|  |
| --- |
| Zac Apelt |
| Unit 4 – Mobile Application |
| Low-Key Karaoke |

|  |
| --- |
|  |

# Problem description

Karaoke was first invented in Japan over 50 years ago and remains popular today, especially in the Philippines, Japan and South Korea [1]. Karaoke has many social and mental benefits, such as building confidence, encouraging positive social interactions and as a means to, for some people, have fun [2].

A mobile app is proposed to centralise information regarding karaoke events so that avid participants have an easy means to access necessary information. Such data includes upcoming karaoke events, the times they occur and their location, allows the user to create their own karaoke event and search for song lyrics to assist in the karaoke.

This report documents the process of developing the proof-of-concept mobile application, involving the interface, data exchange and data security processes.

# Breaking down the problem

## Mind map

A mind map was constructed to better understand the problems and challenges associated with the task.

Diagram

Description automatically generated

## Project Scope

As this is a low fidelity prototype, the app does not need to be complete with all the features, however, there should be a selection of features which are available. The short time period is the greatest limiting factor as this includes the time it takes to learn how to use the Python library Kivy. Kivy allows an app to be developed and built to a mobile phone. It was chosen due to past familiarity with python and a shorter expected learning curve. An API will be used to display song lyrics to the user and the user will have the ability to create their own karaoke events which will be displayed to other users. Users will be able to see events on a map and they will also have the ability to upload short snippets of audio to the database.

# Data components

There are many data components, structures and exchange methods involved in this project and they all must ensure that adequate data security met so that the users can feel safe using the app.

## Data structures

The two main data structures are XML and JSON.

XML was developed in 1997 and stands for Extensible Markup Language. It is divided up with a string of commands and utilizes a tag structure for easy communication of data. These identifying tags are similar to HTML, providing a rigid way to structure the data. The advantages of this method are that the data types are not limited to text or numbers, meaning XML is capable of transmitting images, tables and graphs without needing to convert the data in any way which many people argue makes it the more powerful data structure. XML is used extensively any time money is transferred over the internet, such as stock trading, internet banking and online retail (History Computer, 2022).

JSON was developed in 2001 and stands for JavaScript Object Notation. The data structure is derived from JavaScript and can be easily human readable but also condensed to reduce space. JSON offers many advantages over XML such as being faster due to the easy parsing of the data and faster execution of the data due to the syntax’s light weight nature. The structure makes it easier to format files and is more popular for newer APIs (R. Sharma, 2014).

The following graph shows the relative growth rates of XML and JSON.

Chart, line chart

Description automatically generated

## Data exchange methods

There are three data exchange elements, these are an architectural pattern, a data format and a communication protocol. These three elements are often combined together, for example, the RESTful API mechanism normally uses the Representation State Transfer (architecture), JSON (data format) and the secure HTTPS protocol (communication protocol).

There are five main data exchange mechanisms.

1. Application Programming Interface (API) - This method allows a web service to communicate using the HTTP protocol. Web services include SOAP and REST. “The REST architecture defines a set of guidelines to follow to provide a RESTful web service, for example, stateless existence and the use of HTTP status codes.” (Harvard University, 2020)
2. Extract, Transform and Load (ETL) - A direct connection is established between one application and another application’s database.
3. File Transfer - This method stores data in the form of JSON, XML, CSV or something similar within a file which is transferred to a destination.
4. Remote Procedure Call - A program causes a procedure to run in a different address space, normally on another computer or shared network
5. Data Streaming - One or more data sources continuously transfer data to a receiving process

Various factors determine which data exchange method would be a suitable solution, such as data complexity, frequency of data update and data set size to name a few (Charest, 2020).

## Data security

Its important

### CIA model

yep

# Success criteria

The project can be considered a success if the following criteria are met.

* The user can register a karaoke event
* The user can modify an existing karaoke event they created
* The user can view upcoming karaoke events
* The user can find the lyrics of a song based by entering the name and artist
* The user can find upcoming karaoke events on a map
* The user’s data is stored securely
* The user will be able to upload an audio snippet to the database
* User data such as song searches will be uploaded to the database
* The user can register their interest in attending an event

# Usability Success Criteria

Furthermore, the usability can be considered a success if the following criteria are met

* The events page will display karaoke event data to the user that is accessible and visual through the use of a map (effectiveness)
* The navigating through the app will be intuitive and learnable by utilizing a minimum number of buttons and pages (learnability)
* The app will be visually appealing through style and colour schemes so that the user has a positive experience with the system (visual design)
* The app will be secure for the users’ data (safety)
* The user will be able to create a karaoke event simply and efficiently (effectiveness)
* The user will be able to view the lyrics of a song in a visually appealing and readable format (visual design and effectiveness)
* The user will be able to remove an event they registered and unregister their interest in an event (safety)

# Site Map

The following site map shows how the pages of the app will connect together.Diagram

Description automatically generated

User interface design

The following interface design shows the style that the app will be designed in.

Graphical user interface, text, application

Description automatically generated

# Data flow diagram

he following diagram shows how data moves between the systems.Diagram, schematic

Description automatically generated

# Entity relationship map

Missing some pk but this is the general idea

The following entity table relationship map was created to normalise and demonstrate the structure of the database.Diagram

Description automatically generated

# Pseudo code

# SQL create code

# Test Log

|  |  |
| --- | --- |
| **problem** | **Solution** |
| Screen slide transition not working | Stop the screen being centered in the window as it was preventing the screen to slide. Instead, dynamically change the window size to match the predefined size of the screen |
| Uploading the event registration information to the data wasn’t working | I tried uploading data to a table with more data than the table could hold. The solution was to modify the database to include the necessary fields. |
| Converting a postal address to latitude and longitude geolocation would consistently fail and gave coordinates to Bangladesh. | Geopy required a special data format that it wasn’t being given. Adding another field for the user to input the city name and appending it to the end of the address worked. |
|  |  |
|  |  |
|  |  |
|  |  |

# Evaluation

# Recommendations

# Impacts

## Threats

## Property threatened

## Potential impact

## Countermeasures

# Australian privacy principles

# Security strategy

# References

1. <https://www.howtokaraoke.com/a-brief-history-of-karaoke-where-did-it-all-come-from/>
2. <https://blog.ambient-mixer.com/benefits/9-awesome-benefits-from-singing-karaoke/>
3. <https://ezeelive.com/json-advantages-disadvantages/>
4. <https://history-computer.com/xml-vs-json/>
5. https://enterprisearchitecture.harvard.edu/data-exchange-mechanisms#:~:text=Three%20of%20the%20basic%20data,to%20communicate%20and%20exchange%20data.