# Unity-Workshop Informatiktage 2017

## Software

* Unity 5.6.1f1 <https://store.unity.com/download?ref=personal>
* Android Studio <https://developer.android.com/studio/index.html#tos-header>
* Ev. JDK 1.8 (Windows x64): <http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>
* ~~GIT (?) -> Kopieren per USB~~

## Ablauf

* Vorstellung 5min (Beni, Kaspar)
* Unity introduction (Präsentation) 10min (Beni)
  + Layout «Tall»
  + Split Editor und Game View
  + Project «One Column Layout»
  + Editor
  + Koordinatensystem, Gizmo
  + Navigation
  + Orbit: Alt+Linksklick, Zoom: Mausrad, oder Alt+Rechtsklick
  + Rechts-Drag, WASD
  + Manipulators
  + Hierarchy, Baum
  + Alles ist ein Gameobject
  + Drag and Drop
  + Transform, «Vererbung»
  + Inspector
  + Eigenschaften von ausgewähltem Objekt
  + Components
  + Project
  + Ressourcen
  + Können in mehreren Szenen verwendet werden
  + Prefab erstellen, duplizieren, modifizieren, apply
* Resultat/Ziel zeigen 2min -> nur auf Screen zeigen (Kaspar)
* UI/Editor-Übersicht 15min (Beni)

## Workshop

* Terrain (kasc)
  + Skybox auswählen
  + Heightmap zeichnen, verschiedene Pinsel
  + Texturen zeichnen
  + Bäume hinzufügen (nicht zu viele)
  + In der Mitte Flatten
  + Light:
    - Directional Light hinzufügen
    - Verschiedene Arten erklären
* Cactus Prefab reinziehen
  + Material und Texturen
    - Albedo
    - Normalmap
* CactusMove hinzufügen (beba)

    public float moveSpeed = 0.5f;  
    public Vector3 playerPosition = Vector3.zero;  
  
    void Update()  
    {  
        transform.LookAt(playerPosition);

if (!IsNearPlayer())

{

transform.position += transform.forward \* moveSpeed \* Time.deltaTime;

}

else if (IsReadyToAttack())

{

animator.Play("Attack");

}

    }

* EnemySpawner GO hinzufügen (ks)
  + Script zuweisen
  + Referenz auf Prefab
  + Interval = 5, Radius = 6

    public void StartSpawning()  
    {  
        coroutine = StartCoroutine(SpawnEnemy());  
    }

    if (coroutine != null)

{ StopCoroutine(coroutine);

}

    public void IncreaseDifficulty()  
    {  
        spawnInterval \*= 0.9f;  
    }

* CactusBehaviour (ks)
  + Animation
  + Hitcounter und Die() implementieren

        hitCounter++;  
  
        if (hitCounter >= 3)  
        {  
            Die();  
        }  
    }

* + Destroy Implementieren

    public void DestroyEvent()  
    {  
        Destroy(gameObject);  
    }

* + DecreaseHealthPoints (später zeigen)

    public void HitPlayer()  
    {  
        FindObjectOfType<ScoreManager>().DecreaseHealthPoints(1);  
  
        *// play audio*  
        GetComponent<AudioSource>().clip = audioHit;  
        GetComponent<AudioSource>().volume = 1.0f;  
        GetComponent<AudioSource>().pitch = Random.Range(0.8f, 1.1f);  
        GetComponent<AudioSource>().Play();  
    }

* + IncreaseScore (später zeigen)

    private void Die()  
    {  
        GetComponent<Collider>().enabled = false;  
        GetComponent<Animator>().Play("Die");  
  
        FindObjectOfType<ScoreManager>().IncreaseScore(1);  
    }

* ScoreManager

        *// check if game over*  
        if(healthPoints == 0)  
        {  
            GameOver();  
        }  
    }

* Sound
  + Referenzen neu setzen: grunt, hit, punch
* Builden