**Casual Use Case and Domain Model of**

**Little Animal Horror Zoo**

Groupe Pizza - Martin, Kasper og Mads

**Blue = Doesn't exist in code**

**Green = Nouns = Class/attribut**

**Orange = Verbs = Methods**

**Title:** Play Game

**Primary Actor:** Player

**Main Success Scenario:**

1. **Player starts game**
2. Player **gives input** to **Snake** to **move** and game displays position info.
   1. Snake moves one **square** in given **direction**
   2. **System** moves **Rabbit** 1 square i random direction
3. Repeat **a** and **b** until Snake is beside Rabbit
4. Game display, rabbit **talks**
5. Player moves Snake to same square as Rabbit
6. Snake **eats** Rabbit
7. Game display **victory**!
8. Games ends and data gets sent to developers

**Alternative Flow (Extensions)**

**At all times, if the system fails:**

1. Show error message
2. Reset game
3. Send crash data to developers

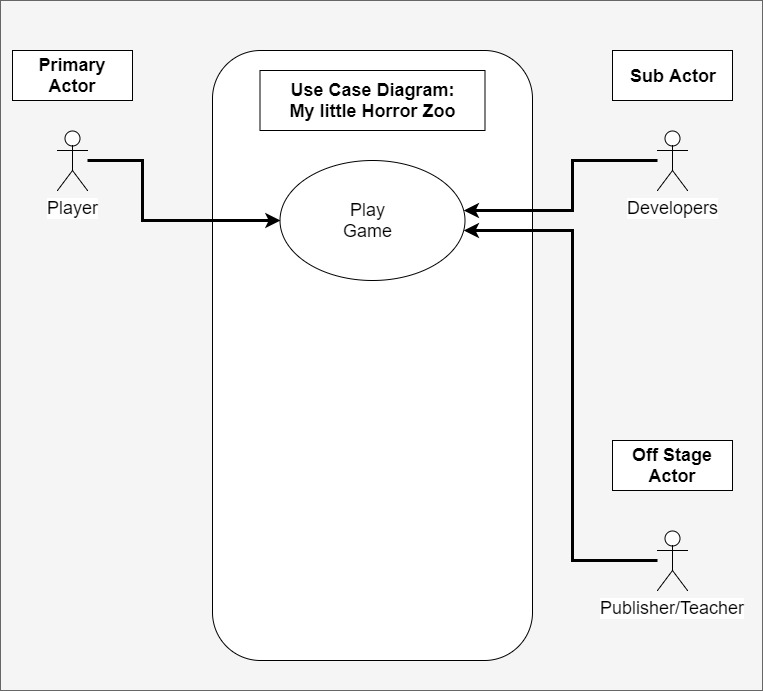
**3A. Rabbit/Snake is next to game area wall and tries to move past the wall:**

1. Moves Rabbit/Snake to other side of board (Periodiske Randbetingelser)

**3B. Snake doesn’t catch Rabbit in allowed moves:**

1. Move counter exceeded
2. Send Game Over message
3. Reset game

**Use Case Diagram:**

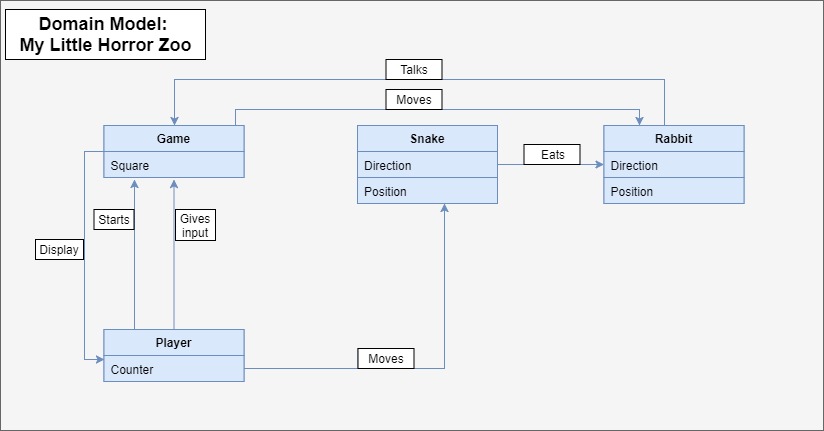
****

**Domain Model:**

**Nouns - (navneord) - Class/attribut:**

1. Player
2. Game
3. Snake
4. ~~Direction~~ - attribut
5. ~~Square~~ - attribut
6. Rabbit
7. ~~Victory~~  - action
8. ~~Error~~ - action
9. ~~Counter~~ - attribut

**Verbs - (Udsagnsord) - Methods:**

1. Starts
2. Display
3. Gives input
4. Talks
5. Moves
6. Eats

**Class Model:**

