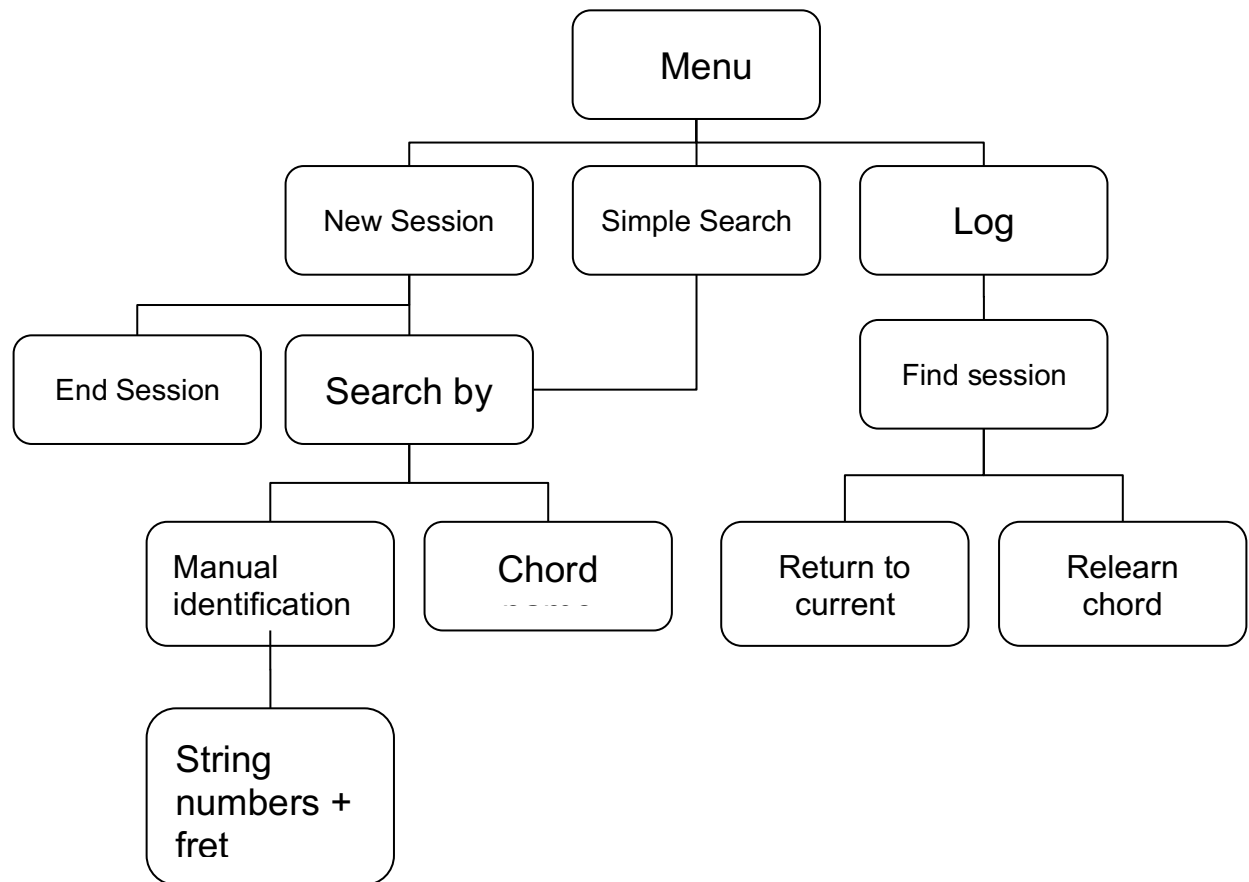


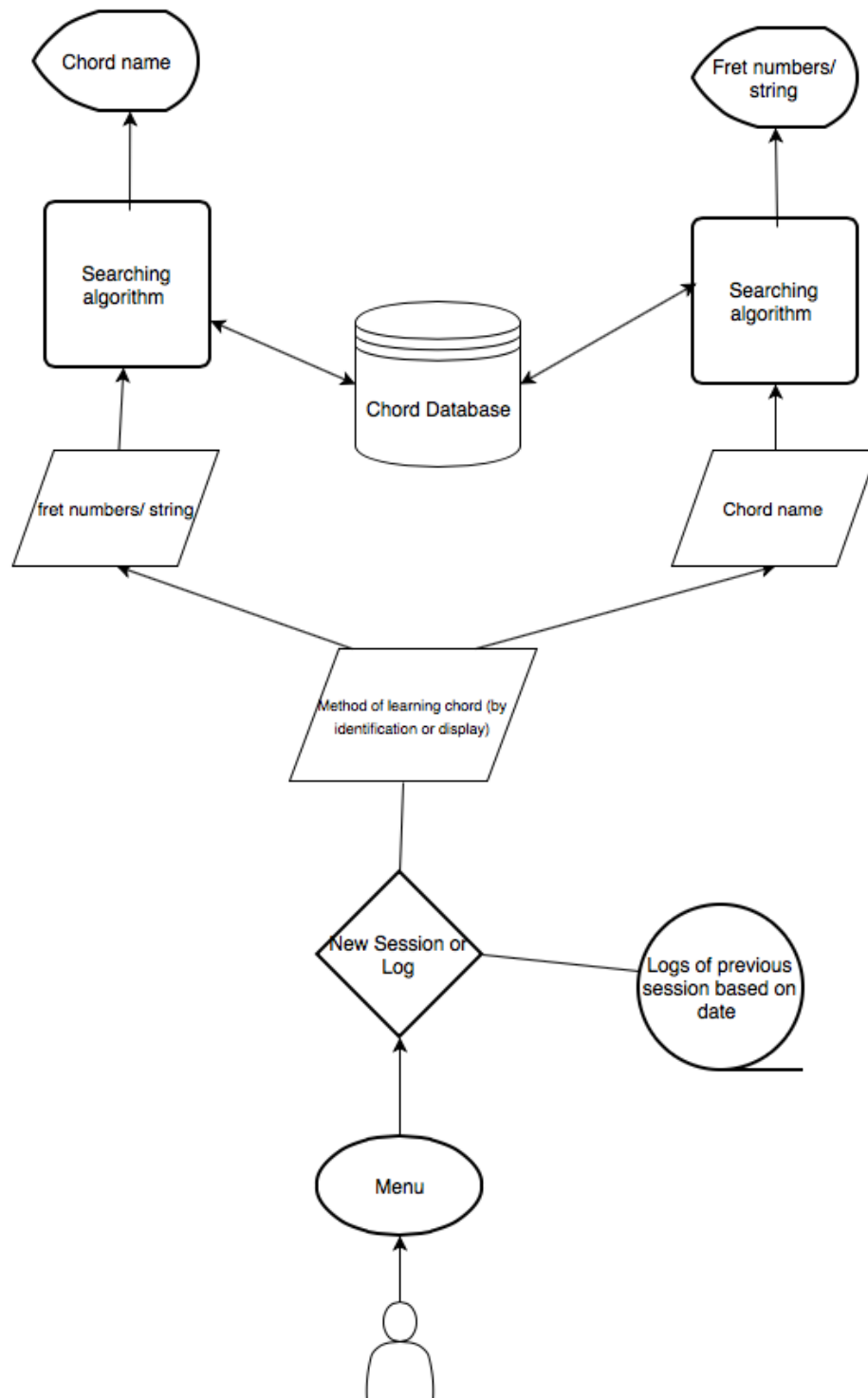
Criterion B: Solution Overview/Design

Designs

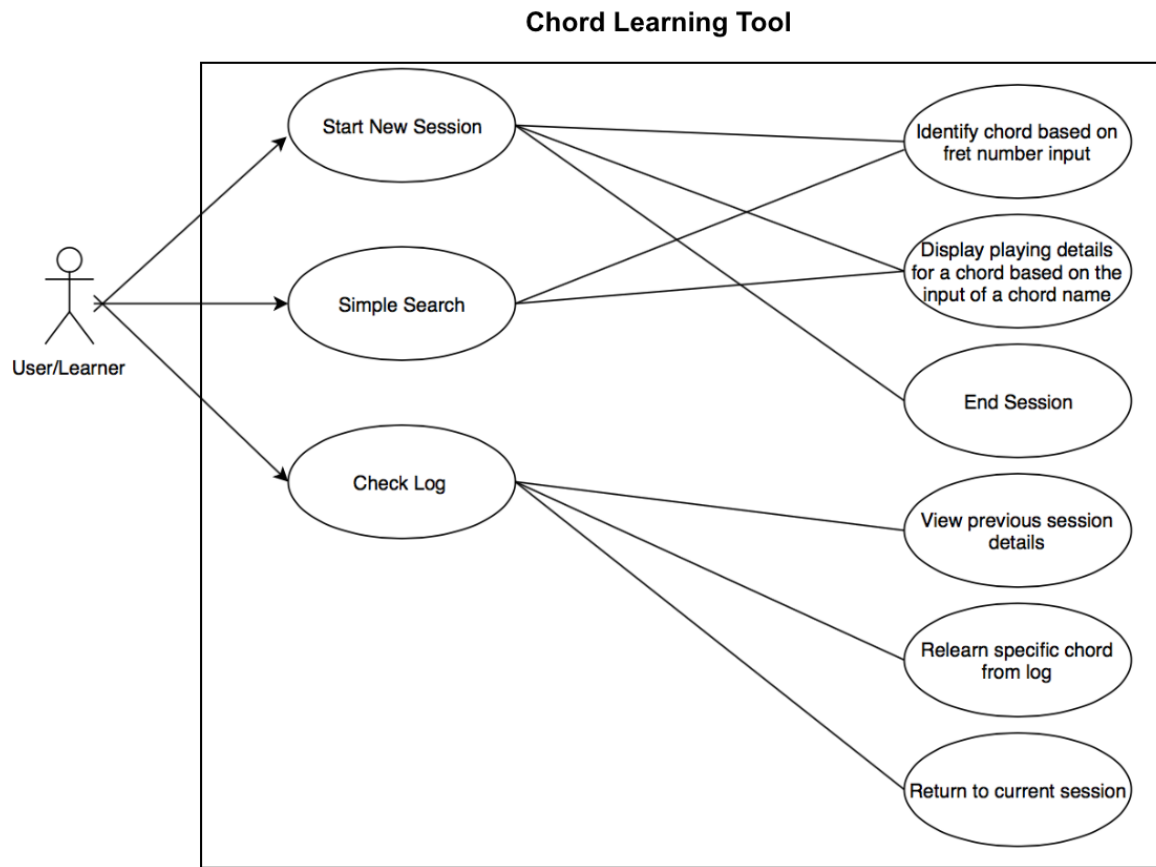
1. **Decomposition Diagram (top-down):** Mapping out the different functions that I will be creating for my solution to be effective is made simpler through the use of this diagram.



2. Systems FlowChart: The systems flowchart is used to categorise different parts of the system in terms of their purpose. For example, the “Chord Database” is identified by a particular symbol. Essentially, this flowchart helps visualise the different parts of the system and their relationships through this syntax of prescribed symbols.



3. Use-case Diagram: This diagram outlines the scope of the proposed solution and provides a basic understanding of the user's interaction with the system. In addition to the textbook <add chapter> by Dimitriou and Hatzitaskos, Fakhroutdinov was also consulted before building the use-case diagram for the solution:



[caption the use-case diagram using MLA formatting, too!]

4. **UML Class Diagrams:** Figure [number] shows a first-draft of the class diagrams I predict I will need to lay the groundwork for my coding. I expect to develop them as the IA process continues.

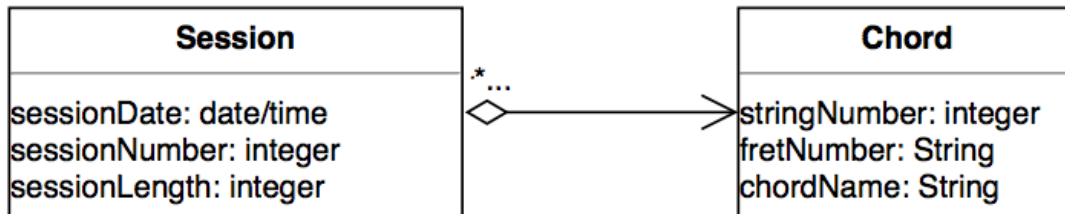


Fig. [number]. Initial UML class diagram.

5. **Data Dictionary:** A glossary of variable names that require user interaction/input and will be used in the development of the system. It is intended to improve user as well as technical documentation of the new system.

Data	Data Type	Description
Session Date	Date/time	The start time and date of the new session; automatically registered in the Session Log
Session Length	Integer	Recorded when Session is “Finished” or exited; automatically generated number of minutes spent in session; displayed when individual session is accessed through Log
Session Number	Integer	Ordered list of sessions over time; automatically recorded for ease of user access in the Log
String Number	String	Guitar string number; auto-display when identifying chord
Fret Number	String	User input fret number in relation to String Number; Must be entered when Identifying Chord; must be 0 or more with the exception of “X” when String is not played at all
Chord Name	String	Mixture of alphabets and numbers; unique

6. File Structures

Two plain text files will be used for permanent storage of data on the client's secondary/backing store drive. They will be read from and written to using sequential access.

chord.txt
String number: integer
Fret number: integer
Chord name: String

session.txt
Session date: date/time
Session number: integer
Session length: integer

Test Plan

Type, length, and range restriction for fret numbers based on the capabilities of the acoustic six-string guitar the client uses.

Data Type	Type of validation	Comment	Input Data	Expected Output	Test Pass/Fail
Normal	- Length check - Range check - Presence check	Correct type (String), within limits ($0 < x < 21$) OR "X"	4, 15	Proceeds with next input/ displays required data	✓
Abnormal		Outside limits, wrong type	67, -1, <null>, b	Error message and input is asked for again	✓
Extreme		On the boundaries of the range	0, 20	Allowed	✓

Type check for Chord Name, which is based on user input for displaying chord patterns.

Data Type	Type of validation	Comment	Input Data	Expected Output	Test Pass/Fail
Normal	• String type check • Presence check	Correct type	A maj, Amaj, Am, Bm7, C sus4	Allowed; program continues to run	✓
Abnormal		Wrong type, no entry	67, ?, hello, <null>	Error message and input is asked for again	✓

8. Pseudocode/ Flowchart

The two main algorithms planned for the solution are the selection sort to meet success criterion [N] and linear search to meet success criterion [M].

Below is the pseudocode listing for the selection sort:

```
ELEMENTS = [1, 5, 3, 86, 256, 420, 9, 510, 51, 24, 60]
MIN = 0
I = 0
TEMP = 0

loop MIN from 0 to 9
  I = MIN
  loop CURRENT from MIN+1 to 10
    I = CURRENT
  end if
end loop
TEMP = ELEMENTS [I]
ELEMENTS [I] = ELEMENTS [MIN]
ELEMENTS [MIN] = TEMP
end loop

output "SORTED ARRAY"
loop C from 0 to 10
  output ELEMENTS [C]
end loop
```

←

←

[modify this pseudocode to make it look more like something to be used in your IA (change variable names, change 9 for LENGTH (length of data structure), etc.) or use a more generic version like the one from the sorting algorithms SL handout

←

←

(Dimitriou and Hatzitaskos) ← use Fig. N. and MLA style of in-text referencing for both items here

The flow chart shown in figure [number] represents the sorting algorithm to be used in the solution.

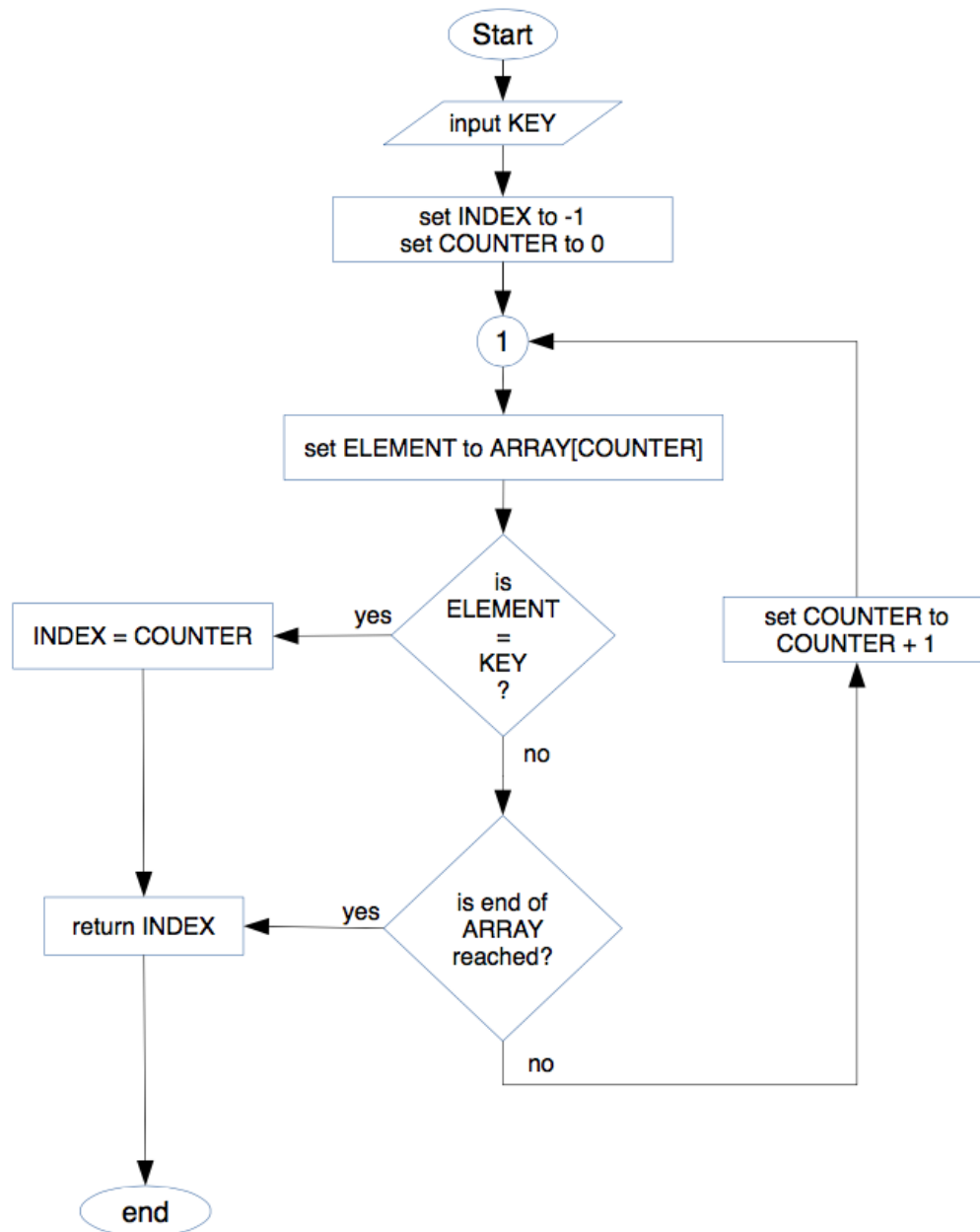



Fig. [number]. Sequential search flowchart (Dimitriou and Hatzitaskos, chapter?) 
double check this source.

WORKS CITED

Dimitriou, Kostas and Hatzitaskos, Markos. *Core Computer Science: for the IB Diploma Program (International Baccalaureate)*. Express Publishing, 2015.

Fakhroutdinov, Kirill. "UML Use Case Diagrams." *UML Graphical Notation Overview, Examples, and Reference.*, 4 January 2014, www.uml-diagrams.org/use-case-diagrams.html.