

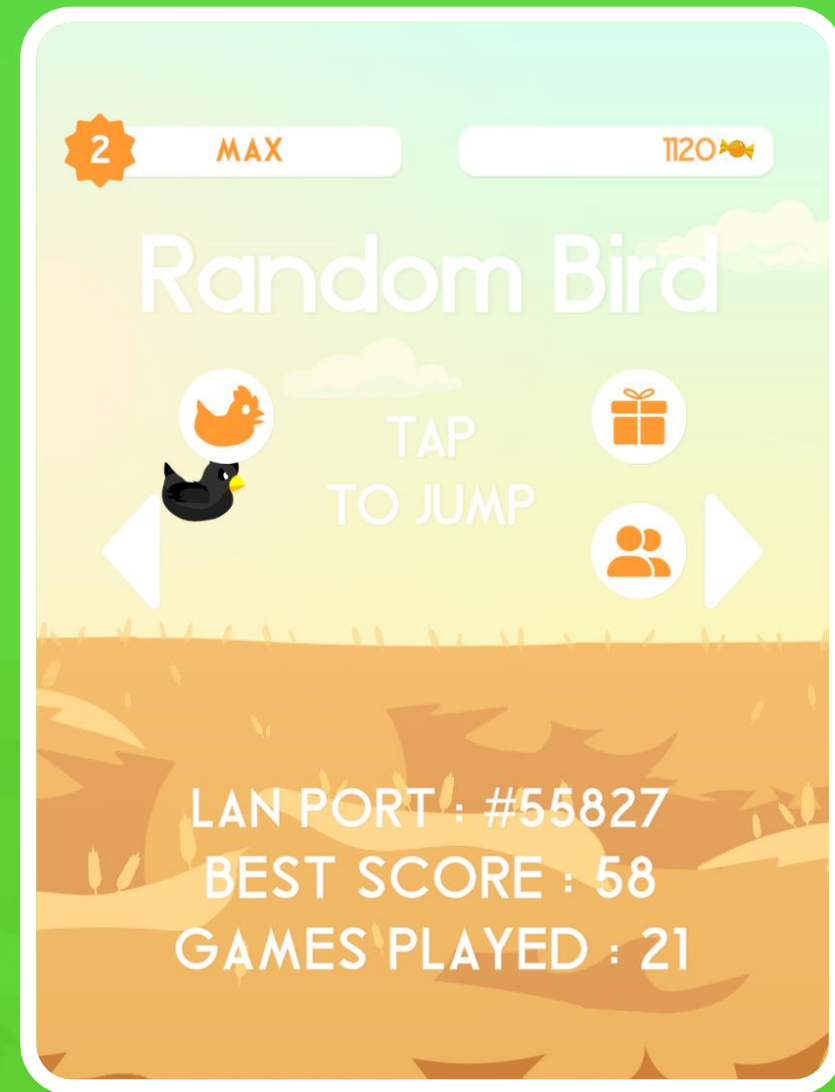


Random Bird



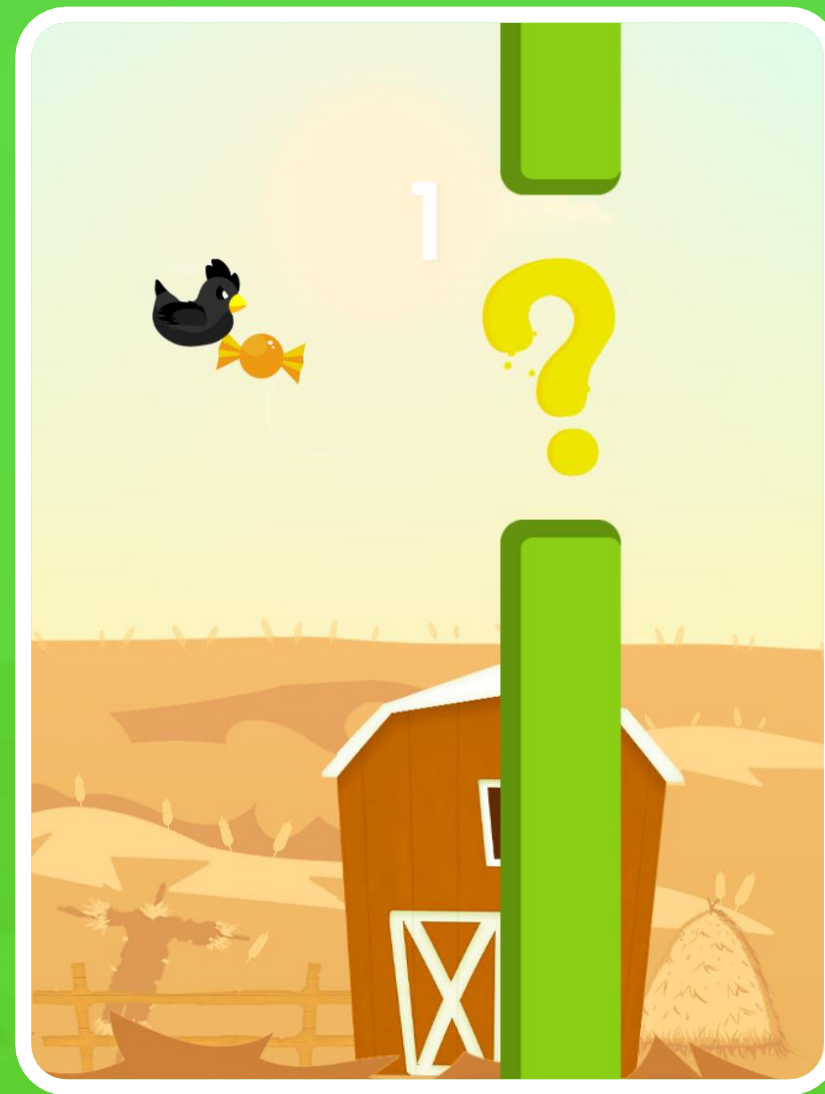
Úvod

- 2D mobilní hra
- Single-player
- Hra pro děti
- Cartoon design



Koncept a cíle hry

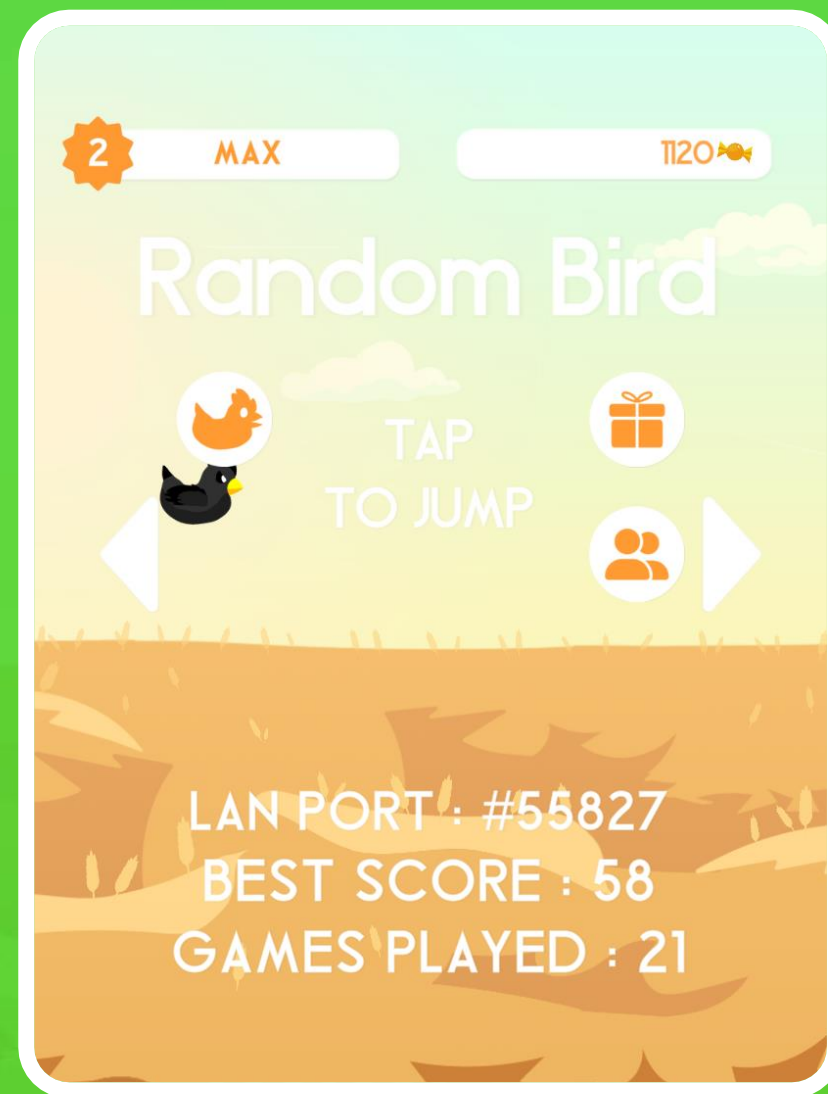
- Nekonečná hra
- Hráč se musí dostávat přes překážky a tím zvyšovat skóre
- Hra se s větším skóre ztěžuje
- Sběr bonbónů = herní měna
- Hlavní cíl hry je odemykání nových map a skinů za bonbóny





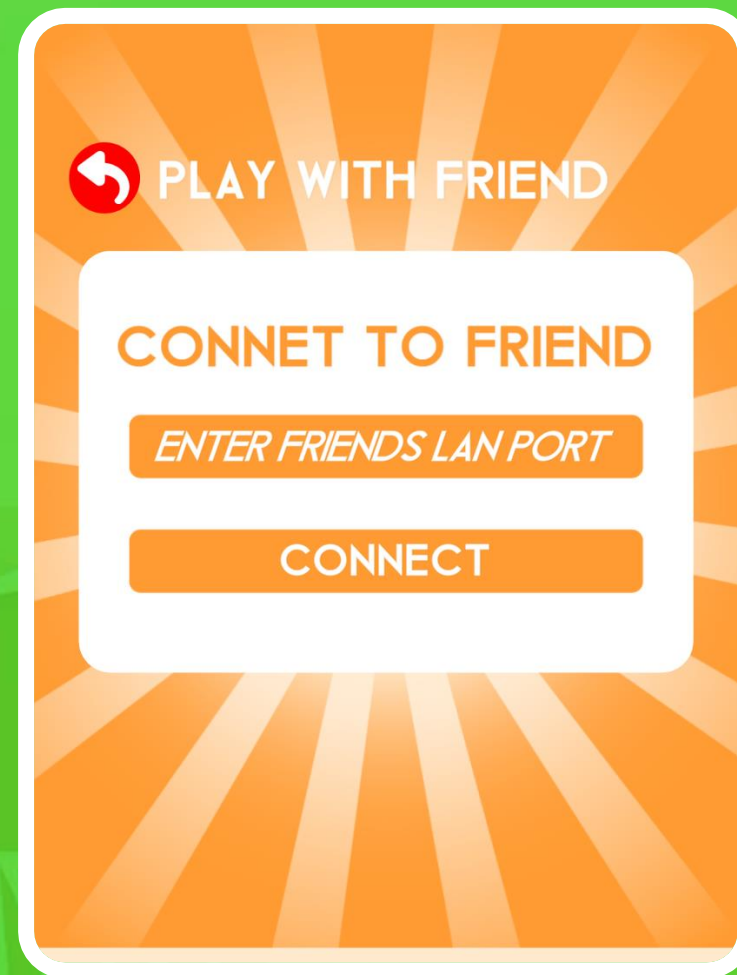
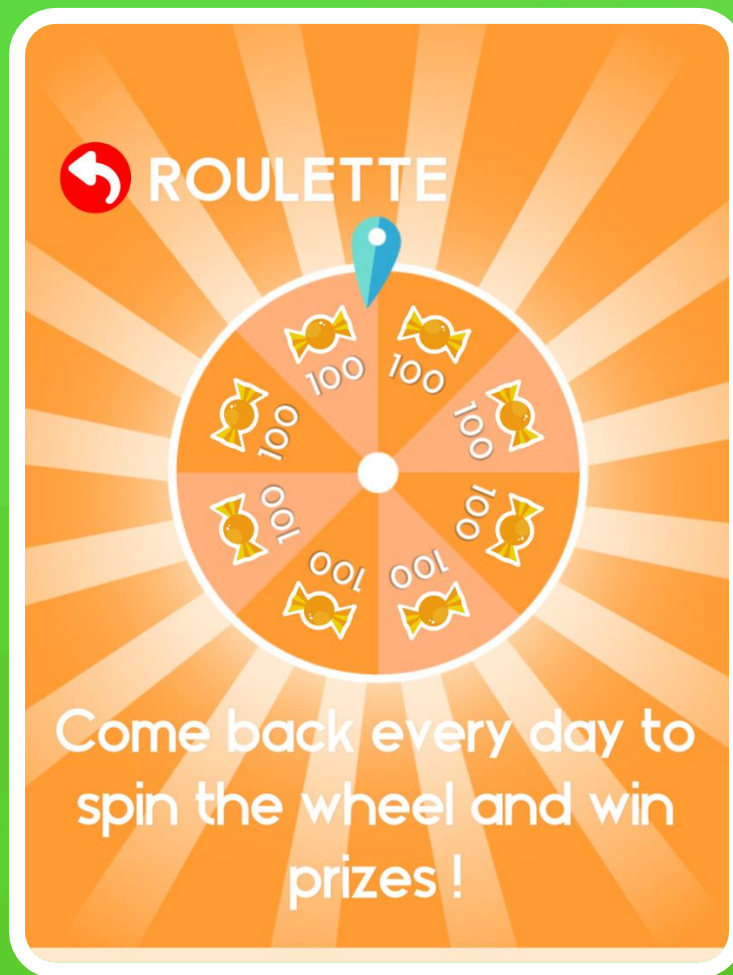
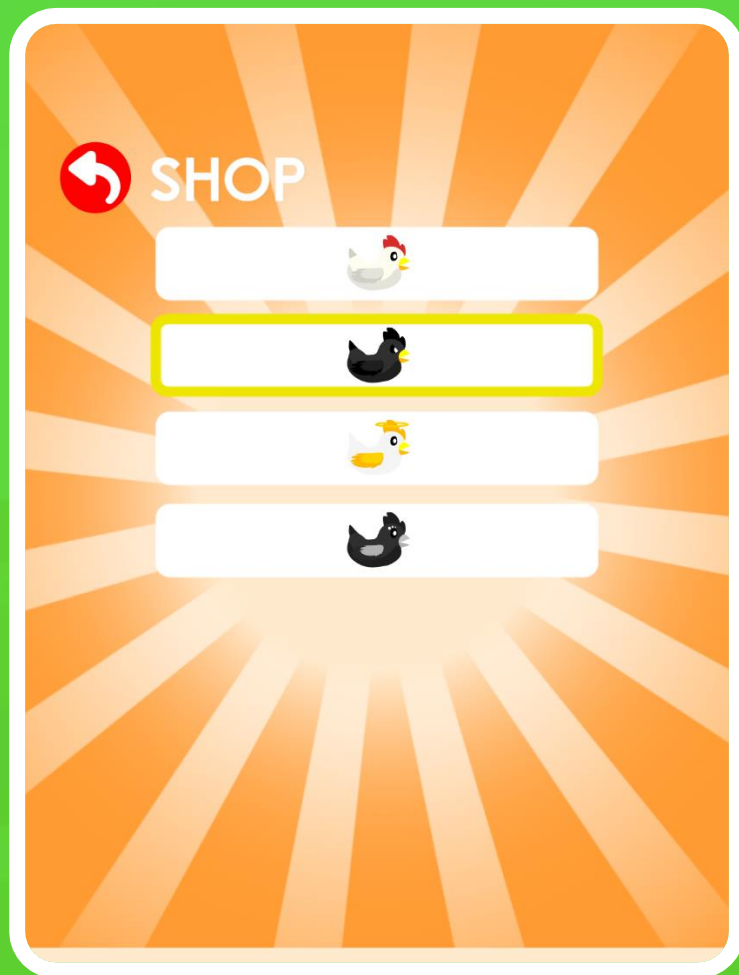
Design a grafika

- 2D
- Design je zaměřen na jednoduchost
- Hráč hraje za slepici
 - Veliký výběr skinů (odemykají se bonbóny)
- Cartoon design
- Každá mapa má svoje prostředí



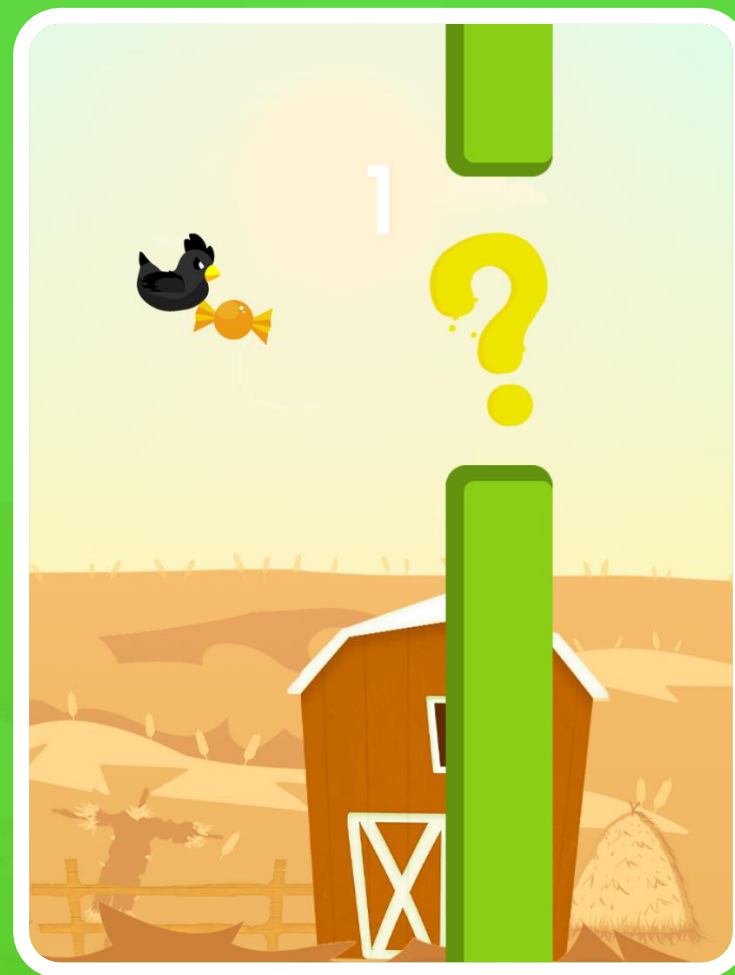
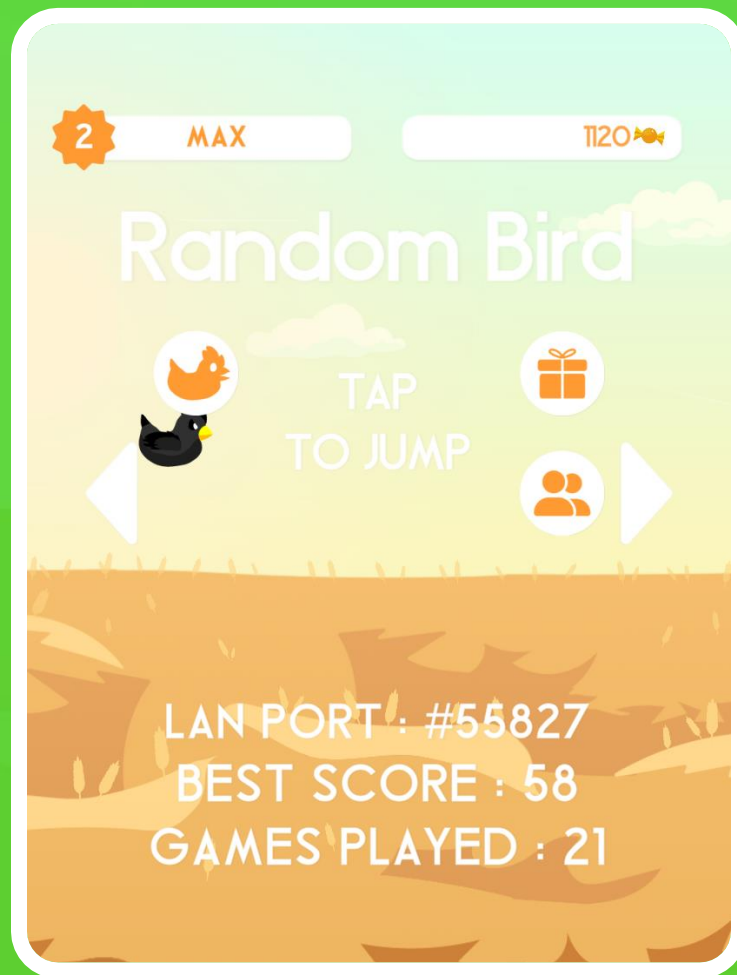


Design a grafika



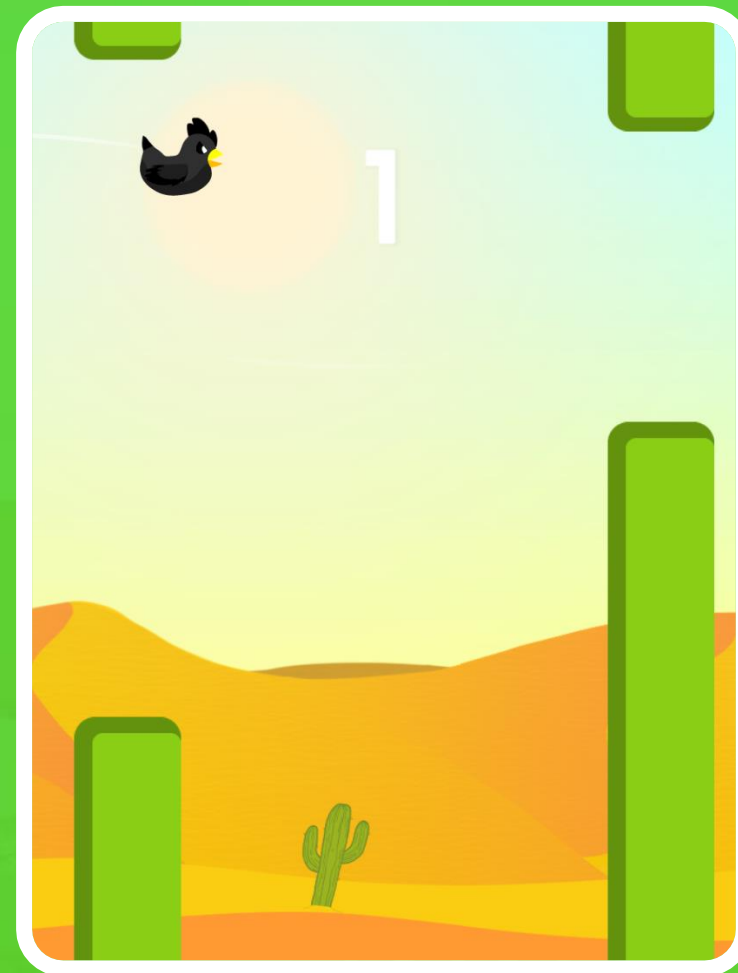


Design a grafika





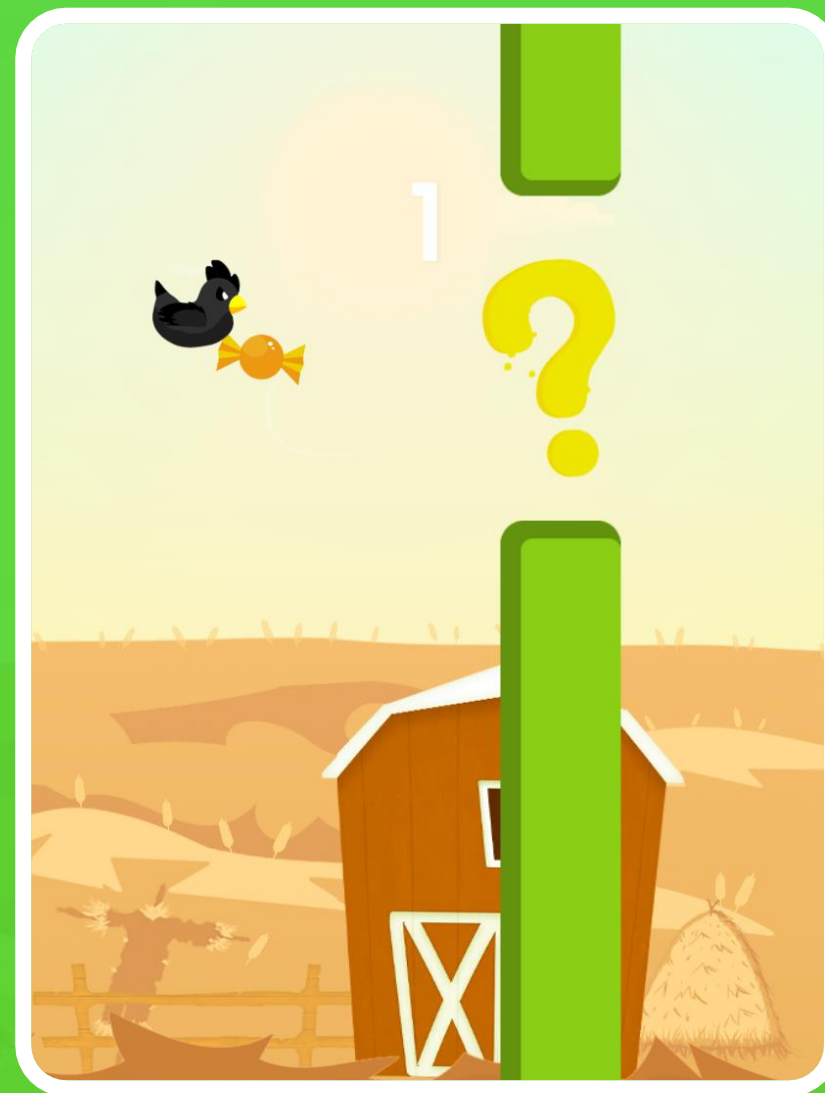
Design a grafika



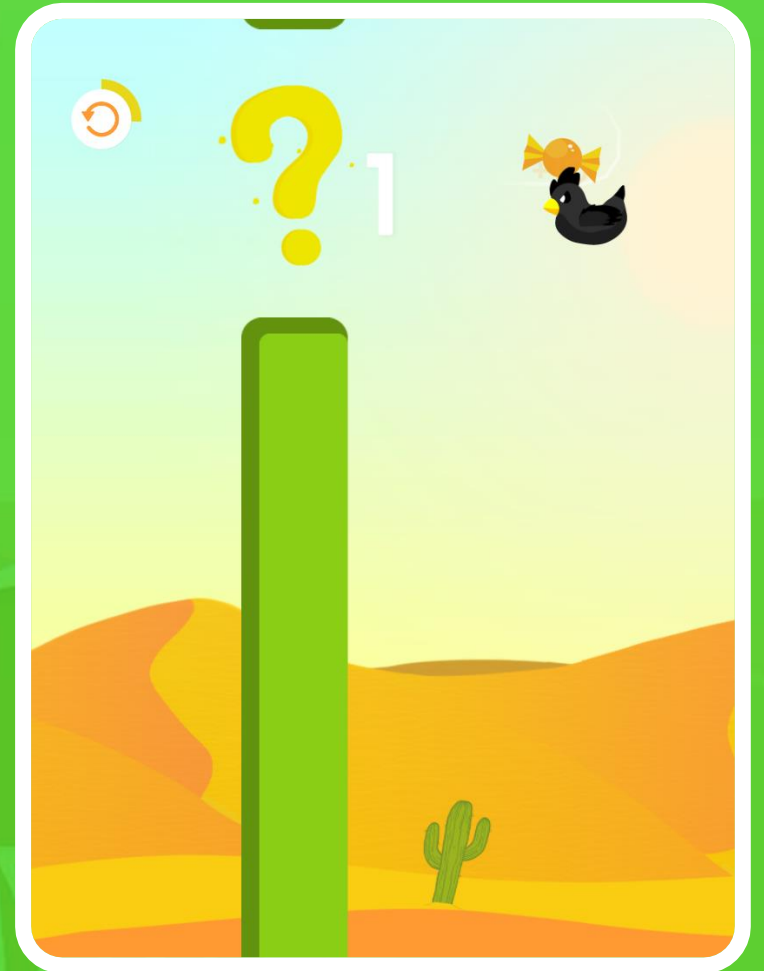
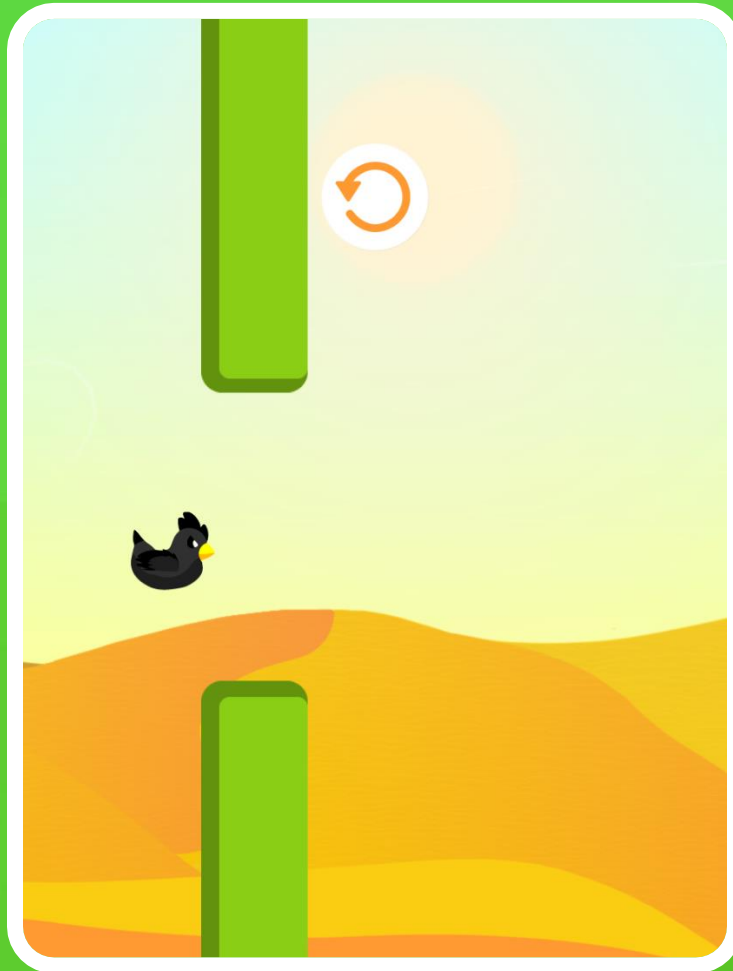


Mechaniky

- Hráč hraje za slepici
- Klepnutí na display = slepice mávne křídly a nadlétne
- Hráč se v průběhu hry potkává s otazníky
- Otazník vybere náhodný efekt
 - Změna směru, vyhýbání se raketám, zhoršení viditelnosti, ...
- Každá mapa má svoje specifické efekty
- Změna dne a noci



Mechaniky





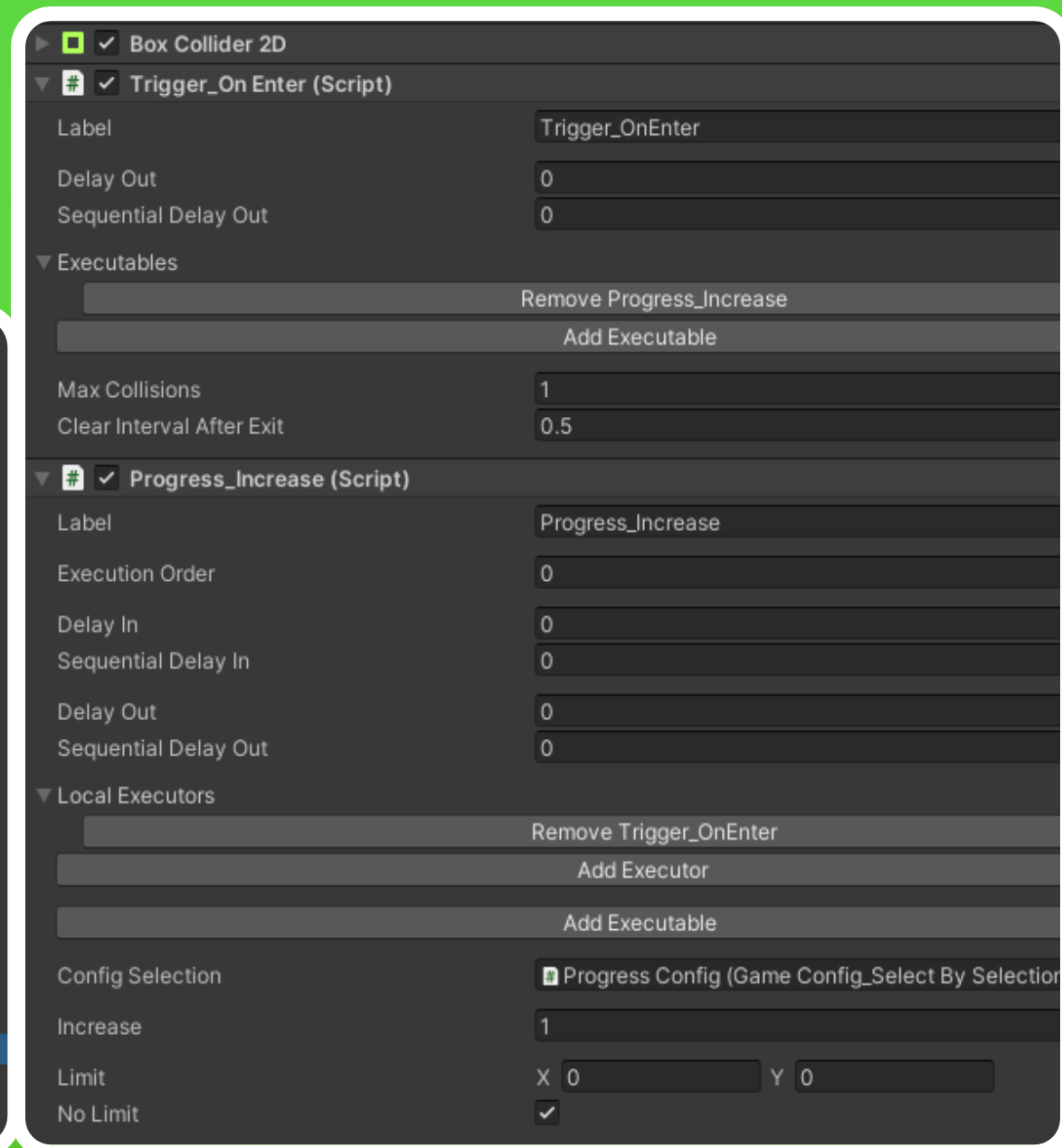
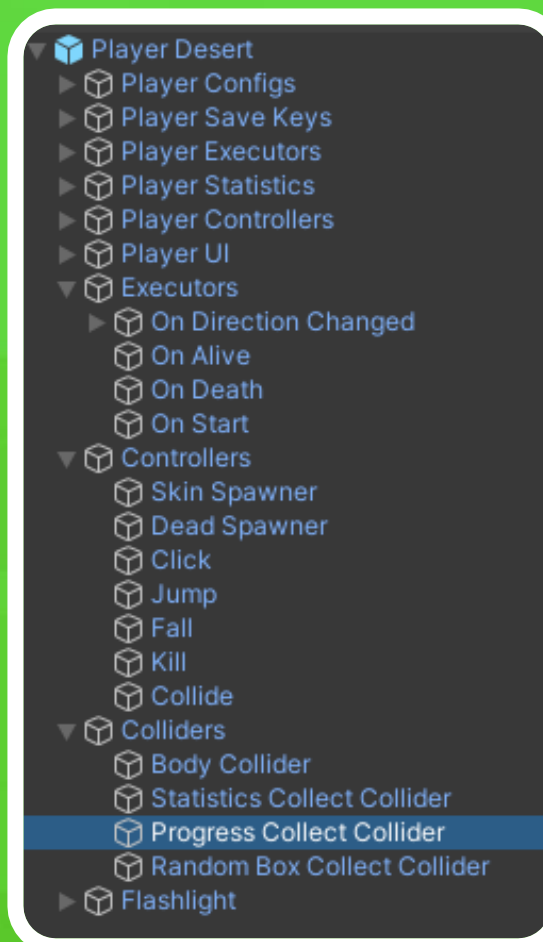
Implementace

- Unity engine
- Plugin Mirror pro multiplayer
- Photoshop a Illustrator



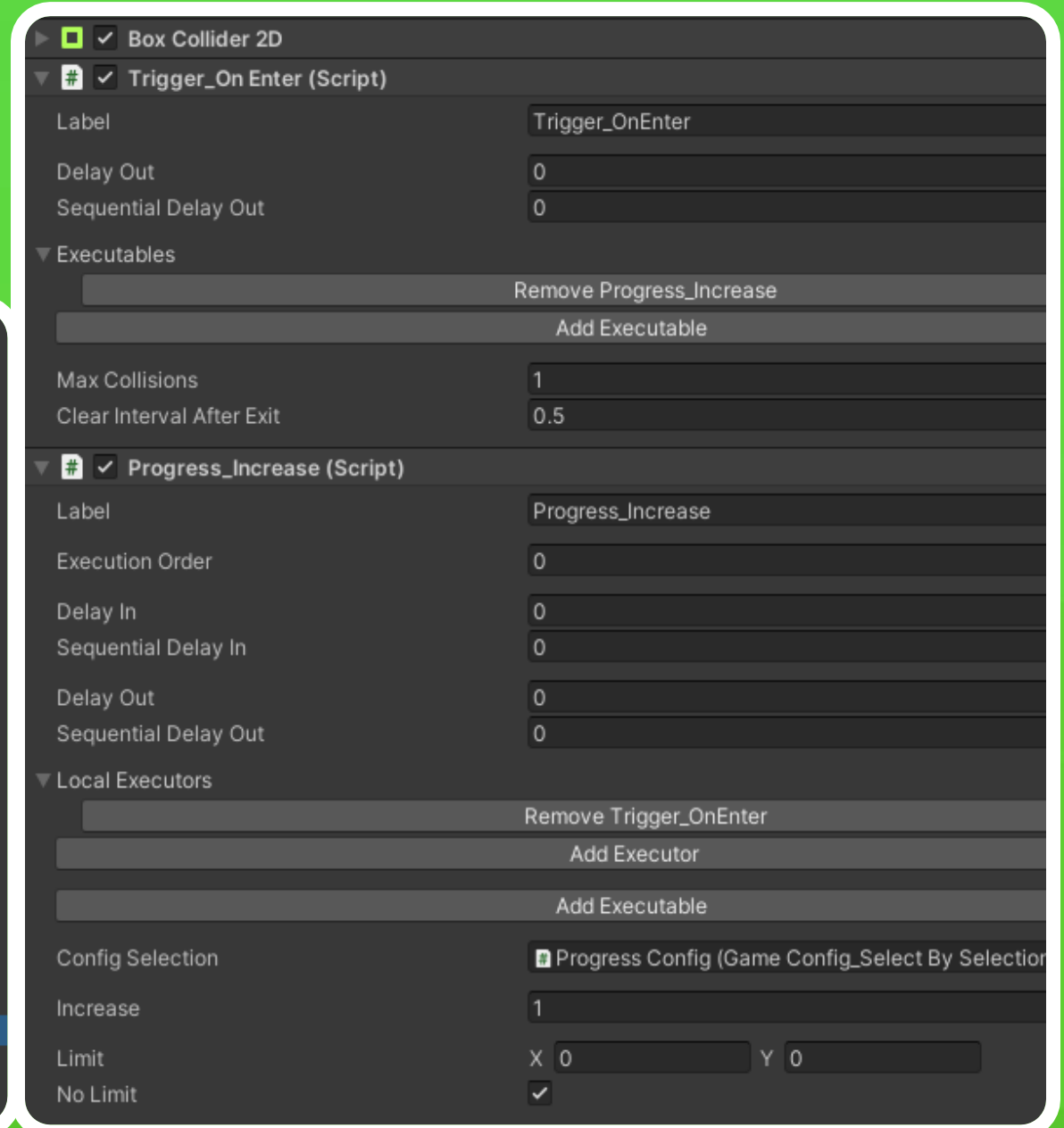
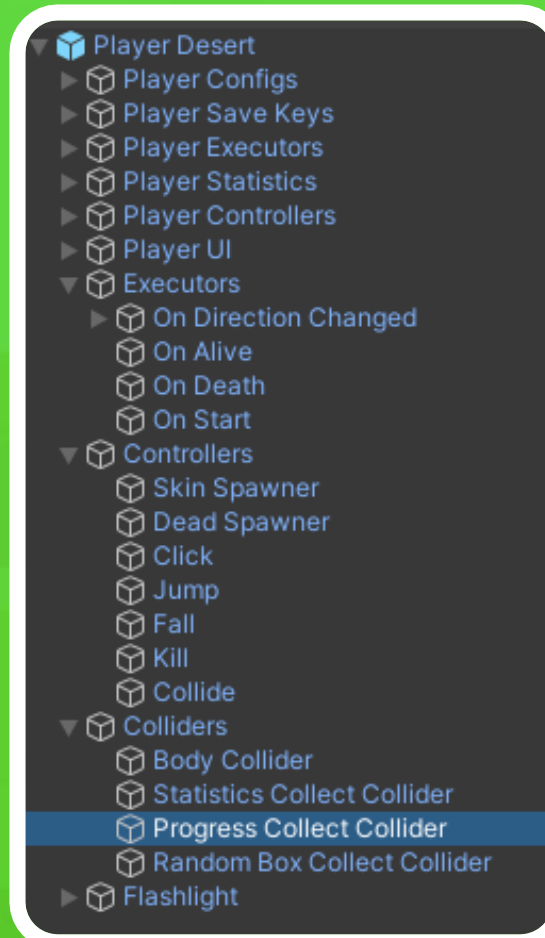
Implementace

- Component design



Implementace

- Executor/executable





Implementace

- Runtime config

```
Unity Script (10 asset references) | 0 references
public class GameConfig_OnHorizontalDirectionChanged : DynamicExecutor
{
    [GameConfigSerializeProperty]
    3 references
    public IGameConfig_HorizontalDirectional Config { get; }

    [OnGameConfigChanged(nameof(Config))]
    0 references
    private void OnConfigChanged(string affected)
    {
        if (GameConfig.IsFieldAffected(affected, nameof(IGameConfig_Horiz
        {
            if (_directionCheckMap[Direction](Config))
            {
                Execute(Time.deltaTime);
            }
        }
    }
}
```

Implementace

- IL post processor
- Knihovna Mono.Cecil

```
[NonSerialized]
private GameConfig _lastConfig__b518905ca5d348c1a9095fd4a6c2c335;
0 references
GameConfig_OnHorizontalDirectionChanged()
{
    AddOnAwakeOnceReceiver(OnAwakeOnceReceiver__35bb3e790ded447bae6241b4496d2677);
    AddOnDestroyReceiver(OnDestroyReceiver__fad54f2ef6474a158c3bcd1575494539);
}

1 reference
private void OnAwakeOnceReceiver__35bb3e790ded447bae6241b4496d2677()
{
    ConfigSelection.AddOnNewConfigSetReceiver(OnNewConfigSetReceiver__f67dbb5c418c46d483cda76761e24c81((GameConfig)ConfigSele
OnNewConfigSetReceiver__f67dbb5c418c46d483cda76761e24c81((GameConfig)ConfigSele
}

1 reference
private void OnDestroyReceiver__fad54f2ef6474a158c3bcd1575494539()
{
    GameConfig_IExtensions.RemoveGameConfigOnBlendReceiver(
        _lastConfig__b518905ca5d348c1a9095fd4a6c2c335, OnConfigChanged);
}

2 references
private void OnNewConfigSetReceiver__f67dbb5c418c46d483cda76761e24c81(GameConfig newConfig)
{
    GameConfig_IExtensions.SetGameConfigOnBlendReceiver(
        _lastConfig__b518905ca5d348c1a9095fd4a6c2c335, newConfig, OnConfigChanged);

    _lastConfig__b518905ca5d348c1a9095fd4a6c2c335 = newConfig;
}

public GameConfig_Selection<IGameConfig_HorizontalDirectional> ConfigSelection;
[GameConfigSerializeProperty]
public IGameConfig_HorizontalDirectional Config { get; }
```





```
string onNewConfigSetMethodName = $"OnNewConfigSetReceiver_{Guid.NewGuid():N}";
MethodDefinition onNewConfigSetMethod = new(onNewConfigSetMethodName, MethodAttributes.None,
onNewConfigSetMethod.Parameters.Add(new("gameconfig", ParameterAttributes.None,
onNewConfigSetMethod.Body = new(onNewConfigSetMethod);
ILProcessor onNewConfigSetWorker = onNewConfigSetMethod.Body.GetILProcessor();
onNewConfigSetWorker.Emit(OpCodes.Ldarg_0);
onNewConfigSetWorker.Emit(OpCodes.Ldfld, lastConfigField);
onNewConfigSetWorker.Emit(OpCodes.Ldarg_1);
onNewConfigSetWorker.Emit(OpCodes.Ldarg_0);
onNewConfigSetWorker.Emit(OpCodes.Ldftn, onConfigChangedMethod);
onNewConfigSetWorker.Emit(OpCodes.Newobj, actionStringCtorReference);
onNewConfigSetWorker.Emit(OpCodes.Call, weaverTypes.GameConfigAddOnBlendReceiver);

onNewConfigSetWorker.Emit(OpCodes.Ldarg_0);
onNewConfigSetWorker.Emit(OpCodes.Ldarg_1);
onNewConfigSetWorker.Emit(OpCodes.Stfld, lastConfigField);

onNewConfigSetWorker.Emit(OpCodes.Ret);
```

```
private void OnNewConfigSetReceiver__f67dbb5c418c46d483cda76761e24c81(GameConfig gameConfig)
{
    GameConfig_IExtensions.SetGameConfigOnBlendReceiver(
        _lastConfig__b518905ca5d348c1a9095fd4a6c2c335, gameConfig, OnConfigChanged);

    _lastConfig__b518905ca5d348c1a9095fd4a6c2c335 = gameConfig;
}
```

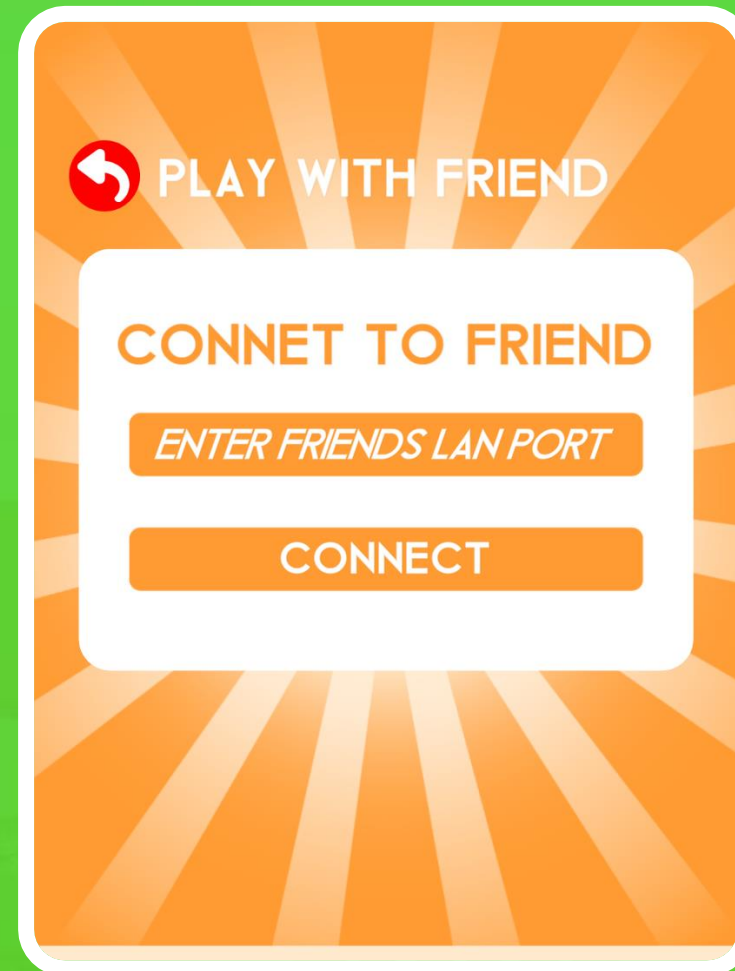
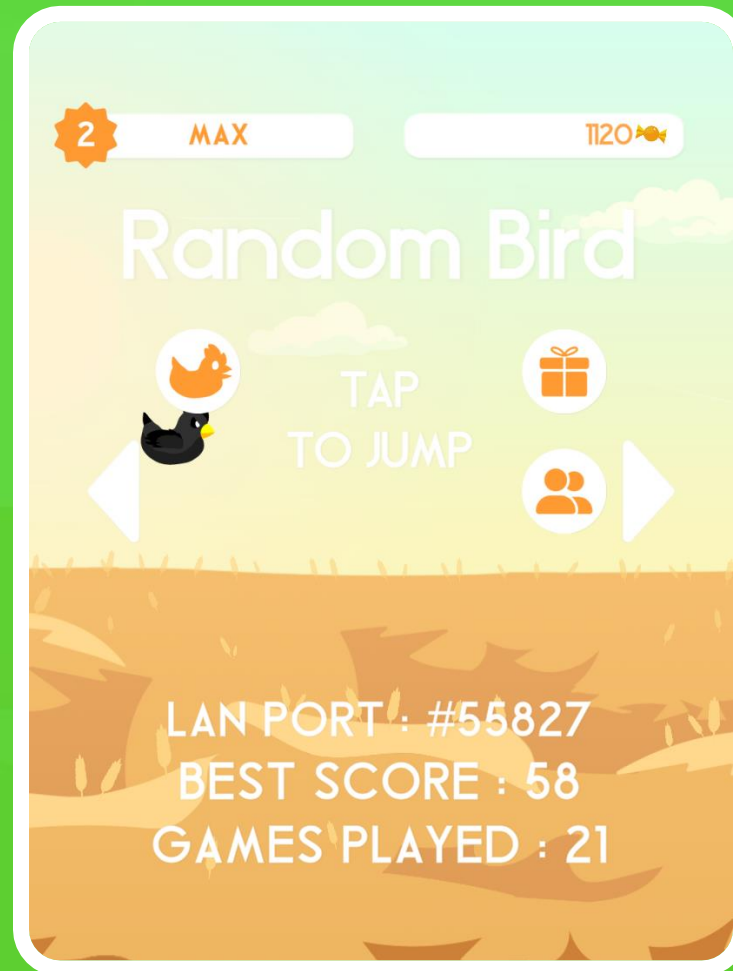



Implementace

- LAN Multiplayer
 - Do budoucna server
- Není hlavním prvkem hry
- Hráči mohou hrát stejně vygenerovanou hru
- TPC protokol pro důležité zprávy
 - Start hry, přeposlání seedů, ...
- UDP protokol pro pohyb hráče



Implementace





Podpora

- iOS
 - Zákaz dynamického programování
- Android





Shrnutí

- Okolo 800 hodin práce
 - Zahrnuje učení se novým technologiím
 - 3x předělání celého projektu
 - Vymýšlení vlastních systémů
- Věci do budoucna
 - Nové mapy, skiny, efekty
 - Globální multiplayer
 - Vydání hry

