



# HKUST CSE FYP 2021-22

## AR3: Steel Of Stalin

(A Turn Based Strategy Game Focusing on Management of Army Logistics)

Presenters:

LEUNG Ho Man Max

LO Yuk Fai (Marcus)

HUI Nathan

KONSTANTINO Hubert Aditya

Advisor:

Prof. Sunil ARYA



01

OVERVIEW

02

Implementation

03

Live Demo



# 01

## OVERVIEW



# Introduction

Why a strategy game?



# Objectives

1. To create an attractive strategy game
2. To develop multiplayer features
3. To implement the above goals into a smooth game experience

# Sources of Inspiration



Advanced Daisenryaku



Starcraft 2



Diplomacy



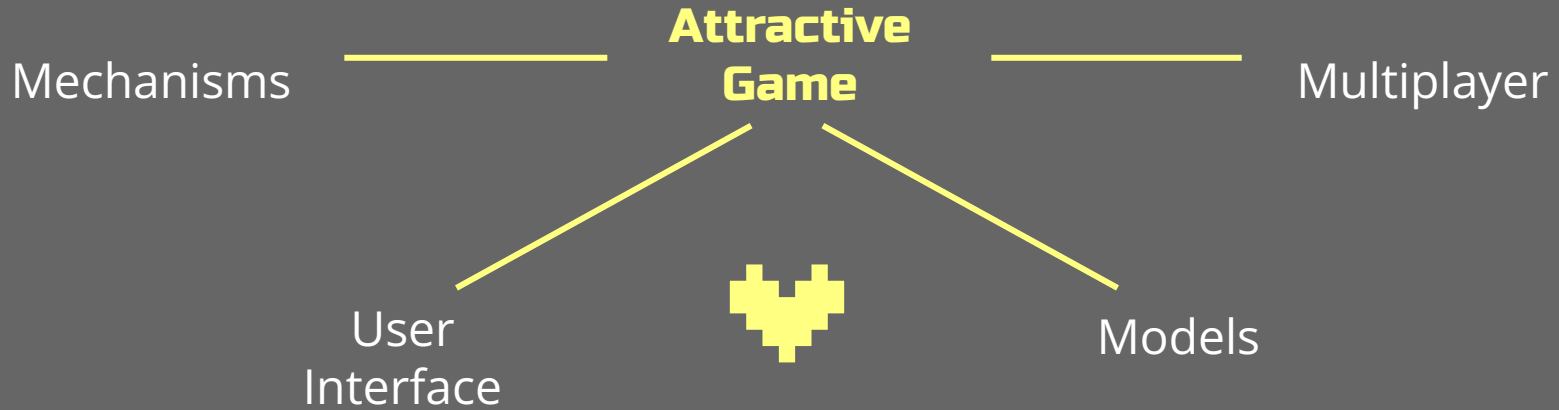
Hearts Of Iron IV

# 02

## Implementation



# Project Overview



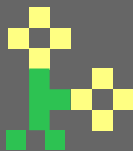


# Project Overview

**Mechanisms**



**User  
Interface**



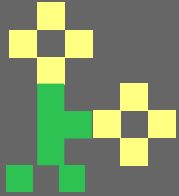
**Models**



**Multiplayer**



# Definitions of in-game terms



## Map

Where the battle takes place, contains hexagonal tiles with different terrains

## Terrain

An abstraction of real-world terrains represented by 3D models

## Prop

Anything on the battle scene (units, buildings, tiles...)

## Unit

Like a chess piece, but can be trained each round by consuming resources

## Round

Where players make decisions by assigning commands to their units

## Command

An action which the player can do, or can assign to his unit



# Mechanisms

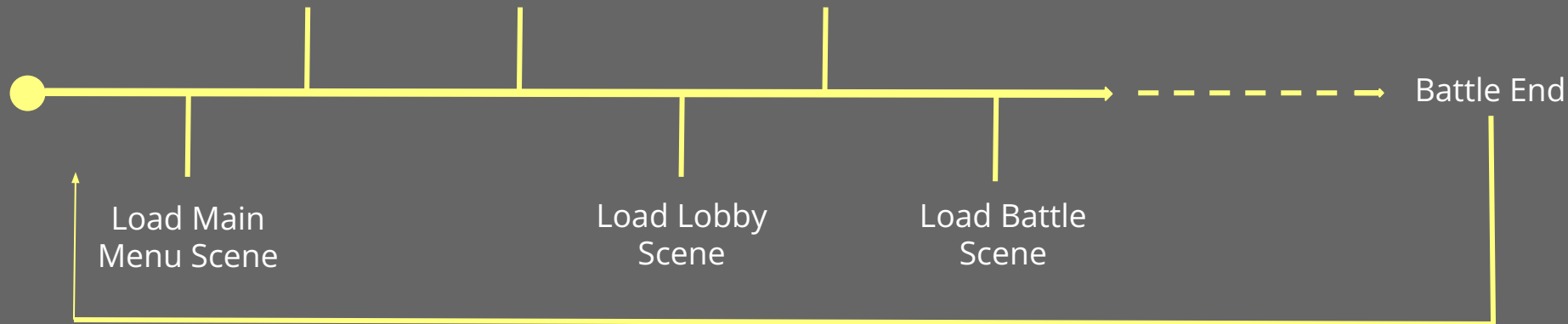
## Game Flow

Backend

Create Script  
Objects

Load Game  
Assets

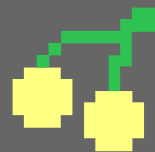
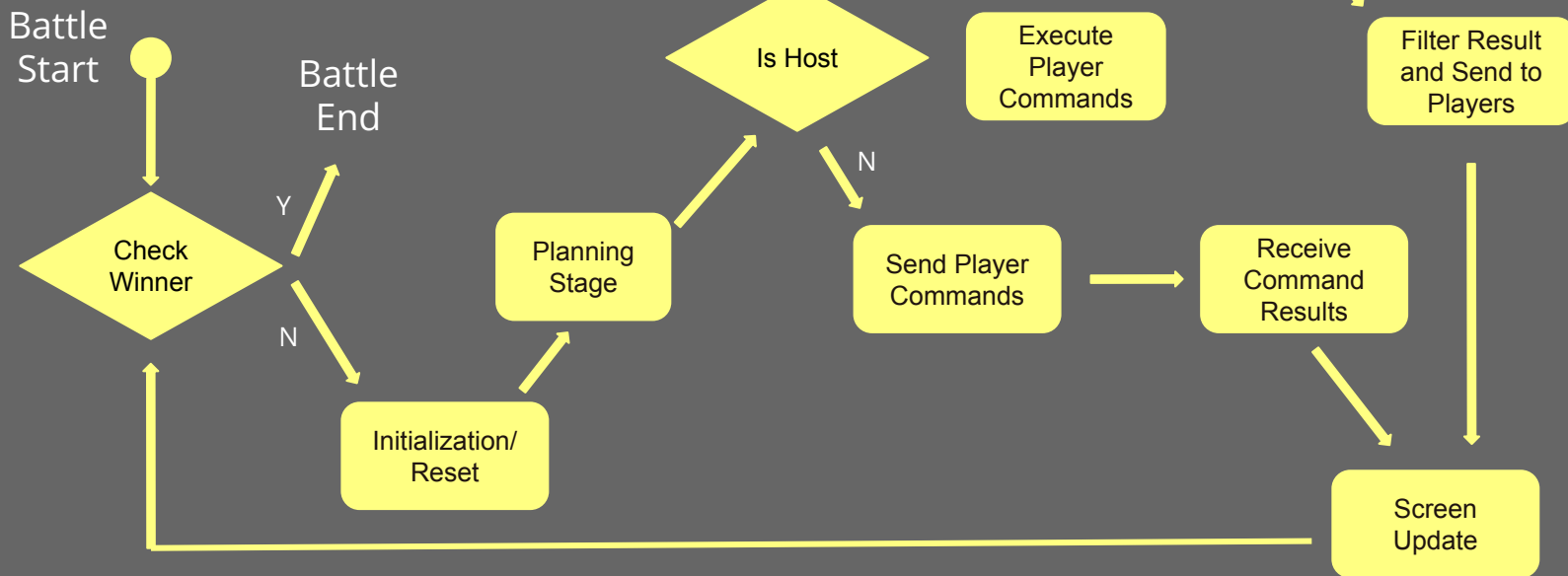
Load Battle



Frontend

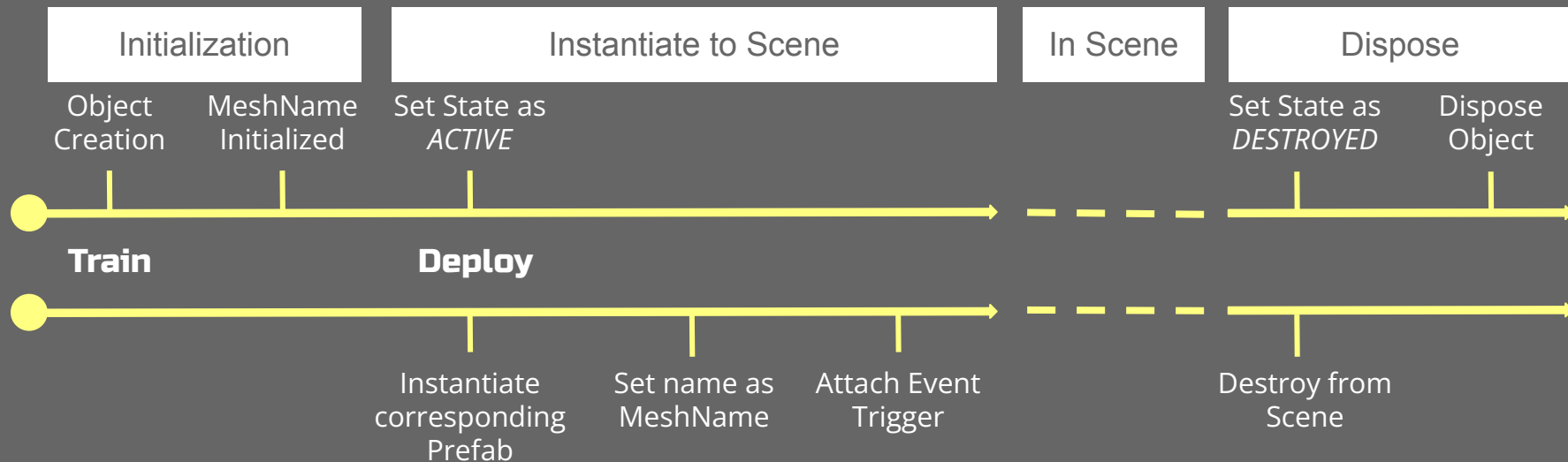
# Mechanisms

## Round Flow



# Mechanisms

## Unit Lifecycle



# Mechanisms

Random map generation



Cities generation

**STEP 3**

Streams generation

**STEP 2**

Terrain generation

**STEP 1**

# Mechanisms

## Random map generation

### Terrain generation

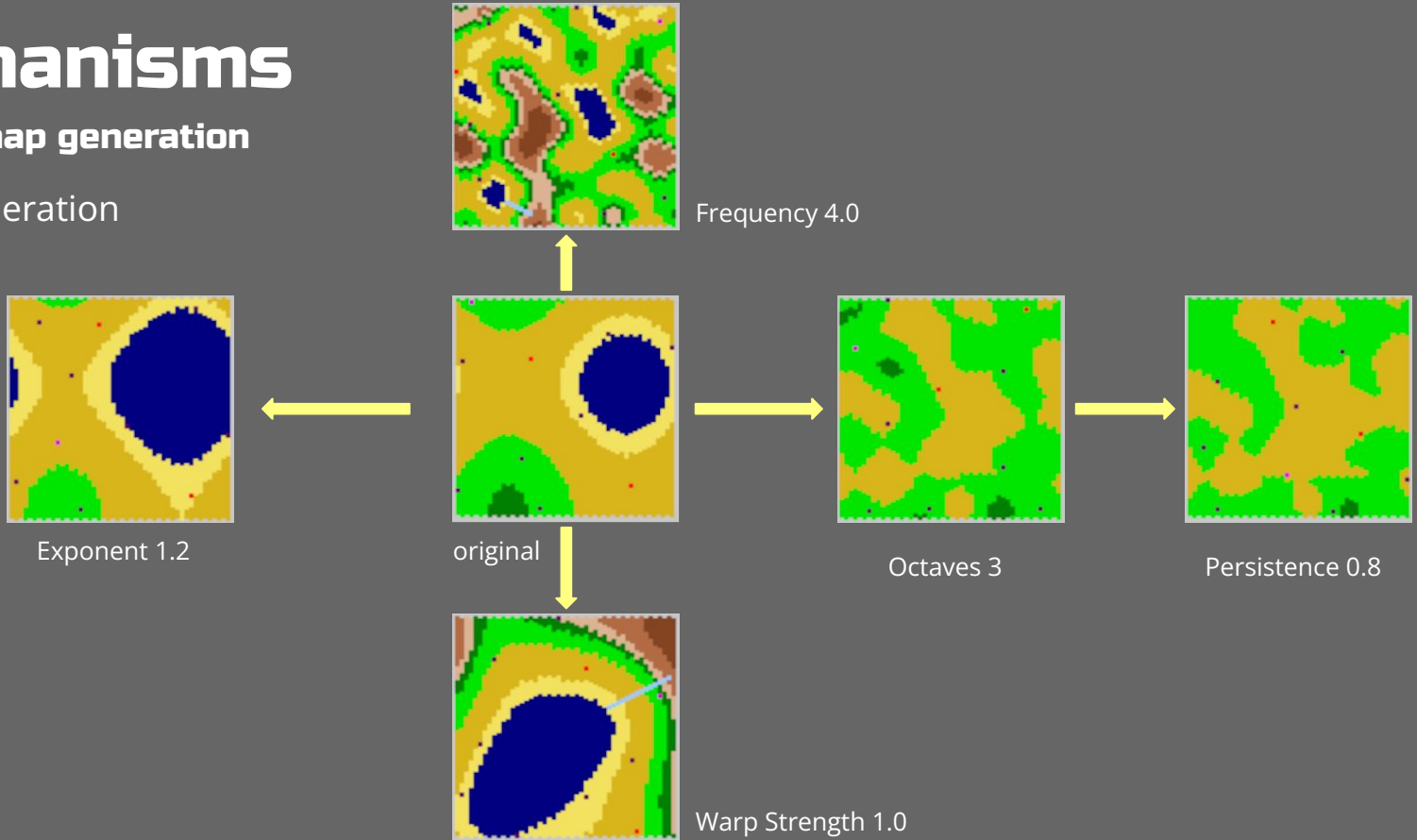
- 2 Perlin noise maps: “Height” and “Humidity”

Parameters	Controls
Frequency	Density of the patterns
Octaves	Details of the patterns
Persistence	Attenuation of octaves
Exponent	Flatness of the patterns
Warp Strength	Degree of distortion of the patterns
Offset	Origin of the sampled area

# Mechanisms

## Random map generation

Terrain generation





# Mechanisms

## Random map generation

Streams generation



STEP 1

STEP 2

STEP 3

STEP 4

Pick a water source

Pick a destination

Weighted A\* algorithm

Overwrite



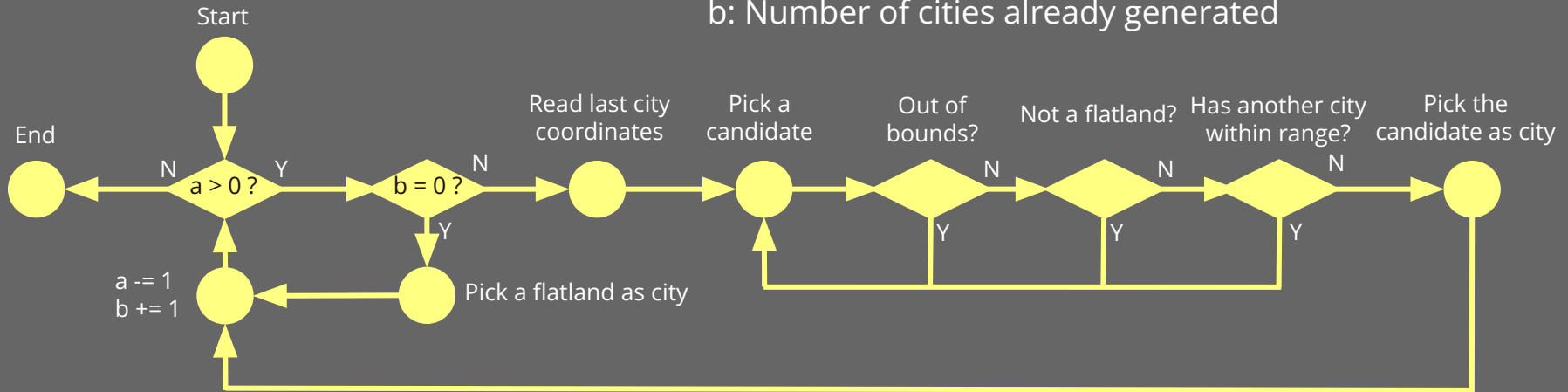
# Mechanisms

## Random map generation

### Cities generation

- Poisson Disk algorithm

a: Number of cities to be generated  
b: Number of cities already generated



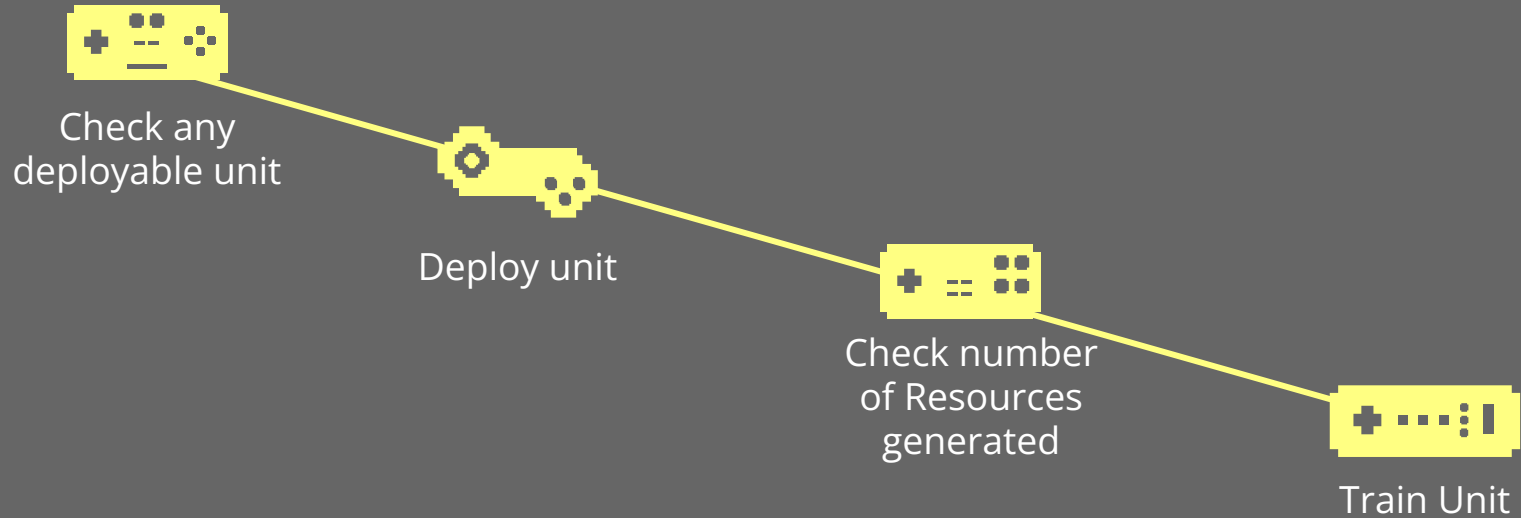
# Mechanisms

## Bot Algorithm



# Mechanisms

## Training



# Mechanisms

## Construction

### Building Inside City Range



STEP 1

Check Missing  
type of  
building



STEP 2

Construct  
the building

### Outpost



STEP 1

Get the  
middle point  
of 2 cities



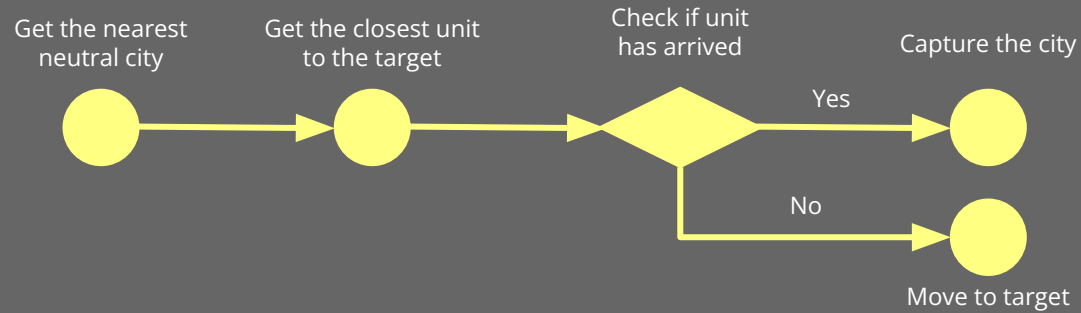
STEP 2

Construct  
outpost

# Mechanisms

## Movement

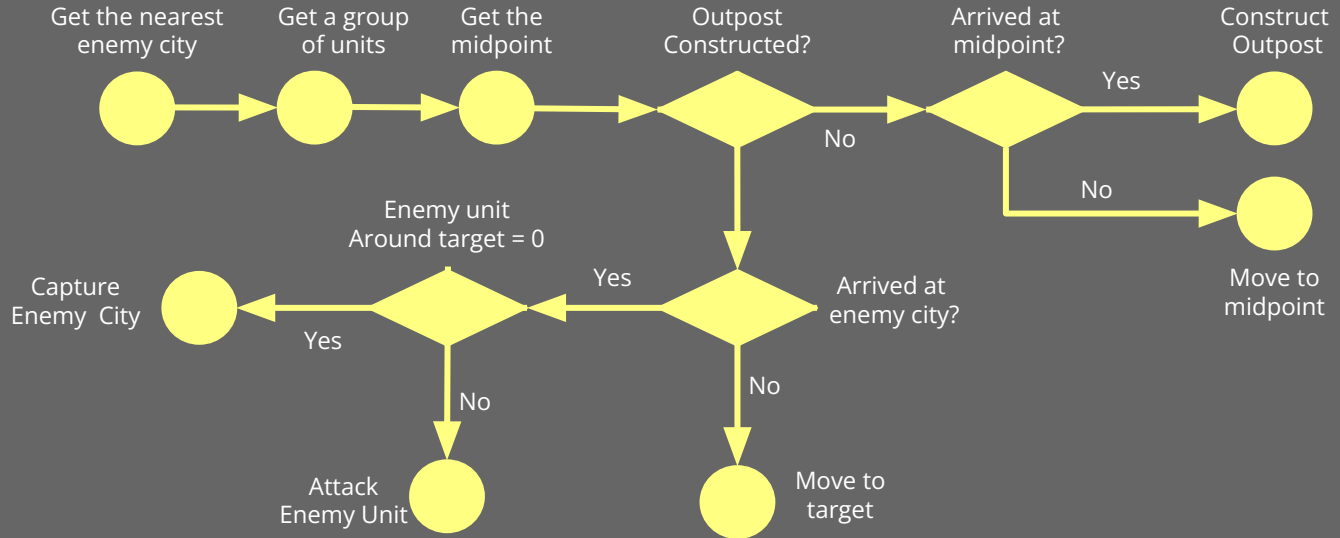
### Capture Neutral City



# Mechanisms

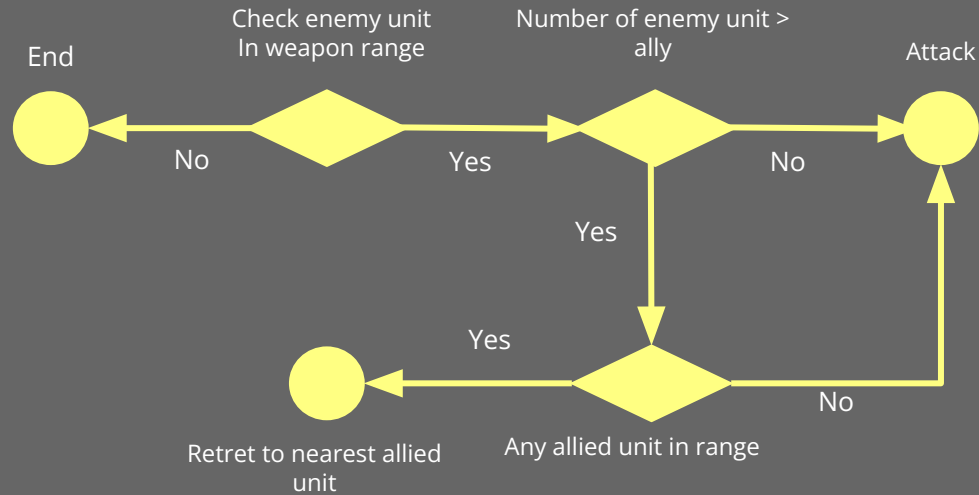
## Movement

### Capture Enemy City



# Mechanisms

## Combat





# User Interface

## Camera Controlling System

### Camera

Controlling the View of Player in the Game World

Focus on an Area  
during Rotation

Point to the  
Game Board Only

Smooth  
Movement

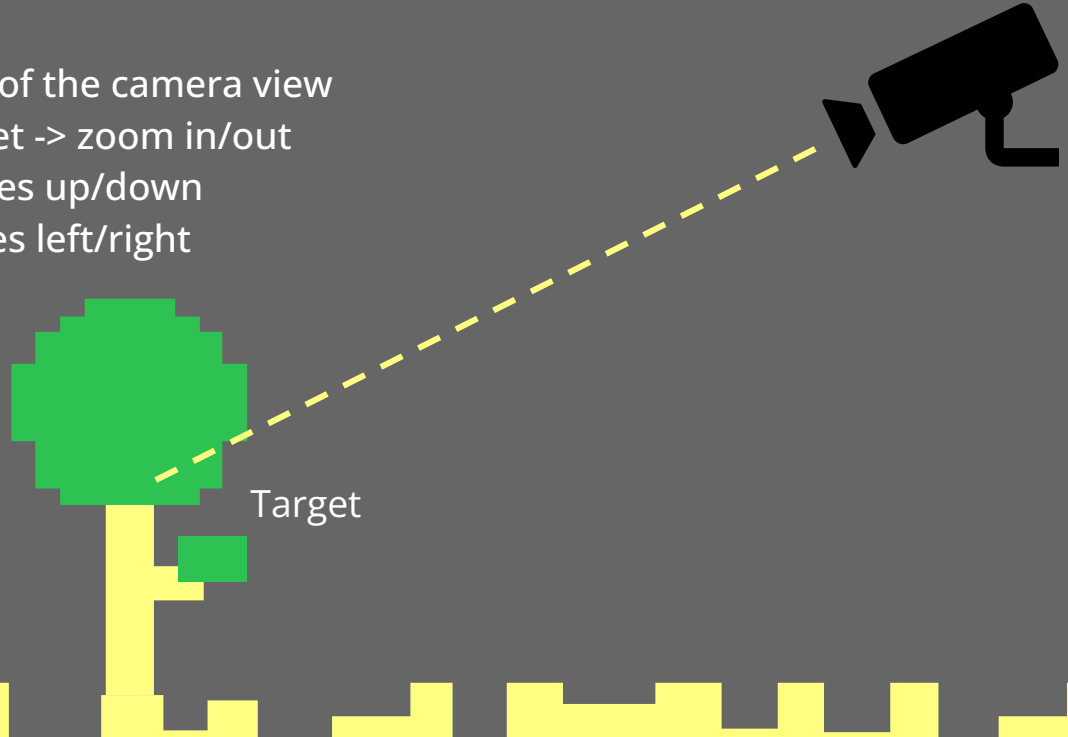


# User Interface

## Camera Controlling System

Variables that control the camera

- Position of Target -> middle of the camera view
- Length from camera to target -> zoom in/out
- Angle of Depression -> rotates up/down
- Horizontal Rotation -> rotates left/right



# User Interface

## Camera Controlling System

Variables that control the camera

- Position of Target -> middle of the camera view
  - $x : 0 - MAP\_WIDTH \times CONSTANT$
  - $y : 0 - MAP\_HEIGHT \times CONSTANT$
  - $z : 0$
- Length from camera to target -> zoom in/out
  - 40 - 400
- Angle of Depression -> rotates up/down
  - $10^{\circ} - 90^{\circ}$
- Horizontal Rotation -> rotates left/right
  - No limitation

# User Interface

## Camera Controlling System

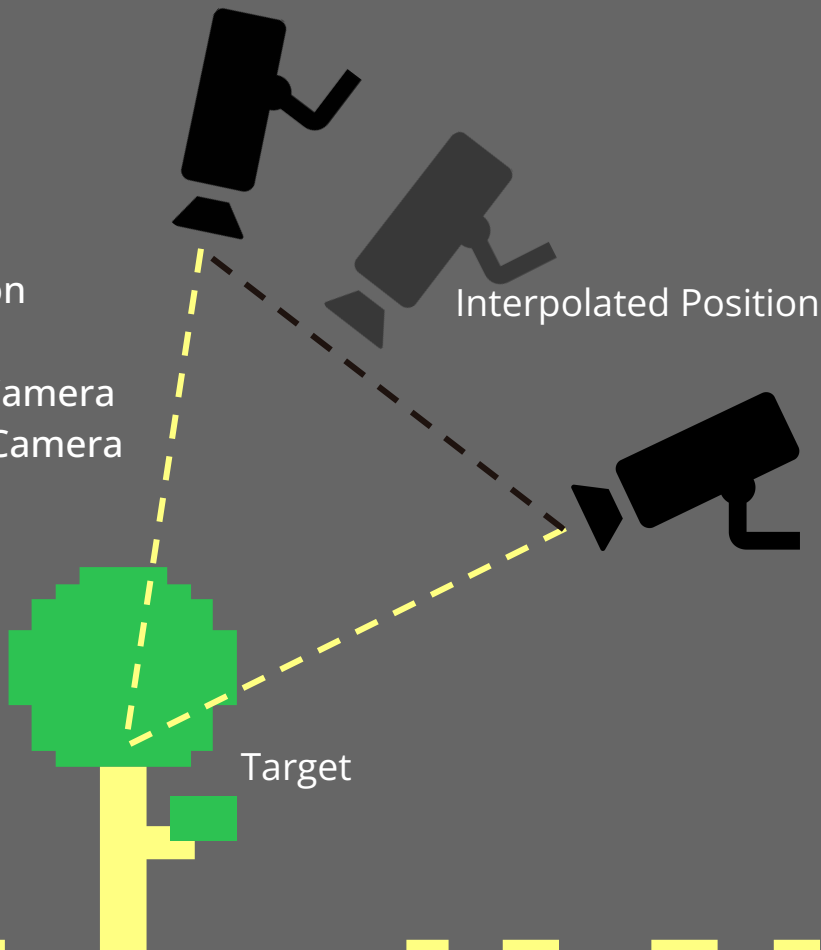
1st Attempt: Linear Interpolation on

- Position of Target
- Local Position of Camera
- Local Rotation of Camera



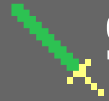
## PROBLEM

The camera zoom in and zoom out during rotation



# User Interface

## Camera Controlling System



## SOLUTION



Aware of the zoom distance

Variables that control the camera

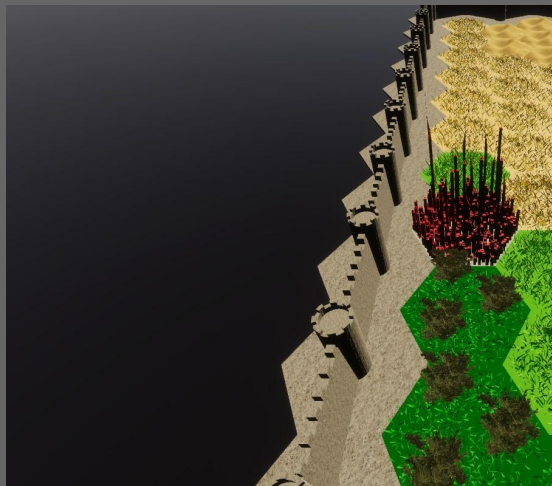
- Position of Target
- Length from camera to target
- Angle of Depression
- Horizontal Rotation

Perform Smoothing on

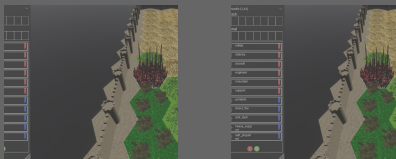
Separately

# User Interface

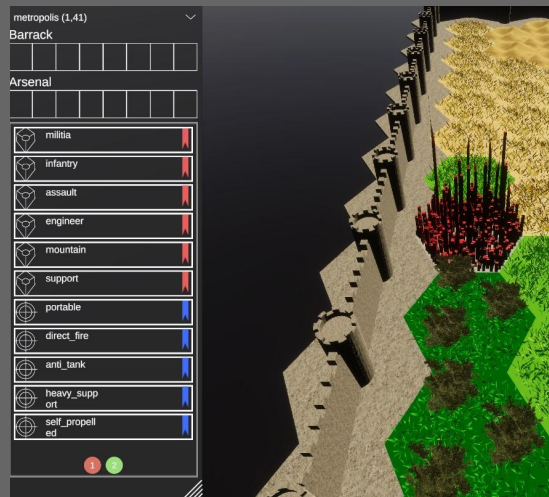
## UI Animation - Tweening



Key Frame



Frames Generated in Between



Key Frame

# User Interface

## PropEventTrigger

- Custom Event Trigger that supports Toggle On/Off

Subscribe/Unsubscribe  
To event trigger



Examine unit  
status

**OR**

Order this  
unit to attack

**OR**

Attack this  
unit

**OR**

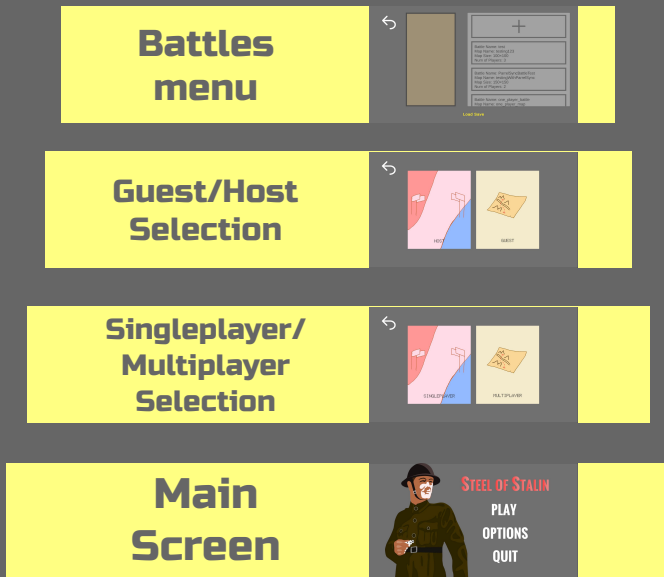
others...

# User Interface

## Menu Navigation

Main Menu: Stack Approach

The GameObject of pages are stored in a stack for navigation purposes





# User Interface

## Menu Navigation

In Battle: Event-Driven Approach

Panels are opened or closed independently according to Pointer OnClick events

**Command  
Panel**



**Resources  
Panel**



120399



104377

**Unit  
Panel**

assault			
Strength	600/600		
Morale	100/100		
Carrying			
Supplies	300/600		
Cartridges	20/40		
Shells	0/0		
Fuel	0/0		
Weapons			
	Soft	Hard	Destruct
carbine	80	8	25
Speed	2		
Recon	6		
Communication	5		

**Train  
Panel**



**Deploy  
Panel**



# Models

## Texture

Define the appearances of surfaces of 3D objects

Types of maps	What is stored	Usage
Bump Map	Height details	Create illusions of “bumps” on surfaces
Normal Map	Surface normals	
Roughness Map	Degree of light scattering	Emulate roughness of real-life objects
Cube Map	Reflections on environments	Skybox




# Models

## Optimization

Object details  ⇒ Polygon count



### PROBLEM

GPU load 

⇒ Framerate 

⇒ Player experience 



### SOLUTION

- Decimate
- Level of Details

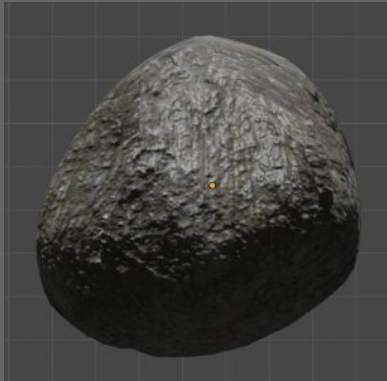


# Models

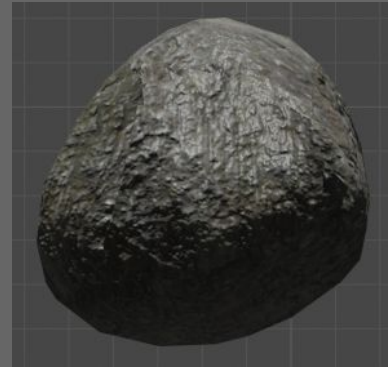
## Optimization

Decimate

- Reduce polygon count
- Does not have great difference in appearance



3072 polygons



306 polygons

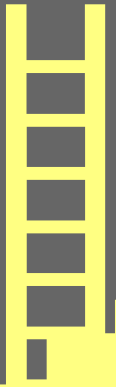
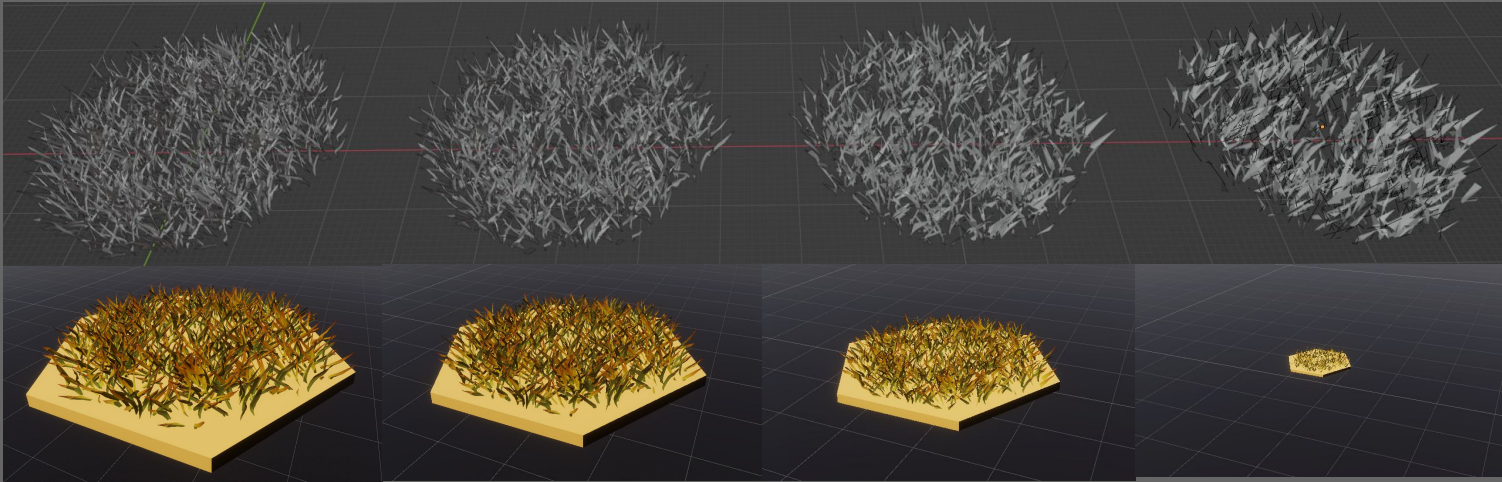


# Models

## Optimization

### Level of Details (LOD)

- Reduce polygon count only when the camera is far away
- Does not have great difference in appearance in camera perspective



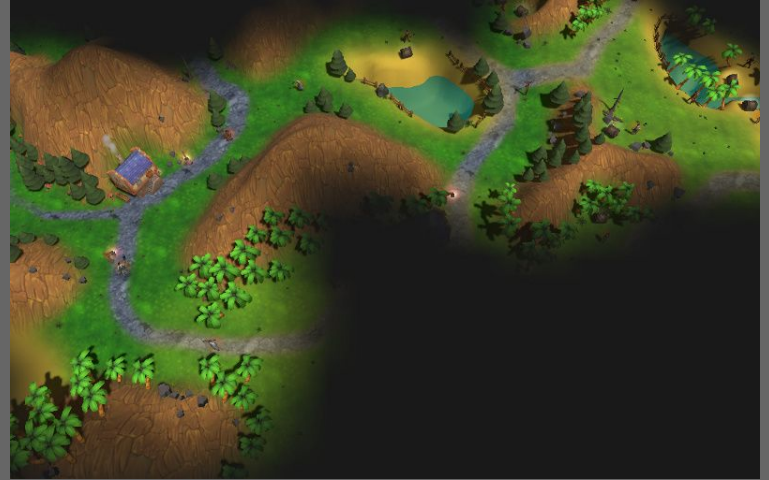
# Multiplayer



# Multiplayer

Our project's needs:

- Simultaneous turns
  - End-of-turn communication
- Fog of War
  - Filtered communication



# Netcode for GameObjects

- Mid-level networking library for Unity
- Players are either Hosts or Clients
- Communication via Remote Procedure Call (RPCs) or Named Messages





# Client-Server Communication

Client

Server

Commands

Filtered Game  
state updates

OK  
confirmation

Next Turn

At each step:

- Wait for timeout
- Gracefully handle error



# Issues and Difficulties

- Not the best documentation
- Synchronization issues
  - Order of initialization important
- Named Message length constraint
  - Solution: Make use of multiple RPCs to shoulder the workload

# 03

## Live Demo



# Demo Session!



# Thanks for listening!

**Credits:** This presentation template was created by **Slidesgo**, including icons by **Flaticon**, and infographics & images by **Freepik**

