DESIGN RATIONALE OF MUCIS FESTIVAL APP

Introduction to Human Computer Interaction
COMP2004 Coursework

Liujia YU

20214735 | scyly1@nottingham.ac.uk

Table of Contents

Introduction

This report is the rationale of the android mobile phone app I designed for this festival. I used Axure as the main tool and photoshop as an auxiliary means. Axure made it possible for me to set links to other pages and URL links which made the prototype interactive and intriguing.

The prototype can be viewed here: https://2amrij.axshare.com. The yellow numbered squares are the annotations of corresponding elements in the interface.

In the rest of the report, I will illustrate why the interface is designed in the way it has represented from 4 perspectives: Cognitive Building blocks, Perceptual Model, Designing GUIs and Participatory Design.

Cognitive Building Blocks

The building Blocks are the most elemental and essential components of a design. They include line, shape, negative space, value, color and texture. [1]This design made combination of the them following the principles of design.

Gestalt principles

Apart from the elements themselves, the method of arranging them is of vital importance. While designing my app, I follow the 5 Gestalt principles and applied 4 of them: proximity, similarity, continuity, symmetry.

Proximity

Proximity is a way of using *negative space* to separate different content, or indicate that the two parts are relatively less related than those without negative spaces.

This principle is extensively applied in the system. For example, in the ***Event*** page, there are four main sections except for the slides of posters in the upper part: *playing today, Stages, Side Shows*, and *Other Facilities*. (Figure 1) *Playing today* and *Stages* are different content separated by negative space.

Similar application of proximity is also in ***Explore*** page and ***Me*** page.

[1] https://www.interaction-design.org/literature/article/the-building-blocks-of-visual-design

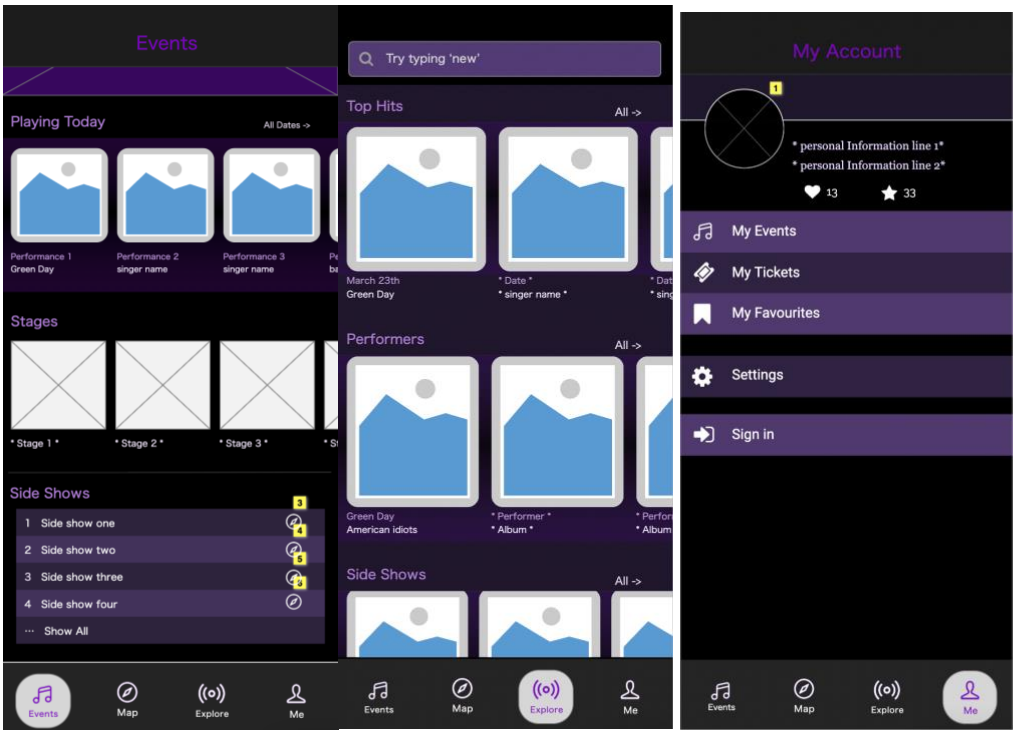


Figure 1 – Different Sections in Event Page, Explore page and Me page

Continuity

Continuity is another widely applied principle in my design. It is assumed in the law of continuity that our eyes will biologically follow the smoothest path when watching lines, despite how the lines were drawn. [2] This is achieved by using varying colour or type of the lines. The profile photo in Me page is an example to illustrate how continuous line can separate the background picture from the user's profile picture (Figure 2).

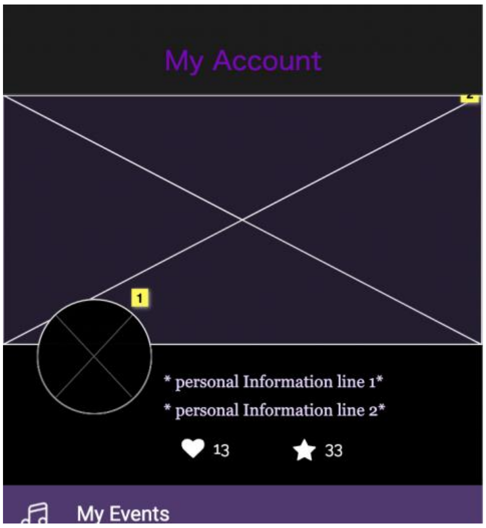

My Account
1
* personal Information line 1*
* personal Information line 2*
13
33
My Events

Figure 2 – the profile picture uses continuous line

This principle is also used in contents that are of the same hierarchy and classification. A continuous line is often broken to indicate that the content is different. Take Side Shows Part in the *Event* page as an instance, the "Show All" option is different from the quick link directly taking the user to

[2] https://www.toptal.com/designers/ui/gestalt-principles-of-design

corresponding side show pages. It is linked to *side shows* page where all side shows are listed. Therefore, the continuous line is broken as the pink line below shows.

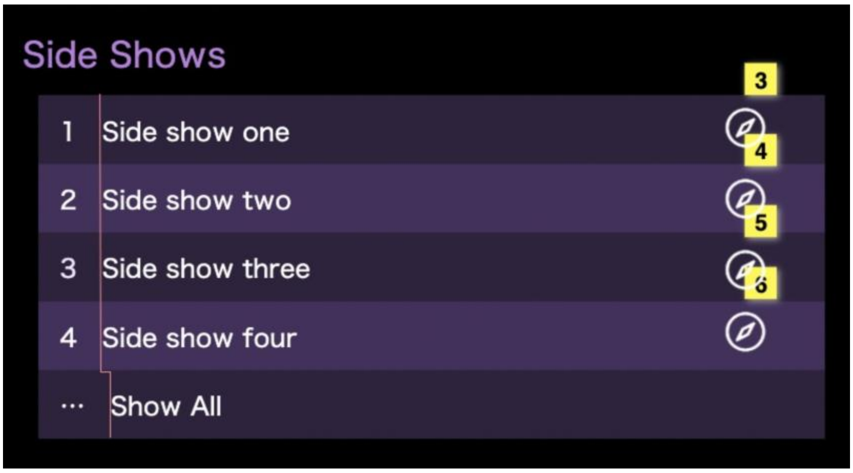


Figure 3 – A continuous line is often broken to indicate that the content is different

Similarity

Similarity is a method to group a content by using a visual pattern. The patterns can be color, shape, or size. Similarity can be used to tie together elements that might not be right next to each other in a design.

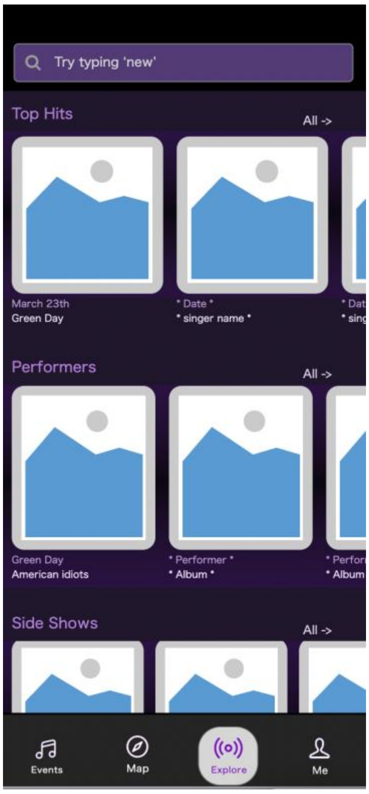


Figure 4 – Similarity groups elements by a visual pattern

In Figure 4, the ***Top Hits*** label is not right next to the other two labels ***Performers*** and ***Side Shows***, but they share a similar visual pattern: same font size and color. The user brain will process them as the same hierarchy classification title, which make this menu page more logical and increases Learnability.

Symmetry

Symmetry is an elegant way of showing the importance or dominance of an element. In the Performance detail page (Figure 5), the name of the performer(s) is aligned in the middle, while the date and place of the performance are symmetrically arranged in both side of the page. Together with the two symmetric ***Navigate*** and ***Add to Favourite*** button, the performer(s) name is emphasized, and this page is perceived as well-structured.

Dominance

Furthermore, dominance is a principle of arranging elements not to be neglected though not part of Gestalt principles. Dominance creates focus on a single element. Dominance can be established by using positioning, shape and colour, among many other factors.[3] The Performer label "Green Day" (Figure 5) is assigned with a larger text size and color contrast (white), making is the hero picture and naturally draws the most attention from users.

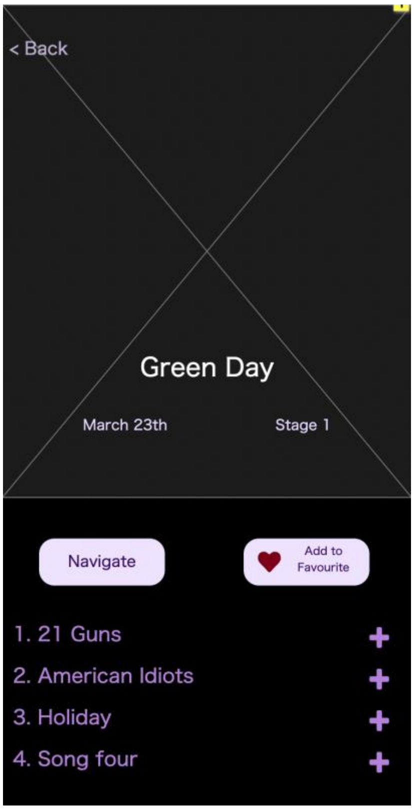


Figure 5 – The Performance Detail Page shows Symmetry

Memory

The way of how human memory works affect the design of interfaces to a large extend. This part of the report will analyse the design decision of this app in the aspect of memory.

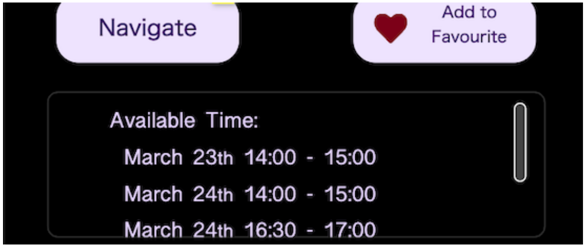
Human brain memory is divided into three hierarchies: Sensory memory, short-term memory, and long-term memory. [4]In the interface designing stage of development, designers should take in account of working memory (short-term memory) as it is closely related to attention.

[3] Dominance can be established by using positioning, shape and colour, among many other factors.
[4] https://uxstudioteam.com/ux-blog/user-attention/

Memory Chunks and Chunking

Presenting information in chunks makes reading or scanning easier for users and can improve their ability to comprehend and remember it. Practically, chunking is about creating meaningful, visually distinct content units that make sense in a larger overall context. [5]

This Idea explains why I made the available timetable in ***Side Show Detail*** page into chunks, in this case different lines. (Figure 6) This makes timetable more readable and memorisable.



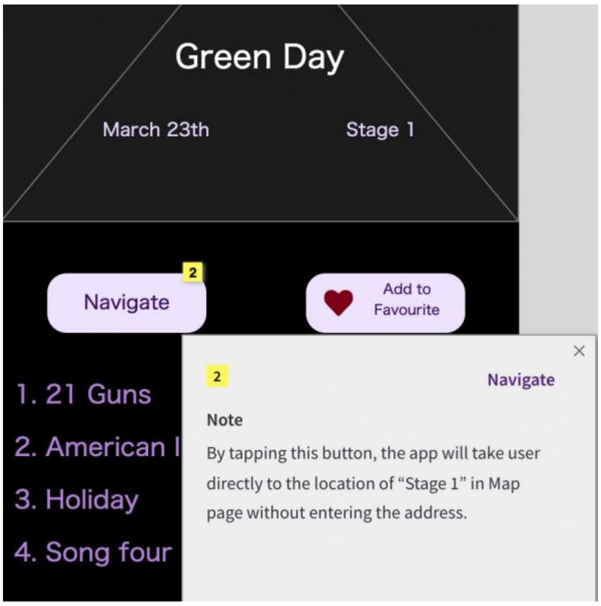
Available Time:
March 23th 14:00 - 15:00
March 24th 14:00 - 15:00
March 24th 16:30 - 17:00

Figure 6 – Information of available time are listed in chunks

Interruptions

Interruptions is inevitable in an app. Resuming a task after an interruption also depends on short-term memory. One of the feasible approaches to cope with interruptions is to provide users with memory aids.

For instance, in the ***Performance Detail*** Page, the button Navigate can redirect users to the ***map*** page and shows the location without remembering and entering any information (Figure 7). This functionality of memory aid is also implemented in all the pages that contains ***Navigate*** button, such as the ***Side Show Detail*** page and ***March 23rd performance List*** page. (Figure 8)



1. 21 Guns
2. American I
3. Holiday
4. Song four

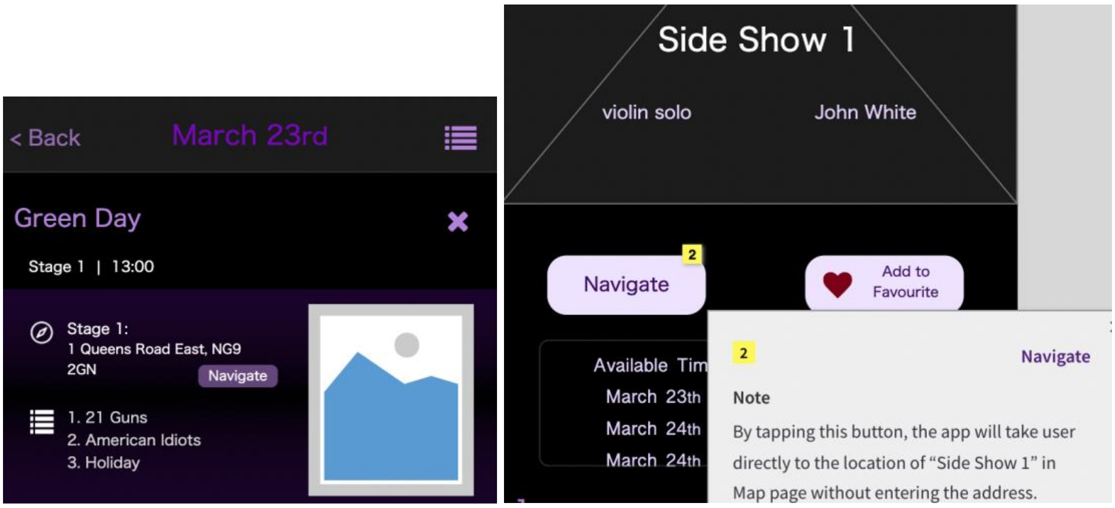
2 Navigate

Note

By tapping this button, the app will take user directly to the location of "Stage 1" in Map page without entering the address.

Figure 7 – Memory aid implemented in Button "Navigate"

[5] https://www.nngroup.com/articles/chunking/



Note

By tapping this button, the app will take user directly to the location of "Side Show 1" in Map page without entering the address.

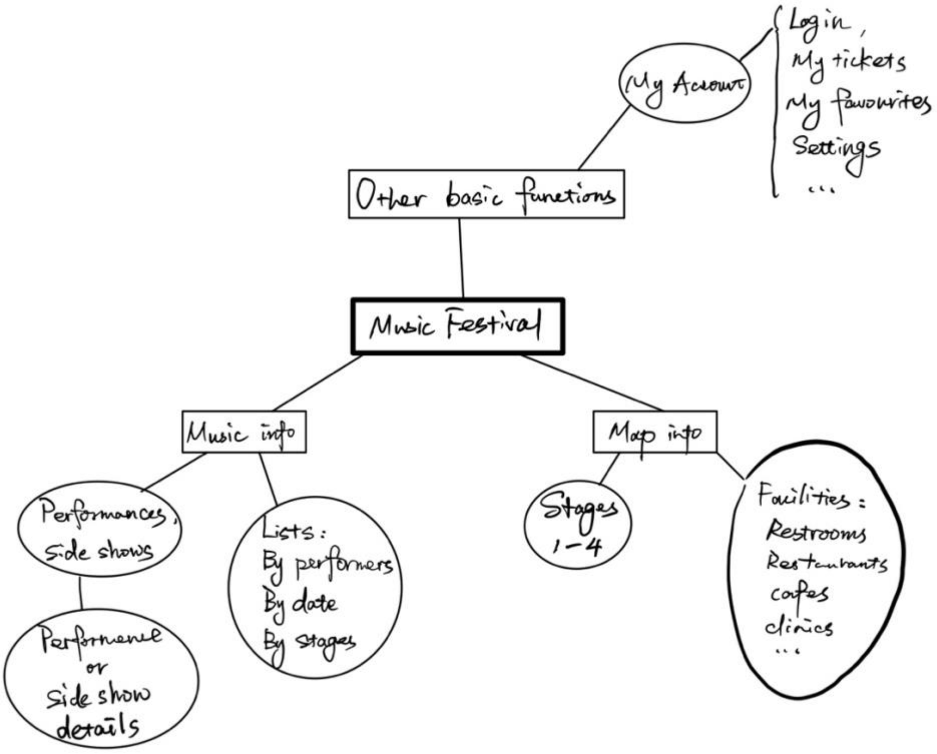
Figure 8 – Memory aid implemented in other pages

Conceptual Model

Constructing an appropriate conceptual model is essential before producing the design into actual prototypes. Inversely, a well-designed conceptual model can also make it easier for users to build an efficient mental model. As *Interaction Design* describes it, conceptual models are formed after synthesising the users' mental model of a system.

Therefore, collecting information of the user's mental models are vital. The main users of this Android app are between 18-35, which is a group of people that are attracted by novelty. I interviewed three targeted people for their thoughts. Most of them imagined this app as a combination of music app (e.g. Apple Music, Spotify) and Map app (e.g. Google Map).

After synthesising collected information, I produced the outline of my conceptual model:



Perceptual Model

Hierarchical task analysis (HTA)

Here is an example of HTA of finding the detail of a performance:

0. In order to view the detail of Performance 1 “Green Day”

1. Log in to the ***Event*** page

2. Find link to performance 1

   2.1. Find link in ***Playing Today*** section

   2.2. Find link in ***All Dates*** page

      2.2.1. Find link in ***March 23rd*** section.

   2.3. Find link in ***Stage 1*** page

      2.3.1. Find the label of “Green Day”

   2.4. Go to ***Explore*** Page

      2.4.1. Find link in ***top hits*** section

      2.4.2. Find link in ***Performer*** section

         2.4.2.1. Tap “All ->” button

         2.4.2.2. Find the label of “Green Day”

      2.4.3. Use the Search engine

         2.4.3.1. Enter search criteria

3. Directed to ***Performance 1 Detail*** page

Plans:

Plan 0: do 1-2-3

Plan 2: do 2.1 if performance 1 is not in playing today section,
do 2.2 or
do 2.3 or
do 2.4

plan 2.4: do 2.4.1 if performance 1 is not in top hits section,
do 2.4.2 or
do 2.4.3

Task Decomposition

Task decomposition can be performed by using HTA. The method of task decomposition can benefit analysing the details of tasks users carry out.

After decomposing the tasks, it is easier in the sense of feeding into design into solid interfaces. Subsequently, the sequences of screens and the list of menus can be determined. For example, the list of menus ***All Dates***, ***March 23rd***, ***Stage 1*** and ***Performer*** all make sense in this task analysis process.

GOMS

Same as HTA, GOMS is not applicable for the whole design interface functionalities. Thus, we choose finding the detail of a performance as an example to illustrate the usage of GOMS principle in my design.

There are 8 ways to visit a performance detail page as list above in the plans of HTA. Most of the methods (such as 1-2.2-2.2.1) are long and complex, but guarantee that the user can find the performance provided that this performance exists in the database.

As **Keystroke level model (KLM-GOMS)** suggests, the fewer the operators, the better predicted time. Therefore, if the task is done more frequent than others, it is reasonable to set up a quick access to the desired page. Assume the "Green Day" performance detail page is visited more than the other detailed pages, putting a link under the **Top Hits** section can reduce time. Similarly, the **Playing Today** section is set up to establish quick links to the performances playing on each day. These quick accesses effectively reduce access time to most frequently-visited pages.

Designing GUIs

This interface is designed for **direct manipulation (DM)** on Android phone. Implementing icons into buttons, applying accordions and dialogue boxes, displaying informative and considerable error messages, realizing user-led repair mechanism and well-arranged menus make it an acceptable interface for market promotion. These characteristics are illustrated as follows.

Icons

Figure 9 is the collections of icons I implemented in my interface. In the unselected mode, all the icons are in white as default, standing out from the dark background.

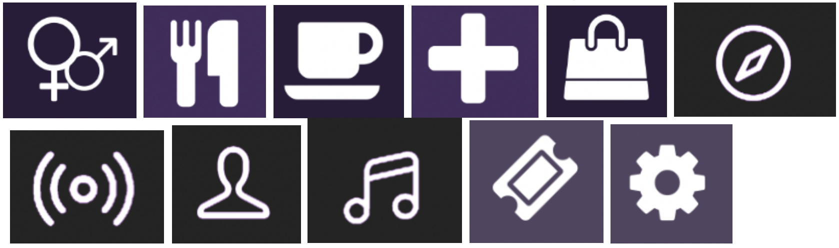


Figure 9 - icons

When the icons in the bottom side bar are selected, they would turn to purple, ensuring that the selected one is visible from others. Figure 10 compares the two statuses of the Map icon. The left-hand side picture is when Event page is opened, while the right-hand side is when Map page is opened.

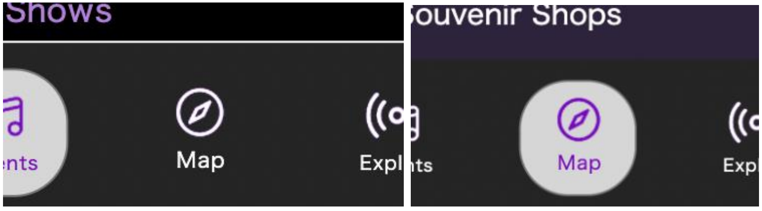


Figure 10 – two statuses of Map icon

Metaphor

The icons are abstract figures of its original meaning, therefore metaphors exist extensively in icons I used. For example, the Map icon ⊘ represents a compass, which is an ancient method to distinguish directions.

Size

According to the Android Icon Size guide for App Interface Icons[6], the Action Bar, dialogue and tab icons should uniformly be 24px. Figure 11 is an example of the settings icon showing its width and height.

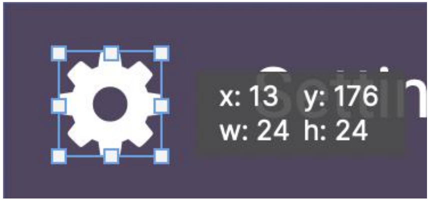


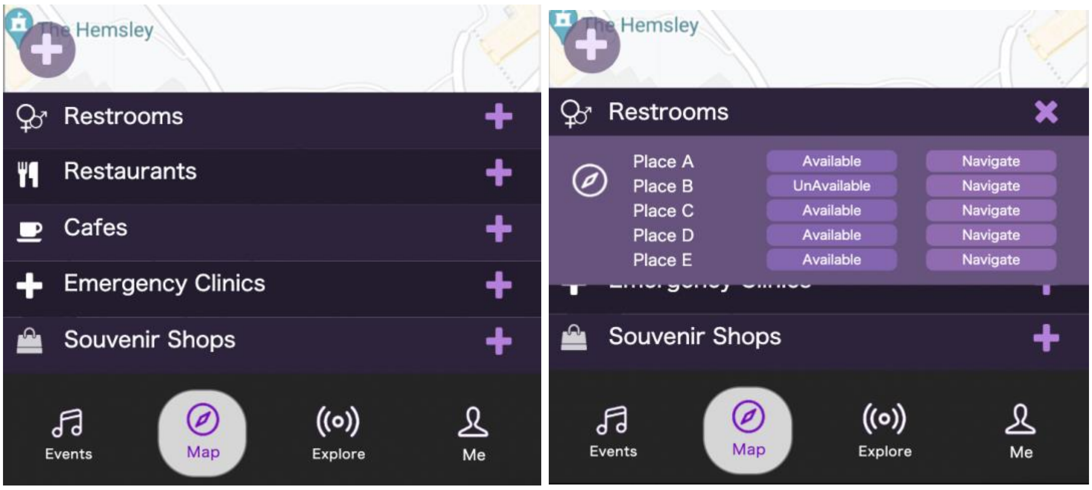
Figure 11 – the Width and Height of the Settings icon

[6] https://www.creativefreedom.co.uk/icon-designers-blog/android-icon-size-guide/

Dialogue boxes and Accordions

Figure 11 shows the dialogue boxes of facilities for users to view and choose. The dialogue boxes are designed to fit in an Android mobile phone. Figure 12 is the accordion of performances in all three days in Dates page.

It is suggested better to contain no more than 4 layout columns in a mobile phone.[7] Thus, the dialogue boxes and accordions shown below is widely applied in my app.



Place A	Available	Navigate
Place B	UnAvailable	Navigate
Place C	Available	Navigate
Place D	Available	Navigate
Place E	Available	Navigate

Figure 11 – dialogue box

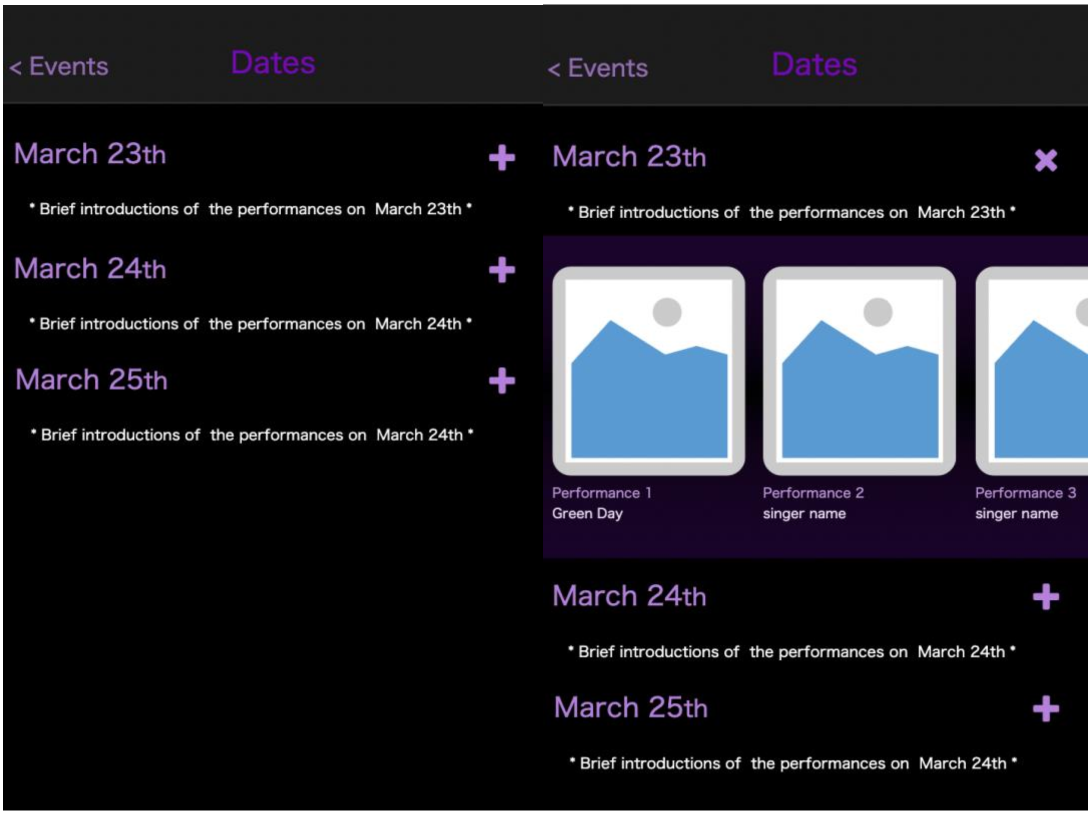
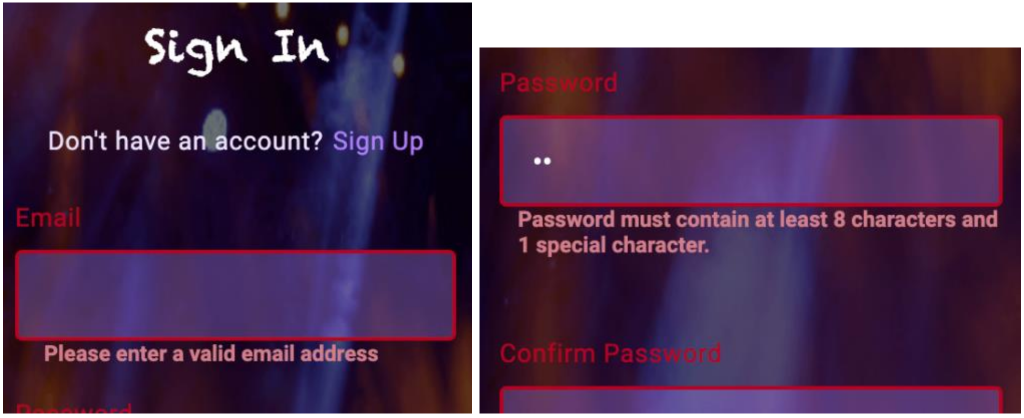

< Events
Dates
March 23th
• Brief introductions of the performances on March 23th •
March 24th
• Brief introductions of the performances on March 24th •
March 25th
• Brief introductions of the performances on March 24th •
< Events
Dates
March 23th
• Brief introductions of the performances on March 23th •
Performance 1
Green Day
Performance 2
singer name
Performance 3
singer name
March 24th
• Brief introductions of the performances on March 24th •
March 25th
• Brief introductions of the performances on March 24th •

Figure 12 – Accordion of all 3 days

[7] https://material.io/design/layout/understanding-layout.html#layout-anatomy

Error Messages

In the Sign In page, entering an invalid Email address would generate an error message. (Figure 13, left) The message is in a positive tone which won't frighten or embarrass users.



Please enter a valid email address

Password must contain at least 8 characters and 1 special character.

Figure 13 – Error Message of entering Invalid Email Address or Password

Entering an invalid password in Password Input bar in Sign Up form generates a message reminding the user of how to correct the input. (Figure 13, right)

Repair Mechanisms

Undoing the any commands is made possible in this interface. The UNDO/REDO commands are realised in the following ways.

a. When being redirected to a new page, the "back" button in the upper left corner of the screen is always there taking users back to the former page.

b. The search bar in Map page or Explore page enables user to clear out the value just entered. (Figure 14)

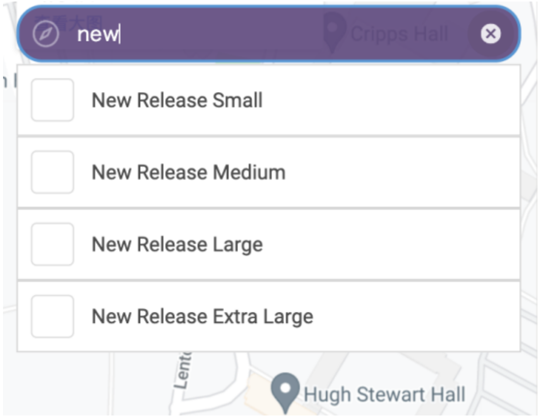


Figure 14 – The clear button on the upper right corner

c. The accordions and dialogue boxes (Figure 11 and 12) are all set up to **toggle**: tapping once hides or shows the accordion, while tapping twice inverses the last command.

Direct manipulation

DM often display continuous representation of the objects of interest, and utilise physical actions instead of syntax[8]. The icons used in buttons is a suitable example: they are abstract

[8] https://www.nngroup.com/articles/direct-manipulation/

figures derived from the reality. Therefore they are easy to recognise and perceive, taking a less learning time compared to recalling all complex syntaxes.

Considering the 4 ways of conceptualising interaction types, this interface falls into the category of manipulating. Depending on what the user intend to achieve, the actions could be swiping, taping, zooming, etc. These common manipulating actions can be found in the app. For example, tapping on labels will take users to the corresponding page, swiping on the poster would show the next slide either left or right (Figure 15). Zooming in or out the Map is also achieved.

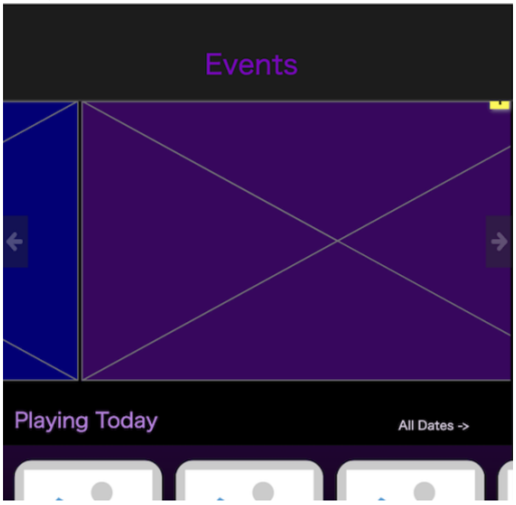


Figure 15 – Swipe for next Slide

Menus

Different forms of menus are implemented: group of pictures, accordions and dialogue boxes.

Structure

Menus of this interface is designed in cyclic structure. (Figure 16) The red lines indicates that the bottom side bar can take user directly back to the Event page. To avoid confusing cycles of menus, options other than back to previous page and root page are all unavailable.

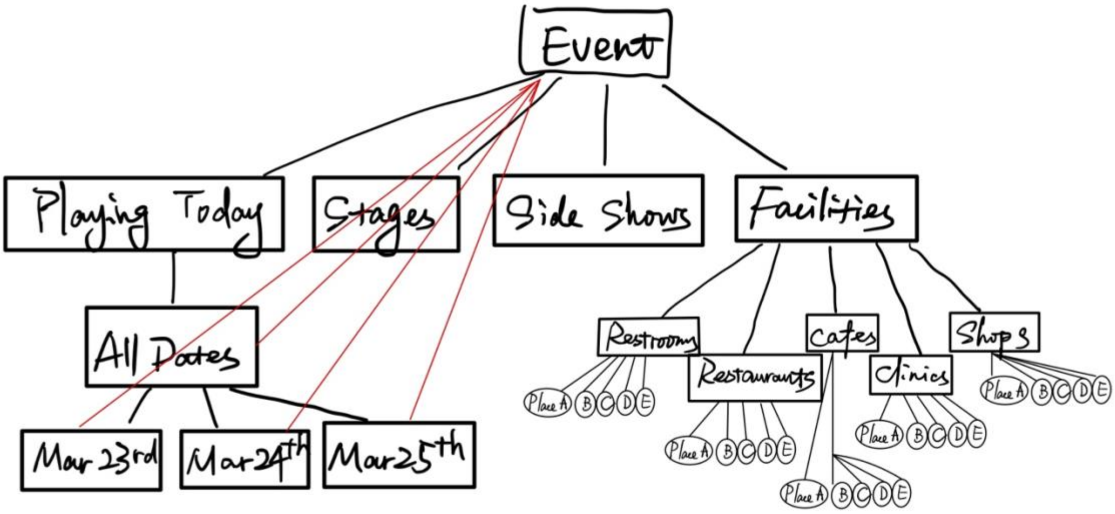
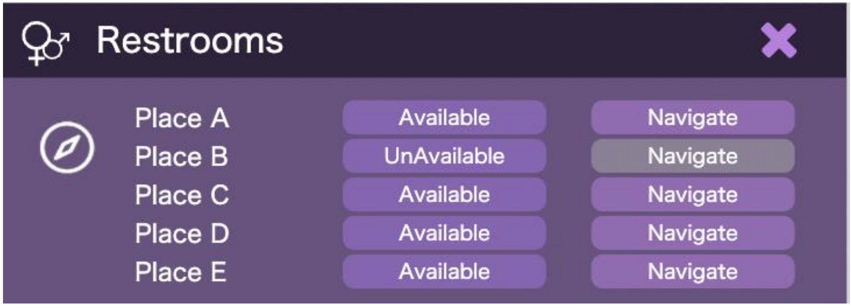


Figure 16 – Acyclic structures

Labelling items

To avoid confusion, unavailable choices in menus are greyed out. Take the restroom menu in Map page as an example (Figure 17).



Restrooms

Place A	Available	Navigate
Place B	UnAvailable	Navigate
Place C	Available	Navigate
Place D	Available	Navigate
Place E	Available	Navigate

Figure 17 – greyed out unavailable choices

Arrangement

From the structure of the menu, it is easy to tell that breadths are not all listed while depths almost are. The reason is that it is tedious and meaningless task to list all the performances in the music festival prototype. As long as the basic idea is presented, those breadths are of minor importance to designers and stakeholders.

Participatory Design and Social Interaction

User stories and Scenarios of use

Here are two user stories that I utilised while designing the system:

1. 20-year old Patrick is a fan of Green Day and don't want to miss one show of them. He bought tickets of March 23rd of this music festival. When entering this festival square, he opens "My Tickets" to show his Admission credential. When time approaches 13:00, Patrick vaguely remember the place and list of songs they are going to perform, so he opens Green Day performance detail page that he has added to his favourites. He tapped on Navigate to find his way to stage 1 on time and enjoyed the show very much.
2. After watching 2 hours of performance, Lisa (30) feels thirsty and tired. She is desperate for a cup of coffee and a cozy chair to sit in. She squeezes out of the croud but realizes she lost her way. She intends to find a café nearest to her current location, so she opens the Event page and scrolls down. By tapping on the compass icon, the app will naviagte her to a nearest café shop.

Types of prototyping

High-fidelity prototype

The tool Axure enables me to create a high-fidelity prototype which largely resemble the actual app.

Horizontal vs. Vertical prototyping

Horizontal prototyping presents a wide variety of functions, but does not fully implement all of them; Vertical prototyping does not attempt to show all the functionality in the system, but rather focuses on implementing a small number of functionality in a nearly complete way. [9]

Viewing the prototype as a whole, it is reasonable to conclude that this is a horizontal prototype. A wide range of functionality are presented, but when actually running every

[9] https://www.usabilityfirst.com/glossary/horizontal-and-vertical-prototypes/index.html

part of the function, it lacks detailed functions to some extent. For example, the Navigation button does not actually take users directly to the navigation page but only redirects them to the Map page.

Usability Components

Guessability

When using the system for the first time, it is common to be confused or over-whelmed by it. This app list performances, stages, dates, etc. in a relatively reasonable arrangement. In addition to the informative icons designed, it is not difficult for 18-35 to guess usages of the buttons.

Learnability

This app is targeted to people who are not computer experts nor geeks, and the scenario of use tends to happen in a daily casual environment, so the time of effort of reaching a competent level of performance is not hard.

System potential

This interface depends hard on search engines and organizing items into the correct list of menus. For example, robust algorithm must be developed to calculate which performances can be listed into the **top hits** menu. As for the search engine, users can use this app as a small map app that allows search for a variety of locations then navigate them (all though this is just an embedded Google Map page in the prototype).

Therefore, this system has a considerably high level of performance provided that all functionalities described above are well-implemented.

Re-usability

This app is bound to have a low re-usability rate since most music festivals only happen once a year in a certain city. This is uncontrollable for developers.

Regardless of this fact, this system could have a high reusability because it takes into account the scenarios of use, and are designed in a user-centered manner.