



SHORT RECAP



```
GameState { MinionStates = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.MinionState], MechStates = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.MechState], GamePhase = DamageMechPhase {}, Board Tiles = System.Collections.Immutable.ImmutableDictionary<2[System.Drawing.Point,System.Boolean], CardDeck = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.Cards.Card], KilledMinions = 1 } }
3ms cycle
GameState { MinionStates = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.MinionState], MechStates = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.MechState], GamePhase = PickDrawableCardsPhase {} , BoardTiles = System.Collections.Immutable.ImmutableDictionary<2[System.Drawing.Point,System.Boolean], CardDeck = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.Cards.Card], KilledMinions = 1 } }
2ms cycle
GameState { MinionStates = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.MinionState], MechStates = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.MechState], GamePhase = DrawCardPhase { MechIndex = 0, DrawableCards = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.Cards.Card], BoardTiles = System.Collections.Immutable.ImmutableDictionary<2[System.Drawing.Point,System.Boolean], CardDeck = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.Cards.Card], KilledMinions = 1 } }
3ms cycle
Select Card from
0: RipsawCard (Blue)
1: ScytheCard (Blue)
2: BlazeCard (Red)
3: Empty
4: Empty
5: Empty
Select Slot from: 0, 1, 2, 3, 4, 5
3
GameState { MinionStates = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.MinionState], MechStates = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.MechState], GamePhase = DrawCardPhase { MechIndex = 1, DrawableCards = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.Cards.Card], BoardTiles = System.Collections.Immutable.ImmutableDictionary<2[System.Drawing.Point,System.Boolean], CardDeck = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.Cards.Card], KilledMinions = 1 } }
3ms cycle
Select Card from
0: ScytheCard (Blue)
1: BlazeCard (Red)
2: Empty
3: Empty
4: Empty
5: Empty
Select Slot from: 0, 1, 2, 3, 4, 5
4
GameState { MinionStates = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.MinionState], MechStates = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.MechState], GamePhase = DrawCardPhase { MechIndex = 2, DrawableCards = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.Cards.Card], BoardTiles = System.Collections.Immutable.ImmutableDictionary<2[System.Drawing.Point,System.Boolean], CardDeck = System.Collections.Immutable.ImmutableList<1[Mechs_Vs_Minions_Abstractions.GameStates.Cards.Card], KilledMinions = 1 } }
3ms cycle
Select Card from
0: ScytheCard (Blue)
1: Empty
2: Empty
3: Empty
4: Empty
5: Empty
Select Slot from: 0, 1, 2, 3, 4, 5
```


GOAL: IMPROVED UX

VISUAL CHANGES

Background



Animations



Game Figures



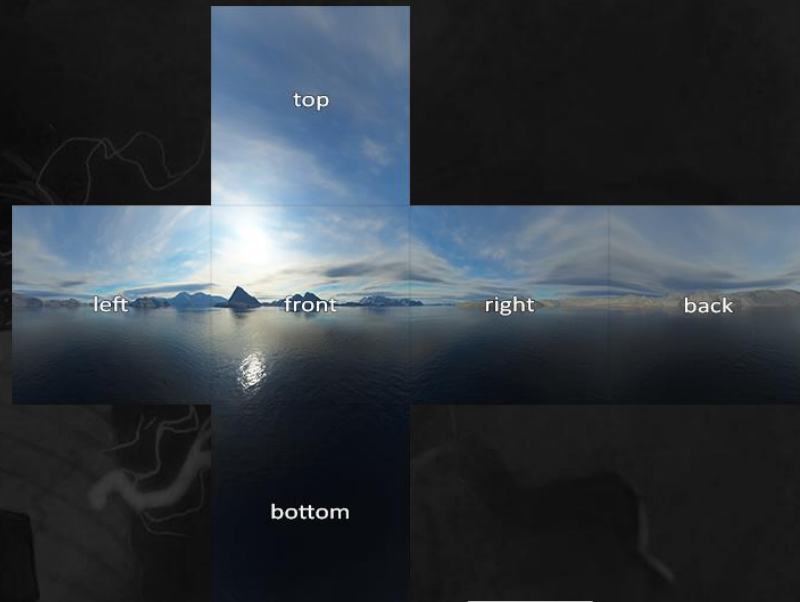
INTERACTIONS

- 🖱️ click-based user interaction
- 🔊 sound effects

GAME LOGIC

- 🎲 game board generation
- + (new card, pushing Mechs)

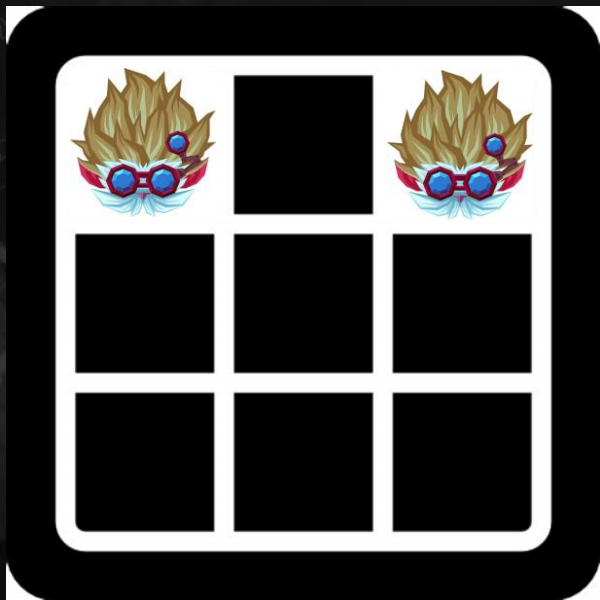
BACKGROUND



!

Remove camera translation!
Don't write to the depth buffer!

ANIMATIONS



GameState: MechAt(0,0)



NoTransition

GameState: MechAt(2,0)

ANIMATIONS



GameState: MechAt(0,0)



MechMoveTransition

GameState: MechAt(2,0)

ANIMATIONS

```
/// <summary>
/// Value between 0 and 1 where 0 means that the animation just started and 1 that it has finished
/// </summary>
1 reference | ubvwp, 16 hours ago | 1 author, 1 change
private float NormalizedAnimationTime => (float)(_playedDuration.TotalMilliseconds / _animationDuration.TotalMilliseconds);

1 reference | ubvwp, 16 hours ago | 1 author, 1 change
private void UpdatePassedTime(TimeSpan passedTime)
{
    _playedDuration = _playedDuration.Add(passedTime);
}

1 reference | ubvwp, 16 hours ago | 1 author, 1 change
private void UpdateInstance()
{
    var currentPosition:Vector3 = CalculateCurrentPosition();
    var currentAngle:float = _orientation.ToFloatAngle();
    _renderableInstanceState.Update(_instanceHandle, mutator: instance => instance with {
        ModelTransform = Matrix4.CreateRotationY(currentAngle) * Matrix4.CreateTranslation(currentPosition)
    });
}

1 reference | ubvwp, 16 hours ago | 1 author, 1 change
private Vector3 CalculateCurrentPosition()
    => _startPosition + Vector3.Multiply(_directionFromStartToEnd, NormalizedAnimationTime);
```

PARTICLES (kind of)



Vec3 Velocity
Vec3 Position
float LifeTime



500x

Future work:

- Update position on GPU
- Update color based on distance
- Smaller particles?

GAME FIGURES



GAME FIGURES

🖼️ ziggs_diffuse.png

🖼️ ziggs_normal.png

📄 ziggs.obj

📄 ziggs.mtl

📄 ziggs.obj.rcinfo

5k



GAME FIGURES

ziggs_diffuse.png

ziggs_diffuse.png

ziggs_normal.png

ziggs.obj

ziggs.mtl

10k

ziggs.obj.rcinfo



GAME FIGURES

ziggs_diffuse.png

ziggs_diffuse.png

ziggs_diffuse.png

ziggs_normal.png

ziggs.obj

ziggs.mtl

ziggs.obj.rcinfo

50k



GAME FIGURES

ziggs_diffuse.png

ziggs_diffuse.png

ziggs_diffuse.png

ziggs_diffuse.png

ziggs_normal.png

ziggs.obj

ziggs.mtl

ziggs.obj.rcinfo

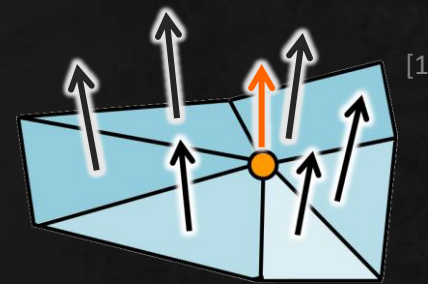
150k



GAME FIGURES

Problem: no normal in the .obj-files

Solution: calculate vertex normals

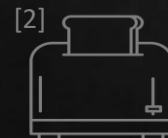


Problem: loading time (20s)

Solution: low resolution mode



polygons: 5k
images: 8192x8192



polygons: 5k
images: 1024x1024

GAME FIGURES

Problem: r

Solution: c

Problem: l

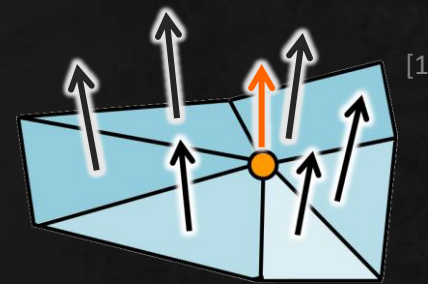
Solution: l



GAME FIGURES

Problem: no normal in the .obj-files

Solution: calculate vertex normals

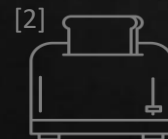


Problem: loading time (20s)

Solution: low resolution mode



polygons: 5k, ... ?
images: 8192x8192



polygons: 5k
images: 1024x1024

GOAL: IMPROVED UX

VISUAL CHANGES

Background



Animations



Game Figures



INTERACTIONS

- 🖱️ click-based user interaction
- 🔊 sound effects

GAME LOGIC

- 🎲 game board generation
- + (new card, pushing Mechs)

CLICK INTERACTION

Goal: let user control the game with mouse clicks
(only Overlay)



CLICK INTERACTION

Goal: let user control the game with mouse clicks
(only Overlay)

Basis: Overlay Scene Graph



CLICK INTERACTION

Goal: let user control the game with mouse clicks
(only Overlay)

Basis: Overlay Scene Graph



CLICK INTERACTION

Goal: let user control the game with mouse clicks
(only Overlay)

Basis: Overlay Scene Graph



Mouse Click

(300,300)



CLICK INTERACTION

Goal: let user control the game with mouse clicks
(only Overlay)

Basis: Overlay Scene Graph



CLICK INTERACTION

Goal: let user control the game with mouse clicks
(only Overlay)

Basis: Overlay Scene Graph



CLICK INTERACTION

Goal: let user control the game with mouse clicks
(only Overlay)

Basis: Overlay Scene Graph



CLICK INTERACTION

Goal: let user control the game with mouse clicks
(only Overlay)

Basis: Overlay Scene Graph



CLICK INTERACTION

Goal: let user control the game with mouse clicks
(only Overlay)

Basis: Overlay Scene Graph



CLICK INTERACTION

Goal: let user control the game with mouse clicks
(only Overlay)

Basis: Overlay Scene Graph



CLICK INTERACTION

Goal: let user control the game with mouse clicks
(only Overlay)

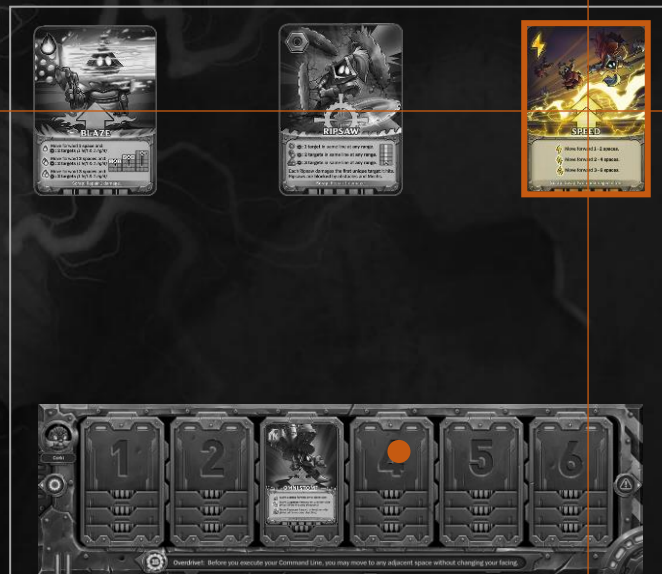
Basis: Overlay Scene Graph



CLICK INTERACTION

Goal: let user control the game with mouse clicks
(only Overlay)

Basis: Overlay Scene Graph



CLICK INTERACTION

Goal: let user control the game with mouse clicks
(only Overlay)

Basis: Overlay Scene Graph



CLICK INTERACTION

Goal: let user control the game with mouse clicks
(only Overlay)

Basis: Overlay Scene Graph



CLICK INTERACTION

Goal: let user control the game with mouse clicks
(only Overlay)

Basis: Overlay Scene Graph



CLICK INTERACTION

Goal: let user control the game with mouse clicks
(only Overlay)

Basis: Overlay Scene Graph



CLICK INTERACTION

Goal: let user control the game with mouse clicks
(only Overlay)

Basis: Overlay Scene Graph



Mouse Click: Command Line



CLICK INTERACTION

Mapping: Overlay Instance → *Action*



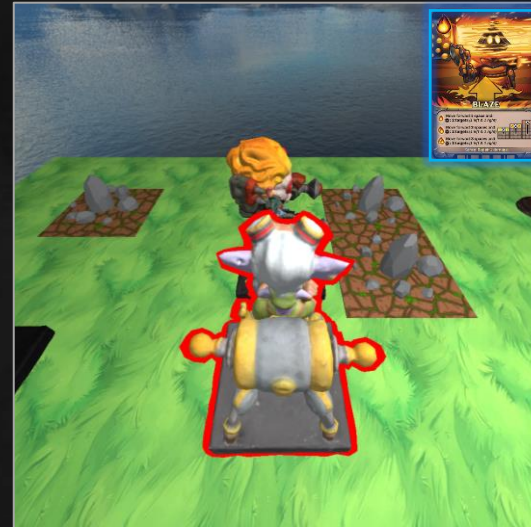
"Select Card ..."

CLICK INTERACTION

Mapping: Overlay Instance → *Action*



“Place card in slot ...”



“Confirm”



“Select direction ...”

GOAL: IMPROVED UX

VISUAL CHANGES

Background



Animations



Game Figures



INTERACTIONS

- 🖱️ click-based user interaction
- 🔊 sound effects

GAME LOGIC

- 🎲 game board generation
- + (new card, pushing Mechs)

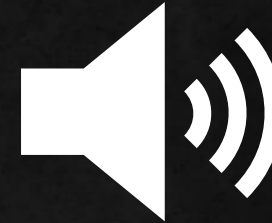
SOUND EFFECTS

1 reference | uekip, 12 days ago | 1 author, 1 change

```
public static void Load()
{
    ALBase.RegisterOpenALResolver();

    var device = ALC.OpenDevice(devicename: null);
    var context = ALC.CreateContext(device, attributeList: Array.Empty<int>());

    ALC.MakeContextCurrent(context);
}
```



1 reference | uekip, 12 days ago | 1 author, 1 change

```
private static AudioData ParseWavFile(string wavFile)
{
    var (format, dataInfo :DataChunk) = WAVFileReader.ReadFile(wavData: File.OpenRead(wavFile));

    var fs :FileStream = File.OpenRead(wavFile);
    fs.Seek(dataInfo.Begin, SeekOrigin.Begin);
    var data = new byte[dataInfo.Size];
    var redBytes :int = fs.Read(data, offset: 0, count: (int)dataInfo.Size);

    if (redBytes < dataInfo.Size) throw new Exception();

    return new AudioData(data, (int)format.SampleRate);
}
```


GOAL: IMPROVED UX

VISUAL CHANGES

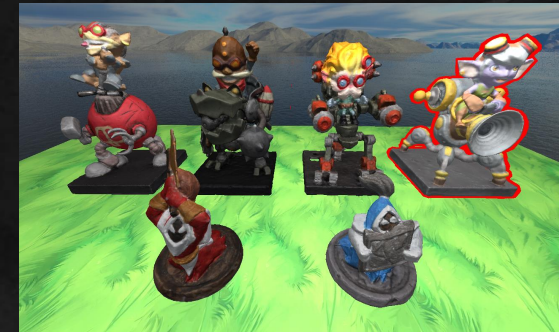
Background



Animations



Game Figures



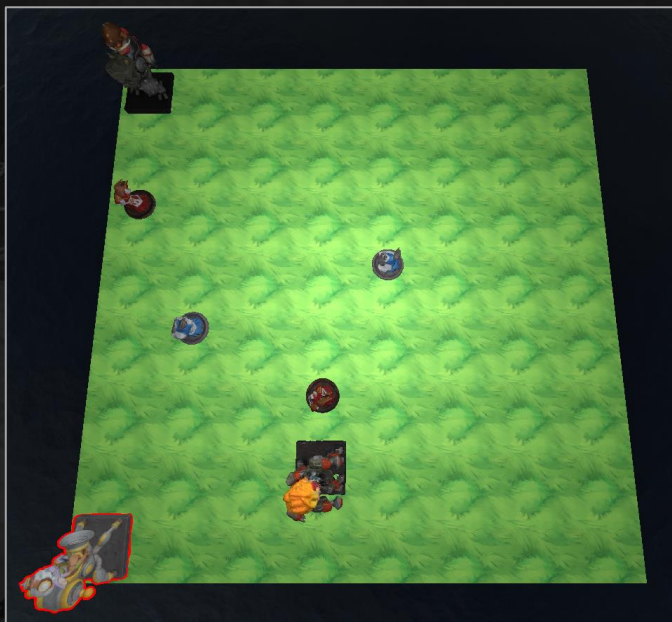
INTERACTIONS

- 🖱️ click-based user interaction
- 🔊 sound effects

GAME LOGIC

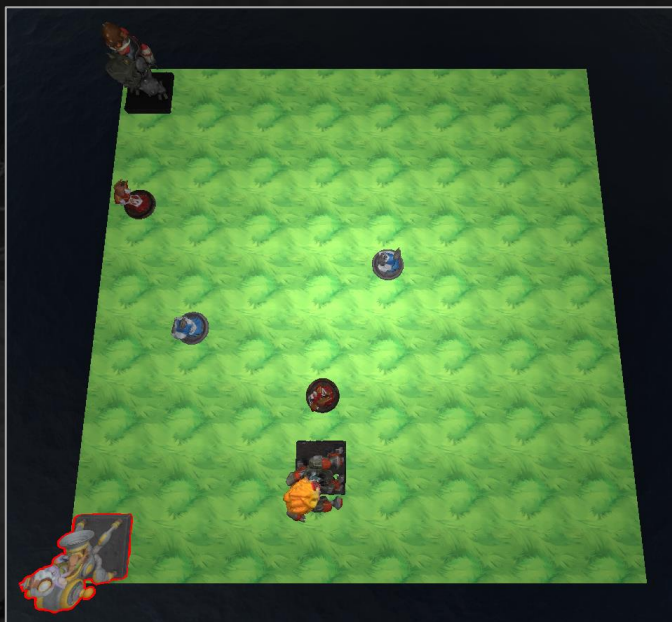
- 🎲 game board generation
- + (new card, pushing Mechs)

MAP GENERATION



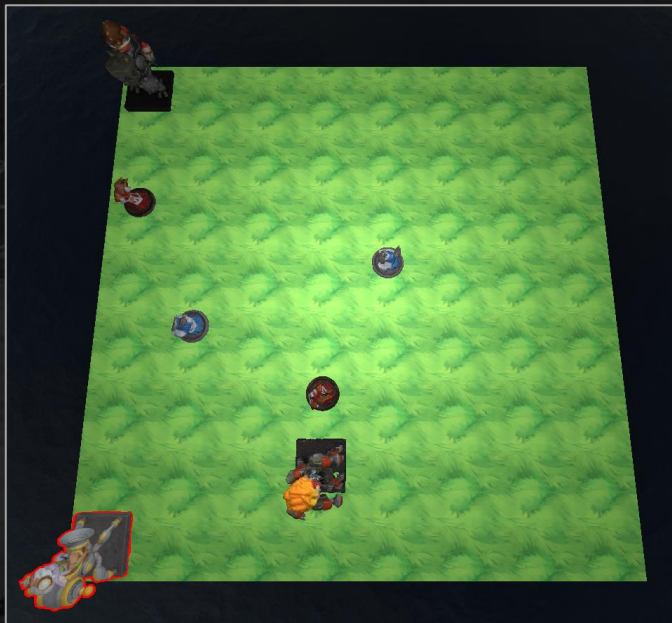
- fixed game board size
- fixed positions for game figures
- open game board

MAP GENERATION



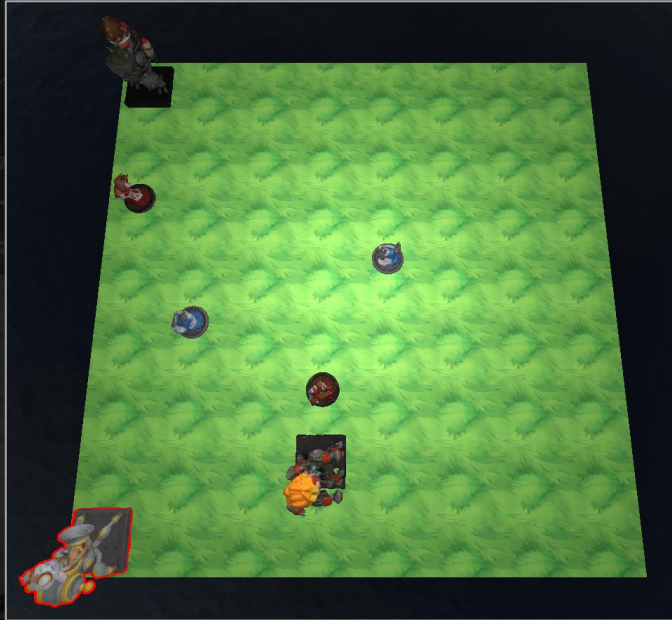
- **arbitrary** game board size
- fixed positions for game figures
- open game board

MAP GENERATION



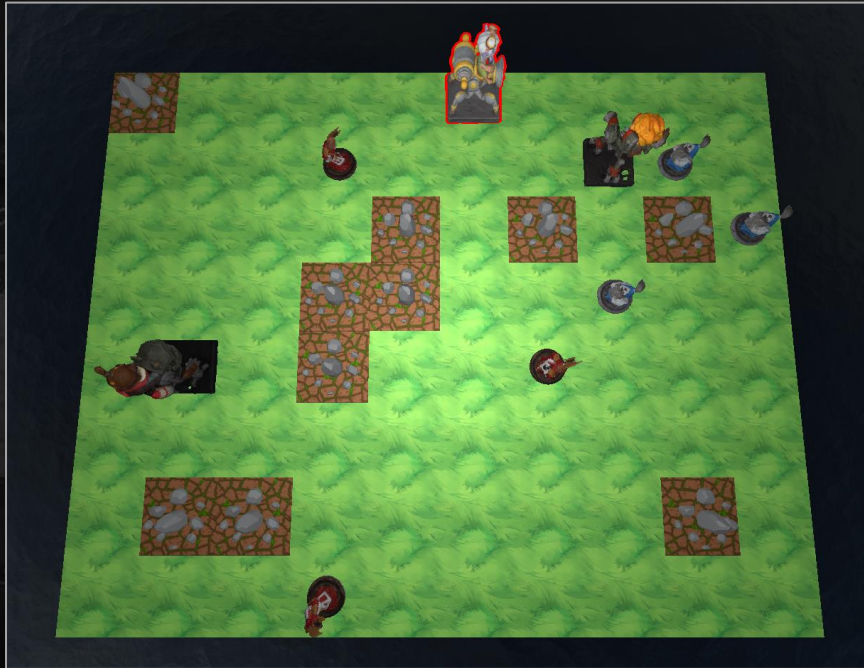
- **arbitrary** game board size
- **random** positions for game figures
- open game board

MAP GENERATION



- **arbitrary** game board size
- **random** positions for game figures
- open game board **with obstacles**

MAP GENERATION



- **arbitrary** game board size
- **random** positions for game figures
- open game board **with obstacles**

MAP GENERATION



rd size

or game figures

with obstacles

MAP GENERATION

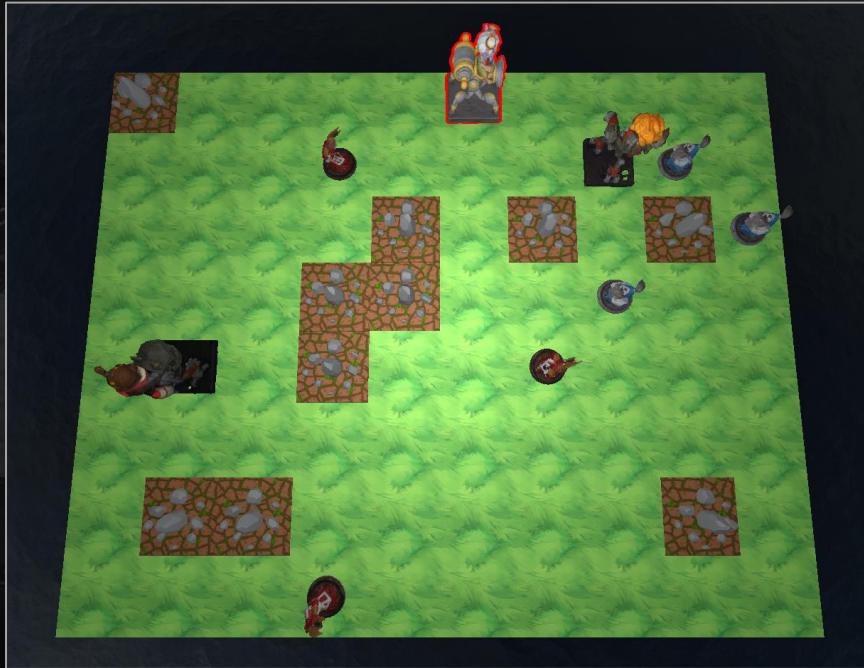


rd size

or game figures

with obstacles

MAP GENERATION



- **arbitrary** game board size
- **random** positions for game figures
- open game board **with obstacles**

= all Mechs can reach
all game figures

The image features a dark, textured background with a metallic, industrial aesthetic. A complex frame of metal and wiring surrounds the central area. The frame includes yellow and black hazard stripes on the left and right sides. At the bottom center, there is a small, glowing blue rectangular light. The word "DEMO" is prominently displayed in the center in a white, serif font, underlined.

DEMO

TOPICS

BACKGROUND



! Remove camera translation! Don't write to the depth buffer!

ANIMATIONS

PARTICLES (kind of)



Vec3 velocity
Vec3 position
Real lifetime

500x

Future work:
• Update position on GPU
• Update color based on distance
• Smaller particles?

GAME FIGURES

ai_ziggs_diffuse.png
ai_ziggs_diffuse.png
ai_ziggs_diffuse.png
ai_ziggs_diffuse.png
ai_ziggs_normal.png
ai_ziggs.obj
ai_ziggs.mtl
ai_ziggs.obj.info



CLICK INTERACTION

Mapping: Overlay Instance → Action

CLICK INTERACTION

Goal: let user control the game with mouse clicks (only Overlay)

Basis: Overlay Scene Graph



Mouse Click: Command Line

SOUND EFFECTS



MAP GENERATION



- arbitrary game board size
- random positions for game figures
- open game board with obstacles

REFERENCES

- mainly: <https://euw.leagueoflegends.com/de/featured/mechs-vs-minions>
- [1] “Computergrafik” slides by Prof. Dr. Dachsbacher
- [2] Cartoon Vectors by Vecteezy <https://www.vecteezy.com/free-vector/cartoon>