

5. Entwurf des Skeletons

31 – marci_und_die_mitbewohner

Konsulent:
Kovács Márton

Mitglieder

Seben Domonkos András
Szapula László
Filip Krisztina
Golej Márton Marcell
Visy Tamás

(ETBCNP)
(DJQOM9)
(QE4L0M)
(V1BYVS)
(CTSJ3H)

domi.seben@gmail.com
szapula.laszlo.99@gmail.com
fkriszta997@gmail.com
golejmarci@gmail.com
tamas.visy@gmail.com

5. Entwurf des Skeletons

5.1 Die reellen Use-Cases des Skeletons

5.1.1 Use-case Diagram



5.1.2 Use-case Beschreibungen

Name der Use-case	Player steps on a stable icefloat
Kurzbeschreibung	Der Spieler tritt auf eine solche Eisscholle, die alle Charakters halten kann.
Aktoren	Player
Drehbuch	1) Der Benutzer wählt Bewegung aus. 2) Der Benutzer gibt die ID der Eisscholle ein.

Name der Use-case	Player steps on an unstable icefloat
Kurzbeschreibung	Der Spieler tritt auf eine solche Eisscholle, die nur eine bestimmte Anzahl von Charakters halten kann.
Aktoren	Player
Drehbuch	1) Der Benutzer wählt Bewegung aus. 2) Der Benutzer gibt die ID der Eisscholle ein.

Name der Use-case	Player steps in a hole
Kurzbeschreibung	Der Spieler tritt in ein Loch und fällt ins Wasser.
Aktoren	Player
Drehbuch	<ol style="list-style-type: none"> 1) Der Benutzer wählt Bewegung aus. 2) Der Benutzer gibt die ID der Eisscholle ein. 3) Der Charakter fällt ins Wasser.

Name der Use-case	Player picks up item
Kurzbeschreibung	Der Spieler nimmt einen Gegenstand von der Eisscholle auf.
Aktoren	Player
Drehbuch	<ol style="list-style-type: none"> 1) Der Benutzer wählt Gegenstand Aufnehmen aus.

Name der Use-case	Player picks up food
Kurzbeschreibung	Der Spieler nimmt ein Lebensmittel auf der Eisscholle auf.
Aktoren	Player
Drehbuch	<ol style="list-style-type: none"> 1) Der Benutzer wählt Gegenstand Aufnehmen aus. 2) Die Temperatur der Benutzer wird erhöht.

Name der Use-case	Player shovels snow
Kurzbeschreibung	Der Spieler schaufelt Schnee auf der Eisscholle, auf der er ist.
Aktoren	Player
Drehbuch	<ol style="list-style-type: none"> 1) Der Benutzer wählt Schaufeln aus.

Name der Use-case	Player builds rocket
Kurzbeschreibung	Der Spieler baut die Leuchtpistole zusammen.
Aktoren	Player
Drehbuch	<ol style="list-style-type: none"> 1) Der Benutzer wählt Leuchtpistole Bauen aus. 2) Falls alle Bestandteile und Spieler auf einer Scholle sind, wird die Leuchtpistole gebaut.

Name der Use-case	Player examines icefloat
Kurzbeschreibung	Der Forscher überprüft, wie viel Spieler eine Eisscholle halten kann.
Aktoren	Researcher
Drehbuch	<ol style="list-style-type: none"> 1) Der Benutzer wählt Eisscholle Überprüfen aus. 2) Der Programm zeigt die Kapazität der Eisscholle.

Name der Use-case	Player builds igloo
Kurzbeschreibung	Der Eskimo baut einen Iglu auf der Eisscholle, auf der er steht.
Aktoren	Eskimo
Drehbuch	<ol style="list-style-type: none"> 1) Der Benutzer wählt Iglu Bauen aus.

Name der Use-case	Icefloat topples over
Kurzbeschreibung	Falls es zu viele Charakter auf einer instabilen Eisscholle gibt, kippt sie um.
Aktoren	Player
Drehbuch	1) Der Benutzer tritt auf die instabile Eisscholle. 1.A) Es gibt zu viele Charakter da und die Eisscholle kippt um und die Charakter darauf sterben.

Name der Use-case	Player drowns
Kurzbeschreibung	Der Spieler ist ins Wasser gefallen, hat keinen Taucheranzug und niemand kann ihn retten.
Aktoren	Player
Drehbuch	1) Der Charakter stirbt.

Name der Use-case	Player swims back
Kurzbeschreibung	Der Spieler ist ins Wasser gefallen, aber hat einen Taucheranzug.
Aktoren	Player
Drehbuch	1) Der Charakter tritt auf die vorige Eisscholle.

Name der Use-case	Player gets rescued
Kurzbeschreibung	Der Spieler ist ins Wasser gefallen, aber ein anderer Spieler auf einer benachbarten Eisscholle hat ein Seil.
Aktoren	Player
Drehbuch	1) Der Charakter tritt auf der Eisscholle, wo der andere Spieler steht, der ein Seil hat.

Name der Use-case	Game lost
Kurzbeschreibung	Eine der Charaktere ist gestorben.
Aktoren	Player
Drehbuch	1) Der Programm schreibt aus, dass das Spiel zu Ende ist.

Name der Use-case	Game won
Kurzbeschreibung	Der Raket wird gebildet.
Aktoren	Player
Drehbuch	1) Das Programm schreibt aus, dass das Spiel gewonnen ist.

Name der Use-case	Snowstorm appears
Kurzbeschreibung	Am Ende aller Runde wird einen Schneesturm gestartet.
Aktoren	Controller
Drehbuch	1) Der Benutzer hat 4-mal Aktionen durchgeführt. 2) Der Controller startet Schneesturm. 2.A) Falls ein Charakter auf einer solchen Scholle steht, wo es Schneesturm und keine Iglu gibt, dann der Charakter verliert Temperatur. 2.A.1) Falls die Temperatur zu 0 sinkt, der Charakter stirbt.

Name der Use-case	Controller generates icefloats
Kurzbeschreibung	Der Controller generiert Eisschollen nach der MudM Algorithmus.
Aktoren	Controller
Drehbuch	1) Der Controller platziert alle Eisschollen.

5.2 Das Benutzerinterface des Skeletons, Dialoge

Für jede Methode werden die einzelne Funktionsaufrufe aufgelistet. Die Konvention ist hier zu lesen:

[Absatz][Name des Objektes, woran die Funktion aufgerufen wurde].[Name der Funktion]

Das Maß der Absatz (Tabulation) wurde dadurch bestimmt, wann die Funktion in der Rufkette gerufen wurde.

Während dem Lauf des Programms werden einige Nachrichten auch erscheinen:
Am Anfang des Spiels wird der folgende Text ausgeschrieben (start() schon zurückgekehrt):

Game started

Oder wenn das n-te Spieler dran ist: *Player n's turn*

Bei jeder Arbeit schreibt es aus: *What to do?* //Viermal während einer Runde, immer vor der Arbeit.

Bei der Veränderung der Körpertemperatur schreibt das Programm aus, was war und auf was wurde sie verändert. Gleichfalls bei Aufnahme eines Gegenstandes, wird das Moment auch in Form einer Nachricht gemerkt.

Wenn jemand verliert, endet das Spiel, und das wird folgenderweise ausgeschrieben: *End game. You loose.*

Falls der Spiel gewonnen wurde: *You win! :)*

Bei jedem abzweigenden Punkt werden die mögliche Befehle aufgelistet, wo der entsprechende Buchstabe eingetippt werden soll um ein Szenario auszuwählen.

Zum Beispiel die möglichen Befehle für einer Arbeit sind:

- *M: moveTo* //Treten an der mit dem Kennzeichen gegebene Eisscholle.
 - *1-m: Eisschollekennzeichen* //Die m Nachbar auswählbar.

- **P: pickUp** //Die gegebene Gegenstand aufnehmen.
 - Es wird immer ausgeschrieben ob ein Gegenstand zur Verfügung steht an der Eisscholle, woran der Spieler getreten hat, folgenderweise: *Item: {Name der Gegenstand}* (food/cartridge/pistol/rope/shovel/diving-suit/signal-flare)
- **I: buildIgloo** // Ein Iglu bauen. (nur ein Eskimo kann das machen)
- **C: checkCapacity** //Ablesen die Kapazität der Eisscholle (nur ein Forscher kann das machen)
 - **I-m: Eisschollekennzeichen** //Die m Nachbar auswählbar.
- **R: buildRocket** // Leuchtpistole zusammenbauen.
- **S: shovel** //Gegebene Einheiten von Schnee wegschaufeln.

Ein Beispiel für einen Befehlgeben für die Arbeit:

What to do?

M: moveTo

P: pickUp

I: buildIgloo

C: checkCapacity

R: buildRocket

S: shovel

>> I

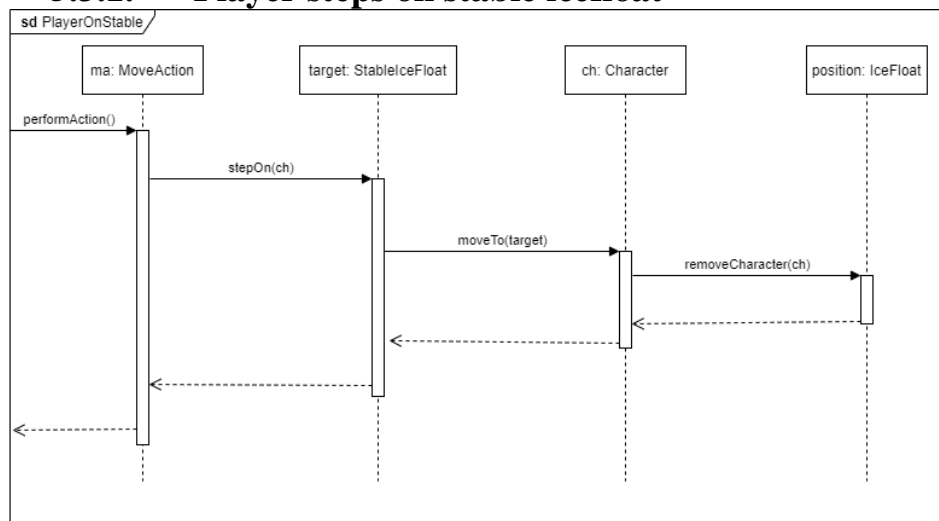
ba.BuildAction()

ba.performAction()

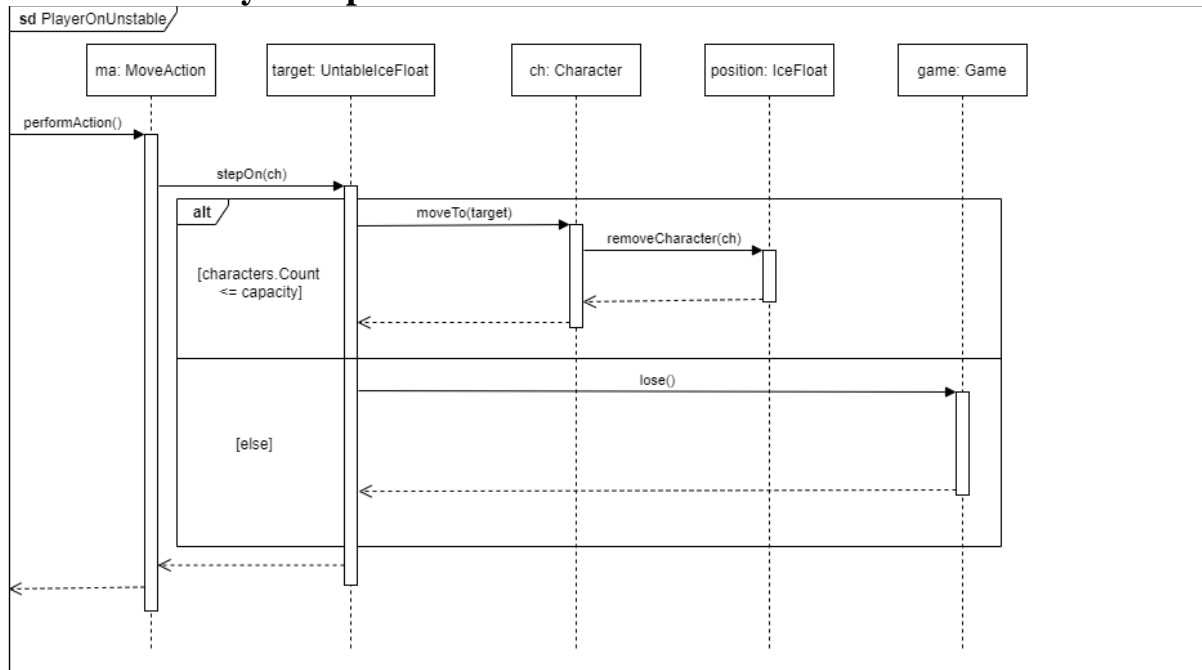
usw.

5.3 Sequenzdiagramme der inneren Arbeitsweise

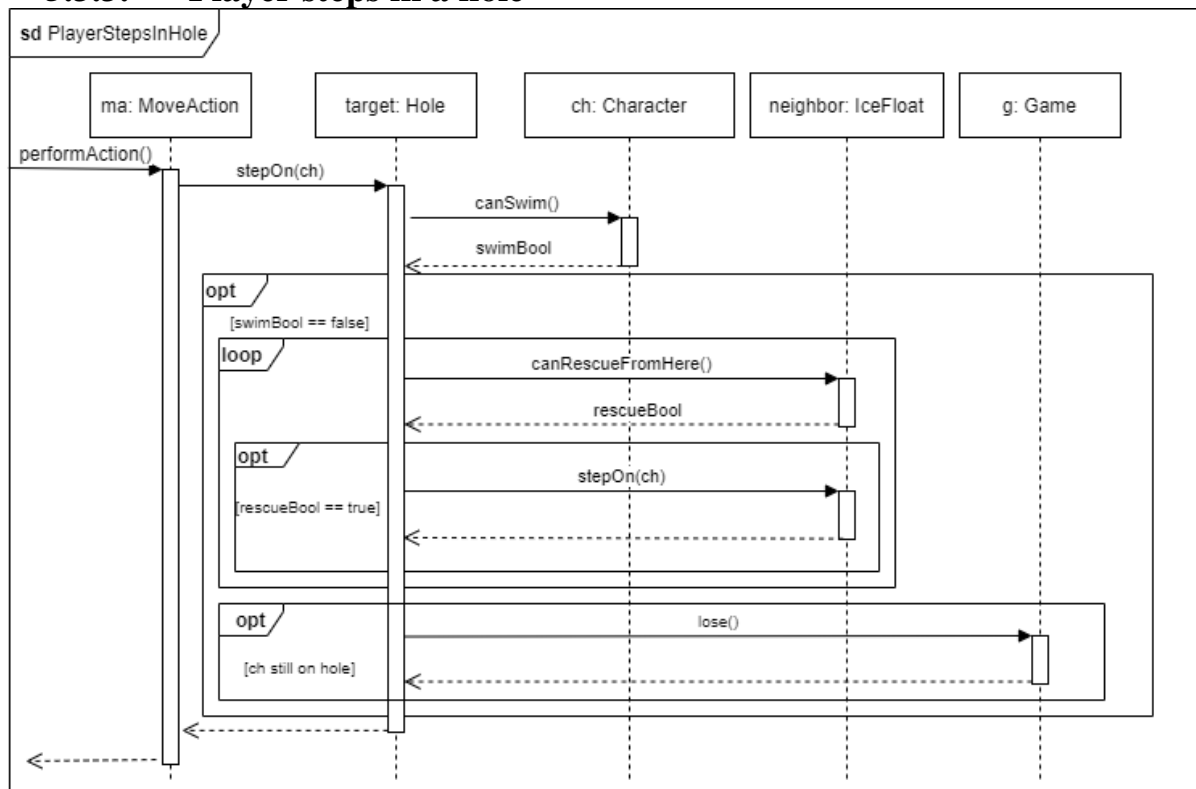
5.3.1. Player steps on stable icefloat



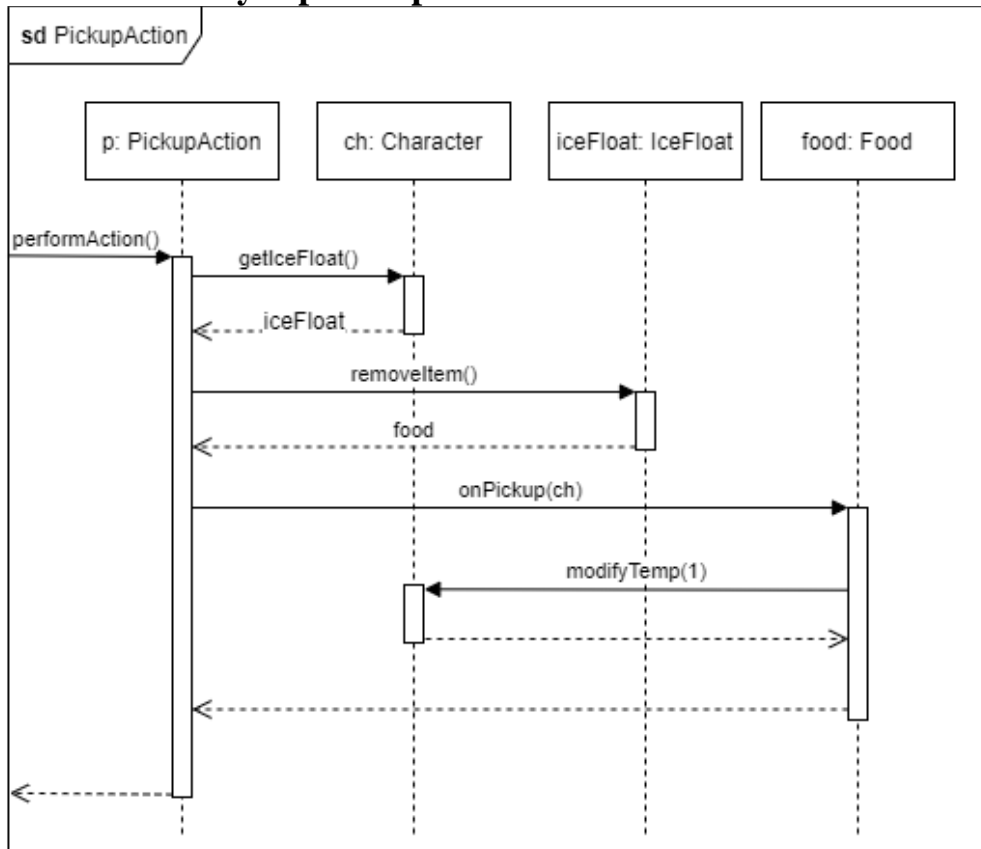
5.3.2. Player steps on unstable icefloat



5.3.3. Player steps in a hole

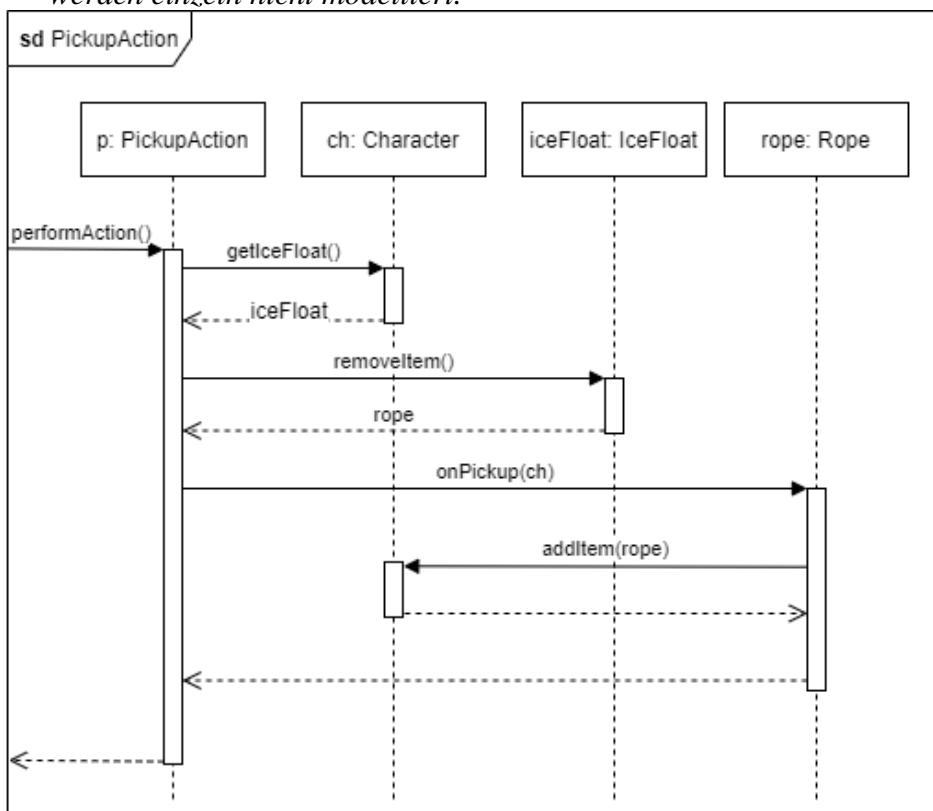


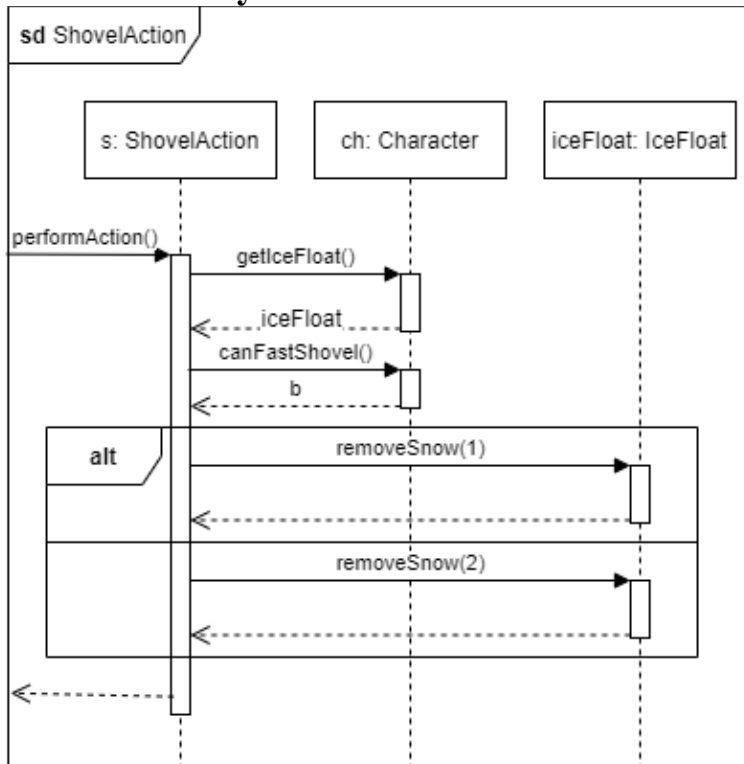
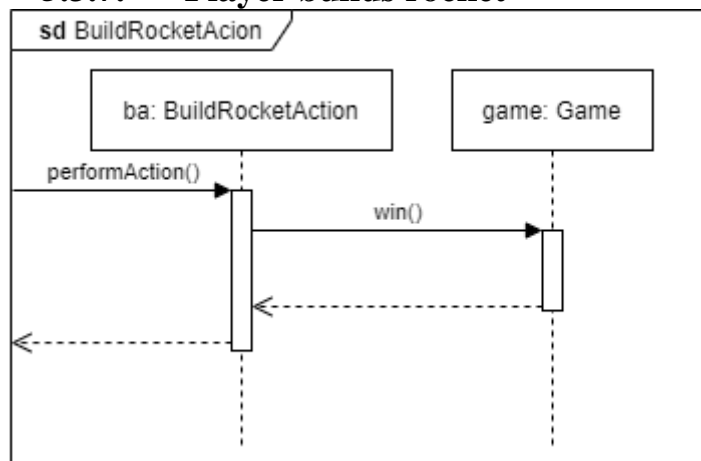
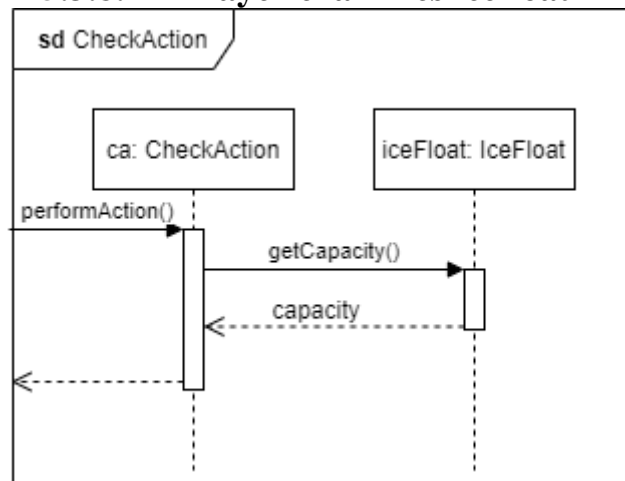
5.3.4. Player picks up food

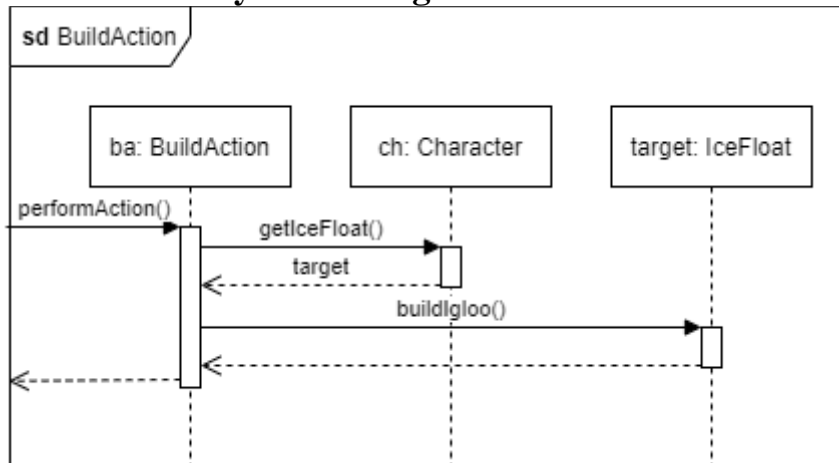
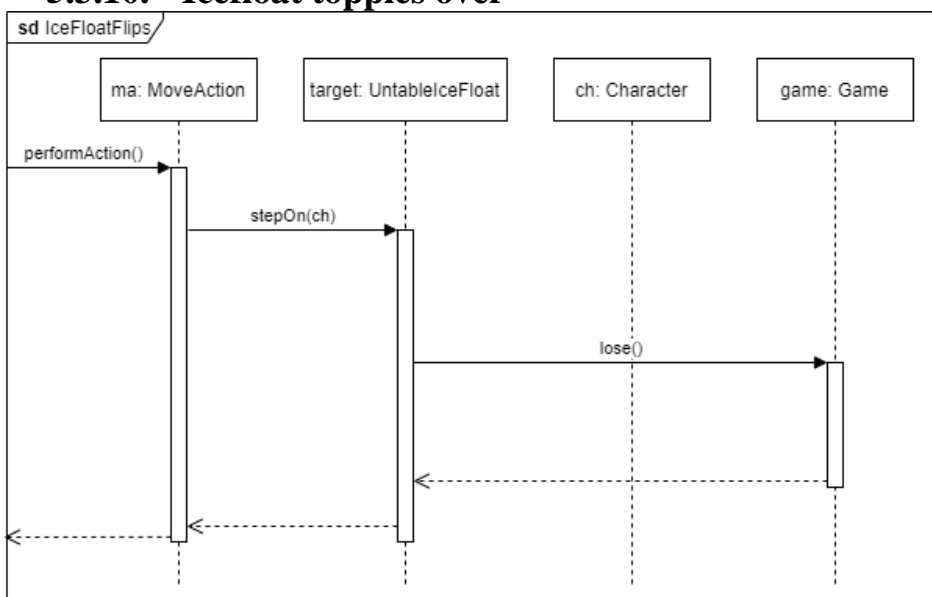
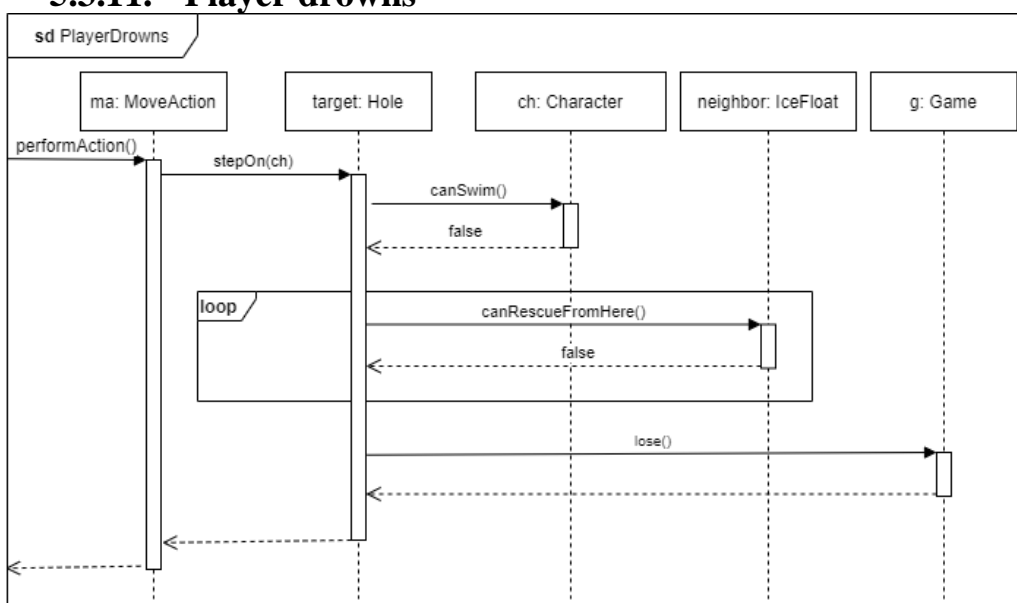


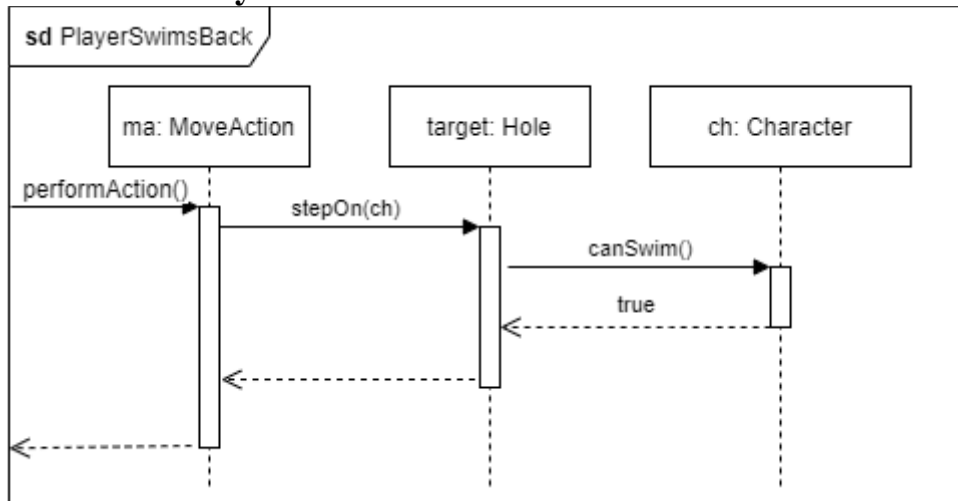
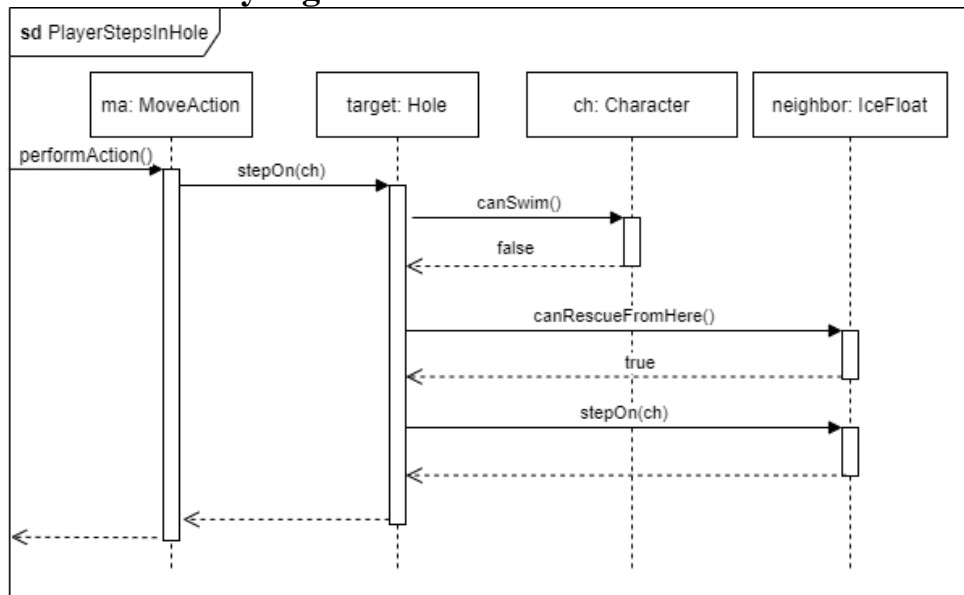
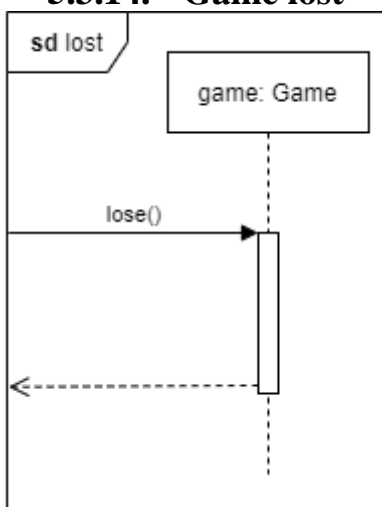
5.3.5. Player picks up rope (/ diving suit / shovel)

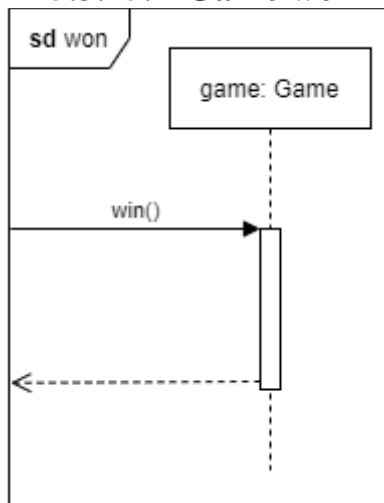
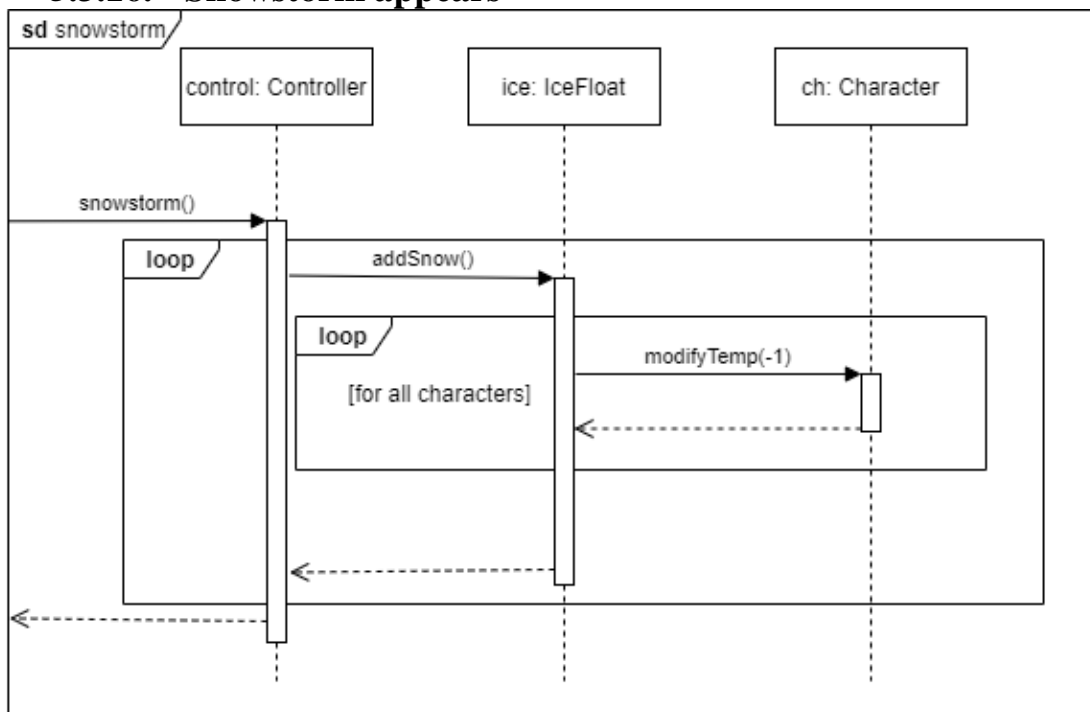
Bei der Aufnahme von dem Taucheranzug und der Schaufel ist derselbe Ablauf gültig, die werden einzeln nicht modelliert.

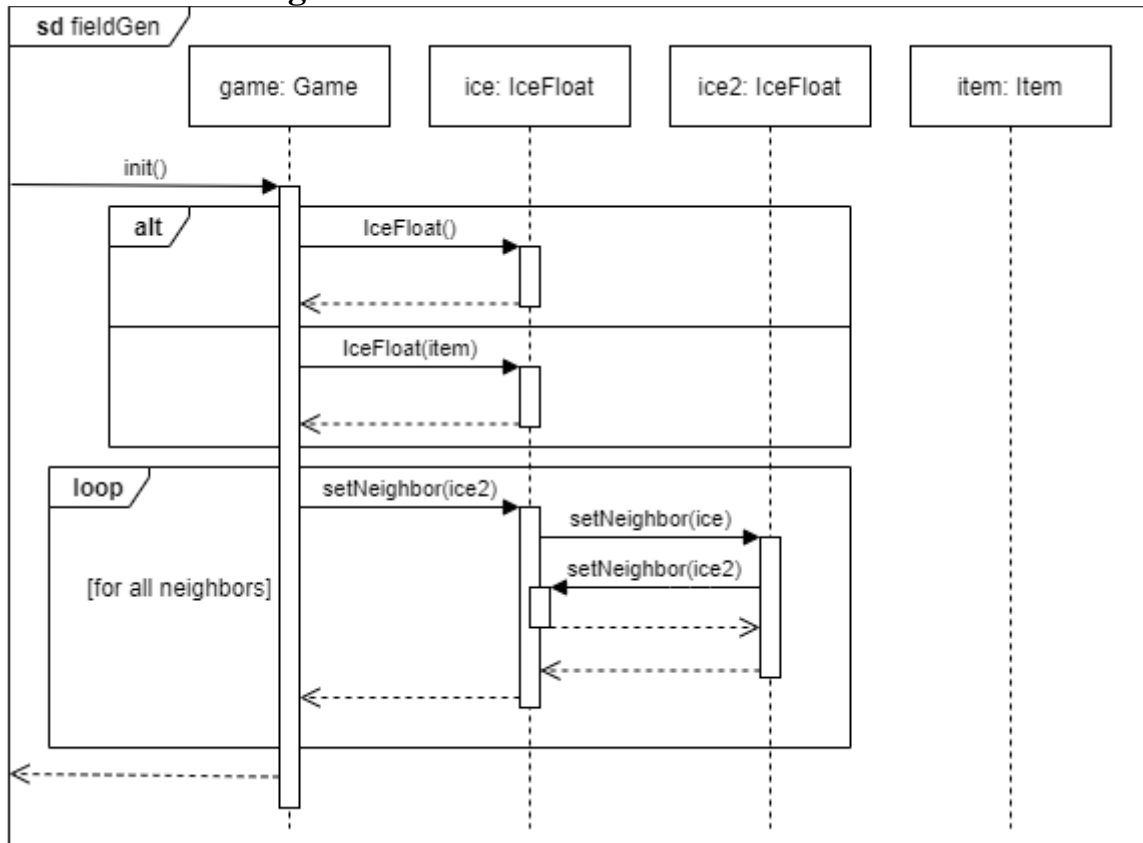


5.3.6. Player shovels snow**5.3.7. Player builds rocket****5.3.8. Player examines icefloat**

5.3.9. Player builds igloo**5.3.10. Icefloat topples over****5.3.11. Player drowns**

5.3.12. Player swims back**5.3.13. Player gets rescued****5.3.14. Game lost**

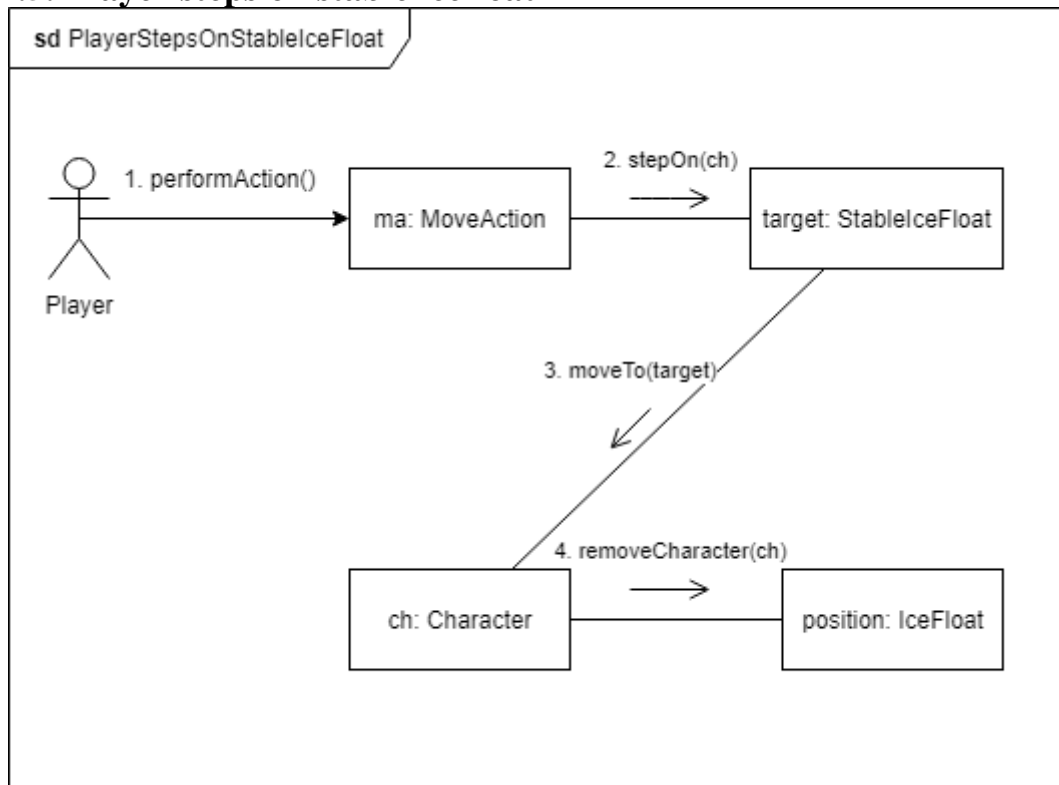
5.3.15. Game won**5.3.16. Snowstorm appears**

5.3.17. Game generates icefloat

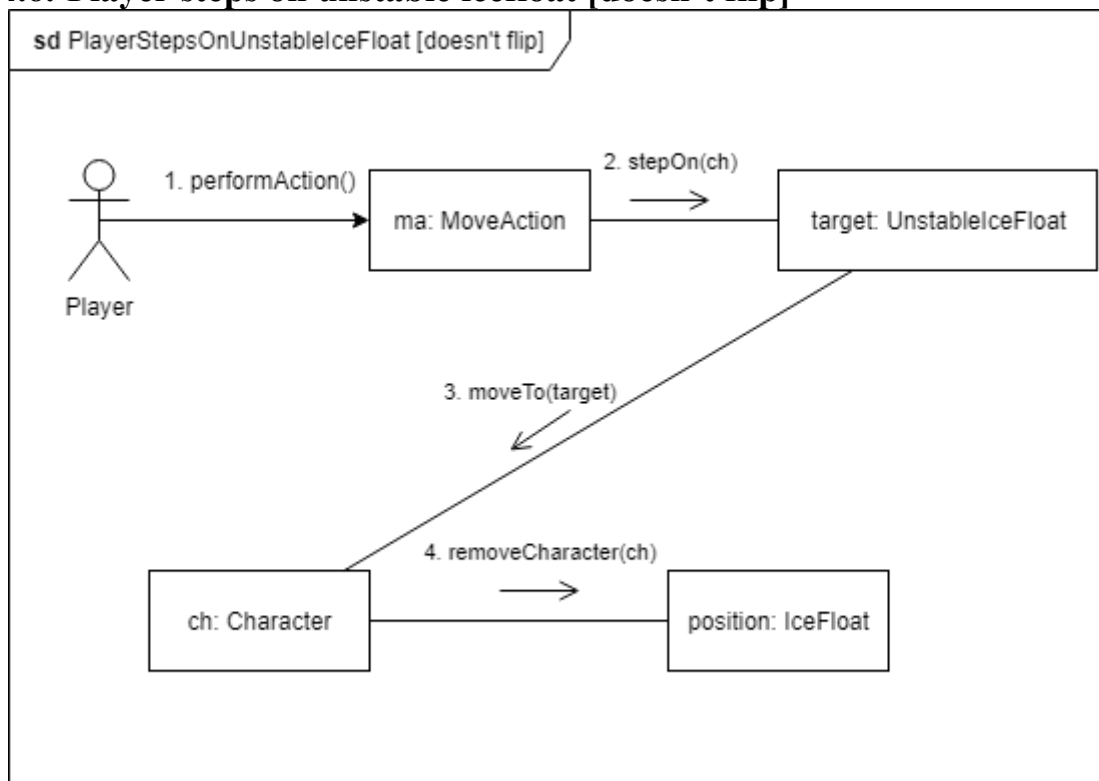
Bemerkung: Dieses Diagramm konzentriert sich nur auf die Tätigkeit, wie die Funktion `init()` Eisschollen generiert und ihre Nachbarschaften einstellt. Um einen allgemeinen Blick auf die ganze Verantwortlichkeit dieser Funktion siehe **4.4.2**

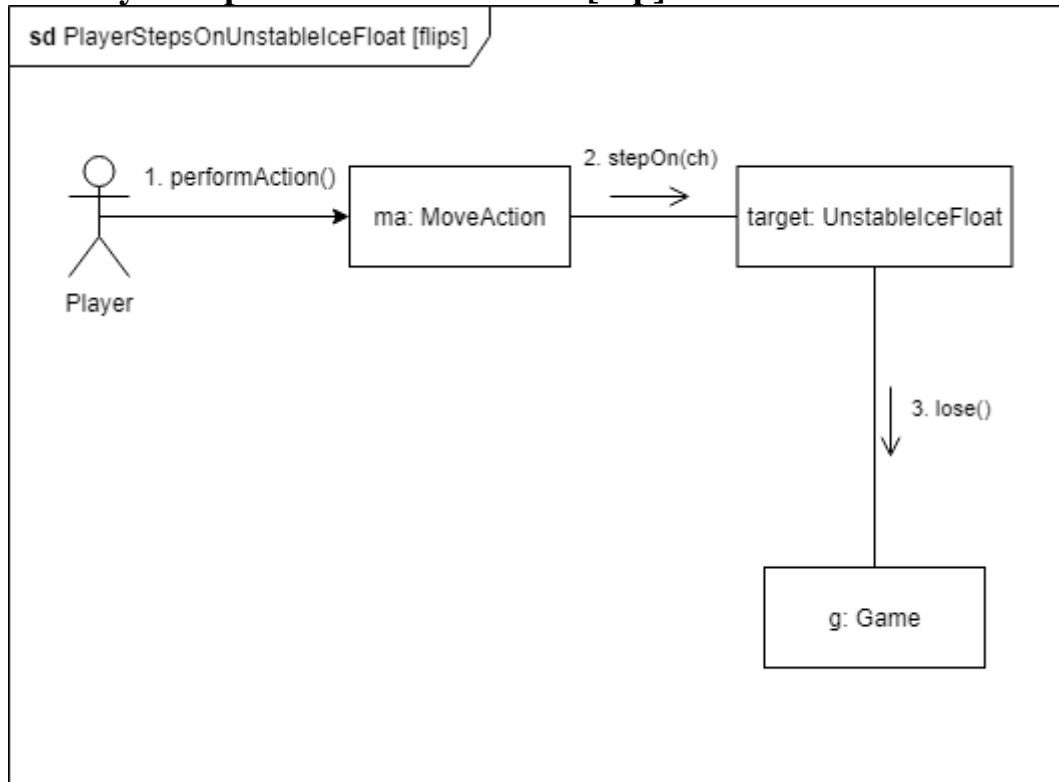
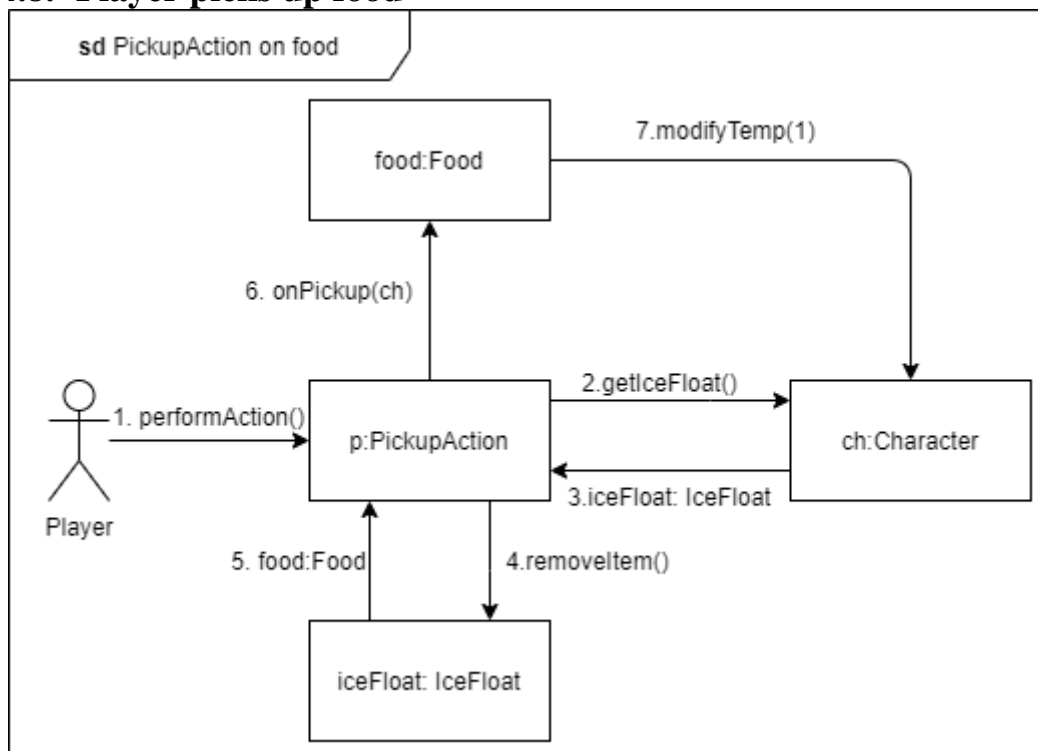
5.4 Kommunikationsdiagramme

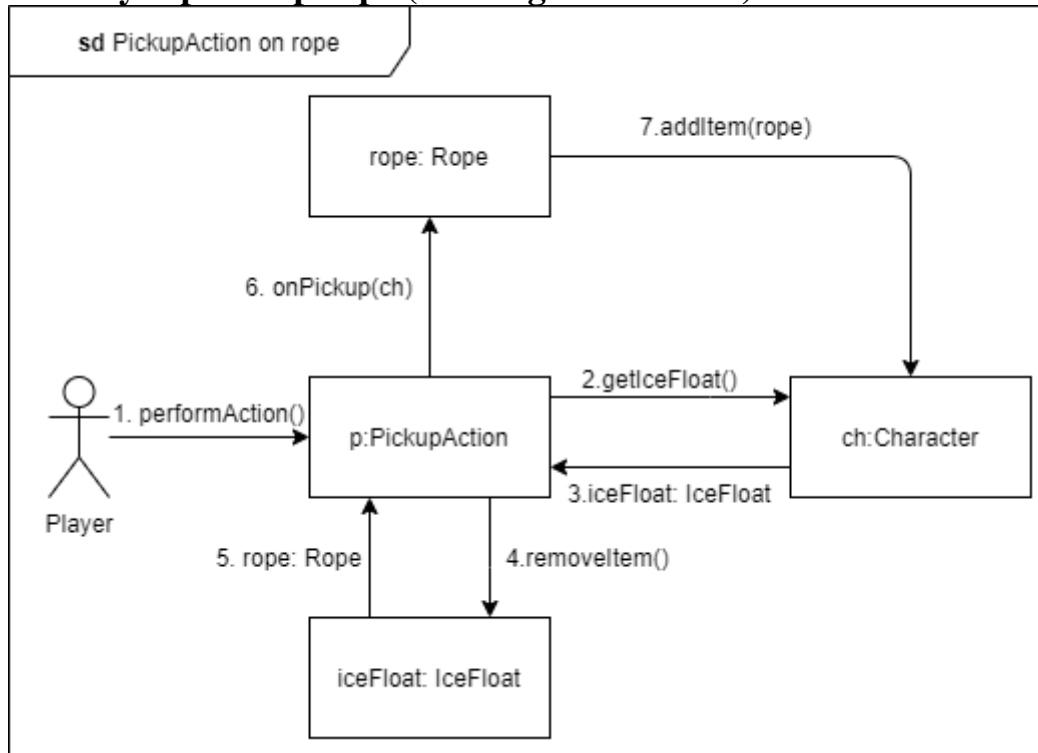
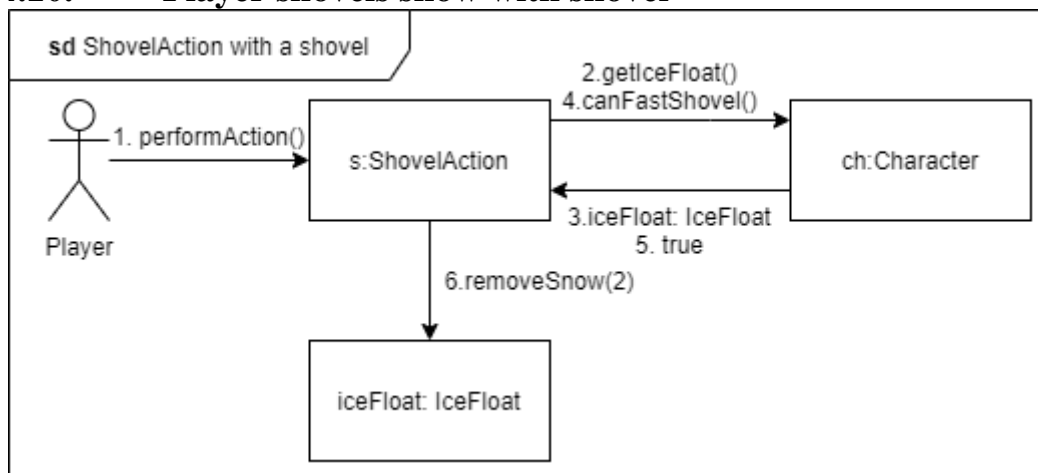
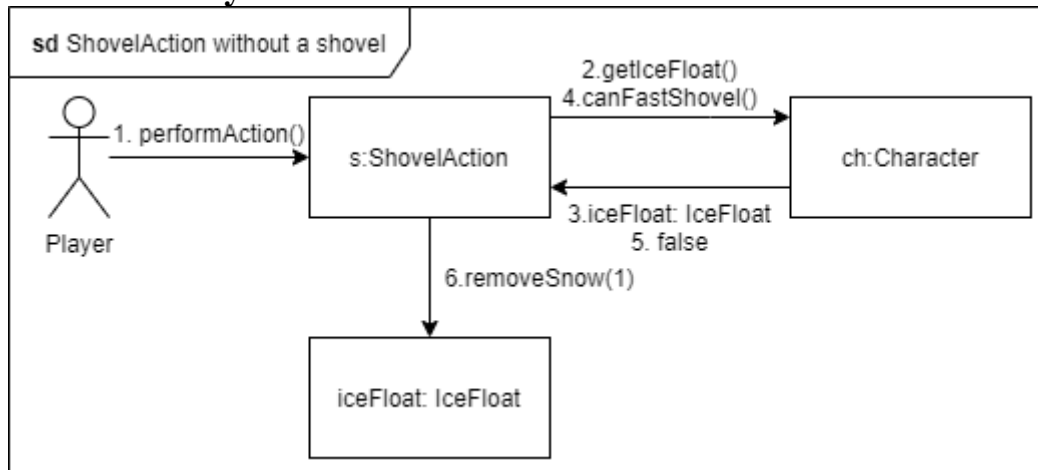
5.4.5. Player steps on stable icefloat

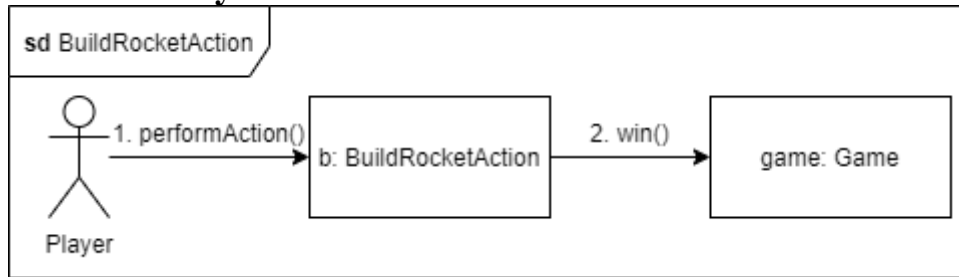
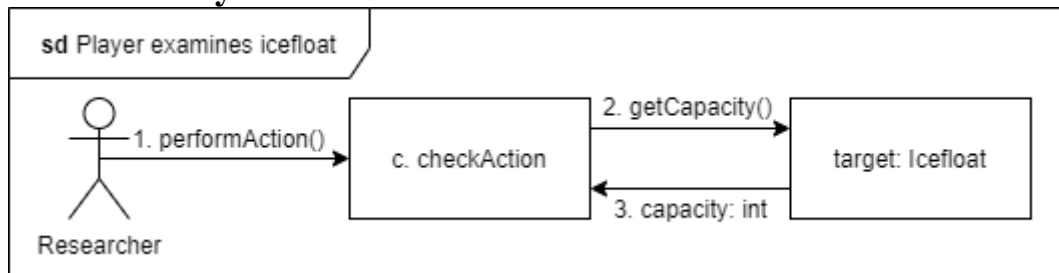
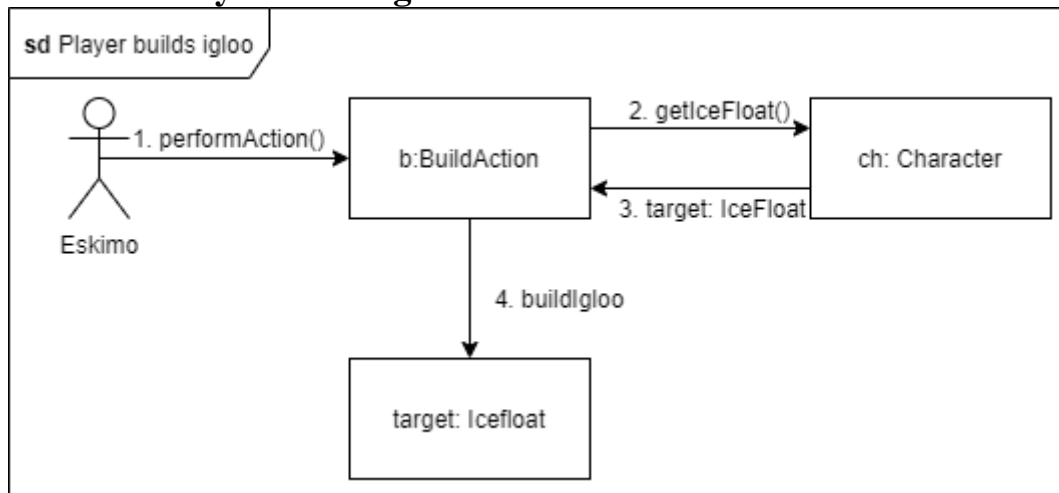


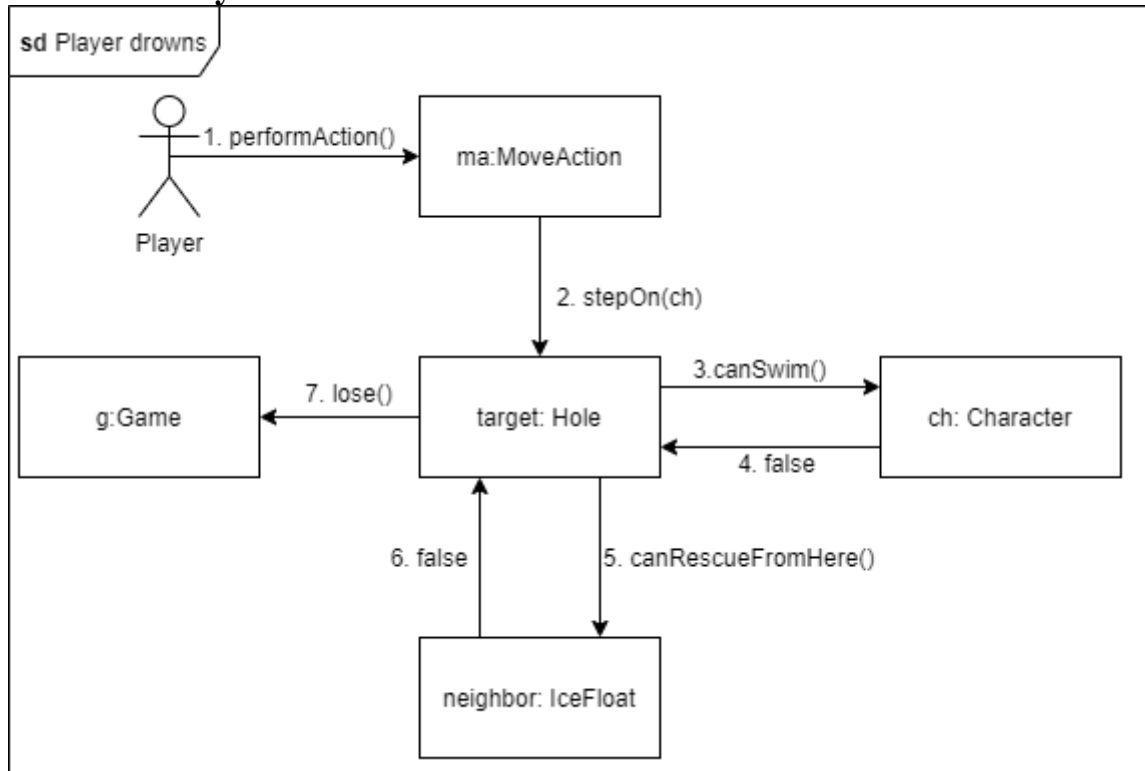
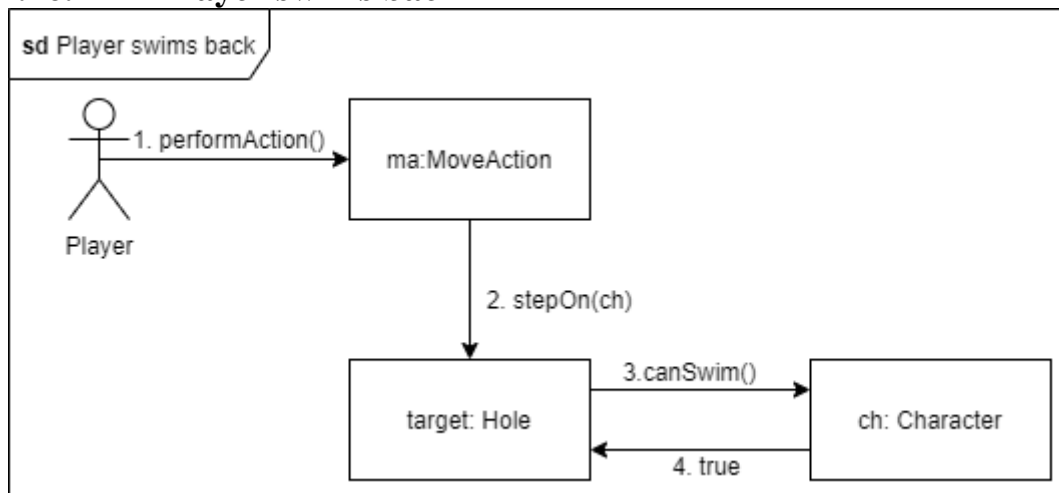
5.4.6. Player steps on unstable icefloat [doesn't flip]

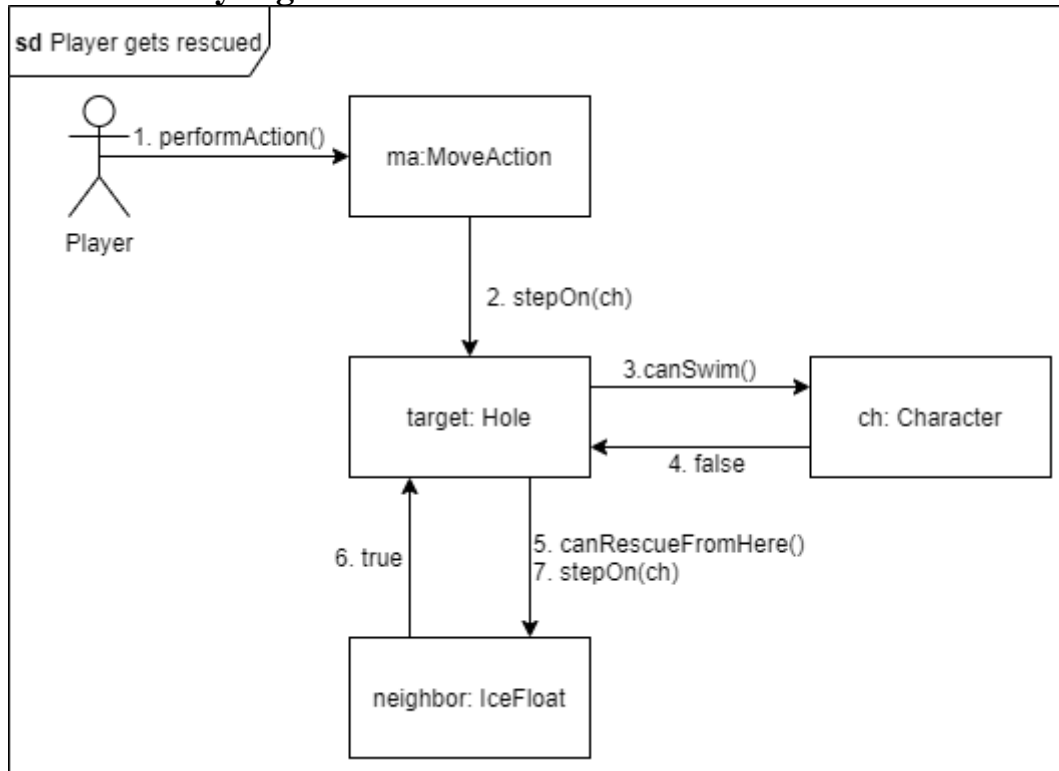
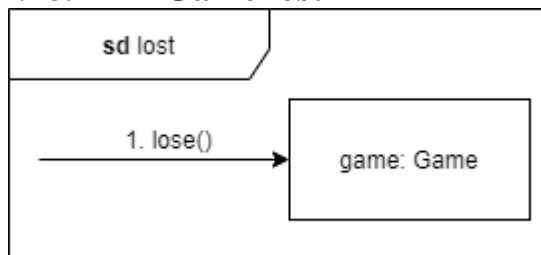
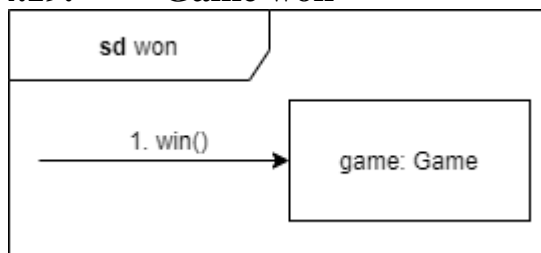


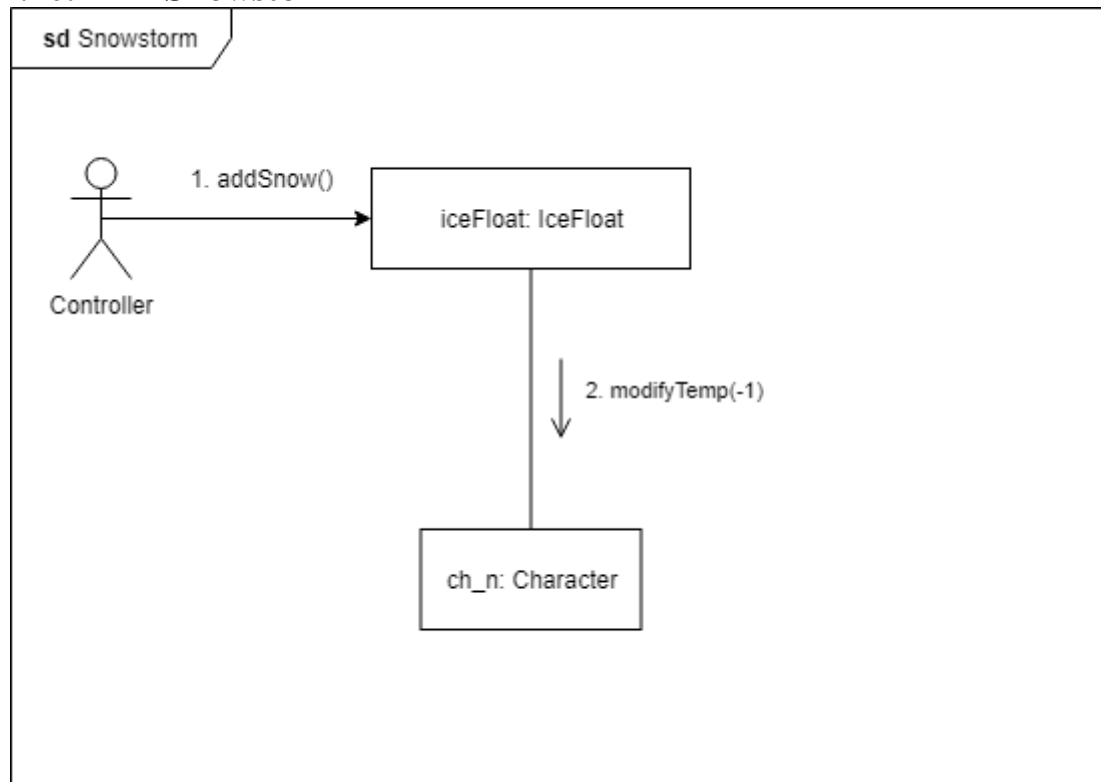
5.4.7. Player steps on unstable icefloat [flip]**5.4.8. Player picks up food**

5.4.9. Player picks up rope (/ diving suit / shovel)**5.4.10. Player shovels snow with shovel****5.4.11. Player shovels snow without shovel**

5.4.12. Player builds rocket**5.4.13. Player examines Icefloat****5.4.14. Player builds igloo**

5.4.15. Player drowns**5.4.16. Player swims back**

5.4.17. Player gets rescued**5.4.18. Game lost****5.4.19. Game won**

5.4.20. Snowstorm**5.4.21. Game generates icefloat**