NORTHWESTERN POLYTECHNIC
UNIVERSITY

UML Diagrams(JukeBox)

Prepared For:

Mr. Henry Chang
Software Quality Assurance and Test Automation CS522
Fall 2021
Northwestern Polytechnic University

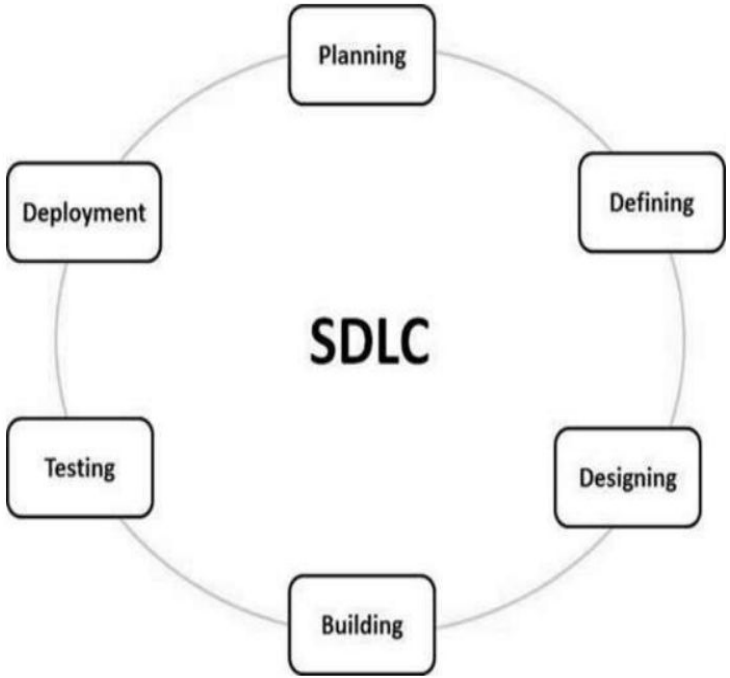
Prepared By:

Ms. Nagalla, Santhi Sree ID:19568

Table of Contents

Software Development Life Cycle (SDLC)

- SDLC is a process followed for a software project, within a software organization. It consists of a detailed plan describing how to develop, maintain, replace and alter or enhance specific software. The life cycle defines a methodology for improving the quality of software and the overall development process.



UML(Unified Modeling Language)

- Unified Modeling Language (UML) is a general purpose modelling language. The main aim of UML is to define a standard way to visualize the way a system has been designed.
- UML is linked with object oriented design and analysis. UML makes the use of elements and forms associations between them to form diagrams.

UML(Unified Modeling Language)

Diagrams in UML can be broadly classified as:

Structural Diagrams – Capture static aspects or structure of a system. Structural Diagrams include: Component Diagrams, Object Diagrams, Class Diagrams and Deployment Diagrams.

Behavior Diagrams – Capture dynamic aspects or behavior of the system. Behavior diagrams include: Use Case Diagrams, State Diagrams, Activity Diagrams and Interaction Diagrams.

Use Case Diagram

- Use Case Diagrams are used to depict the functionality of a system or a part of a system. They are widely used to illustrate the functional requirements of the system and its interaction with external agents(actors). A use case is basically a diagram representing different scenarios where the system can be used. A use case diagram gives us a high level view of what the system or a part of the system does without going into implementation details.

Class Diagram

- The most widely use UML diagram is the class diagram. It is the building block of all object oriented software systems. We use class diagrams to depict the static structure of a system by showing system's classes,their methods and attributes. Class diagrams also help us identify relationship between different classes or objects.

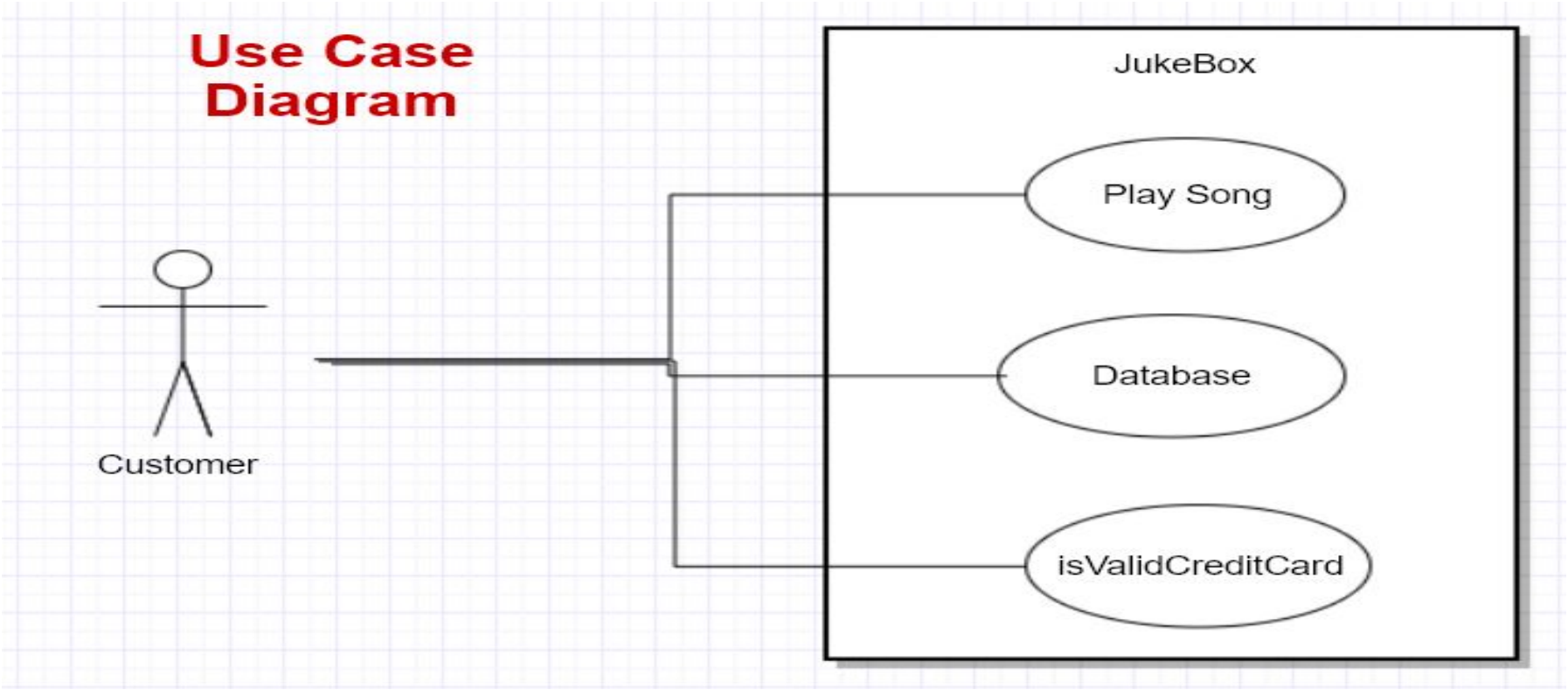
Sequence Diagram

- A sequence diagram simply depicts interaction between objects in a sequential order i.e. the order in which these interactions take place. Sequence diagrams describe how and in what order the objects in a system function. These diagrams are widely used by businessmen and software developers to document and understand requirements for new and existing systems.

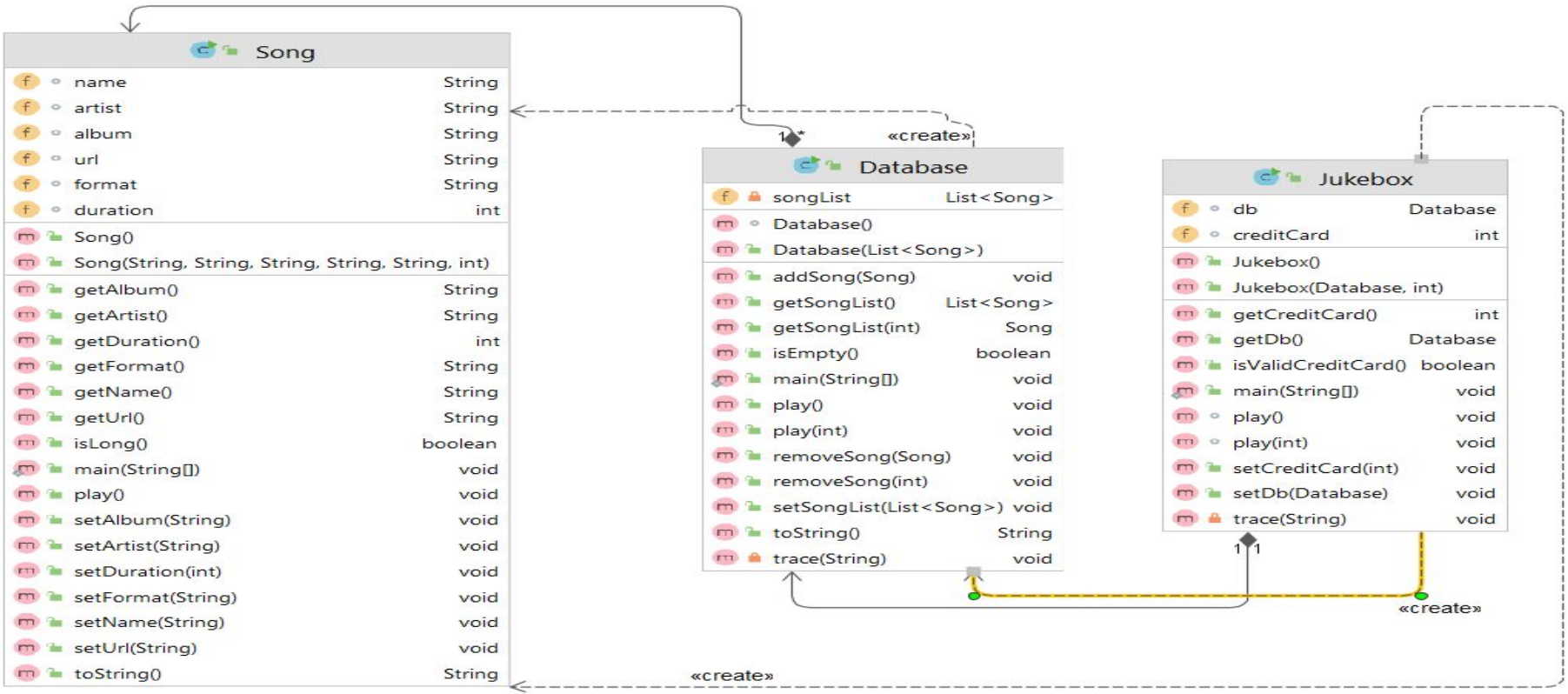
JukeBox

- Juke Box spec
  - Allow customers to
    - select songs they want to play.
    - submit a playlist that they have already created previously.
  - The Juke Box can search other Juke Boxes from Internet for songs that are not contained by a local Jukebox.
  - To provide a mechanism for owners, record companies and artists to earn a profit. The Juke Box contains
    - A coin drop
    - Cash feed mechanism
    - A card swipe mechanism
    - A cell dial payment capability.

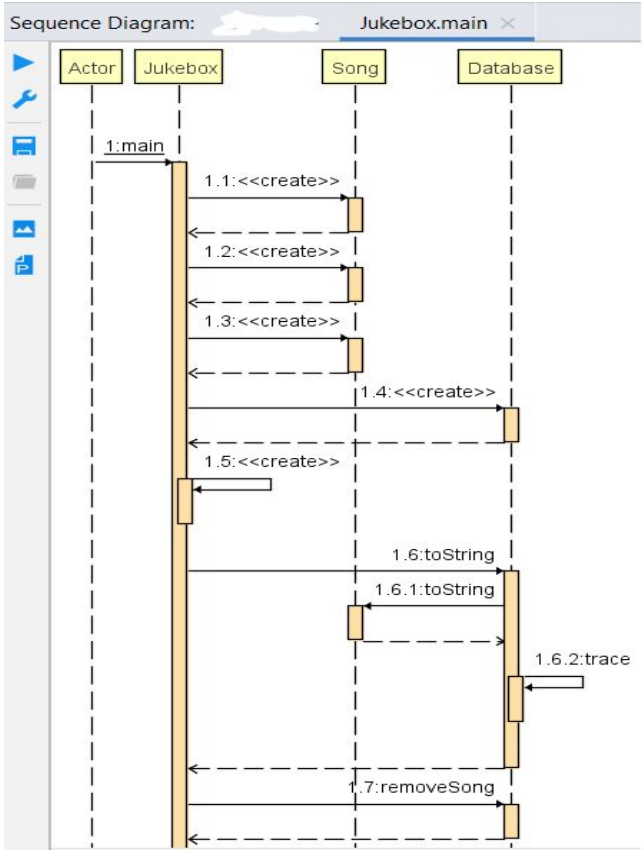
JukeBox - Use Case Diagram

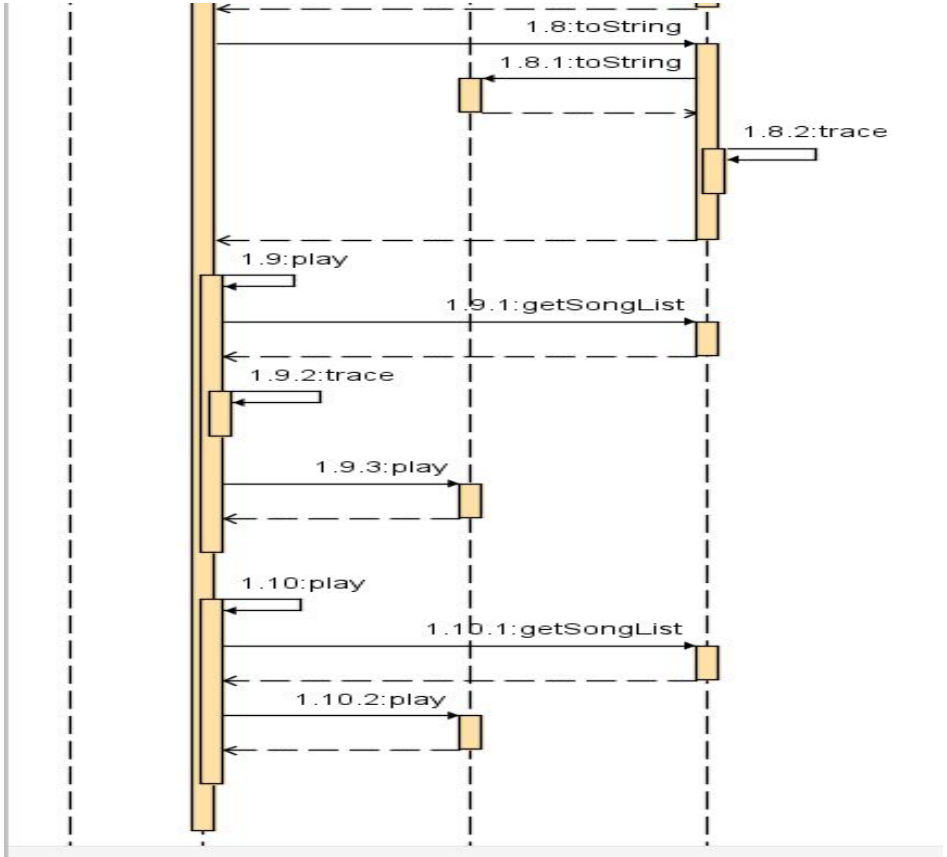


JukeBox - Class Diagram



JukeBox - Sequence Diagram





References

- GitHub URL - https://github.com/santhinagalla/Software-Quality-Assurance-and-Test-Automation/tree/main/Software%20Development%20Life%20Cycle%20(SDLC)/UML/JukeBox
- https://www.geeksforgeeks.org/unified-modeling-language-uml-introduction/
- https://npu85.npu.edu/~henry/npu/classes/oo/uml/slide/use_case_diagram.html
- https://npu85.npu.edu/~henry/npu/classes/introjava/java_class/hw/christy_jaeson_augustine/jukebox/index_jukebox.html
- https://npu85.npu.edu/~henry/npu/classes/oo/uml/slide/class_diagram.html
- https://npu85.npu.edu/~henry/npu/classes/oo/uml/slide/sequence_diagram.html