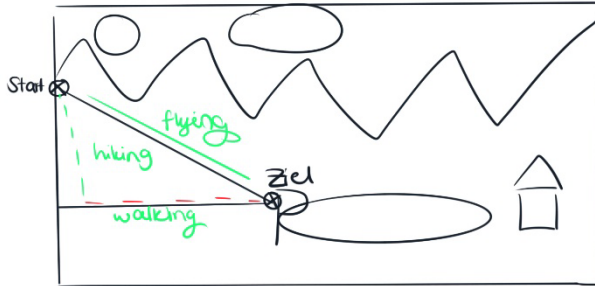


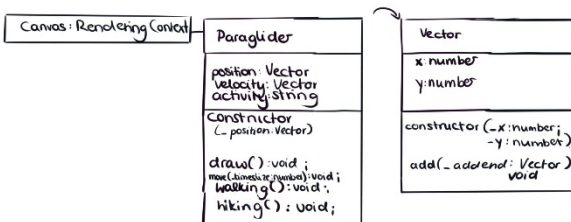
Scribble mit Idee hinter dem Konzept



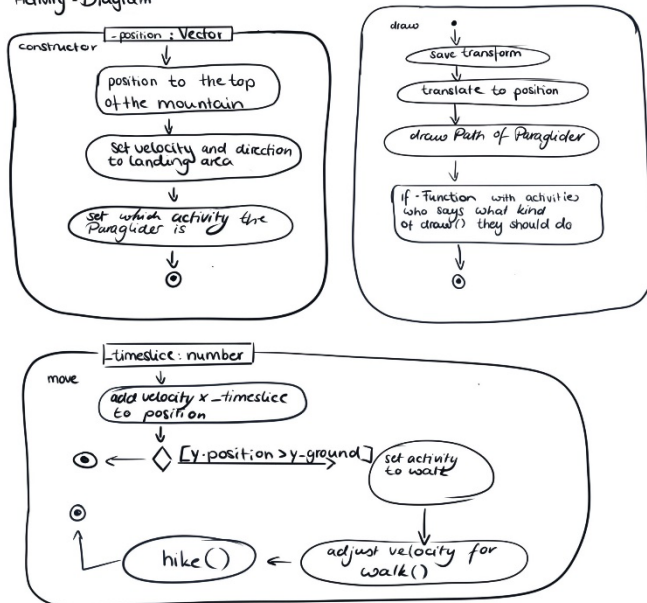
AD und Klassendiagramm

Paraglider : Class - Diagram

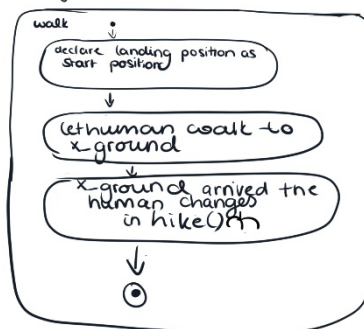
Was hat es?
Was kann es?
Was weiß es?



Activity - Diagram



Paraglider AD:



Paraglider AD:



y-ground = koordinaten
des Erdes

y+x koordinate

if (x-ground)
{ hiking() }

x-ground
 $x > 50 ; x < 10$

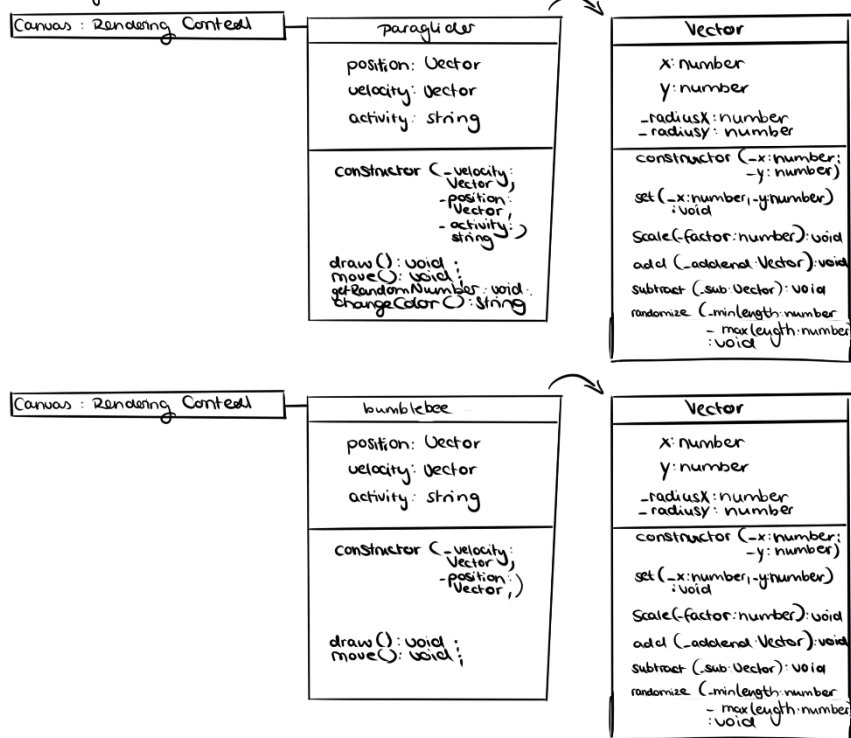
Notizen:

Das war meine erste Idee die ich auch weitestgehend umsetzen konnte. Ich habe den Code aufgeteilt in background.ts, einem main script.ts, bumblebee.ts, vector.ts und FlyingPerson.ts.

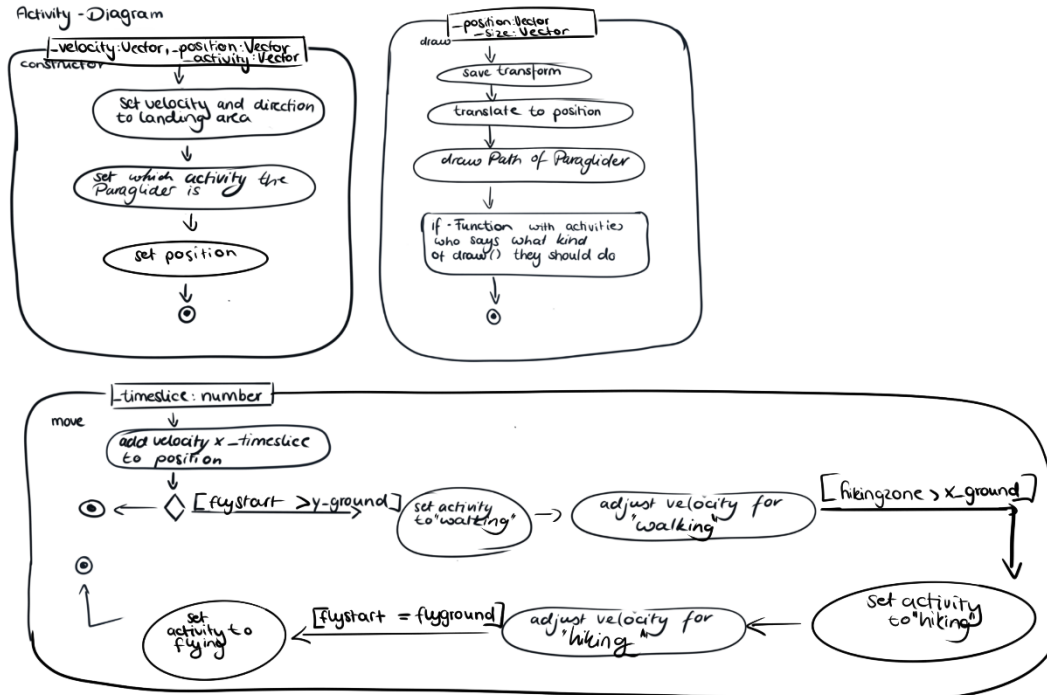
Das half mir zwar den Überblick zu behalten, hab aber extreme Probleme mit der Animation gehabt UND es haben sich ein paar Sachen aufgetan, die ich während dem schreiben noch geändert und habe daher mein Konzept überarbeitet.

Das Klassendiagramm habe ich gekürzt, da paraglider die Methoden fly, walk und hike nicht mehr benötigte, da draw() und move() alles inzwischen regeln. Das hat die Funktionen unnötig gemacht, daher sind die auskommentiert. Bei Vector habe ich ein paar Möglichkeiten hinzugefügt (W3C und Jirkas AsteroidenVideos haben da geholfen)

Class - Diagramm

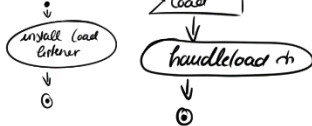


Flying Person .ts
Activity - Diagram

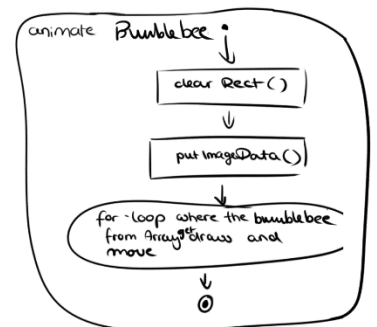
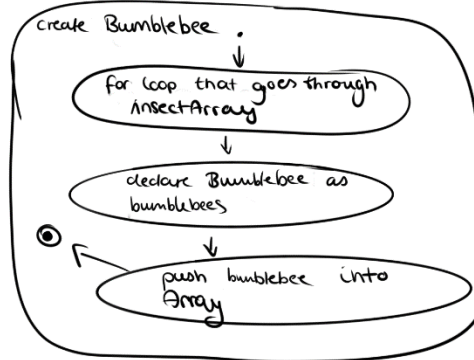
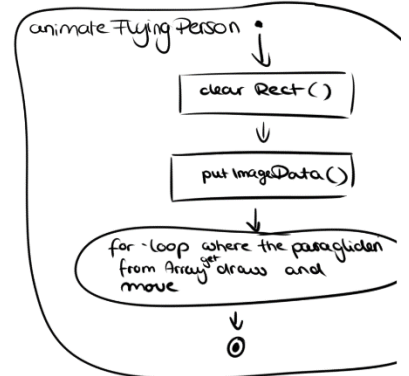
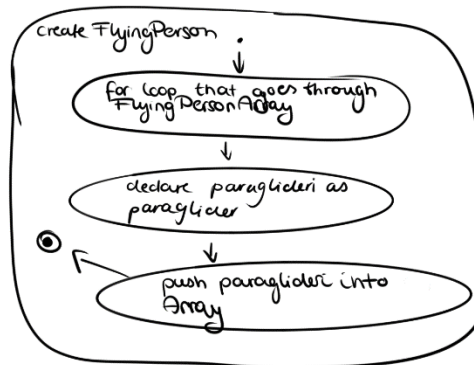
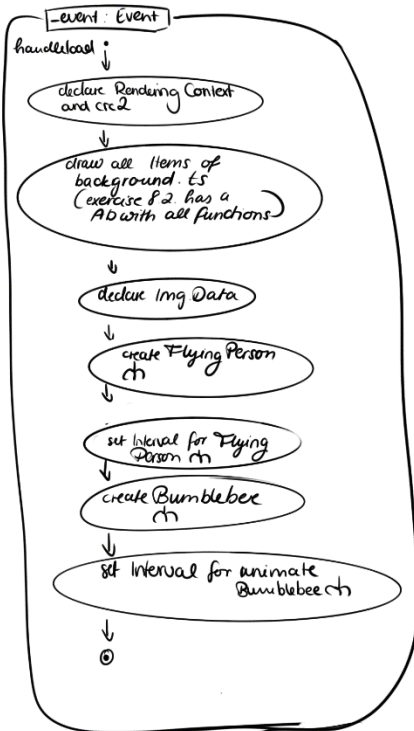


Hier hab ich das Zeugs nochmals so umgeschrieben, wie move() als methode abläuft. Wesentlich platzsparender und an sich funktioniert es eigentlich, nur bewegt sich der Spaß nicht, was meines Erachtens an setInterval im script.ts liegt. (siehe Kommentare im Code)

script ES / main



declare Array for
bumblebees and
paragliders
declare : landing zone, biking zone & flystart



In meinen Kommentaren im Code sieht man dann ja was ich noch herausgefunden habe, zum Teil läuft er, zum Teil ist er noch Überarbeitungsdrftig. Das waren so meine Gedankengänge zwischen rein. 😊