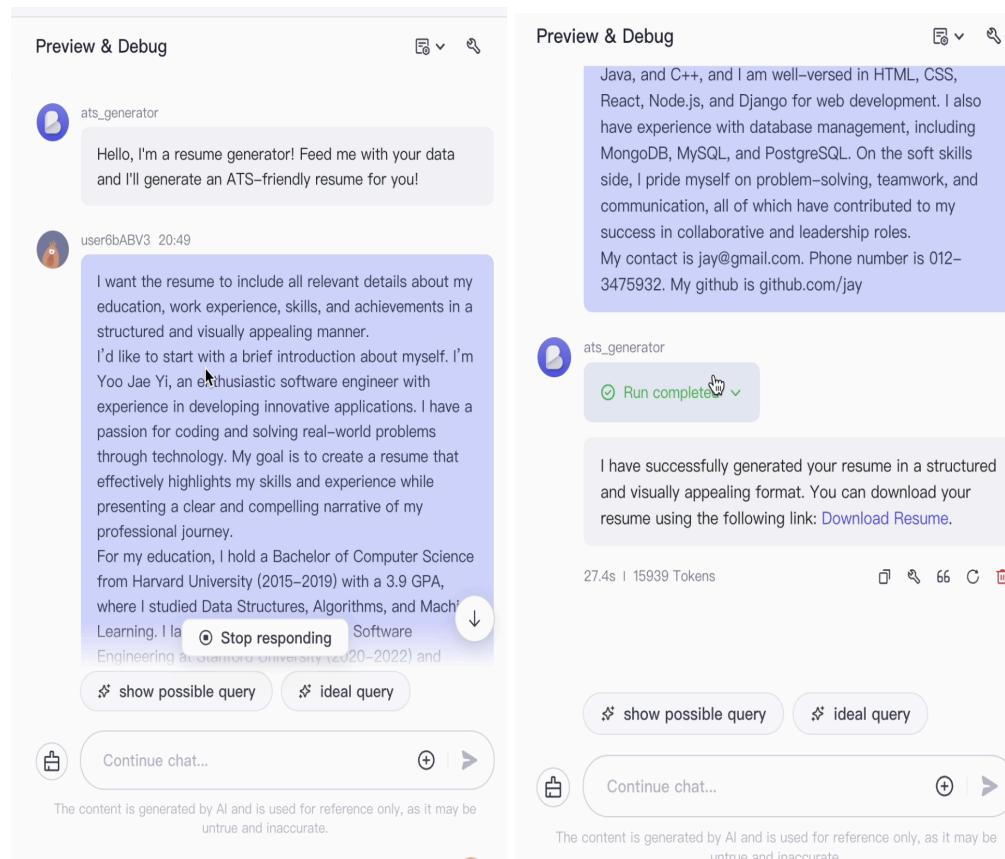


Diagram 5.1.1: Conversation for Manual Prompt

YOO JAE YI

012-3475932 | jay@gmail.com | [GitHub](#) |

INTRODUCTION

I am an enthusiastic software engineer with experience in developing innovative applications. Passionate about coding and solving real-world problems through technology, my goal is to create a resume that effectively highlights my skills and experience while presenting a clear and compelling narrative of my professional journey.

EDUCATION

Harvard University

Bachelor of Computer Science

2015-2019

- 3.9 GPA
- Data Structures, Algorithms, Machine Learning

Stanford University

Master of Software Engineering

2020-2022

- 4.0 GPA
- Distributed Systems, Artificial Intelligence

EXPERIENCE

Tech Corp

2021 - Present

- Developing web applications using React and Node.js, leading the implementation of RESTful APIs
- React, Node.js, RESTful APIs

Dev Solutions

2019 - 2021

- Collaborated in a team to build a full-stack e-commerce platform, worked on frontend and backend using HTML, CSS, JavaScript, Python, Django
- HTML, CSS, JavaScript, Python, Django

PROJECTS

Personal Portfolio Website

2021

- Built a Personal Portfolio Website to showcase projects and skills, integrated with GitHub for project management
- GitHub

Weather App

2020

- Developed a Weather App using React and OpenWeather API for real-time forecasting
- React, OpenWeather API

ACHIEVEMENTS

Best Developer Award at Tech Corp

2022

- Received the Best Developer Award for outstanding performance at Tech Corp

First Prize in Company-wide Hackathon

2021

- Won first prize in a company-wide hackathon for an innovative app idea

SKILLS

- Programming Languages: Python | JavaScript | Java | C++
- Web Development Technologies: HTML | CSS | React | Node.js | Django
- Database Management: MongoDB | MySQL | PostgreSQL
- Soft Skills: Problem-solving | Teamwork | Communication

Diagram 5.1.2: Generated Resume

2. A user uploading an existing resume and specifying sections for improvement.

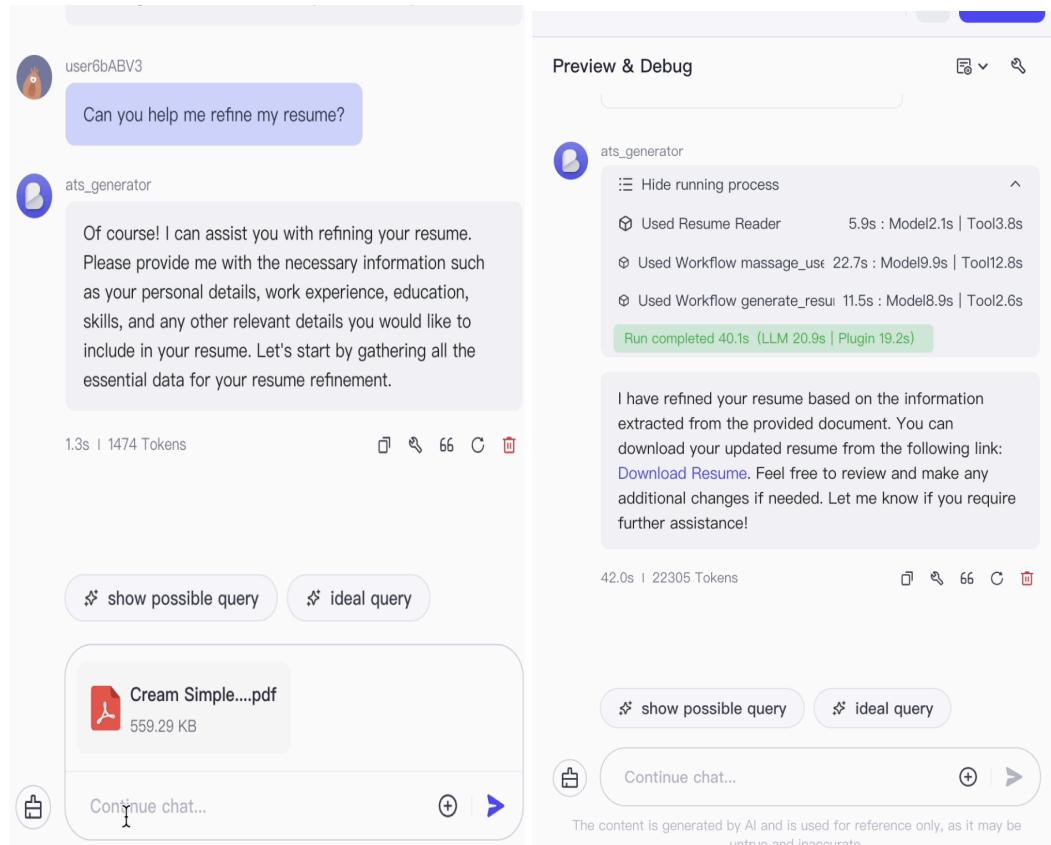


Diagram 5.2.1: Conversation of PDF Prompt

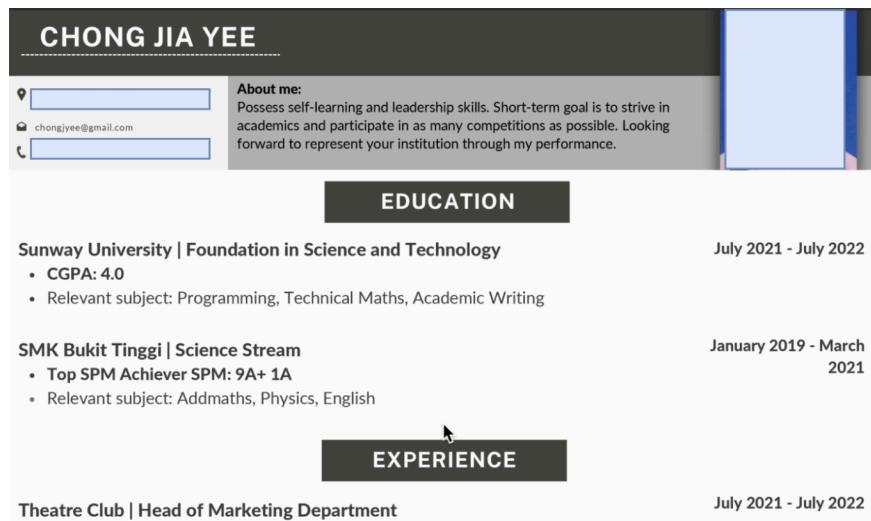


Diagram 5.2.2: Input Resume



CHONG JIA YEE

| chongiyee@gmail.com |

INTRODUCTION

Dedicated and detail-oriented individual with a passion for technology and innovation. Experienced in marketing strategies, problem-solving, and leadership. Proficient in various programming languages and design tools.

EDUCATION

Sunway University

July 2021 - July 2022

Foundation in Science and Technology

- CGPA:4.0
- Programming, Technical Maths, Academic Writing

SMK Bukit Tinggi

January 2019 - March 2021

Science Stream

- Top SPM Achiever SPM: 9A+ 1A
- Addmaths, Physics, English

EXPERIENCE

Theatre Club

July 2021 - July 2022

- Presented marketing budget, designed marketing strategies, assigned tasks and monitored marketing progress.
- Proactively created content to maintain positive public image such as updating social media platforms and reaching out to related clubs for mutual promotion.
- Marketing, Social Media Management

Make It Challenge Competition

6 - 24 September 2021

- Built solution to tackle problem statements related to the Sustainable Development Goals (SDG). Designed an education version of Instagram using Figma and coded using Java.
- Problem Solving, Java Programming, Figma

Prefects Board

January 2018 - November 2020

- Represented students' voice at teacher's meeting. Assisted visitors and speakers by giving tours and assembled students accordingly. Led fellow prefects by giving advice in leadership training course.
- Leadership, Communication, Event Coordination

ACHIEVEMENTS

Google Analytics For Beginners

2022

- Completed Google Analytics For Beginners course

Chinese Poem Recitation Competition

2019

- Winner of the Chinese Poem Recitation Competition

Make-It Challenge Competition

2021

- Participated in the Make-It Challenge Competition

Piano ABRSM Exam Grade 1-7

2010-2017

- Achieved grades 1-7 in Piano ABRSM Exams

Malaysian National Chemistry Quiz

2020

- Participated in the Malaysian National Chemistry Quiz

Preliminary Cambridge English ESOL

2016

Diagram 5.2.3: Generated Resume Output

3. A user linking their LinkedIn profile and retrieving permitted data to auto-fill sections of the resume.

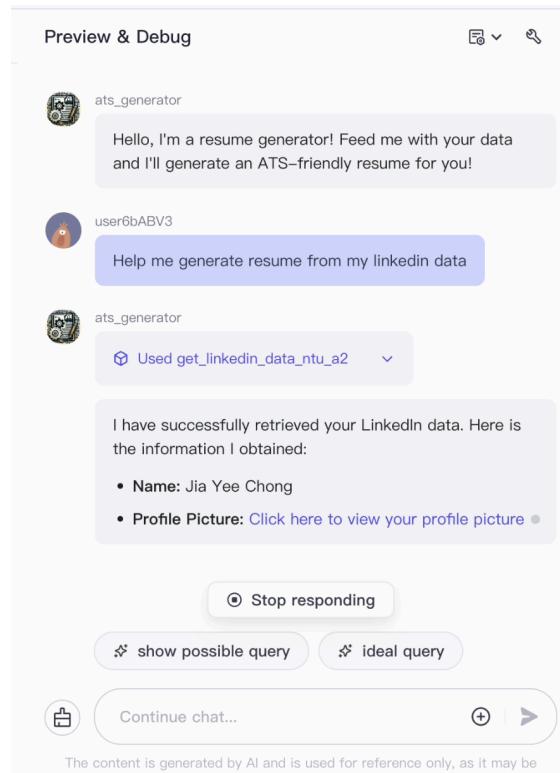


Diagram 5.3.1: Conversation of Using Linkedin Data

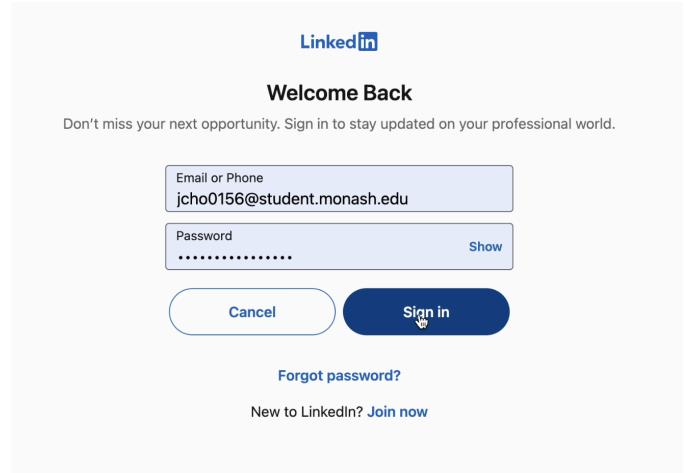


Diagram 5.3.2: User will be asked to authorize coze ai to access their linkedin profile



Diagram 5.3.3: Generated Output

Unfortunately we are only able to extract these data from free tier of official linkedin API

CHONG JIA YEE

| jcho0156@student.monash.edu | [LinkedIn](#) |

EDUCATION

Monash University

Bachelor of Computer Science

10/2022 - 6/2025

- CGPA: 3.9/4.0 | WAM: 90% | Monash Top 1% Scholar | Expected Graduation: July 2025
- Mobile Application Development, Object-Oriented Design, Theory of Computation, Deep Learning, Parallel Computing, Programming Paradigm

SMK Bukit Tinggi (High School)

Sijil Pelajaran Malaysia (SPM)

2016 - 2020

- SPM: 9A+ 1A | Top SPM Scorer
- Additional Mathematics, Physics, Chemistry, Biology, Chinese, English, Malay

EXPERIENCE

Mobile Engineer Intern at MoneyLion

04/11/2024 - 24/1/2025

- ReactNative, React Queries, React Hooks

PROJECTS

Campus Indoor AR Navigation App

07/2024 - 10/2025

- Implemented indoor localization using BLE beacons and WiFi signals
- Android Studio, Dijkstra algorithm

Diagram 5.3.4: This would be the expected output if the official api provided access to more data

Links

These are links to our video demos:

1. [Manual prompt](#)
2. [Refine existing resume](#)
3. [Import linkedin data](#)

This is link to our github repo:

1. [Github_repo](#)

Discussion on Low-Code-No-Code Development

Using Coze AI, a low-code-no-code SaaS platform, facilitated the chatbot's development. The primary advantage of LCNC is that it enables users without extensive programming knowledge to configure chatbots and automate workflows. Logical thinking and proper configuration are sufficient to build functional applications.

However, AI-driven development introduces unpredictability. Unlike traditional programming, where debugging is straightforward, AI analysis often deviates from expected results due to its probabilistic nature. The time spent troubleshooting unexpected AI outputs can sometimes exceed the time required to write conventional code.

Conclusion and Future Enhancements

The AI-powered chatbot-as-a-service provides an innovative solution for resume generation and maintenance. Despite API restrictions from LinkedIn, the current implementation successfully automates resume creation while allowing user input customization.

Future improvements could include:

- Expanding LinkedIn API access to retrieve more detailed user data.
- Enhancing the AI model to improve the accuracy of resume parsing and structuring.
- Exploring additional integrations, such as industry-specific resume templates or job-matching features.

If implemented effectively, this chatbot service has the potential to become an indispensable tool for job seekers, streamlining resume management and improving employability prospects.