Name_____Jeffrey Zhang_____ Mark _____/50

[**Instructions**: Remove everything that is not a heading below and fill in with your own diagrams, etc.]

1. Brief introduction __/3

The dynamic sound engine will play sound effects, stingers, and various other sound cues upon being triggered by an event or by the user when interacting with the game.

2. Use case diagram with scenario _14

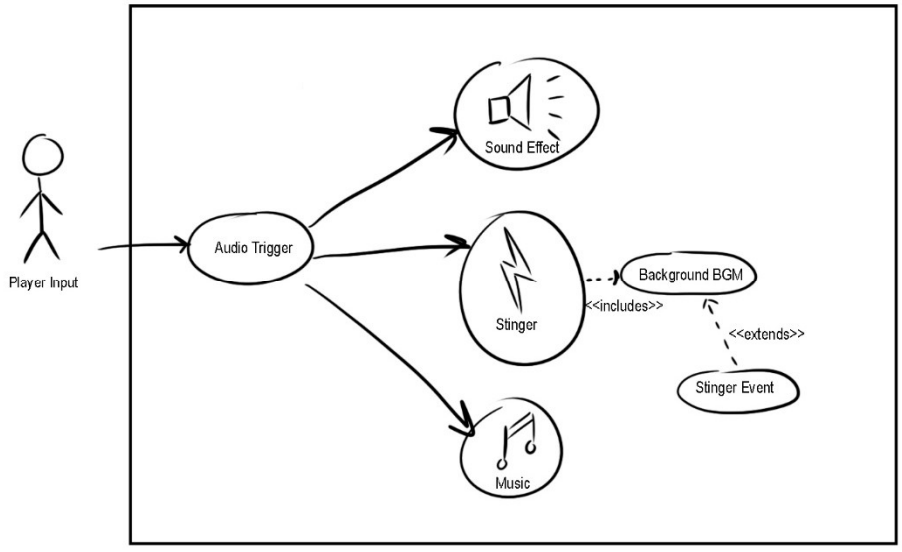
[Use the lecture notes in class.

Ensure you have at least one exception case, and that the <<extend>> matches up with the Exceptions in your scenario, and the Exception step matches your Basic Sequence step.

Also include an <<include>> that is a suitable candidate for dynamic binding]

Example:

Use Case Diagrams



Scenarios

[You will need a scenario for each use case]

Name: Play UI Button Click Sound [Audio Trigger]

Summary: A sound effect plays when the player interacts with the game's UI.

Actors: Player, Game System, Sound Handler

Preconditions: A UI button is clicked.

Basic sequence:

Step 1: Player clicks a button in the UI (e.g., settings cog, start game).

Step 2: The game system registers the click event.

Step 3: The sound handler plays a corresponding sound effect.

Step 4: The sound handler plays a UI sound effect.

Post conditions: A sound effect confirms the button press.

Priority: 2

ID: S01

Name: Play Item Pickup Sound [Audio Trigger]

Summary: A sound effect plays when the player picks up an item.

Preconditions: The player collects an item in the world.

Basic sequence:

Step 1: Player navigates over an item.

Step 2: The game system adds the item to the player's inventory.

Step 3: The sound handler plays a corresponding sound effect.

Post conditions: A sound plays to confirm that the item has been picked up.

Priority: 2

ID: S02

Name: Play Item Use Sound

Summary: A sound effect plays when the player uses an item.

Actors: Player, Game System, Sound Handler

Preconditions: A player uses an item from the inventory.

Basic sequence:

Step 1: Player selects an item and activates it.

Step 2: The game system processes the item’s effect.

Step 3: The sound handler plays the corresponding sound effect.

Post conditions: A sound confirms the item’s activation.

Priority: 2

ID: S03

Name: Player Enemy Interaction Sound

Summary: A sound effect plays when the player interacts with an enemy.

Actors: Player, Enemy, Game System, Sound Handler

Preconditions: The player attacks or gets hit by an enemy via projectiles or melee weaponry.

Basic sequence:

Step 1: Player or enemy shoots or swings at the player.

Step 2: The game system registers the interaction.

Step 3: The sound handler plays a corresponding sound event (i.e. hit marker, hit sound, attack sound).

Post conditions: An effect confirms the interaction.

Priority: 2

ID: S04

Name: Trigger Audio Events

Summary: The game tells the sound handler to initialize and play selected audio track.

Actors: Player, System, Sound Handler

Preconditions: UI Button Click || Item Swap || Item Use || Enemy Entity Interaction with player || Static World Entity Interaction || level load/unload

Basic sequence:

Step 1: Player activates a precondition.

Step 2: The game system processes the interaction.

Step 3: The sound handler plays a corresponding sound effect.

Step 4: Volume of effect is selected and output.

Post conditions: A sound effect plays to confirm interaction.

Priority: 2

ID: S01

Name: Play Static World Interactable object sound

Summary: A sound effect plays when the player interacts with a static object.

Actors: Player, Game System, Sound Handler

Preconditions: The player interacts with a destructible or interactable world object (i.e. traps, environmental projectiles, etc.)

Basic sequence:

Step 1: Player collides with or interacts with a world entity.

Step 2: The game system processes the interaction.

Step 3: The sound handler plays a corresponding sound effect.

Post conditions: A sound effect plays to confirm interaction.

Priority: 2

ID: S05

Name: Trigger Stinger Music Cue

Summary: A stinger plays during key in-game events.

Actors: Game System, Sound Handler

Preconditions: A critical game event occurs.

Basic sequence:

Step 1: Player enters next level, player gets introduced to boss (Flying Spaghetti Monster), other critical events.

Step 2: The game system registers the event.

Step 3: The sound handler plays a stinger.

Post conditions: A unique stinger highlights the event.

Priority: 2

ID: S06

<<includes>>: "Background BGM" to ensure smooth transitions.

Name: Background BGM Loop (Action, Passive tracks)

Summary: The game plays looping background music.

Actors: Game System, Sound Handler

Preconditions: The player loads and enters a new level (Action) The player eliminates all hostiles on the map (Passive).

Basic sequence:

Step 1: Player loads and enters a new level.

Step 2: The game system determines the appropriate BGM action track. (Mapped per level).

Step 3: The background music loops until all hostiles have been eliminated.

Step 4: Stinger plays and Passive track loops until level end.

Post conditions: BGM plays until replaced or stopped.

Priority: 1

ID: S07

Name: Level Load Music Transition

Summary: The game smoothly transitions background music when loading a new level.

Actors: Game System, Sound Handler

Preconditions: A level change occurs and (possible) score screen pops up.

Basic sequence:

Step 1: Player completes and unloads a level.

Step 2: The game system loads the next level.

Step 3: The sound handler plays transition track.

Step 4: The sound handler stops and plays the new level action track upon player entry.

Post conditions: Background music seamlessly transitions between levels.

Priority: 2

ID: S08

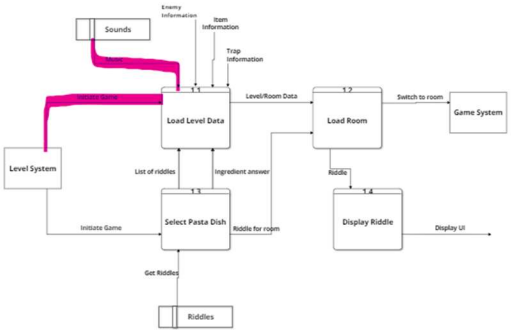
<<extends>>: "Background BGM" for continuous audio flow.

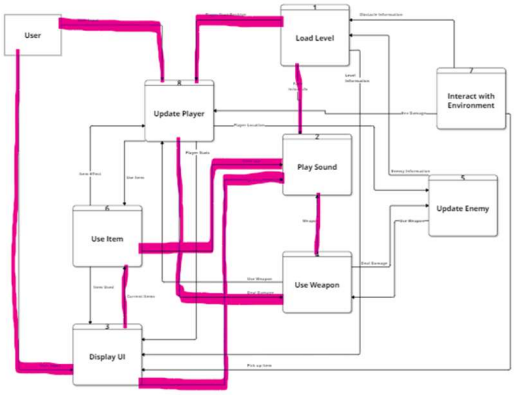
3. Data Flow diagram(s) from Level 0 to process description for your feature _____14

[Get the Level 0 from your team. Highlight the path to your feature]

Example:

Data Flow Diagrams





Process Descriptions

Triggering Sound Events:

IF player interacts with UI → Trigger UI sound effect.
IF player picks up an item → Trigger Pickup sound effect.
IF player uses an item → Trigger Item use sound.
IF player interacts with an enemy → Trigger Combat sound effect.
IF player interacts with a world object → Trigger Interaction sound effect.
IF level starts or changes → Trigger Level transition sound.

Route Audio Event:

IF event type = UI → Play from Sound Effect Handler.
IF event type = Item → Play from Sound Effect Handler.
IF event type = Enemy interaction → Play from Sound Effect Handler.
IF event type = Environmental interaction → Play from Sound Effect Handler.
IF event type = Major event (Boss spawn, cutscene) → Play from Stinger Handler.
IF event type = Level transition → Play from Music Handler.

Managing Background Music

IF level starts → Load corresponding Background BGM.
WHILE level is active:
IF no stinger event → Continue looping Background BGM.
IF stinger event occurs:

Pause Background BGM.
Play Stinger Music.
Resume Background BGM after stinger ends.
IF level changes → Crossfade to new Background BGM.

Handling Overlapping Sounds
Limit active sound effects to 5 simultaneous instances.
IF a new sound effect starts while 5 are already playing:
IF the oldest sound is non-essential (e.g., footsteps) → Stop it.
ELSE reject the new sound request.
For looping sounds (e.g., background ambiance):
Only restart when a different track is triggered.
Maintain smooth transitions to avoid abrupt stops.

Stinger Event Logic
IF a major event occurs (boss spawn, cutscene, cinematic trigger):
Reduce Background BGM volume.
Play Stinger Music.
Wait for Stinger Music to end.
Restore Background BGM volume.
IF multiple stinger events occur simultaneously:
Queue additional stingers rather than playing them at the same time.

Fallback and Fail-Safes
IF a required sound file is missing → Play a default "error" sound.
IF a looping sound fails to stop correctly → Force-stop after 10 seconds.
IF two identical sounds play within 0.1 seconds → Merge them into one instance.

4. Acceptance Tests _____9

[Describe the inputs and outputs of the tests you will run. Ensure you cover all the boundary cases.]

Test Case	Input	Expected Output	Notes
UI Button Click Sound	Click UI Button	Corresponding UI sound Plays	
Item Pickup Sound	Pick up an item	Pickup SFX plays	
Item Use Sound	Use a consumable	Use sound effect plays	
Enemy Interaction	Attack an enemy/vice versa	Combat sound plays	Combat
Static Object Interaction	Interact with a destructible/trap	Object interaction sound plays	Static world traps, destructible blocks, etc.
Stinger Event Trigger	Finish a puzzle,	Stinger music plays	Possibly more cases planned

	perform combos		
Level Load Transition	Load new level	BGM changes smoothly	BGM -> Loading music -> BGM

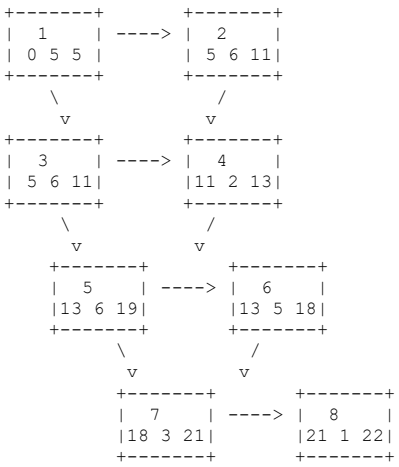
5. Timeline _____/10

[Figure out the tasks required to complete your feature]

Work items

Task	Duration (PWks)	Predecessor Task(s)
1. Requirements Collection	2	-
2. Sound Asset Collection	3	1
3. Audio Engine Integration	4	2
4. Implement Sound trigger Logic	5	3
5. BGM system	3	3
6. Stinger Handling	4	5
7. Testing	3	4, 6
8. Final Adjust, Optimization, Deployment.	1	8

Pert diagram



Gantt timeline