



Department of Computing

**Development Project
(55-608850)**

Name: Joe Hall

Student ID: 30010893

Degree Route: BEng SE

Supervisor's Name: Cyncia Matsika

Table of Contents

1.	<i>Analysis</i>	3
1.1.	User Stories and Personas	3
1.1.1.	Roles	3
1.1.2.	User Stories	3
1.1.3.	Personas	8
1.1.4.	Requirement Traceability Matrix	13
1.2.	Use Cases	18
1.2.1.	Use Case Diagram	18
1.2.2.	Use Case Descriptions	19
1.3.	MoSCoW Analysis	23
2.	<i>Design</i>	24
2.1.	Class Diagram	24
2.2.	Entity Relationship Diagram	25
2.3.	Sequence Diagram	26
2.4.	Architecture Overview	27
3.	<i>Testing</i>	28
3.1.	Unit Testing	28
3.1.1.	Unit Tests	28
3.1.2.	Unit Test Results	28
3.2.	Integration Testing	29
3.3.	User Acceptance Testing	34
3.3.1.	Questions	34
3.3.2.	Results	34
4.	<i>GitHub Repository</i>	37

1. Analysis

1.1. User Stories and Personas

1.1.1. Roles

Role Name	Description
User	Account that can scan images using image processing and talk to a conversation AI about what is in the image. Users can also access their gallery to view previous image scans and conversations.
Administrator	Account that has administrative privileges over the application. Administrators can remove/create accounts, edit account details and remove gallery image.

1.1.2. User Stories

Title: Logging into the system	Priority: Medium	Estimate: 3 Days
As a Logged-out user.		
I want to Log into the system.		
So that I can utilise the functionality of the application.		
Acceptance Criteria		
Given I am a logged-out user on the login page, and I have an account on the system.		
When I enter the 'Username' and 'Password' credentials for my account.		
Then The system will log me in.		
Requirement Traceability		
Requirement ID – RQ1		
Current Stage of Implementation – TC01 Pass, TC02 Pass, TC03 Pass		

Title: Signing Up	Priority: Medium	Estimate: 2 Days
As a Logged-out user.		
I want to Create an account on the system.		
So that I can utilise the functionality of the application.		
Acceptance Criteria		
Given I am on the sign-in page, and I do not currently have an account on the system.		

When I fill in the 'Username', 'Name', 'Email', 'Password' and 'Re-type password' fields (and the username is not already associated with an account), and I interact with the 'Sign-Up' button.

Then The system will create an account for me and log me in.

Requirement Traceability

Requirement ID – RQ2

Current Stage of Implementation - TC04 Pass, TC05 Pass, TC06 Pass, TC07 Pass

Title: Scanning an image

Priority: High

Estimate: 7 Days

As a User.

I want to Scan an image to analysed and summarised by AI.

So that I can understand what is in the image.

Acceptance Criteria

Given I am a user with a valid OpenAI API key, and I am on the home page.

When I interact with the scan button and take a photo with my camera or upload an image.

Then The system will process the image and take me to a new page that shows the uploaded image as well as a summary of what is in the image.

Requirement Traceability

Requirement ID – RQ3

Current Stage of Implementation - TC08 Pass, TC09 Pass, TC10 Pass, TC11 Pass, TC12 Pass

Title: Conversing with the chatbot

Priority: High

Estimate: 9 Days

As a User.

I want to Have a conversation with the chatbot that has the context of my uploaded image.

So that I can go into more detail about certain aspects of the image rather than just a summary.

Acceptance Criteria

Given I am a user with a valid OpenAI API key, and I have just scanned an image.

When I enter a message in the text input field in the chatbot section of the page and send it.

Then The system will process the message and reply with an answer if the message relates to the image, if the message does not relate to the image, then it will politely decline to respond and prompt the user to ask something related to what is in the image.

Requirement Traceability

Requirement ID – RQ4

Current Stage of Implementation – TC13 Pass, TC14 Pass, TC15 Pass

Title: Viewing an image and related conversation in the gallery

Priority: Low

Estimate: 5 Days

As a User.

I want to Open the gallery section of the application.

So that I can View a previously uploaded image and corresponding chatbot conversation.

Acceptance Criteria

Given I am a user that has previously uploaded an image to the application to be scanned.

When I enter the gallery page of the application and interact with one of the previously uploaded images displayed on the page.

Then The system will open a page displaying information about the scan, such as: the image, the date it was uploaded, the initial summary of the image and the chatbot conversation history.

Requirement Traceability

Requirement ID – RQ5

Current Stage of Implementation – TC16 Pass, TC17 Pass, TC18 Pass

Title: Adding an API key

Priority: High

Estimate: 2 Day

As a User.

I want to Add an OpenAI API key to my account.

So that I can Use all of the functionality of the application.

Acceptance Criteria

Given I have an account on the system.

When I navigate to the 'Account' page from the menu and interact with the 'Edit' button underneath the 'OpenAI API Key' header.

Then The system will prompt me to enter an API key into the text field, from there the save button can be pressed to confirm the API key.

Requirement Traceability

Requirement ID – RQ6

Current Stage of Implementation – TC19 Pass

Title: Deleting your account

Priority: Low

Estimate: 1 Day

As a User.

I want to Delete my account from the system.

So that I can Have my data removed from the system.

Acceptance Criteria

Given I currently have an account on the system.

When I navigate to the 'Account' page from the menu and interact with the 'Delete Account' button underneath the 'Remove Account' header.

Then The system will prompt me to make sure I want to delete my account, once I confirm, the system will remove my account and redirect me to the login page.

Requirement Traceability

Requirement ID – RQ7

Current Stage of Implementation – TC20 Pass

Title: Creating an account

Priority: Low

Estimate: 3 Days

As a Administrator.

I want to Create a new account on the system.

So that I can Add a new account to the system for usability testers to use.

Acceptance Criteria

Given I am logged into the system using an Administrator account.

When I navigate to the 'Settings' page from the menu and interact with the 'Add User' button and fill in the 'Name', 'Username', 'Password', 'Re-type password' and 'Role' fields.

Then The account will be created on the system.

Requirement Traceability

Requirement ID – RQ8

Current Stage of Implementation – TC21 Pass

Title: Deleting an account

Priority: Low

Estimate: 2 Days

As a Administrator.

I want to Delete an account on the system.

So that I can Remove an account that is no longer needed.

Acceptance Criteria

Given I am logged into the system using an Administrator account.

When I navigate to the 'Settings' page from the menu and search for a user and then interact with the 'Delete Account' button.

Then The account and all of its associated data will be deleted from the system.

Requirement Traceability

Requirement ID – RQ9

Current Stage of Implementation – TC22 Pass

1.1.3. Personas

1.1.3.1. User

Jordan Clark



AGE	23
WORK	University Student
FAMILY	Lives with parents and a younger sister
LOCATION	Glasgow, Scotland

“ Vision does not define us, it is our ability to adapt and overcome that shapes our lives.

Personality

Empathetic Resilient

Intuitive Charismatic

Bio

Jordan lost his sight at the age of 15 due to a rare genetic condition. He quickly adapted to his new way of living with the help of technology and a strong support network. He's currently pursuing his bachelor's degree with a focus on social behavior and inclusion. Jordan is an active member of his university's accessibility committee and plays the keyboard in a band with his friends. He uses technology to aid his studies and social interactions.




Goals

- To graduate with honours and pursue a career in social work.
- To develop a campus community program that pairs sighted students with visually impaired students for mutual learning and support.




Frustrations

- Difficulty in accessing textbooks and academic resources that are not available in accessible formats.
- Apps and websites with poor accessibility features that hinder his academic and personal activities.

Brands



Preferred Channels



Voice Assistant Mobile App Social Media

Scenario 1:


Jordan Clark is university student at The University of Glasgow in Scotland. Jordan lost his sight at the age of 15 and since has adapted to the new lifestyle that being blind entails. Jordan is always looking to use new technology to help him navigate his way through life and is optimistic about recent advances in artificial intelligence. Jordan hears about a new website that allows for ai image processing using only your phone camera, this allows visually impaired users to get a summary of what is in the image using generative artificial intelligence. Jordan accesses the website using his phone and is happy to know that the website is built with accessibility in mind as it works with his phones inbuilt screen reader. Jordan creates an account on the website and is greeted by the home page where he uses his phones camera to capture the environment

around him. He presses the results button and is presented with an AI summary of the image; he taps on the speaker icon to hear the summary out loud using text to speech technology.

Scenario 2:

Jordan Clark is university student at The University of Glasgow in Scotland. Jordan lost his sight at the age of 15 and since has adapted to the new lifestyle that being blind entails. Jordan has recently been using AI Assistant for the Visually Impaired to help him identify objects and scenery in the world around him. Jordan is out at a social gathering with his friends and decides to show them some of the photos he scanned a couple of days ago, he goes to login to the website and after logging in successfully he navigates to the gallery page. From there Jordan can go through all his past image scans and show them to his friends.

Alex Jensen



AGE	29
WORK	Software Developer
FAMILY	Single
LOCATION	San Francisco, California

“ Technology should bridge gaps, not create them. I'm fascinated by tools that bring us closer to one another, regardless of our abilities.

Personality

PragmaticInnovativeInnovativeFriendly

Bio

Alex is a software developer at a tech startup known for its inclusive software design. Although not visually impaired, Alex is deeply interested in assistive technologies and their potential to enhance the lives of those who have disabilities. Driven by a passion for universal design principles, Alex volunteers at local coding bootcamps teaching accessible web development. In leisure, Alex enjoys hackathons, gaming, and exploring VR technologies.




Goals

- To stay at the forefront of emerging accessibility technologies and advocate for their adoption.
- To learn new programming languages and technology.




Frustrations

- Apps and websites that are not designed with accessibility in mind.
- The slow pace at which some companies adopt inclusive design practices.

Brands



Preferred Channels



Video GamesMobile AppMessage

Scenario 1:

Alex Jensen is a software developer at a tech start-up in San Francisco, California. Alex is not herself visually impaired or blind, however she is very interested in new emerging technologies,


and she is particularly keen at the recent advances in AI technology. Alex has heard of a new web application that uses image analysis to help visually impaired users to identify things in the world around them. Alex goes to the webpage of the image analysis tool to sign up for an account. She proceeds to fill out the sign-up form and continue to the application. Alex doesn't notice that the website is built to prioritise accessibility because she isn't using a screen reader however, she notices on the About page of the website that this is the case. Alex goes to the applications home page to try the application for herself, she scans an image that is saved on her laptop and proceeds to press the results button. She is pleased to see that the AI summary that is generated using the image has a feature to allow for text to speech.

Scenario 2:

Alex Jensen is a software developer at a tech start-up in San Francisco, California. Alex is not herself visually impaired or blind, however she is very interested in new emerging technologies, and she is particularly keen at the recent advances in AI technology. Alex has recently signed up for a new application that allows blind and visually impaired users to understand the world around them using artificial intelligence. She decides that she wants to show this application to some of her fellow software developers, she opens the gallery page on the website and scrolls down to an image she scanned a few days prior. She interacts with the image and shows the image and chatbot history to her colleague.

1.1.3.2. Administrator

David Bennett



AGE	42
WORK	Accessibility Consultant
FAMILY	Single, lives with a guide dog named Max
LOCATION	Leeds, England

“ Disability is not an obstacle to success. It's a matter of embracing your abilities and finding the way that works for you.

Personality

Proactive Sociable

Patient Friendly

Bio

David was born with retinitis pigmentosa, progressively losing his vision since childhood. He has adapted well to his condition, thanks to his proactive nature and the support from his community. David works to make digital content more accessible for people with disabilities. In his free time, he enjoys listening to audiobooks and is an avid swimmer. He's also active in advocacy groups for the visually impaired.




Goals

- To ensure digital platforms are universally accessible and user-friendly for people with visual impairments.
- To lead workshops that empower visually impaired individuals to use technology effectively.




Frustrations

- Apps and websites that are not designed with accessibility in mind.
- The general lack of awareness about the capabilities of visually impaired individuals.

Brands



Preferred Channels



Voice Assistant Mobile App Podcast

Scenario 1:

David Bennett is an accessibility consultant in Leeds, England. David was born with retinitis pigmentosa, a disease which progressively deteriorates his vision, this has left him blind for most of his adult life. Due to his disability, he has been inspired to pursue new and emerging accessibility technologies that can help him and others in day-to-day life; this led David to becoming an administrator for the AI Assistant for the Visually Impaired website. He is very excited about recently becoming an administrator for the site, so he decides to present a demo of the application to the people that attend his workshops. He logs into his account on the site and navigates to the settings tab in the menu, from there he is greeted with the administrator settings page. He uses his laptops inbuilt screen reader to identify where the 'Add User' button is and proceeds to create temporary accounts for the people in his workshop so they can also test the application before creating their own accounts.

Scenario 2:

David Bennett is an accessibility consultant in Leeds, England. David was born with retinitis pigmentosa, a disease which progressively deteriorates his vision, this has left him blind for most of his adult life. After a recent workshop that David led where he created temporary accounts for other visually impaired people in the workshop, he notices that they have all created their own accounts on the system. David then decides that he may as well remove the temporary accounts from the system as they are no longer in use. David navigates to the settings page from the menu, then he finds the accounts that he created from the list of user accounts presented to him. David goes through and removes all the accounts that are no longer in use.

1.1.4. Requirement Traceability Matrix

Requirement ID:	RQ1
Requirement Priority:	Medium
Description:	Users should be able to log into the system so that they can utilise the applications functionality.
Rationale:	The user of the system will be able to log into the system from a login screen that will check their credentials against others in the system in order to identify them.
Inputs:	Once the user accesses the website via a web browser, they will be able to input their username and password into the input fields, they can then press the blue Login button.
Outputs:	The user will be logged into their account if their credentials are associated with an account on the system. If the credentials are incorrect, they will be given an error message.
Persistent Changes:	No persistent changes for this output.
Test Case ID:	TC01, TC02, TC03
Status:	TC01 – Pass TC02 – Pass TC03 - Pass

Requirement ID:	RQ2
Requirement Priority:	Medium
Description:	Users should be able to create an account on the system by signing up with a name, email, username, and password.
Rationale:	The user of the system will be able to create an account on the system from the sign-up page.
Inputs:	Once the user has accessed the website, they will be able to interact with the sign-up button from the login page. They will then be directed to the sign-up page where they can input a name, email, username, and password.
Outputs:	The user will then be redirected to the login page where they can input the username and password for the account that has just been created for them on the system.
Persistent Changes:	A new account will be added to the systems database containing the user's credentials.
Test Case ID:	TC04, TC05, TC06, TC07
Status:	TC04 – Pass TC05 – Pass TC06 – Pass

	TC07 - Pass
--	-------------

Requirement ID:	RQ3
Requirement Priority:	High
Description:	Users should be able to upload an image from their device so that it can be analysed by the system.
Rationale:	The user of the system will be able to upload an image so that it can be analysed using AI and return a summary of the image.
Inputs:	Once the user is logged into the system and on the home page, they will be able to interact with the scan button in the centre of the page, they will then be prompted to upload an image.
Outputs:	The user will be redirected to a scan results page which will show them the image they just uploaded, along with a title for the image, an AI summary of the image, a chatbot to converse with and a text-to-speech button.
Persistent Changes:	A new conversation object will be added to the database containing the title, the file path of the image, the file path of the text-to-speech audio, AI summary of the image and the date the conversation took place.
Test Case ID:	TC08, TC09, TC10, TC11, TC12
Status:	TC08 – Pass TC09 – Pass TC10 – Pass TC11 – Pass TC12 – Pass

Requirement ID:	RQ4
Requirement Priority:	High
Description:	Users should be able to converse with a chatbot to ask more in-depth questions about the uploaded image.
Rationale:	The user of the system will be able to converse with chatbot that has the context of the image they uploaded to the system, allowing for more in-depth descriptions of specific parts of the image.
Inputs:	Once the user has uploaded an image to be analysed by the system, they will be able to interact with the chatbot on the scan results page. At the bottom of the chatbot container there is an input field which the user can input a question into. The user can then send the message by pressing enter.

Outputs:	The chatbot container will display a response to the user's question using AI, underneath the message there is also a button that allows the user to hear the response using text-to-speech.
Persistent Changes:	Two new message objects will be created and added to the database. One for the user's message and one for the AI response.
Test Case ID:	TC13, TC14, TC15
Status:	TC13 - Pass TC14 - Pass TC15 - Pass

Requirement ID:	RQ5
Requirement Priority:	Low
Description:	Users should be able to access a gallery of all their previously uploaded images.
Rationale:	The user of the system will be able to access the gallery page to view their previously uploaded images and chatbot conversations.
Inputs:	Once the user is logged into the system and has previously scanned an image using the system, they will be able to navigate to the Gallery tab. A list of their previous image scans will appear on the screen as well as a title for the image. The user can then click on one of the images in order to see more details.
Outputs:	The application will display more details about the conversation such as the date of the conversation, the initial image summary, and any messages with the chatbot.
Persistent Changes:	No persistent changes for this output.
Test Case ID:	TC16, TC17, TC18
Status:	TC16 - Pass TC17 - Pass TC18 - Pass

Requirement ID:	RQ6
Requirement Priority:	High
Description:	Users should be able to add an OpenAI API key to their account.
Rationale:	The user of the system will be able to add an OpenAI API key to their account in order to use the full functionality of the system.
Inputs:	Once the user is logged into the system and navigates to the Account tab, they will be greeted with the account page. From there they will be able to interact with the Edit button underneath the OpenAI API Key header.

Outputs:	The application will present the user with an input field where they can enter an API key.
Persistent Changes:	A new API key object will be added to the database that is linked to the user's account.
Test Case ID:	TC19
Status:	TC19 – Pass

Requirement ID:	RQ7
Requirement Priority:	Low
Description:	Users should be able to delete their account on the system.
Rationale:	The user of the system will be able to delete their account on the system as well as all their saved data.
Inputs:	Once the user is logged into the system and have navigated to the Account tab, they will interact with the red Remove Account button. The system will then display a pop-up screen to confirm the account deletion.
Outputs:	Once the user has confirmed the account deletion their account will be deleted from the system, and they will be redirected to the login screen.
Persistent Changes:	The users account will be deleted from the database as well as their API key and any conversations and associated messages.
Test Case ID:	TC20
Status:	TC20 – Pass

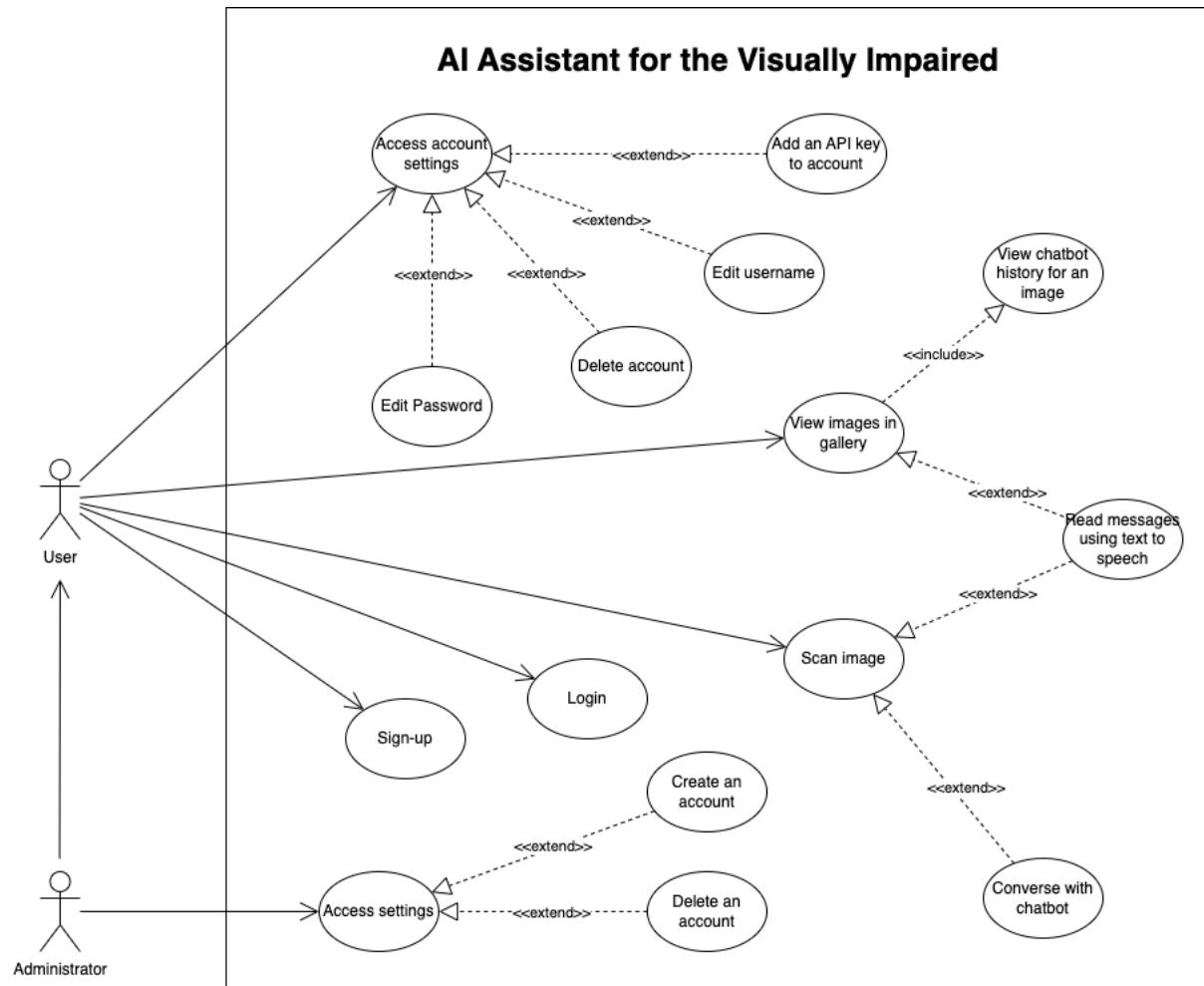
Requirement ID:	RQ8
Requirement Priority:	Low
Description:	Administrators should be able to create new accounts on the system.
Rationale:	The administrator of the system will be able to create new accounts on the system for testing purposes.
Inputs:	Once the administrator is logged into the system and navigated to Settings tab the system will display an administrator only settings page. From this page they can interact with the Add User button, a pop-up window will appear, allowing the administrator to input account details. Then the administrator will press the blue Create Account button.
Outputs:	A new account will be added to the system, this will be apparent as the new account will be added to the list of accounts on the system on the administrator only settings page.
Persistent Changes:	A new account will be saved to the database.

Test Case ID:	TC21
Status:	TC21 – Pass

Requirement ID:	RQ9
Requirement Priority:	Low
Description:	Administrators should be able to delete accounts on the system.
Rationale:	The administrator of the system will be able to delete accounts on the system, such as accounts used for testing purposes.
Inputs:	Once the administrator is logged into the system and navigates to the Settings tab, the system will display the administrator only settings page. From this page they will be able to see a list of every account on the system, the administrator will search for the account they want to delete and then interact with the red bin icon next to the account. The browser will then display a pop-up window to confirm the deletion of the account, the administrator can then press OK to delete the account.
Outputs:	Once the administrator has deleted the account the account will be removed from the list of accounts on the system.
Persistent Changes:	The account will be removed from the database along with any associated API key, conversations, and messages.
Test Case ID:	TC22
Status:	TC22 – Pass

1.2. Use Cases

1.2.1. Use Case Diagram



1.2.2. Use Case Descriptions

Sign-up	
Actors	User
Pre-Conditions	User does not already have an account on the system.
Primary Path	<ol style="list-style-type: none">1. User navigates to the applications 'Login' page.2. User interacts with the 'Sign Up' button at the bottom of the page.3. The user input the necessary details into the 'Name', 'Username', 'Email', 'Password' and 'Re-type password' fields.4. The user interacts with the 'Sign Up' button.5. The system creates an account for the user and saves the details into the database.6. Use Case ends with Success
Post-Conditions	User has an account on the system.
Alternative Path	4a. System already has an account with the same username. <ol style="list-style-type: none">1. The system displays an error message indicating the username is already taken and prompting the user to try again.2. Continue from step 3. of Primary Path.

Login	
Actors	User, Administrator
Pre-Conditions	The actor already has an account on the system.
Primary Path	<ol style="list-style-type: none">1. Actor navigates to the applications 'Login' page.2. Actor enters in their credentials into the 'Username' and 'Password' fields.3. The actor then interacts with the 'Login' button.4. The system takes the inputted credentials and verifies that they match an account in the database.5. Use Case ends with success
Post-Conditions	Actor logs into the system successfully.
Alternative Path	3a. System doesn't recognise an account in the database with the username and password combination. <ol style="list-style-type: none">1. The system displays an error message to indicate the username or password does not match an account on the system and prompts the actor to try again.2. Continue from step 2. of Primary Path.

Scan Image	
Actors	User, Administrator
Pre-Conditions	The actor has an account on the system, and a valid OpenAI API key associated with the account, and is logged in.
Primary Path	<ol style="list-style-type: none">1. Actor navigates to the 'Home' page.2. Actor interacts with the button in the middle of the screen that reads 'Tap to scan image with your camera'.3. Actor uses their devices camera to take a photo to be scanned and interacts with the 'Results' button.4. The system uses the OpenAI API to analyse the image.5. The system displays a page with the results of the scan including, the image, the AI generated summary of the image and a chatbot that allows the user to ask more questions.

	6. Use Case ends with success
Post-Conditions	Actor successfully scans an image.
Alternative Path	<p>3a. Error returning results from OpenAI API.</p> <ol style="list-style-type: none"> 1. The system displays an error informing the actor that there has been an issue connecting to the external API. 2. Use Case ends in failure. <p>5a. Having a conversation with the chatbot.</p> <ol style="list-style-type: none"> 1. The actor inputs a prompt for the chatbot, into the input field, regarding the image. 2. The chatbot responds with a response to the actor's question using generative AI. 3. Use Case ends with success. <p>5b. System reading message out loud using Text to Speech.</p> <ol style="list-style-type: none"> 1. The actor interacts with the speaker icon located underneath the chatbot conversation message. 2. The system uses the Text to Speech to read the message out loud. 3. Use Case ends with success.

View Image in Gallery	
Actors	User, Administrator
Pre-Conditions	The actor has an account on the system and is logged in.
Primary Path	<ol style="list-style-type: none"> 1. Actor interacts with the menu icon to the right on the header. 2. Actor interacts with the 'Gallery' menu item. 3. The system displays the actor's Gallery. 4. The actor interacts with an image tile. 5. The system displays a page including the image, the date the image and conversation took place, the AI generated summary and the chatbot history. 6. Use Case ends with success.
Post-Conditions	Actor successfully views gallery image.
Alternative Path	<p>3a. System cannot find any previous images.</p> <ol style="list-style-type: none"> 1. The system cannot display any previously scanned images on the actor's account. 2. The system displays a message indicating there are no previously scanned images. 3. Use Case ends in failure. <p>5a. System reading message out loud using Text to Speech.</p> <ol style="list-style-type: none"> 1. The actor interacts with the speaker icon located underneath the chatbot conversation message. 2. The system uses Text to Speech to read the message out loud. 3. Use Case ends with success.

Access Account Settings	
Actors	User, Administrator
Pre-Conditions	The actor has an account on the system and is logged in.
Primary Path	<ol style="list-style-type: none"> 1. Actor interacts with the menu icon to the right on the header. 2. Actor interacts with the 'Account' menu item. 3. The system displays the Account page for the actor logged in.

	4. Use Case ends with success.
Post-Conditions	Actor successfully accesses the account settings.
Alternative Path	<p>3a. Actor adds API key to their account.</p> <ol style="list-style-type: none"> 1. Actor interacts with the 'Edit' button underneath the 'OpenAI API Key' header. 2. The system displays an input field for the actor to input their API key. 3. The actor interacts with the save button. 4. Use Case ends with success. <p>3b. Actor edits their username.</p> <ol style="list-style-type: none"> 1. Actors interacts with the 'Edit' button underneath the 'Username' header. 2. The system displays an input field for the actor to input their new username. 3. The actor interacts with the save button. 4. Use Case ends with success. <p>3c. Actor edits their password.</p> <ol style="list-style-type: none"> 1. Actor interacts with the 'Edit' button underneath the 'Password' header. 2. The system displays an input field for the actor to enter their new password. 3. The actor interacts with the save button. 4. Use Case ends with success. <p>3d. Actor deletes their account.</p> <ol style="list-style-type: none"> 1. Actor interacts with the red 'Delete Account' button underneath the 'Remove Account' header. 2. The system displays a screen asking if the actor is sure they want to delete their account and informing them that it involves erasing all the accounts data. 3. The actor interacts with the 'Delete' button. 4. Actor is logged out of the system and the account, as well as all the data associated with it, is removed from the database. 5. Use Case ends with success.

Access Settings	
Actors	Administrator
Pre-Conditions	The administrator is logged in.
Primary Path	<ol style="list-style-type: none"> 1. Administrator interacts with the menu icon to the right on the header. 2. Administrator interacts with the 'Settings' menu item. 3. The system displays the settings page. 4. Use Case ends with success.
Post-Conditions	Administrator successfully accesses the settings.
Alternative Path	<p>3a. Creating a new account on the system.</p> <ol style="list-style-type: none"> 1. Administrator interacts with the blue 'Add user' button underneath the 'Settings' header.

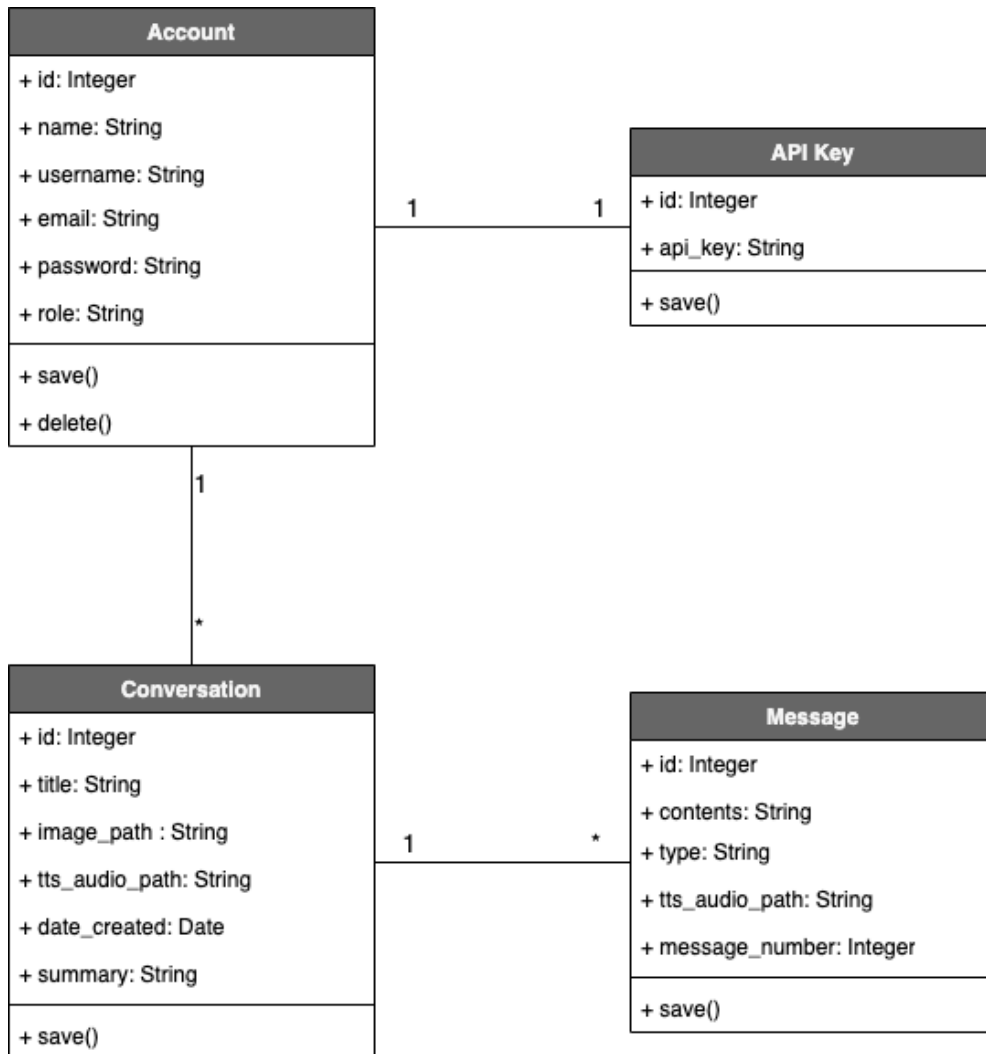
	<ol style="list-style-type: none">2. The system displays a page allowing the administrator to fill in the 'Name', 'Username', 'Email', 'Password', 'Re-type password' and 'Role' input fields.3. The administrator confirms the new account and interacts with the 'Create' button.4. A new account is created on the system.5. Use Case ends with success. <p>3b. Deleting an account on the system.</p> <ol style="list-style-type: none">1. Administrator locates the account they want to remove from the system and interacts with the 'Delete Account' button.2. The system prompts them to confirm if they want to remove the account and all associated data.3. The Administrator interacts with the 'Delete' button.4. The system removes the account and its data from the database.5. Use Case ends with success.
--	--

1.3. MoSCoW Analysis

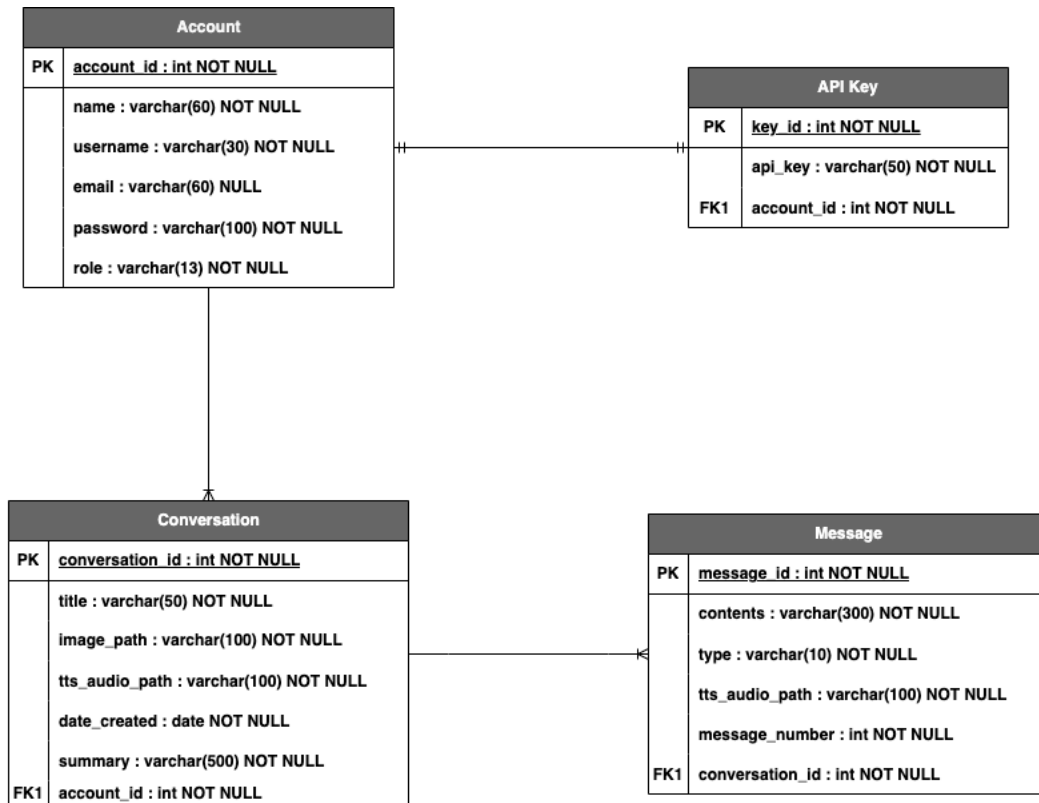
Must Have	Should Have
<ul style="list-style-type: none">• Sign-up/Login/Create an account: Essential for using the application.• Add API Key: Required functionality to allow for image scanning and generative AI chatbot.• Image scanning functionality: Core functionality for a visually impaired user to interact with the application.• Account settings: Essential to add an API key to the users account.	<ul style="list-style-type: none">• Read message aloud using Text to Speech: Crucial for accessibility however most devices do include screen reading technology.• Conversing with the chatbot: Functionality that should be in the application, however it would still function without it.• Gallery image view: Functionality that should be available in the first release of the application
Could Have	Won't Have (for initial release)
<ul style="list-style-type: none">• Edit password/Edit username: Important for account security and personalisation but not necessarily critical functionality.• Delete account: Important but essential for functionality.• Administrator system settings: Ideal for Administrator accounts to manage the system, however not required functionality.	<ul style="list-style-type: none">• Speech to Text: Functionality that would be great for accessibility however it cannot be added in the projects initial time frame.• Social media integration: Functionality that would be a welcome feature in a future release but not necessary for the initial release.• Multilingual support: Useful functionality that would broaden the audience able to use the application.

2. Design

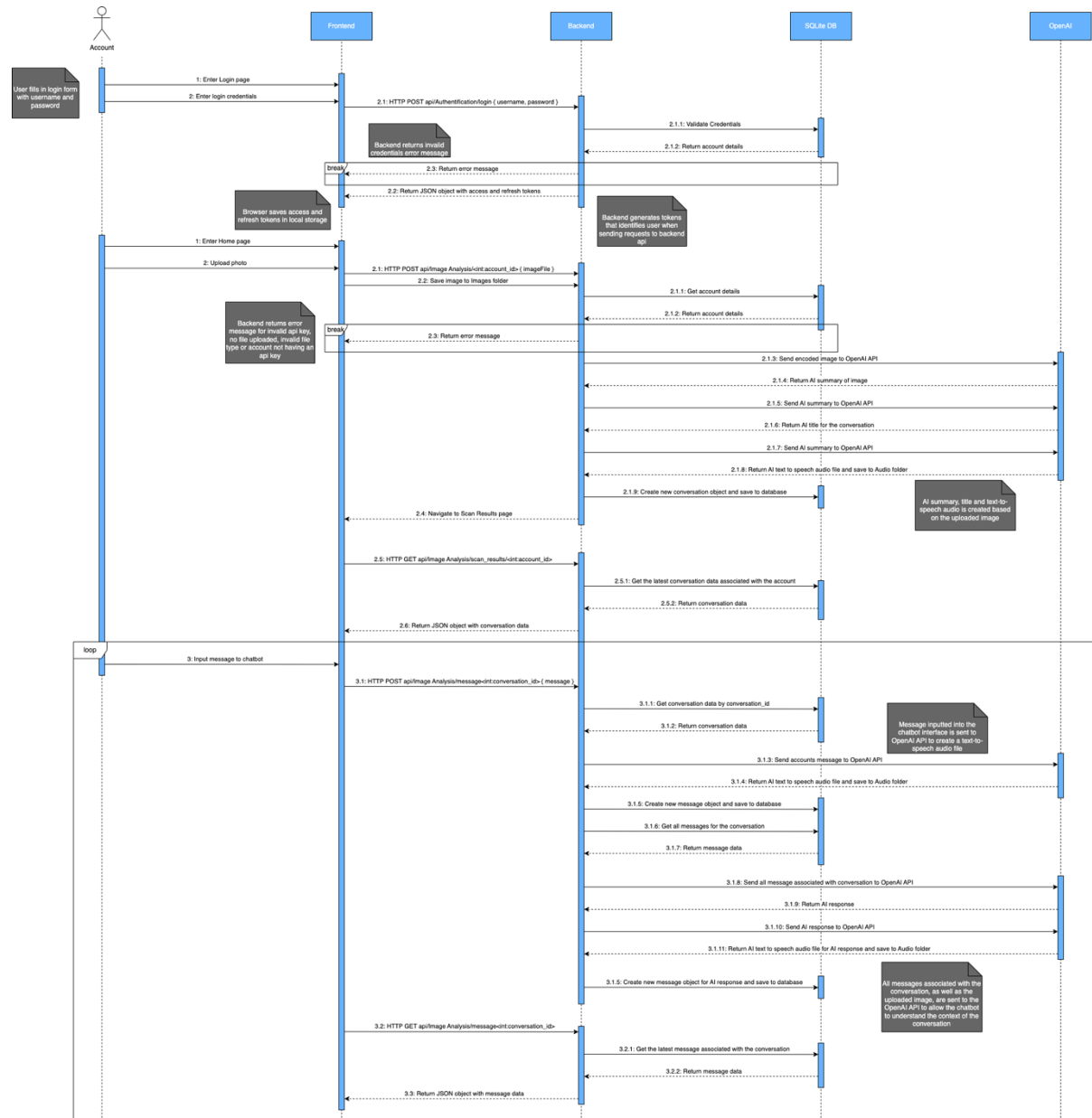
2.1. Class Diagram



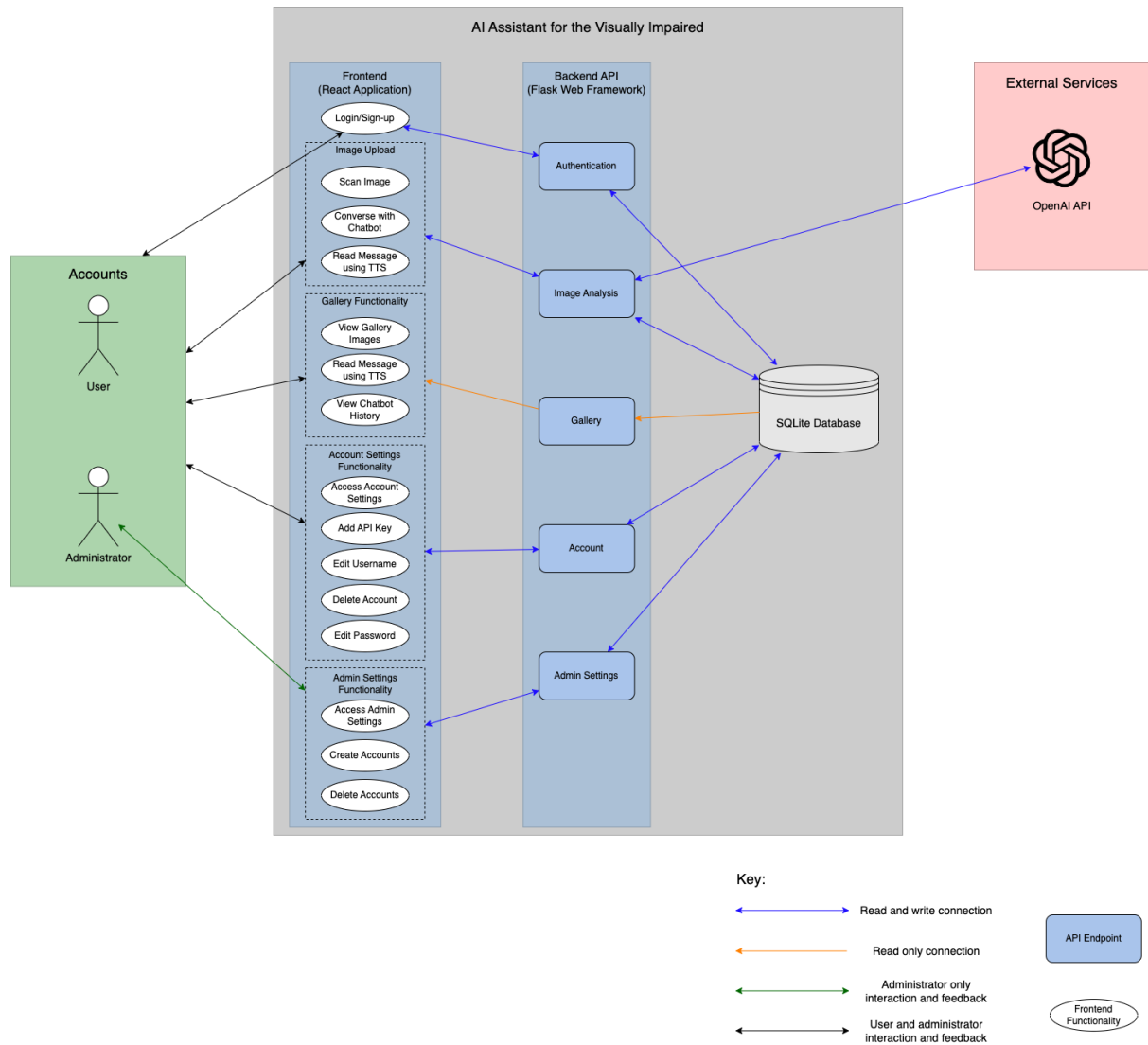
2.2. Entity Relationship Diagram



2.3. Sequence Diagram



2.4. Architecture Overview



3. Testing

3.1. Unit Testing

3.1.1. Unit Tests

```
backend > test_api.py > APITestCase > test_login
1 import unittest
2 from main import create_app
3 from config import TestConfig
4 from exts import db
5
6 class APITestCase(unittest.TestCase):
7     def setUp(self):
8         self.app = create_app(TestConfig)
9         self.client = self.app.test_client(self)
10
11         with self.app.app_context():
12             db.create_all()
13
14     # AUTHENTICATION TESTS
15
16     # Test the signup endpoint
17     def test_signup(self):
18         signup_response=self.client.post('/Authentication/signup', json={
19             "name": "Test User",
20             "username": "test_user123",
21             "email": "test@gmail.com",
22             "password": "password123"
23         })
24         self.assertEqual(signup_response.status_code, 201)
25
26     # Test the login endpoint
27     def test_login(self):
28         signup_response=self.client.post('/Authentication/signup', json={
29             "name": "Test User",
30             "username": "test_user123",
31             "email": "test@gmail.com",
32             "password": "password123"
33         })
34         login_response=self.client.post('/Authentication/login', json={
35             "username": "test_user123",
36             "password": "password123"
37         })
38         self.assertEqual(login_response.status_code, 200)
39
```

3.1.2. Unit Test Results

```
PROBLEMS 18 OUTPUT TERMINAL PORTS DEBUG CONSOLE
./Users/joehall/Coding/AAVI/AAVI/backend/api/account.py:145: LegacyAPIWarning: The Query.get() method is considered legacy as of the 1.x series of SQLAlchemy and becomes a legacy construct in 2.0. The method is now available as Session.get() (deprecated since: 2.0) (Background on SQLAlchemy 2.0 at: https://sqlalche.me/e/b8d9)
  db_account = Account.query.get(account_id)
./Users/joehall/Coding/AAVI/AAVI/backend/api/account.py:96: LegacyAPIWarning: The Query.get() method is considered legacy as of the 1.x series of SQLAlchemy and becomes a legacy construct in 2.0. The method is now available as Session.get() (deprecated since: 2.0) (Background on SQLAlchemy 2.0 at: https://sqlalche.me/e/b8d9)
  db_account = Account.query.get(account_id)
./Users/joehall/Coding/AAVI/AAVI/backend/api/image_analysis.py:198: LegacyAPIWarning: The Query.get() method is considered legacy as of the 1.x series of SQLAlchemy and becomes a legacy construct in 2.0. The method is now available as Session.get() (deprecated since: 2.0) (Background on SQLAlchemy 2.0 at: https://sqlalche.me/e/b8d9)
  user = Account.query.get(account_id)
.
Ran 20 tests in 5.356s
OK
```

3.2. Integration Testing

Test Case ID	Test Case Objective	Test Case Description	Expected Result	Actual Result
TC01	Check the interface between the Login and Home react components.	Enter correct account credentials into input fields and interact with the blue login button.	To be directed to the home page of the application.	Directed to the home page
TC02	Check the error handling on the Login component when entering invalid credentials.	Enter incorrect account credentials into input fields and interact with the blue login button.	To be shown a red error message displaying 'Invalid Credentials'.	Error message displayed.
TC03	Check the error handling on the Login component when not entering any credentials.	Not enter any credentials into input fields and interact with the blue login button.	To be shown a red error message displaying 'Please fill in all fields'.	Error message displayed.
TC04	Check the interface between interface between the Login and Signup components.	Interact with the grey sign-up button to be directed to the signup page, fill out the input fields and interact with the blue sign up button to be directed back to the login page.	To be directed to the signup page to create an account and then directed back to the login page.	Directed to sign up page, account created on the system, then directed back to the login page.
TC05	Check the error handling on the Signup component when entering username that already exists on the system.	Enter a username into the into the input field that already exists on the system and interact with blue sign-up button.	To be shown a red error message displaying 'Username already exists'.	Error message displayed.
TC06	Check the error handling on the Signup component when not entering any details into input fields.	Not enter any details into sign-up input fields and interact with blue sign-up button.	To be shown a red error message displaying 'Please fill in all fields'.	Error message displayed.
TC07	Check the error handling on the Signup	Fill out sign-up form but enter a different string into the re-type	To be shown a red error message displaying	Error message displayed.

	component when entering a different password into the re-type password input field.	password input field than what is in the password input field.	'Passwords do not match'.	
TC08	Check the interface between the Home and ScanResults components.	Interact with the large grey button in the centre of the home page to upload a photo and then interact with the blue results button. The system will then display a loading screen, then promptly navigate to the scan results page. The uploaded image, AI generated title and summary, text-to-speech button and chatbot interface will be displayed.	To be directed to scan results page, after uploading an image, showing the uploaded image, AI generated title and summary, text-to-speech button and chatbot interface.	Directed to scan results page with all expected content displayed.
TC09	Check the error handling on the Home component when attempting to upload an image without an API key.	With no API key associated with the account. Interact with the large grey button in the centre of the home page to upload a photo and then interact with the blue results button.	To be shown a red error message displaying 'Account does not have an API key'.	Error message displayed.
TC10	Check the error handling on the Home component when interacting with results button without uploading an image.	With no uploaded image. Interact with the blue results button.	To be shown a red error message displaying 'No selected file'.	Error message displayed.
TC11	Check the error handling on the Home component when uploading a file type other than '.png', '.jpg', '.jpeg' or '.webp'.	After uploading an invalid file type, interact with the blue results button.	To be shown a red error message displaying 'Invalid file type'.	Error message displayed.

TC12	Check the error handling on the Home component when attempting to upload an image with an invalid API key.	With an account that has an invalid API key. After uploading an image, interact with the blue results button.	To be shown a red error message displaying 'Invalid API key'.	Error message displayed.
TC13	Check the chatbot interface on the scan results page and the interaction with the message component.	After uploading an image to be analysed by the system and being navigated to the scan results page. Interact with the chatbot interfaces input field and enter a question regarding the uploaded image.	To be shown a grey message container with the inputted message as well as blue message container displaying the text 'Loading...'. Once the AI response has been retrieved it will replace the loading message.	Chatbot interface acts as expected, showing the user inputted message, loading message and AI response.
TC14	Check the response in the chatbot interface if being asked a question unrelated to the uploaded image.	Interact with the chatbot input field and enter a question completely unrelated to the uploaded image.	To be shown a blue message AI response detailing the image and explaining that the message is unrelated to the topic and suggesting talking about the image instead.	AI response explains that the question asked is not related to the uploaded image and suggests talking about the image instead.
TC15	Check that the chatbot interface can use text-to-speech in order to read the AI message to the user.	Interact with the button underneath the message text.	To be played an audio file that reads out the contents of the message.	Audio file containing the message is read aloud.
TC16	Check the interface between the Gallery and GalleryImage Components.	From the home page, interact with the gallery tab in the navigation bar.	To be shown a grid of previously scanned images associated with the account, along with the AI generated titles.	Gallery page is displayed along with GalleryImage components for all scanned images.
TC17	Check the interface between GalleryImage	From the gallery page, interact with a previously uploaded image in the gallery to	To be shown a page displaying conversation data such as: the image,	Conversation data is displayed as expected.

	components and Conversation components.	display conversation information.	date of the conversation, summary of the image, chatbot messages and text-to-speech icons for user and AI messages.	
TC18	Check the error handling in the Gallery component when attempting to view the gallery page when no images have been scanned.	With an account that has not previously uploaded any images to the application. From the home page, interact with the gallery tab in the navigation bar.	To be shown a message stating 'No images found'.	Message displayed on the gallery page.
TC19	Check the interface for the Account component when attempting to save an API key to account.	Having previously logged in. From the home page, interact with the account tab in the navigation bar. From the account page, interact with the blue edit button underneath the 'OpenAI API Key' header. Input a valid API key into the input field and press the blue save button.	To be directed to the account page after interacting with the account tab, after inputting a valid API key into the input field the user will be able to use the apps main functionality.	API key is successfully saved to the account.
TC20	Check the interface for the Account component when attempting to delete account.	Having previously logged in and navigated to the account page, interact with the red delete account button. A pop up will appear to confirm with the account holder whether they want to delete their account, informing them that all account data will be deleted forever. The user will then interact with the red delete account button.	To be shown a pop up after interacting with the delete account button. After pressing the delete account button again to confirm the deletion the user will be redirected to the login page and all account data deleted from the system.	Redirected to login page and account data removed from the system.
TC21	Check the interface for the	Having previously logged into an	To be shown a pop up to create an	A new account is

	Settings component when attempting to create a new account on the system.	administrator account, interact with the settings tab in the navigation bar. Interact with the blue add user button. A pop up will appear prompting the administrator to input account information into the input fields as well as selecting the role of the account. Then the administrator interacts with the blue create account button.	account on the system, after filling out the details and interacting with the create account button the pop up will disappear and a new account will be added to the list of user accounts on the settings page.	added to the system, evident by the new account added to the list of users.
TC22	Check the interface for the Settings component when attempting to delete an account on the system.	Having previously logged into an administrator account, interact with the settings tab in the navigation bar. After finding an account from the list of accounts on the settings page to delete, interact with the red trash icon, a pop up will be displayed to confirm the deletion.	To be shown a list of accounts on the system, after interacting with the red trash icon and confirming the account deletion the account will be removed from the list of accounts on the system.	The account will be removed from the list of accounts on the system and deleted.

3.3. User Acceptance Testing

3.3.1. Questions

1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.

3.3.2. Results

3.3.2.1. Non-Blind folded user results

Participant 1

Q1: Strongly Agree
Q2: Disagree
Q3: Strongly Agree
Q4: Agree
Q5: Strongly Agree
Q6: Disagree
Q7: Strongly Agree
Q8: Disagree
Q9: Strongly Agree
Q10: Neutral

SUS Score: 70

Participant 2

Q1: Strongly Agree
Q2: Disagree
Q3: Strongly Agree
Q4: Disagree
Q5: Strongly Agree
Q6: Disagree
Q7: Strongly Agree
Q8: Neutral
Q9: Strongly Agree
Q10: Disagree

SUS Score: 65

Participant 3

Q1: Strongly Agree
Q2: Disagree
Q3: Agree
Q4: Disagree
Q5: Strongly Agree
Q6: Neutral
Q7: Strongly Agree
Q8: Disagree
Q9: Strongly Agree
Q10: Disagree

SUS Score: 62.50

3.3.2.2. Blind folded user results

Participant 4

Q1: Neutral
Q2: Disagree
Q3: Agree
Q4: Agree
Q5: Agree
Q6: Disagree
Q7: Agree
Q8: Neutral
Q9: Disagree
Q10: Agree

SUS Score: 55

Participant 5

Q1: Neutral
Q2: Agree
Q3: Disagree
Q4: Agree
Q5: Neutral
Q6: Disagree
Q7: Disagree
Q8: Agree
Q9: Disagree
Q10: Strongly Agree

SUS Score: 52.20

Participant 6

Q1: Agree
Q2: Neutral

Q3: Agree
Q4: Strongly Agree
Q5: Agree
Q6: Disagree
Q7: Disagree
Q8: Disagree
Q9: Neutral
Q10: Agree

SUS Score: 57.50

4. GitHub Repository

[joeHall02/AAVI: AI Assistant for the Visually Impaired \(github.com\)](https://github.com/joeHall02/AAVI)