

Introduction to Software Engineering

Software Design

***Design Document** for the assigned course project, following the attached template.*



Software Engineering Department
Faculty of Information and Technology
University of Science

Table of Contents

1	Member Contribution Assessment	2
2	Conceptual Model	3
3	Architectural Design.....	4
3.1	Architecture Diagram	4
3.2	Class Diagram	4
3.3	Class Specifications	6
3.3.1	Class Bot:.....	6
3.3.2	Class Message:.....	6
3.3.3	Class Group:.....	6
3.3.4	Class BotService:	7
3.3.5	Class MessageService:	7
3.3.6	Class GroupService:	7
3.3.7	Class CreateBotPage:	8
3.3.8	Class BotConfigurationPage:.....	8
3.3.9	Class ChatPage:	8
3.3.10	Class GroupPage:.....	8
4	Data Design.....	10
4.1	Data Diagram	10
4.2	Data Specification	10
5	User Interface and User Experience Design.....	17

Software Design

Objectives

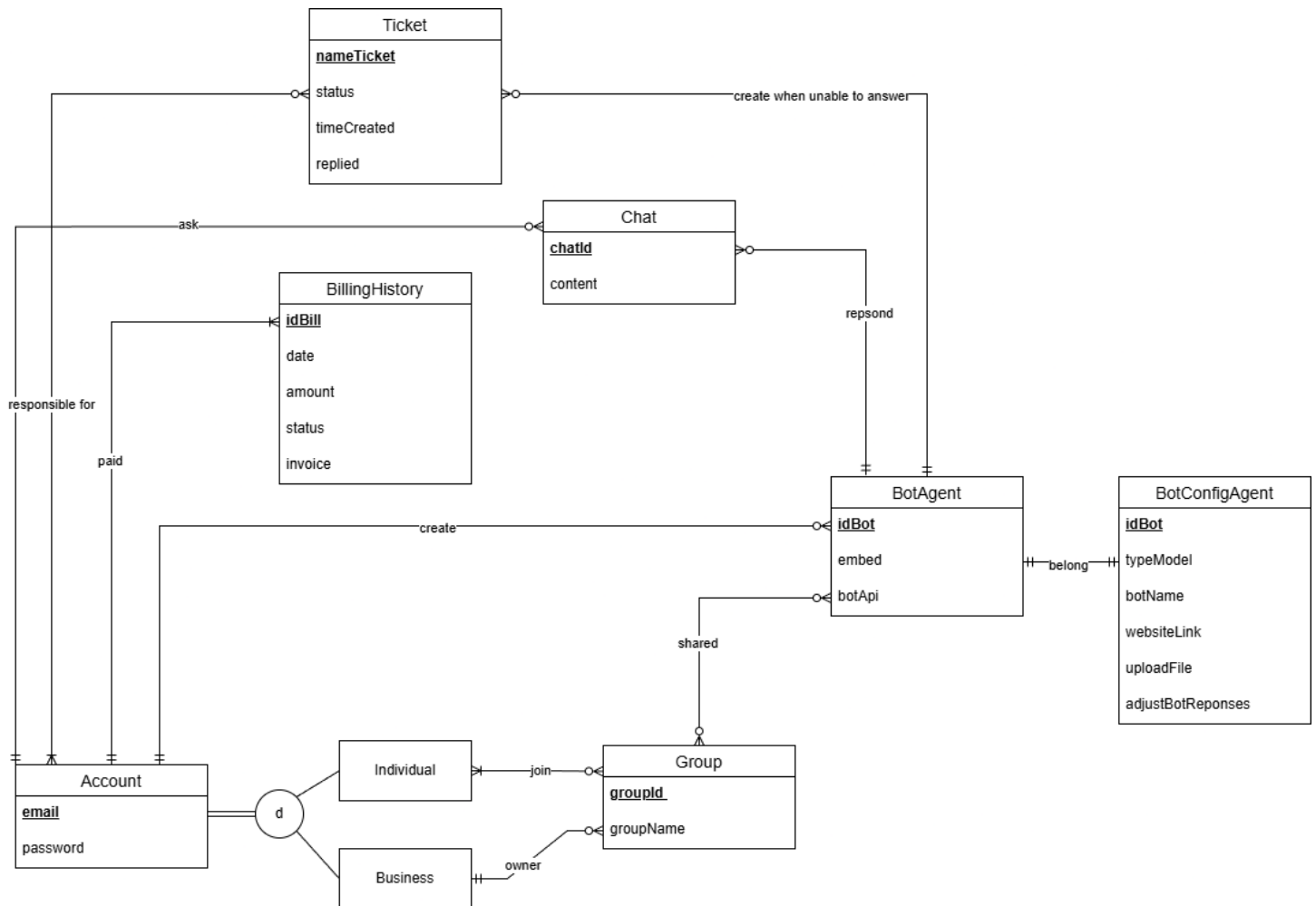
This document focus on the following topics:

- ✓ Complete the Software Design Document with the following contents:
 - Conceptual Model
 - Architectural Design
 - Data Design
 - User Interface Design
- ✓ Understanding the Software Design Document.

1 Member Contribution Assessment

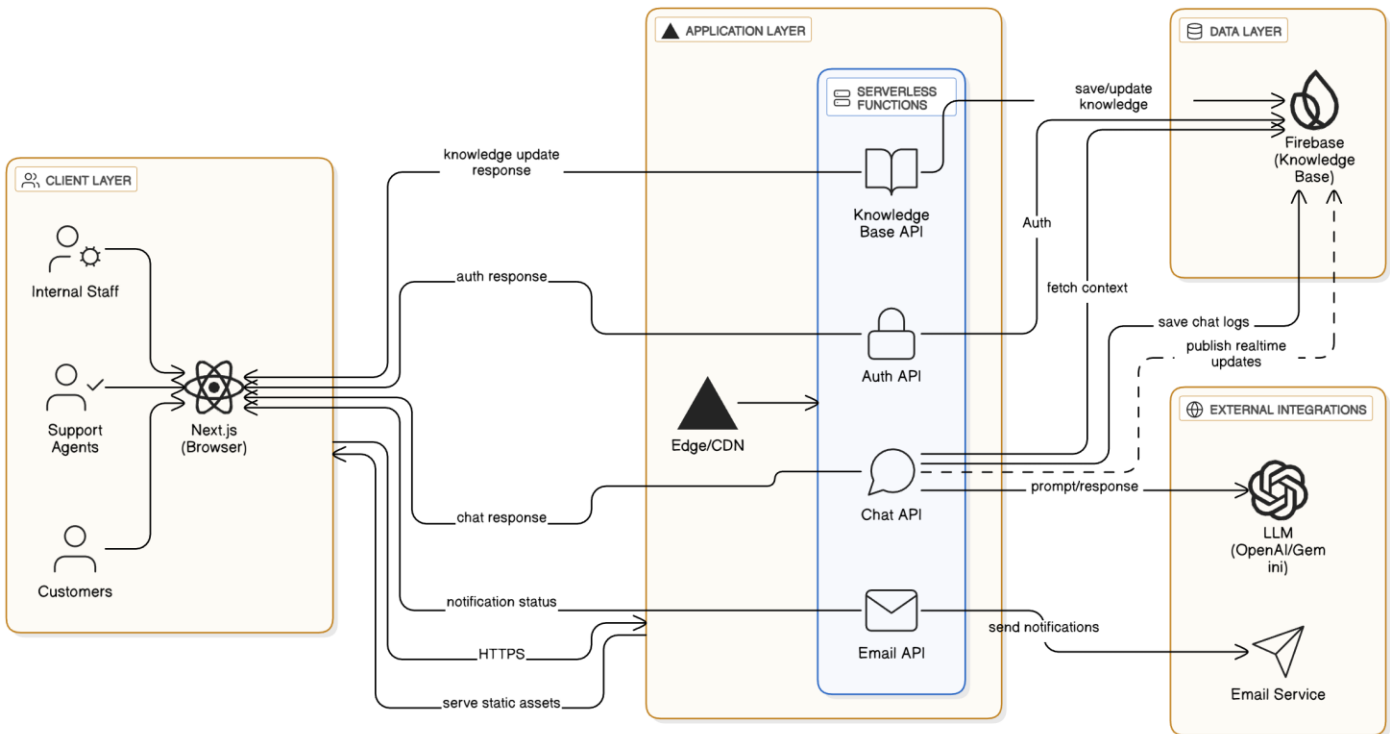
ID	Name	Contribution (%)	Signature
23127097	Trần Trí Nhân	100%	
23127168	Cao Trần Bá Đạt	100%	 Cao Trần Bá Đạt
23127193	Bùi Quang Hùng	100%	 Bùi Quang Hùng
23127234	Nguyễn An Nghiệp	100%	 Nguyễn An Nghiệp
23127238	Trần Hoài Thiện Nhân	100%	 Trần Hoài Thiện Nhân

2 Conceptual Model

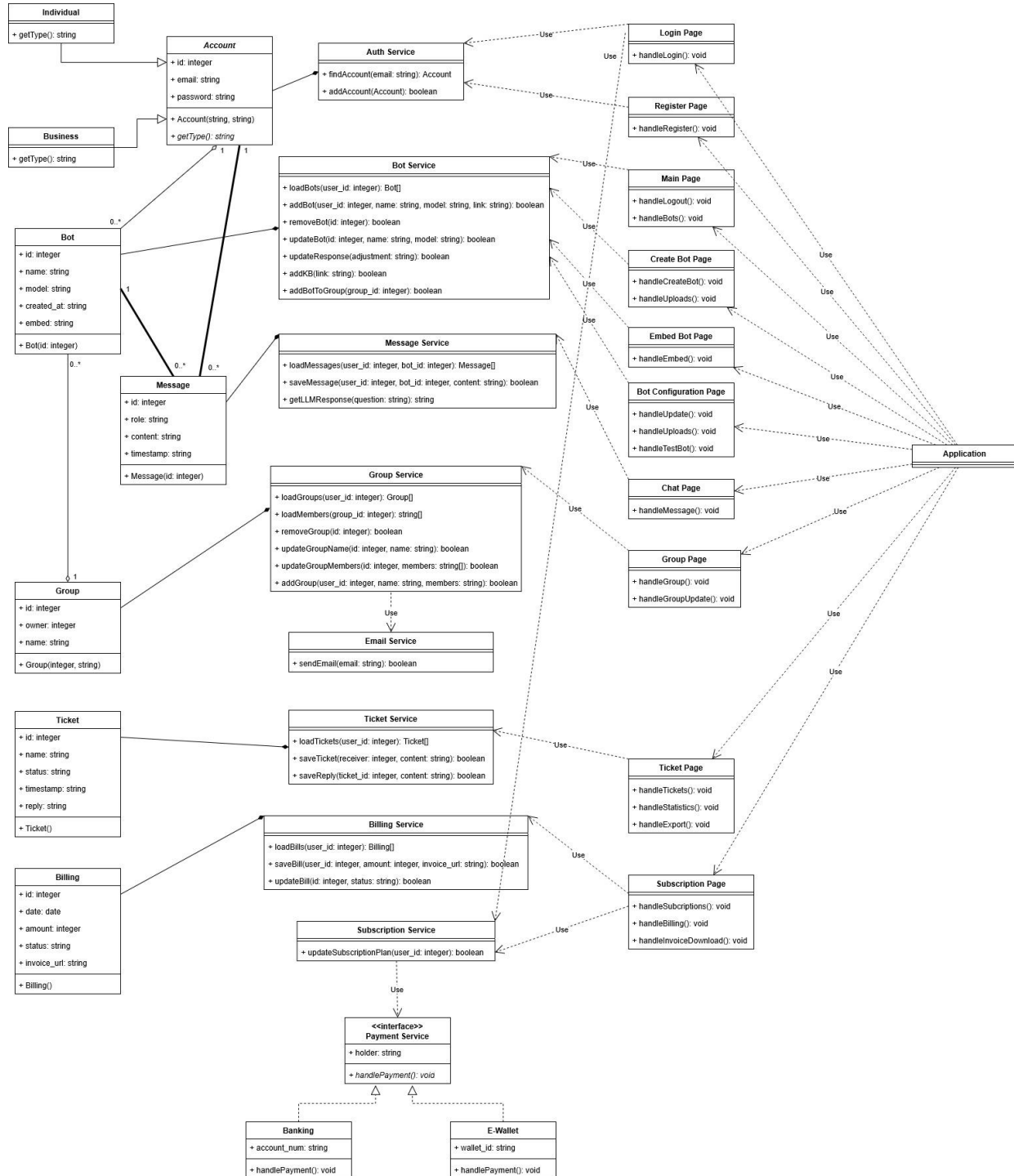


3 Architectural Design

3.1 Architecture Diagram



3.2 Class Diagram



3.3 Class Specifications

3.3.1 Class Bot:

Seq	Property	Modifier	Constraint	Description
1	id	public	integer, positive	ID of a bot
2	name	public	string	Name of a bot
3	model	public	string	LLM model of a bot
4	created_at	public	string, YYYY-MM-DD format	Creation date of a bot
5	embed	public	string, HTML format	Embedded code for a widget of a bot

Seq	Operation	Modifier	Constraint	Description
1	Bot(id: integer)	public		Bot's constructor

3.3.2 Class Message:

Seq	Property	Modifier	Constraint	Description
1	id	public	integer, positive	ID of a message
2	role	public	string, 'user' or 'bot'	Owner of a message
3	content	public	string	The content of a message
4	timestamp	public	string, HH:MM AM/PM format	The time when a message got sent

Seq	Operation	Modifier	Constraint	Description
1	Message(id: integer)	public		Message's constructor

3.3.3 Class Group:

Seq	Property	Modifier	Constraint	Description
1	id	public	integer, positive	ID of a group
2	owner	public	integer, positive	ID of group owner
3	name	public	string	The name of a group

Seq	Operation	Modifier	Constraint	Description
1	Group(owner: integer, name: string)	public		Group's constructor

3.3.4 Class BotService:

Seq	Operation	Modifier	Constraint	Description
1	loadBots(user_id: integer)	public		Get list of bots of a user
2	addBot(user_id: integer, name: string, model: string, link: string)	public		Add new bot
3	removeBot(id: integer)	public		Delete a bot
4	updateBot(id: integer, name: string, model: string)	public		Update name or model of a bot
5	updateResponse(adjustment: string)	public		Adjust response of a bot
6	addKB(link: string)	public		Update knowledge base of a bot
7	addBotToGroup(group_id: integer)	public		Share a bot to a group

3.3.5 Class MessageService:

Seq	Operation	Modifier	Constraint	Description
1	loadMessages(user_id: integer, bot_id: integer)	public		Load the conversation between a user and a bot
2	saveMessage(user_id: integer, bot_id: integer, content: string)	public		Add a message to database
3	getLLMResponse(question: string)	public		Retrieve a response from the AI model used by a bot

3.3.6 Class GroupService:

Seq	Operation	Modifier	Constraint	Description
1	loadGroups(user_id: integer)	public		Load list of groups of a user

2	loadMembers(group_id: integer)	public		Load list of members' emails of a group
3	removeGroup(id: integer)	public		Remove a group
4	updateGroupName(id: integer, name: string)	public		Change group name
5	updateGroupMembers(id: integer, members: string[])	public		Change members of a group
6	addGroup(user_id: integer, name: string, members: string)	public		Create a new group

3.3.7 Class CreateBotPage:

Seq	Operation	Modifier	Constraint	Description
1	handleCreateBot()	public		Control bot creation
2	handleUploads()	public		Validate and process files uploaded by a user

3.3.8 Class BotConfigurationPage:

Seq	Operation	Modifier	Constraint	Description
1	handleUpdate()	public		Control configurations of bot's name, bot's model and bot sharing
2	handleUploads()	public		Validate and process files uploaded by a user
3	handleTestBot()	public		Control the bot testing when configuring a bot

3.3.9 Class ChatPage:

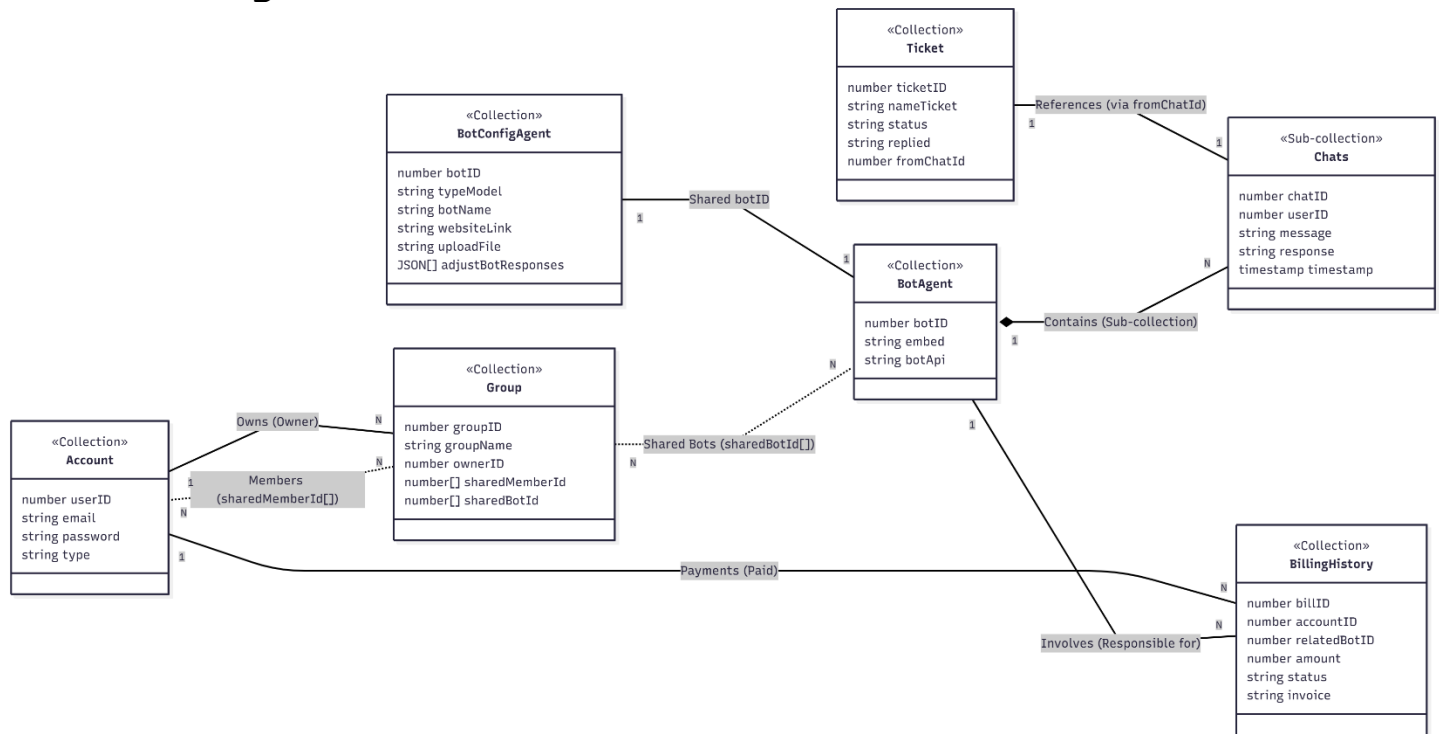
Seq	Operation	Modifier	Constraint	Description
1	handleMessage()	public		Control the messages sent by users and bots

3.3.10 Class GroupPage:

Seq	Operation	Modifier	Constraint	Description
1	handleGroup()	public		Control group listing and creation
2	handleGroupUpdate()	public		Control group editing

4 Data Design

4.1 Data Diagram



4.2 Data Specification

NoSQL Recap: Since the system uses Firebase (NoSQL database), data is organized as Collections and Documents instead of traditional relational tables. Foreign Key constraints are managed through application logic instead of hard constraints in the DB.

1. Collection: Account

- Description: Stores user and business account information*
- Sample JSON structure (Format):*

```
{
  "userID": "6",
  "email": "example@email.com",
  "password": "hashed_password_string",

```

"type": "business" // or "individual"

}

- *Property details:*

Attribute name	Data type	Constraints	Describe
<i>userID</i>	<i>Integer</i>	<i>X</i>	<i>Unique user ID</i>
<i>email</i>	<i>String</i>	<i>Unique, Not null</i>	<i>Login email</i>
<i>password</i>	<i>String</i>	<i>Hashed</i>	<i>Encrypted password</i>
<i>type</i>	<i>String</i>	<i>"business" or "individual"</i>	<i>Level of login account</i>

2. Collection: BotConfigAgent

- *Description: Save configuration for each bot*
- *Sample JSON structure (Format):*

```
{
  "botID": "3",
  "typeModel": "gpt",
  "botName": "Customer Support",
  "websiteLink": "https://example.com",
  "uploadFile": "link_to_drive_file",
  "adjustBotResponses": [
    {
      "question": " what is your company name
      " answer": " the company name is Sun "
    }
    {
      "question": " What is your boss's name?
      "answer": " My boss's name is Khoa"
    }
  ]
}
```

- *Property details:*

Attribute name	Data type	Constraints	Describe
-----------------------	------------------	--------------------	-----------------

<i>botID</i>	Integer	X	<i>Bot identifier</i>
<i>botName</i>	String	Not Null	<i>Bot Display Name</i>
<i>typeModel</i>	String	Valid values: "gpt 5", "gemini 3 pro", "gemini 3"	<i>Type of AI model used</i>
<i>websiteLink</i>	String	URL	<i>Website link of data learning bot (Scraping)</i>
<i>uploadFile</i>	String	URL (Storage)	<i>Link to file (PDF/Doc) uploaded to Firebase Storage</i>
<i>adjustBotResponses</i>	Array<String>	Question & answer	<i>Custom Response Directive (System prompt)</i>

3. Collection: BotAgent

- *Description: Contains embedded bot information and chat history*
- *Note: Chat is an array so it will be designed as a Sub-collection in Firebase*
- *Sample JSON structure (Format):*

```
{
  "botID": "3",
  "embed": "<iframe>code</iframe>",
  "botApi": "api_key_string"
}
// Sub-collection: Chats
"chats": [
  {
    "chatID": 6_1
    "userID": "user_6",
```

```

    "message": "Hello bot",
    "response": "OK"
    "timestamp": 1765021351
  }
]

```

- *Property details:*

Attribute name	Data type	Constraints	Describe
<i>botID</i>	<i>Integer</i>	<i>X</i>	<i>Bot identifier</i>
<i>embed</i>	<i>String</i>		<i>Embed bot code on website</i>
<i>botApi</i>	<i>String</i>	<i>Not Null</i>	<i>API Key to connect</i>

Sub-collection: chats (Located inside the BotAgent document)

- *Description: Save chat history between User and Bot*

Attribute name	Data type	Constraints	Describe
<i>chatID</i>	<i>String</i>	<i>X_Y</i> <i>X is for user X</i> <i>Y is for bot X</i>	<i>ID of the chat causing the error - create when unable to answer relationship</i>
<i>userID</i>	<i>Integer</i>	<i>X</i>	<i>ID of the person chatting with the Bot</i>
<i>message</i>	<i>String</i>	<i>Not Null</i>	<i>Content of the message sent by the user</i>
<i>response</i>	<i>String</i>		<i>Bot reply content</i>

<i>date</i>	<i>Unix Timestamp</i>	<i>Default: Now</i>	<i>Message time</i>
-------------	-----------------------	---------------------	---------------------

4. Collection: Group

- *Description: Manage user groups*
- *Sample JSON structure (Format):*

```
{
  "groupId": "1",
  "groupName": "Marketing Team",
  "ownerID": "6",
  "sharedMemberId": ["2", "3"],
  "sharedBotId" : ["2", "3"]
}
```

- *Property details:*

Attribute name	Data type	Constraints	Describe
<i>groupId</i>	<i>Number</i>	<i>X</i>	<i>Group identifier</i>
<i>groupName</i>	<i>String</i>	<i>Not Null</i>	<i>Group display name</i>
<i>ownerID</i>	<i>Integer</i>	<i>X</i>	<i>ID of the group creator (Group Admin)</i>
<i>sharedMemberID</i>	<i>Array<Integer></i>		<i>List of IDs of members invited to the group</i>
<i>sharedBotID</i>	<i>Array<Integer></i>		<i>List of Bot IDs shared in the group</i>

5. Collection: BillingHistory

- *Description: Stores user payment history, invoices and transaction status*
- *Sample JSON structure (Format):*


```
{
  "billID": "4",
  "accountID": "6",
  "date": "1765021351",
  "relatedBotID": "3",
  "amount": 200000,
  "status": "paid",
  "invoice": "https://firebasestorage.googleapis.com/.../invoice.pdf"
}
```

- *Property details:*

Attribute name	Data type	Constraints	Describe
<i>billID</i>	<i>Integer</i>	<i>X</i>	<i>Unique invoice identifier</i>
<i>accountID</i>	<i>Integer</i>	<i>X</i>	<i>ID of the account making the payment (used to query the user's bill history) - paid relationship</i>
<i>relatedBotID</i>	<i>Integer</i>	<i>X</i>	<i>Paid bots - responsible for relationships</i>
<i>date</i>	<i>Unix Timestamp</i>	<i>Not Null</i>	<i>Transaction execution time</i>
<i>amount</i>	<i>Integer</i>	<i>X</i>	<i>Payment amount</i>
<i>status</i>	<i>String</i>	<i>"paid" or "failed"</i>	<i>Transaction status</i>
<i>invoice</i>	<i>String</i>	<i>URL (Storage)</i>	<i>Link to invoice file (PDF/Image) saved on Storage</i>

6. Collection: Ticket

- *Description: Support tickets when the Bot cannot answer*
- *Sample JSON structure (Format):*

```
{
  "ticketID": "1",
  "nameTicket": " Payment error ",
  "question": "Why I can not pay? ",
  "status": "resolved",
  "timeCreated": "1765021351",
  "replied": " Refunded ",
  "fromChatId": "6_1"
}
```

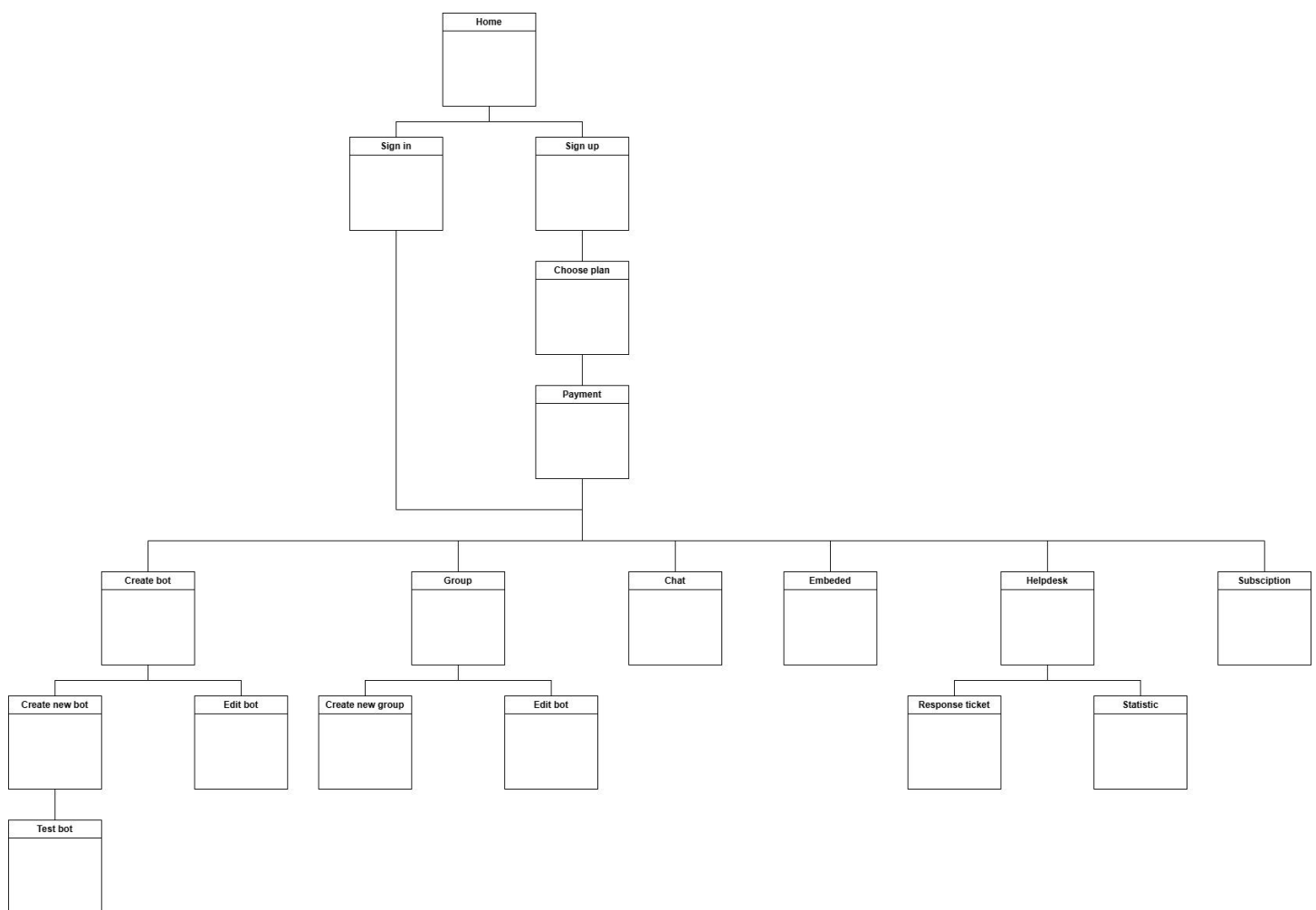
- *Property details:*

Attribute name	Data type	Constraints	Describe
<i>ticketID</i>	<i>Integer</i>	<i>X</i>	<i>Unique ticket identifier</i>
<i>nameTicket</i>	<i>String</i>	<i>Not Null</i>	<i>Problem Title</i>
<i>question</i>	<i>String</i>	<i>Not Null</i>	<i>Overview of the problem</i>
<i>status</i>	<i>String</i>	<i>"resolved" or "pending"</i>	<i>Processing status</i>
<i>timeCreated</i>	<i>Unix Timestamp</i>	<i>Default: Now</i>	<i>Time to create ticket</i>
<i>replied</i>	<i>String</i>		<i>Feedback content from admin</i>
<i>chatID</i>	<i>String</i>	<i>X_Y</i> <i>X is for user X</i> <i>Y is for bot X</i>	<i>ID of the chat causing the error - create when unable to answer relationship</i>

5 User Interface and User Experience Design

5.1 Screen Diagram.

- Screen diagram:



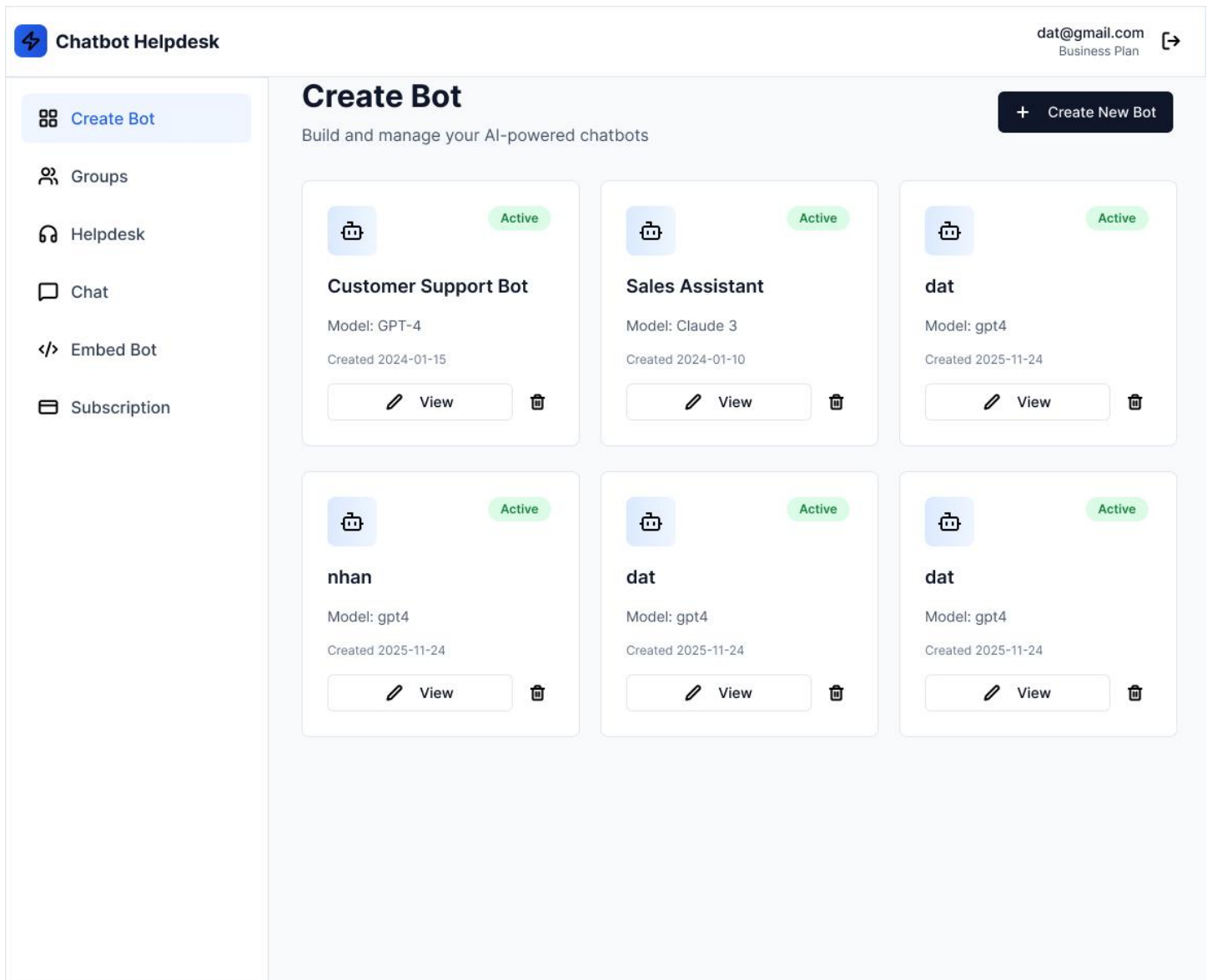
- List the screens.

Seq	Screen	Description
-----	--------	-------------

1	Home	Website home page interface
2	Sign in	Interface for users to log in to their accounts
3	Sign up	Interface for users to register for an account
4	Choose plan	Select subscription packages after entering your subscription information
5	Payment	Enter payment information
6	Create bot	Bot management page
7	Create new bot	The new bot creation page is not in the list
8	Test bot	Test bot page after creating new bot to edit bot answers
9	Edit bot	Edit page for created bots
10	Group	Group management page
11	Create new group	The new group creation page is not yet on the list.
12	Edit group	Page to edit existing groups
13	Chat	Chat page with created bots
14	Embedded	Page to embed bot on other website
15	Helpdesk	Page for general customer management
16	Response ticket	Page answers customers' specific questions
17	Statistic	Statistics of business indicators
18	Subscription	Page to adjust account subscriptions

5.2 Screen Specifications.

5.2.1 Create bot.



a) Page Title and Description.

- Create Bot
- Subtitle: *Build and manage your AI-powered chatbots*

b) Main Action Button: Create New Bot (primary call-to-action, located at the top-right of the content area).

c) Bot List (Bot Cards Grid).

Each chatbot is presented as an individual card containing:

Element	Description
---------	-------------

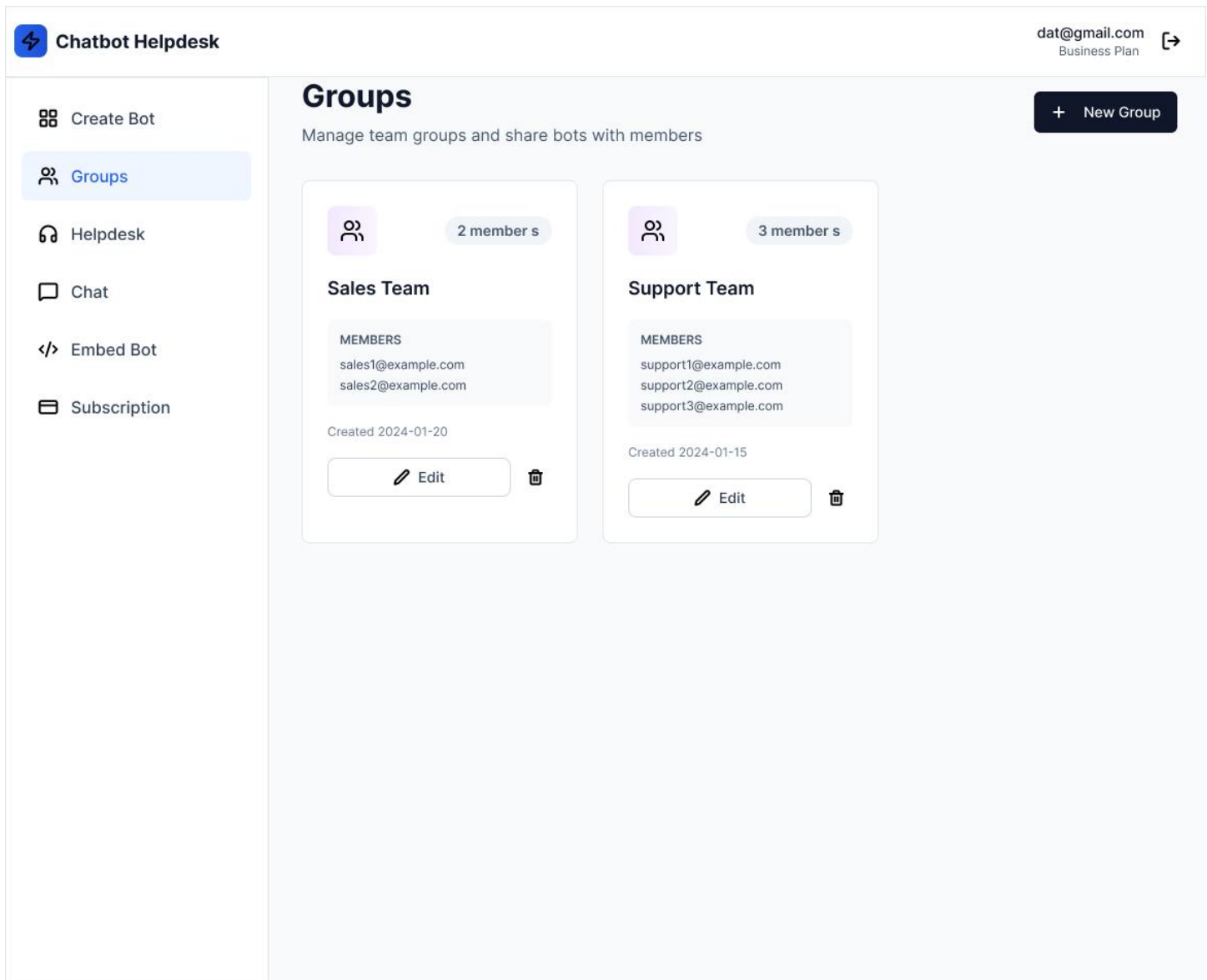
Bot icon	A small robot icon
Status badge	Active (green badge)
Bot name	e.g., Customer Support Bot, Sales Assistant
Model	e.g., GPT-4, Claude 3, gpt4
Created date	e.g., <i>Created 2024-01-15</i>
Edit button	Pencil icon
View button	“View” button
Delete button	Trash icon

Cards are displayed in a 3-column responsive grid.

d) Event Handling.

Event	System Behavior
Click “Create New Bot”	Redirects to “Create new bot” screen.
Click the View button	Opens “Edit bot” page for modifying bot name, model, settings, etc.
Click the Delete icon	Delete the bot containing this button
Hover over buttons	Visual highlight effect to improve focus and recognition.
Left page buttons	Move to other pages

5.2.2 Group.



a) Title and Description.

- Groups
- Subtitle: *Manage team groups and share bots with members*

b) Primary Action Button: New Group (Top-right corner) - This button allows users to create a new team group.

c) Group Cards Grid.

Each group is displayed as a card containing:

Element	Description
---------	-------------

Group icon	A small group/people icon
Member count badge	e.g., “2 members”, “3 members”
Group name	e.g., Sales Team, Support Team
Members list	Email addresses of team members
Created date	e.g., Created 2024-01-20
Edit button	Pencil icon ("Edit" label)
Delete button	Trash icon

d) Event Handling.

Event	System Behavior
Click “New Group”	Opens the “Create new group” screen where the user enters group name, members, permissions, etc.
Click the Edit button	Redirects to the “Edit group” screen, allowing modification of member list
Click the Delete icon	Delete the group containing this button
Hover over buttons	Highlights the buttons to indicate interactivity.
Left page buttons	Move to other pages

5.2.3 Helpdesk.

Chatbot Helpdesk dat@gmail.com Business Plan

Helpdesk

Tickets Statistics

Open Tickets
2

Resolved Tickets
1

Avg Resolution Time
2.5h

All Tickets

- Bot not responding to greetings
open high
- Integration issue with API
resolved high
- Feature request: custom responses
open low

Select a ticket to view details and reply

The Helpdesk dashboard is divided into two tabs:

- Tickets
- Statistics

a) Metrics Overview Cards.

Displayed horizontally at the top:

Metric	Description
Open Tickets (2)	Shows the total number of currently open tickets.
Resolved Tickets (1)	Displays the number of tickets resolved by the support team.

Avg Resolution Time (2.5h)	Displays the average time required to resolve previous tickets.
----------------------------	---

Each card includes a small icon and uses a clean numerical emphasis.

b) Ticket List Panel (Left Section).

A list titled All Tickets, showing individual ticket summaries.

Each ticket item includes:

Element	Example from UI
Title	Bot not responding to greeting
Status badge	open / resolved
Priority badge	high / low
Additional spacing and background highlight	indicates click ability

Badges use color-coded labels:

- open → yellow
- resolved → green
- high → red
- low → blue

This enhances scannability of ticket state and urgency.

c) Ticket Detail Panel (Right Section)

When a ticket is selected, this panel will display its details:

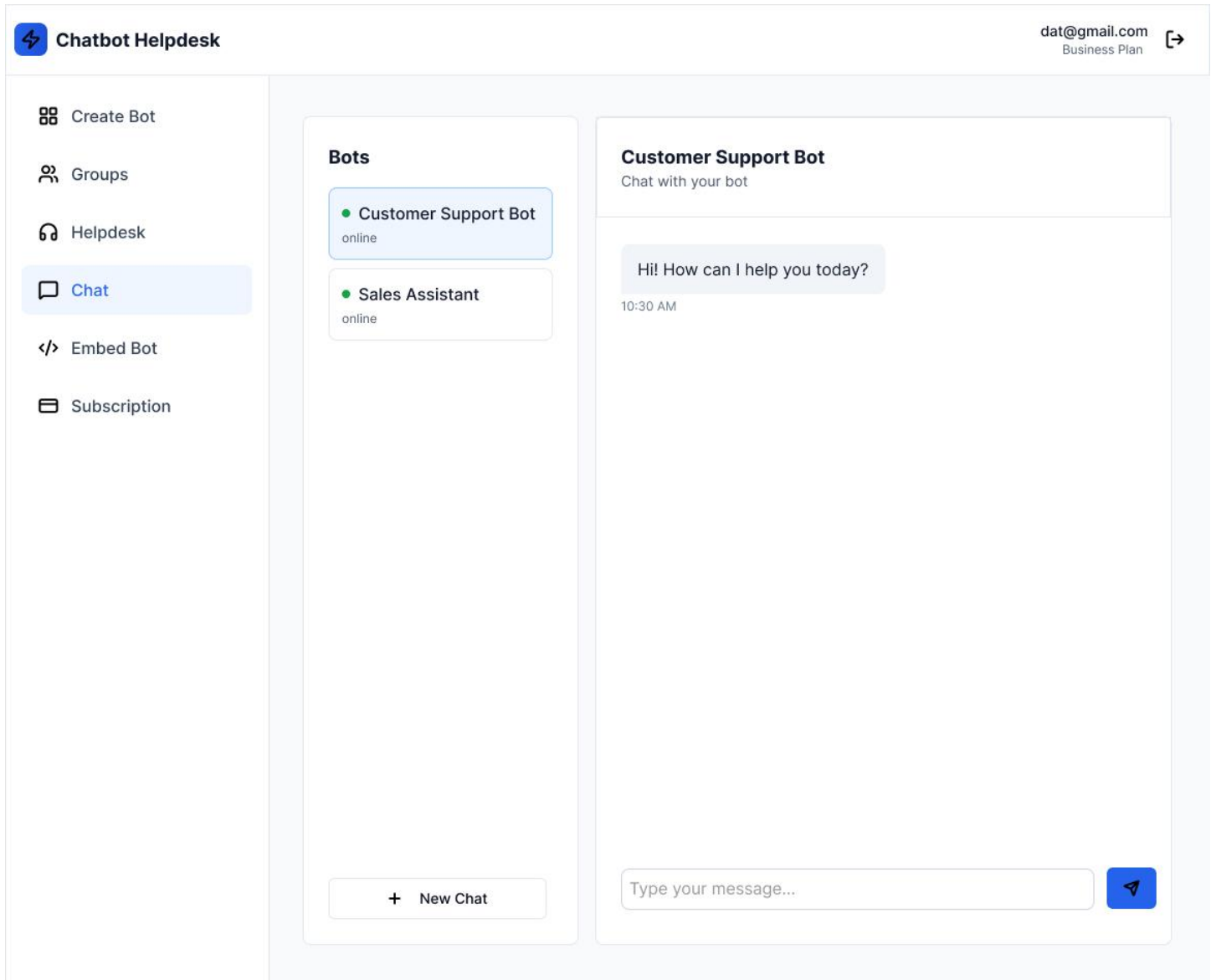
- Full description
- Conversation history
- Reply box
- Status update actions (resolve / reopen)

d) Event Handling.

Event	System Behavior
Click "Tickets" tab	Displays the ticket management dashboard (current view)
Click "Statistics" tab	Go to the "Statistics" page
Select a ticket in the list	Loads ticket details into the right panel for viewing and replying.

Hover over buttons	Button highlights to show interactivity.
Left page buttons	Move to other pages

5.2.4 Chat.



a) Bot List Panel (Left Side).

Displayed under the header Bots, this panel contains:

- A list of bots available for conversation

- Each bot row includes:
 - Bot name (e.g., *Customer Support Bot*, *Sales Assistant*)
 - Status indicator: green dot with label online
 - Highlighted selection state (blue outline) when active
- “+ New Chat” button located at the bottom, allowing users to start a new conversation session with the selected bot.

b) Chat Window Panel (Right Side).

The chat panel displays:

Chat Header

- Bot name: *Customer Support Bot*
- Subtitle: *Chat with your bot*

Message Area

- Conversation messages displayed chronologically

Messages are contained within rounded chat bubbles (light gray for bot messages).

Message Input Bar

Located at the bottom of the chat window:

- Text input field labeled “**Type your message...**”
- Send button (paper-plane icon)

This allows users to input messages and send them to the bot.

c) Event Handling.

Event	System Behavior
Select a bot from the bot list	Loads the bot’s conversation history into the chat window and highlights the selected bot.
Click “New Chat”	Initiates a new conversation session with the currently selected bot; clears the chat window.
User types a message	Input is captured in the message textbox.
Click Send button or press Enter	Sends the message to the bot, appends it to the chat window, and triggers a response from the backend.
Bot sends a response	The message appears as a bot bubble with timestamp.

Scroll in chat area	Allows reviewing older messages.
Left page buttons	Move to other pages

5.2.5 Embedded.

The screenshot shows the 'Embed Bot' interface of a Chatbot Helpdesk. On the left is a sidebar with navigation options: 'Create Bot', 'Groups', 'Helpdesk', 'Chat', 'Embed Bot' (highlighted), and 'Subscription'. The main area is titled 'Embed Bot' and contains the following elements:

- A header: 'Select a bot and embed it on your website'.
- A 'Select Bot' section with two cards:
 - 'Customer Support Bot' (Active) with a blue selection checkbox.
 - 'Sales Assistant' (Active) with an unselected checkbox.
- An 'Embed Code' section with a black text box containing the code and a blue 'Copy Code' button.
- An 'Installation Instructions' box with three steps:
 1. Copy the embed code above
 2. Paste it into your website's HTML, usually before the closing tag
 3. The bot widget will automatically appear on your website

The top right of the interface shows the user 'dat@gmail.com' on a 'Business Plan'.

1. Select Bot Panel (Left Side).

A list of bots that can be embedded into external websites. Each bot is displayed as a selectable card containing:

Element	Description
Bot name	e.g., <i>Customer Support Bot, Sales Assistant</i>
Status badge	Active (indicates bot availability)
Selection indicator	Radio/checkbox UI for selecting the bot to embed

The selected bot card is highlighted with a blue border and filled selection indicator.

b) Embed Code Panel (Right Side).

This section displays:

- A black code box representing the embed script (HTML/JS snippet)
- A Copy Code button with copy icon

Pressing the button copies the script to the user's clipboard.

c) Installation Instructions Section (Bottom Panel).

A highlighted instructional box provides step-by-step installation steps:


1. Copy the embed code above
2. Paste it into your website's HTML, usually before the closing `</body>` tag
3. The bot widget will automatically appear on your website


This aims to ensure non-technical users can integrate the bot with minimal difficulty.


d) Event Handling.


Event	System Behavior
Select a bot card	Highlights the chosen bot and updates the embed code field to reflect that bot's unique widget script.
Click "Copy Code"	Copies the embed script to the clipboard; may trigger a confirmation toast such as "Code copied!".
Hover over buttons	Applies a soft shadow or outline to indicate clickability.
Left page buttons	Move to other pages


5.2.6 Subscription.


 Chatbot Helpdesk


dat@gmail.com
Business Plan 


 Create Bot

 Groups

 Helpdesk

 Chat

 Embed Bot

 Subscription

Subscription

Manage your plan and billing

Current Plan

Business - 500,000 VND/Month + 100 VND/Token

Next billing date: March 22, 2024

[Cancel Subscription](#)

Available Plans

Individual

Perfect for personal projects

200,000 VND/month

- ✓ Unlimited bots
- ✓ 1 AI model connection
- ✓ Basic analytics
- ✗ Team collaboration
- ✗ Priority support

[Upgrade to Individual](#)

Business

For growing businesses and teams

500,000 VND/month + 100 VND/token

- ✓ Unlimited bots
- ✓ All AI models
- ✓ Team collaboration
- ✓ Advanced analytics
- ✓ Priority support

[Current Plan](#)

Billing History

Date	Amount	Status	Invoice
Feb 22, 2024	500,000 VND	Paid	Download
Jan 22, 2024	500,000 VND	Paid	Download

a) Current Plan Panel.

A highlighted box shows the user's active subscription:

- Plan name: *Business – 500,000 VND/Month + 100 VND/Token*

- Next billing date: *March 22, 2024*
- Cancel Subscription link (top-right corner)

This panel provides a summary of the user's current plan and billing cycle.

b) Available Plans Section.

This section displays subscription tiers the user can choose from:

- Individual Plan Card:

- Price: 200,000 VND/month
- Description: *Perfect for personal projects*
- Features:
- Upgrade to Individual button

- Business Plan Card (Selected):

Visually highlighted with a blue border.

- Price: 500,000 VND/month + 100 VND/token
- Description: *For growing businesses and teams*
- Features:
- Button displays Current Plan (disabled)

This layout allows quick comparison between subscription tiers.

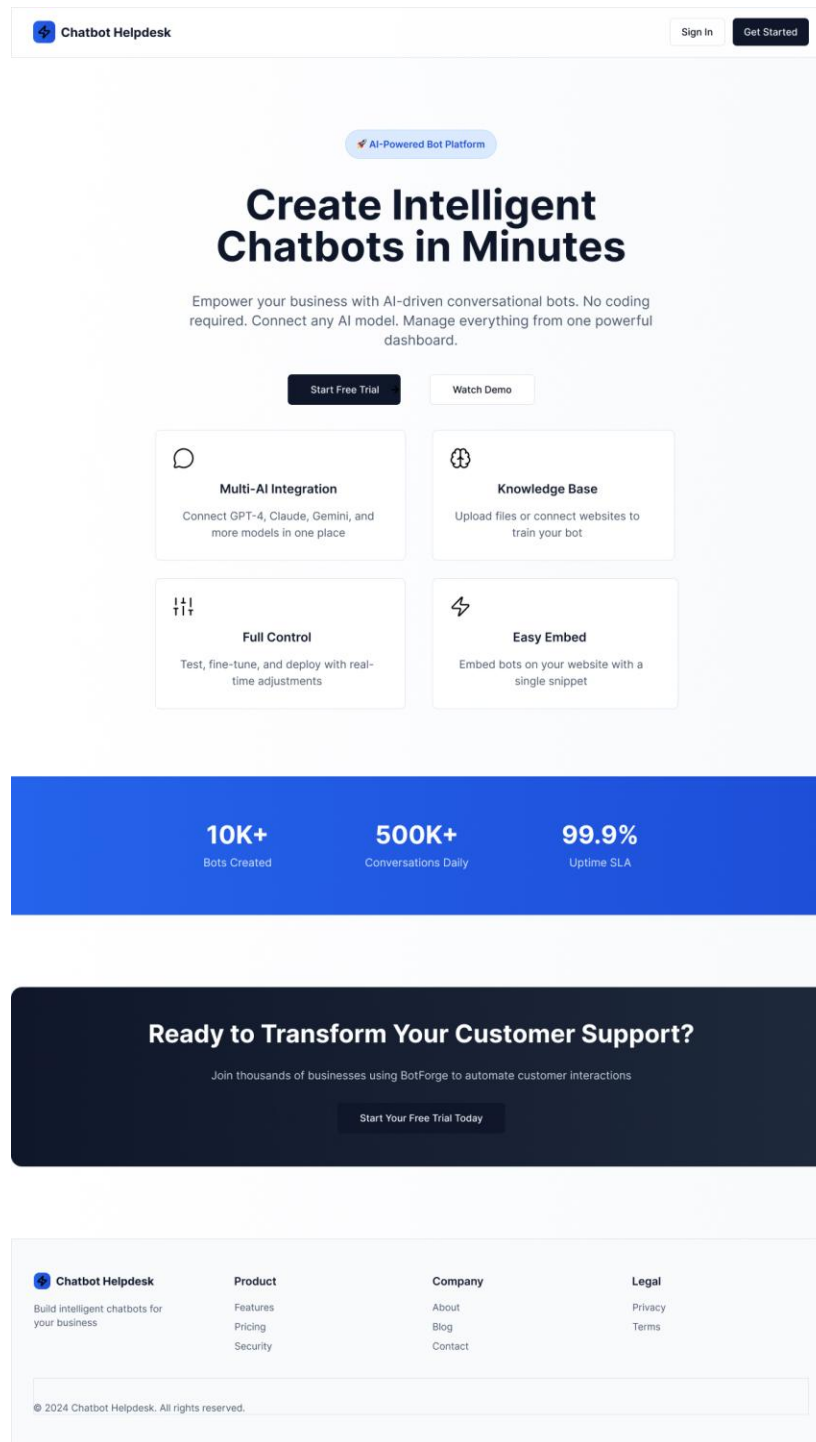
c) Billing History Section.

- A table listing previous invoices.
- This provides financial transparency and easy access to past invoices.

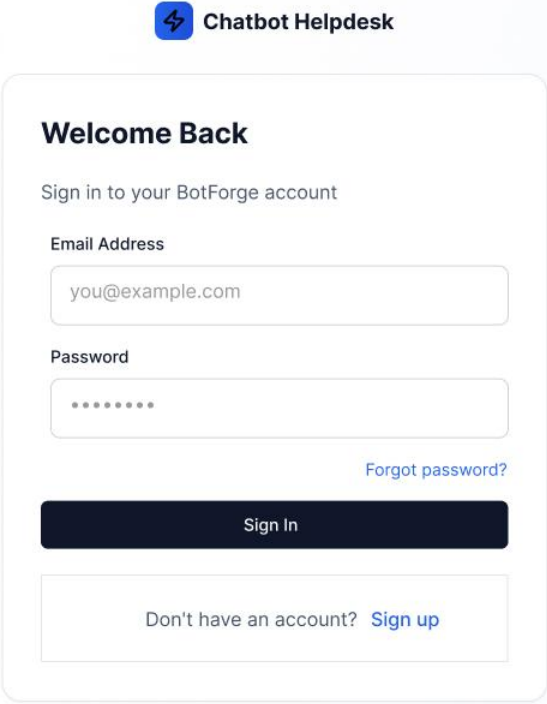
d) Event Handling.

Event	System Behavior
Click "Cancel Subscription"	Cancel current subscription
Click "Upgrade to Individual"	Switches the user's active plan to Individual; updates billing and plan features.
Click "Current Plan"	No action (button disabled).
Click "Download" in Billing History	Downloads the corresponding invoice PDF file.
Hover over buttons	Applies a soft shadow or outline to indicate clickability.
Left page buttons	Move to other pages

5.2.7 Home.



5.2.8 Sign in.



The image shows a sign-in form for BotForge. At the top, there is a logo with a blue square containing a white lightning bolt icon, followed by the text "Chatbot Helpdesk". Below this is a white rounded rectangle containing the sign-in form. The form has a heading "Welcome Back", a subheading "Sign in to your BotForge account", and two input fields: "Email Address" with the placeholder "you@example.com" and "Password" with a masked password ".....". To the right of the password field is a link "Forgot password?". Below the input fields is a dark blue "Sign In" button. At the bottom of the form is a link "Don't have an account? Sign up". Below the form is a line of text: "By signing in, you agree to our Terms of Service".

Chatbot Helpdesk

Welcome Back

Sign in to your BotForge account

Email Address

Password

[Forgot password?](#)

Sign In

Don't have an account? [Sign up](#)

By signing in, you agree to our Terms of Service

5.2.9 Sign up.



Create Account

Get started with Chatbot Helpdesk

Email Address

Password

Continue

Already have an account? [Sign in](#)

By signing up, you agree to our [Terms of Service](#)

5.2.10 Choose plan.



Choose Your Plan

Select the plan that fits your needs

Individual

Perfect for personal projects and small tasks

- ✓ Unlimited bots
- ✓ 1 AI model connection
- ✓ Basic analytics

200,000VND/month

Business

Popular

Ideal for growing businesses and teams

- ✓ Unlimited bots
- ✓ All AI models
- ✓ Team collaboration
- ✓ Advanced analytics
- ✓ Priority support

500,000VND/month

+ 100 VND per token

By signing up, you agree to our Terms of Service

5.2.11 Payment.



Payment Method

Choose your payment method for your subscription

Bank Account

Transfer directly from your bank account

E-Wallet

Pay using popular e-wallet services
(GCash, PayMaya, etc.)

Bank/E-Wallet Details

Account Holder Name


Verification Code


Create Account & Subscribe


[Back](#)


By signing up, you agree to our [Terms of Service](#)


5.2.12 Create new bot.


 Chatbot Helpdesk


dat@gmail.com
Business Plan 


 Create Bot


 Groups

 Helpdesk

 Chat

 Embed Bot

 Subscription

 **Create New Bot**
Configure your AI-powered chatbot

AI Model

GPT-4
Most advanced

Claude 3
Balanced & fast

Gemini
Cost effective

Share with Groups

Select Groups (1)


Sales Team x


Bot Configuration

Bot Name *

e.g., Customer Support Bot

Knowledge Base

 **Add Website Link**
Train bot with website content

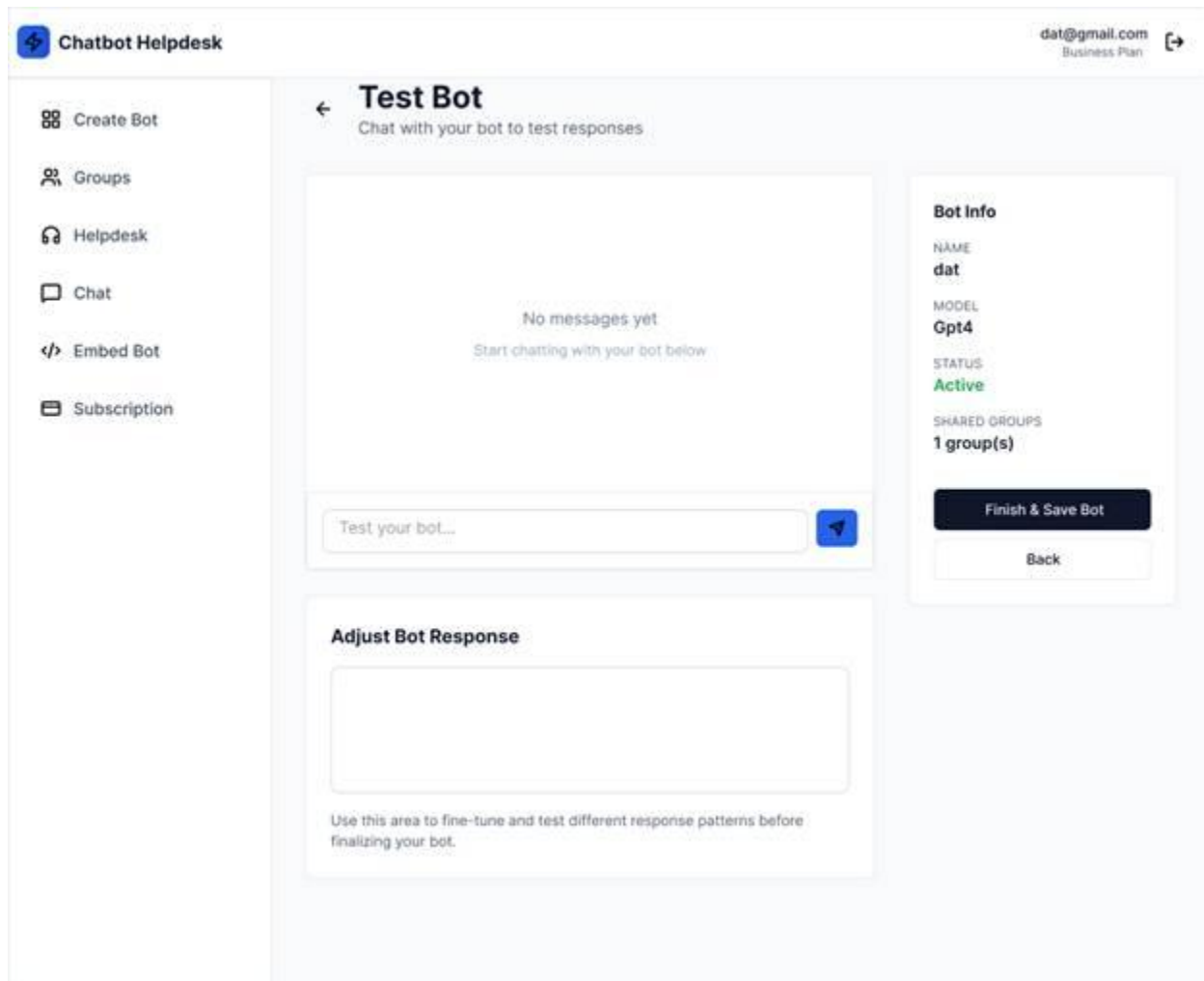
 **Upload File**
PDF, TXT, or DOCX files

Create Bot

5.2.13 Test bot.

HCMUS | SE Dept.

36



5.2.14 Edit bot.

The screenshot shows the 'Bot Details & Testing' interface of a Chatbot Helpdesk. The interface is divided into a left sidebar and a main content area. The sidebar contains navigation links: 'Create Bot', 'Groups', 'Helpdesk', 'Chat', 'Embed Bot', and 'Subscription'. The main content area is titled 'Bot Details & Testing' and includes a subtitle 'Edit and test your chatbot configuration'. It features three main sections: 'Bot Configuration', 'Knowledge Base', and 'Test Chat'. The 'Bot Configuration' section includes a 'Bot Name' field, an 'AI Model' dropdown (with options GPT-4, Claude 3, and Gemini), a 'Save Changes' button, and a 'Share with Groups' section. The 'Knowledge Base' section includes a 'Website Link' field and an 'Upload File' button. The 'Test Chat' section includes a 'Test message...' input field and a 'Send' button. Below the 'Test Chat' section is an 'Adjust Bot Response' section with a text area for customizing responses.

Chatbot Helpdesk dat@gmail.com Business Plan

Bot Details & Testing

Edit and test your chatbot configuration

Bot Configuration

Bot Name *

AI Model

GPT-4
Most advanced

Claude 3
Balanced

Gemini
Fast

Save Changes

Share with Groups

Select Groups (0)

Knowledge Base

Website Link
Train bot with website content

Upload File
PDF, TXT, or DOCX files

Test Chat

Interact with your bot


No messages yet
Start testing below


Test message...


Adjust Bot Response


Customize responses to fine-tune how your bot answers user messages.


5.2.15 Create new group.


 Chatbot Helpdesk


dat@gmail.com
Business Plan 


 Create Bot

 Groups

 Helpdesk

 Chat

 Embed Bot

 Subscription

Groups

Manage team groups and share bots with members


Create New Group

Group Name *

Member Emails (comma separated) *

Enter valid email addresses separated by commas

Create GroupCancel


 2 member s


Sales Team

MEMBERS

sales1@example.com
sales2@example.com

Created 2024-01-20

Edit


 3 member s

Support Team


MEMBERS


support1@example.com
support2@example.com
support3@example.com


Created 2024-01-15


Edit


5.2.16 Edit group.


 Chatbot Helpdesk


dat@gmail.com
Business Plan 


 Create Bot

 Groups

 Helpdesk

 Chat

 Embed Bot

 Subscription

Groups

Manage team groups and share bots with members


Edit Group

Group Name *

Member Emails (comma separated) *

Enter valid email addresses separated by commas

Update GroupCancel


 2 member s


Sales Team

MEMBERS

sales1@example.com
sales2@example.com

Created 2024-01-20

Edit


 3 member s

Support Team


MEMBERS


support1@example.com
support2@example.com
support3@example.com


Created 2024-01-15


Edit


5.2.17 Response tickets.


 Chatbot Helpdesk


dat@gmail.com
Business Plan 


 Create Bot

 Groups

 Helpdesk


 Chat


 Embed Bot


 Subscription

Helpdesk

Tickets Statistics

Open Tickets
2 

Resolved Tickets
1 

Avg Resolution Time
2.5h 

All Tickets

Bot not responding to greetings
open high


Integration issue with API
resolved high

Feature request: custom responses
open low

Bot not responding to greetings


Open High Created 2024-01-22

The bot fails to respond when users greet it with common phrases.


 Replies


No replies yet


Reply to this ticket


 Reply


5.2.18 Statistics.


 Chatbot Helpdesk


dat@gmail.com
Business Plan 


 Create Bot

 Groups

 Helpdesk

 Chat

 Embed Bot


 Subscription

Helpdesk

Tickets Statistics

Export Statistics

Daily Resolutions



Ticket Trends

