CanvasRenderingContext

Cloud

position: Vector;
velocity: Vector;

constructor (): void;
draw(): void;
move(_timeslice:number): void;

Bee

position: Vector;
velocity: Vector;

constructor (): void;
draw(): void;
move(_timeslice:number): void;

Flower

type: string;
color: string;
position: Vector;
size: number;
velocity: Vector;

constructor (): void;
drawFlowers(): void;
move(_timeslice:number): void;

Flower

type: string;
color: string;
size: number;
position: Vector;
velocity: Vector;

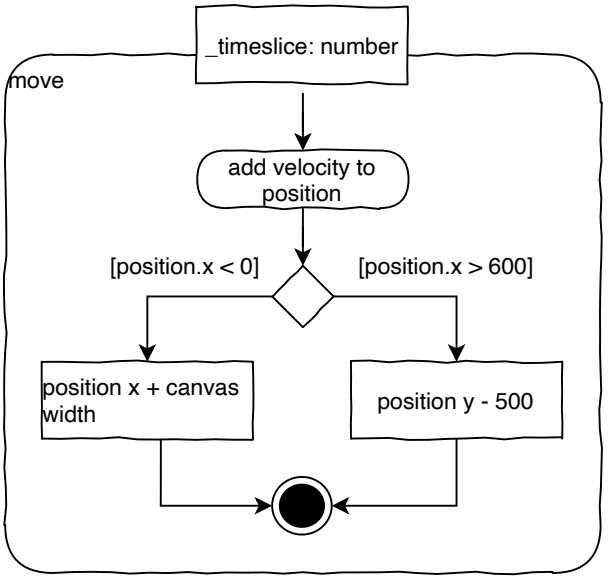
constructor (): void;
draw(): void;
move(_timeslice:number): void;

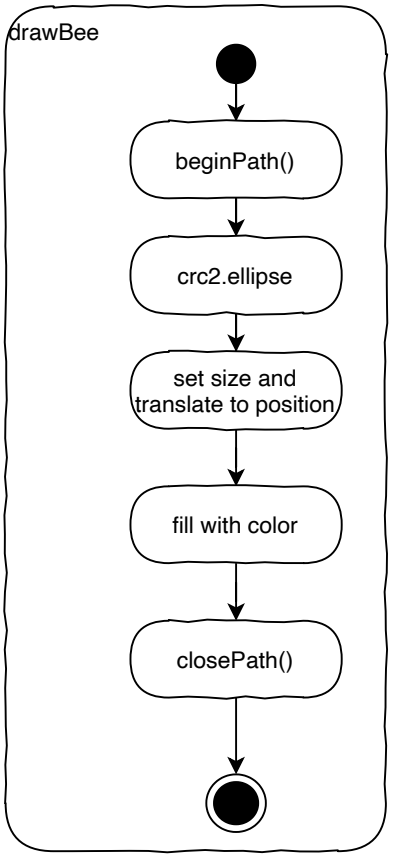
Vector

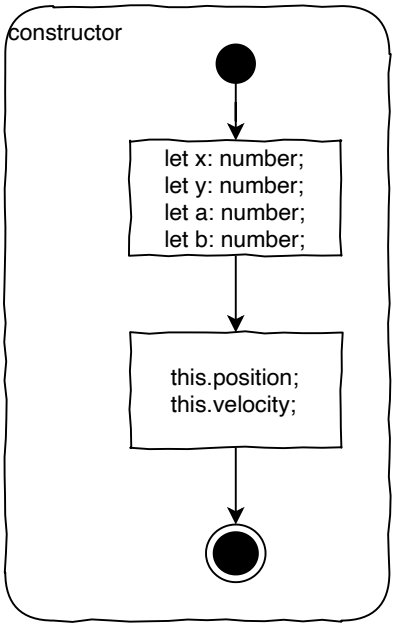
position: Vector;
velocity: Vector;

constructor (): void;
draw(): void;
move(_timeslice:number): void;

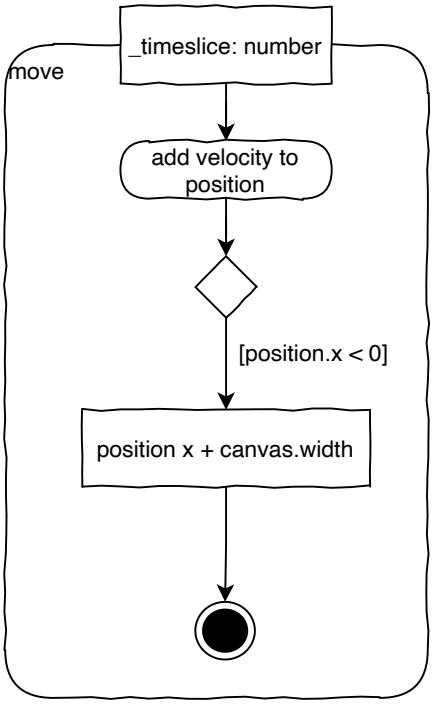
Activity Diagram Bee

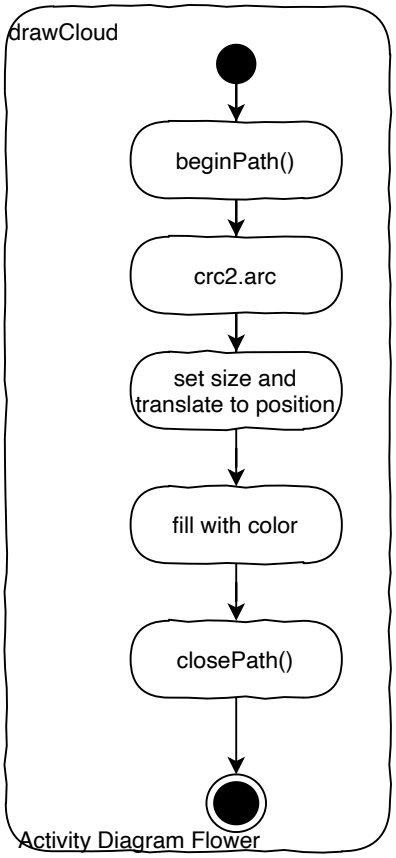


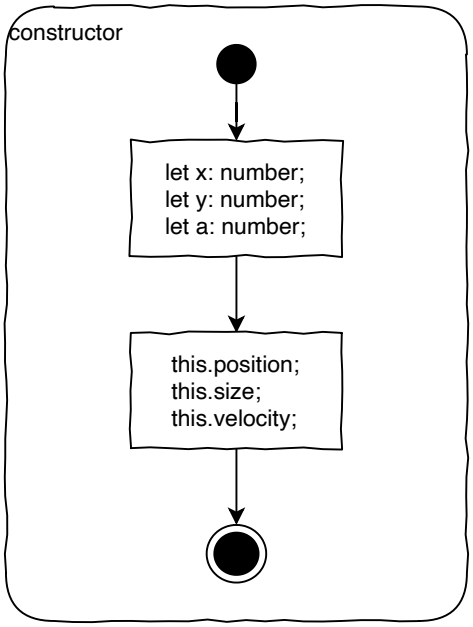




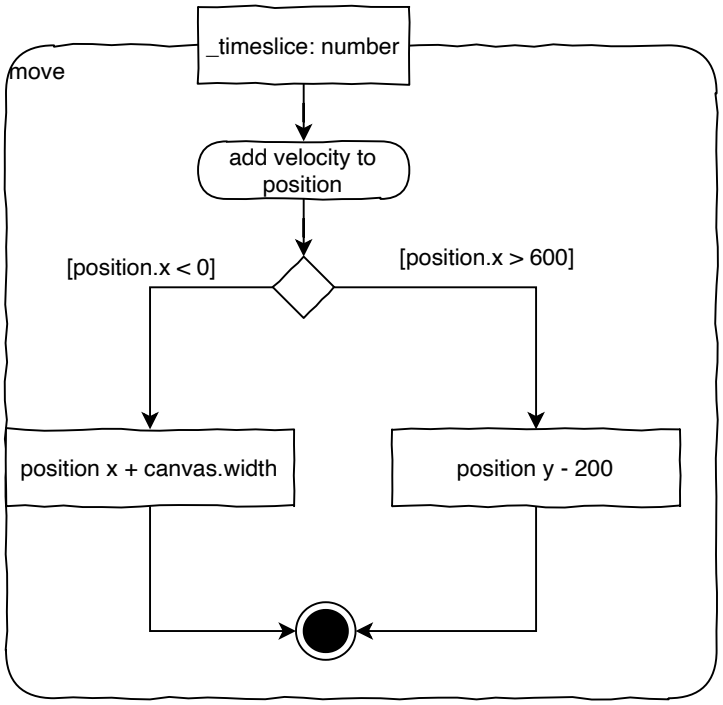
Activity Diagram Cloud

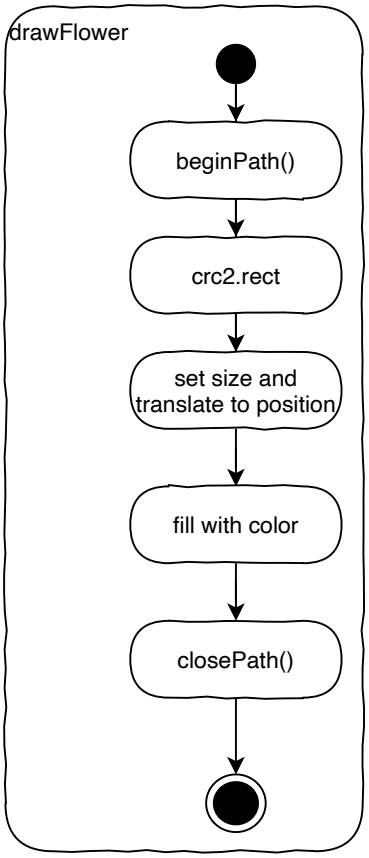


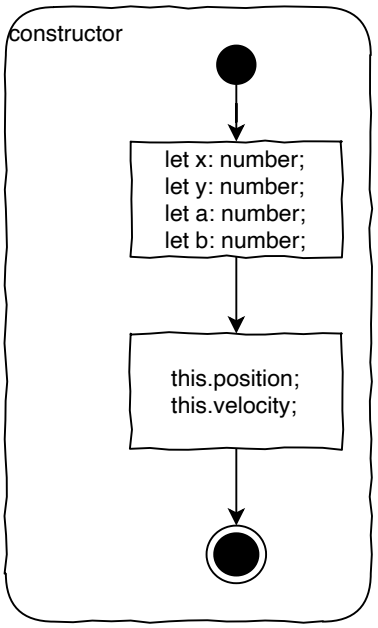




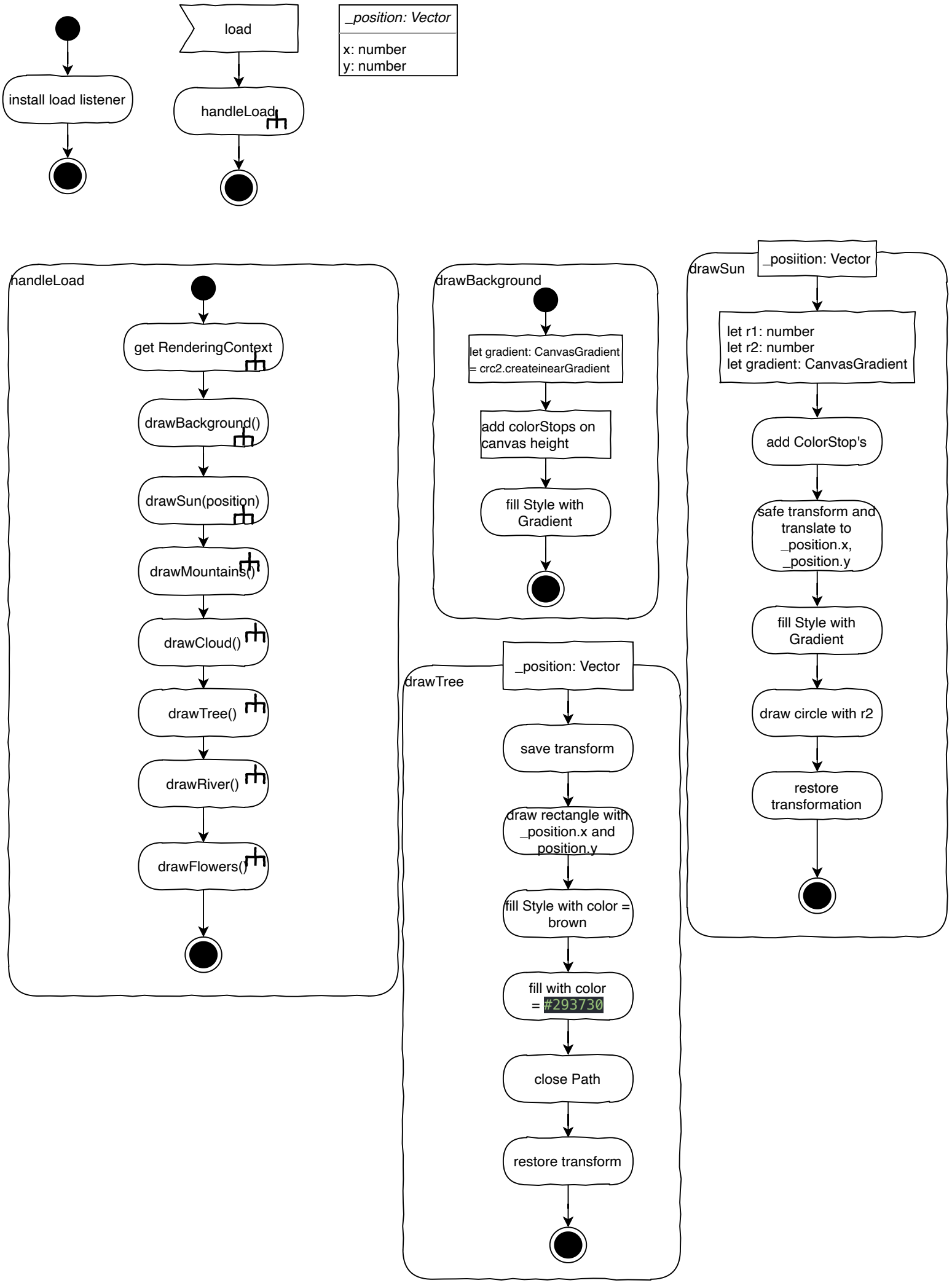
Activity Diagram Flower

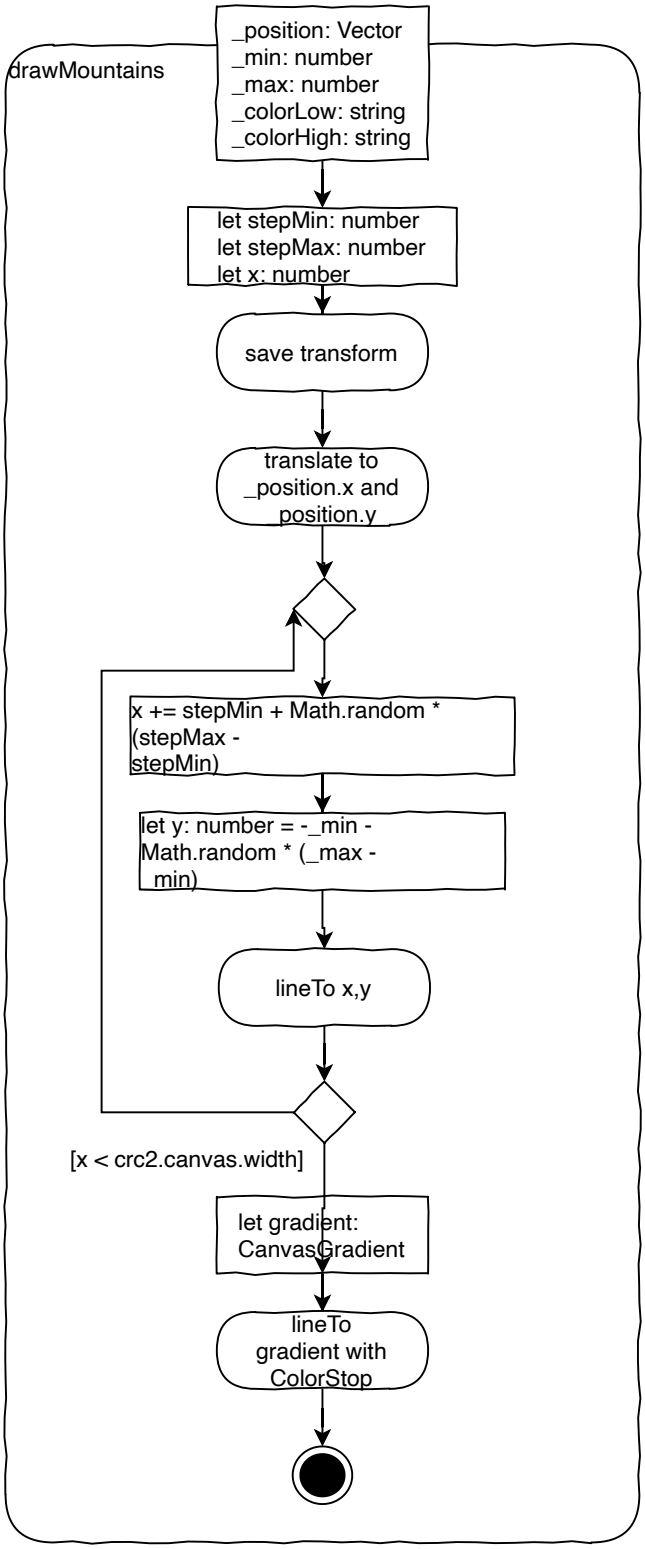


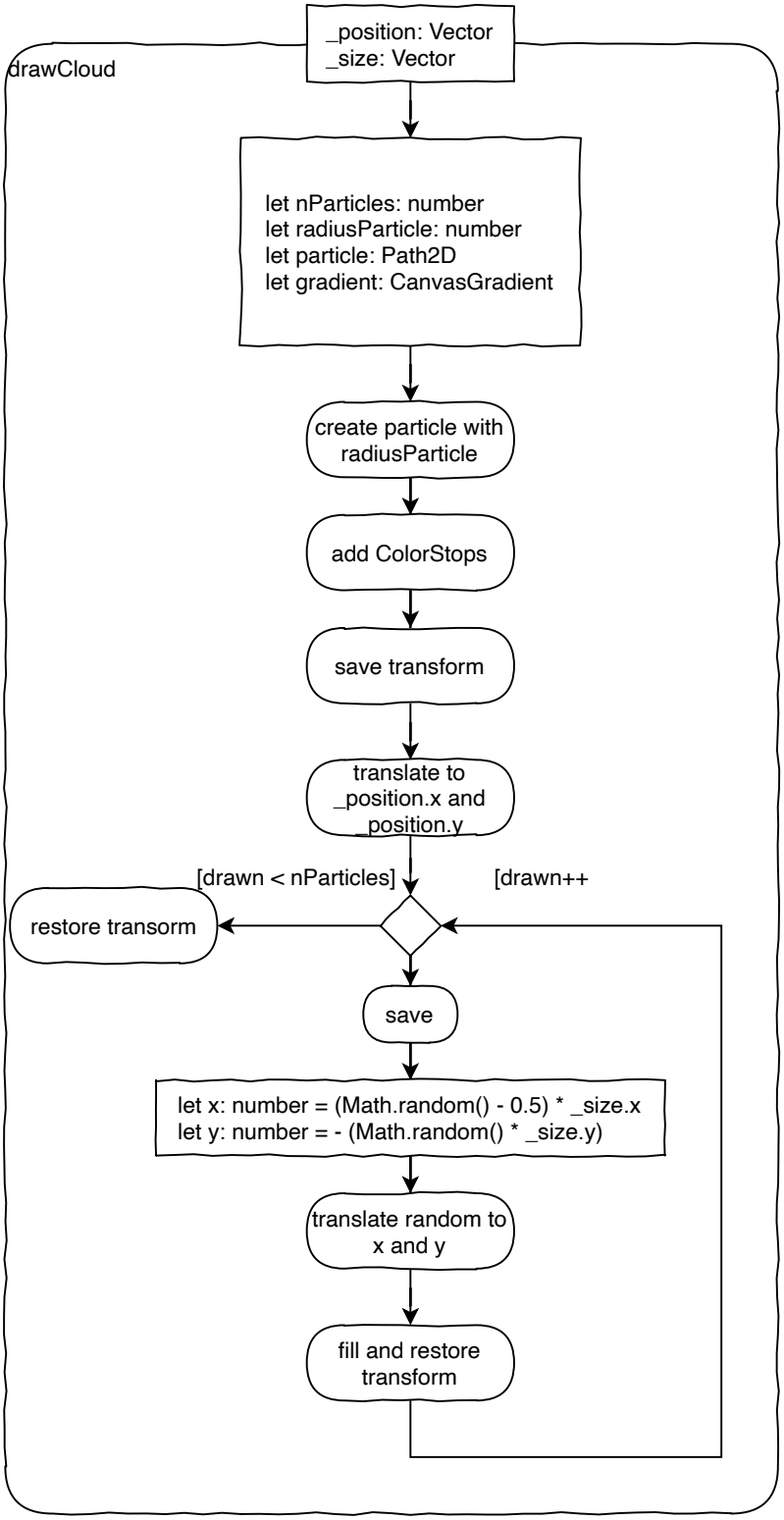


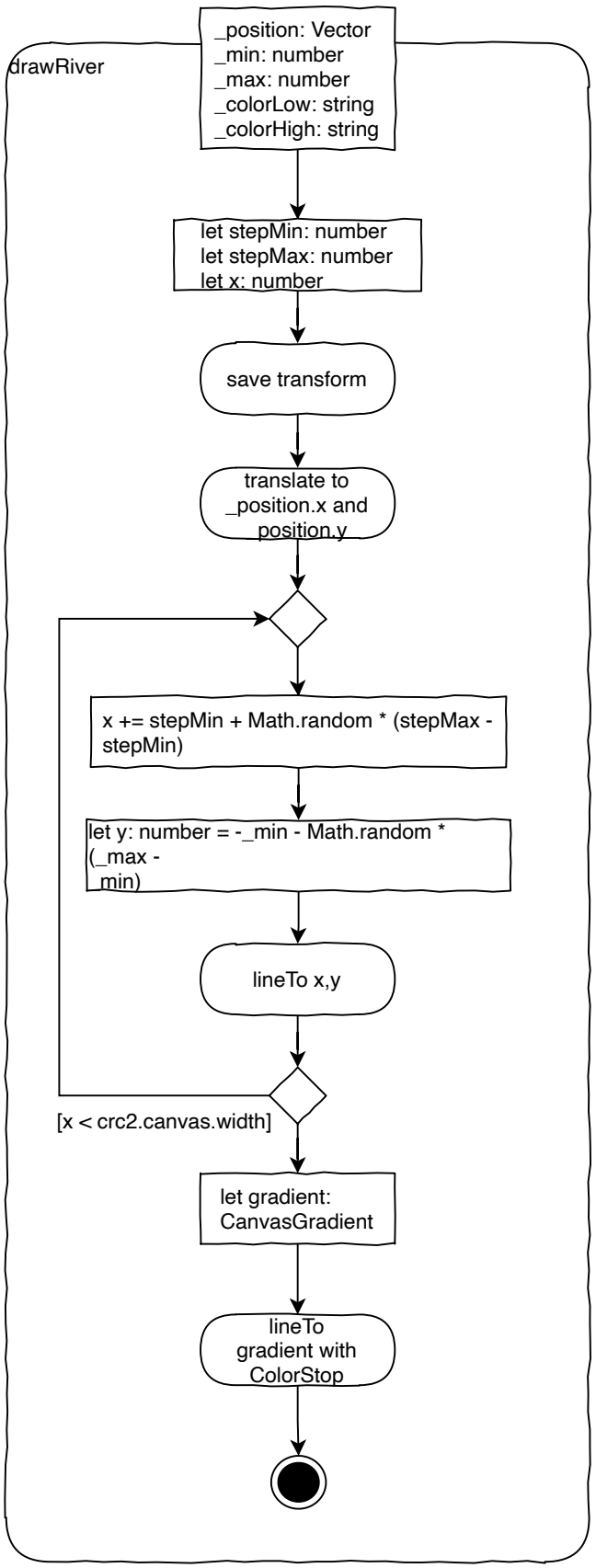


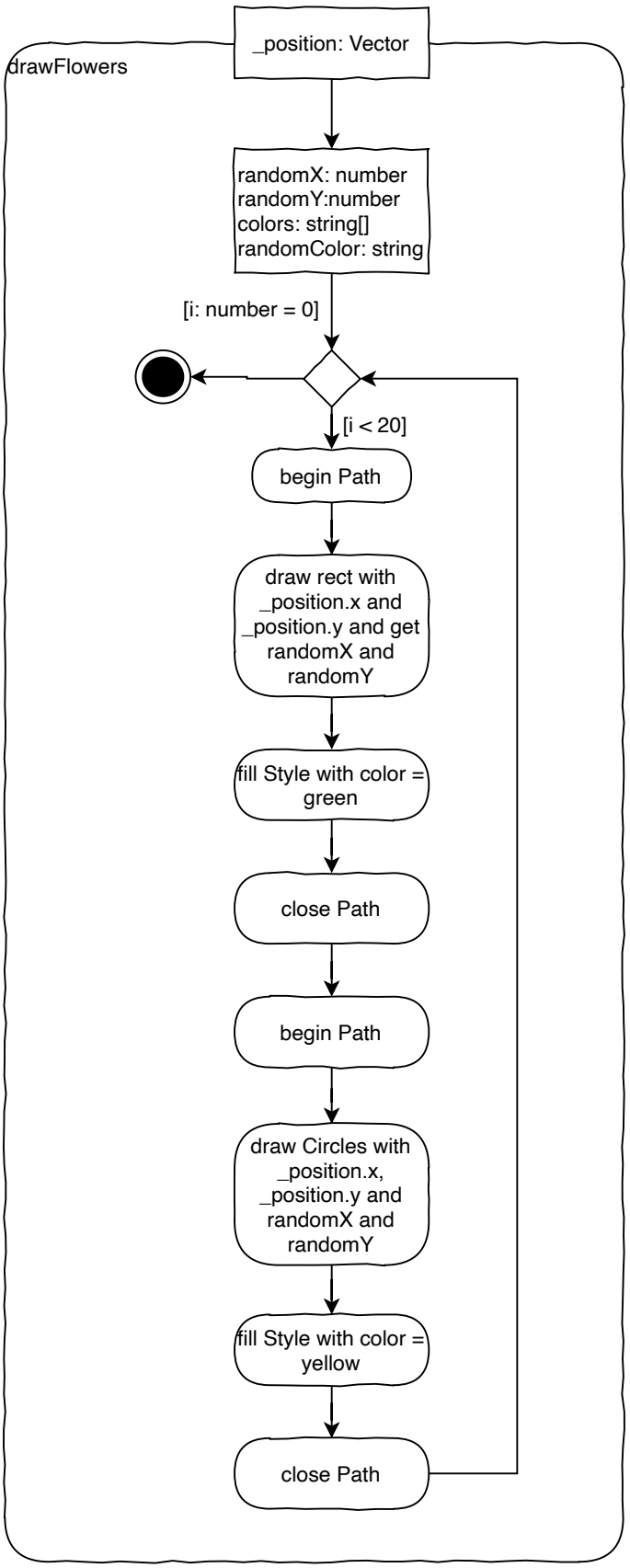
Acitivity Diagram, Rebecca Räschke, L08.2_Blumenwiese




drawMountains
_position: Vector
_min: number
_max: number
_colorLow: string
_colorHigh: string
let stepMin: number
let stepMax: number
let x: number
save transform
translate to
_position.x and
position.y
x += stepMin + Math.random *
(stepMax -
stepMin)
let y: number = -_min -
Math.random * (_max -
min)
lineTo x,y
[x < crc2.canvas.width]
let gradient:
CanvasGradient
lineTo
gradient with
ColorStop


drawCloud
_position: Vector
_size: Vector
let nParticles: number
let radiusParticle: number
let particle: Path2D
let gradient: CanvasGradient
create particle with
radiusParticle
add ColorStops
save transform
translate to
_position.x and
_position.y
[drawn < nParticles]
[drawn++
restore transorm
save
let x: number = (Math.random() - 0.5) * _size.x
let y: number = - (Math.random() * _size.y)
translate random to
x and y
fill and restore
transform


drawRiver
_position: Vector
_min: number
_max: number
_colorLow: string
_colorHigh: string
let stepMin: number
let stepMax: number
let x: number
save transform
translate to
_position.x and
position.y
x += stepMin + Math.random * (stepMax -
stepMin)
let y: number = -_min - Math.random *
(_max -
_min)
lineTo x,y
[x < crc2.canvas.width]
let gradient:
CanvasGradient
lineTo
gradient with
ColorStop


drawFlowers
_position: Vector
randomX: number
randomY:number
colors: string[]
randomColor: string
[i: number = 0]
[i < 20]
begin Path
draw rect with
_position.x and
_position.y and get
randomX and
randomY
fill Style with color =
green
close Path
begin Path
draw Circles with
_position.x,
_position.y and
randomX and
randomY
fill Style with color =
yellow
close Path