

**Software Engineering**  
**Modeling and Implementation Project**

---

**“Simulator of a Cassette Deck”**

**- Diagrams -**

---

By Constant Théo and Essafsyfy Abdelkrim

Academic year 2018-2019

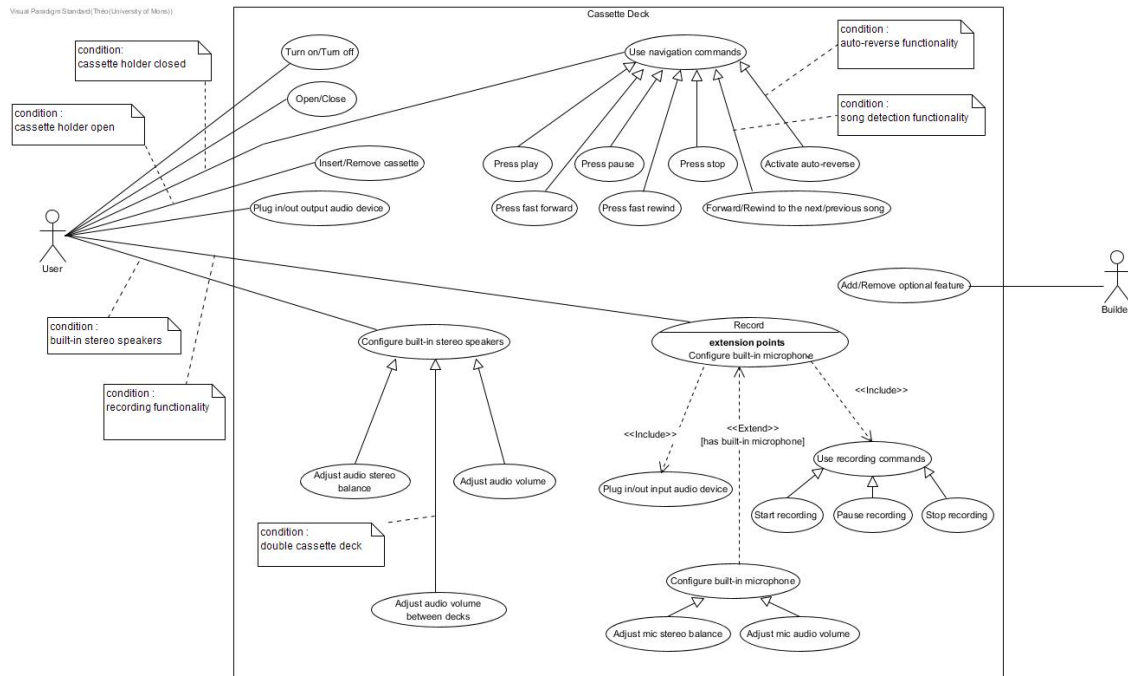
## Contents

<b>1</b>	<b>Use case diagram</b>	<b>1</b>
1.1	Diagram . . . . .	1
1.2	Cases' description . . . . .	1
1.2.1	Turn on/Turn off . . . . .	1
1.2.2	Open/Close . . . . .	2
1.2.3	Insert/Remove cassette . . . . .	2
1.2.4	Plug in/out audio device . . . . .	3
1.2.5	Use navigation commands . . . . .	3
1.2.6	Press play . . . . .	3
1.2.7	Press pause . . . . .	4
1.2.8	Press stop . . . . .	4
1.2.9	Press fast forward . . . . .	4
1.2.10	Press fast rewind . . . . .	5
1.2.11	Forward/Rewind to the next/previous song . . . . .	5
1.2.12	Activate auto-reverse . . . . .	6
1.2.13	Configure built-in stereo speakers . . . . .	6
1.2.14	Adjust audio stereo balance . . . . .	6
1.2.15	Adjust audio volume . . . . .	7
1.2.16	Adjust audio volume between decks . . . . .	7
1.2.17	Record . . . . .	8
1.2.18	Plug in/out input audio device . . . . .	8
1.2.19	Configure built-in microphone . . . . .	8
1.2.20	Adjust mic stereo balance . . . . .	9
1.2.21	Adjust mic audio volume . . . . .	9
1.2.22	Adjust mic audio volume . . . . .	9
1.2.23	Use recording commands . . . . .	10
1.2.24	Start recording . . . . .	10
1.2.25	Pause recording . . . . .	10
1.2.26	Stop recording . . . . .	11
<b>2</b>	<b>Class diagram</b>	<b>12</b>

<b>3</b>	<b>State Charts Diagrams (<i>Yakindu</i>)</b>	<b>13</b>
3.1	Diagram 1 . . . . .	13
3.2	Diagram 2 . . . . .	13
3.3	Diagram 3 . . . . .	14
<b>4</b>	<b>Sequence diagrams</b>	<b>15</b>
4.1	Music Player . . . . .	15
4.2	Music Recorder . . . . .	16
<b>5</b>	<b>Activity diagram</b>	<b>17</b>

# 1 Use case diagram

## 1.1 Diagram



## 1.2 Cases' description

### 1.2.1 Turn on/Turn off

**Use case name :** Turn on/Turn off

**Summary :** Allows the user to turn the cassette deck on or off.

**Actors :** User.

**Assumptions :** Cassette deck has access to a power source.

**Preconditions :** Cassette holder is closed

**Basic course of action :**

1. User presses power button on.
2. Cassette deck is powered on.

**Postconditions :** Same as preconditions.

**Alternate courses :**

2a : Cassette deck is powered off.

### 1.2.2 Open/Close

**Use case name :** Open/Close

**Summary :** Allows the user to open or closes the cassette holder.

**Actors :** User.

**Assumptions :** Cassette deck is plugged in.

**Preconditions :** Cassette deck is powered on.

**Basic course of action :**

1. User presses eject button.
2. Cassette holder opens.

**Postconditions :** Same as preconditions.

**Alternate courses :**

2b : Cassette holder closes.

### 1.2.3 Insert/Remove cassette

**Use case name :** Insert/Remove cassette

**Summary :** Allows the user to insert or remove a cassette from the cassette holder

**Actors :** User.

**Assumptions :** Cassette deck is powered on.

**Preconditions :** Cassette holder is open.

**Basic course of action :**

1. User presses eject button.
2. Cassette holder opens.
3. User inserts a cassette into the cassette holder.

**Postconditions :** Cassette holder is closed

**Alternate courses :**

3a : User removes a cassette from the cassette holder.

#### 1.2.4 Plug in/out audio device

**Use case name :** Plug in/out audio device

**Summary :** Allows user to plug in or out an external audio device

**Actors :** User.

**Basic course of action :**

1. User plugs in an external audio device.

**Alternate courses :**

1a : User plugs out an external audio device.

#### 1.2.5 Use navigation commands

**Use case name :** Use navigation commands

**Summary :** An abstract use case containing common sub classes descriptions.

**Actors :** User

**Assumptions :** Cassette deck is powered on.

**Preconditions :** Cassette inserted into the deck.

#### 1.2.6 Press play

**Use case name :** Press play

**Summary :** Allows the user to press play button to play music.

**Actors :** Same as parent use-case.

**Assumptions :** Same as parent use-case

**Preconditions :** Magnetic head engaged and cassette deck must be in idle state

**Basic course of action :**

1. User presses play button.

2. Magnetic head engages.

3. Music starts playing.

**Postconditions :** Magnetic head disengaged

### 1.2.7 Press pause

**Use case name :** Press pause

**Summary :** Allows the user to press pause button to pause music.

**Actors :** Same as parent use-case.

**Assumptions :** Same as parent use-case.

**Preconditions :** Same as parent use-case and Magnetic head engaged.

**Basic course of action :**

1. User presses pause button.
2. Music pauses playing.

**Postconditions :** Magnetic head disengaged

### 1.2.8 Press stop

**Use case name :** Press stop

**Summary :** Allows the user to press stop button to stop music.

**Actors :** Same as parent use-case.

**Assumptions :** Same as parent use-case

**Preconditions :** Same as parent use-case.

**Basic course of action :**

1. User presses play button.
2. Magnetic head disengages.
3. Music stops playing.

### 1.2.9 Press fast forward

**Use case name :** Press fast forward

**Summary :** Allows the user to press fast forward button

**Actors :** Same as parent use-case.

**Assumptions :** Same as parent use-case.

**Preconditions :** Same as parent use-case, magnetic head disengaged and cassette deck must be in idle state

**Basic course of action :**

1. User presses fast forward button.
2. Magnetic head disengages.
3. Music is fast forwarded.

**Postconditions :** Magnetic head disengaged

**1.2.10 Press fast rewind**

**Use case name :** Press fast rewind

**Summary :** Allows the user to press fast rewind button.

**Actors :** Same as parent use-case.

**Assumptions :** Same as parent use-case.

**Preconditions :** Same as parent use-case, magnetic head disengaged and cassette deck must be in idle state

**Basic course of action :**

1. User presses fast rewind button.
2. Magnetic head disengages.
3. Music is fast re-winded.

**Postconditions :** Magnetic head disengaged

**1.2.11 Forward/Rewind to the next/previous song**

**Use case name :** Forward/Rewind to the next/previous song

**Summary :**

**Actors :** Same as parent use-case.

**Assumptions :** Same as parent use-case.

**Preconditions :** Same as parent use-case, magnetic head disengaged, and cassette deck must be in idle state

**Basic course of action :**

1. User presses forward to next song.
  2. Deck fast forwards to the next song.
  3. Return to idle state.
- Postconditions :**

**Alternate courses :**



- 1a. User presses rewind to previous song.
- 2a. Deck fast rewinds to the previous song.

### 1.2.12 Activate auto-reverse

**Use case name :** Activate auto-reverse

**Summary :** Allows the user to press auto-reverse button.

**Actors :** Same as parent use-case.

**Assumptions :** Same as parent use-case.

**Preconditions :** Same as parent use-case and deck has an auto-reverse functionality

**Basic course of action :**

- 1. User presses auto-reverse button.
- 2. Magnetic head reaches the end of the tape.
- 3. Magnetic head on one side disengages.
- 4. Magnetic head on the other engages.

### 1.2.13 Configure built-in stereo speakers

**Use case name :** Configure built-in stereo speakers

**Summary :** An abstract use case containing common sub classes descriptions.

**Actors :** User

**Assumptions :** Cassette deck is powered on.

**Preconditions :** Built-in stereo speaker

### 1.2.14 Adjust audio stereo balance

**Use case name :** Adjust audio stereo balance

**Summary :** Allows the user to adjust the audio stereo balance of the deck

**Actors :** Same as parent use-case.

**Assumptions :** Same as parent use-case.

**Preconditions :** Same as parent use-case.

**Basic course of action :**

1. User turns audio stereo balance knob
2. Audio stereo balance increases

**Alternate courses :**

2a : Audio stereo balance decreases

**1.2.15 Adjust audio volume**

**Use case name :** Adjust audio volume

**Summary :** Allows the user to adjust the audio volume of the deck

**Actors :** Same as parent use-case.

**Assumptions :** Same as parent use-case.

**Preconditions :** Same as parent use-case.

**Basic course of action :**

1. User turns audio volume balance knob
2. Audio volume increases

**Alternate courses :**

2a : Audio volume decreases

**1.2.16 Adjust audio volume between decks**

**Use case name :** Adjust audio volume between decks

**Summary :** Allows the user to adjust the audio volume of two decks

**Actors :** Same as parent use-case.

**Assumptions :** Same as parent use-case.

**Preconditions :** Same as parent use-case and double cassette deck

**Basic course of action :**

1. User turns audio volume balance knob
2. Audio volume increases

**Alternate courses :**

2a : Audio volume decreases

**1.2.17 Record**

**Use case name :** Record

**Summary :** Allows the user to record audio using a microphone

**Actors :** User

**Assumptions :** Cassette deck is plugged in.

**Preconditions :** Recording functionality available

**Basic course of action :**

1. User presses the record button.
2. Deck starts recording audio.

**Postconditions :** Same as preconditions.

**1.2.18 Plug in/out input audio device**

**Use case name :** Plug in/out input audio device

**Summary :** Allows the user to plug in or out an input audio device to the deck

**Actors :** User

**Assumptions :** Cassette deck in turned on.

**Preconditions :**

**Basic course of action :**

1. User plugs in an in input audio device.

**Postconditions :** Same as preconditions.

**Alternate courses :**

1. User plugs out an in input audio device.

**1.2.19 Configure built-in microphone**

**Use case name :** Configure built-in microphone

**Summary :** An abstract use case that allows the user to configure a built-in microphone

**Actors :** User

**Assumptions :** Cassette deck in turned on.

**Preconditions :** Built-in microphone

**Basic course of action :**

1. User configures a built-in microphone.

**Postconditions :** Same as preconditions.

**1.2.20 Adjust mic stereo balance**

**Use case name :** Adjust mic stereo balance

**Summary :** Allows the user to adjust deck's mic stereo balance

**Actors :** Same as parent.

**Assumptions :** Same as parent.

**Preconditions :** Same as parent.

**Basic course of action :**

1. User adjust mic stereo balance.

**Postconditions :** Same as parent.

**1.2.21 Adjust mic audio volume**

**Use case name :** Adjust mic audio volume

**Summary :** Allows the user to adjust deck's audio volume

**Actors :** Same as parent.

**Assumptions :** Same as parent.

**Preconditions :** Same as parent.

**Basic course of action :**

1. User adjust audio volume.

**Postconditions :** Same as parent.

**1.2.22 Adjust mic audio volume**

**Use case name :** Adjust mic audio volume

**Summary :** Allows the user to adjust deck's audio volume

**Actors :** Same as parent.

**Assumptions :** Same as parent.

**Preconditions :** Same as parent.

**Basic course of action :**

1. User adjust audio volume.

**Postconditions :** Same as parent.

### 1.2.23 Use recording commands

**Use case name :** Use recording commands

**Summary :** An abstract use case that allows the user to use recording commands

**Actors :** User

**Assumptions :** Cassette deck in turned on.

**Preconditions :** Built-in microphone

**Postconditions :** Same as preconditions.

### 1.2.24 Start recording

**Use case name :** Start recording

**Summary :** Allows the user to start recording audio

**Actors :** Same as parent.

**Assumptions :** Same as parent.

**Preconditions :** Same as parent.

**Basic course of action :**

1. User presses the start recording button.
2. Deck starts recording audio.

**Postconditions :** Same as parent.

### 1.2.25 Pause recording

**Use case name :** Pause recording

**Summary :** Allows the user to pause recording audio

**Actors :** Same as parent.

**Assumptions :** Same as parent.

**Preconditions :** Same as parent.

**Basic course of action :**

1. User presses the pause recording button.
2. Deck pauses recording audio.

**Postconditions :** Same as parent.

### 1.2.26 Stop recording

**Use case name :** Stop recording

**Summary :** Allows the user to stop recording audio

**Actors :** Same as parent.

**Assumptions :** Same as parent.

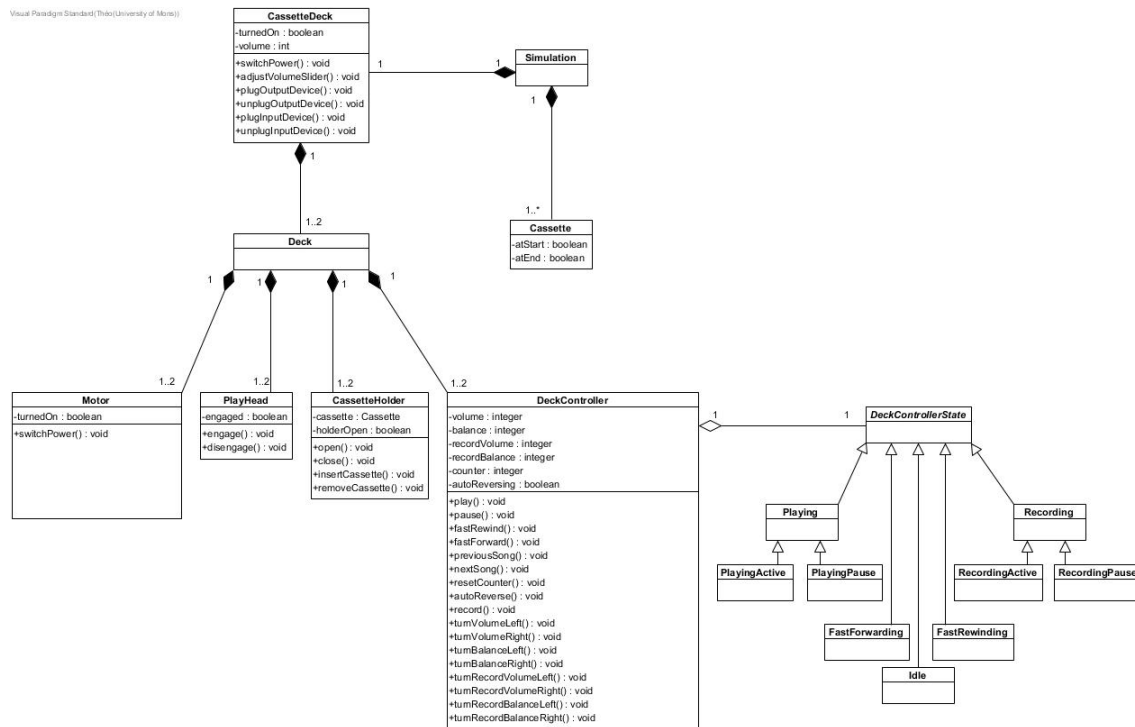
**Preconditions :** Same as parent.

**Basic course of action :**

1. User presses the stop recording button.
2. Deck stops recording audio.

**Postconditions :** Same as parent.

## 2 Class diagram



This diagram shows the different classes we'll use in the implementation. First, we can see a Simulation class : it corresponds to a simulation session and contains one CassetteDeck object and one or several Cassette objects. The Cassette class allows us to instantiate cassettes, with possibly an audio file in them.

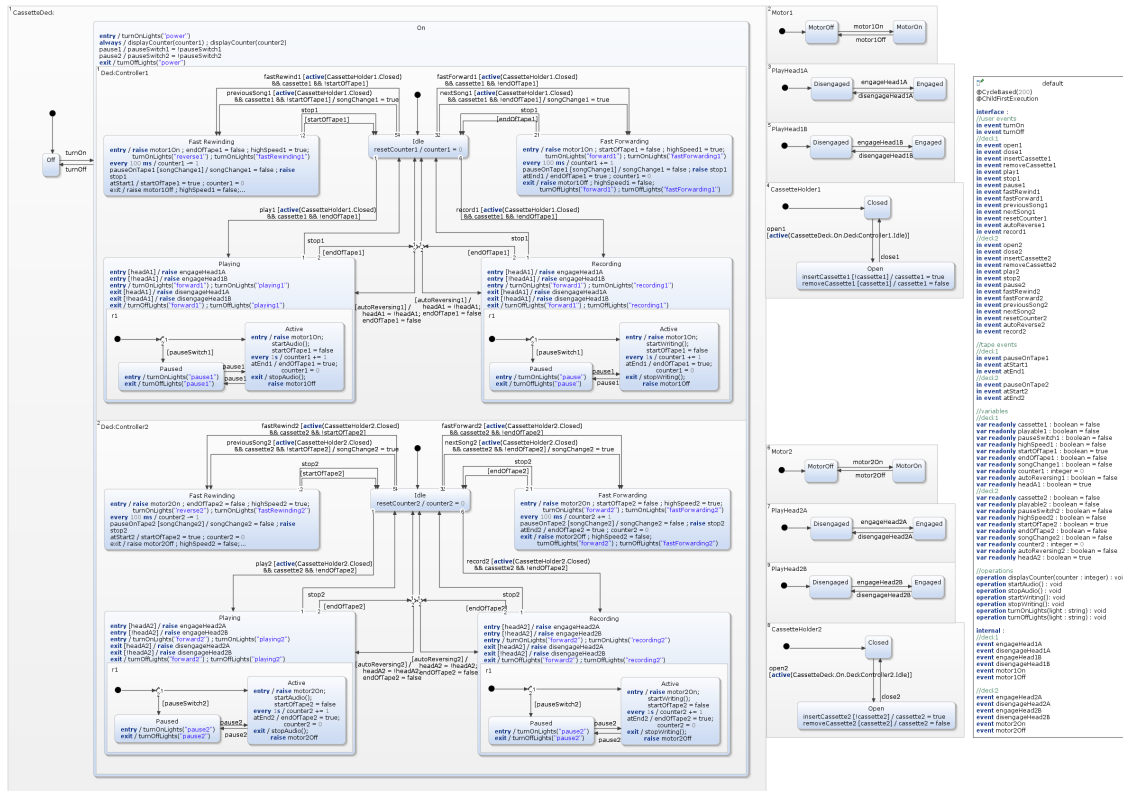
The CassetteDeck class is the device itself. It handles the general buttons like the power button or the volume slider. It contains one or two Deck objects, according to the type of device (some are double cassette decks). Each Deck contains a Motor, a CassetteHolder and a DeckController object, and one or two PlayHead objects (two if it has an auto-reverse functionality).

The DeckController class is the most important class for the user : it contains almost all the commands he can perform, like play, stop or fastRewind. This object can have different states. Therefore, we have decided to implement the state design pattern for it. The class contains an abstract DeckControllerState class, whose inherit seven possible states. Playing and Recording are also abstract. According to the state design pattern, all those classes have the same operations than DeckController (for some clarity, we didn't write them on the diagram).





## 3.3 Diagram 3



Those diagrams show all the different states in the application and their relations, for a simple cassette deck, a more complex one, and finally a double cassette deck. This last diagram is the most complete one, so this is the one we'll be describing here.

There is a **CassetteDeck** region, that is separated in two concurrent states, for the two decks. There are also a **Motor** region, two **PlayHead** regions and a **CassetteHolder** region for each deck. Those two decks are living separately, so the user can record the other one while it is playing a cassette. When the user turns the cassette deck on, the object enters in the "On" state. Each deck starts in the "Idle" state, and this is the central state: the deck has to be in this state before entering most of the other states.

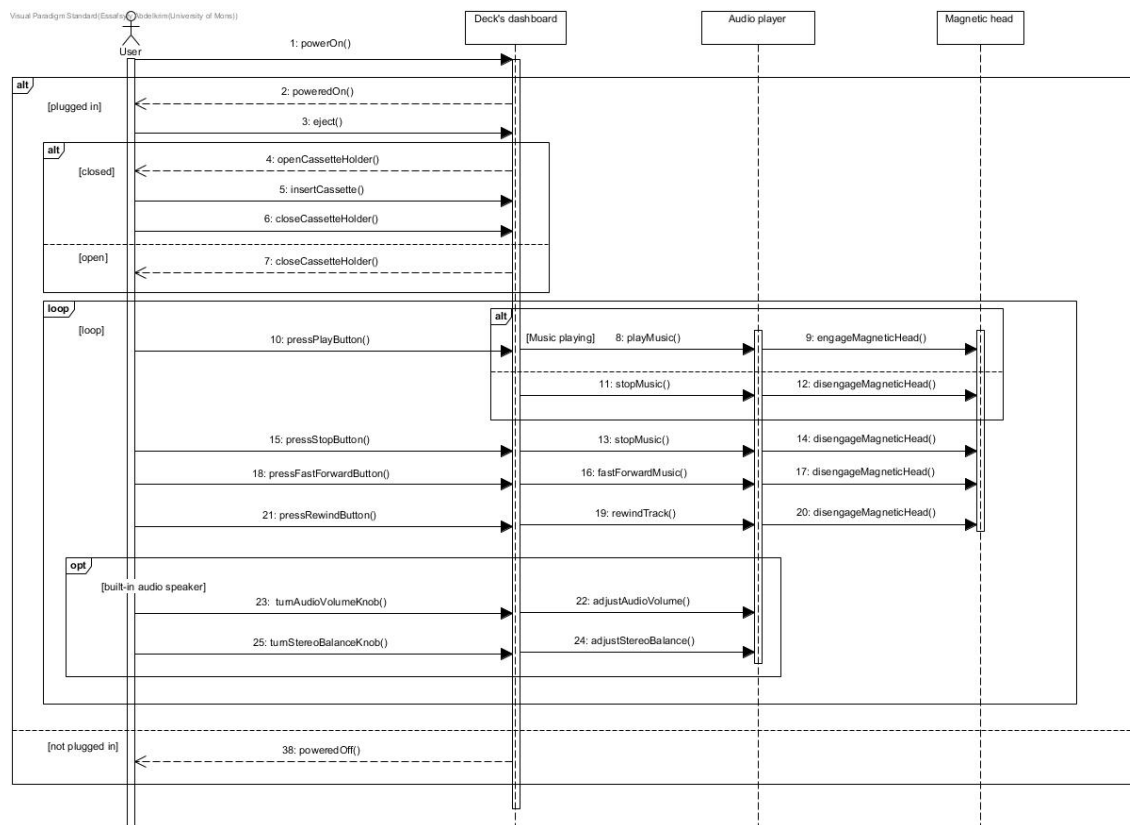
Playing and Recording states contain Paused and Active states. The pause button is a switch button and can be switched at any time, unlike the other buttons.

## 4 Sequence diagrams

The sequence diagram describes the interactions between its components, it also formalizes the behavior described by the Use case diagram.

In this instance, we used two example diagrams to describe this behavior; Music player and Music recorder.

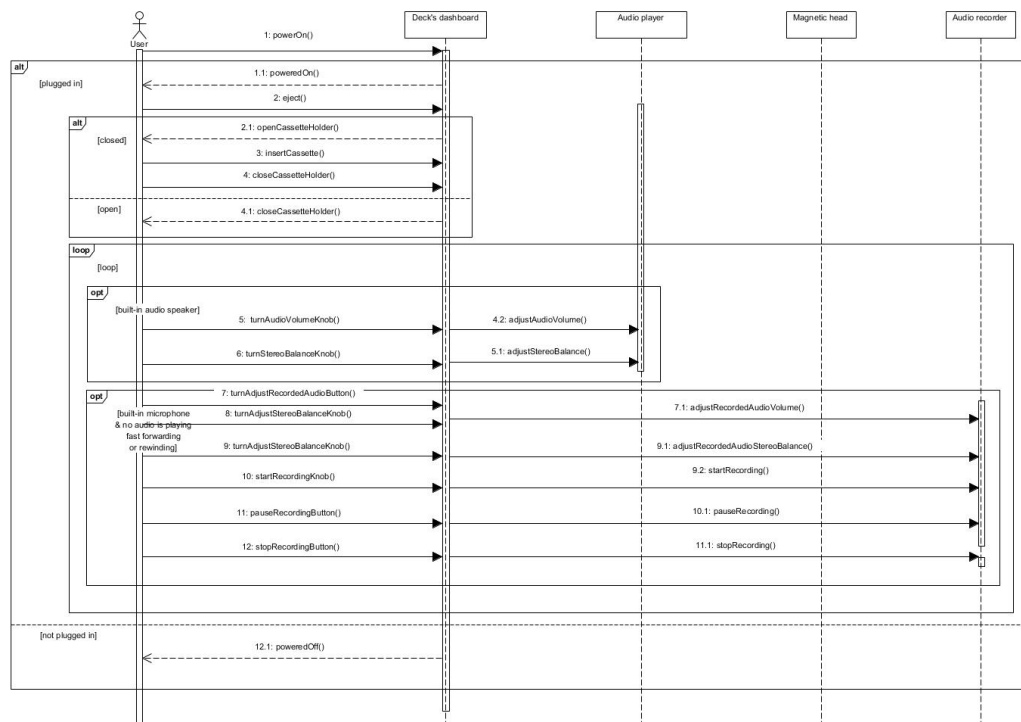
### 4.1 Music Player



This diagram describes the steps a user follows to play, pause, stop, fast forward or rewind music using the deck's dashboard. All these features can be looped, thus allowing the user to play, pause then play, for instance. If the deck has a built-in audio speaker, the user is able to adjust the audio volume and the stereo balance using the respective knob. All operations are only feasible if the deck is plugged in to a power source.

## 4.2 Music Recorder

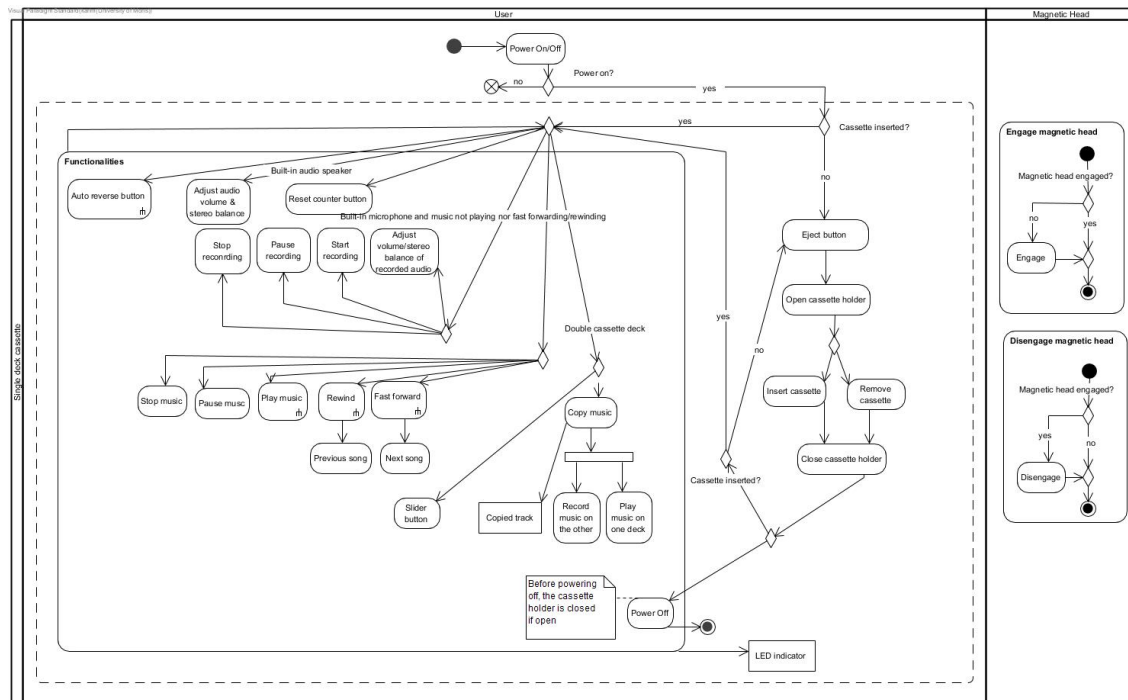
□ Use-Case Diagram Standard (Copyright © 2004 University of Miami)



This diagram describes the steps a user follows to record audio using the deck's audio recorder, provided the former has a built-in microphone. Similar to the Music player diagram, the user has to power on, eject the cassette holder and insert a cassette and finally close the holder to record audio.

The User is able to adjust the recorded audio's volume as well as its stereo balance using the respective knob.

## 5 Activity diagram



The activity diagram specifies in which order and under what conditions the different cases of the Use case diagram will be executed. For example, the user has to power the deck on in order to perform any operation.

Two horizontal *swimlanes* are used to differentiate between the user and the machine's action. in this case, the **Magnetic Head**, which operates whenever the user performs actions related to the cassette. The **User** *swimlane* contains the basic course of action to manipulate the deck i.e. ejecting the cassette holder, inserting a cassette, playing it, recording...

Some features -such as play, pause, stop...- require the deck's magnetic head to be either engaged or disengaged. To visualize this, a symbol  $\bowtie$  is used to demonstrate this relation.

Furthermore, all operations are assembled in an activity called **Functionalities**. The latter being recursive, the user can perform these operations over and over again until the **Power** button is pressed, thus the use of an *interruptible activity region*.

If need be, a link to the git repository used for this project:

<https://github.com/Virtuosek/CassetteDeck>