

# Assignment #1, part 1

- Vytvořte strukturovaný seznam požadavků na aplikaci.
- Vytvořte Use Case Diagram včetně popisu dle standardu UML.

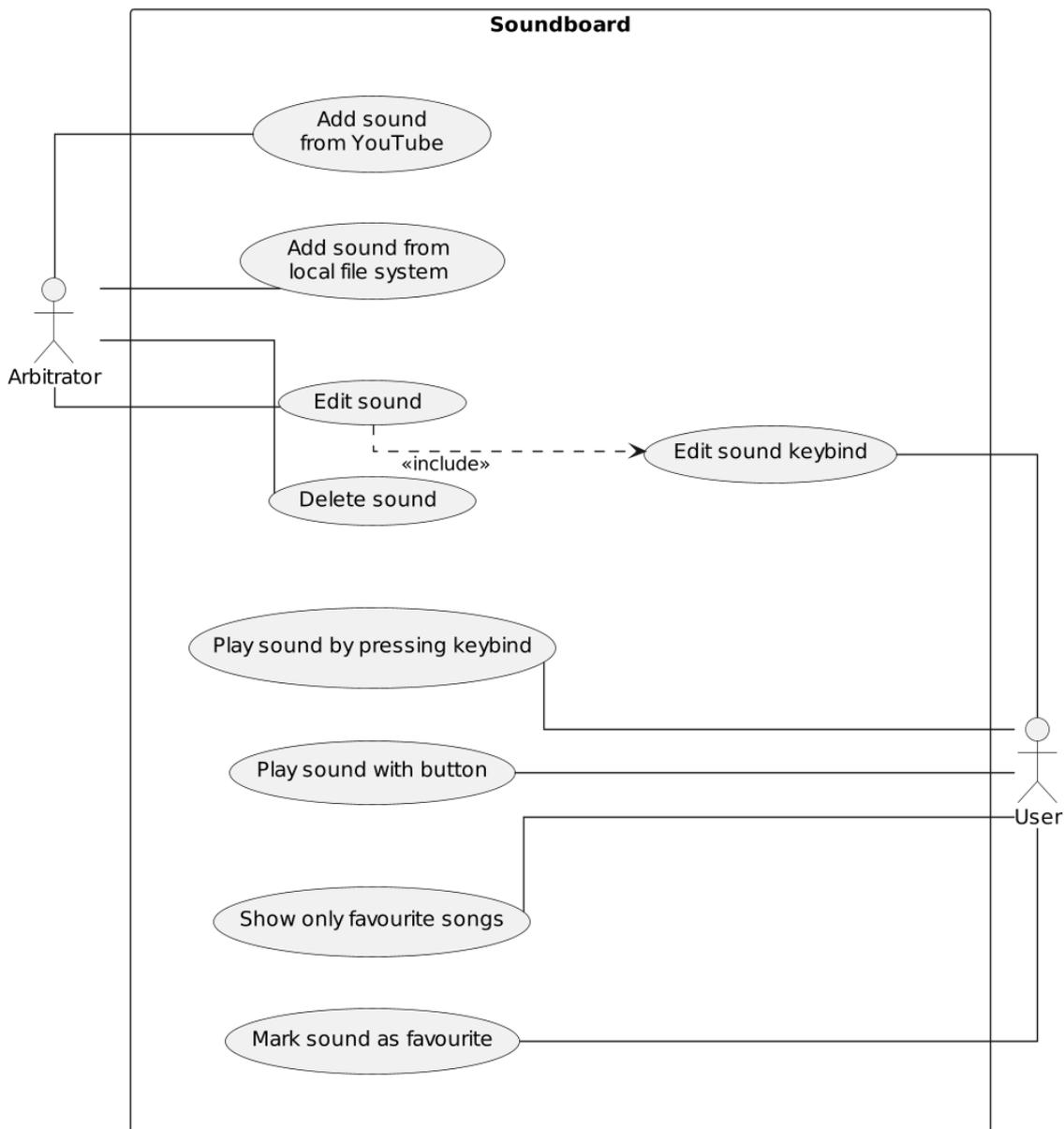
## Functional requirements

ID	Name	Description
1	Play Sound with mouse	User should be able to play a sound by clicking a button.
2	Play Sound with keyboard	User should be able to play a sound by pressing a key, mapped to the sound.
3	Add local sound	Arbitrator should be able to add a sound from his computer to be played later.
4	Add YouTube sound	Arbitrator should be able to add a sound found on YouTube to be played later.
5	Mark sound as favourite	User should be able to mark a sound as favourite.
6	Show only favourite songs	User should be able to show only favourite songs.
7	Edit sound	Arbitrator should be able to edit sound information.
8	Delete sound	Arbitrator should be able to delete a previously added sound.

## Non-functional requirements

ID	Name	Description
1	Language used	The app should be developed using the latest .NET 6.0 standard.
2	UI framework used	The app should be developed using the cross-platform UI framework "Avalonia".
3	UI responsivity	The UI should be responsive and not freeze on time-costly operations.
4	UI understandability	The UI should be easy to understand, and more complex actions should be explained, e.g. with a tooltip.

## Use case Diagram



## Use case Specifications

---

### Add sound from local filesystem

#### Primary actor

- Arbitrator

#### Main flow

1. Actor clicks on a "Add local sound" button
2. A filesystem browser dialog opens
3. Actor picks a valid sound file, clicks Ok in the dialog
4. Sound is added to DB and visible Sound list

### **Alternative flow**

- Actor clicks Cancel in the dialog, meaning no sound will be added
- 

## **Add sound from YouTube**

### **Primary actor**

- Arbitrator

### **Main flow**

1. Actor clicks on a "Add YouTube sound" button
2. A custom dialog opens
3. Actor inputs a YouTube link to the sound they want downloaded
4. Actor passes path on the filesystem where the sound will be downloaded
5. Actor clicks Ok in the dialog
6. Sound is appropriately downloaded
7. Sound is added to DB and visible Sound list

### **Alternative flow**

- Actor clicks Cancel in the dialog, meaning no sound will be added
  - Actor passes an invalid or inaccessible path on the filesystem, an error message will be shown
  - Actor passes an invalid YouTube video link, an error message will be shown
- 

## **Play sound with button**

### **Primary actor**

- Arbitrator, User

### **Preconditions**

- The desired sound has been added to the sound list

### **Main flow**

1. Actor clicks on the Play button next to the desired sound
2. The desired sound will start playing

### **Postconditions**

- The desired sound has started playing
- The Play button will show a Stop symbol for the duration of the sound playing, allowing to prematurely stop the playing of the sound

### **Alternative flow**

- The sound cannot be played (e.g. the sound file has been delete outside the app), an error message will be shown
- 

## **Play sound by pressing keybind**

### **Primary actor**

- Arbitrator, User

### **Preconditions**

- The desired sound has been added to the sound list
- The desired sound has had a keybind assigned to it

### **Main flow**

1. Actor presses the key corresponding to the desired sound
2. The desired sound will start playing

### **Postconditions**

- The desired sound has started playing
- The Play button will show a Stop symbol for the duration of the sound playing, allowing to prematurely stop the playing of the sound

### **Alternative flow**

- The sound cannot be played (e.g. the sound file has been delete outside the app), an error message will be shown
- 

## **Mark sound as favourite**

### **Primary actor**

- Arbitrator, User

### **Preconditions**

- The desired sound has been added to the sound list

### **Main flow**

1. Actor presses the toggleable button next to the desired sound
  2. The desired sound will be marked as favourite
- 

## **Show only favourite songs**

### **Primary actor**

- Arbitrator, User

### **Preconditions**

### **Main flow**

1. Actor presses the Show Favourites button
2. The sound list will be filtered accordingly

### **Postconditions**

- The sound list will have ben filtered accordingly to show only favourite songs/to show all songs, based on the toggle state
- 

## **Edit sound**

### **Primary actor**

- Arbitrator

### **Preconditions**

- The desired sound has been added to the sound list
- The desired sound is selected (i.e. highlighted) in the sound list

### **Main flow**

1. Actor clicks on the Edit button

2. A custom edit dialog opens
3. Actor can edit the path to the sound
4. INCLUDE (Edit sound keybind)
5. Actor presses the Ok button
6. Sound is edited

**Postconditions**

- Sound changes are visible in the sound list

**Alternative flow**

- Actor presses the Cancel button in the dialog, resulting in no changes being made
- 

**Edit sound keybind****Primary actor**

- Arbitrator

**Main Flow**

1. IF Actor presses the Delete key
    1. the current keybind is cleared
  2. IF Actor presses the Escape key
    1. the keybind editing is finished
  3. Actor presses any combination of keys to edit the keybind
- 

**Delete sound****Primary actor**

- Arbitrator

**Preconditions**

- The desired sound has been added to the sound list
- The desired sound is selected (i.e. highlighted) in the sound list

**Main flow**

1. Actor clicks on the Delete button
2. A confirmation dialog opens, asking to confirm the delete operation
3. Actor presses the Yes button
4. Sound is deleted

**Postconditions**

- The deleted sound is no longer visible in the sound list
- The deleted sound is removed from the database

**Alternative flow**

- Actor presses the No button in the dialog, resulting in no changes being made