



Faculty of Computer Science and Information Technology

A Real-time Interactive Event Management Chatbot

FINAL REPORT

ENRIQUE SHANNA ANAK UJAI

Bachelor of Computer Science with Honours (Software Engineering)

2023

A REAL-TIME INTERACTIVE EVENT MANAGEMENT CHATBOT

ENRIQUE SHANNA ANAK UJAI

This project is submitted in partial fulfilment of the requirements for the degree of Bachelor of
Computer Science and Information Technology

Faculty of Computer Science and Information Technology

DECLARATION

I hereby declare that this project is my original work. I have not copied from any other student's work or from any other sources except where due reference or acknowledgement is not made explicitly in the text, nor has any part had been written for me by another person.



24/1/2023

(Enrique Shanna Anak Ujai)

Date

ACKNOWLEDGEMENT

I would like to express my sincere appreciation to my Final Year Project supervisor, Dr. Mohamad Nazim Jambli, for his constant guidance and support throughout the project. He was an excellent mentor, providing valuable insights and feedback that allowed me to develop my project to its fullest potential. His expertise and guidance played a crucial role in my successful completion of the project. I am also very grateful to the final year project coordinator, Professor Dr. Wang Yin Chai, for his guidance and support regarding the formal documentation of my final year project report. His expert advice helped me ensure that the report met the necessary standards and was of high quality. I would also like to extend my gratitude to my family and friends for their support and encouragement throughout this project. Their support and confidence in me served as a constant motivation and without them, I would not have been able to complete my project successfully.

ABSTRACT

Chatbots have proven useful in many other contexts to automate tasks and improve the user experience, such as automated customer services, education, and e-commerce. Moreover, existing reports emphasize that chatbot design will become a key ability in IT hires soon.

This report is focused on the development of a real-time interactive event management chatbot assist students acquire information for upcoming events of their interests within or outside university. Students can put their technical knowledge to the test by preparing and presenting events or activities or, purchase merchandises or miscellaneous in UNIMAS based on their own interests and the system will automatically matches to other users with similar views or opinions. The chatbot system can empower students to put their technical knowledge to the test by preparing and presenting events or activities and allows to purchase merchandises or miscellaneous in UNIMAS based on their own interests and matches to other users with similar views or opinions. The proposed system can also keep track of events or activities within local institutions or communities within local region. This fully interactive recommendation chatbot system allows students to engage in a unique blend of social interaction or events and academic discussion in their local geographic region or within UNIMAS that matches user's preferences.

Table of Contents

Catalog

ACKNOWLEDGEMENT	ii
ABSTRACT	iii
Table of Contents	4
List of Figures	5
CHAPTER 1: INTRODUCTION	7
1.1 Project Title	7
1.2 Introduction	7
1.3 Problem Statement	7
1.4 Scope	8
1.5 Aims and Objectives	9
1.6 Brief Methodology	9
1.7 Significance of Project	10
1.8 Project Schedule	10
1.9 Expected outcomes	12
1.10 Summary	12
CHAPTER 2: LITERATURE REVIEW	12
2.1 Introduction	12
2.2 Overview of Objectives	13
2.3 Review on Similar Existing Web-Application	13
2.4 Review on Tools and Technology	23
2.5 Summary	26
CHAPTER 3: REQUIREMENT ANALYSIS AND DESIGN	27
3.1. Overview	27
3.2. Waterfall Development Process	27
3.3 Implementation	35
3.4 Verification or Testing	36
3.5 Summary	36
CHAPTER 4: IMPLEMENTATION	38
4.1 Introduction	38
4.2 System Configurations	38
4.3 Deployment	55
4.4 Web Development	56
4.5 Proposed System	62
4.6 Summary	64
CHAPTER 5: TESTING	65
5.1 Introduction	65
5.2 Software Testing	65
5.3 Non-Functional Testing	72
5.4 Summary	82
CHAPTER 6: CONCLUSION AND FUTURE WORKS	82
6.1 Introduction	82
6.2 Achievements	82
6.3 Limitations and Constraints	83
6.4 Future Works	83
6.5 Summary	84
References	85

List of Figures

Figure 1 :FYP1 Gantt Chart	11
Figure 2 : Gantt Chart FYP2	11
Figure 3 : <i>ManyChat Dashboard Goals and Strategies Tab</i>	14
Figure 4 :ManyChat Dashboard Bot Overview Tab	15
Figure 5 : ManyChat Dashboard Metrics Tab	16
<i>Figure 6</i> : ManyChat Audience Dashboard	17
<i>Figure 7</i> : ManyChat Audience Dashboard Collaborative Filtering	17
Figure 8 : ManyChat Growth Tool Dashboard	18
Figure 9 : ManyChat Import Audience	19
Figure 10 : ManyChat Export Audience	19
<i>Figure 11</i> : ManyChat Events Dashboard	20
Figure 12 :Chatbot.ai Dashboard	21
Figure 13 : Chatbot.ai Filter	22
<i>Figure 14</i> : Sublime	23
<i>Figure 15</i> : Microsoft Bot Framework	24
Figure 16 : XAMPP	25
Figure 17 : Localhost	25
Figure 18 : Personal Laptop	26
Figure 19 : Class Diagram	32
Figure 20 : Sequence Diagram	34
<i>Figure 21</i> : <i>Screenshot of Microsoft Azure for Students Website</i>	39
Figure 22 : Screenshot of Microsoft Azure for Students Registration Page	39
Figure 23 : Screenshot of Microsoft Azure for Students Page	40
Figure 24 : Create a Resource (Resource Group)	41
Figure 25 : Create a Resource (Resource Group) Details Required	42
Figure 26 : Successful Creation of Resource Group	42
Figure 27 : Creating a Web App	43
Figure 28 : Fill in details required and choose your runtime stack that you wanted your bot built on.	44
Figure 29 : Successful in creating Web App	45
Figure 30 : Creating an Azure Bot	45
Figure 31 : Fill in Details Required while Creating an Azure Bot	46

Figure 32 : : Success Building an Azure Bot	47
Figure 33 : Create a Language Azure Services	47
Figure 34 :Language Services Portal	48
Figure 35 : Language Services Features	49
Figure 36 : Language Services Features Details Required	50
Figure 37 : Successfully creating Language Services	50
Figure 38 : Successfully creating all necessary components	51
Figure 39 : Test Bot in Web Chat	52
Figure 40 : Knowledge Base Custom QnA	53
Figure 41 : Get Embed Codes	54
Figure 42 : Embed Codes	54
Figure 43 : Secret Key of Embed Code	55
Figure 44 : CSS Code of style.css	57
Figure 45 : HTML Code of index.php	58
Figure 46 : JavaScript Code of script.js	59
Figure 47 : PHP Code of config.php	60
Figure 48 : SQL Code	61
Figure 49 : Proposed System UI	62
Figure 50 : Proposed Chatbot UI	62
Figure 51 : Proposed Chatbot UI	63
Figure 52 : Admin UI	64
Figure 53 : IT Background Survey	74
Figure 54 : Q1 Survey	75
Figure 55 : Q2 Survey	75
Figure 56 : Q3 Survey	76
Figure 57 : Q4 Survey	77
Figure 58 : Q5 Survey	78
Figure 59 : Q6 Survey	79
Figure 60 : Q7 Survey	80
Figure 61 : Q8 Survey	80
Figure 62 : Q9 Survey	80

CHAPTER 1: INTRODUCTION

1.1 Project Title

A Real-time Interactive Event Management Chatbot

1.2 Introduction

This recommendation chatbot system assist students acquire information for upcoming events of their interests within or outside university. Students can put their technical knowledge to the test by preparing and presenting events or activities or, purchase merchandises or miscellaneous in UNIMAS based on their own interests and the system will automatically matches to other users with similar views or opinions. Chatbots have proven useful in many other contexts to automate tasks and improve the user experience, such as automated customer services, education, and e-commerce. Moreover, existing reports emphasize that chatbot design will become a key ability in IT hires soon. [1]. A In addition, through this fully interactive recommendation chatbot system participation allows students to engage in a unique blend of social interaction or events and academic discussion in their local geographic region or within UNIMAS that matches user's preferences. One of the common recommender systems techniques is collaborative filtering (CF). The basic idea of CF is the extraction of information about the old behaviour or opinions to user that exist in society, any elements were expected that the current user of the system are likely to have willingness to be selected, or similar to his taste [2].

1.3 Problem Statement

There is no current recommendation chatbot system for students to acquire information for upcoming events within or outside university other than WhatsApp which can be hard to track as too many messages coming in. In addition, students tend to overlook events of their interests within university or outside institution on social struggles to acquire information of events, activities, within student community and to local communities outside the institution of their likings. There is no active current recommendation system that sells UNIMAS merchandises or miscellaneous of their likings and provides privilege such as discount.

1.4 Scope

Table 1.0: Scope

Target User	UNIMAS Students, lecturers and staffs.
Programming languages	<ol style="list-style-type: none">1. Java2. PHP3. MySQL4. Python5. C++
Software System Requirement	<p>Software</p> <ol style="list-style-type: none">1. Sublime2. XAMPP3. LocalHost <p>Hardware</p> <ol style="list-style-type: none">1. Laptop2. PC3. Tablet
System Features	<ol style="list-style-type: none">1. User can upload their event's poster or information through Chatbot.2. Chatbot can redirect user to find clubs or activities, or events based on their own interests.3. Student can see list of posters or events posted based on their likings.4. Chatbot can redirect students to see merchandises that they might like based on their interests.5. Chatbot can filter events that matches

	<p>user's preferences</p> <ol style="list-style-type: none"> 6. Student can interact with the Chatbot. 7. Chatbot able to acquire real-time information of event's outside of university.
--	---

1.5 Aims and Objectives

- This recommendation chatbot system for UNIMAS students to acquire information for upcoming events of their interests within or outside university.
- Students no need to rely on WhatsApp to acquire information of upcoming events since it can be difficult because of messages.
- To develop and design a system that student can have access to e-commerce of UNIMAS miscellaneous of their own likings.

1.6 Brief Methodology

The methodology that being chose for this project is Waterfall Method. There are five phases in this methodology which are:

1.6.1 Requirements Phase

- At this phase, every possible requirement for the system that will be implemented is assessed and defined in a requirement specification document.

1.6.2 Design Phase

- Throughout this phase, the required requirements from the first phase are assessed. The system design is established once the data is analyzed. This system design help in deciding on the overall system design as well as the hardware and software system requirements.

1.6.3 Implementation Phase

- Unit testing and system testing are implemented to ensure a reliable system is produced.

1.6.4 Verification Phase

- After testing each unit developed throughout the implementation, phase, the entire system is unified. After integration, the full system is examined for faults and failures.

1.6.5 Maintenance Phase

- During this stage, small changes are made to improve the system developed during implementation and maintenance tasks is performed. Bugs or errors are identified that may have been overlooked during the testing phase.

1.7 Significance of Project

The significance of this project is to provide a real-time interactive event management Chatbot system for the UNIMAS students acquire information for upcoming events within or outside university based on their interests. The system can facilitate and ease those process since it can be accessed if there is a stable internet connection and hardware devices that connected to it. Hence, this proposed system can solve the problems and issues such as student struggles to acquire information of events, activities, within student community and to local communities outside the institution of their preferences.

1.8 Project Schedule

The project is expected to be completed within two semesters following the course Final Year Project I and Final Year Project II.

Table 2.0: Schedule of each task for Final Year Project I

No.	Task	Start	Finish	Days to Complete	Predecessor
1	Brief Proposal	10-10-22	04-11-2022	26	
2	Full Proposal	05-11-22	15-11-2022	11	1
3	Chapter 1 : Introduction	16-11-22	21-11-2022	6	2
4	Chapter 2 : Literature Review	22-11-22	09-12-2022	18	3
5	Chapter 3 : Methodology	10-12-22	30-12-2022	21	4
6	FYP 1: Report	02-01-23	11-01-2023	10	5

FYP 1 - A Real-Time Interactive Event Management Chatbot

Gantt Chart

PROCESS	2022				2023	
	Sep	Oct	Nov	Dec	Jan	Feb
Brief Proposal		█				
Full Proposal			█			
Chapter 1: Introduction			█			
Chapter 2: Literature Review				█		
Chapter 3: Methodology				█		
FYP 1: Report				█		

Figure 1:FYP1 Gantt Chart

FYP 2 - A Real-Time Interactive Event Management Chatbot

Gantt Chart

PROCESS	2023					
	Jan	Feb	Mar	Apr	May	Jun
Chapter 4: Implementation	█	█	█	█		
Chapter 5: Testing		█	█	█	█	
Chapter 6: Conclusion and Future Works					█	█
FYP 2: Final Report				█	█	█

Figure 2: Gantt Chart FYP2

1.9 Expected outcomes

The expected outcome of this project is fully interactive recommendation system that serves as a collaborative filtering application that assist students in findings information of forthcoming events based on their interests and it can be accessed via personal computer, laptop, and smartphones as it will bring benefit to all the students and lecturers in UNIMAS. Besides, the system also can be used in a long-term period and have user-friendly interfaces. The students can easily acquire information of events, activities, within student community and to local communities outside the institution based on their own interests and likings that matches user's preferences. Recommender systems are considered as devices and programming engineering organization that is utilized for generating advantages suggestion to the users, as well as to help them in the procedure of decision making. [3]

1.10 Summary

To sum up, chapter 1 includes a clear understanding of this project. This is followed by introduction of the project, problem statement, scope, objectives, methodology used, significance of project and project schedule. The chapter concludes with the expected outcome of this project.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

A literature review surveys books, scholarly articles, and any other sources relevant to a particular issue, area of research, or theory, and by so doing, provides a description, summary, and critical evaluation of these works in relation to the research problem being investigated. Literature reviews are designed to provide an overview of sources you have explored while researching a particular topic and to demonstrate to your readers how your research fits within a larger field of study [4].

Chapter 2 of the FYP 1 paper will provide a comparison of A Real-time Interactive Event Management Chatbot with comparable existing systems, and to

learn more about the similar system that currently exist, and it will assist the development phase of the FYP2 project later. Along with analysing the current existing system which is to carefully consider the technology and tools to create the system, will be carry out later. This assists in directing us during the FYP2 development phase. This includes elements such as the required technology and software to create a suitable system of this project with proportionate functionality as current existing system.

2.2 Overview of Objectives

The objectives of a real-time interactive event management chatbot are to provide users with a more efficient way to find clubs of their interests and users can promote their events to students who have the same interests as well. The chatbot can help to streamline the planning and execution of events by providing automated reminders, notifications, and updates, as well as personalized responses to user inquiries and finding people who have the same interests. Additionally, the chatbot can provide personalized event recommendations and suggestions to users with the same likings and provide real-time analytics of event data. By utilizing a chatbot for event management, event organizers can save time, money, and resources, while providing users with a more seamless and enjoyable experience.

The purpose of this chapter is to carefully analyze current existing system comparable to this project which called A Real-time Interactive Event Management Chatbot. The second objective is to develop the Real-time Interactive Event Management tool itself, which will be done in Final Year Project 2, and the third objective is to evaluate the efficacy of the chatbot tool that has been created, assessing its functionality, and reviewing its performance.

2.3 Review on Similar Existing Web-Application

2.3.1 ManyChat by Mikita Mikado

One example of a chatbot that can be used to manage events is ManyChat. ManyChat is a chatbot platform that allows users to create and manage chatbots for their websites, Facebook pages and mobile apps. It provides features such as automated messages, conversations with customers, and the ability to create custom bot flows. It also has a

mobile app that allows users to manage their chatbots on-the-go and to stay connected with their customers. Additionally, ManyChat provides an API that allows users to integrate their chatbots into their own websites. This chatbot can be used to manage events by providing automated reminders, notifications, and updates; helping to process payments; providing personalized event recommendations and suggestions; and providing real-time analytics of event data. [5]

Dashboard in ManyChat allows businesses to view the status and activity of their subscribers. This includes the total number of subscribers, new subscribers, and lost subscribers. Additionally, businesses can filter subscribers by their status (Active, Subscribed, or Unsubscribed) and segment them into lists. The dashboard also displays the open and click rates of messages sent to subscribers. And now we are in ManyChat Dashboard Home first tab which is Goals and Strategies. The Goals & Strategies dashboard in ManyChat allows businesses to track their customer service and marketing performance. This includes tracking metrics such as open rate, click rate, and engagement rate. Additionally, businesses can set objectives and create campaigns to reach their goals. They can also customize messages and create automated workflows to improve customer experience.

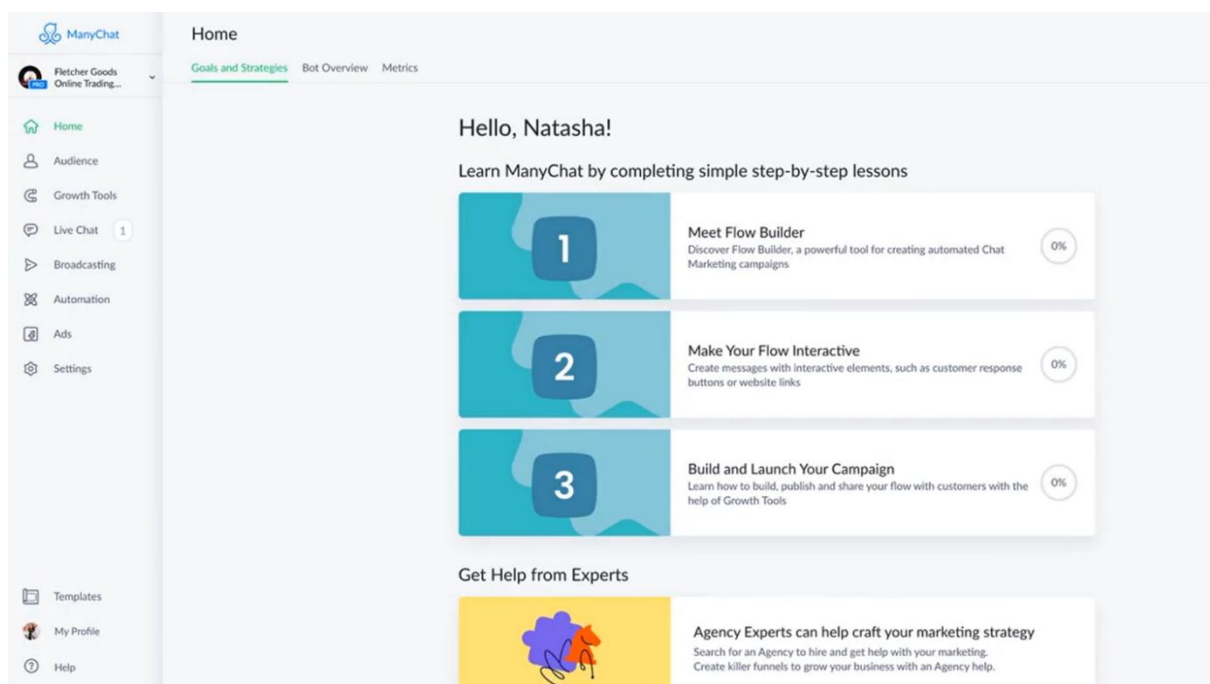


Figure 3: ManyChat Dashboard Goals and Strategies Tab

The Bot Overview dashboard in ManyChat allows businesses to view and manage their automated customer service and marketing bots. This includes viewing the total number of bots, the number of active bots, and the number of conversations. Additionally, businesses can customize their bots by setting messages and automated workflows. They can also track the performance of their bots by viewing metrics such as open rate, click rate, and engagement rate.

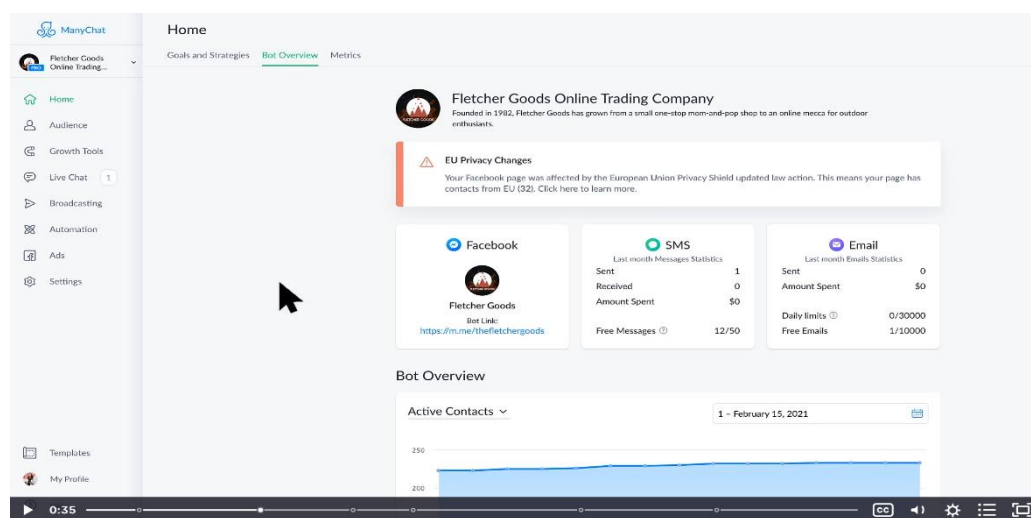


Figure 4: ManyChat Dashboard Bot Overview Tab

The Metrics dashboard in ManyChat allows businesses to view their performance metrics. This includes open rate, click rate, and engagement rate. Additionally, businesses can view the total number of messages sent, the total number of subscribers, and the total number of conversations. The dashboard also displays the total number of new subscribers and the total number of lost subscribers.

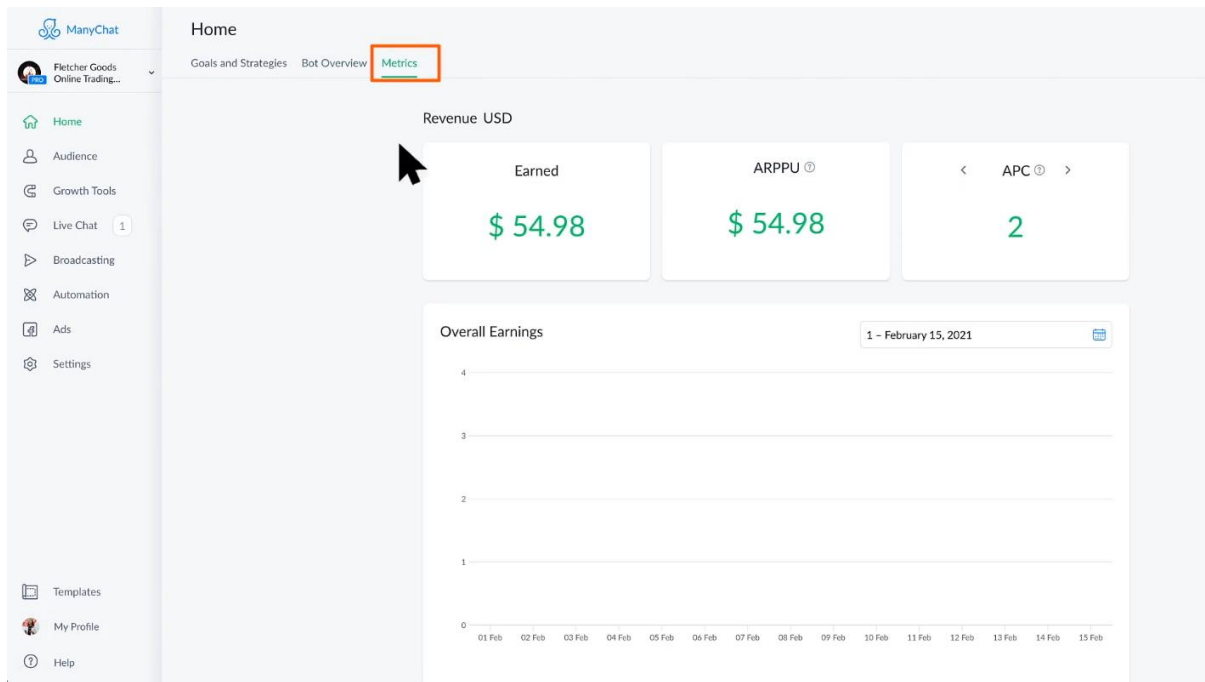


Figure 5: ManyChat Dashboard Metrics Tab

Manychat's Audience Dashboard is an online tool that provides users with an overview of their audience's growth and engagement. It allows users to track their subscribers, profile data, conversations, and more. The dashboard also provides insights into audience engagement, such as which messages are the most effective, what content is resonating with subscribers, and which features are driving the most conversions. Additionally, users can customize their dashboard with custom widgets, such as the Live Chat Widget, to gain even more in-depth insights into their audience. The Audience Dashboard also allows users to filter and search their data to better understand their audience. This feature allows users to filter their data by specific criteria, such as demographics, locations, or interests, so they can better target their messages to their desired audiences.

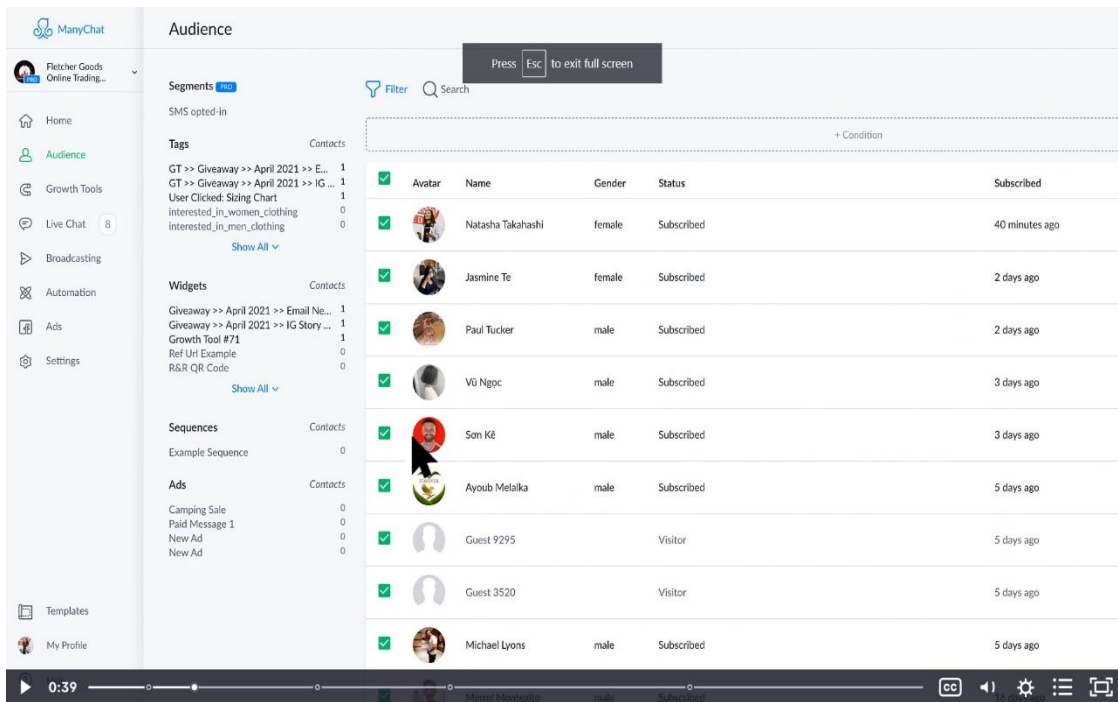


Figure 6: ManyChat Audience Dashboard

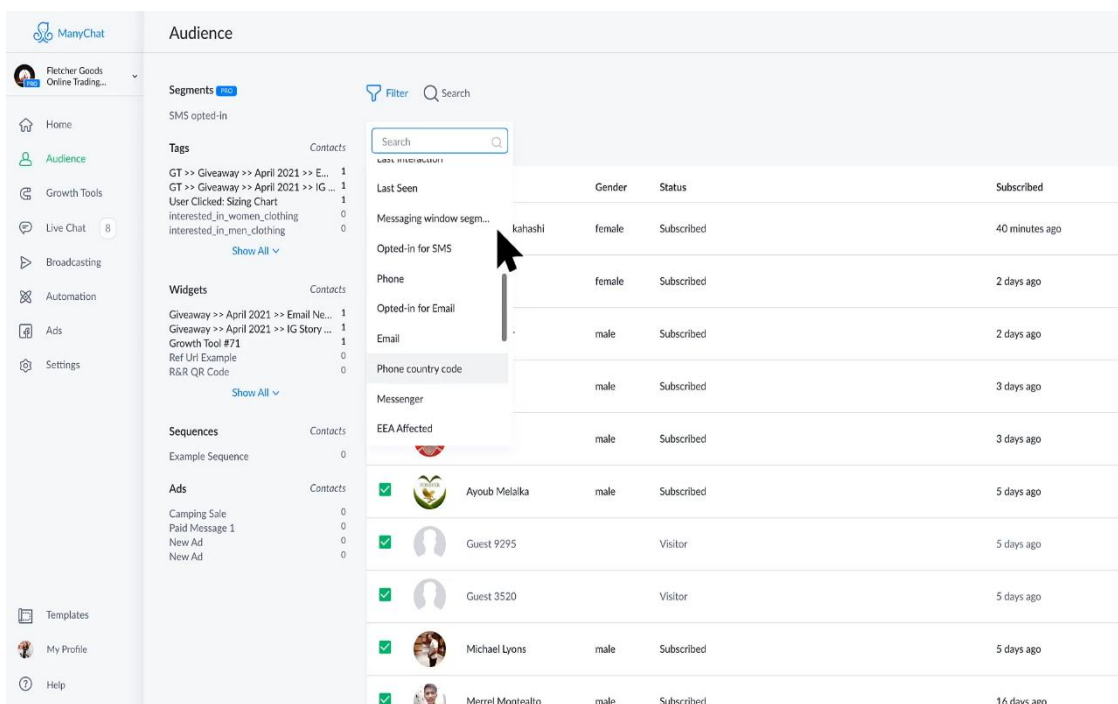
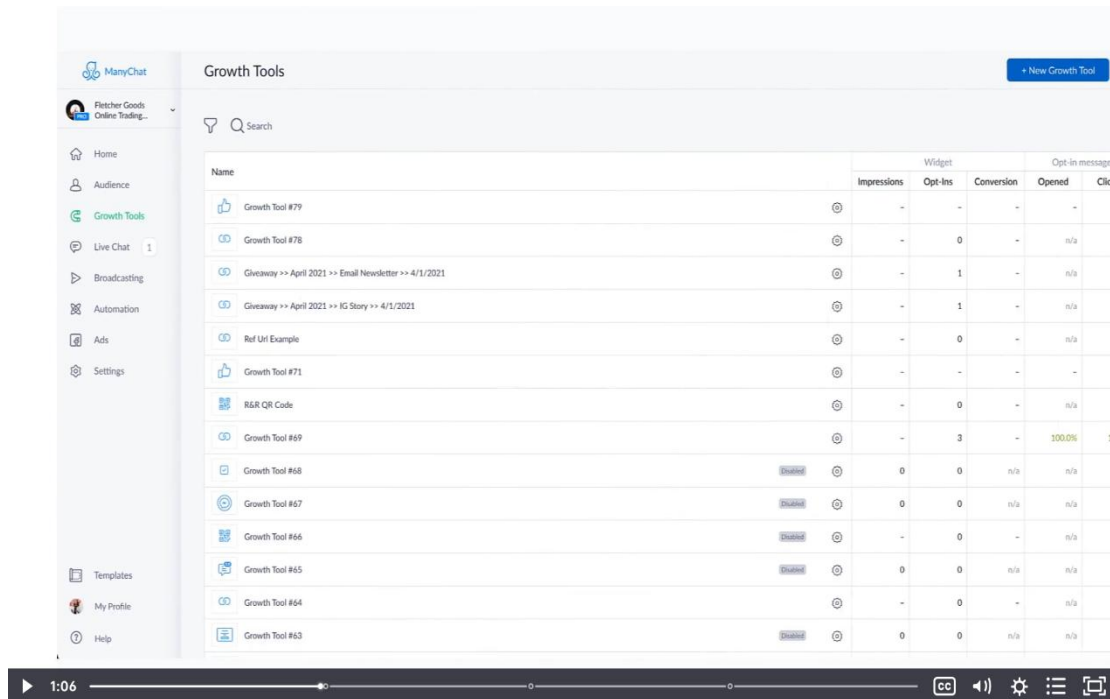


Figure 7: ManyChat Audience Dashboard Collaborative Filtering

Manychat's Growth Tool is an automated tool designed to help users grow their audiences quickly and efficiently. It enables users to easily create custom URLs, QR codes, and referral codes to share with their audience, as well as targeting options to ensure they reach the right people. Additionally, the Growth Tool provides detailed analytics on the

performance of campaigns, allowing users to optimize their strategies and maximize their reach.



The screenshot displays the ManyChat Growth Tools dashboard. On the left is a navigation sidebar with options like Home, Audience, Growth Tools, Live Chat, Broadcasting, Automation, Ads, and Settings. The main area shows a table of growth tools with columns for Name, Impressions, Opt-ins, Conversion, and Opened. A search bar and a '+ New Growth Tool' button are at the top right.

Name	Impressions	Widget		Opt-in message	
		Opt-ins	Conversion	Opened	Click
Growth Tool #79	-	-	-	-	-
Growth Tool #78	-	0	-	n/a	-
Giveaway >> April 2021 >> Email Newsletter >> 4/1/2021	-	1	-	n/a	-
Giveaway >> April 2021 >> IG Story >> 4/1/2021	-	1	-	n/a	-
Ref Url Example	-	0	-	n/a	-
Growth Tool #71	-	-	-	-	-
R&R QR Code	-	0	-	n/a	-
Growth Tool #69	-	3	-	100.0%	1
Growth Tool #68	0	0	n/a	n/a	-
Growth Tool #67	0	0	n/a	n/a	-
Growth Tool #66	-	0	-	n/a	-
Growth Tool #65	0	0	n/a	n/a	-
Growth Tool #64	-	0	-	n/a	-
Growth Tool #63	0	0	n/a	n/a	-

Figure 8: ManyChat Growth Tool Dashboard

Manychat allows users to import and export contacts from other services, such as CSV, XLS, and TXT files. This feature allows users to quickly and easily add subscribers to their ManyChat account, as well as export their contacts for use in other services. Additionally, ManyChat also provides an API (Application Programming Interface) that allows developers to integrate their applications with ManyChat, allowing users to sync their contacts across multiple services.

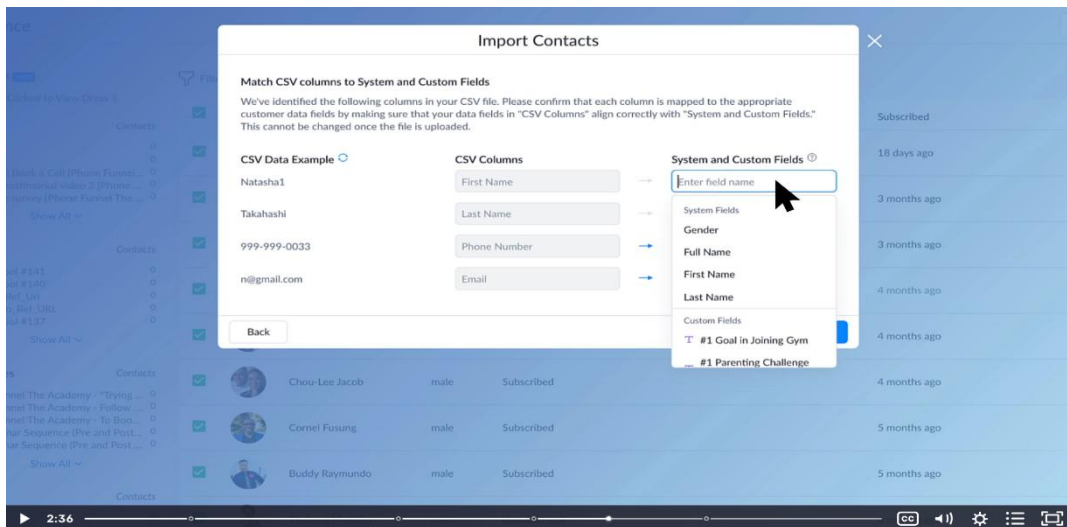


Figure 9: ManyChat Import Audience

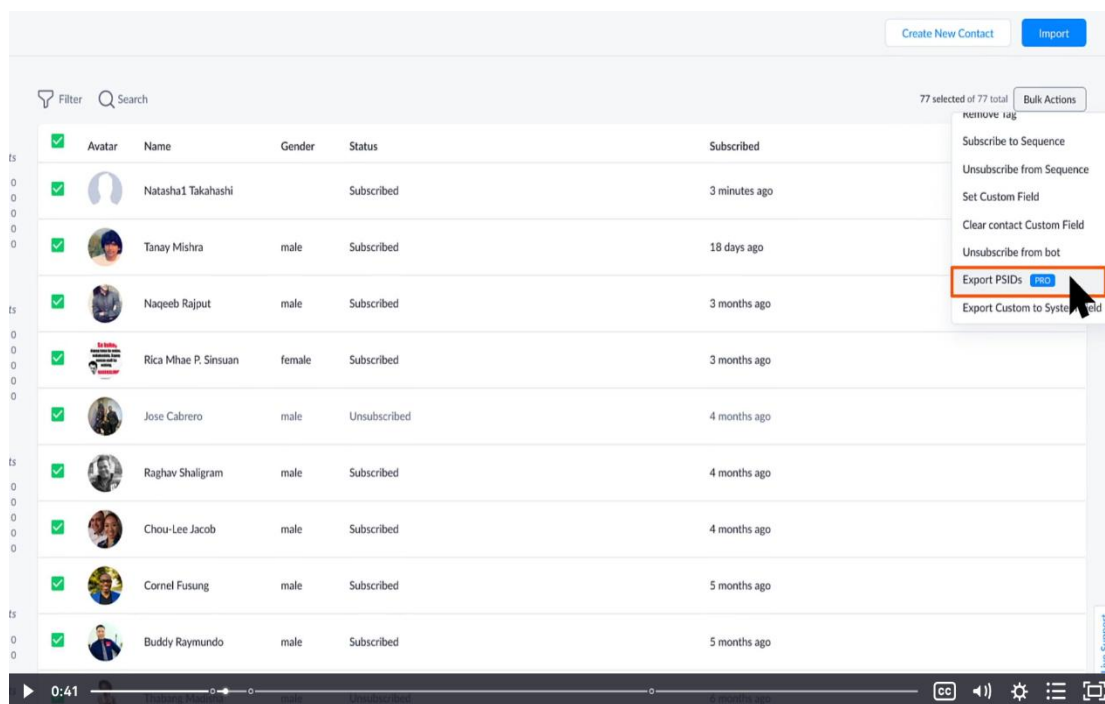


Figure 10: ManyChat Export Audience

ManyChat's Events feature allows users to easily create and host webinars, allowing them to engage with their audience in a more interactive and engaging way. Users can create their webinar, set up their registration page, and then send out invitations to their audience. The webinar can then be broadcasted on ManyChat and users can chat with each other in

real-time. Additionally, users can also access analytics on the performance of their webinars, allowing them to optimize their strategies and maximize their reach. In addition, ManyChat's Events Dashboard allows users to filter and search their data to better understand their audience. This feature allows users to filter their data by specific criteria, such as demographics, locations, or interests, so they can better target their messages to their desired audiences.

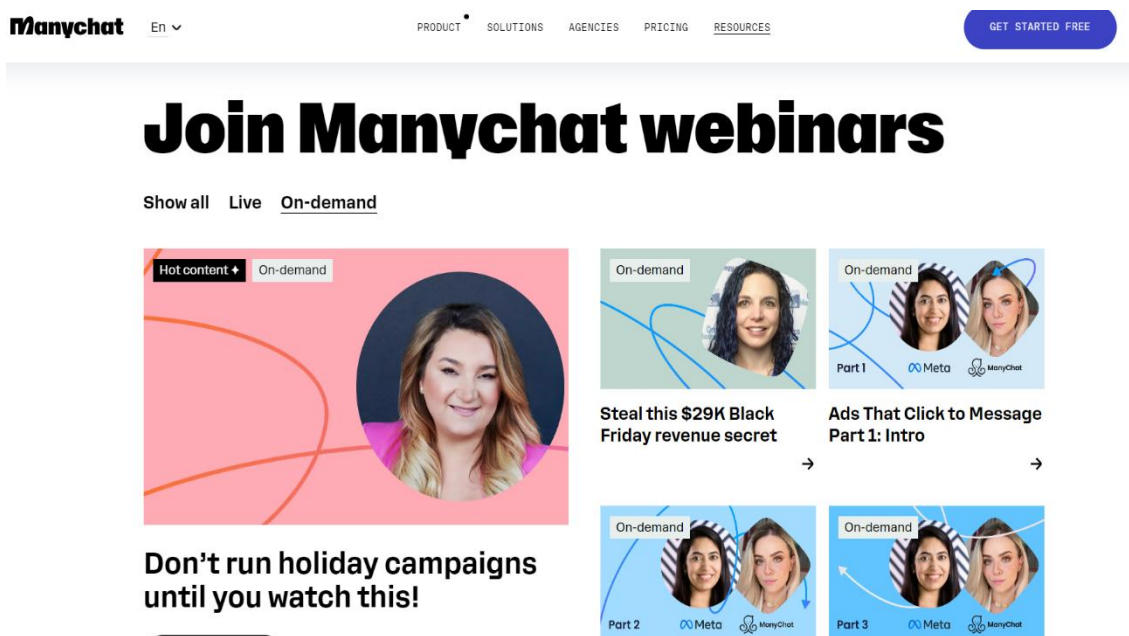


Figure 11: ManyChat Events Dashboard

2.3.2 Chatbot.ai

Chatbot.ai was developed by a team of engineers and software developers from IEEE, Harvard, and Cornell. The team is led by CEO and founder Shilpa Dhar, who has experience in artificial intelligence and natural language processing. The team also includes engineers with expertise in data science, machine learning, and computer engineering. [6] One example of an existing chatbot that uses Python, Java, C++, MySQL, and PHP is Chatbot.ai. This chatbot is an AI-powered virtual assistant that helps developers learn and build applications using these programming languages. Chatbot.ai offers tutorials, code snippets, and other resources to help users get the most out of their programming experience.

Additionally, the chatbot can provide advice and support on debugging, testing, and other topics related to coding. [7]

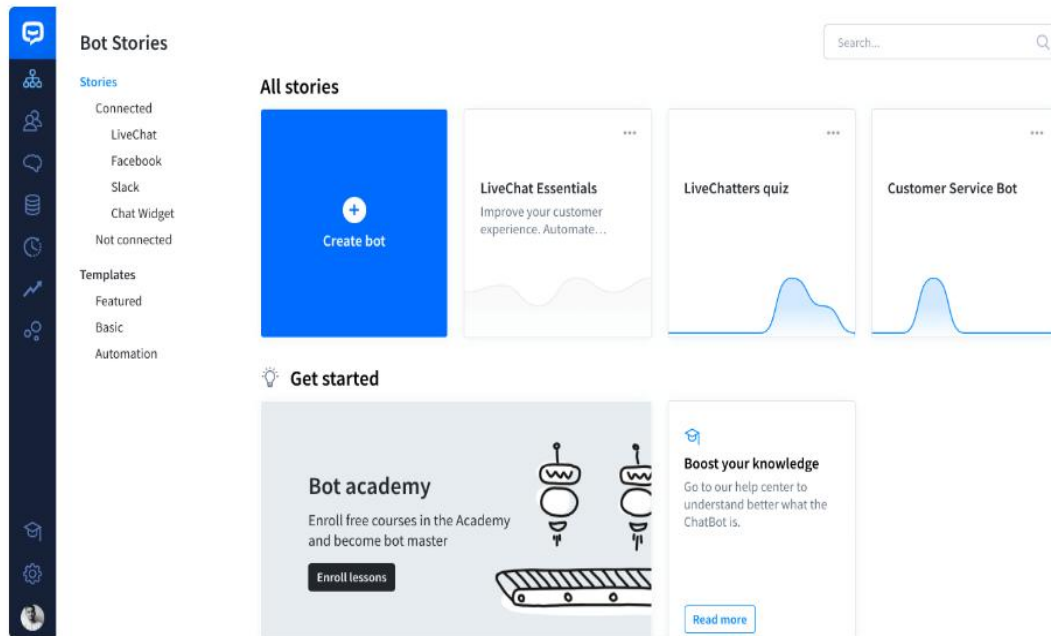


Figure 12: Chatbot.ai Dashboard

What differs between Chatbot.ai and ManyChat are that Chatbot.ai is a more complex platform that provides an AI-driven chatbot creation tool, allowing you to create rules-based chatbot conversations. ManyChat, on the other hand, is a more user-friendly platform that lets you create chatbots using ready-made templates, using a drag-and-drop interface. ManyChat also offers a range of marketing automation tools, such as lead generation and customer segmentation. The similarities are that Chatbot.ai can filter and search using collaborative filtering. It offers a range of features such as natural language processing (NLP) capabilities, sentiment analysis, and recommendation systems. It also uses machine learning algorithms to identify patterns in customer conversations and deliver personalized recommendations same as ManyChat.



Figure 13: Chatbot.ai Filter

2.3.3 Table 3.0 for comparisons between Chatbots Application

Features	ManyChat	Chatbot.ai
Platform Complexity	Low	High
AI-Driven Chatbot Creation	No	Yes
Ready-Made Templates	Yes	No
Drag-and-Drop Interface	Yes	No
Marketing Automation Tools	Yes	No
Natural Language Processing	Yes	Yes
Sentiment Analysis	Yes	Yes