

ARCHITECTURE

Personal Audio Book Desktop App

Author: Shraddha Giradkar

Abstract

3

1 Error! Bookmark not defined.

1.1 4

1.2 Error! Bookmark not defined.

1.3 Error! Bookmark not defined.

2. Logging

4

3 Error! Bookmark not defined.

4 Error! Bookmark not defined.

5 6

Abstract:

Due to the busy lifestyle even for the people who loves books, finding the opportunity to read books can be a challenge. Best of all, audiobooks let all the people read anytime and at any place. Audiobooks allowed listeners freedom from physical media such as cassettes, CD, DVD. Basically audiobooks allows multitasking. It can be listened during exercising, cooking, travelling with public transport or crafting and similar activities. They can be a valuable tool to increase reading comprehension, boost self-confidence and also saves time which leads to better outcomes. This work discusses the implementation of the a fully automatic desktop app for personal audio book system, where the main objective is to allow user to upload any book of his choice in pdf format and after uploading the book it convert the pdf text to speech.

1 Introduction

1.1 Why this Low-Level Design Document?

The goal of LLD or a low-level design document (LLDD) is to give the internal logical design of the actual program code for Personal Audiobook Desktop App System. LLD describes the class diagrams with the methods and relations between classes and program specs. It describes the modules so that the programmer can directly code the program from the document. The LLD phase is the stage where the actual software components are designed.

During the detailed phase the logical and functional design is done and the design of application structure is developed during the high-level design phase.

1.2 Scope:

This software system will be a desktop application This system will be designed to the convert the pdf text to speech.

1.3 Constraints:

The Personal Audio Book Desktop app system should be user friendly, simple and user should not be required to know any of the working of the system.

2. Logging

We should be able to log every activity done by the user.

- The System identifies at what step logging required
- The System should be able to log each and every system flow.
- Developers can choose logging methods. You can choose database logging or File logging as well.
- System should not be hung even after using so many loggings. Logging just because we can easily debug issues so logging is mandatory to do.

3. Technology stack:

Front End	Tkinter
Backend	Python
Deployment	AWS

Deployment:

AWS:



4. Proposed Solution:

The solution proposed here is a fully automatic desktop app for personal audio book system can be implemented to perform above mention use cases. In first case, user is allowed to upload any book of his choice in pdf format. After uploading a book, user have to click on the submit button. After clicking submit button the uploaded book which is in pdf format gets converted into speech.

5.User I/O workflow

