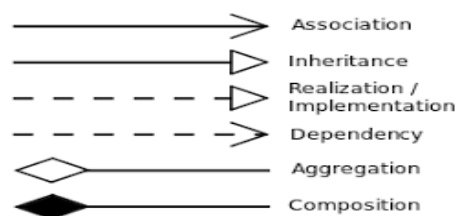


# Design Documentation

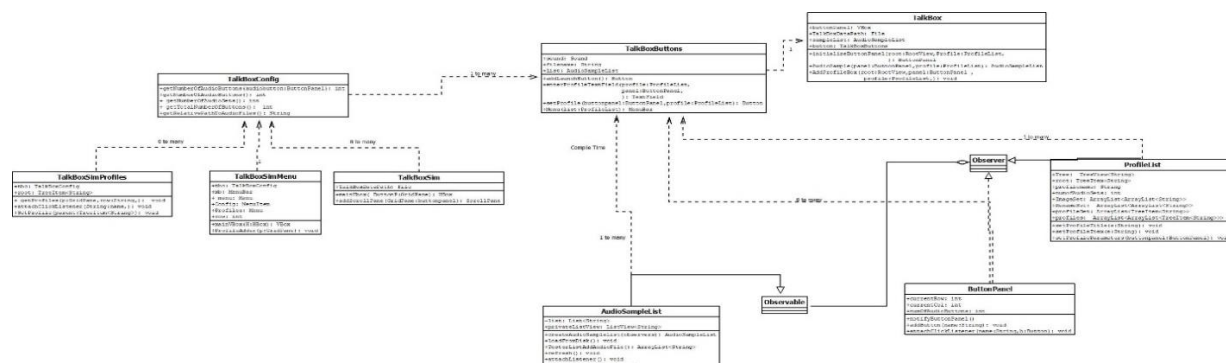
## Introduction and Purpose

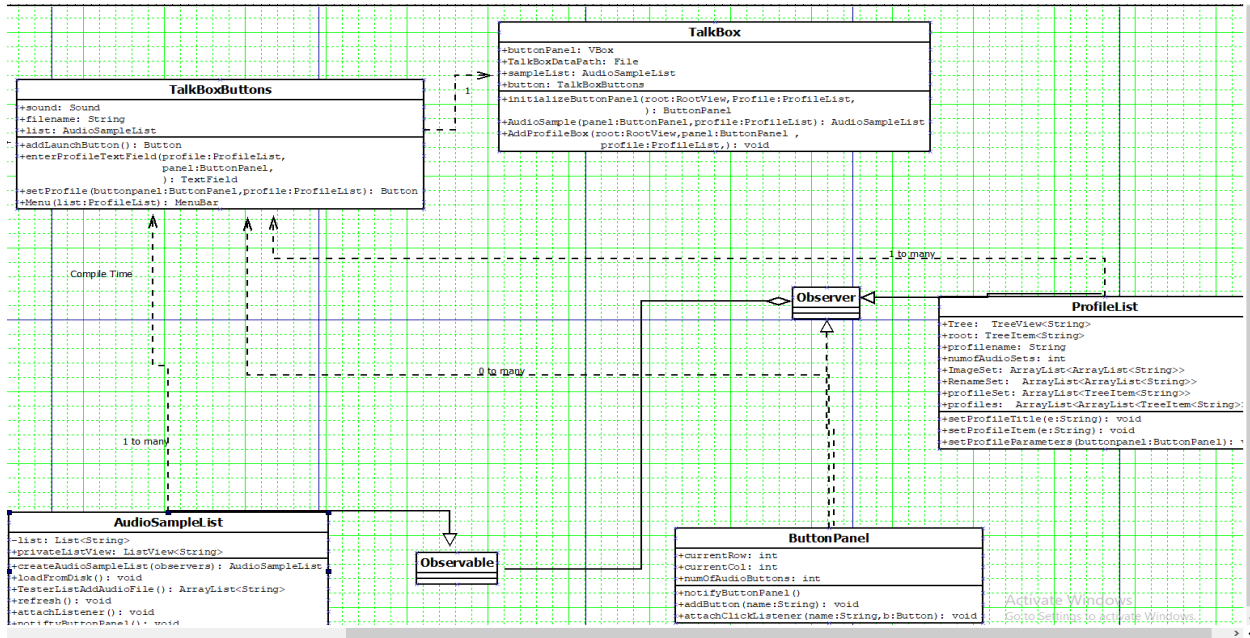
This document describes the design process of the Talk Box Application. The application is delivered into three systems: configurator, simulator and Logger. The device is implemented for users that have speech impairment issues and using this device will provide a way of communicating with friends and family. The class diagram is broken into three phases correlating to the different applications. This document displays multitude class diagrams, sequence diagrams. The class diagrams are not displaying all the classes but rather the important classes, fields and methods. This makes it so a developer can understand the important schemes of the design implementation making the application. The sequence diagram elucidates the interaction between the user and the multiple systems interacting with each other. As a design pattern, the TalkBox application follows the MVC(Model-View-Controller) design and Observer pattern.



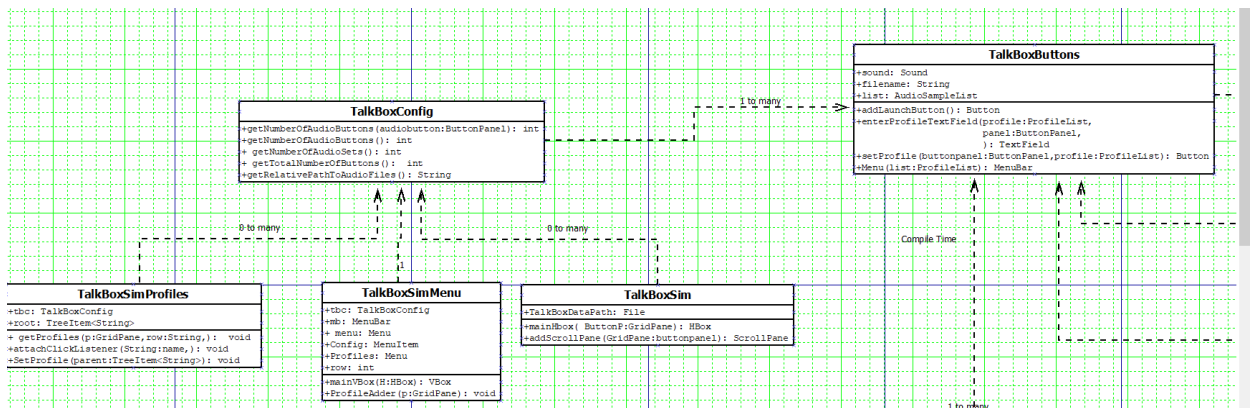
UML Class Diagram Legend

The below image is the overall design implementation of the TalkBox.



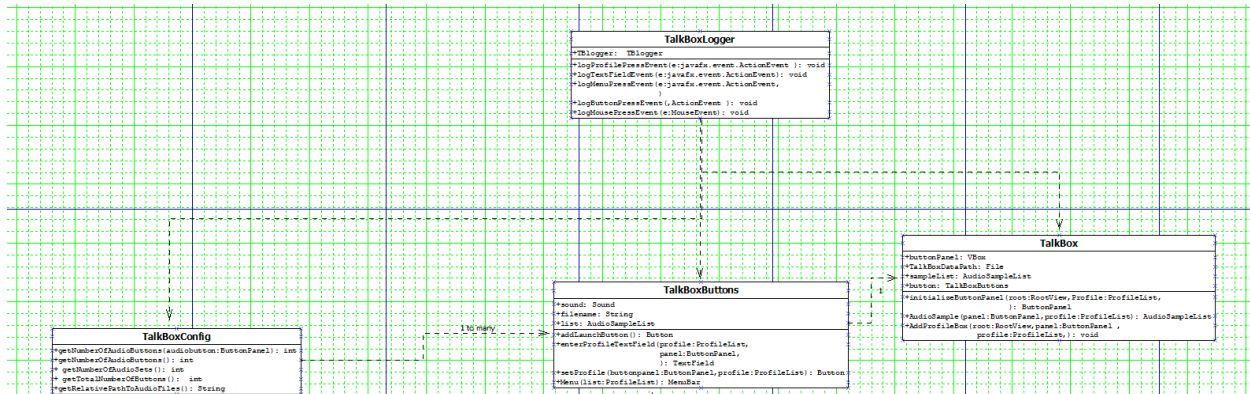


This class diagram shows how the observer pattern is being utilized. So, when the user clicks an imported audio file which is found in the AudioSampleList, it is the Observable, and sends a signal to the Observer class which is extended from the ButtonPanel class and creates a button on the main view found in the \_TalkBox class.



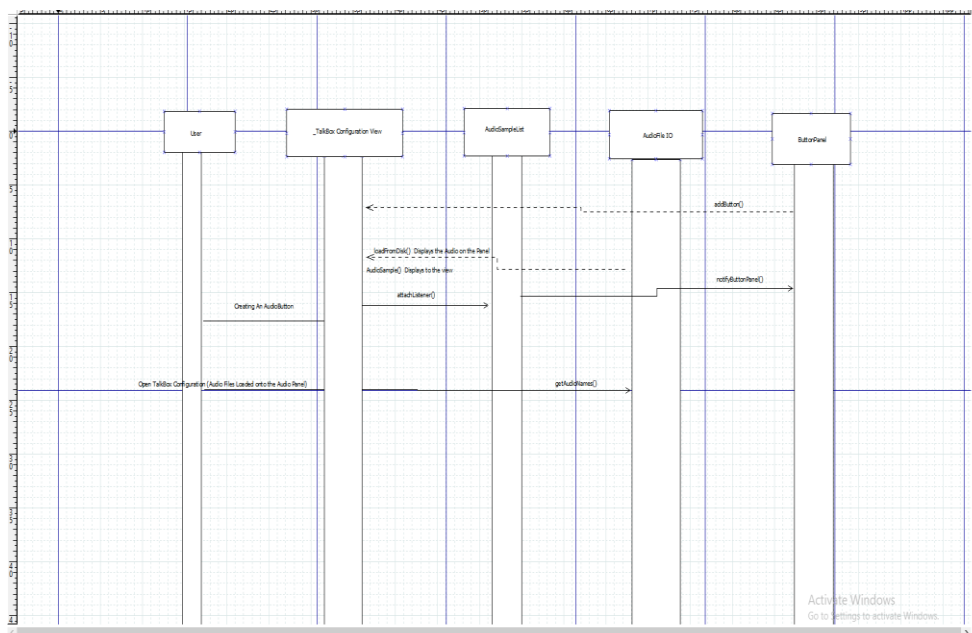
This class diagram shows the relationship between the TalkBox configurator in relation to the Simulator

class and dependency otherwise generalization the classes have on the serialized information contained in TalkBox Configuration class.

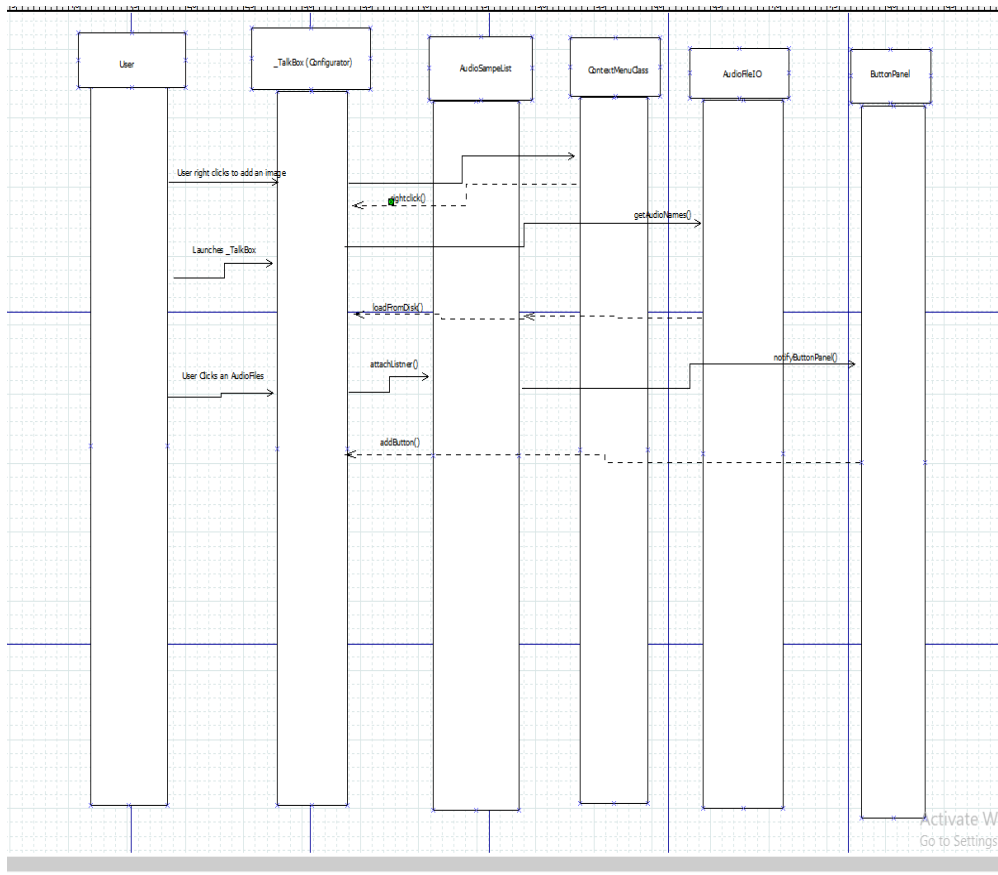


This class diagram shows the new implementation of the TalkBox Logger where it is found in multiple methods used to keep track of the buttons clicked or audio sets created.

## Sequence Diagram



This sequence diagram shows how the user opens the TalkBox application and then clicks on an audio file and subsequently an audio button is created.



This sequence diagram shows the previous action but adding the additional function where the user right-clicks an existing button and has 2 options: load images and rename.

