

# Change History

| Version | Notes |
| --- | --- |
| 0.1 | * First Iteration |
| 0.2 | * Changed format of NFRs * Defined business value and difficulty for requirements * Added colours to the requirement tables * Removed use cases * Removed some irrelevant definitions |
| 1.0 | * Moved “Key Actors” to section 2 * Split functional and non-functional requirements into subsections. * Reformatted section 1 * Changed colours of the requirement tables |
| 2.0 | * Changed some requirements from “should have” to “must have” |

# Glossary

| Term | Definition |
| --- | --- |
| Framework | A platform for developing software applications. Contains in-built functionality to assist in the development of software. |
| Front-end | The front-end of an application is defined as what the users can see and directly interact with. |
| Back-end | The back-end of an application relates to the server-side tasks and database communication. |
| Database | A structured set of data held in a computer or server. |
| SDK  (Software Development Kit) | A collection of software development tools in one package. |
| API  (Application Programming Interface) | A software interface which offers a service to other pieces of software. |
| WebSocket | A web communication protocol allowing two-way communication between a client and a server. |
| FIFO | A process for ordering data, where the initial data that is added to a queue will be the data that is taken first from the queue. |

Contents

[**Change History**](#_heading=h.bwu00cx13jnv) **2**

[**Glossary**](#_heading=h.jpy9isf3la1) **2**

[**1 Introduction**](#_heading=h.n9bdx1pa2en0) **4**

[1.1 Purpose](#_heading=h.x4wecth77ukv) 4

[1.2 Audience](#_heading=h.3o7alnk) 4

[1.3 Project Background](#_heading=h.23ckvvd) 4

[**2 Overall Description**](#_heading=h.yzkh776mlsmq) **5**

[2.1 Key Actors](#_heading=h.86a8a06d4siw) 5

[2.2 Product Functions](#_heading=h.swotgrq2hkpo) 5

[2.3 Assumptions and Dependencies](#_heading=h.lnxbz9) 6

[**3 External Interface Requirements**](#_heading=h.5t1xo2akgl58) **7**

[3.1 User Interfaces](#_heading=h.1ksv4uv) 7

[3.2 Hardware Interfaces](#_heading=h.44sinio) 7

[3.3 Communications Interfaces](#_heading=h.z337ya) 7

[**4 Functional Requirements**](#_heading=h.m119jqscl2jv) **8**

[4.1 Evaluation Criteria](#_heading=h.rohcougq6xz4) 8

[4.2 Requirements per Product Function](#_heading=h.ln5uglxygkif) 8

[4.2.1 User Accounts and Storage](#_heading=h.83vewyf4etd4) 8

[4.2.2 Virtual Jukebox Sessions](#_heading=h.7uzdchbr20m4) 9

[4.2.3 Song Queue](#_heading=h.tdbbu9eqv5rk) 10

[4.2.4 Chat Room](#_heading=h.3g2702ym0n4t) 11

[4.2.5 Virtual Jukebox Map](#_heading=h.segff0rvt4xu) 12

[4.2.6 Music Playback](#_heading=h.n050s11yxko2) 12

[**5 Non-Functional Requirements**](#_heading=h.3gxd60s45ugr) **14**

[5.1 Evaluation Criteria](#_heading=h.z9q1frz1lb9x) 14

[5.2 Requirements per Category](#_heading=h.6ccj0cdmh7cb) 14

[5.2.1 Performance Requirements](#_heading=h.xkht1758j0n4) 14

[5.2.2 Reliability Requirements](#_heading=h.6iydz7j5b5wb) 15

[5.2.3 Usability Requirements](#_heading=h.vqlajeaiw61) 15

[5.2.4 Security Requirements](#_heading=h.iaayczcg4vqk) 16

[5.2.5 Legal Requirements](#_heading=h.lw58m5dxhb1v) 16

# 

# 1 Introduction

## 1.1 Purpose

This report details the functional and non-functional requirements for the Virtual Jukebox application. This document will additionally summarise the key actors, product functions, and external interface requirements that are defined for the application.

## 1.2 Audience

This report is aimed at the stakeholders involved in the development of the Virtual Jukebox application. The main stakeholder is Amristar, which is the supervisor and client of the project. Stakeholders also include the staff involved with Computing Capstone Project 1 at Curtin University, which act as the co-supervisors, and the Developers of the application.

## 1.3 Project Background

Jukeboxes were once a common sight at bars, discos and any social gathering. They allowed users to choose a song from a playlist to add to a queue that would eventually be played at the gathering. However, due to the explosive rise of music streaming services, jukeboxes have since become nothing more than a novelty. The Virtual Jukebox application aims to reproduce the functionality of a physical jukebox in a digital, web application format. Users will be able to connect to a local jukebox by selecting a location on a map or by scanning a QR code, and be able to vote for songs from a playlist which will be added to the queue.

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 2 Overall Description

## 2.1 Key Actors

**Virtual Jukebox Application:** The web application itself. The Virtual Jukebox application is responsible for providing users the ability to set up and join Virtual Jukebox sessions, managing these sessions’ song queues, and in turn, handling music playback via Spotify (each defined in section 2.2). It will also be responsible for managing host and guest accounts, and managing each chat room per Virtual Jukebox session. It will also support the Virtual Jukebox map (defined in section 2.2).

**Hosts:** Type of user. Hosts must create and log into an account to be able to host Virtual Jukebox sessions. Hosts may create a Virtual Jukebox session, and can control the music playback. Hosts can vote for songs in the song queue, and add songs from the session playlist into the song queue.

**Guests:** Type of user. Guests can create and log into an account, join a session, queue songs in a session, send messages, and vote on the current song queue. Guests are not required to log into an account to join sessions.

## 2.2 Product Functions

**User Accounts and Storage:** The Virtual Jukebox application will allow hosts to create a Virtual Jukebox account and login using their credentials. A host must authenticate an existing Spotify Premium account to their Virtual Jukebox account. The application will also allow guests to generate and use a guest account to join a session, which will be generated with a random username.

**Virtual Jukebox Session:** A Virtual Jukebox Session is set up by hosts, and joinable by guests. These sessions can be public, meaning they can be joined by any guest within the specified range of the jukebox without a password, or can be private (requires a password). Guests can join a Virtual Jukebox session in one of two ways. One way is by scanning a session’s QR code (or directly visiting the link), which is generated when a host creates a session (bypassing passwords if any). The other way is by selecting the session on the Virtual Jukebox map. Each Virtual Jukebox session will have a playlist acting as a pool of songs which users can choose from to add to the song queue. This playlist will be imported from the host’s Spotify account.

**Song Queue:** Virtual Jukebox sessions will utilise a song queue which is viewable by all guests as well as the host. The queue will control the ordering of song playback on the host device. The user-queued songs will be ordered in a first-in-first-out manner by default, but songs with votes will be prioritised in playback order. Songs with the same amount of votes will remain in a FIFO order. When no songs are queued by users, the host device will randomly play songs in the host defined playlist.

**Chat Room:** The Virtual Jukebox application will support a chat room in which guests and hosts connected to the same session can send and receive chat messages. Messages will be filtered for profanity, and a popup will display for any guest or host which sends a message which triggers the profanity filter.

**Virtual Jukebox Map:** The Virtual Jukebox application will support an interactive map which will show the location of all active Virtual Jukebox sessions. Guests can join a Virtual Jukebox session from the map if they are in a certain range. Hosts are able to select their location on the map using either a GPS or manual input when setting up a Virtual Jukebox session.

**Music Playback:** The Virtual Jukebox application will support music streaming through the Spotify SDK. Streaming will be played through the host’s device, and the host can control playback by skipping, pausing or playing the current song.

## 2.3 Assumptions and Dependencies

For the software to remain fully functional, it will require a web server with a database to store user account data. It will also require access to the Spotify SDK to enable music streaming functionality in Virtual Jukebox sessions. Additionally, the Mapworks SDK will be necessary for Virtual Jukebox map functionality.

Hosts will also require a Spotify Premium account, and link it to their Virtual Jukebox accounts to enable the ability to host Virtual Jukebox sessions.

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 3 External Interface Requirements

## 3.1 User Interfaces

The user interface for the Virtual Jukebox application will be compatible with both mobile and desktop devices, and will support a scalable UI for a consistent user experience between desktop and mobile devices alike.

## 3.2 Hardware Interfaces

To operate the software, users will need access to a mobile phone, laptop, tablet, or desktop computer. Hosts will require access to devices with speakers (either connected internally, via cable, or wirelessly) to act as a music source.

## 3.3 Communications Interfaces

Users will require access to a web-browser, specifically Microsoft Edge, Safari, Google Chrome, or Mozilla Firefox within the latest two versions.

Other communication interfaces include client-server communication which will be achieved via the Django back-end framework, HTTPS, encryption, and WebSocket for synchronisation.

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 4 Functional Requirements

## 4.1 Evaluation Criteria

**Difficulty:** Either “low”, “medium”, or “high”. Evaluates difficulty according to an estimated amount of time taken. Difficulty is relative to the other requirements; a requirement with a “high” difficulty has a higher estimated time requirement than a “medium” difficulty requirement.

**Business Value:** Based on the MoSCoW method. A requirement which is deemed to be a “must have” will have a “high” business value. “Should have” requirements will have a “medium” business value. and “could have” requirements will have a “low” business value.

## 4.2 Requirements per Product Function

Note: “All users” refers to guests and hosts.

### 4.2.1 User Accounts and Storage:

| **Requirement** | **Description** | **Difficulty** | **Business Value** |
| --- | --- | --- | --- |
| VJ-1 | The system must allow all users to create a Virtual Jukebox account with an email and password. | High | High |
| VJ-2 | The system must allow all users to login to their Virtual Jukebox account with an email and password. | Medium | High |
| VJ-3 | The system must allow all users to connect their Virtual Jukebox account with a Spotify Premium account. | Low | High |
| VJ-4 | The system should allow the host to view the existing playlists of a connected Spotify Premium account. | Low | Medium |
| VJ-5 | The system must allow guests to login with a guest account, which will generate a random username. | Low | High |
| VJ-6 | The system could allow guests to set a username for their guest account. | Low | Low |
| VJ-7 | The system could allow hosts to change the username of their existing account. | Low | Low |
| VJ-8 | The system could allow hosts to change the public profile picture of their existing account. | Medium | Low |
| VJ-9 | The system should allow users to reset the password of their Virtual Jukebox account from the login page. | Medium | Medium |
| VJ-10 | The system could support two factor authentication for logging in to a Virtual Jukebox account. | Medium | Low |

### 4.2.2 Virtual Jukebox Sessions:

| **Requirement** | **Description** | **Difficulty** | **Business Value** |
| --- | --- | --- | --- |
| VJ-11 | The system must allow hosts to use an existing Spotify playlist to use as the Virtual Jukebox session playlist. | Low | High |
| VJ-12 | The system must allow hosts to create either a public or private (requires password) Virtual Jukebox session. | Low | High |
| VJ-13 | The system should allow hosts to enter a publicly displayed name for their hosted Virtual Jukebox session. | Low | Medium |
| VJ-14 | The system should allow hosts to enter a publicly displayed description for their hosted Virtual Jukebox session. | Low | Medium |
| VJ-15 | The system should hosts to generate a QR code which will allow guests to connect to the currently hosted Virtual Jukebox session. | Medium | High |
| VJ-16 | The system should allow guests to join a public or private Virtual Jukebox session with a QR code. | Medium | High |
| VJ-17 | The system should allow hosts to generate a URL which will allow guests to join a Virtual Jukebox session directly. | Medium | High |
| VJ-18 | The system should allow guests to use a URL to join a Virtual Jukebox session directly. | Medium | High |
| VJ-19 | The system must allow all users connected to a Virtual Jukebox session to view the current song queue. | Low | High |
| VJ-20 | The system must allow all users connected to a Virtual Jukebox session to view the playlist selected by the host for the session. | Low | High |
| VJ-21 | The system must allow all users connected to a Virtual Jukebox session to view the currently playing song. | Low | High |
| VJ-22 | The system should allow all users connected to a Virtual Jukebox session to search the current session playlist by song title. | Low | Medium |
| VJ-23 | The system could allow all users connected to a Virtual Jukebox session to search the current session playlist by artist name. | Low | Low |
| VJ-24 | The system must end a Virtual Jukebox session if the host selects to end the session. | Medium | High |
| VJ-25 | The system must end a Virtual Jukebox session if the host disconnects from the session. | Medium | High |

### 4.2.3 Song Queue:

| **Requirement** | **Description** | **Difficulty** | **Business Value** |
| --- | --- | --- | --- |
| VJ-26 | The system must allow all users connected to a Virtual Jukebox session to vote on songs in the song queue. | Medium | High |
| VJ-27 | The system should allow all users connected to a Virtual Jukebox session to earn credits over time. | Low | Medium |
| VJ-28 | The system must allow all users connected to a Virtual Jukebox session to use credits to add a song from the Virtual Jukebox session playlist into the song queue. | Medium | High |
| VJ-29 | The system must sort the current song queue by number of votes. | Low | High |
| VJ-30 | The system should sort the current song queue by the order songs were added, provided the number of votes are equal. | Low | Medium |
| VJ-31 | The system must play the Virtual Jukebox session playlist on shuffle playback if no songs are currently in the song queue. | Low | High |
| VJ-32 | The system should restrict all users from adding songs to a Virtual Jukebox session’s song queue that are already queued. | Low | Medium |
| VJ-33 | The system should restrict all users from adding the currently playing song in the Virtual Jukebox session to the song queue. | Low | Medium |
| VJ-34 | The system should not allow all users to have more than 5 current credits at any given time. | Low | Medium |
| VJ-35 | The system should enforce a minimum length for a Virtual Jukebox session playlist of 10 songs. | Medium | Medium |
| VJ-36 | The system could support all users removing votes on the song queue once they have been placed. | Low | Low |

### 4.2.4 Chat Room:

| **Requirement** | **Description** | **Difficulty** | **Business Value** |
| --- | --- | --- | --- |
| VJ-37 | The system should allow all users connected to a Virtual Jukebox session to view the chat room | Low | Medium |
| VJ-38 | The system should allow all users connected to a Virtual Jukebox session to send and receive messages in the chat room. | High | Medium |
| VJ-39 | The system should allow all users connected to a Virtual Jukebox session to view all sent and received messages in the chat room since they have joined. | Medium | Medium |
| VJ-40 | The system must filter profanity from all Virtual Jukebox session chat rooms. | Medium | High |
| VJ-41 | The system could display a popup for a user if a message they send is filtered by the profanity filter. | Low | Low |
| VJ-42 | The system should limit the number of messages a user can send in a period of time in a Virtual Jukebox chat room. | Low | Medium |

### 4.2.5 Virtual Jukebox Map:

| **Requirement** | **Description** | **Difficulty** | **Business Value** |
| --- | --- | --- | --- |
| VJ-43 | The system should allow hosts to select a location on the Virtual Jukebox Map to choose the location of a Virtual Jukebox session. | Low | Medium |
| VJ-44 | The system could allow hosts to select a radius in which guests must be within to join for a hosted Virtual Jukebox session. | Low | Low |
| VJ-45 | The system should allow guests to view all Virtual Jukebox sessions on the Virtual Jukebox Map. | Medium | High |
| VJ-46 | The system should allow guests to join a public Virtual Jukebox session by clicking the session on the Virtual Jukebox map, provided they are within the radius specified by the host. | Medium | High |
| VJ-47 | The system should allow guests to join a private Virtual Jukebox session by clicking the session on the Virtual Jukebox map, and entering a password, provided they are within the radius specified by the host. | Medium | High |

### 4.2.6 Music Playback:

| **Requirement** | **Description** | **Difficulty** | **Business Value** |
| --- | --- | --- | --- |
| VJ-48 | The system must play the Virtual Jukebox session’s music through the host’s connected device. | Low | High |
| VJ-49 | The system should allow the host to pause, play or skip the current playing song for a Virtual Jukebox session. | Low | Medium |

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 5 Non-Functional Requirements

## 5.1 Evaluation Criteria

**Difficulty:** Either “low”, “medium”, or “high”. Evaluates difficulty according to an estimated amount of time taken. Difficulty is relative to the other requirements; a requirement with a “high” difficulty has a higher estimated time requirement than a “medium” difficulty requirement.

**Business Value:** Based on the MoSCoW method. A requirement which is deemed to be a “must have” will have a “high” business value. “Should have” requirements will have a “medium” business value. and “could have” requirements will have a “low” business value.

## 5.2 Requirements per Category

Note: “All users” refers to guests and hosts.

### 5.2.1 Performance Requirements

| **Requirement** | **Description** | **Difficulty** | **Business Value** |
| --- | --- | --- | --- |
| VJ-50 | The system must update the song queue within **two** seconds of a vote being placed. | High | High |
| VJ-51 | The system must update the chat-room within **two** seconds of a message being sent. | Medium | High |
| VJ-52 | The system must generate a QR code (for joining a Virtual Jukebox session) within **five** seconds of a Virtual Jukebox session being created. | High | High |
| VJ-53 | The system must be able to process up to **100** messages per second. | High | High |
| VJ-54 | The system must be able to process up to **300** votes per second. | High | High |
| VJ-55 | The system must retrieve and validate user details for Virtual Jukebox accounts from the database within **three** seconds after a login request. | High | High |
| VJ-56 | The system must load each page within **two** seconds (to be evaluated on an internet connection of **5MBps speed)**. | High | High |
| VJ-57 | The system must support song queue sizes of up to **10 000** songs. | Medium | High |
| VJ-58 | The system must support playlist sizes of up to **10 000** songs. | Medium | High |
| VJ-59 | The system should send a password reset email within one minute of a password reset request being received. | High | Medium |
| VJ-60 | The system must create Virtual Jukebox accounts within **five** seconds of the creation form being submitted | High | High |

### 5.2.2 Reliability Requirements

| VJ-61 | The system must be available for **99.9%** of any given month. | High | High |
| --- | --- | --- | --- |
| VJ-62 | The system’s profanity filter should fail for no more than **one in 1000** messages. | High | Medium |
| VJ-63 | The system must play the most voted song **99.99%** of the time. | High | High |
| VJ-64 | The system must display the Virtual Jukebox session on the correct map location **99.99%** of the time. | High | High |

### 5.2.3 Usability Requirements

| VJ-65 | The system must allow guests to join a Virtual Jukebox session as a guest within **three** clicks (minus passcode entry). | Medium | High |
| --- | --- | --- | --- |
| VJ-66 | The system must allow all users to queue a song while in a Virtual Jukebox session within **three** clicks. | Medium | High |
| VJ-67 | The system must display the song queue and the chat on one screen on a 16:9 screen. | Medium | High |
| VJ-68 | The system should display media controls to the host within a Virtual Jukebox session at all times. | Medium | Medium |
| VJ-69 | The system must provide a consistent UI across common desktop and mobile screens. | High | High |
| VJ-70 | The system must support the following sizes, ratios and browsers ([1920x1080, 2k, 4k], [16:9, 4:3], [Chrome, Firefox, MS Edge, and Safari]). | High | High |
| VJ-71 | The system must provide a UI such that **95%** of guests can navigate to join a Virtual Jukebox session without assistance. | High | High |

### 5.2.4 Security Requirements

| VJ-72 | The system must store encrypted Virtual Jukebox account passwords. | Medium | High |
| --- | --- | --- | --- |
| VJ-73 | The system should store encrypted Virtual Jukebox session passwords. | Medium | Medium |

### 5.2.5 Legal Requirements

| VJ-74 | The system must display a disclaimer stating that the application is not to be used commercially due to Spotify’s terms and conditions. | Low | High |
| --- | --- | --- | --- |

## 