Deliverable #2 Template

SE 3A04: Software Design II – Large System Design

1 Introduction

This document will go in depth into the design of Merlin, a song lookup application, using various different analysis methods such as a Use Case Diagram, Analysis Class Diagram, Architectural Design Analysis, as well as Class Responsibility Collaboration (CRC) Cards.

1.1 Purpose

The purpose of this documentation is to state, describe and analyze the architecture of the Merlin song identification app. This will be done through a set of predetermined and proven analysis methods and is intended to be read by the implementation team and client.

1.2 System Description

The application, Merlin, contains a back end system that will be discussed in this report. The system is constructed using multiple expert classes and each class will render a different set of code. A user may input the information about a song using the Tempo, Lyric or using the Artists name. The app will then use these experts to find the song that the user is trying to search for from a database of all songs. The user will have their previous search information stored on the device. This application will run on any android application, and will be available for download through the android app store.

1.3 Overview

This document has 5 sections including this one. Every section analyzes the architecture of the Merlin applications system in a different way using either diagrams or words. Following is a breakdown of what each section analyzes:

1) Use Case Analysis and Diagram: This section will demonstrate all possible ways of the apps interaction with the user.

2) Analysis Class Diagram: This diagram will outline the three different types of classes (boundary, controller or entity) and specify which class falls under which category. This section will also show the different connections between the classes.

3) Architectural Design Class: This section will overview and justify the reason for designing the application in the chosen way.

4) Class Responsibility Collaboration Cards: These cards will show the different classes that make up the song identification app, and their key responsibilities. It will also state which classes each other interact with.

2 Use Case Diagram

View Previous Searches The user must be able to review their previous searches. All searches should be saved so that they may be displayed in this list. The data saved should consist of the inputs entered to return the result.
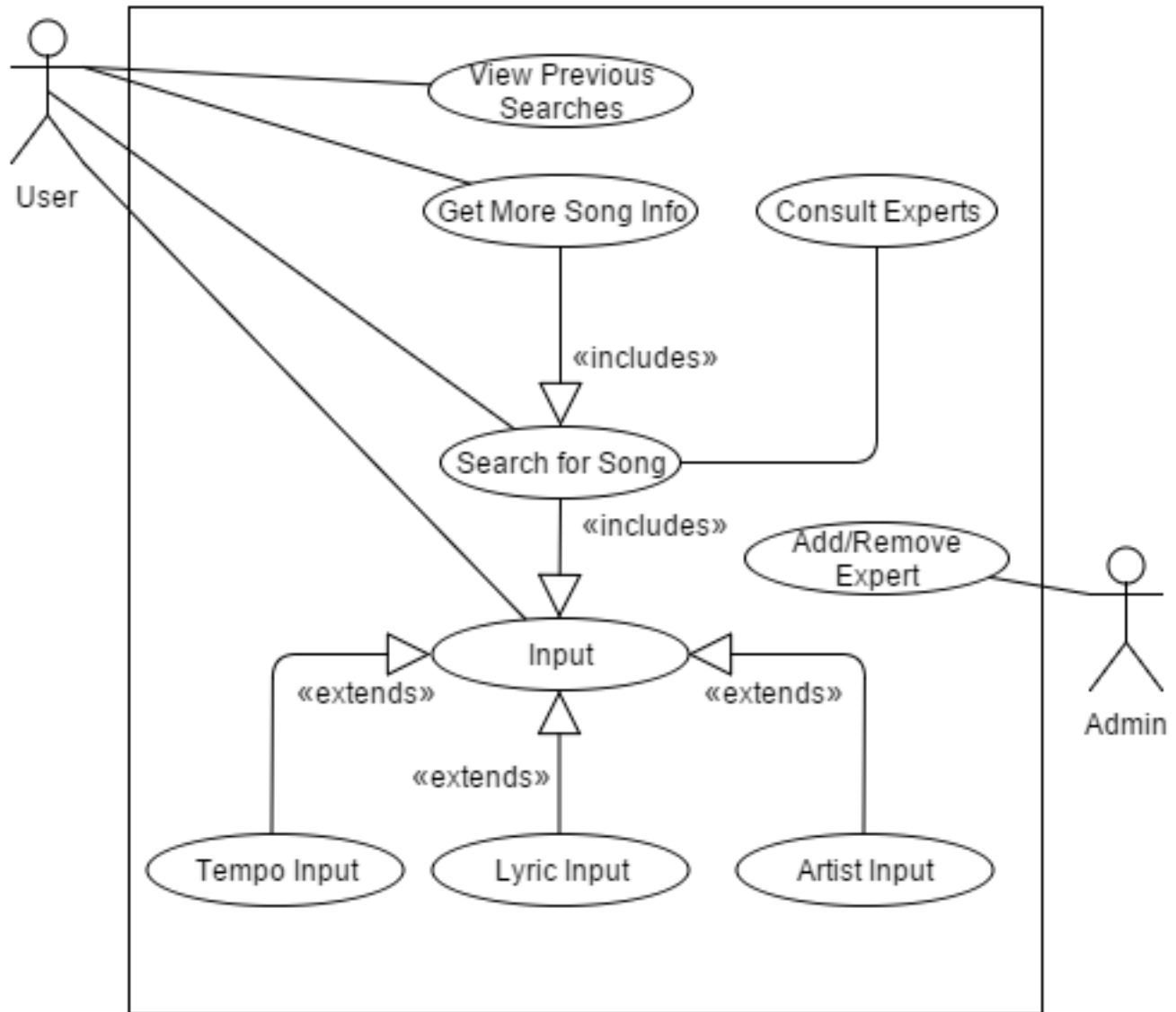
Figure 1: Use Case Diagram

Get More Song Info Each song result is linked to more detailed information about the artist and song. The user should have access to this more detailed information on request. This use case cannot be executed unless a song is searched for first.

Search for Song The user can search for a song based on the inputs given.

Input This use case is a generic use case that is extended by more specific input fields. These fields include Tempo Input, Lyric Input and Artist Input.

Tempo Input To input a tempo the user must tap on the screen at the desired tempo. This tempo is converted to a measurement in BPM for searching.

Lyric Input To input lyrics users will have the option to type some lyrics into the search field or speak into the device's microphone and translate their voice to text via the speech-to-text functionality. These lyrics are a sequence of words that can be found in the desired song.

Artist Input The artist input is the name of the artist that preforms the desired song. To input the artist the user may type the name into the search field or speak into the device's microphone and translate their voice to text via the speech-to-text functionality.

Consult Experts Experts will be consulted to attempt to identify the desired song. They will work together to combine their opinions to decide on a song.

Add/Remove Expert The administrator may wish to take an expert out of the system or introduce an new expert. In order to do so they must contact the developer who will directly preform the switch.

3 Analysis Class Diagram

3.1 Noun Extraction

user No (outside of system)

previous searches Yes (boundary class) PreviousSearchPage

searches refers to search process, leads to the creation of controller class SearchController

success Yes (boundary class) SearchSuccessPage or SearchResults

error Yes (boundary class) SearchErrorPage

song info Yes (entity class) SongInformation

input refers to input process, leads to the creation of controller class InputController

tempo input Yes (boundary class)

lyric input Yes (boundary class)

artist input Yes (boundary class)
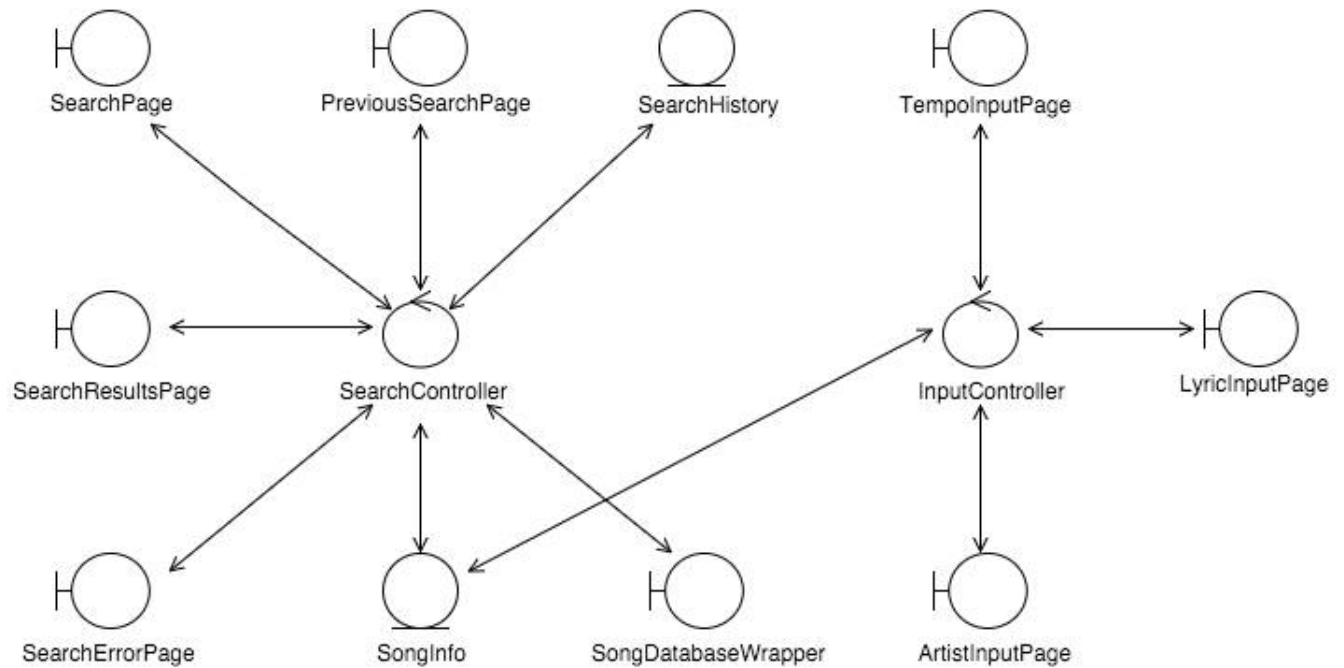
song database Yes (boundary class), Wrapper

Figure 2: Analysis Case Diagram

4 Architectural Design

4.1 System Architecture

a) A blackboard architectural design will be used.

b) The system will primarily use the information obtained by 3 experts to determine a final result. The nature of this structure lends itself to the chosen type architecture.

4.2 Subsystems

a) **Artist Expert:** The expert charged with identifying songs associated with a given artist.

b) **Lyric Expert:** The expert charged with identifying songs associated with a given lyric.

c) **Tempo Expert:** The expert charged with identifying songs associated with a given tempo.

d) **Blackboard:** The subsystem charged with combining data from all other subsystems and producing a final result for the user interface.

e) **User Interface:** The subsystem charged with handling interactions with the user, including input and output.

5 Class Responsibility Collaboration (CRC) Cards

This section should contain all of your CRC cards.

a) Provide a CRC Card for each identified class

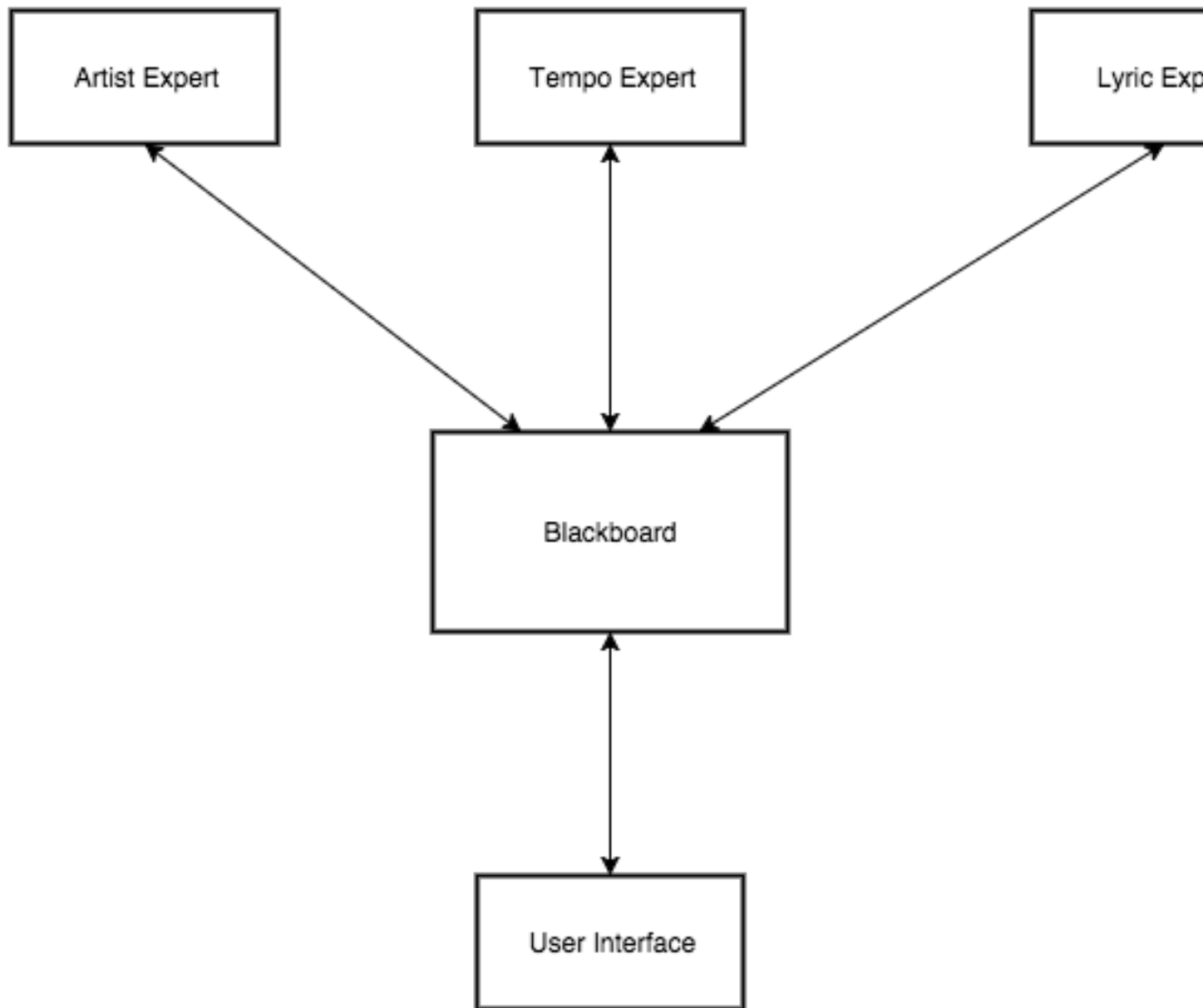b) Please use the format outlined in tutorial, i.e.,

Figure 3: System Architecture Diagram

A   Division of Labour

Include a Division of Labour sheet which indicates the contributions of each team member. This sheet must be signed by all team members.

Class Name:	
Responsibility:	**Collaborators:**

IMPORTANT NOTES

- Please document any non-standard notations that you may have used

  - *Rule of Thumb*: if you feel there is any doubt surrounding the meaning of your notations, document them

- Some diagrams may be difficult to fit into one page

  - It is OK if the text is small but please ensure that it is readable when printed
  - If you need to break a diagram onto multiple pages, please adopt a system of doing so and thoroughly explain how it can be reconnected from one page to the next; if you are unsure about this, please ask about it

- Please submit the latest version of Deliverable 1 with Deliverable 2

  - It does not have to be a freshly printed version; the latest marked version is OK

- If you do <u>NOT</u> have a Division of Labour sheet, your deliverable will <u>NOT</u> be marked