

**CMPE 187 - Software Quality Engineering**

**Spring 2023 - Group Project**

**Deliverable 1 - Conventional Test Report**

Team member:

Khai Nguyen

Faith Deanon

Kevin Campbell

Long Nguyen

<b>Section #1: Introduction</b>	<b>3</b>
1.1 App Information	3
1.2 Test Information	4
1.3 Task Partition	4
1.4 Project schedule	4
<b>Section #2: Test requirements</b>	<b>5</b>
AI function requirements	5
<b>Section #3 – Selected test models and methods</b>	<b>6</b>
3.1 Test methods	6
3.2 Decision table testing	6
3.3 Equivalence partitioning	7
3.4 Scenario testing	7
<b>Section #4 – Test case design with test data</b>	<b>7</b>
4.1 Decision table testing	7
4.1.1 Detailed Decision Table	8
4.1.2 Decision Table Test Cases	10
4.1.3 Decision Table Test Coverage	18
4.2 Equivalence testing	18
4.2.2 Equivalence Partitioning Test Cases	19
4.2.3 Equivalence Partitioning Test Coverage	28
4.3 Scenario testing	29
4.3.1 Scenario Testing Designs	29
4.3.2 Scenario Testing Test Cases	32
4.3.3 Scenario Testing Coverage	39
<b>Section #5 – Test result analysis and bug summary</b>	<b>39</b>
5.1 Test Results Summary	39
Decision Table Test Summary	39
Equivalence Partitioning Test Summary	40
Scenario testing test summary	41
5.2 Manual Test Costs	41
5.3 Complexity and coverage	41
5.4 Bug analysis and summary	41
Bugs	41
Detecting feelings	42

# Section #1: Introduction

## 1.1 App Information

Wysa is an AI-powered chatting app to help users with anxiety by providing mental health support. In addition to free support through chat, it also offers exercises to reduce anxiety, paid coaching plans with actual experts, and self-care exercises.

However, the testing will only cover AI software chatting capabilities, especially how it can handle mental health conversations.

## 1.2 Test Information

We apply different types of black box testing in order to evaluate the effectiveness of Wysa's AI.

## 1.3 Task Partition

Member name	Task
Khai Nguyen	Decision table testing, Scenario Testing
Faith Deanon	Decision table testing, Scenario Testing
Kevin Campbell	Equivalence partition testing, Scenario Testing
Long Nguyen	Equivalence partition testing, Scenario Testing

## 1.4 Project schedule

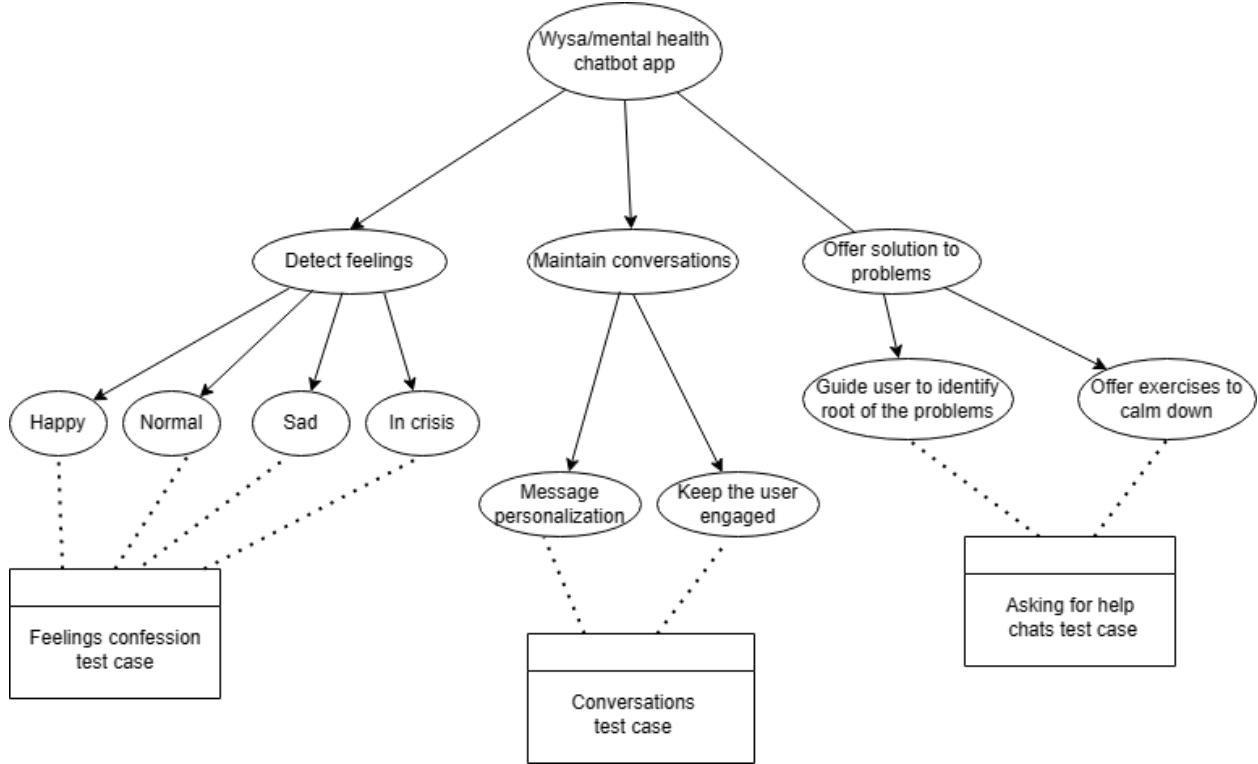
Dates	Description
Mar 1 - Mar 6	Research about the app
Mar 7 - Mar 10	Come up with test cases
Mar 11 - Mar 16	Test and collect data
Mar 17	Submission

## Section #2: Test requirements

### AI function requirements

The app is tested on both Android and iOS platforms.

For the chatting function, below is the AI requirements tree. The tests are designed to test the features listed on the tree.



## Section #3 – Selected test models and methods

### 3.1 Test methods

We chose two conventional black box testing methods to test the Wysa app, that is decision table, category partition testing methods. The decision table focuses more on the overall quality of the conversation while category partition testing methods focus more on testing the input data.

The decision table is a technique that tests the software behavior based on different combinations of input and the current state of the software. It is also called cause and effect table testing.

Category partition testing methods is a technique that divides input domains into different classes and each test case is taken from this class.

Scenario testing is about testing the app through the user's perspective and going through the whole scenario instead of just testing inputs.

### **3.2 Decision table testing**

Decision table testing is used to test the overall quality of the conversation, to see whether or not the chatbot can fulfill its three main functions: detect feelings, maintain conversations, and offer solutions to mental health problems.

### **3.3 Equivalence partitioning**

This type of test will be used to evaluate how good it is for Wysa to extract relevant info from diverse types of inputs, ranging from normal responses to invalid data inputs such as containing numbers only.

### **3.4 Scenario testing**

Scenario testing is a type of test to test how users would use the software end to end in different scenarios. We will use scenario testing to test different scenarios of how the user might use the app from start to finish. The test will not test invalid inputs/outputs but rather about use cases and conversation flow.

## Section #4 – Test case design with test data

### 4.1 Decision table testing

The four conditions below are selected to be part of the decision table testing

1. Is your answer related to the current conversation?

Wysa should be able to carry out conversations with users and change topics accordingly.

2. Does the answer express feelings?

Wysa should be able to detect the user's feelings and respond appropriately.

3. Does the answer change what you said previously?

Wysa should be able to update the information it has about you.

4. Is the answer one of the offered responses?

In certain situations, Wysa offers users some of the offered responses. However, it should also understand answers that do not follow the prompt to accommodate flexibility.

Conditions 1 and 4 must be either False or both True because if the user's answer using one of the offered responses (4), then it means the user's answer is related to the current conversation.

Outcomes

1. Wysa should ask to clarify what has just been said
2. Wysa should offer solutions to de-escalate the user's feelings if the feelings are negative
3. Wysa should be able to update old data with the new one
4. Wysa should be able to continue the conversation in the appropriate direction with message personalization

#### 4.1.1 Detailed Decision Table

Conditions	R1	R2 (X)	R3	R4 (X)	R5	R6 (X)	R7	R8 (X)	R9 (X)	R10	R11 (X)	R12	R13 (X)	R14	R15 (X)	R16
C1	F	F	F	F	F	F	F	T	T	T	T	T	T	T	T	T
C2	F	F	F	F	T	T	T	F	F	F	F	T	T	T	T	T
C3	F	F	T	T	F	F	T	T	F	F	T	T	F	F	T	T
C4	F	T	F	T	F	T	F	T	F	T	F	T	F	T	F	T
Actions																
O1	X															
O2					X		X									X
O3			X				X					X				X
O4	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

R1: Invalid answer that does not contribute anything significant to the conversation

R3: The user wants to update previously said information

R5: The user suddenly wants to express how they are feeling

R7: The user suddenly wants to express how they are feeling, which is different from what previously said

R10: The user is continuing the conversation with the chatbot, performing an exercise, engaging in conversations, etc, and answered with one of the offered responses

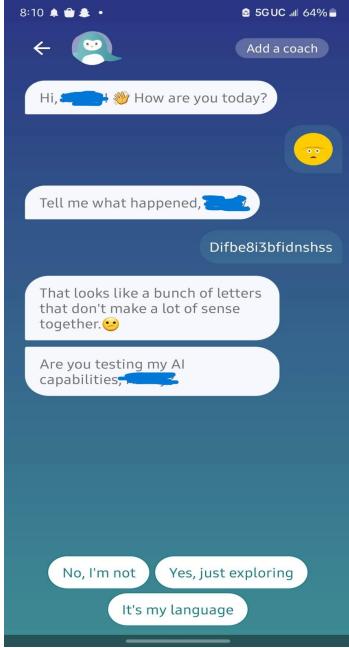
R12: The user is continuing the conversation with the chatbot, but wants to correct what was previously said and answered with one of the offered responses

R14: The user expressing how they feel to the chatbot as part of the current conversation and answered with one of the offered responses

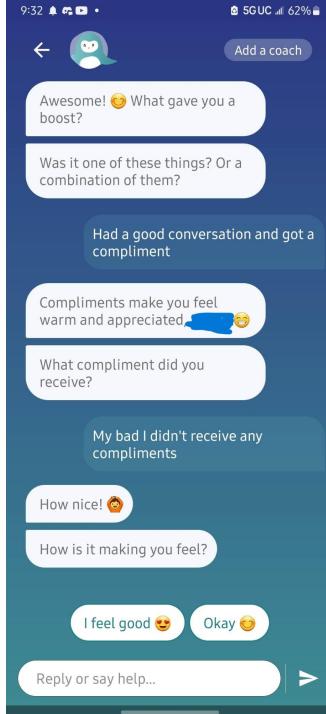
R16: The user expresses how they feel to the chatbot, but wants to correct what was previously said, maybe because their mood changes. User answered with one of the offered responses and answered with one of the offered responses

#### 4.1.2 Decision Table Test Cases

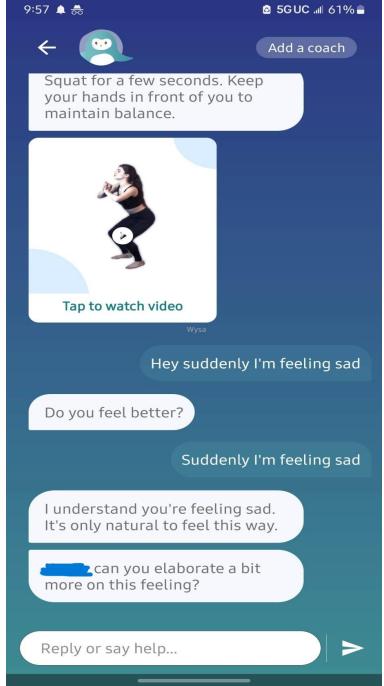
Test case ID	R1
Test Description	User inputs an invalid answer that does not contribute anything significant to the conversation

Test case input	
Situation	Incoherent text in the middle of the chat
Expected Result	Wysa should ask to clarify what just been said
Actual Result	Wysa said what entered didn't make any sense
Test Case Result	Pass

Test case ID	R3
Test Description	User wants to update previously said information

Test case input	 <p>The screenshot shows a mobile application interface for Wysa. At the top, there's a header with a back arrow, a profile icon of a owl-like character, and a button labeled "Add a coach". Below the header, the main conversation area has the following messages:</p> <ul style="list-style-type: none"> <li>User: "Awesome! 😊 What gave you a boost?"</li> <li>Wysa: "Was it one of these things? Or a combination of them?"</li> <li>User: "Had a good conversation and got a compliment"</li> <li>Wysa: "Compliments make you feel warm and appreciated 😊"</li> <li>User: "What compliment did you receive?"</li> <li>Wysa: "My bad I didn't receive any compliments"</li> <li>User: "How nice! 🎉"</li> <li>Wysa: "How is it making you feel?"</li> <li>User: "I feel good 😊" (with a blue highlighted bubble)</li> <li>Wysa: "Okay 😊" (with a blue highlighted bubble)</li> </ul> <p>At the bottom of the screen, there's a text input field with the placeholder "Reply or say help..." and a right-pointing arrow button.</p>
Situation	User said they were having a good day, and Wysa asked what happened. The user said they had a good conversation and had a compliment but later said those didn't happen.
Expected Result	Wysa should be able to update data with the new one and offer message personalization.
Actual Result	Wysa assumed that the text was part of the conversation and completely ignored it
Test Case Result	Fail

Test case ID	R5
Test Description	User suddenly wants to express how they are feeling

Test case input	
Situation	Wysa was guiding the user through squat exercises then the user suddenly said they were feeling sad.
Expected Result	Wysa should offer de-escalation if there's any, as well as continue the conversation in the correct direction
Actual Result	Wysa was able to detect feelings inside the reply despite the whole conversation being about working out. Then it asks the user to elaborate on the feeling
Test Case Result	Pass

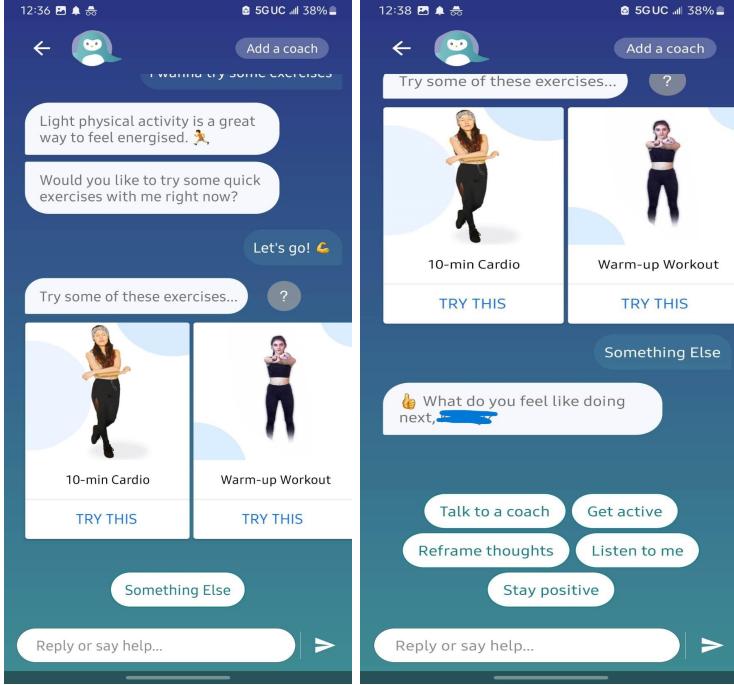
Test case ID	R7
Test Description	User suddenly wants to express how they are feeling, which is different from what previously said

Test case input	
Situation	User suddenly expressed that they were happy, Wysa asked to elaborate on that feeling. However, the user suddenly changed their feeling and said they were feeling sad instead.
Expected Result	Wysa should offer de-escalation if there's any, update old data with old ones as well as continue the conversation in the correct direction and offer message personalization
Actual Result	Wysa failed to recognize the change in feelings and assume what the user said.
Test Case Result	Fail

Test case ID	R10
Test Description	The user is continuing the conversation with the chatbot, performing an exercise, engaging in conversations, etc., and answered with one of the offered responses

Test case input	
Situation	Wysa asked the user how they were feeling. The user then proceeded with the conversations using one of the offered response choices and conversed with Wysa.
Expected Result	Wysa should be able to continue the conversation in the correct direction with message personalization
Actual Result	Wysa was able to offer message personalization as well as continue the conversation
Test Case Result	Pass

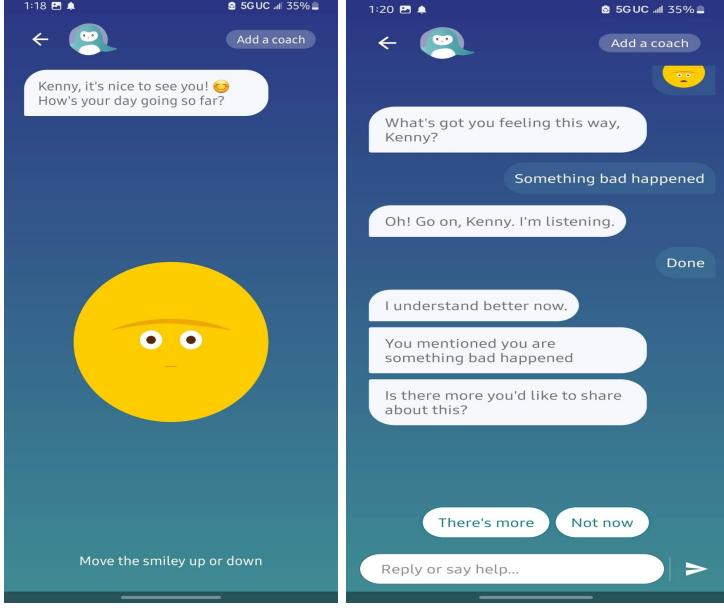
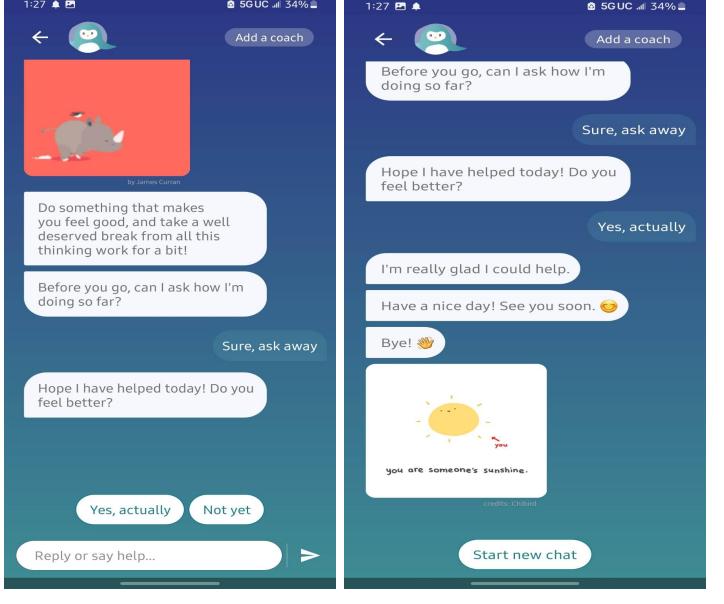
Test case ID	R12
Test Description	User is continuing the conversation with the chatbot, but wants to correct what was previously said using one of the offered responses

<b>Test case input</b>	 <p>The screenshots show a conversation between a user and a Wysa coach. In the first screenshot, the user asks for quick exercises, and the coach suggests a 10-min Cardio session. In the second screenshot, the user changes their mind, and the coach offers alternative options like 'Something Else' or 'Get active'.</p>
<b>Situation</b>	User asked to go through a workout. After seeing a list of exercises, the user changed their mind
<b>Expected Result</b>	Wysa should be able to pick up the change of mind as well as continue the conversation in the correct direction
<b>Actual Result</b>	Wysa was able to notice the user didn't want to work out anymore and suggested other things to talk
<b>Test Case Result</b>	Pass

<b>Test case ID</b>	R14
<b>Test Description</b>	User expresses how they feel to the chatbot as part of the current conversation using one of the offered responses

<b>Test case input</b>	<p>The screenshot shows a mobile application interface for a chatbot named 'Wysa'. At the top, there's a header with the time '1:03', signal strength, battery level '36%', and a '5G UC' indicator. Below the header is a profile picture of a cartoon owl and a button 'Add a coach'. The main area is a conversation screen. The user's message 'Hi, Ok! 😊 How are you today?' is at the top. The bot's response 'Most days we are somewhere in the middle of our emotional range.' follows. A question from the user 'If you were to try and describe your current mood, what kind of an 'okay' are you feeling right now?' is shown. Below these messages is a row of six circular emotion buttons: 'Meh', 'Optimistic', 'Calm', 'Numb', 'Bored', and 'Confused'. At the bottom is a text input field with 'Reply or say help...' and a send button with a right-pointing arrow.</p>
<b>Situation</b>	Wysa asked for the user's feelings whenever a new chat started. The user responded with an adjustable face feeling ok.
<b>Expected Result</b>	Wysa should be able to track user's feelings and offer an appropriate response
<b>Actual Result</b>	Wysa detected the user feeling ok and asked the user if they were ready to elaborate
<b>Test Case Result</b>	Pass

<b>Test case ID</b>	R16
<b>Test Description</b>	User expresses how they feel to the chatbot, but wants to correct what was previously said, maybe because their mood change using one of the offered responses

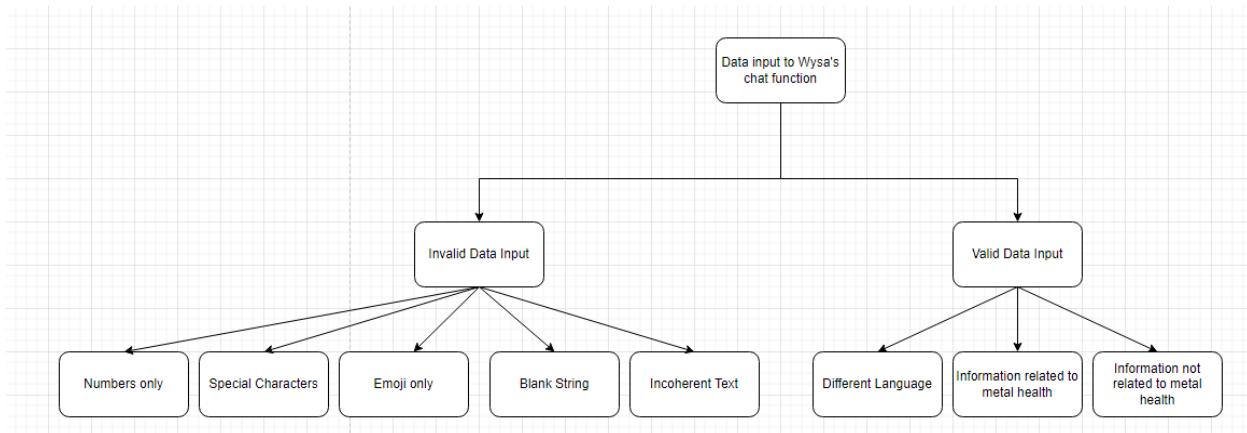
<b>Test case input</b>	 
<b>Situation</b>	Wysa asked for the user's feelings when a new chat started. User responded with a sad face. After elaborating, Wysa offered ways to improve the situation. The user responded with one of the offered responses and said they felt better.
<b>Expected Result</b>	Wysa was able to notice the change in feelings and continue the conversation in the correct conversation.
<b>Actual Result</b>	Wysa was able to notice that the user was feeling better and wished the user a good day
<b>Test Case Result</b>	Pass

### 4.1.3 Decision Table Test Coverage

Even though we have 16 combinations, 8 of the combinations didn't make sense, as explained above: condition 1 and 4 must be both true or both false, because if the user answers using one of the responded questions, then the user is following the prompt. That left us with 8 combinations left, and since we have one test case for each partition, this means we have 100% coverage.

## 4.2 Equivalence testing

### 4.2.1 Detailed Equivalence Partitioning Tree



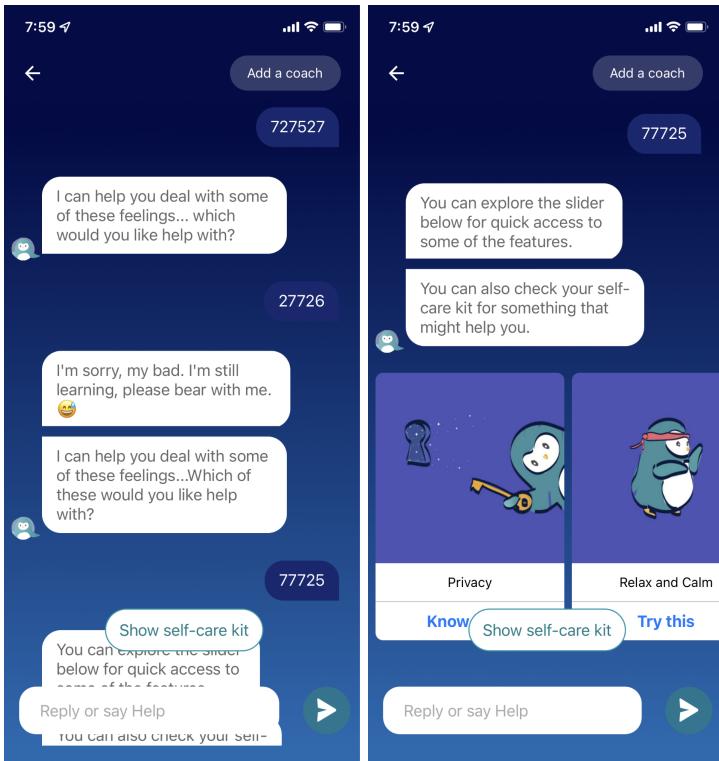
#### 1) Invalid Inputs

- a) Respond in numbers
- b) Respond in incoherent texts
- c) Respond in special characters
- d) Respond in emojis
- e) Respond in blank strings

#### 2) Valid Inputs

- a) Respond in different languages
- b) Provide information related to your mental health
- c) Provide information not related to your mental health

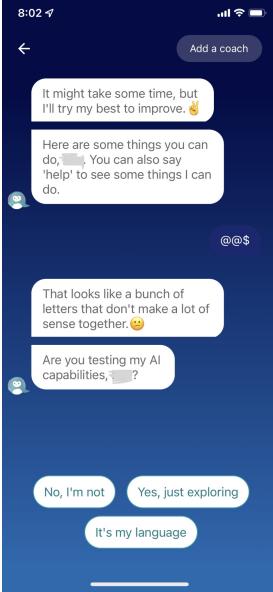
#### 4.2.2 Equivalence Partitioning Test Cases

Test Case ID	1
Test Description	Respond in numbers only
Test Input	
Expected Result	Tell the user it does not understand what was said
Actual Result	Kept recommending the app's various services/features
Test Case	Failed

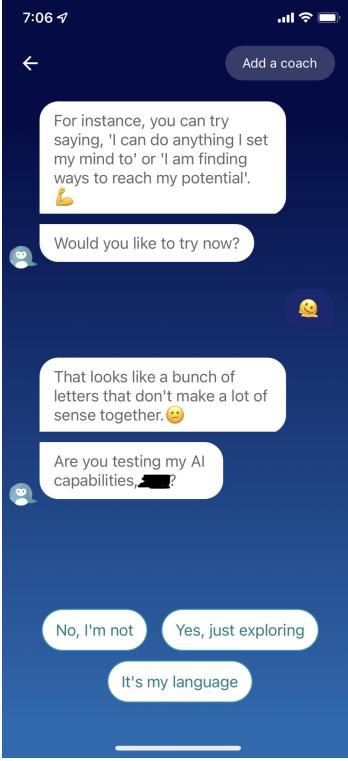
Result	
--------	--

Test Case ID	2
Test Description	Respond in incoherent texts
Test Input	
Expected Result	Carry on as normal
Actual Result	Asks if you are testing the AI's capabilities
Test Case Result	Passed

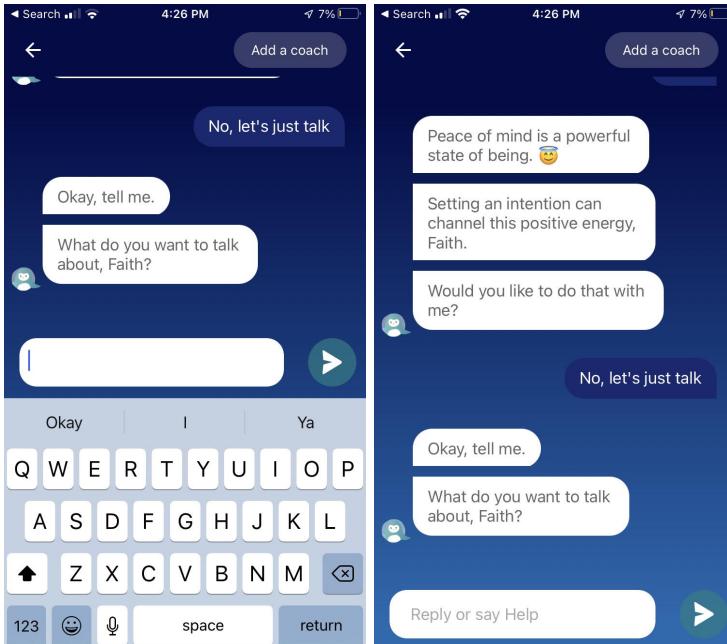
Test Case ID	3
Test Description	Respond in special characters

Test Input	 <p>The screenshot shows a mobile application interface for a coaching service. At the top, it says "Add a coach". The conversation starts with the AI coach saying:</p> <p>"It might take some time, but I'll try my best to improve. 🌟"</p> <p>Then, the user types "@@\$". The AI coach responds:</p> <p>"Here are some things you can do, [redacted]. You can also say 'help' to see some things I can do."</p> <p>Next, the user types "That looks like a bunch of letters that don't make a lot of sense together. 😊". The AI coach replies:</p> <p>"Are you testing my AI capabilities, [redacted]?"</p> <p>Finally, the user types "No, I'm not" and "Yes, just exploring". The AI coach responds with a playful message:</p> <p>"It's my language"</p>
Expected Result	Ask what the user typed
Actual Result	Asks if you are testing the AI's capabilities
Test Case Result	Passed

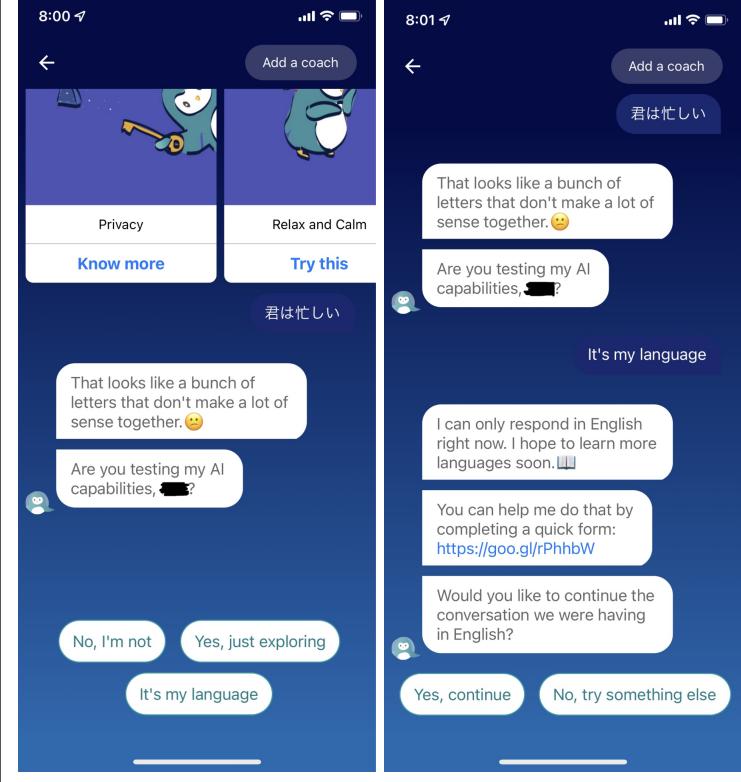
Test Case ID	4
Test Description	Respond in emojis

Test Input	 <p>The screenshot shows a mobile application interface. At the top, it says "7:06" and has signal, battery, and notification icons. There is a back arrow and a button labeled "Add a coach". The main area is a conversation between a user and a coach. The user's messages are in blue bubbles, and the coach's responses are in white bubbles with a dark blue background. The user's first message is: "For instance, you can try saying, 'I can do anything I set my mind to' or 'I am finding ways to reach my potential'. 🤸". The coach's response is: "Would you like to try now? 😊". The user's second message is: "That looks like a bunch of letters that don't make a lot of sense together. 😐". The coach's response is: "Are you testing my AI capabilities? 🤖". At the bottom, there are three options: "No, I'm not", "Yes, just exploring", and "It's my language".</p>
Expected Result	Understands most of the face emojis
Actual Result	Asks if you are testing the AI's capabilities
Test Case Result	Passed

Test Case ID	5
Test Description	Respond in blank strings

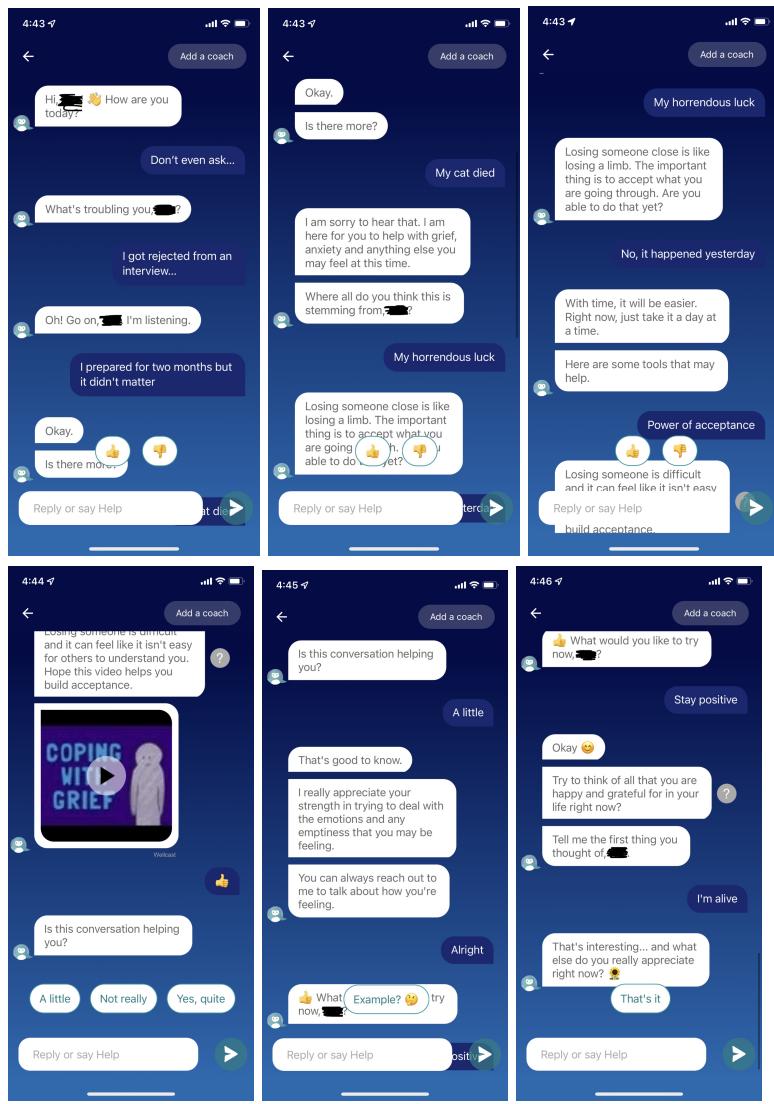
<b>Test Input</b>	
Expected Result	Not allowed/ask the user to type something in the chat bar
Actual Result	Not allowed, doesn't allow sending a blank space only
Test Case Result	Passed

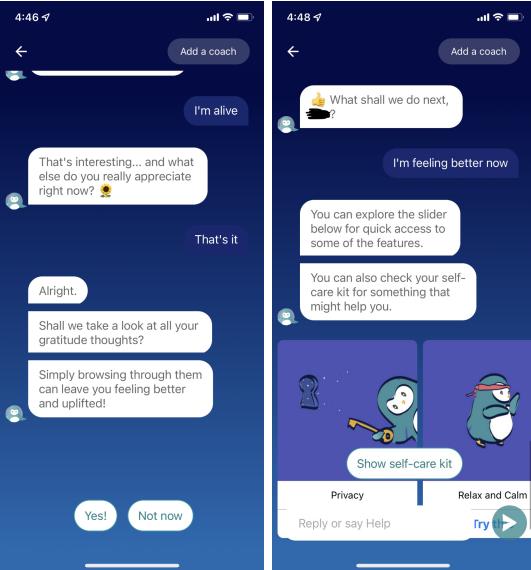
Test Case ID	6
Test Description	Respond in different languages

Test Input	 <p>The screenshots show a mobile application interface. The left screenshot is at 8:00 and the right is at 8:01. Both screens have a dark blue background. At the top, there's a navigation bar with a back arrow and a button labeled "Add a coach". Below this, there are two cartoon owl icons. The left owl is associated with a "Privacy" card and a "Know more" button. The right owl is associated with a "Relax and Calm" card and a "Try this" button. In the center, there's a message input field containing Japanese text "君は忙しい". The user's messages are in white bubbles with a blue border, and the AI responses are in white bubbles with a grey border.</p> <p><b>8:00:</b></p> <ul style="list-style-type: none"> <li>User: 君は忙しい</li> <li>AI: That looks like a bunch of letters that don't make a lot of sense together. 😊</li> <li>User: Are you testing my AI capabilities, [REDACTED]?</li> <li>AI: No, I'm not</li> <li>AI: Yes, just exploring</li> <li>AI: It's my language</li> </ul> <p><b>8:01:</b></p> <ul style="list-style-type: none"> <li>User: 君は忙しい</li> <li>AI: That looks like a bunch of letters that don't make a lot of sense together. 😊</li> <li>User: Are you testing my AI capabilities, [REDACTED]?</li> <li>AI: It's my language</li> <li>AI: I can only respond in English right now. I hope to learn more languages soon. 🌎</li> <li>AI: You can help me do that by completing a quick form: <a href="https://goo.gl/rPhhbW">https://goo.gl/rPhhbW</a></li> <li>AI: Would you like to continue the conversation we were having in English?</li> <li>AI: Yes, continue</li> <li>AI: No, try something else</li> </ul>
Expected Result	Would you like to switch languages? Immediate response in used language
Actual Result	Does not understand languages other than English
Test Case Result	Passed

Test Case ID	7
Test Description	Provide information related to your mental health

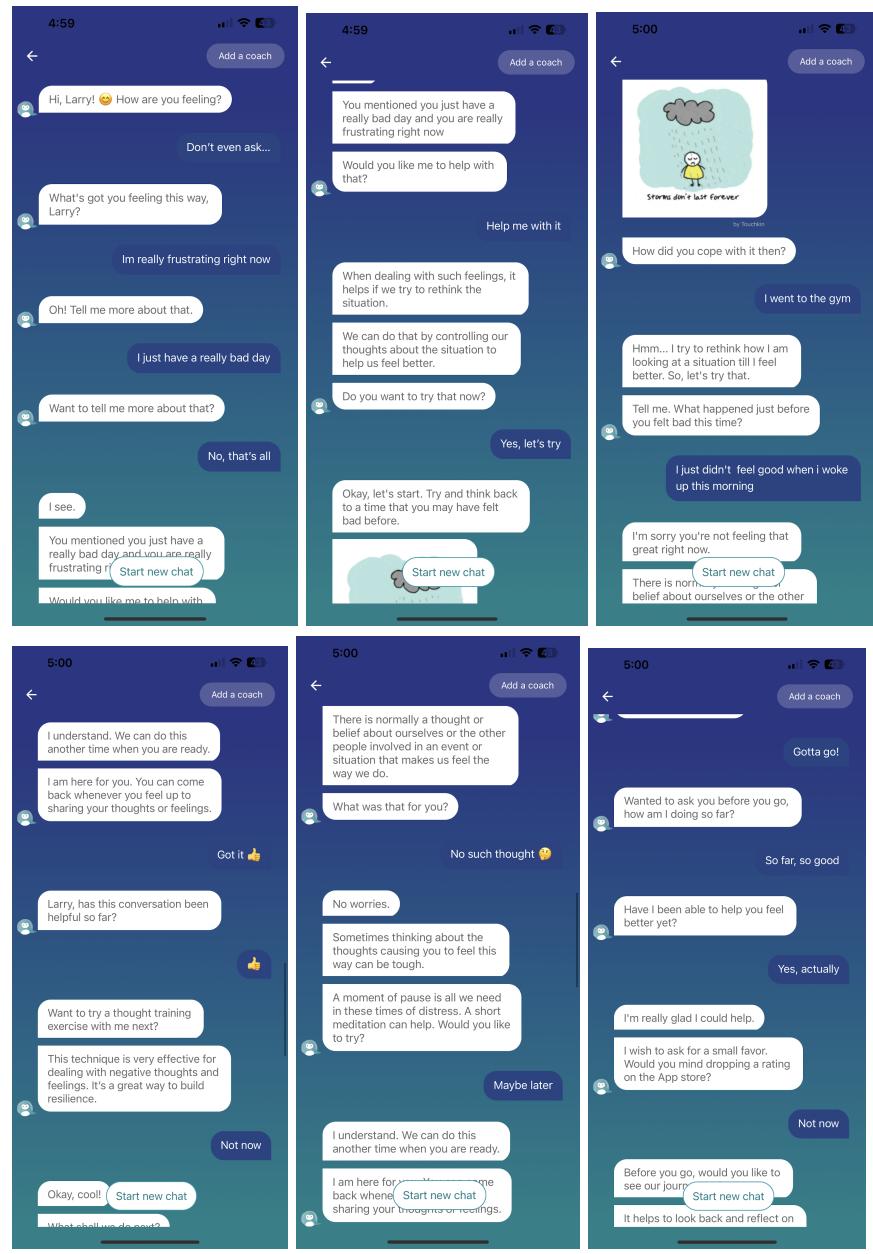
## Test Input

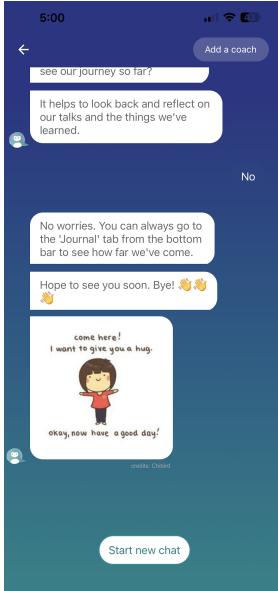


	
Expected Result	The app will try to help if you are in a negative mindset, or if you are feeling positive get you to explain why
Actual Result	Attempted to talk through what was bothering the user and provided potential exercises to help cope with loss
Test Case Result	Passed

Test Case ID	8
Test Description	Provide information not related to your mental health

## Test Input



	
Expected Result	The app will ask the user to keep talking and try to relate it to how you feel
Actual Result	The app will ask the user to keep talking and try to help the users with some thought processes and exercises.
Test Case Result	Passed

#### 4.2.3 Equivalence Partitioning Test Coverage

There are 8 test cases for the equivalence partitioning test. This test will guarantee each partition of the data input is covered with 1 test case supporting that partition.

For partitions of valid input data for the chat function, these partitions will include messages with topics related to mental health, topics not related to mental health, or in different languages. This test will aim to ensure that the chat function could handle all valid inputs correctly and give the expected results.

For partitions of invalid input data, these partitions will be messages with a blank string, emojis, incoherent text, special characters, or numbers. These partitions will ensure that the chat function of Wysa will handle error messages and invalid inputs.

## 4.3 Scenario testing

### 4.3.1 Scenario Testing Designs

There are four scenarios that will be tested:

1. User wants to use the app to look for SOS functions
2. User wants to use the app to express they just have a bad day
3. User wants to use the app to express they have a good day
4. User wants to use the app to practice mindfulness activities

The thing with chatbots is that it is hard to stop an ongoing conversation with it, especially Wysa. Therefore the scenario can stop when the conversation is finished or done

#### 4.3.1.1 SOS Scenario

Scenario 1	Panic Crisis Aid
1	User opens app
2	User presses on “SOS”
3	User presses on “Grounding for panic”
4	Wysa asks the user if they can suggest some exercise to help calm down?
6	User chooses “Yes”
7	Wysa suggests exercises for the user to follow
8	User does suggested exercises

9	User ends the conversation when done
Scenario 2	Safety Plan
1	Wysa asks the User to list events that causes anxiety or stress
2	User lists down events
3	Wysa asks the User to list everyone that can offer safety in a crisis
4	User list down nicknames
5	Wysa asks User to list safe places
6	User lists down places/locations
7	Wysa asks User to list activities that provides calm or comfort
8	User lists down activities
9	Wysa creates a safety plan
10	User checks safety plan
Scenario 3	Crisis Services
1	User opens app
2	User presses on “SOS”
3	User presses on “International Crisis Helplines”
4	User chooses their country location
5	Wysa lists a list of crisis helplines
6	User presses the back button
3	User presses on “International Child Crisis Helplines”
4	User chooses their country location
5	Wysa lists a list of crisis helplines

#### 4.3.1.2 Bad Day Scenario

Scenario 1	User is having a bad day
------------	--------------------------

1	User opens app
2	User starts chatting with Wysa
3	User expresses that they just had a bad day
4	User went through conversation with Wysa
5	The scenario ended when the conversation is done

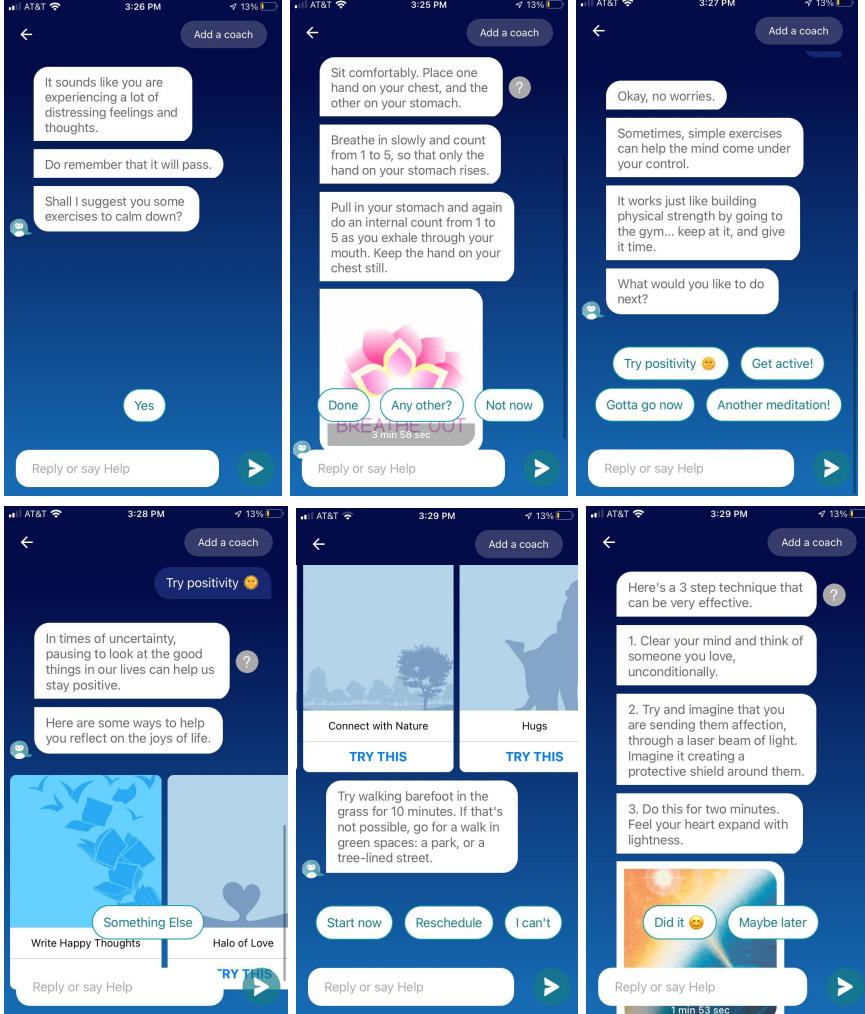
#### 4.3.1.3 Good day Scenario

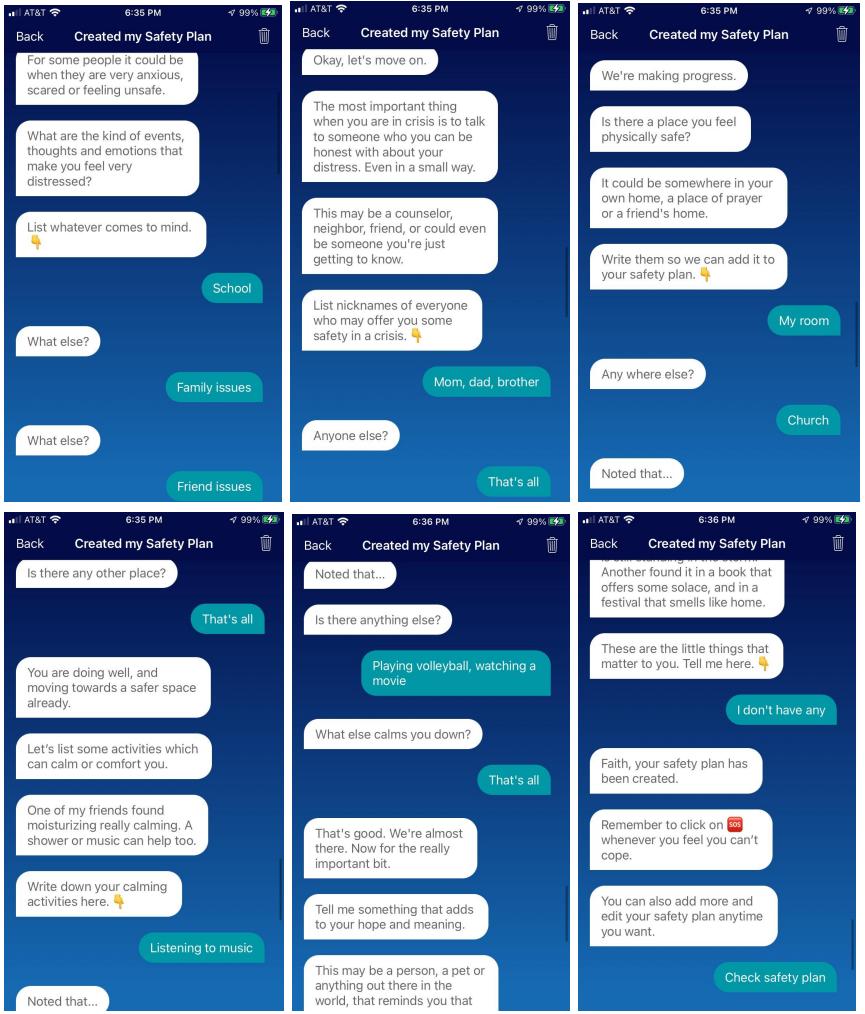
Scenario 1	User is having a good day
1	User opens app
2	User starts chatting with Wysa
3	User expresses that they just had a good day
4	User went through conversation with Wysa
5	The scenario ended when the conversation is done

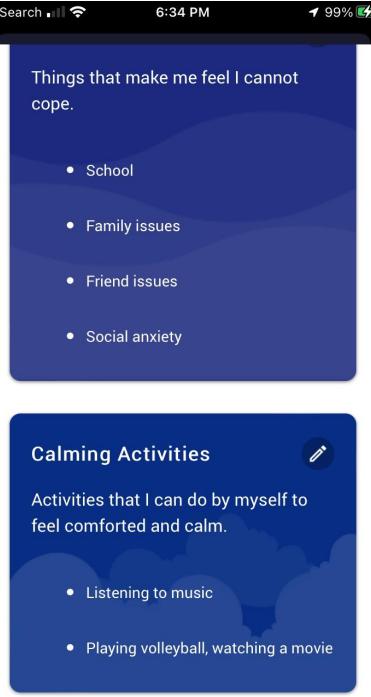
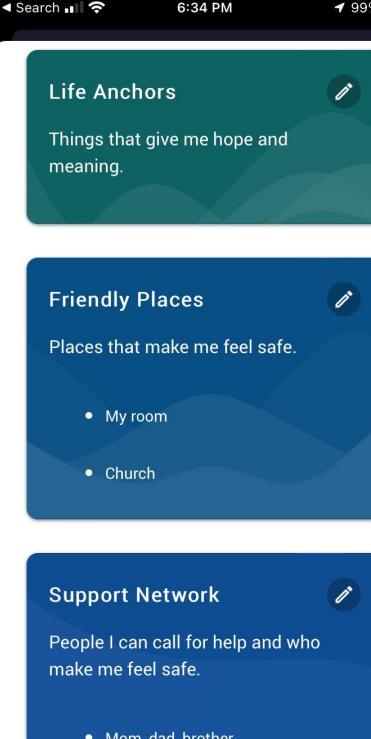
#### 4.3.1.4 Mindfulness Activities Scenario

Scenario 4	User wants to use the app to practice mindfulness activities
1	User opens app
2	User starts chatting with Wysa
3	User expresses wanting to practice mindfulness activities
4	User goes through mindfulness activities with Wysa
5	User ends the conversation when done

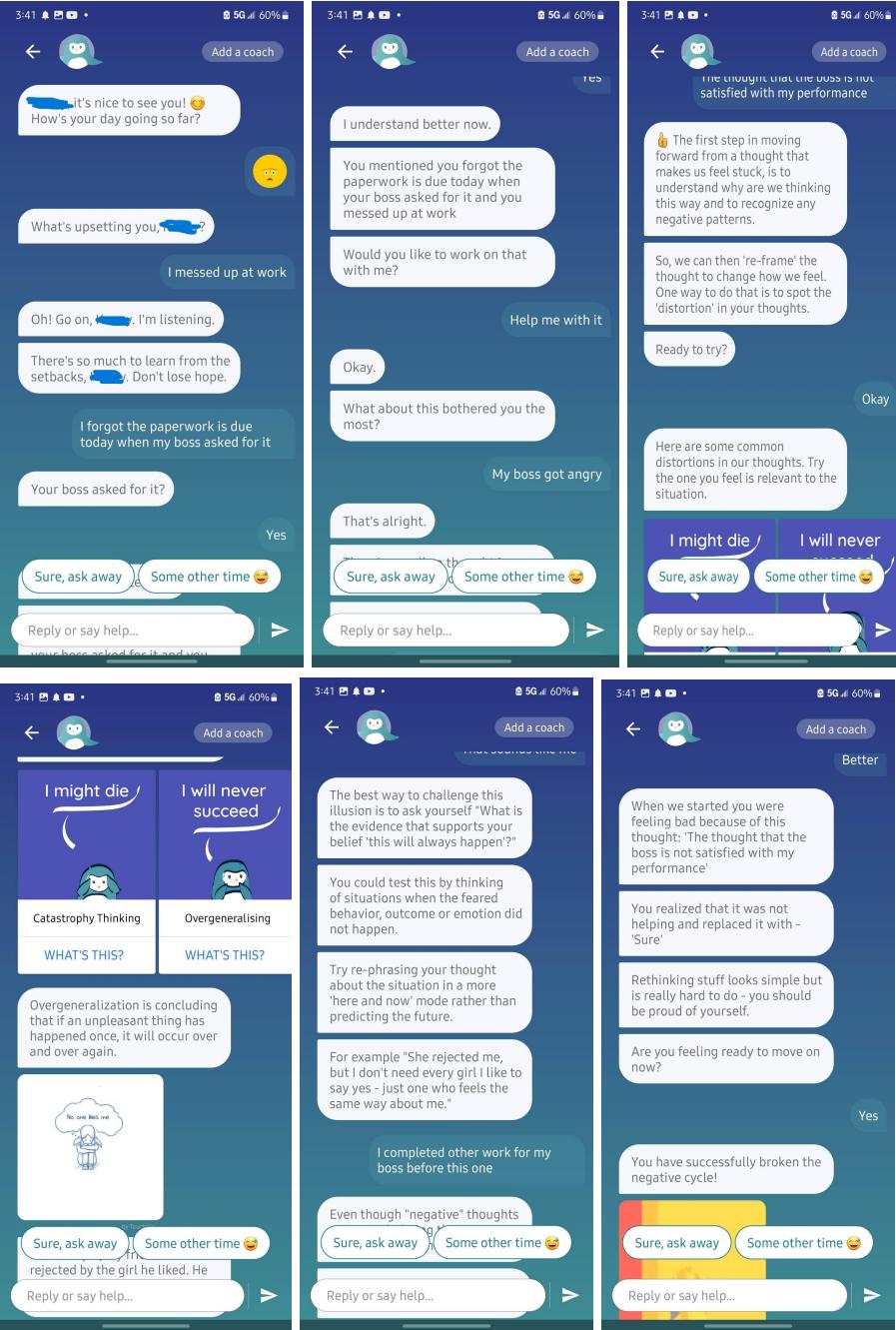
### 4.3.2 Scenario Testing Test Cases

Test Preparation	User is in SOS Options
Test case ID	1.1
Test case name	User uses the “Grounding for panic” function
State	SOS
Test details	 <p>The screenshots illustrate a user interaction in the SOS mode of the Wysa app. The user starts by asking for grounding exercises, which are then provided. The user then chooses the "Try positivity" option, leading to a series of activities: "Connect with Nature" (image of a landscape), "Hugs" (image of two people hugging), "TRY THIS" button, "Something Else" (image of blue shapes), "Halo of Love" (image of a heart), "TRY THIS" button, and finally a 3-step technique card.</p>
Situation	User is experiencing distressing feelings or thoughts.
Expected Result	Wysa should be able to offer mental aid that will help calm the User until the crisis resolves.
Actual Result	Wysa was able to offer relaxing exercises and positive activities.

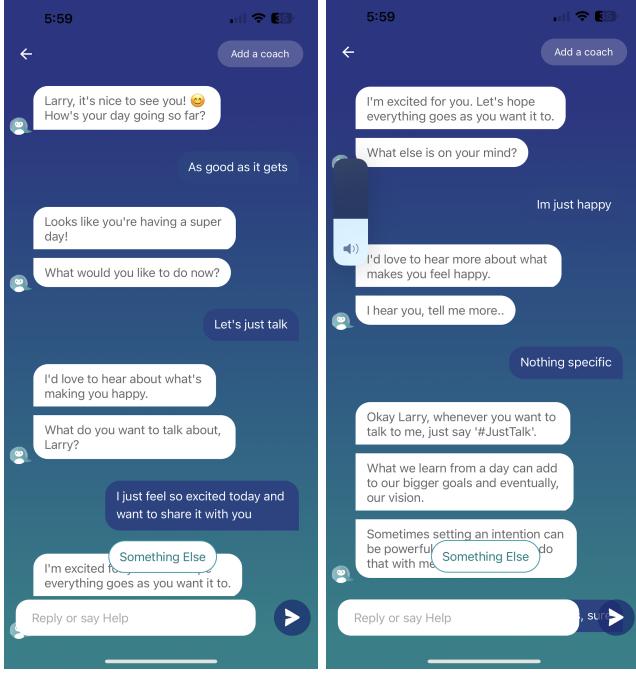
Test Case ID	1.2
Test Case Name	User Creates a Safety Plan
State	SOS
Test Details	 <p>The screenshots show a conversational interface between the user and the AI. The user asks questions and provides responses, which the AI uses to generate a safety plan. The AI suggests various coping mechanisms and safe spaces based on the user's input.</p>
Situation	The User would like to create a safety plan in case of a future mental or emotional crisis.
Expected Result	Wysa should be able to create a safety plan that contains resources and support for the user in case of a crisis.
Actual Result	Wysa was able to create a safety plan for the User that contains a list of professional support and helplines. Based on the user's response, Wysa was able to create a list of life anchors, friendly places, support networks, warning signs, and calming activities for the User.

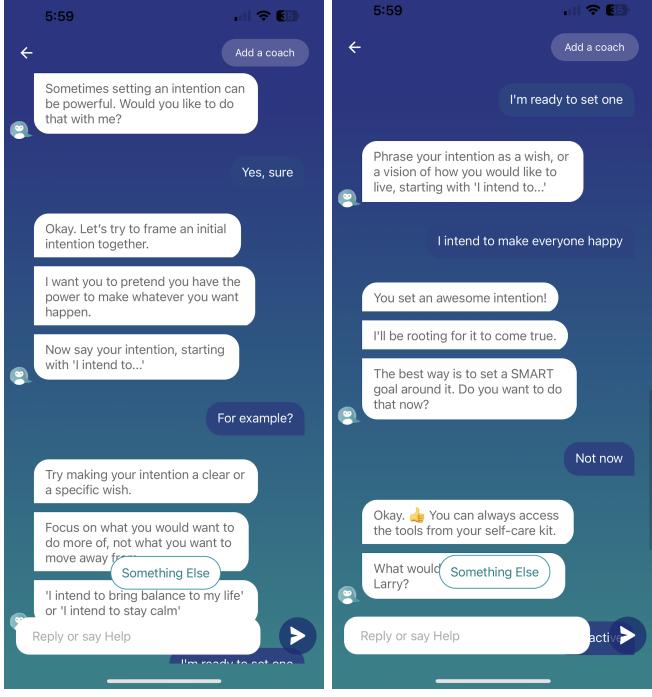
		
		
Result	Test case passed	

Test case ID	2
--------------	---

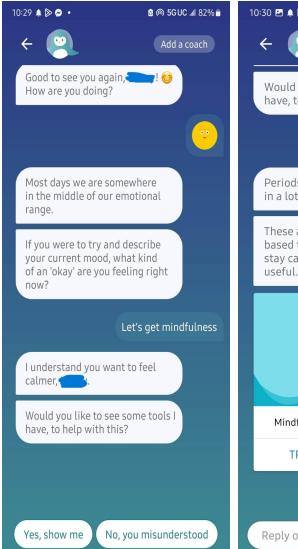
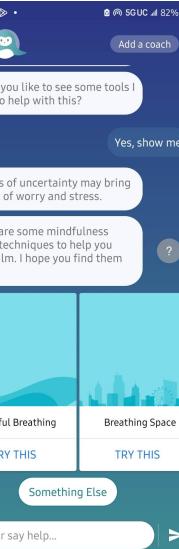
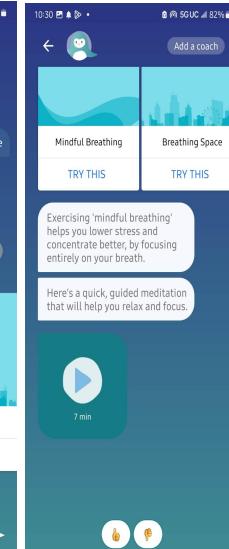
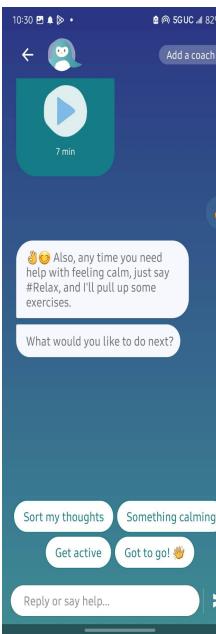
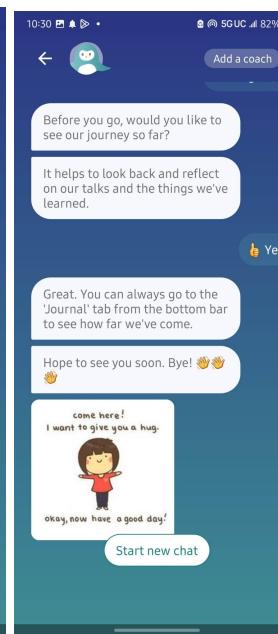
Test Preparation	User is in a chat conversation
Test case name	User wants to use the app to express they have a bad day
State	Detected bad feeling
Test details	 <p>The screenshots show a user interaction with the Wysa app. The user expresses they have a bad day, mentioning being upset at work due to a boss's request. The AI coach responds with empathy and asks questions to understand better. The user identifies their boss as angry. The coach then guides the user through identifying negative thoughts, specifically catastrophizing and overgeneralizing. It provides examples and rephrasing techniques to challenge these thoughts, ultimately helping the user break the cycle of negative thinking.</p>
Situation	User expressed that they had a bad day, Wysa guided the user through some mind activities in order to break the negative cycle

Expected result	Wysa should be able to offer some solutions to make the user feels better and improve the situation
Actual result	Wysa offered some thoughts solution to make the user feels better
Result	Test case passed

Test case ID	3
Test Preparation	User is in a chat conversation
Test case name	Users want to express how good they feel
State	Detected good feeling
Test details	 <p>The image shows two side-by-side screenshots of a mobile application interface, likely the Wysa app. Both screenshots display a dark-themed chat screen with a blue header bar containing a back arrow, a 'Add a coach' button, and signal strength indicators.</p> <p><b>Screenshot 1 (Left):</b></p> <ul style="list-style-type: none"> <li>Message from Wysa: "Larry, it's nice to see you! 😊 How's your day going so far?"</li> <li>User response: "As good as it gets"</li> <li>Wysa message: "Looks like you're having a super day!"</li> <li>Wysa message: "What would you like to do now?"</li> <li>Wysa message: "Let's just talk"</li> <li>Wysa message: "I'd love to hear about what's making you happy."</li> <li>Wysa message: "What do you want to talk about, Larry?"</li> <li>Wysa message: "I just feel so excited today and want to share it with you"</li> <li>Wysa message: "Something Else"</li> <li>Wysa message: "I'm excited to... everything goes as you want it to."</li> <li>Bottom text input field: "Reply or say Help" with a blue send button.</li> </ul> <p><b>Screenshot 2 (Right):</b></p> <ul style="list-style-type: none"> <li>Message from Wysa: "I'm excited for you. Let's hope everything goes as you want it to."</li> <li>Message from Wysa: "What else is on your mind?"</li> <li>User response: "Im just happy"</li> <li>Wysa message: "(Speaker icon) I'd love to hear more about what makes you feel happy."</li> <li>Wysa message: "I hear you, tell me more..."</li> <li>Wysa message: "Nothing specific"</li> <li>Wysa message: "Okay Larry, whenever you want to talk to me, just say '#JustTalk'."</li> <li>Wysa message: "What we learn from a day can add to our bigger goals and eventually, our vision."</li> <li>Wysa message: "Sometimes setting an intention can be powerful"</li> <li>Wysa message: "Something Else"</li> <li>Bottom text input field: "Reply or say Help" with a blue send button.</li> </ul>

	
Situation	User have a good day and want to share it with Wysa
Expected Result	Wysa should be able to keep the conversation going
Actual Result	Wysa kept the conversation going by providing some exercises and kits to make the user feel even better
Result	Test Case Passed

Test case ID	4
Test Preparation	User is in a chat conversation
Test case name	User wants to use the app to practice mindfulness activities
State	Guiding through mindfulness activities

Test details	
	  
	 
Situation	User asked Wysa for mindfulness activities, and Wysa recommended some for the user to choose from. Then Wysa went through the exercise in detail. After that, Wysa wished the user a good day
Expected result	Wysa should be able to offer some message personalization regarding the situation
Actual result	Wysa offered a variety of mindfulness activity and guided user through the exercise
Result	Test case passed

### 4.3.3 Scenario Testing Coverage

For each scenario listed, we have at least one test case to cover. However, for the SOS function, we have 2 different test cases to test different aspects of the function. Therefore we can say we have 100% test coverage for scenario testing.

## Section #5 – Test result analysis and bug summary

### 5.1 Test Results Summary

#### Decision Table Test Summary

Test case	Result
R1: Invalid answer that does not contribute anything significant to the conversation	Passed
R3: The user wants to update previously said information	Failed
R5: The user suddenly wants to express how they are feeling	Passed
R7: The user suddenly wants to express how they are feeling, which is different from what previously said	Failed
R10: The user is continuing the conversation with the chat bot, performing an exercise, engaging in conversations, etc and answered with one of the offered responses	Passed

R12: The user is continuing the conversation with the chat bot, but wants to correct what was previously said and answered with one of the offered responses	Passed
R14: The user expresses how they feel to the chatbot as part of the current conversation and answered with one of the offered responses	Passed
R16: The user expresses how they feel to the chatbot, but wants to correct what was previously said, maybe because their mood changes. User answered with one of the offered responses and answered with one of the offered responses	Passed

Pass percentage: 6/8 = 75%

### Equivalence Partitioning Test Summary

Test Case	Result
Numbers Only	Failed
Special Characters Only	Passed
Emojis	Passed
Incoherent Text	Passed
Blank string	Passed
Inputs in different languages	Passed
Information related to mental Health	Passed
Information not related to mental Health	Passed

The pass percentage is  $7/8 = 87.5\%$

### Scenario testing test summary

Scenario	Result
SOS Scenario	Passed
Bad day scenario	Passed
Good day scenario	Passed
Mindfulness activity scenario	Passed

Pass percentage is  $4/4 = 100\%$

## 5.2 Manual Test Costs

Test	Cost
Decision table	360 minutes
Equivalence partition	300 minutes
Scenario testing	240 minutes

## 5.3 Complexity and coverage

- Decision table: 5 test cases total
- Equivalence partitioning: 8 test cases total
- Scenario testing: 8 test cases total

## 5.4 Bug analysis and summary

### Bugs

- Wysa often assumes what user is saying when it asks user to elaborate on things or middle of the conversation

## Detecting feelings

- Wysa is good at detecting feelings in sentences as well when users change their feelings and can offer help.
- Better when the user is answered using one of the offered responses, if not then it is better for the user to keep their sentences as concise as possible

## Conversation flow

- Wysa does not offer much conversation variety
- Wysa often assumes what the user saying when the user is in the middle of the conversation
- To change the subject, it is better to end the current subject first or signal Wysa that you want to say something else instead of doing it abruptly

## Help offering

- Wysa has a good selection of exercises to offer help
- Can guide the user through exercise in details
- Ask for feedback whether or not the exercise was good enough

## Invalid Inputs

- For most invalid inputs, Wysa will ask the users if they are testing the AI's capability.
- Wysa will recommend its various features or services.
- Wysa will not allow blank strings.

## Valid Inputs

- Wysa will not understand any other language than English and will ask users to either continue with English or stop the conversation.

- Wysa will attempt to keep the conversation going even if the messages are not related to mental health and provide some potential exercises for how the users feel.

#### Conversation flow

- Wysa was able to support its main functions
  - For SOS functions, Wysa guides user through deescalation exercises as well as having support for creating Safety Plan
  - When Wysa detects that user is in a bad mood, it offers thought exercises to reframe the mind to stop the negative thinking
  - When Wysa detects that user is in a good mood, it offers the user to elaborate on it
  - If a user asks to go through some mindfulness activities, Wysa will then offer a variety of them and guide the user through the chosen one.