

A-VOID

Control a planet and avoid asteroids. The game ends if the planet runs out of lives and it is won if it has survived for a set time.

MVP (DOM - CANVAS)

- Moving the planet in all directions,
- Checking for collisions with asteroids.
- Set a timer.

BACKLOG

- Enemies:
 - Comet: goes directly for the player.
 - Great Asteroid: A bigger asteroid.
- PowerUps (should get stored and be called when needed.)
 - Watercloud: Life Up.
 - Nova: destroy some asteroids.

DATA STRUCTURE

Game.js:

```
Game(){
  this.canvas;
  this.ctx;
}
Game.prototype.startGame(){
}
Game.prototype.finishGame(){
}
StartLoop(){
  - setTimer()
  - New Planet
  - Define Loop {
    - Clear canvas(),
    - Update canvas(),
    - Draw canvas()
    - If timer > 0 || lives > 0
      - Request animation frame, loop.
  }
}
```

```
    Request animation frame, Loop()
}

Game.prototype.updateAll(){
}
Game.prototype.clearAll(){
}
```

Planet.js:

```
Character(){
  this.x;
  this.y;
  this.speed;
  this.lives;
  this.size;
  this.canvas;
  this.ctx;
}

Planet.prototype.move(){
}
Planet.prototype.getDamage(){
}
Planet.prototype.hittingAsteroid(asteroid){
}
Planet.prototype.update(){
}
Planet.prototype.draw(){
}
```

Asteroid.js:

```
Asteroid (){
  this.x;
  this.y;
  this.size;
  this.canvas;
  this.ctx;
}
```

```
Asteroid.prototype.movement(){  
}  
Asteroid.prototype.update(){  
}  
Asteroid.prototype.draw(){  
}
```

STATES AND THEIR TRANSITIONS

```
- buildMainMenuScreen(){  
    -destroyWinScreen(if ){  
    }  
  
}  
- buildGameScreen(){  
    -destroyMainMenuScreen(){  
    }  
    -destroyGameOverScreen(if){  
    }  
  
}  
- buildGameOverScreen(){  
    -destroyGameScreen(){  
    }  
  
}  
- buildWinScreen(){  
    -destroyGameScreen(){  
    }  
  
}
```

Task

Links

Git

URIs for the project repo and deploy

[Link Repo](https://github.com/EricCapdevila/A-Void)

[Link Deploy](http://github.com)