

---

# SOFTWARE REQUIREMENTS SPECIFICATION

for

<Project>

Version 1.0 approved

Prepared by <author>

<Organization>

June 29, 2018

# Contents

<b>1</b>	<b>Introduction</b>	<b>5</b>
1.1	Purpose . . . . .	5
1.2	Document Conventions . . . . .	5
1.3	Intended Audience and Reading Suggestions . . . . .	5
1.4	Project Scope . . . . .	5
1.5	References . . . . .	6
<b>2</b>	<b>Overall Description</b>	<b>7</b>
2.1	Product Perspective . . . . .	7
2.2	Product Functions . . . . .	7
2.3	User Classes and Characteristics . . . . .	7
2.4	Operating Environment . . . . .	7
2.5	Design and Implementation Constraints . . . . .	8
2.6	User Documentation . . . . .	8
2.7	Assumptions and Dependencies . . . . .	8
<b>3</b>	<b>External Interface Requirements</b>	<b>9</b>
3.1	User Interfaces . . . . .	9
3.2	Hardware Interfaces . . . . .	9
3.3	Software Interfaces . . . . .	9
3.4	Communications Interfaces . . . . .	9
<b>4</b>	<b>System Features</b>	<b>11</b>
4.1	System Feature 1 . . . . .	11
4.1.1	Description and Priority . . . . .	11
4.1.2	Stimulus/Response Sequences . . . . .	11
4.1.3	Functional Requirements . . . . .	11
4.2	System Feature 2 (and so on) . . . . .	11
<b>5</b>	<b>Other Nonfunctional Requirements</b>	<b>12</b>
5.1	Performance Requirements . . . . .	12
5.2	Safety Requirements . . . . .	12
5.3	Security Requirements . . . . .	12
5.4	Software Quality Attributes . . . . .	12
5.5	Business Rules . . . . .	12
<b>6</b>	<b>Other Requirements</b>	<b>13</b>
6.1	Appendix A: Glossary . . . . .	13

6.2	Appendix B: Analysis Models . . . . .	13
6.3	Appendix C: To Be Determined List . . . . .	13

## Revision History

Name	Date	Reason For Changes	Version
Cindy	06/28	wrote five sections	1
4	32	33	34

# 1 Introduction

## 1.1 Purpose

藉Line Bot系統實現根據使用者回傳的對話內容關鍵字，判斷使用者當下心情，並回傳相應的心情歌曲給使用者。希望透過與我們的Line Bot可以以音樂去安慰不开心的人，同時也可以讓开心的有更好的心情。另外藉由對話中出現的歌手名稱，可隨機推薦該歌手的影片MV連結。這是此程式的第一個版本，可以自己運作的主程式，而不是其他程式的部分元件。

## 1.2 Document Conventions

<定義文件中使用的技術性字詞Describe any standards or typographical conventions that were followed when writing this SRS說明任何寫這份文件需要的標準或排版慣例，such as fonts or highlighting that have special significance比如有特殊涵義的字或醒目標註. For example, state whether priorities for higher-level requirements are assumed to be inherited by detailed requirements, or whether every requirement statement is to have its own priority.>

## 1.3 Intended Audience and Reading Suggestions

<Describe the different types of reader that the document is intended for, such as developers, project managers, marketing staff, users, testers, and documentation writers. Describe what the rest of this SRS contains and how it is organized. Suggest a sequence for reading the document, beginning with the overview sections and proceeding through the sections that are most pertinent to each reader type.>

## 1.4 Project Scope

使用的軟件分別是Microsoft Azure和Line Bot。Microsoft Azure 中的文字分析的功用是可以對輸入文字進行情緒分析，這API會為這句文字計算出一個心情數值由 0 到 1 之間，愈小表示心情愈不好，相反愈大表示愈開心。而Line Bot是Line提供的API，可以與機器人進行構通。為了達到產品可用音樂改善使用者的心情的這目的，而結合這兩項軟件。任何人在任何時間都可以利用Line Bot去傳信息給機械人並把收到的文字用Azure的文字分析得到使用者心情 再從歌單中選出音樂推薦傳送給使用者。

## 1.5 References

1. **LINE developers**  
<https://developers.line.me/console/channel/1587699018/basic/>
2. **Heroku**  
<https://dashboard.heroku.com>
3. **LineBot+Python**, 輕鬆建立聊天機器人  
<https://yaoandy107.github.io/line-bot-tutorial/>
4. 文字分析API — **Microsoft Azure**  
<https://azure.microsoft.com/zh-tw/services/cognitive-services/text-analytics/>
5. **LINE BOT** 通訊  
<https://engineering.linecorp.com/tw/blog/detail/183>

## 2 Overall Description

### 2.1 Product Perspective

這部分還沒弄Diagram圖表在

<Describe the context and origin of the product being specified in this SRS. For example, state whether this product is a follow-on member of a product family, a replacement for certain existing systems, or a new, self-contained product. If the SRS defines a component of a larger system, relate the requirements of the larger system to the functionality of this software and identify interfaces between the two. A simple diagram that shows the major components of the overall system, subsystem interconnections, and external interfaces can be helpful.>

### 2.2 Product Functions

<Summarize the major functions the product must perform or must let the user perform將主要需要使產品運作的或需要使用者觸發的函數做個總結. Details will be provided in Section 3這部分的細目會在Section3中呈現, so only a high level summary所以不用寫得太細 (such as a bullet list比如使用項目清單) is needed here. Organize the functions to make them understandable to any reader of the SRS在可以讓所有閱讀者了解的情況下. A picture of the major groups of related requirements functions and how they relate一張呈現主要需求的函式與他們之間關係的圖表, such as a top level data flow diagram or object class diagram, is often effective.>

### 2.3 User Classes and Characteristics

本line bot預期的主要用戶為年輕人、line的中文用戶，因此本line bot推薦的音樂主要為中文的流行歌曲，並且在輸入方面也要求中文輸入。

### 2.4 Operating Environment

這個程式因為連結了line developer的後台系統，依賴著line現有的應用程式進行bot的實作。因此這個程式僅可以在支持line系統安裝的裝置上運作。目前智慧型手機主要使用的兩大作業系統，Android與IOS均有支持line系統的安裝。這個程式本身會回傳的語言為中文。只要裝置能安裝支持line bot系統使用的line應用程式，並能正常顯示繁體中文字，就可以正常使用本程式。

## **2.5 Design and Implementation Constraints**

<Describe any items or issues that will limit the options available to the developers. These might include: corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer's organization will be responsible for maintaining the delivered software).>

## **2.6 User Documentation**

<List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards.>

## **2.7 Assumptions and Dependencies**

<List any assumed factors (as opposed to known facts) that could affect the requirements stated in the SRS. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external factors, such as software components that you intend to reuse from another project, unless they are already documented elsewhere (for example, in the vision and scope document or the project plan).>



## 3 External Interface Requirements

### 3.1 User Interfaces

<Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., help) that will appear on every screen, keyboard shortcuts, error message display standards, and so on. Define the software components for which a user interface is needed. Details of the user interface design should be documented in a separate user interface specification.>

### 3.2 Hardware Interfaces

<Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware, and communication protocols to be used.>

### 3.3 Software Interfaces

<Describe the connections between this product and other specific software components (name and version), including databases, operating systems, tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Refer to documents that describe detailed application programming interface protocols. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.>

### 3.4 Communications Interfaces

User使用Line Bot前端介面輸入訊息後，Line平台為了通訊的安全問題，資料都要會透過加密通道。所以在架Messaging API的Webhook時，一定是用HTTPS來進行通訊協定。而訊息傳出後經過根據Webhook URL（即Heroku的domain位置），通過Message API把訊息導到後端Heroku處理，用HTTPS傳回到Message API，Heroku會根據Line

Bot的Channel secret及Channel access token兩個key知道是哪個Line Bot 最後把訊息再發到平台上回覆使用者。[5]

## 4 System Features

<This template illustrates organizing the functional requirements for the product by system features, the major services provided by the product. You may prefer to organize this section by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>

### 4.1 System Feature 1

<Don't really say "System Feature 1." State the feature name in just a few words.>

#### 4.1.1 Description and Priority

<Provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. You could also include specific priority component ratings, such as benefit, penalty, cost, and risk (each rated on a relative scale from a low of 1 to a high of 9).>

#### 4.1.2 Stimulus/Response Sequences

<List the sequences of user actions and system responses that stimulate the behavior defined for this feature. These will correspond to the dialog elements associated with use cases.>

#### 4.1.3 Functional Requirements

<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use "TBD" as a placeholder to indicate when necessary information is not yet available.>

<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>

REQ-1: REQ-2:

### 4.2 System Feature 2 (and so on)

## 5 Other Nonfunctional Requirements

### 5.1 Performance Requirements

由於本程式的將伺服器建置在heroku的免費平台上，因此，倘若在30分鐘以內皆無使用者對此程式進行輸入，則伺服器會自動關機，並在下一次接收到輸入時才重新啟動，所以使用者第一次輸入時可能會有5-10秒的延遲時間。另外，因本程式在伺服器端主要運算皆仰賴於爬蟲，所以如果同始有多位使用者進行輸入會造成伺服器頻繁的項目標網站送出請求，可能會導致伺服器產生錯誤，無法正確回應使用者。

### 5.2 Safety Requirements

本程式為依賴著line bot系統的一個互動程式。因為系統主要擷取使用者對話判斷其心情並做進一步處理，因此line系統本身擁有的「拍照並傳送圖片」、「傳送圖片」、「錄製語音並傳送」、「傳送貼圖」等等其他功能，使用者雖然可以正常發送，但本程式撰寫出來的bot系統無法對其內容做出處理。僅會從預寫的文字訊息中隨機回覆一句，告知使用者已收到訊息，但無法處理。

### 5.3 Security Requirements

本程式程式本體儲存在Heroku，藉由Heroku系統將文字傳給line系統，因此文字訊息內容的安全性主要依賴line系統本身的加密系統。並不需要額外的加密系統。

### 5.4 Software Quality Attributes

<Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.>

### 5.5 Business Rules

<List any operating principles about the product, such as which individuals or roles can perform which functions under specific circumstances. These are not functional requirements in themselves, but they may imply certain functional requirements to enforce the rules.>

## 6 Other Requirements

<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>

### 6.1 Appendix A: Glossary

<Define all the terms necessary to properly interpret the SRS, including acronyms縮寫 and abbreviations縮寫. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>

### 6.2 Appendix B: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

### 6.3 Appendix C: To Be Determined List

<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>