CM3203

One Semester Individual Project

Final Report – A personal diary app



Author: Eloise Collings Student Number: C1523126 Project Supervisor: Alia I Abdelmoty Project Moderator: Carl Jones

Abstract

Personal diaries have traditionally been kept using physical methods of paper and pen throughout human history. These diaries are used to store memories and as an organisational tool to plan out daily life among many other activities. This report outlines how a simple mobile diary application can be created to allow users to fulfil many of their diary wishes on their smart devices.

The application is created to include benefits such as photograph entries, location tagging and emotion tagging. Events and their respective countdowns can be added, as well as simple to-do task list for organisation of everyday life.

This report shows how the design, approaches and implementation of these features can be achieved to create a functioning personal diary application. Testing and evaluation have been carried out on the system to assess how well this application achieves the set objectives. Related user tests and a user questionnaire have been conducted to further evaluate how well this application meets user's expectations.

Acknowledgements

I would like to thank my friends and family for their support, patience and encouragement in this project, especially the continued motivation to keep me working hard. I greatly appreciate their help with their valuable responses to the questionnaire and for acting as volunteers for user research and user testing. I am grateful for all their help and opinions they have offered me.

I would also like to thank my supervisor Alia Abdelmoty for her continued support throughout the project. With her insightful guidance, advice and encouragement, she has helped to lift the quality of this project to a standard I am proud of.

Contents

1.	1. Introduction						
2.	Background						
	2.1.	Exis	ting Solutions7				
	2.2.	Use	r Research – Personas12				
	2.3.	Iden	tified issues and benefits15				
	2.4.	Con	straints15				
3.	Ар	proach	٠17				
	3.1.	Plan	ning17				
	3.2.	Lear	ning17				
	3.2	2.1.	Android Development Courses				
	3.2	2.2.	Android developer guides17				
	3.2	2.3.	Location				
	3.2	2.4.	Data Storage				
	3.2	2.5.	User Interface Design19				
4.	Sp	ecifica	tion20				
	4.1.	Fund	ctional Requirements				
	4.2.	Non	-Functional Requirements22				
	4.3.	Use	Cases24				
5.	Desig	gn					
	5.1.F	Prototy	rpe				
	5.1	1.1. In	itial Prototype32				
	5.1	1.2.	Improved Prototype41				
5.	Im	pleme	ntation44				
	5.1.	Data	abase44				
	5.1	1.1.	UML Class Diagram46				
	5.1	1.2.	Activity Diagrams				
	5.2.	Syst	em48				
	5.2	2.1.	Database Retrieval				
	5.2	2.2.	Navigation				
	5.2	2.3.	Diary entry51				
	5.2	2.4.	To Do Lists				
	5.2	2.5.	Event lists for date60				
	5.2	2.6.	Home Page61				
	5.2	2.7.	Analysis				
6.	Ev	aluatio	n63				

6	1. Test Cases								
6	6.2. Usability Testing								
	6.2.1. Usability Test Results								
6	6.3. Questionnaire								
	6.3.1. Methodology								
	6.3.2. Key Results								
7.	Future Work								
8.	Conclusions75								
9.	Reflection on Learning77								
10.	References								

1. Introduction

The aim of this project was to design and create a personal diary app that would allow a user to create journal entries on their smartphone or other smart device that would be beneficial enough to encourage the use of digital diary apps. The motivation behind this app was to give the user the freedom and flexibility of having everything they could create in a physical diary in one complete application.

The objectives set out for this project were too:

- Allow entries to be made with the included benefit of photographs, location tagging and emotion tagging. These entries are flexible and allow the user to enter their data using multiple methods.
- Allow events to be created for the users specified date. Countdowns for specified events are available for the users review
- Allow to-do tasks to be created for the users specified date. These tasks can be easily accessed throughout the day, with the ability to check off any completed task as required. These tasks can be created efficiently, with the ability to create repeated tasks if required.
- Reminders are available for all events and tasks added by the user if required. Reminders can also be added to remind users to create their entry for the day

The focus of the project was to create a user-friendly app, with the user interface and interactions within the app being simple to use and easy to understand. It will allow the user to analyse any information they have entered, including the analysis of the emotional tags to highlight any patterns and relationships between given dates. Users will also be able to analyse the text to identify any connections between specified words and the location and emotion tags.

To achieve the above, the design and all functionality was created with a user-centred design approach. Each function and layout were evaluated using Nielsen's 10 heuristic evaluation principles to ensure that the resulting designs were as appropriate as possible and would meet user expectations in terms of navigating through the system efficiently.

In the implementation of the system, an Agile based approach was used to ensure all previously set requirements and objectives were achievable throughout the projects development. Reviewing the functionalities currently possible to the desired functionalities, each method could be refined and adapted where necessary to ensure objectives were being met.

2. Background

2.1. Existing Solutions

The starting point for this project was a review of existing diary apps to establish what was currently available. This research also gathered information on the views and opinions of real users that had a need for an app like this project. Users were asked their opinion on the good and the less favourable elements of the app, and the data collected was used to inform the development process and to ensure the final product met the expectations and needs of real users.

The background research for the development of this app included a review of mobile applications which provided a journal function. The intent was to further develop the basic journal function by including additional elements such as emotional analysis, location tracking and reminder applications to give the user an enhanced product with increased functionality.

To fully understand what was already available on the market, research into some of the currently available journal apps was undertaken. This involved the trial use of a selection of apps to enable the features of each to be explored in detail. The results of this trial are summarised in the tables below.

Name: Day One
Rating: 4.6/5 (Apple Store)
Summary:
Is the top recommended journaling app on the app store, has also been Apple Editor's Choice multiple
times.
Offers a wide range of features, including; journal entries with photos and videos, location tagging,
weather tagging, motion and activity tracking. Currently playing music and step count can also be included
with each entry.
Customisable reminders available to reminder users to complete their journal entry throughout the day.
Free version of the app offers all features mentioned above. Subscription cost of \$34.99 (£27.12) per year
to receive premium features that include being able to sync across multiple devices, allowing storage of
more than one photo per entry and allowing multiple journals per account.
User Reviews:
"Beautifully organised"
"Whilst expensive [full superscription price] it seemed a worthwhile expense. Worth the long-time
investment in my memories"
"The new features such as 'on this day' and multiple journals make it well worth the upgrade"
"[Since the update] Quick access to location and photo picker has been removed, those options are now
hidden away"
Things to take from app:
Users have been willing to pay a premium for extra features such as multiple photos per entry, this will be a good feature to include with this project as standard.
Usability is the most important aspect of all applications, ensuring all main features are easily accessible to
the user is a must. A customisable aspect within this project may be beneficial to allow users to 'hide'
features they may not require, but ensure that for a first-time user, all features are accessible.
Experience using app:
Help tips were available on first launch to help show features and navigation around app. But other than
these pop-up hints, not much more help was given. The user must trial and error all features in the
application to discover what they do.
Not much text on home screen, mainly icons which were clear if you have experience using similar mobile
apps, but may not be clear to inexperienced users

Lots of features were unavailable unless a premium paid member, however, features that were available were easily accessible and simple to use, would be good for occasional/simple journaling on a basic account.

Adding an entry: simple to use, auto finds location and weather information, logs time and date of entry. Can also add user's own tags. Can have multiple entries for same day – to access each entry was confusing and inefficient. However, the idea of having multiple entries for one day was beneficial to have more than one location available to tag etc.

User can view their journal using different visual methods; via a map of all locations tagged, or a calendar view, or a photo album type screen with all photographs entered.

Entries can be filtered by many options: Favourite, Tag, Place, Audio, Year, Weather, Creation Device, Activity, Music

On this day feature, that shows all previously entered entries for same date at different years Can activate passcode if user want their journal private

Can export entries to json, plain text or pdf. The user can specify the date range of entries, and have included/excluded tag

Can connect social media – Instagram or Foursquare

Can search text to find relating entries

Who would benefit using this app:

The app would suit all users who are interested in journaling their everyday lives, the use of different entry medias would benefit different people, as you could enter journal info using plain text, speech to text (premium only), drawings (premium only) and photographs. The flexibility gives the user the choice. Dedicated users would see the benefit of the subscription features, with the ability to sync with multiple devices and back up of data would, again, give the user more flexibility.

This app did not seem to have many features to do with reminders, other than to remind the user to complete the journal every day, so users looking to use the app as an organisational tool would not benefit from this

Available at:

http://dayoneapp.com/?utm_source=zapier.com&utm_medium=referral&utm_campaign=zapier

Name: Diarium

Rating: 4.4/5 (Google Play Store)

Summary:

This is a simple and "feature rich" app, it offers multiple media types in journal entries, including speak-totext, drawing, multiple photos and rating of individual journal.

Can sync across multiple android or Windows devices and can export journal entries to a file type of the user's choice.

App can sync with other social media, including Twitter or Facebook, to include feed posts into the journal entry.

Allows a reminder to be shown once a day for the user to complete their journal entry

Free version of the app available on Android, or £2.32 for premium app that allows synching and exporting of journal entries.

User Reviews:

"Best diary app I have found. Works really well and can add photos. It's great that it syncs with Windows as I use this dairy on my laptop and android devices"

"Appreciate the audio dictation & export capabilities"

"I use this app to gather my thoughts every day and plan out what I will do tomorrow"

"Very clean interface. I appreciate the daily notification reminding me to write something"

Things to take from app:

The option to have daily reminders to complete the journal entry is a good idea. This can also help to keep people using the app, which will allow for better analysis to be made, as more data entered

Experience using app:

Home page is very well laid out. Icons and text are self-explanatory, would be easy to use for beginners without much help needed.

Ability to export entries to word doc, web page, plain text, these can include or exclude photos, tag, people, location, weather, motion data of the user's choice.

Creating an entry is very simple. Application gives a prompt to start entry, user can add own tags, can tag people to the entry by accessing contacts list manually tagged.

The user can give the day a rating as a way to analyse their entries.

Noticed the auto location feature is less precise than other applications that use user location

Very nice calendar interface, if you add a photo to your entry, this photo become the icon for the day in the calendar.

Statistics of the user's journal can be reviewed, the types of statistics available include the number of diary entries, the total number of photographs entered, the average rating of their journals, their most used tag and many more

Who would benefit using this app:

As with the DayOne app, this app would be suitable for all users wanting to journal their everyday lives. The simple layout would suit all users from technology beginner to advance from the start. The layout and features are not too advanced and not too simplified that beginners have the help and hints to correctly use features and advanced users could use the app without being slowed down by oversimplified features. As before this app is limited in the reminder features available, so would not be suitable for users looking an organisational tool

Available at:

https://timopartl.com/?utm_source=zapier.com&utm_medium=referral&utm_campaign=zapier

Name: Five Minute Journal

Rating: 4.7/5 (Google Play Store)

Summary:

A simpler approach to journaling, this app breaks down a journal entry to have the user spend five minutes a day to list the important things that the user wishes to achieve in the morning then prompts them (using the same 5 questions each day) to complete the list of things they have achieved in the evening.

This app was based on positive psychology research that uses the idea that if a user can clearly see all the achievements they have made in the day, this will improve their self-gratitude and self-reflection.

The app offers text and photographic entries. It has a simple timeline view, with easy to navigate screens and activities. A nice feature of the app includes the option of receiving positive daily quotes and weekly challenges that can be shared across the user's social media

The app costs £4.99 on the Google play store which includes all the features mentioned above

User Reviews:

"I use it to get me focused for the day and reflect on my achievements in the evening. I would definitely recommend this app"

"[This app] put me in a good mood, focus my attention on positive things, it also reminds me to be proactive in my day to change the things I can"

"I love this app; it really helps me to focus on gratitude and it doesn't take a lot of time"

"Very simple but clean interface, helps me journal every day. I like that you can add a picture"

Things to take from app:

This is a different approach to journaling by using a set of simple questions to help prompt the user to plan their day and journal the achievements they have made. The shorter entries seem to be helpful to users who are new to journaling as they are more motivated to complete their set achievements every day, with the morning and evening reminders.

Experience using app:

As this app costs £4.99 to download, this was not downloaded, but research into how other users (eg using in app photos taken by users and promotional images (Intelligent Change, 2019), YouTube videos on user's experience (Keep Productive 2017) (Overall Adventures 2017)

The layout of the days is split into morning and evening entries that the user is prompted to enter every day. There are 3 questions/prompts the user will enter in the morning: "I am grateful for...", "What will I do to make today great" and "Daily affirmations. I am...". And in the evening 2 different

questions/prompts are presented: "3 amazing things that happened today..." and "How could I have made today even better?"

The layout of the individual entries is nice as they use the photograph for the day's entry as a cover photo, the user can scroll through these entries as a timeline layout. The UI is simple and straightforward, without clutter, it seems to be simple to navigate for both beginners and experienced users, with icons and text hints available.

Who would benefit using this app:

This app would be perfect for users that lack motivation for keeping up with journal entries, the prompts that are given in the app would encourage users to create an entry. Although some may find the repetitive nature of the same questions being asked every day tedious and may tire of these same questions. The app focuses on positivity on simple questions and answers, with the prompt this seems less like a memoire journal and more to make a positive impact on how a user sees each day.

Available at: https://play.google.com/store/apps/details?id=com.intelligentchange.fiveminutejournal

Name: BZ Reminder

Rating: 4.7/5 (Google Play Store)

Summary:

A simple to-do list app. Helps the user to set reminders to organise their day in a simple layout. User can set multiple reminders and timers throughout the day to consistently create a notification to remind the user to complete the task. Users can use at the monthly calendar, that can be colour coordinated, to have as an organisational tool

User Reviews:

"It's very easy to add a task and to set it for the exact time and date you want without too many clicks" "Solid UI that is very easy to just glance at and get all the information I need"

"Consistent reminders until you complete the task, very useful"

Things to take from app:

The users of this app praise the simplicity of the app highly, the layout is decluttered and only the most important features are displayed as required. An apps layout should be easy for the user to navigate, with recognisable icons to access features. The app should be accessible for both new users and experienced users, so not relying on user's memorisation or knowledge before using the app, and keeping all features consistent

Experience using this app:

Very simple interface, with simple and helpful icons and text. Dedicated screens for each feature give a clean UI.

The calendar screen displays small coloured dots within the date to show on which days there are reminders set for, using different colours for different types of tagged reminders.

Reminders are clear and consistent. If user does not acknowledge first reminder, a second will be sent etc. until the user acknowledges – which makes sure user does not miss reminder.

Who would benefit from this app:

Any user who is looking for daily reminders (either one off events, or repeating events) this app is simple to use and set up the reminders. The information entered for each item is limited, only name, date and time so the user will only be able to enter the basic information per item.

Available at: https://play.google.com/store/apps/details?id=com.bzzzapp

After reviewing each app, DayOne was identified as the one most similar to the ideas and requirements that this project intended to achieve. A more thorough evaluation of this app was carried out to assess the heuristics in more detail. The full heuristic evaluation can be found in Appendix 1: Heuristic Analysis of DayOne App.

Going through the app as a user, each screen will be shown and evaluated, a heuristic evaluation of each screen will be made. Any favourable elements will be noted to influence this project. Any issues or elements that do not seem appropriate will be evaluated and will be noted to avoid in this project.

The evaluation looks at each of the 10 Nielsen Heuristic elements (Nielsen, J 1994a) that are appropriate for the current screen, and gives them a score (Very Poor, Poor, Moderate, Good, Excellent) to determine how well this application meets each of the specified heuristic evaluators. Additional comments justifying the score will be made.

Exact location address and photographs may be blurred for privacy reasons.

All information found during this evaluation will be used to inform the design and implementation of this project. The heuristics scores will be considered to try and improve any elements that the application fell short on and will try and sustain any high scores it achieved by using similar methods and features within this project.

2.2. User Research – Personas

These personas are based on the conventional users of diaries/journals.

Information gained from the existing solutions, including user reviews and the types of features available for each application, as well as informal information collected from potential diary app users, has been used to give 3 personas based on the three main types of users that may wish to use a mobile diary application.

Name: Georgia Lockwood	Age: 24						
Photo:	Quote:						
"I wish to improve my journaling experience of enhancements to help me reflect on my past							
Description Georgia has been journaling since she was 14. She writes an entry into her journal book every day,							
describing her day and emotions. She often refle	cts on her past by re-reading old journal entries. She has a						
smartphone and is comfortable using apps so long as they are simple to use, and no technical knowledge is							

and time by transferring to a digital journal that she can access whenever she requires Goals

- Be able to continue her daily journaling efficiently
- Have access to all previous journal entries, retrieving entries is simple and fast
- The diary app will allow her to enhance her journal entries using photographs of the day's events

required to operate them efficiently. She is looking for an app to replace her physical diaries to save space

• A map of the locations she has recorded in her previous entries can be displayed, showing how often this location has been visited and on what dates

Technical Knowledge

Current Diary Usage

Name: Jake Granger	Age: 32					
Photo:	Quote:					
Fileto. Guote: "I want an app to help me observe my life, seeing how past experiences affect my emotions" Experiences affect my emotions affect						
Description Jake is a business man who is constantly on the move, travelling around the world for his work. He has recently become interested in how his current life experiences are affecting his emotions and would like a way to analyse his emotions compared to the places he visits and the experiences he has. As he is does not currently keep a journal, he is not in the habit of writing his experiences daily and has no previous expectations on what he would expect in a diary app. He is therefore hoping for a simple to use and self-						
Goals						
 Allow reviews of all past entries, accessib Determine average emotions felt for loca anxious etc. Be able to track experiences and emotion patterns in emotions felt on days these experiences and average emotions 	le simply and efficiently for his own analysis ations visited, find the places he is most happy, upset, hs, through keywords found in the journal entries, show aperiences were documented					
Technical Knowledge						
Current Diary Usage						

Name: Sophie Matthews	Age: 19
Photo:	Quote:
	"I would like an app to help me plan my everyday life, using simple to-do checklists and countdowns to big life events"
Description Sophie is a student with a busy social life, she ofte	en has events lined up in her future that she is excited for
and counts down the days until the event. As a st	udent she often like to plan small tasks to fulfil her days

and counts down the days until the event. As a student she often like to plan small tasks to fulfil her days to make sure she keeps on top of her work/life balance. She would like an application that is mobile and easily accessible 24/7 to keep up with her busy life. She likes minimalist designs to allow smooth and easy to use availability to everything she does without wasting time on cluttered or hard to navigate activities

Goals

- Countdowns to important events will be available in an exciting and simple format
- A check list can be added, and editable throughout the day to help her organise her daily tasks, these tasks can be checked off as she accomplishes them
- Reminders for events and tasks can be set for specific times and dates



2.3. Identified issues and benefits

- It is important that the user has full navigational capabilities around the system. Ensuring to keep user control to avoid errors by allowing the user to undo/redo actions is important to keep the flow of the system as efficient as possible. By not making the user do extra tasks or work to resolve avoidable mistakes will be paramount to making the system enjoyable for the user
- Keeping a minimalist design to reduce clutter and unnecessary information will give the system a cleaner look. This will also benefit the user as only relevant and informative information will be shown, so no confusion in the data shown will occur.
- Keeping the data available for the user to enter relatively flexible, by not limiting (where sensible) the data the user can choose from. This will give the user more flexibility and gives them a more unique experience in using the application, being able to customise their data in a way that suits them. While at the same time, it may be beneficial to have some pre-set data available for novice users to simply select from rather than inputting their own.
- Giving the user feedback on the current state of the journal can give the user chance to analyse and evaluate how they are using it. This could in turn encourage use of the journal as the user can clearly see when they last made an entry and what sort of entries they made (eg text or image). The use of 'On This Day' features would be beneficial to some users to see their progress in life or just as a curiosity of how different or similar their lives were years before
- Being able to efficiently search through the entries will be important as this is another way to encourage the use of mobile diary app over physical copies, being able to easily and quickly search can help users organise and review their entries without manually reading each entry. This can also help with user reflection.
- Another important feature to show the benefit to using mobile diary applications is the flexibility in how the user can use the application. This can be in the types of input data they can have as well as having different ways for the user to view this data. For example, having a photo gallery to display the images connected to entries as well as a text list, or having a map of location tags.

2.4. Constraints

There are a number of constraints that may affect the outcome of this project. These are discussed and where possible any steps that can be taken to minimise these will be outlined

The first and most prominent constraint, will be that the nature of this project is not one that I have previously experienced before. The scale of this project is much larger than any solely undertaken before, this will mean a lot of planning and organisation throughout the projects' lifecycle will be required. I have not used or created an application using the Android platform before, so before the project begins, more research into creating an Android application to plan what will be possible for this project in terms of requirements, accessibility and scalability will be undertaken.

Another constraint will be the strict timeline this project will be completed in. From research to final submission this project will take at most 15 weeks (assuming no extensions are required due to extenuating circumstances). This will have an impact on the types of features that can be implemented into the project within the time limit. It will be important to plan how and when each activity will be implemented to ensure at least a basic functioning system is delivered at the deadline. Extensions and extra features will be planned further along the project's timescale to ensure this is the case.

This project is not being conducted for one set client, this will mean volunteers will need to be found that will act as users to allow for accurate testing and analysis of the system. The sooner test users can be found

the more accurate the project will be, as having user opinions throughout the project (from prototype to final product) will improve the standard of the system.

3. Approach

3.1. Planning

The main approach to conducting this project will be based on the Agile development methodology. The Agile approach will allow the project to adapt and develop throughout the process as more information and methods are learnt. Breaking each section of the project into smaller phases, to allow for more time to be spent on elements that will require additional research and learning. As this is an individual project, some methods adopted from the Agile methodology are not applicable to an individual, but the main development process will be used. These will include the smaller sections of work, that are regularly tracked to ensure they are being carried out in the most efficient way. The system will be built with the focus of delivering a working system, with each development phase working on implemented features to improve them in an efficient cycle. Regular reviews of the system as a whole either as a developer or with users will be carried out to ensure the system is on track to meet the requirements that were laid out before, and if any changes need to be made either to the system itself or to the requirements acceptance criteria.

Each phase will be planned carefully to ensure all basic functionality of the system is built, keeping designs and activities simple at the beginning, and then allow for improvements and expansions as is required for the project. This will keep the strict timescale for the project manageable with the number of features and activities that the project is hoping to achieve sensible and realistic. Any extra features will only be implemented once the basic functionality is working to an appropriate standard.

3.2. Learning

3.2.1. Android Development Courses

The decision to take on this project required the acknowledgment and acceptance that new skills will need to be learnt quickly and effectively to be able to achieve a high standard at the end of this project. The decision to create an Android app was made as Android seemed to have more learning resources available (as oppose to other platforms like iOS). These resources included the Android Development online courses available on Udacity (Udacity 2019), created by Android employees. This ensures this is a reliable source of learning material. Although the original content of the course uses Oreo as the main operating system and therefore can be assumed the course was created in 2017, the material is constantly being updated. Where necessary, links and updates to features mentioned in the course videos are provided, providing up to date content and information.

The course involves creating toy and example applications to show how the android system operates through practical applications. This was important for me to see how the features and structures of android applications work together, being able to play around with these features myself helped me to understand the technology enough to be able to apply them to this project.

3.2.2. Android developer guides

On top of the course, there are many guides and write ups of the features included in the android platform. These guides go into more implementation and background detail than the course. These are useful to expand the knowledge gained through the course to better fit the projects individual requirements (Android Developers 2019a).

These guides are written by Android and are consistently updated to show any deprecated methods, giving useful hints and guides on how best a method should be used and implemented. Sample codes are also available to show working code examples which will be invaluable to understand how methods work together on the platform.

3.2.3. Location

To implement all features that use location, this project will make use of the Google Location Services API. (Android Developers 2019b) This framework allows for user location tracking and visualising locational data on maps. It is a highly documented method of location which will be useful to the project when developing and debugging. The framework also makes use of the Google Maps Android API, to display the data on a map using the Google Maps servers, which will include all benefits of using Google Maps (including data stored on Maps servers, touch gestures and all necessary UI elements for the user to effectively navigate the map and customisation like markers and overlays) Markers will be used to show the user more detail about the places they have tagged from their diary entries.

To implement the Google Maps Android API, all devices must have access to the Google Play Store through Google Play services, which for majority of Android devices that this project will be developed for, will have access to.

The main feature using location within this project will be using the user's location to tag diary entries. Google Play services can access the user's last known location (which in most cases is also the user's current location, unless the device has no signal to access the location service). Google services can find the user's location in the background (if the app is not currently in use) which will make the diary app entry process quicker and more efficient, the app will be able to suggest the current location based on the continuously updating last known location. No extra time is spent calculating location each time the app is accessed. This interval between updating location can be modified, using the APIs *setInterval()* or *setFastestInterval()* methods. This will allow the optimal interval time to either be manually entered or automatically calculated for the specific device the app is running on.

The API could also be used for the advanced feature of tracking activity. The Activity Recognition Transition API can tell when a user is doing a specific activity (eg walking, running, driving) and all information related to this (eg speed, mileage, time spent doing activity). This will be a good enhancement for the project if the time if available, to allow automatically detected activity tags.

3.2.4. Data Storage

This project depends on the storage and modification of user data. It is therefore essential that the method chosen to store this data is reliable and has sufficient methods to support the data entered. For this reason, MongoDB will be used to store the data. Using a NoSQL approach to data storage will mean the framework around the data will be more flexible, allowing for varying entry types without wasting storage space in fields that may be used in one entry but not in another.

Each document within the MongoDB collection will be used to store one diary entry, these documents can have differing schemas, based on the data that has been entered for the specified entry.

To access a MongoDB database from within the app, MongoDB Stitch will be used. MongoDB Stitch is a platform developed by MongoDB to allow communication from programs such as mobile apps to a user specified database (MongoDB 2019a). Stitch is accessible from a MongoDB Atlas account, this is a fully automated cloud service built by MongoDB, allowing for smooth and reliable clusters creations and database management. A free cloud database is available for smaller or personal applications that have up to 512MB of storage. For this project, the database will be created using the free cloud database package available, with the idea of if this application was expanded and developed to be accessible to more users, the upgrade to a dedicated cluster with larger storage and improvements in performance can be implemented easily and with minimal change to the application code.

As MongoDB Stitch was created by MongoDB, the documentation and tutorials on implementation stitch into an app are extensive, allowing for informal and complete learning experiences to achieve high performing communication between the app and the database.

3.2.5. User Interface Design

When developing the design of the system, the way the user interacts with the system and the overall aesthetic will be the priority. The design of the system should be unique for this application, but also follow the standards a user would expect from a mobile application. This will be important to allow the users to freely navigate through the system without the need to learn a new UI approach.

The standard guidelines that Android recommends (Android Developers 2019c) ensures the way this application looks and behaves is consistent with most other applications on the Android platform. Material.io (Google Design2019) include the guidelines supported by Android, therefore the practices and visual aspects recommended here will be incorporated into this application to ensure the UI is functional, practical and visually pleasing.

4. Specification

4.1. Functional Requirements

4.1.1. Must have

FR1: The system will allow the user to enter text-based journal entries on specified dates Acceptance Criteria:

• User can select their specified date and enter text that will be stored for this date

FR2: The entry can optionally also include photographs for any selected entry Acceptance Criteria:

• For the users specified entry, they will have the option to take a photograph from their mobile device, and/or select previously taken photographs found in their device's camera roll. These photographs will be stored for the date specified

FR3: The entry can optionally also include location tags for any selected entry Acceptance Criteria:

• For the users specified entry, they will have the option to add a location tag to the entry. This will include the user selecting a position from a map by using the user's current or last known location, and/or by pin pointing the location on a map presented to the user by selecting the position on the map or typing in the name of the location which will automatically pin point this location. The location tagged will be stored for the date specified

FR4: The entry can optionally also include emotion tags for any selected entry Acceptance Criteria:

• For the user's specified entry, they will have the option to add an emotion tag to the entry. The user can select an option from one of the pre-presented emotion tag options. This tag will be stored for the date specified

FR5: The user can review and edit any previous entries Acceptance Criteria:

- The user can select any date where an entry has been previously created, and all data entered will be displayed to the user (including any text, photographs, location tags and/or emotion tags)
- The user can select to edit the data for this entry. The user will be able to change any and all data, including deleting the entry including all data associated with it.

FR6: The user can create event reminders for their personal events on specified dates Acceptance Criteria:

- The user will select the date for their event, they will enter information related to the event, including the name, time and location (as necessary) and the user will select if and when they would like a reminder for this event
- The event will be displayed to the user on the applications home screen as well as detailed information on the specified date
- If a reminder time and date has been set by the user, the application will create and notify the user of the event when specified

FR7: The application can display countdowns to events specified by the user as the user requires Acceptance Criteria:

- While the user is creating the event reminder (as seen in FR6), they can also select to have a countdown displayed for this event. The countdown will be displayed on the applications home screen as well as a dedicated countdown screen.
- Any and all events selected to be displayed as a countdown will be displayed on the home screen in chronological order (with the closest event being displayed first in the list). The display will include the events name, and then the countdown of months, days, hours
- The countdown screen within the app, will display more detailed information of the events, including the name, time and date, location and the countdown in months, days, hours. Selecting an event from this list on this screen will allow the user to edit the information in this event.

FR8: The user can enter daily tasks or "to-do" tasks which can be checked off as the user requires Acceptance Criteria:

- The user can enter 'to-do' tasks from the applications home screen, where they will be displayed a list with currently unchecked boxes. The tasks will also be accessible through the date where they are stored, allow for the user to review the tasks completed/not completed on this date at future dates
- The user can access the list and check off any event they complete as they require
- The application will remind the user of incomplete tasks at intervals throughout the day, specified by the user

FR9: The application can create repeatable tasks as the user specifies

Acceptance Criteria:

- The user can enter task information in the same way as FR8, in the 'to-do' list. The user can specify this is a repeatable task
- The user can specify the interval between the task reminders (eg every day, every week, every Tuesday etc.) until a specified date (or indefinitely)
- The repeatable task will appear on the specified task list for the days specified without the user's involvement

4.1.2. Should Have

FR10: The application can perform text search analysis of the journal entries, displaying emotions and location tags that match the inputted search text

Acceptance Criteria:

- The user will enter a search term that may appear in their text journal entries.
- The application will search for any occurrences of this word in the user's journal entries. A total number of found entries (one count per date, not per occurrence of word) will be return to the user, selecting this number can give the user a list of the dates an occurrence of the word appeared, the date can be selected for the user to review the entry for the selected date
- The application will also display a map of locations that may have been tagged for entries that have occurrences of the search term. The map will pin point the locations with a number displaying the times the location and search term has occurred in the same entry. Selecting this location will give the user a list of dates of occurrences for location and search term
- The total number of each emotion tag found in entries containing the search term will be displayed, showing the user how this term seems to relate to a certain emotion. Again, selecting an emotion tag will display a list of dates where the search term and emotion tag occur

• The average emotion tagged for this result from the search will be displayed for the user's analysis

FR11: A map of all locations visited at least once can be displayed to the user, entry data relating to the location can be displayed when location selected

Acceptance Criteria:

- A map with single pin points where a location has been previously tagged at least once is displayed to the user (i.e. at most one pin for location, not per time visited)
- The user can select the location pin and any emotion tags relating to the location and date entry will be displayed to the user

4.1.3. Could have

FR12: The user can use the application to track progression of personal goals, with analysis of the user's progress being calculated by the app as appropriate

Acceptance Criteria:

- The user will enter their main end goal, including all information of their current circumstance and the goal criteria they wish to meet, and by which date they wish to meet this goal (if required)
- The user has the option to enter milestones they wish to meet on specified dates
- The user can update information on the app to show the progress made. The app will use this updated data with the goals and milestones to show; whether the goal has been met, how far away their goal date is, if their milestones have been met, how far away the user currently is from any milestones/goals
- If applicable, the app will analyse the data to show the relationship between progress made so far, and make projections/predictions on if the current progress is repeated, if the goal is reachable in the timeframe/when milestones may be met

FR13: The application will remind the user to complete their journal entry if not done so by a certain time

Acceptance Criteria:

- The application will notify the user to complete their entry for the day if they have not yet entered data for this day's entry
- This feature can be disabled by the user if required

4.2. Non-Functional Requirements

4.2.1. Must Have

NFR1: All entries must be updated in the database whenever an edit is made

Acceptance Criteria:

- If a new entry is created, this entry is stored in the database immediately after the entry is submitted
- Any edits made to previously created entries are updated in the database immediately after the edit is submitted

NFR2: All entries will always be accessible to the user

Acceptance Criteria:

- Whenever the user accesses the application, all previously created data will be available to the user, to view or edit
- If connection to the database is not available, an appropriate error message will inform the user of the current error and give appropriate solutions to rectify

NFR3: All activities on the application must be clear to the user on how to complete them

Acceptance Criteria:

- Sufficient hints and help must be given to support the user
- The UI will be simple and easy to navigate with clear and clean interfaces and icons
- The features within the application will have consistent layouts, with similarities in usage where possible to avoid confusion and make the users experience as clean as possible

4.2.2. Should Have

NFR4: The application can be password protected as the user requires

Acceptance Criteria:

- An option to have a password/passcode will be available, the user will select their password and only this password will grant access to the information within the app for the specified device
- The user has the option to disable/enable this feature whenever they require

NFR5: The response time for any activity completed on the app will be an acceptable time

Acceptance Criteria:

- Any and all activities attempted on the app, will have a response (either success of activity or error message) within 500ms maximum
- If response time reaches over 500ms the activity will time out and a prompt for the user to explain cause of time out will be given and user will be asked to try activity again if possible

4.2.3. Could Have

NFR6: Application will allow for future developments

Acceptance Criteria:

- All development of application will be documented to allow for current progress to be reviewed
- The activities created in the app will allow for developments by not creating unrepeatable code

4.3. Use Cases

The following diagram shows the use cases a user will be able undertake within the application, showing the main/primary tasks available.



Use	UC01	Use Case Adding a new entry Rating: Must have		Must have	
Case		Name:			
Number					
Goal:		The user can o	create a new diary entry for the	current date.	
Pre-condit	tions:	An entry for t	he current date has not yet bee	n submitted	
Descriptio	n:	The user will b	pe entering a diary entry for the	current date. The	e entry will include
		text data, an e	emotion tag, a photograph and	a location tag	
Basic Flow	<i>'</i> :	1. User open	s app, the diary entry screen is	displayed	
		2. User selec	ts to Add new entry		
		3. User enter	rs text data for the entry		
		4. User selec	ts an emotion tag for the entry		
		a.	Select from list		
		5. User selec	ts to a photograph for the entry		
		a.	User can choose to take a ne	w photograph	
		6. User input	s location tag for entry		
		a.	User selects current location		
		7. User saves	s entry for the current date		·····
Alternate	FIOW:	A-1 User is air	eady running app; the user will	navigate to the d	liary entry screen from
		the navigation	a sta ta abaasa a abata graab fra	itry button	
		A-5.d User sei	ects to choose a photograph inc	cation name or	ninnointing on man
Related II					
Basic Flow	v STN	-			
		Diary Entry Screen	elect to New Entry" New row added to entry list Select to input location tag	List of available emotion tags nput text into entry Location map Cu Dialogue box for type of photograph entry – Take New or Choose from Camera Roll	Data inputted into entry selects to take new photograph
Alternate	Flow ST	N Diary Entry Screen	Select to New Entry", New row added to entry list Select to input Select to input location tag Discrete to information Select to input location tag	List of available emotion tags Input text into entry Location map Inpu Dialogue box for type of photograph entry – Take New or Choose from Camera Roll	an emotion tag Uts own location tag Selects to choose Selects to choose photo from camera foll

Use	UC02	Use Case	Find previous diary entry	Rating:	Must Have		
Case		Name:					
Number							
Goal:		User can revi	ew previously submitted entries				
Pre-condit	ions:	Entries have l	peen previously submitted for th	ne users chosen	date		
Descriptio	n:	If a user wou	d like to review the entry they h	ave previously	entered, this will be		
		accessible. Al	l data previously entered (includ	ling any text da	ta, emotion tag,		
		photographs	or location tags) will be displaye	d			
Basic Flow	<i>'</i> :	1. User nav	1. User navigates to 'Home Page' via the navigation bar.				
		2. User sele	2. User selects chosen date from calendar				
		3. The entr	3. The entry previously submitted for this date is retrieved and shown to the user				
Related Us	se Cases	: -					
Basic Flow	' STN						
		Home Scre	Page Select required date fro	om calendar	Diary Entry Screen of selected date		

Case Number Name: Goal: User will be able to review, edit or delete any/all data from previous entry Pre-conditions: Description: A user will be able to review a previous entry from the users chosen date. The us will be able to delete the whole entry. Any edits or deletions will be confirment to be correct by the user when prompted by the app Basic Flow: 1. User navigates to the entry of the specified data 2. (Optional) User can choose to select new emotion tag 3. (Optional) User can choose to select new emotion tag 3. (Optional) User can choose to select new location tag 6. (Optional) User can choose to select new location tag 6. (Optional) User can choose to select new protograph 5. (Optional) User can choose to select new location tag 6. (Optional) User can choose to select new protograph 5. (Optional) User can choose to select new protograph 7. After the user onfirms the deletion of the entry, they will select the 'delete' button 7. After the user confirms the deletion of the entry, the entry will now be blank Alternate Flow: A4. (Optional) User can choose to select new photograph from devices camera ro generation tag Basic Flow STN UC02 - to navigate to the entry of the specified date UC01 - the input of data will be using the same method Basic flow STN Basic Flow STN Select to edit location tag Selects new map Selects new map	Use	UC03	Use Case	Edit previous entry	Rating:	Must Have			
Number User will be able to review, edit or delete any/all data from previous entry Pre-conditions: Entries have been previously submitted for the users chosen date Description: A user will be able to review a previous entry from the users chosen date. The us will be able to delete the whole entry. Any edits or deletions will be confirmed to be correct by the user when prompted by the app Basic Flow: 1. User navigates to the entry of the specified data 2. (Optional) User can choose to edit text data 3. (Optional) User can choose to select new emotion tag 4. (Optional) User can choose to select new emotion tag 6. (Optional) User can choose to select new emotion tag 6. (Optional) User can choose to select new emotion tag 6. (Optional) User can choose to delete the entire entry, they will select the 'delete' button 7. After the user confirms the deletion of the entry, the entry will now be bland Alternate Flow: A4. (Optional) User can choose to select new photograph from devices camera row (alter to edit) Select to edit) Basic Flow STN UC02 - to navigate to the entry of the specified date UC01 - the input of data will be using the same method Basic Flow STN Basic Flow STN Select to edit is using the same method Basic Flow STN Select to edit is using the same method	Case		Name:						
Goal: User will be able to review, edit or delete any/all data from previous entry Pre-conditions: Entries have been previously submitted for the users chosen date Description: A user will be able to review a previous entry from the users chosen date. The us will be able to delete the whole entry. Any edits or deletions will be confirmed to be correct by the user when prompted by the app Basic Flow: 1. User navigates to the entry of the specified date 2. (Optional) User can choose to edit text data 3. (Optional) User can choose to select new emotion tag 4. (Optional) User can choose to select new location tag 6. (Optional) User can choose to select new location tag 6. (Optional) User can choose to select new photograph 5. (Optional) User can choose to select new photograph from devices camera reconfirms the deletion of the entry, they will select the 'delete' button 7. After the user confirms the deletion of the entry, the entry will now be blank Alternate Flow: A4. (Optional) User can choose to select new photograph from devices camera reconfirmate entry into tag Basic Flow STN UC02 - to navigate to the entry of the specified date UC01 - the input of data will be using the same method Basic flow strip select need to edit text data in entry Select to edit Select to edit	Number								
Pre-conditions: Entries have been previously submitted for the users chosen date. Description: A user will be able to review a previous entry from the users chosen date. The us will be able to delete the whole entry for the users chosen date. The us will also be able to delete the whole entry. Any edits or deletions will be confirmed to be correct by the user when prompted by the app Basic Flow: 1. User navigates to the entry of the specified date 2. (Optional) User can choose to edit text data 3. (Optional) User can choose to select new emotion tag 4. (Optional) User can choose to select new enotion tag 6. (Optional) User can choose to select new location tag 6. (Optional) User can choose to select new photograph 5. (Optional) User can choose to select new photograph 7. After the user confirms the deletion of the entry, the entry will select the 'delete' button 7. After the user confirms the deletion of the entry, the entry will now be bland Alternate Flow: A4. (Optional) User can choose to select new photograph from devices camera re Related Use Cases: UC02 - to navigate to the entry of the specified date UC01 - the input of data will be using the same method Basic Flow STN Select to edit Select new motion tag 9 Select to edit Select to edit Select to edit 9 Otoggraph Select to edit Select to edit 9 Otoggraph	Goal:		User will be a	ble to review, edit or delete any/	'all data from p	revious entry			
Description: A user will be able to review a previous entry from the users chosen date. The us will be able to edit/delete any specified data within the entry as they chose. The will also be able to delete the whole entry. Any edits or deletions will be confirmed to be correct by the user when prompted by the app Basic Flow: 1. User navigates to the entry of the specified date 2. (Optional) User can choose to edit text data 3. (Optional) User can choose to select new emotion tag 4. (Optional) User can choose to select new location tag 6. (Optional) User can choose to select new location tag 6. (Optional) User can choose to select new photograph 7. After the user confirms the deletion of the entry, the entry will select the 'delete' button 7. After the user confirms the deletion of the entry, the entry will now be blank Alternate Flow: A4. (Optional) User can choose to select new photograph from devices camera ro Related Use Cases: UCO2 - to navigate to the entry of the specified date UCO1 - the input of data will be using the same method Basic Flow STN Select to edit 9 Select to edit Selects new or choose from case 9 Select to edit Selects new or choose from case 9 Select to edit Selects new or choose from case 9 Select to edit Selects new or choose from c	Pre-condit	tions:	Entries have l	peen previously submitted for the	e users chosen	date			
will be able to edit/delete any specified data within the entry as they chose. The will also be able to delete the whole entry. Any edits or deletions will be confirmed to be correct by the user when prompted by the app Basic Flow: 1. User navigates to the entry of the specified date 2. (Optional) User can choose to select new emotion tag 4. (Optional) User can choose to select new emotion tag 5. (Optional) User can choose to select new location tag 6. (Optional) User can choose to select new location tag 6. (Optional) User can choose to select new location tag 7. After the user confirms the deletion of the entry, the entry will select the r delete' button 7. After the user confirms the deletion of the entry, the entry will now be blank Alternate Flow: A4. (Optional) User can choose to select new photograph from devices camera ro Related Use Cases: UC02 - to navigate to the entry of the specified date UC01 - the input of data will be using the same method Basic Flow STN Diary Entry Screen of required Alter to edit Select to edit Iocation tag Selects on tag Selects on the Photograph entry - Take New or Choose from Choose from Cho	Descriptio	n:	A user will be	able to review a previous entry f	from the users of	chosen date. The user			
will also be able to delete the whole entry. Any edits or deletions will be confirmed to be correct by the user when prompted by the app Basic Flow: 1. User navigates to the entry of the specified date 2. (Optional) User can choose to edit text data 3. (Optional) User can choose to select new enotion tag 4. (Optional) User can choose to select new location tag 6. (Optional) User can choose to select new location tag 6. (Optional) If the user wishes to delete the entire entry, they will select the ridelete' button 7. After the user confirms the deletion of the entry, the entry will now be blank Alternate Flow: Related Use Cases: UC02 - to navigate to the entry of the specified date UC01 - the input of data will be using the same method Basic Flow STN Diary Entry Screen of required date Diary Entry Screen of required date Diary Entry Screen of required date Diary Entry Screen of required Cocation tag Select to edit Diary Entry Screen of required Cocation tag Dialogue box for type of photograph entry - Take New or Choose from Camera Roll Camera Roll			will be able to	o edit/delete any specified data v	vithin the entry	as they chose. The user			
to be correct by the user when prompted by the app Basic Flow: 1. User navigates to the entry of the specified date 2. (Optional) User can choose to select new emotion tag 3. (Optional) User can choose to select new emotion tag 4. (Optional) User can choose to select new location tag 6. (Optional) User can choose to select new location tag 6. (Optional) User can choose to select new location tag 6. (Optional) User can choose to select new location tag 7. After the user confirms the deletion of the entry, the entry will now be blank Alternate Flow: A4. (Optional) User can choose to select new photograph from devices camera reference Related Use Cases: UC02 - to navigate to the entry of the specified date UC01 - the input of data will be using the same method Basic Flow STN Select to edit VC01 - the input of data will be using the same method Basic Flow STN Select to edit VC01 - the input of data will be using the same method Basic Flow STN Select to edit Very Screen of required date Ust of selects new into tag Very Screen of required date Select to edit Very of photograph entry Select to edit Very of photograph entry Select to edit Very of photograph entry Select to edit			will also be al	ple to delete the whole entry. An	y edits or delet	ions will be confirmed			
Basic Flow: 1. User navigates to the entry of the specified date 2. (Optional) User can choose to edit text data 3. (Optional) User can choose to select new emotion tag 4. (Optional) User can choose to select new location tag 5. (Optional) User can choose to select new location tag 6. (Optional) If the user wishes to delete the entire entry, they will select the 'delete' button 7. After the user confirms the deletion of the entry, the entry will now be blank Alternate Flow: A4. (Optional) User can choose to select new photograph from devices camera recombined Related Use Cases: UC02 - to navigate to the entry of the specified date UC01 - the input of data will be using the same method Basic Flow STN Select to edit entry fabre fabre for available emotion tag 9 Select to edit location tag 9 <td></td> <td></td> <td>to be correct</td> <td>by the user when prompted by t</td> <td>he app</td> <td></td>			to be correct	by the user when prompted by t	he app				
 2. (Optional) User can choose to edit text data 3. (Optional) User can choose to select new emotion tag 4. (Optional) User can choose to retake photograph 5. (Optional) User can choose to select new location tag 6. (Optional) If the user wishes to delete the entire entry, they will select the 'delete' button 7. After the user confirms the deletion of the entry, the entry will now be blank Alternate Flow: A4. (Optional) User can choose to select new photograph from devices camera received and the entry of the specified date UC02 - to navigate to the entry of the specified date UC01 - the input of data will be using the same method Basic Flow STN 	Basic Flow	/:	1. User nav	igates to the entry of the specifie	ed date				
 3. (Optional) User can choose to select new emotion tag 4. (Optional) User can choose to retake photograph 5. (Optional) User can choose to select new location tag 6. (Optional) If the user wishes to delete the entire entry, they will select the 'delete' button 7. After the user confirms the deletion of the entry, the entry will now be blank Alternate Flow: A4. (Optional) User can choose to select new photograph from devices camera rc Related Use Cases: UC02 - to navigate to the entry of the specified date UC01 - the input of data will be using the same method Basic Flow STN 			2. (Optiona	I) User can choose to edit text da	ita				
 4. (Optional) User can choose to retake photograph 5. (Optional) User can choose to select new location tag 6. (Optional) If the user wishes to delete the entire entry, they will select the 'delete' button 7. After the user confirms the deletion of the entry, the entry will now be blank Alternate Flow: A4. (Optional) User can choose to select new photograph from devices camera rc Related Use Cases: UC02 - to navigate to the entry of the specified date UC01 - the input of data will be using the same method Basic Flow STN 			3. (Optiona	I) User can choose to select new	emotion tag				
 S. (Optional) User can choose to select new location tag 6. (Optional) If the user wishes to delete the entire entry, they will select the 'delete' button 7. After the user confirms the deletion of the entry, the entry will now be blank Alternate Flow: A4. (Optional) User can choose to select new photograph from devices camera ro Related Use Cases: UC02 - to navigate to the entry of the specified date UC01 - the input of data will be using the same method Basic Flow STN Basic Flow STN 			4. (Optiona	I) User can choose to retake phot	tograph				
 6. (Optional) If the user wishes to delete the entire entry, they will select the 'delete' button 7. After the user confirms the deletion of the entry, the entry will now be blank Alternate Flow: A4. (Optional) User can choose to select new photograph from devices camera row select low select new photograph from devices camera row select low select low select low select low select new photograph from devices camera row select low select low select new photograph from devices camera row select low select low select low select new photograph from devices camera row selects new photograph from tage for the emotion tag for the select new photograph for the s			5. (Optiona	I) User can choose to select new	location tag				
'delete' button 7. After the user confirms the deletion of the entry, the entry will now be blank Alternate Flow: A4. (Optional) User can choose to select new photograph from devices camera ro Related Use Cases: UC02 - to navigate to the entry of the specified date UC01 - the input of data will be using the same method Basic Flow STN 			6. (Optiona	I) If the user wishes to delete the	entire entry, tl	ney will select the			
 After the user confirms the deletion of the entry, the entry will now be blank Alternate Flow: A4. (Optional) User can choose to select new photograph from devices camera row for the specified date UC01 - the input of data will be using the same method Basic Flow STN Basic Flow STN Basic Flow STN Diary Entry Screen of required date 			'delete' k	putton					
Alternate Flow: A4. (Optional) User can choose to select new photograph from devices camera ro Related Use Cases: UC02 - to navigate to the entry of the specified date UC01 - the input of data will be using the same method Basic Flow STN Basic Flow STN Diary Entry Screen of required date Diary Entry Select to edit location tag Select to edit location tag Dialogue box for type of photograph Choose from Choose from Choose from Choose from Choose from Camera Roll			7. After the	user confirms the deletion of the	e entry, the ent	ry will now be blank			
Related Use Cases: UC02 - to navigate to the entry of the specified date UC01 - the input of data will be using the same method Basic Flow STN	Alternate	Flow:	A4. (Optional) User can choose to select new p	photograph froi	m devices camera roll			
UC01 – the input of data will be using the same method Basic Flow STN Diary Entry Screen of required date Diary Entry Screen of required date Diary Entry Screen of required cation tag Select to edit location tag Select to edit location tag Dialogue box for type of photograph entry – Take New or Choose from Camera Boll Camera Boll Camera Boll Control Con	Related U	se Cases	: UC02 - to nav	UC02 - to navigate to the entry of the specified date					
Basic Flow STN Select to edit select to edit emotion tag Edit text data in entry Screen of required date Select to edit location tag Select to edit bialogue box for type of photograph entry – Take New or Choose from Camera Boll Camera Boll Camera Boll Cist of available emotion tag Selects an emotion tag Data inputted into entry Data inputted into entry Select so edit Select to edit Select			UC01 – the in	UC01 – the input of data will be using the same method					
Press and hold entry wishing to delete delete deletion Press and hold entry Confirmation dialog of deletion Confirm deletion journal	Basic Flow	/ STN	Diary Entry Screen of required date	Select to edit emotion tag Edit text data Select to edit location tag Select to edit location tag Select to edit photograph Press and hold entry Wishing to delete	selects an em a in entry a in entry a in entry select locat e box for hotograph ike New or se from tra Roll re Confirm del	botion tag Data inputted into entry ects to add pects to add w photograph w photograph Entry removed from journal			

Use Case UC04 Us		Us	se Case Name:	Add new event	Rating:	Must Have	
Goal:			User will be al	ple to add a new event for their	specified date		
Pre-conditi	ons:		-				
Descriptior	1:		The user wish user wishes to	es to add an event (eg birthday have a countdown for the eve	, anniversary) o nt displayed	n a specified date. The	
Basic Flow:			 User navigates to home page via the navigation bar User selects day they wish to enter the event on the calendar The entry for the date selected will be displayed. User selects the "Today's Events" button User selects to "Add new event" from the Events today screen User completes form for new event (including event name, date of event, time of event, event location) If user requires countdown to be displayed for this event the "create 				
Alternate F	low:		A2 - A4. User	selects the "+" button on the up	coming events		
Related Use Cases:			-				
			Home Page Screen Selec required from cale	t date ndar Screen of required date Select "Today's Events" button Select "Add N Event	een date to lew t ["] Event entry screen Complet form for event	e Event created for specified date	

Use	UC05	Us	e Case	Add a To-Do task for specified	Rating:	Must Have
Case		Na	ime:	date		
Number						
Goal:			User can add a	a task in their to-do list for a speci	fied date	
Pre-condi	tions:		-			
Descriptio	on:		The user woul	d like to add tasks to their to-do l	ist for a specifie	d date, these tasks
			can have remi	nders set as the user requires.		
Basic Flov	v:		1. User navi	gates to the Home Page via the na	avigation bar	
			2. User seled	cts day they wish to enter the tasl	on the calenda	r
			3. The entry	for the date selected will be disp	layed. User sele	cts the "To-Do
			Today" bu	utton		
			4. User selec	cts to "Add new task" from the To	-Do today scree	n
			5. User com	pletes form for new task (includir	ig task name and	d date)
			6. If the use	r would like a reminder for the tas	sk, they will che	ck the option to do
			so, and in	put a time for reminder (this rem	inder is set for t	he same date as the
			task)			
			7. User seled	cts the option to repeat task, user	completes all re	equired fields
			including	frequency of reminder and if a tir	ne reminder is r	equired
Alternate Flow:			A2 -A4. The us	ser selects the "+" button in the To	p-Do Today sect	ion
Related Use Cases:			-			
			Home Page Screen Select required of from caler	date hdar Screen of required date Select "To-Do Today" button To-Do Scree for given da Select to "Add Ne Task"	en te o W To-Do Tasks Screen Complete form for Task	Task added for specified date

Use Case	UC06	Use	Case Name:	Analys	e diary entries via user	Rating:	Should Have
Number			r	specifi	ed search terms		
Goal:			User can ente	er search	n terms (text terms, emot	ions, locati	ons) to find analysis
			of all entries	that ma [.]	tch this term		
Pre-condition	ns:		Multiple entr	ies have	been previously entered	with know	n terms written in
			the text, emo	otions or	locations		
Description:			The user will	be able	to search terms written ir	n diary entr	ies, emotions tagged
			for entries or	location	n names tagged. The resu	lts of these	searches return all
			entries that include these terms, showing the emotion tags relating to these				
			entries				
Basic Flow:			1. User navigates to the "Search Plus" analysis page via the navigation bar				
			2. User will enter their search terms into the search bar				
			3. The results of the search are displayed for the user to review				
Related Use Cases:			-				
Basic Flow STN							
			Analysi: Scre	s Page en	Input search term and se	earch	Results of search displayed
							approyed

Use Case	UC07	Use	Case Name:	View location map and	Rating:	Should Have
Number			·	analyse		
Goal:		The user can review all locations previously tagged locations on one map, with				
		extra details available to review				
Pre-conditions:		Multiple entries have been previously entered with a location tag				
Description:		A map of all previously tagged locations is displayed, selecting a marker on the				
		emotion tag.				
Basic Flow:		1. User navigates to the "Search Plus" analysis page via the navigation bar				
-		2. User navigates to the map analysis page from the "Map All Entries" button				
			in the "Se	arch Plus" analysis page		
			3. User can	review all locations displayed	on map	
			4. On select	ing one of the markers, the us	er can review	entries that have this
			location t	agged, the emotions tagged fo	or these entrie	s can be reviewed
			5. Selecting	a diary entry will navigate the	user to this er	ntry, showing all data
Delated Lice (Included	with this entry		
Related Use			-			
Basic Flow STN		Analysis Page Scree Select All Er	en : "Map atries" Location Map Screen, pinpoints on all locations tagged Select pinpoint on map	Relating entri pinpoint sele displayed Select entr from list	es of cted y Diary Entry Screen of required date	

5. Design

5.1. Prototype

To ensure the application will be functional in the sense that the user will be able to navigate through and access all activities available on the app, a prototype was created to show a mock-up of how the user will interact with the system to get their intended results.

These designs were created with the functional requirements and the use cases in mind to make sure every requirement has the ability to be carried out and the use cases show how the user is expecting to interact with the system, showing the results as expected

Due to some restrictions on the programme used to create the designs some aspects have not been visually achieved, but care has been taken to consider these and they will be outlined as required, giving a description of any extras that will be implemented.

5.1.1. Initial Prototype

PTS (prototype screen) number

Photos of today Photos of today Continue Continue Continue Continue </th <th>PTS1 - Diary Entry This is the main screen for the diary entry. This is the screen that will be displayed when the app is first opened. The entry for the current date will be accessible easily via the navigation bar. Within this screen the user will be able to enter their entry, including; text data, emotion tag, photo, location tag If an entry has previously been written for a specified data, all relating data will be displayed in the respective containers Use Case(s): UC01, UC02, UC03, UC04, UC05, UC06</th>	PTS1 - Diary Entry This is the main screen for the diary entry. This is the screen that will be displayed when the app is first opened. The entry for the current date will be accessible easily via the navigation bar. Within this screen the user will be able to enter their entry, including; text data, emotion tag, photo, location tag If an entry has previously been written for a specified data, all relating data will be displayed in the respective containers Use Case(s): UC01, UC02, UC03, UC04, UC05, UC06
How did you feel today?	 PTS2 - Emotion Selector An example of how the user will select the emotion tag to be linked to the current entry. The user selects the emoji that relates to how they have felt this day. Use Cases(s): UC01, UC02, UC03

Photo Selection Toke Photo Toke Photo From Camera Roll	PTS3 - Photo Selector The dialog that will be displayed when the user selects to add a photo to the entry. The user has the choice to take a new photo, or select one from the device's camera roll Use Case(s): UC01, UC02, UC03 PTS4 - Home Page This page allows the user to find and review previously entered diary entries. The user can select a date from the calendar and will be taken to this date's diary entry in the 'Diary Entry' screen. If an entry has been made for this event, a small icon will be displayed on the date on the calendar to show the user an entry has been made for this date. The 'Today's To-Do' container displays any tasks the user has entered for the current date; the user will be able to check off any task form this list. The user can also select to add a new task by selecting the "+" button relating to this list. Selecting this will take the user to the 'Add new To-Do Item' screen The 'Countdowns' container displays any event the user has selected to have a countdown for, the list shows each event name, with the months, weeks and days left until the event. The user can select to add a new event by selecting the "+" button relating to this list. Selecting this will take the user to the 'Add new roent' screen
	Use Case(s): UC04, UC05, UC06

To-Do for 22nd February 2019 Complete Coursework Leave house for work Leave house for work Dook Theatre Tickets Phone Grandma Pet Dog Take Medicine 1800 Add New Task	PTS5 - To-Do Today List Screen to display the to do tasks the user has set for the given date. User can check off any completed task, the task will remain in the list but will be shown as completed. If a task has been set as needed a timed reminder, this will be shown as an icon relating to the task Selecting a task from this list the user can choose to edit or delete a specific task From this screen the user can choose to add new tasks for the current date Use Case(s): UC05
To-Do Creator Task Name: Date: / Ø Reminder for task? Time for reminder: Save Clear Delete	PTS6 - Add new To-Do item This is the dialog that will be displayed when a user selects to enter a new To-Do task. The user will complete the fields in the form as required, task name and date are compulsory, if required the user can check the 'Reminder for task' checkbox where they will then be prompted to input the time the user would like to be reminded. Selecting save, if the task name or date has not been inputted the user will be prompted to complete these fields until they are able to save the task. If a user selected to edit a previously entered task, this dialog will be displayed with all the relevant data displayed in the respective containers, the user can edit the data, or delete the task completely Use Case(s): UC05

Contractions for 22nd February 2019 Dog's Birthday New Sofa Arriving Add New Event Contractions	PTS7 - Events Today List The events screen displays any events previously entered for the given date. Selecting an event from this list the user can choose to edit or delete a specific event From this screen the user can choose to add a new event for the current date Use case(s): UC04
Event Creator Event Name: Date of Event: / / Time of Event: Event Location: Create Countdown for event? Create Countdown for event?	PTS8 - Add new event This is the dialog that will be displayed when the user chooses to enter a new event. The user will enter all required fields. The event name and date will be compulsory fields, time of event, event location and whether the user would like to have a countdown for the event are optional fields. Selecting save, if the event name or date has not been inputted the user will be prompted to complete these fields until they are able to save the event. If a user selected to edit a previously entered event, this dialog will be displayed with all the relevant data displayed in the respective containers, the user can edit the data, or delete the event completely
	Use Case(s): UC04

	PTS9 - Repeat Reminders List
Repeat Reminders Take Medicine Bins Out Choir Rehersal Leave house for work Create new reminder	This screen displays all repeat reminders the user has entered, showing the name, frequency of task and the time the user requires a reminder The user can select to add a new repeatable reminder from this screen Use Case(s): UC06
Reminder Creator Name: Time for reminder: Days to remind Monday Tuesday Wednesday Thursday Friday Saturday Starting from: / / Ending on: / Delete	PTS10 - Add new repeatable reminderThis is the dialog that will be displayed when the user selects to add a new repeat reminder.The user will enter all required fields. All fields in this form are required to create the new reminder.Selecting save, if any field has not been inputted the user will be prompted to complete these fields until they are able to save the event.If a user selected to edit a previously entered reminder, this dialog will be displayed with all the relevant data displayed in the respective containers, the user can edit the data, or delete the reminder completelyUse Case(s): UC06


Heuristic Evaluation

Using Nielson's 10 principles for User Interface Design (Nielsen, J 1994a), the following show if these current prototype designs follow the heuristics outlined.

I will be conducting this evaluation, as finding users is impractical. I will ask for advice and recommendations on what potential users may want to change, but as a first prototype, time is not available to have a full evaluation from a user

Severity Ratings (Nielsen, J 1994b)

- 0. I don't agree that this is a usability problem at all
- 1. Cosmetic problem only: need not be fixed unless extra time is available on project
- 2. Minor usability problem; fixing this should be given low priority
- 3. Major usability problem; important to fix, should be given high priority
- 4. Usability Catastrophe; imperative to fix before product can be released

Heuristic Evaluation ID	HE01	
Task/Problem	Emotion tags not labelled in emotion selector	
Relating Screen(s)	PTS2	
Severity Rating	3	
Applicable Heuristics	Recognition rather than recall	
Problem Description	As no labels are given to the user on what each emoji is, there may be confusion on what one emotion may be. When adding a tag, the user would be able to always choose the same tag for the same emotion without having to recall what each emoji meant When analysing or searching for entries the user might not use the correct search term for their chosen tag, the users name for an emotion may not match the name given by the application	
Alterations	Label the emotion tag images with their tag name	

Heuristic Evaluation ID	HE02	
Task/Problem	Repeat Reminders not accessible from anywhere in the app	
Relating Screen(s)	PTS9	
	PTS10	
Severity Rating	5	
Applicable Heuristics	User control and freedom	
	Flexibility and efficiency of use	
Problem Description	Currently in this design the repeat reminders screen and option to add a new	
	repeat reminder is not available to the user.	
Alterations	As there is no direct or obvious link to reach this list. Combining the to-do entry	
	and repeat reminders entry will be more useful for the user,	

Heuristic Evaluation ID	HE03	
Task/Problem	User can only enter one entry per day	
Relating Screen(s)	PTS1	
Severity Rating	4	
Applicable Heuristics	Flexibility and efficient of use	
	User control and freedom	
Problem Description	The user can only enter one diary entry per day, this includes only one emotion	
	tag, one photo and one location tag. A user may want to include multiple	
	emotion tags, locations or photos for different events during their day	
Alterations	Allow the user to enter multiple entries per day	
	These multiple entries should be available to view from one page per day, and	
	the user can scroll through them and edit/delete any data within them	

Heuristic Evaluation ID	HE04	
Task/Problem	Event information not available after entry	
Relating Screen(s)		
Severity Rating	3	
Applicable Heuristics	Consistency and standards	
Problem Description	After the user has entered the information into the Event entry page (including	
	event location and time) this is not available for the user to see in the events	
	main page	
Alterations	Have this information included in the event list	

Heuristic Evaluation ID	HE05	
Task/Problem	Unclear icons for navigation bar	
Relating Screen(s)	All	
Severity Rating	3	
Applicable Heuristics	Match between system and real world	
	Recognition rather than recall	
	Help and documentation	
Problem Description	The icons used in the navigation bar may be unclear to users. The Home Page	
	icon does not match the users recognised icon of a house.	
Alterations	Change the icons to match the user's expectations	
	Add labels to the icons to give hints to what screen will be navigated to from this	
	button	

Heuristic Evaluation ID	HE06	
Task/Problem	Consistency between name labels for data entry	
Relating Screen(s)	PTS6	
	PTS8	
	PTS10	
Severity Rating	2	
Applicable Heuristics	Consistency and standards	
Problem Description	The labels used to indicate the data that should be entered into the entry	
	screens is not consistent in the way they are written, for example Task Name,	
	Name and Event Name. Date of Event, Date and Date of Task	
Alterations	Keep the labels consistent by having "Name of", "Date of"	

Heuristic Evaluation ID	HE07	
Task/Problem	No confirmation of edits or deletions	
Relating Screen(s)	All	
Severity Rating	4	
Applicable Heuristics	Error prevention	
	Help users recognise, diagnose and recover from errors	
Problem Description	When a user makes an edit, saves or deletes any data there is currently no	
	prompts or confirmation dialogues that confirm the users action	
Alterations	For deletions a confirmation dialog should be shown asking the user to confirm	
	the action to allow for recovery from accidently button presses	

Heuristic Evaluation ID	HE08	
Task/Problem	Unclear use for emotion graph	
Relating Screen(s)	PTS11	
Severity Rating	2	
Applicable Heuristics	Aesthetic and minimalist design	
Problem Description	The graph is currently unclear on the use or need for it.	
Alterations	As the new design will make use of multiple entries per day, the graph may not	
	be suitable for this data, it would therefore be more suitable to remove this	
	graph from the application	

5.1.2. Improved Prototype

Taking into account all issues found during the heuristic evaluation, some screens needed to be redesigned to ensure the alterations were carried out fully. The following is the alterations that have been made to the prototype design to ensure the issues raised have been solved.



How did you feel today? Happy Loved Sad And	ngry Scared Meh	Emotion Selector Screen To improve the recognisability for this screen, the emojis have been labelled to let the user know what each emoji represents. This will allow them to later search the correct term for the emotion tag to return all entries that have been tagged with the specified emotion tag
Related Heuristic ID H	HE01	

		To-Do Task Creator
To-Do Task Creator Task Name: [Task Date [Add reminder	/ /	This entry screen has combined the To-Do Entry and Repeat Reminders Entry screens to allow users a more minimalist design, reducing the clicks needed from a user to add both a 'one off' to-do task and a repeated to-do task.
Time for Reminder	ask? ay Dednesday Saturday Denday	The user can check the boxes to indicate if they would like a timed reminder for the task. And they can check the box to indicate this task is to be repeated when they specify. When either the 'add reminder' or 'repeat task' checkboxes are checked the relating fields are shown to the user. Keeping the data available to the user to only that that relates to what the user wants to achieve, reduces the potential errors that could be made and keeps the design minimal. Any task entered here will be added to the respective dates to-do list. If a task is to be repeated, the task will automatically appear on the future dates to-do lists as required
Related Heuristic ID	HE02 HE06	
		For a list





During this prototype design it was decided that the name of this application will be "everyDaybook", an icon and logo has been designed as follows:



This will be used as the launcher icon for the app and will then be incorporated where appropriate throughout the application to tie it all together and keep the consistent theming.

5. Implementation

The following section details how the system has been built. Extracts of relating code and screenshots from the app are provided to aid in the explanations of how each feature was implemented into the application. Where necessary if a feature has not been successfully implemented or where improvements could be made to a feature, explanations of why this feature is not up to the standard hoped from this system have been given.

5.1. Database

Using MongoDB Atlas Stitch as detailed previously, the application is able to communicate with the database quickly and without the need for extensive storage of the diary entries within the device the application is installed on.

The following is an example of the schema that has been used, as this is a MongoDB database, the schema is flexible and has allowed for users to be able to input multiple entries per day that can include or exclude which ever fields they require for each individual entry

The basic schema for the diary entry is as follows, an example of this schema used in the database with test data is also shown:

{ id : ObjectID owner_id : String date : String	_id:ObjectId("5cab4c73c0f86629b235ff70") owner_id:"5c98d67779f87ab5406c6a8b" date:"Mon, 08 Apr, 19"
<pre>entries : Document { [text : String emotion : String photo : GridFS Object longitude : String latitiude : String location : String</pre>	<pre>v entries: Array v 0: Object text: "test 8th april: Today I went to the shops" emotion: "Happy" photo: "photo" longitude: "20.20" latitude: "0.1278" location: "Location: "Location: "LK"</pre>
] } toDoTasks : Document { [taskName : String checked : Boolean reminderTime: String] } }	<pre>> 1: Object > 2: Object > toDoTasks: Array</pre>

Collection: diaryEntries

Each device that uses this application is set an owner_id, only the relating owners can see their entries. Each entry is identified by the entry date. For each date, there is an array of documents available for each entry, this allows for easy access to each entry for reviewing and editing from the application. Each entry can contain text data, an emotion tag, a photo object, the latitude and longitude coordinates of the location tag and the street address name of the location tag. The data types for each element have been chosen to ensure easy access and manipulation within the system. The photo element within each entry was not implemented in time and is therefore not a GridFS object and a placeholder of String is set instead, the details of this will be discussed later in the Evaluation.

The to do tasks for each date are also contained in an array of documents in 'toDoTasks'. Each task can have a task name, the 'checked' field relates to if this item has been ticked off on the checklist inside the application. If the user requires a reminder for the task, this is stored in the 'reminderTime' field.

For the events stored for the journal, another collection was made. This allowed all events to be accessible via one array, for easy access within the application. An alternative way the events could have been stored was to store a document of events within the document of the specified date of the event. This would have resulted in a much larger overhead to search every document within the diaryEntries collection to retrieve any inputted events. Instead having the separate collection allowed for less overhead in searching the database to find the required events and allowed for easier computation of events countdowns.

```
_id: ObjectId("5c98e5001c9d440000766ef4")
                                                              owner id: "5c98d67779f87ab5406c6a8b"
                                                            v events: Array

    0: Object

  {
                                                                   eventName: "Bob's Party"
        id : ObjectID
                                                                   eventDate: "10/05/19"
       owner id: String
                                                                   eventTime: "15:35"
       events : Document {
                                                                   eventLocation: "Bobs House"
            ſ
                                                                   eventCountdown: false
                  eventName : String
                                                               v1: Object
                  eventDate : String
                                                                   eventName: "New Years Eve"
                                                                   eventDate: "31/12/19"
                  eventTime : String
                                                                   eventTime: "00:00"
                  eventLocation : String
                                                                   eventLocation: "Hilton Hotel"
                  eventCountdown : Boolean
                                                                   eventCountdown: true
            1
                                                               > 2: Object
       }
                                                               > 3: Object
                                                               > 4: Object
 }
                                                               > 5: Object
                                                               > 6: Object
                                                               > 7: Object
Collection: eventEntries
```

To relate the events to the correct journal, the owner_id field acts as a reference key, to ensure only the

owner of the journal currently accessed has access to the relating entries and events for their journal. As shown in this Entity Relationship Diagram.



5.1.1. UML Class Diagram

The following diagram shows the basic structure of all classes within the system. All crucial variables and methods within each class are shown, to simplify the diagram some variables and methods have not been added.

Within all main classes there is also the option to navigate to and therefore instantiate MainActivity, HomePage and AnalysisHome. This is dues to the navigation bar accessible from most activities, but again to simplify the diagram these instantiations have not been added



5.1.2. Activity Diagrams

To further show how each class can access all others and to show the available activities within each of the three main sections of the system, activity diagrams have been created.

As previously mentioned, the ability to access each of the three main sections via the navigation bar have not been added to the diagrams unless stated otherwise. Within the navigation bar there is also the option to return to the previous activity using the 'Back' button, but for simplicity this has not been included in any of the diagrams.

5.1.2.1. Diary Entry Activities



5.1.2.2. Home Page Activities



5.1.2.3. Search Plus Activities



5.2. System

5.2.1. Database Retrieval

The original design of this system did not include a login system. The system originally started up directly to the diary entry screen, using the owner_id to find the appropriate collection and creating a new entry for today's date. This collection can only be found once the database has been retrieved, in most home or private internet connections, the database is retrieved quickly and the so the system can find all entries without the user noticing a long wait. However, when the system was tested using the university wi-fi an issue arose that the system tried to complete other activities that used the data retrieved from the database, before the data was completely retrieved. This issue is supposedly due to the firewalls and blocks the system needs to go through before the database can be accessed that is not normally present (to such a

large extent) on home networks. To overcome this, the idea of a login screen was found, this will act as both security to access the user's private diary entries and as a buffer for the system to retrieve all entries before trying to use them within the activities. Even with this login buffer screen, there is still a slight delay, so here a progress alert dialog is shown, where the user is told that the system is retrieving their entries. This ensures the user does not try and access any activities the system is not ready to run until all data is found.

The idea of the login system will also allow for future updates in the systems database management. The logins allow for different users to have access to the system, which will mean each user could store their entries within their own collection within the MongoDB database. This would allow for quicker retrieval and searching as the system would only need to search through the individual user's collection. But due to finding the above issue close to the deadline, the simple solution given was enough to ensure the system will work even for protected internet connections.

5.2.1.1. Login/password reset

When the user accesses the application for the first time, they will be shown the 'Create User Login', here they will be asked to create a new login with username and password. Once created the user can start creating their entries as they choose.

When they next access the application, their login details will be automatically pulled. Their username will be set in the login screen and the user will be asked to confirm their passcode in order to access the entries. Due to this feature being implemented close to the deadline as explained above, currently the application pulls the password as plaintext and compares it to the plaintext string inputted by the user. In order to increase security, it would be suggested for future work to encrypt these passcodes and compare the encryptions rather than the plaintext.

To allow the user to reset their passcode if they choose, the option is available by selecting the 'Settings' button in the top right corner of the 'Diary Entry' screen, the 'Home Page' screen and the 'Search Plus' screen. From this the user will have to re-enter their old passcode as verification, and then they can reset to a new passcode of their choosing.

5.2.2. Navigation



Navigation is available throughout all main screen of the application via the navigation bar, found at the bottom of the screen. In



order the buttons are: "Back", "Todays Entry", "Home" and "Search Plus"

Shown here is how the navigation bar looks within the app. When not selected, the icons appear greyed out and the title is not visible. When selected the icons are coloured and the title of the screen the user is navigating to is displayed, as shown by the highlighted 'Back' button.

On all screen (except the Diary Entry screen as explained below) the 'Back' button simply returns the user to the last actively used screen by finishing the currently running activity.

The 'Todays Entry' button will navigate the user to the Diary Entry screen. Using intents, the current date is passed to the Diary Entry screen so todays entries can be found

The 'Home' button will start the Home Page using intents, as does the "Search Plus" button opening the Search plus screen.

Within the Diary Entry screen, the 'Back' button has additional safety catch where it prompts the user to check that all entries have been saved. Currently this is a brute force way of minimising errors as the program does not see if any data has been changed, it just assumes that it has and asks the user to save all data. The user can ignore this alert if all data has been changed or they do not wish to save the data by selecting the 'Continue' option in the alert dialog



5.2.3. Diary entry

All methods used to retrieve and handle diary entry data are found in the MainActivity class

When starting this activity, which activity it has been called from is found using Intents.



The method looks for a string passed through the intent that represents the date that has called the method. If there is no such string passed through, this means the diary entry for the current date is being called, and the *date_of_entry* will therefore equal the current date. The string passed through the intent is used when an entry is required for a particular date, for example when a date is selected on the calendar or the user wishes to expand the entry found through a search or event list.

Originally this system was set up differently. The diary entry screen was used by two activities, one for the current diary entry called '*MainActivity*' and a second for retrieving the entries from a date selected from the calendar called '*EntryGivenDate*. This was changed because the original code for retrieving the entries for the given date did not update correctly when the date changed, and it was found that both types of entry retrieval (todays entries and given date entries) could be accessed through this one class.

The system instead calls the *dbRetrieveFind()* method which uses the find() method with MongoDB's *RemoteFindIterable* variable. This method uses the given date (either the current date or date passed by an intent) to find any documents that match this date. If the task finds a document, all required data is retrieved from it including all entries and all to-do tasks for this date. As previously mentioned, events are stored in a separate collection and therefore are not affected by these document retrievals

If multiple documents are returned, there has been an error with previously inputted entries. This is a very rare occurrence due to the verification and validation steps being used within the system to avoid this. However, as a precaution, the system deals with this by only retrieving the first document found. This is a brute force approach as it does not check which entry is the most appropriate, but will allow the system to continue to work if a previous error had been made somehow.

If no document is found, this means that the given date has not previously been accessed by this user. The system will therefore create a document for this date, to allow for data to now be inputted for this date including any diary entries or to-do tasks.



Once the document has been retrieved or create the system now sets the appropriate fields ready for user entry. The method *SetEntries()* is called. This method creates the arrays required to be passed to the recycler view to show the entries appropriately. The arrays (*enTexts* etc.) are set to the size of the number of entries that have been found within the document. For each entry within the document all the entry data fields are retrieved and added to their respective arrays. If there are no documents retrieved, the arrays are given a blank

string as a placeholder to be read by the recycler view. Once all entries have been processed the recycler view is called, passing the arrays to it, to display appropriately in the main list.



To implement the multiple entries per date design a recycler view was used to allow the list to update dynamically. As the amount of data stored in each row can become guite large, the recycler view can handle this amount of data better than a standard list view and adapter. The View Holder helps to improve the performance of the system by keeping the views of each element within the row constant, and not re-finding the view each time the row is accessed. The use of the recycler view was a late addition to the project, originally an adapter was used, however this soon ran into the issue that the data within each row was not updating correctly or frequently enough, so data was often out of date to the users input. These issues were time consuming to solve and caused problems with other activities within the project by supplying them with incorrect data. But once this was updated to use the recycler view, all data was correctly updated, and other activities using this data could all run more smoothly.

Unfortunately, due some issues with using and debugging MongoDB Atlas, the entries cannot be deleted. The system has implemented the feature to delete an entry by using an *onLongClickListener* where the user would be able to long click select a row in the entries list, and once confirmed the deletion in the dialog that is presented, the system then attempts to delete the selected row. However, all methods attempted to remove the document within the entries array (within the database) were unsuccessful. An example of a method used is shown below, where the document of the entry is attempted to be pulled from the array. But as this entry is a document, within an array, within the parent document, there were some issues accessing and removing the correct entry. As this project works with MongoDB Atlas, there is little in the way of debugging available to see where the process is being run and on what data. When running this method shown, the only feedback from the database was that one document was modified, but as far as can be seen, no documents were removed or updated.



5.2.3.1. Add New Entry

To add a new entry for the given date (i.e. to add another row to the given date's entries list) the user will select the "Add New Entry" button. The original design for adding a new row included saving a new empty document within the dates entries array in the database, however this meant that if a user added multiple new rows, without inputting any data to them, even without saving the next time this date's entries were accessed the empty rows would still be shown in the list. To overcome this this method now keeps track of a variable called *extraRows*. This variable is increased by one every time the user selects the "Add New Entry" button, after this increment the *dbRetrieveFind()* method is called again, which will call the *SetEntries()* method when the entries are retrieved. As mentioned, the arrays created are set to the size of the number of entries retrieved, however, to allow more rows to be added, the size is increased by the amount of *extraRows*.



This means if the new row is not saved, when the user returns to the date of the entry, there will only be the number of previously saved entries and no blank additional rows will appear.

This does mean however that some data may be lost if the user inputs or edits data in a row

and then selects to "Add New Row", as the entries are reset to the arrays that have been saved and retrieved by the *dbRetireveFind()* method. To avoid this error of losing data an alert dialog is set to remind the user to save all entries required before adding a new row.

5.2.3.2. Emotions



The emotions were implemented to keep the recognisability and simplicity as determined during the heuristic evaluation of the prototype. To ensure this, a simple list design was used, using the combination of the emoji image and the relating string to allow users to recognise and link the two relating items together. This will also allow users to later search the correct term for the emotion tag to return all entries that have been tagged with the specified emotion tag. Or when reviewing analysis of an entry will know which emotion is being referred to.

When the user selects an emotion from the list, the relating emoji image is displayed within the entry. If the user selects the "Remove Emotion Tag" button the emotion tag will be set back to the default "empty" emoji face and no tag will be saved will the entry.

Currently the user is limited to select from a list of 8 emotions, these were based on the 8 basic human emotions (Gautam, S, 2011) the emotions were also chosen on the availability of emoji images and whether these emotions could be portrayed accurately through an emoji.

Having a limited number of emotions, results in all code relating to these emotions being hardcoded.

The images of emojis were all found through JoyPixel, that has a free licence to use the images within personal projects. All images available from www.joypixels.com.

5.2.3.3. Locations

The location tags for the entry is completed using Google Maps API. In order to use Google Maps API, an API key had to be obtained using console.developers.google.com. This key is restricted to Android apps only and uses this projects SHA1 key to restrict the use this this app. Maps SDK for android and Places API enabled to allow the application to use the Google Maps method and be able to return location addresses when the user searches for a location string, selects points on the map or uses current user location.

When the user selects the location tag field within the entry, an activity is started using the *startActivityForResult* method with an Intent. The intent will take any current location tag that may have

Intent mapIntent = new Intent(packageContext: MainActivity.this, LocationSetter.class); boolean locationData = false; String locationCur = String.valueOf(viewHolder.location.getText()); String latitudeCur = String.valueOf(viewHolder.longHolderTemp.getText()); String latitudeCur.equals("")) { mapIntent.putExtra(name: "longitude", longitudeCur); mapIntent.putExtra(name: "location", locationCur); locationData=true; } mapIntent.putExtra(name: "locationData", locationData); startActivityForResult(mapIntent, REQUEST_NEW_LOCATION); currentView=v; locationView=viewHolder.location; latHolder=viewHolder.location; latHolder=viewHolder.location; longHolder=viewHolder.longHolderTemp; been previously entered for this entry and passes it to a new activity called LocationSetter. Before the map can be displayed the system will gain permission from the user that location services may be used by this application. This includes allowing current location to be found and map locations can be searched. If permission is granted by the user, in the LocationSetter class, any coordinates passed through from the intent are found and mapped on the Google Map MapView set in the layout once the map has been initialised. If no coordinates have been passed an



empty map (without any pins) is displayed.

There are many ways the user can pin a location on the map. The first is they can set the location as their current location. Here the system checks the permission to access user location has been granted, if it has the system will use the *getLastLocation()* method to find the last known location of the device.

The Geocoder class is used to get the address of the location, this is used as the location name to be recognised by the user instead of using coordinates. To give the user more flexibility when a location name is found, it will be displayed to the user to edit in an alert dialog. The user can choose to save the location name as given, or they can edit it to be whatever name they choose. This allows the user to



use tags personal to them ('Home' for example), this allows for memorable search terms for them to search if they wish.



Another way for a user to tag their location is to use the search bar, here they can search an address and using the Places SDK for android, autocomplete suggested locations will be available for the user to select from. This feature is implemented using a Autocomplete Support Fragment. When the user selects the search bar, the fragment is displayed showing a list of all autocomplete suggestions. Selecting one of the suggestions will return a Place. Using this Place's LatLng coordinates and name, the required location fields are set ready for the location intent and the marker is placed onto the map. The location name is again verified by the user to give this flexibility of location names stored in the database

```
String apiKey = getResources().getString(R.string.google_maps_api_key);
Places.initialise(getApplicationContext(), apiKey);
autocompleteFragment = (AutocompleteSupportFragment) getSupportFragmentManager()
        .findFragmentById(R.id.autocomplete_fragment);
autocompleteFragment.setPlaceFields(Arrays.asList(
        Place.Field.LAT_LNG, Place.Field.NAME, Place.Field.ADDRESS));
autocompleteFragment.setOnPlaceSelectedListener(new PlaceSelectionListener() {
        @Override
        public void onPlaceSelected(Place place) {
        LatLng latLng = place.getLatLng();
        mLatitude = latLng.latitude;
        mLongitude = latLng.longitude;
        mLongitude = latLng.longitude;
        mLongitude = nlarg.getLatLng();
```

The final way to tag a location is to long click select any location on the map using the Google Map *setOnMapLongClickListener()* method. As before the coordinates are found using the Geocoder methods and the location address is verified and changed by the user as required.



The *moveCamera* method is used frequently whenever a new location is wanting to be pinned on the map. This method will remove any already placed pins, as only one location per entry should be made.

A new pin is added to the specified

location, the name specified by the user is set as the title. The default zoom value when moving the camera is set to 15*f*, to keep the consistency of the map visually.



When a location has been chosen, the user will select the "Set Location" button an Intent is made to pass back the chosen locations coordinate and location name to the diary entry. As this activity was called using a *startActivityForResult()* method, intent is returned by setting the result and finishing the activity

To remove the location tag the user can select the "Remove Location" button, after confirming the action in the displayed alert dialog, the location currently displayed will be removed and the user will be returned to the diary entry where now the location field will be blank.

5.2.3.4. Photo

The implementation of allowing photographs to be added to entries has been partially successful. Currently all methods to input a photograph into an individual entry are working well. In order to first access the devices camera or camera roll, requests are made for the user to grant permission for the application to access its required target.



Once permission has been granted, the user will be prompted via a dialog box to add a photo via selecting from the device's camera roll or taking a new photo from the device's camera. Both options have been successfully implemented using intents to the requested activities within the device. On result of the activity an image URI is found and set within the entry, a small icon of the photograph is displayed in the entry. When reselecting the photo field, a larger image is shown within the dialog box for the user to view easily.

Currently the photo cannot be stored within the database. This would have been implemented using a GridFS object available to work with MongoDB, but due to lack of knowledge and time, this was not successfully implemented. Further explanation to this issue is given within the Evaluation section.



5.2.3.5. Saving an Entry

Each entry is saved individually, this decision was made to reduce the potential errors a user could make by overwriting data changed in different entries when they only meant to save one. To complement this design, it was initially planned that the system will check for any changes made within any and all of the entries when the user navigates away from the Diary Entry screen. However, due to time constraints this was not implemented fully and therefore there are no checks made to whether data that has been changed has been saved.

The data is saved by using the *updateOne* method for MongoDB. As each entry is stored within an array of Documents, a new Document is created using all the data entered for this entry row. If any data fields have not been inputted, this data field is given an empty string value as a placeholder for when this entry is retrieved or updated, meaning no extra work is needed to input these fields into the document later. The

update operator is "\$set". \$set also allows updating of fields that have already been inputted, it will replace the current data with the new data. If the entry being saved is new and therefore does not have a place in the array, the update option of *upsert(true)* will update any existing entries but will also create a new document in the array otherwise.

A toast message is displayed to the user on success of the save, this is to inform the user that the entry has been saved. This ensures the user is aware of the update success and will remove any doubts or confusion.



5.2.4. To Do Lists



5.2.4.1. Add one off task

Within the To-Do task entry screen (as shown), the user can create a one-off task for the date they specify or create multiple repeated tasks with the frequency they specify, as explained below.

When the user selects to add a new task they are taken to the To-Do Task Creator screen, initally the only fields visible are the Task Name and Task Date, the other two secotions (reminder time and repat task) are accessible when the user checks the corresponding check box. This will reduce the potential errors made by users, by limiting the fields displayed to a user until they choose to access them.

The Task Date and Final Date use DatePickerDialogs and the Reminder Time uses a TimePickerDialog. Both are Android methods to ensure the correct format of time or date is inputted to remove errors caused by invalid data stored in the database.

A one-off task requires a Task Name and a Task Date, optionally the user can input a Reminder Time if required. When the user selects "Save" the system will run some verification checks, if the user has

not entered a Name or Date they will be prompted to do so until the Save is accepted. Only if the "Add reminder for task" checkbox has been checked does the system verify that a Reminder Time has been entered and will prompt the user if not.

5.2.4.2. Add repeatable task

To add a repeatable to-do task the user will specify the frequency of the task, by the days applicable and the frequency of weeks specified. The Final Date for task will be the last date this task will be added on (if task falls on this date).

The method to calculate which dates this task should be saved is as follows. The system starts with the current date saving this to a temporary variable. While the current date does not equal the final date, the system will look at the day of the week of this current date and compare it to the selected days that the user would like to repeat the task on. If there is a match, the system will also check that the number of weeks passed since the last time this day of the week was added is equal to the frequency of the number of weeks the user would like the task to be repeated in (for example every 2 weeks, so the number of weeks would be 2). If this is also equal this task can successfully be saved for this date. To do this the system will try and retrieve the entry for this date, if an entry is successfully retrieved, the task is added to the list of to-do tasks for this date (or a new list is created if no to-do tasks exists for this entry). Otherwise if no entry for the given date is found an entry needs to be created first in order for the to-do list to be stored correctly.

When the date has been checked and if necessary saved, the current date is increased by one. This new date is checked as above and so on until the current date equals the final date.

if(mNumberWeeks==1){

When checking the day of the week to Sunday, the number of weeks will be increased (if the user has set the frequency to every week, this will not be increased) if the number of weeks just counted (eg weekcounter-1) is equal to the number of weeks specified, then the count is restarted at 1, so the next loop will save the entries as the number of weeks that have past is equal to the number of weeks the user has specified to pass.

Shown here is the while loop used to only look at dates before the final date. Within here, the checks made to see if the current day of the week matches to a day of the week checked by the user. And checks for if the number of weeks passed is equal to the number of weeks

specified.

5.2.4.3. Checking off tasks

The to-do list found in both the main to-do list screen and the Home Page use the same adapter to simply display the name of the task, the reminder time set (if specified) and the checkbox that relates to if the user has checked this task off. When a checkbox is modified (checked or unchecked) the database is updated without any further user interaction needed, by using the *onCheckedChangeListener*.

5.2.4.4. Reminders

Unfortunately, the system does not currently remind the user of any of the tasks they have selected to have a reminder. This was due to lack knowledge of how to implement this feature, and eventually the time constraints of the project meant this could not be implemented successfully. Android has the method of AlarmManager that would be able to create alarms specified by the system and remind the user of their tasks. This would be able to run in the devices background and 'wake up' the device when specified. This method would therefore be ideal to use for the reminder features, and for future improvements of the project it would be hoped to be implemented to do so.

Event Creator	
Event Name	
Event Date	10/06/19
Event Time	
Event Location	
	ountdown for Event?
CANCEL	SAVE

5.2.5. Event lists for date

5.2.5.1. Add Event

The screen to add a new event implements simple methods to save the event and display them to a list using an adapter. The user will need to enter the Event Name and Event Date, optionally they will enter the Event Time and Event Location. If the user selects to add a countdown for an event and the user has not set the Event Time, the time will be set as 00:00.

5.2.5.2. Edit Event

The option to edit an event is available when a user long click selects them. This will return the user to the Event creator screen, but the related date fields are filled in with the corresponding data. The user can modify the data and save the event again which will then update in the database and thus all corresponding lists

Events for: Tue, 31	I Dec, 19
New Years Eve	Hilton Hotel 00:00

5.2.6. Home Page



5.2.6.1. Event countdowns

If a user has chosen to have a countdown for an event, they will be displayed in the countdown list on the Home Page. To do this, the system goes through all events in the database to find each entry that has been set to have a countdown. The system compares the current date to the date of the event, using JodaTime, the time difference between the two dates is calculated. If the time difference is positive the event is stored in a temporary array, if the difference is negative, the event has passed and therefore will not be stored in the array. This temporary array is then sorted into descending order of time difference, so the event that will be happening soonest is first in the array. This sorted array is then displayed in a list using an adapter, showing the event name, and the countdown until the date in months, weeks, days and hours.

Long click selecting an event gives the user the option to go to the corresponding diary entry for the date of the event

5.2.6.2. Date entry calendar

Selecting a date from the calendar found in the Home Page will

navigate the user to the chosen date's Entry screen. This is simply done using Intents and passing the selected date through and retrieving the entries for this selected date.

5.2.7. Analysis

5.2.7.1. Search terms

The user can search any test string, and the system will return any entry that contains this string in the text, emotion tag or location tag. This is implemented by retrieving all entries that match the owner_id of the current device, and for each entry that is returned, the text, emotion and location is checked if they match the given search term. If they do, the entry date, emotion tag and location tag are entered into an array which is then displayed to the user, to show all matching entries.

Each entry can be selected, and the user can navigate to the corresponding diary entry screen.

If required, the user can search without a search term and all entries created by the user will be displayed

Currently this way of searching is efficient and fast enough for there to be no noticeable delay to the user. In the future as the number of entries inputted by the user, the overhead will increase, and the user may start to see a decrease in performance when searching. It would

therefore be ideal to improve this method of searching by not having to retrieve every entry made by the user each time a search is made.

happy		Q ₩ ▼
Mon, 08 Apr, 19	Нарру	London, UK
Mon, 08 Apr, 19	Нарру	
Mon, 08 Apr, 19	Нарру	Borkou, Chad
Tue, 09 Apr, 19	Нарру	
Wed, 24 Apr, 19	Нарру	
Wed, 01 May, 19	Нарру	West Bridgford Nottingham,
Thu 02 May 10	Hannv	South Bank
MAP S	EARCH RESULTS	
Average Emotion(s):	Нарру	
	MAP ALL EN	VTRIES
	<u>64</u>	Ð

5.2.7.2. Advanced search

The user is able to make an advance search by selecting the small dropdown button within the search bar, this will give them the option to only search within the text, emotion tag and/or location tag. This gives the user more flexibility with their searches.

5.2.7.3. Average Emotion

The average emotion(s) of all entries returned by the search is displayed.

The average is found by the most common emotion tagged in all the returned entries. If there is more than one emotion that occurs the most often, they are all displayed.

The average is calculated by creating a temporary map, the keys of which are the 8 emotions available, the values will keep a count of the number of times the emotion tag is found. When the system is searching through each entry, if an entry matches the search string, the emotion tagged (if any) will increase the value of the corresponding position in the map. Once all entries have been searched, the values are put into an array and sorted, the largest value is found. The keys that map to the same largest value are retrieved and these are therefore set as the average emotion.



5.2.7.4. Map results

For another way to view the search results the user can choose to map the results. For this the system takes the coordinates of the resulting search entries and maps them. The pins mapped represent at least one entry has been tagged with this location. Each pinpoint on the map combines the entry with the same location, the number of entries in this group is shown when selecting the pin. The display of entries is shown when selected, and the average emotion for this group of entries is also displayed.

To implement this, when the entries are searched, the coordinates (longitude and latitude) for the entries are stored in a HashMap as a LatLng key. Similar to the search emotion Map variable, this counts the number of times this location is tagged, this will be used as the count above the pin when selected. The entry date and emotion tag are also added to an array, which will be passed to the map intent.

The Map All Entries button will map *all* locations tagged in any entry inputted using the same method as described above.

When selecting a pin, the relating entries to this location are

displayed. Long click selecting one of the displayed entries allows the user to navigate to the full entry for the given date.



6. Evaluation

6.1. Test Cases

Test cases were created to test functionality and accuracy of the system. These test cases were carried out as true to the case as possible to ensure no bias or work arounds were made. The tests were written to ensure all requirements have been met and the system is functional for different types of users, each with different requirements of the system using different types of data. Each test is given a Pass or Fail status, depending on if the tests successfully completed the tasks with the provided test data. The test data was written to test for:

- Valid data, this data should always be accepted by the system. The data is based on potential common user tasks.
- Erroneous data, this data should never be accepted by the system. The data was also based on potential user tasks, but tests for common errors that may occur with the user. This data will test how the system handles any erroneous data, testing for error checking and subsequently error messages if error not handled by system effectively
- Extreme data, where applicable, data that is at the highest or lowest possible values to ensure the system can handle data at the extremes, valid extreme data should always be accepted by the system

See Appendix 2: Test Cases

The results of the test cases show 82% of tested functionality were successful. This high rate shows that the majority of functionality requirements initially outlined for the project were successfully implemented and function appropriately with both valid and invalid data.

There are however some failed test cases, these relate to the functional requirements FR2, FR6 and FR8.

FR2 was not successfully implemented due to the fact that the images selected within the application cannot be stored within the database. As explained previously with the photo data implementation, the photos can be inputted into the entry, but the methods required to input these into the database were unsuccessful. It was planned to input these photos into the database by converting them to GridFS objects. GridFS objects are datatypes that are acceptable within a MongoDB database, allowing large sized objects that may exceed the document size limit. (MongoDB 2019b) However due to the limited time constraint, no method was found to convert the image to the GridFS object to be stored in the database, or to be able to retrieve the GridFS object and display it appropriately in the entry.

Therefore, although aspects including retrieving a photograph from the devices camera roll and allowing the user to take new photographs from the device's camera. This test case ultimately failed as this data could not be saved or retrieved.

FR6 and FR8 are related to getting reminders from the system about events and to-do tasks inputted by the user. These test cases failed as there is currently no method for having reminders or alerts implemented into the system. This feature was not implemented in time, although the relating and required data fields to implement this feature have been added. Therefore, with further work, this feature can be added relatively simply with little modification to existing methods and activities.

Testing for FR12 and FR13 has not been carried out as these features were not implemented in time. However, due to these requirements being labelled as "Could Have" requirements and disclaimers made previously, these features were only expected if there was time spare at the end of implementing all other features successfully. Therefore, the lack of these features has not affected the main functionality of the project, only additional features that were not main requirements.

6.2. Usability Testing

To test the system with a set of potential real users, a moderated user testing approach is used. Testing with real users that will be similar to the personas outlined in the background research will help evaluate how the system will work with users and if it meets expectations.

The tests will begin by showing the user how the system works, giving brief tour of functionality available. This will make the tests more effective to execute as the user will have an idea of how the tasks should be carried out.

The users will then be given a user a number of tasks that will simulate common tasks a user will want to achieve while using this system. Users will be encouraged to think out loud while carrying out the tasks and no prompts will be given unless they become lost or significantly go over the allocated time. This will show user expectations in how they natural navigate system and if there are common user errors.

Following the tasks, the user will be asked if they have any general comments or opinions on how the app functions and what improvements they may suggest making navigation through the system more efficient or self-explanatory. This may also include if they had different expectations in how they thought the app would unction or where they thought a task should be accessible from.

Task Number	UT01
Task Goal	Create New User Login
Task Explanation	This will be the first step for a user, whether they can complete this task will determine if they will be able to access the application.
	For efficiency the user may use the following test data: Username: userOne Passcode: 1234
Time Allocated	40 seconds

Task Number	UT02
Task Goal	Add and save entry for today
Task Explanation	This is the main feature of this application, completing this task is paramount in the reason this app exists
	For efficiency the user may use the following test data, this will also ensure all features are tested by the user: Text: Today I went to the London Eye Emotion: Happy Photo: Select photo of London from camera roll Location: London Eye
Time Allocated	2 minutes

Task Number	UT03
Task Goal	Add to do task for today
Task Explanation	User will be asked to add a one-off task to add to their to do list
	For efficiency the user may use the following test data:
	Task Name: Phone Dad
	Reminder Time: 17:30
Time Allocated	40 seconds

Task Number	UT04
Task Goal	Add Repeatable To-Do task
Task Explanation	User will be asked to add a task that will be repeated multiple times, this will test if the user is aware of the "Repeatable task" feature, and that they do not think they will need to enter each task individually and manually
	For efficiency the user may use the following test data:
	Task Name: Take out Bins
	Repeat every other week (every 2 weeks) on Wednesday's and Saturday's
Time Allocated	1 minute

Task Number	UT05
Task Goal	Add Event with countdown
Task Explanation	User will be asked to add an event with a countdown. The event will be for a future
	date, this will test how the user navigates the system to complete this task
	For efficiency the user may use the following test data:
	Event name: Jane's Party
	Event date: 23/07/2019
	Time: 09:45
	Location: Village Hall
	Add countdown
Time Allocated	1 minute 30 seconds

Task Number	UT06
Task Goal	Check off task from to do list
Task Explanation	User will be asked to check off one of the tasks for todays to do list, this will used to shown if users use the shortcut to access the to do list in the home page or if they go to the to do list page, via the diary entry
Time Allocated	30 seconds

Task Number	UT07
Task Goal	Review Diary entry for given date
Task Explanation	User will be asked to review the entry previously entered for the 28 th April 2019. This will see if users are able to navigate the calendar on the Home page to review previous entries
Time Allocated	30 seconds

Task Number	UT08
Task Goal	Search for all entries containing a given search term
Task Explanation	The user will be asked to search all entries for the term "Happy", they will be asked to search in the text only. This will see if user can navigate to the search and if they use the advanced search appropriately The user will be asked if they can identify the average emotion of these search results
Time Allocated	1 minute

Task Number	UT09
Task Goal	Map all location tags entered into journal
Task Explanation	The user will be asked to map all locations, this will ensure they know the difference
	between mapping the searched entries and mapping all entries
Time Allocated	45 seconds

Task Number	UT10
Task Goal	Reset passcode for account
Task Explanation	The user will be asked to reset the passcode they previously entered when creating
	their login. This will ensure the user is able to navigate to the passcode reset screen
Time Allocated	45 seconds

6.2.1. Usability Test Results

To complete this usability testing, three users were asked to carry out the tasks outlined above individually. As little guidance as possible was given, only when the user asked for guidance when they could not see how to complete the task.

The users selected were selected to try and give three unique opinions as each user has different backgrounds, ages and experience with diary usage. Ideally these tasks would be tested by as many users as possible to get as much feedback using the system as possible as users all have different expectations and will try to complete tasks in their own way, however due to time constraints only three users were able to carry out the tasks.

User 1: A current diary user. He currently writes in a diary book daily, often short passages of what has happened that day. Has never tried a mobile diary app but has very good technical knowledge and is moderately proficient using mobile apps

User 2: Does not currently keep a diary. Did attempt to a few years ago but often forgot to create an entry and soon lost interest. Would be interested in restarting a diary given a more interesting way to keep one. Has high proficiency in using mobile apps

User 3: Does not currently keep a diary and has previously had no desire to start one. This user has very high proficiency using mobile apps.

UC01	
User 1	Pass
User had no problem creating new login	
37 seconds	
User 2	Pass
User had no problem creating new login	
24 seconds	
User 3	Pass
User had no problem creating new login	
27 seconds	

UC02		
User 1	Pass	
The user selected the "Add new Entry" button even though there was an empty entry available to them,		
thus creating an empty row in the list.		
User felt the location set field may not be large enough to select accurately but navigated well otherwise.		
User was hesitant whether how to save the location tag, eventually correctly selected 'Set Location'		
button		
2 minutes 38 seconds		
User 2	Pass	
User navigated through all data input fields well, only	unsure of how to save entry, did not realise that the	
save button relating to the row only saves that entry, eventually saved after trialling save buttons		
2 minutes 4 seconds		
User 3	Pass	
User found entry simple, saved entry successfully		
1 minute 42 seconds		

UC03		
User 1	Pass	
Initially forgot where to find To-Do tasks, found eventually through diary entry screen. Otherwise navigated well, no issues with entry		
57 seconds	Γ	
User 2	Pass	
Accessed through main diary entry screen To-Do List. Navigated and succeeded very easily		
29 seconds		
User 3	Pass	
Accessed via Home Page to-do list, entered successfu 30 seconds	lly	

UC04	
User 1	Pass
Navigated well. Initially did not enter 'Final Date' as r	equired, but app prompted user and was then
successful	
1 minute 6 seconds	
User 2	Pass
Navigated well, again did not initially enter 'Final Dat	e' but when prompted by app, successfully added
tasks	
1 minute 8 seconds	

User 3	Pass	
Navigated very well, entered all required fields without prompts		
58 seconds		

UC05		
User 1	Pass	
Navigated to the date first via the Home Page calendar, then to the events list from the Diary Entry screen		
displayed. Successfully created		
1 minute 32 seconds		
User 2	Pass	
Navigated to from today's Diary Entry screen, then cl	nanged date from within events entry screen. Event	
successfully created		
42 seconds		
User 3	Pass	
Navigate to the date first via Home Page calendar, th	en to events list. Successfully created	
59 seconds		

UC06		
User 1	Pass	
Found via Diary Entry screen To-Do list. Wondered if needed to save but after not finding one accepted saved automatically		
37 seconds		
User 2	Pass	
Found via Diary Entry screen To-Do list. Successfully checked		
11 seconds		
User 3	Pass	
Found via Home Page To-Do List, successfully checked		
16 seconds		

UC07	
User 1	Pass
Successfully navigated and reviewed	
16 seconds	
User 2	Pass
Successfully navigated and reviewed	
28 seconds	
User 3	Pass
Successfully navigated and reviewed	
18 seconds	

UC08		
User 1	Fail	
User could not find the advance search button and needed to be prompted. After prompting search		
successfully made		
1 minute 31 seconds		
User 2	Pass	
Successfully found search area, thought advanced search was hard to see, but found eventually and successfully searched		

32 seconds		
User 3	Pass	
Initially could not find advanced search, found eventually, successfully searched		
43 seconds		

UC09	
User 1	Pass
Found easily and successfully mapped all results	
12 seconds	
User 2	Pass
Initially did not know what this task was asking them to do, after searching around eventually successfully	
mapped	
27 seconds	
User 3	Pass
Navigated well and successfully mapped	
16 seconds	

UC10		
User 1	Pass	
Found easily via Home Page due to seeing settings menu when carrying out previous task, successfully		
changed login		
40 seconds		
User 2	Pass	
Found easily via Search Plus page, successfully changed login		
27 seconds		
User 3	Pass	
Found easily via Home Page, successfully changed log	gin	
38 seconds		

General comments:

All users found most of the screens and features easy to navigate. One comment was the 'Diary Entry' screen icon in the navigation bar was not clear what it was for, they could not give an example of what they were expecting but did not think the icon chosen was ideal.

Other comments mentioned that some buttons, including the advanced search drop down button and the area for location tag selection, was too small. For some screen sizes, especially mobile phones, these buttons may not be able to be selected accurately.

One user did not feel they would use the map locations or search results section of the application, so they were unsure of the use of this feature.

Some features require to be double clicked, this includes the calendar picker for date selectors in the event entry and to-do tasks entry, were not obvious to users. They felt the calendar picker should appear on first click of these fields.

For some of the alert dialogs, for example the emotion select dialog, one user was unaware that selecting anywhere outside of the screen will close the dialog. They suggested that some of these dialogs might benefit a 'Cancel' or similar button to let users close the dialog if they do not want to select an item within the dialog are unsure how to close it otherwise.

6.3. Questionnaire

As there were constraints to getting enough users for testing a questionnaire was developed to gather user opinion on the app. It would have been preferable to have been completed the questionnaire survey before implementation, but the time taken to gain ethics approval was longer than expected and the tight deadlines for completion of this project prevented this. Instead the results found will be used to contrast and compare the ideas and opinions of real users and what has actually been achieved in this project. This will also help to identify improvements and potential development of this application in the future.

6.3.1. Methodology

The questionnaire was shared via social media asking potential participants to complete a voluntary and anonymous online questionnaire. In total 34 responses were received, with participants ranging in age from 18-25 to over 65. Participants were asked whether they were regular diary users to see identify whether there were differences of opinion on what they considered should be in a diary app between those who write journals every day and people who have never kept a journal before. Opinions on what functions users would find useful in a mobile diary app were collected. This data will be used to analyse whether the current features of the created application match the opinions of real users

Full questionnaire shown in Appendix 3: Questionnaire

6.3.2. Key Results

Full Results shown in Appendix 3.1: Questionnaire - Raw Results

The following is the breakdown of the 34 participants

- Age Groups
 - o **38% 18-29**
 - o **21% 30-45**
 - o **35% 46-65**
 - o 6% Over 65
- Gender
 - o 82% Female
 - o 18% Male

Out of the 34 participants, 53% reported that they consider themselves diary users. These current users ranged from 12% reporting relatively short time use of up to 1 year of use. 44% of users said they have been keeping a form of diary between 2 to 5 years. While the remaining 44% have said they kept their diaries for over 10 years. These results show that keeping a diary is often based on consistency and may be due to a habit built over many years. 61% of current diary users reported that they write daily in some capacity, which verifies diaries to be a daily activity that becomes part of a person's routine.

83% of current diary users have said they only use book or paper forms of a diaries, and the remaining 17% use a combination of book and digital forms of diaries. This shows there is current lack of use of mobile diary apps, this may be to do with, as mentioned before, writing diaries is traditionally written in physical copies and as majority of users have been keeping journals for many years, most before mobile diary apps were available or good enough for users to make the switch from physical to digital.

Out of the 34 participants the remaining 47% reported to not consider themselves diary users. However, 94% reported to have previously tried to keep a diary in some capacity, but due to numerous reasons have not in recent years. These reasons include; forgetting to write (reported by 25% of non-user), losing interest

(25% of non-users), feeling a diary did not benefit their lives (19% of non-users) and 25% felt they did not have the time to complete their entries.

Many of these reasons could be answered using a mobile app, the use of reminders to create an entry each day or at the users chosen frequency could help user to remember to keep up to date with their journals. Whether a mobile app would be beneficial or would be able to keep a user's interest would depend on the user's individual needs but using a mobile app could offer the user more features to bring diary keeping up to date. In the most recent reports at least 94% of adults in the UK own a mobile phone (it is unreported how many of these are smartphones, but it is assumed that majority of adults will own a smartphone or a device that has capabilities of working with applications) (Henshaw, S 2019). This could attract both current diary users and non-users to create a diary on their devices that will be more flexible and offer more features. The participants were asked to rank their proficiency in using mobile application, 97% of participants said they were at least moderately proficient (3 out of 5), this shows that majority of users would be happy to use mobile apps means majority of users would be happy to use the user interface designs and features that have been implemented into this application.

Within the questionnaire, the participants were asked their opinions of what sort of features they felt would be beneficial to them and what features they would use if they were to use a mobile diary app. The following shows the percentage of users that agreed the feature would be beneficial:

- Photographs 88%
- Event Reminders 82%
- Location Tagging 79%
- Daily To-do lists 65%
- Emotion Tagging 47%
- Repeat Reminders 44%
- Event Countdowns 41%
- Goal Tracking 29%

The available extra data inputs are interesting to users to keep entries modern and give users flexibility in how they keep their thoughts and memories. Although photographs can be stored on smartphones already, storing and organising these photographs by date and enhancing text diary entries with photographs give the diary app an advantage over physical diary books.

The responses show that as well as diary entries, users would like the app to have extended features like event reminders and daily to-do lists. The is encouraging for the project created as both features have been implemented successfully to create events and to-do tasks. Users would like the remind feature involved with these, so further work would be recommended to implement the reminder functionality to fulfil the user expectations.

Currently this application does not offer the use of photographs within the entries, because of the issue of photographs not being stored in the database. This situation would not be ideal for real users as this was the highest rated feature users would most appreciate. The feature of goal tracking was not implemented in time, but from analysing these results not many users would see the benefit of this within a diary app. Otherwise all other high rating features have been successfully implemented so the application has met many of the expectations and wishes real users would like to see in a personal diary application.

Users were asked for any other features that were not listed that they would like to see in a mobile diary app. Some of the comments and features mentioned were features including 'on this day' reminders where users can see what was inputted on the current day years ago, other input types including drawing and

weather tags and the ability to output their entries in a format of their choice. Users commented on wanting to be able to synchronise their diary between devices, which could include different platforms (between Windows, Android and iOS for example). The final comment made was to allow flexible designs for user to be able to make the application their own, with the use of colour coding between different types of entries (work, home etc.). This would allow users to make their diaries unique and most efficient for them, as with physical diaries users often make them their own, either by the design of the book, or the layout of their entries, so it will be important to continue this in mobile apps to not take away from the benefits of physical books.

On top of this users were asked if they would be interested in analysis of their entry, 59% of users agreed that they would like this analysis in some sense. This is inconclusive as there may be many ways an entry could be analysed and depends on what sort of entries the user likes to make. If a user does not make use of one type of input (emotion tags for example) this would narrow the amount of analysis that could be made on the entry. The use of 'On this day' feature seems popular as it is often used in other social media (Facebook memories for example (Facebook 2018) and diary applications (as found in the DayOne app). This could be a simple way of analysing a journal that users would enjoy and would be interested in.
7. Future Work

The current application is functional and implements the majority of the features initially planned. There are a few features that were not completely executed to the standard that would have been expected, these would be the first areas that would be worked on further, given more time. These include allowing photographs to be stored within entries, this was a highly expected feature as shown from previous application research and the responses from the user questionnaire. The issues faced were to do with getting the images from the device to be stored as GridFS objects within the database, retrieving them successfully and displaying them in the entries when required. The structure for implementing this feature is set up, and with more time and knowledge this should be a relatively simple task to achieve.

Another planned feature that was not fully executed is the ability for the application to alert the user of the reminders they may have set for tasks and events, and a reminder to write their entry every day. This, again, seems like a relatively simple task to achieve, and given more time to research into methods of how to implement this function, could be added to the system with little change of code or functionality.

The current method used to connect the application to the database is via MongoDB Atlas. This method requires an internet connection, which although it can be assumed that internet access (whether through Wi-Fi or data roaming) is available in the majority of areas around the UK. (Office for National Statistics 2018) It would be ideal to allow users to access their diary entries offline, giving the user more flexibility and having the diary available at all times. This may include changing the database access or method of retrieval which would cause a significant change to the system and would therefore only be recommended given enough research and reason to do so.

Another feature identified is the improvement of the security of the login system. Improving this system will involve the use of separate collections within the MongoDB database for each user, accessed only through the correct login details. This will also allow for synchronisation between devices, using the login system, the user can login from multiple devices and still see all the same data. The security behind the login feature will need to be improved to keep all data secure and private only to the user, this could be done via encryption of passwords and/or entries.

Activities that would add to the functionality of the app include the 'on this day' reminders and allowing more flexibility within the app. This could include allowing the user to customise the design to suit their aesthetic and 'hide' features they may not require. Making the application personalised to the user will improve efficiency, this includes remembering user entered locations, for example having a 'Home' or 'Work' location tag that can be entered by the user and will then be recognised in future tagged entries.

One of the requirements that was set out at the beginning of the project was the plan to implement goal tracking into the application. Although from the evaluation and questionnaire analysis it seems this feature may not be as beneficial to users as other features mentioned above. It would therefore be recommended that if this feature was implemented that it would be flexible and not intrusive to the rest of the system if the user does not want to regularly use the feature. Keeping these additional activities separate from the main operations would be suggested, to not take away from the main aims of the project but still give users these additions to the application.

To improve the analysis of the diary entries, different methods of the analysis could be implemented. For example, showing how the user's average emotion has changed over a period of time for a location or search term. This could be displayed in many different visual ways with graphs or maps. The user could have the ability to change which average is used to find the average emotion, not just the mode but instead using an emotion 'scale' and using this to find an average position on the scale.

One major change that would need to be considered if this application was to be published would be reviewing how the application's features work on larger scales with multiple users. The current database would need to be upgraded from the current fixed size sandbox account to a larger more personalised tier. This would allow more data to be inputted and the sending and retrieving of data to be efficient and give a high performance while multiple users are using the application's database.

8. Conclusions

This project was created to offer users a simple and efficient way of keeping a mobile diary. There are many target users for this app, including users who would like to make their current physical diaries digital, offering them more ways to input their entries via text, photographs, location tagging and emotion tagging. Some users may be looking for a way to help observe and review their life's events with easy ways to search through and review the analysis of their entries. Other users may use the app to help organise and plan their everyday lives, using the features of the to-do lists and event countdowns. The flexibility within the app is important to allow users to use which ever features they feel benefit them without their unused features getting in the way.

The main aims of this project were to give users an opportunity to enhance their journaling experience, offering features that are not possible when using physical books. Although there are many current solutions to this, this app was created to combine the features loved by current users of digital diaries and improve the features that may not be available or well received. The main features that this project aimed to implement included:

- Diary entries that can be completed with text, photographs, location tags and/or emotion tags
- Allowing users to review and edit any entry previously entered easily and efficiently
- Allowing users to create events for specified dates and create countdowns for these events if required by the user
- Creating daily to-do lists that can be checked off by the user with reminders available to complete the tasks
- Adding additional efficiency when creating the to-do task, the user can create repeated tasks for all specified dates
- Searching and analysing methods for all entries. The results of which can be displayed to the user in different ways including simple results lists, analysis of emotion tags and mapping locations tagged
- Goal tracking for the user. Giving feedback, analysis and advice of how much progress the user has made towards their goal

Out of all the features initially planned to include as described above, the majority of the features were implemented successfully and achieved all aspects planned. There were however a few features that were not successfully implemented, whether due to running out of time, lack of knowledge, struggles with methods and execution or being overly ambitious on what would be able to be achieved with the scale of this project. The main aim of the project was to give users a flexible platform to carry out a variety of activities that can benefit their journaling experiences and help improve organisation of their lives; this has been achieved to a large extent.

Some of the issues faced were due to issues getting the system and the database to work efficiently together, this was primarily due to lack of experience in using both the Android methodologies and MongoDB Atlas. This included allowing photographs to be stored in the database, this was expected to be one of the simpler tasks to achieve but ended up taking a lot of time and work to try and get this feature to work correctly. This unplanned delay in implementation meant the time planned to implement additional features like the goal tracking was cut short and therefore these features were not implemented either.

There has been a close attention to detail throughout the project which has helped make the system more user friendly and has improved functionality of the simple requirements initially laid out for this project. These small additions give the user more flexibility using the app, with simple features such as being able to add multiple entries for a date, with these entries being displayed in a simple and aesthetic way. The user has the option to navigate through the system using different paths, for example being able to add a new to-

do task through both the Diary Entry task list, and the Home Page task list. This flexibility in navigating around the system gives the user freedom to access the features they require in a way they recognise best.

The analysis and searching of the system allow the user to review all their previously entered entries. These entries can be grouped by search terms entered by the user, and if the user wants a more visual representation, the mapping feature gives users this opportunity.

This system met most expectations. The main structure and methods of all features planned to be implemented (with the exception of goal tracking) have been successful. With more time and experience the issues within the application should be easily solved with only minor changes to code and functionality. The main recommended improvement for the system would be to improve functionality for use for multiple users, and to ensure features work on large scales with larger amounts of data.

9. Reflection on Learning

Throughout this project, I have faced many new challenges, primarily the new challenge of creating a mobile application. Having not undertaken this sort of project before I was prepared to work hard and knew the importance of planning my development of the project carefully. As I learnt, I found there were details within the development courses I was learning from that were not necessary for my use. It was important to me to read through each lesson, ensuring that the time taken learning this material will be beneficial to the project in the end. Not doing this from the beginning meant some of the time spent on research was not as beneficial as it could have been. I have learnt to be more selective with learning courses and found that it was better for me to learn the basics before starting implementation, and then, as and when I required, find more specialised and related lessons to aid my work.

Allowing myself to be ambitious with the project ideas from the start has kept me motivated, as always having another goal to work towards meant every detail was scrutinised to be the best I could make it. This improved my standard of work, bringing in little details that were not considered at the beginning, but have proved to be the highlights of the project. Proving to myself that these features and functionalities were possible given my skill set and the time constraints, pushed the motivation behind the project. With each new feature shown to be working a new motivation for a new task was found. This was helped by using a version of the Agile approach. Although a full agile approach was not used, using sprints for example, the idea that constantly reviewing the requirements of the project and what is wanting to be achieved helped to keep the project moving. Where as with other project, motivation and enthusiasm towards the project may be lost, improving the aims to best suit the projects direction kept the motivation high. This will be my new approach to individual projects in the future. I will incorporate this approach in any future team projects by encouraging self-reflection on projects, making sure that the aims are still manageable and achievable to keep the team motivated to finish at the highest standard we would be able to achieve.

While working with new programs and facing new challenges, I often found, when features were not working the way I had hoped, that I would become distracted and less focused on the task at hand. It was important to me to keep trying different approaches to issues found, and when a problem was proving difficult to find, to move to a different task so not to get unmotivated. This could have been a reason to why some of the main functionalities are not completely working (eg storing photographs within the database and retrieving them. And reminders for events and tasks etc.) but being able to admit that the current approach was not going to work within this project was important. This meant I was able to take a step back and try a different approach, and as mentioned there are a few approaches I have found that may be suitable to solve the missing functionalities. But in the end, there was not enough time to properly try these methods with the system, and so these would be the first additions I would make given more time.

I would say that I organised the development of this project well. Setting out what I planned to do each week with the work plan written in the Initial Report helped keep track of where I was within the project. In previous projects, I had not followed a timeline as closely, but I found with this project, following the timeline meant all planned development was organised well. Creating a more detailed work plan for all my future projects, will benefit me immensely. Keeping track of the project's development will help keep the project organised and developed efficiently. This project has taught me to not be afraid to take on new challenges, as the work produced in the time available is at a high standard, and a lot has been achieved.

10. References

Intelligent Change. (2019). *The Five Minute App*. Available at: https://www.intelligentchange.com/pages/five-minute-journal-app [Accessed: 6 February 2019]

Keep Productive. (2017). *The 5-Minute Journal App | My Full Review*. Available at: https://www.youtube.com/watch?v=NCHE1N-ZbkE [Accessed: 6 February 2019]

Overall Adventures. (2017). *Top 5 Journaling Apps with Francesco D'Alessio*. Available at: https://www.youtube.com/watch?v=RLGghuGmaTg [Accessed: 6 February 2019]

Udacity. (2019). *Developing Android Apps by Google*. Available at: https://eu.udacity.com/course/new-android-fundamentals--ud851 [Accessed: 30 January 2019]

Android Developers. (2019a). *Build anything on android*. Available at: https://developer.android.com/ [Accessed 10 February 2019]

Android Developers. (2019b). *Location and Maps*. Available at: https://developer.android.com/guide/topics/location/ [Accessed 10 February 2019]

Android Developers. (2019c). *Design for Android.* Available at: https://developer.android.com/design/index.html [Accessed 16 February 2019]

MongoDB. (2019a). *MongoDB Atlas*. Available at: https://www.mongodb.com/cloud/atlas [Accessed 11 February 2019]

Google Design. (2019). Design Available at: https://material.io/design/ [Accessed 16 February 2019]

Nielsen, J. (1994a). Enhancing the explanatory power of usability heuristics. *CHI '94 Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. pp152-158

Nielsen, J. (1994b). *Severity Ratings for Usability Problems* Available at: https://www.nngroup.com/articles/how-to-rate-the-severity-of-usability-problems/ [Accessed 20 February 2019]

Gautam, S. (2011). *Basic Emotions*. Available at: https://the-mouse-trap.com/2011/08/31/basic-emotions/ [Accessed 9 March 2019]

MongoDB. (2019b). *GridFS.* Available at: https://docs.mongodb.com/manual/core/gridfs/ [Accessed 22 April 2019]

Henshaw, S. (2019). *Mobile Phone Usage Statistics in the UK: How many smartphone users are there*. Available at: https://www.tigermobiles.com/blog/mobile-phone-usage-statistics/_[Accessed 29 April 2019]

Facebook. (2018). All of Your Facebook Memories Are Now in One Place. Available at: https://newsroom.fb.com/news/2018/06/all-of-your-facebook-memories-are-now-in-one-place/ [Accessed 29 April 2019]

Office for National Statistics. (2018). *Internet access – households and individuals, Great Britain: 2018*. Available at:

https://www.ons.gov.uk/peoplepopulationandcommunity/householdcharacteristics/homeinternetandsocial mediausage/bulletins/internetaccesshouseholdsandindividuals/2018 [Accessed 1 May 2019]