



LUDWIG-  
MAXIMILIANS-  
UNIVERSITÄT  
MÜNCHEN

FAKULTÄT FÜR MATHEMATIK, INFORMATIK UND STATISTIK  
INSTITUT FÜR INFORMATIK  
ARBEITSGRUPPEN MEDIENINFORMATIK UND  
MENSCH-MASCHINE-INTERAKTION



# Blockpraktikum Multimediatechnische Programmierung

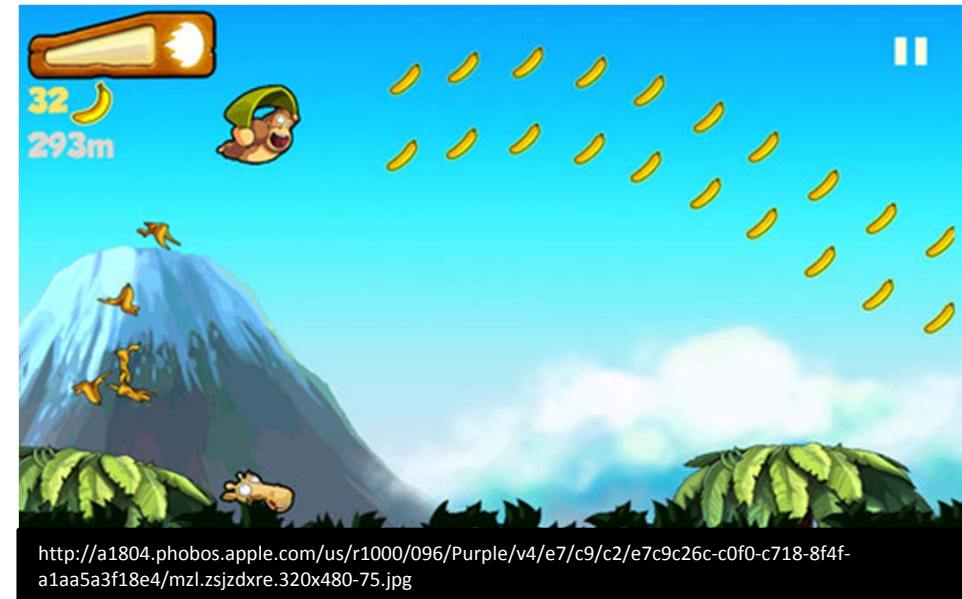
Henri Palleis  
Aufgabenstellung

# Das Projekt

- Vier Teams à 5/6 Personen
- Aufgabenstellung für alle Teams identisch
- Trotzdem möglichst innovatives und kreatives Ergebnis
- Spielspaß!
  - Spielspaß genauso wichtig wie Umsetzung der Funktionalität

# Die Aufgabenstellung

## *ENDLESS RUNNING*



# Weitere Beispiele

2D:

- Jetpack Joyride, Tiny Wings, Ski Safari, Punch Quest

3D:

- Temple Run, Pitfall, Subway Surf, Run with Friends

# Grundsätzliche funktionale Anforderungen



- Anwendung besteht neben dem Spiel selbst aus:
  - Startmenü
  - Impressum
  - (Kurz-)Anleitung (z.B. In-Game-Anleitung)
- Spiel soll auf Physik-Engine basieren
- Level-Aufbau soll zufällig generiert werden
- Eigenes einzigartiges Spielkonzept

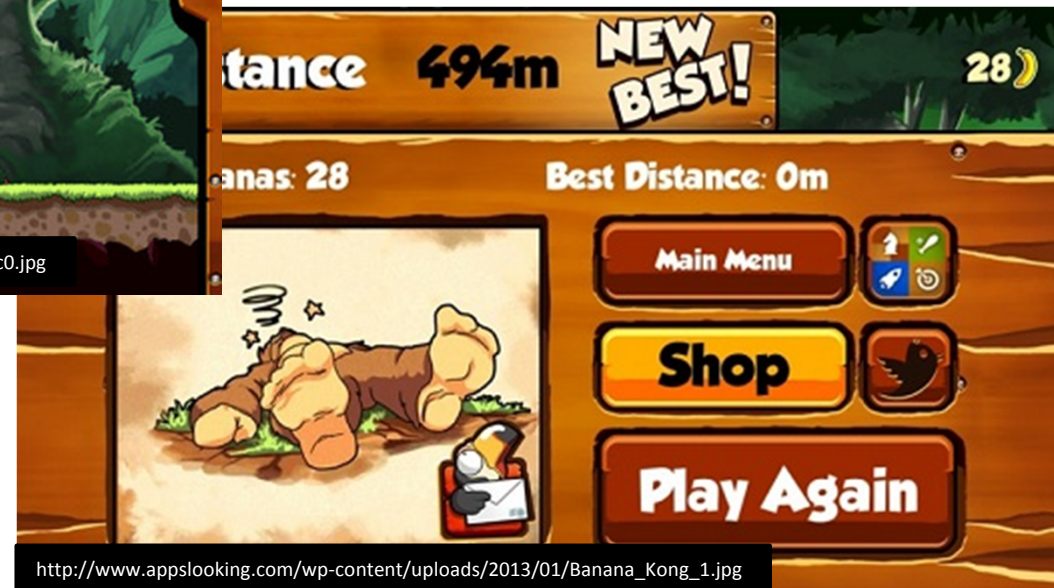
# Inhaltliche Anforderungen

- Gameplay
  - Spiel wird mit der Zeit schwieriger
    - Ziel: Distanz maximieren – So lange wie möglich am Leben bleiben
  - Gegenstände sammeln (Münzen, Bananen etc.)
    - Können z.B. als Währung in integriertem Store verwendet werden
  - Hindernisse, evtl. Verfolger
    - Ausweich-/Zerstörungsmanöver
    - Power-Ups
    - Etc.

# Nicht-funktionale Anforderungen

- Animation und Interaktion
- Eigenes gestalterisches Thema
- Besonderes Aufgabenmerk auf das Feeling
  - Richtiger Schwierigkeitsgrad
  - Gute Benutzbarkeit
  - Spaß am Spiel
- Flexible Anwendungsstruktur
- Robustheit, Fehlerfreiheit
- Ergebnis soll publizierbar sein
  - Kein urheberrechtlich geschütztes Material verwenden (gilt natürlich auch für Code)
  - Abgeschlossenheit

# Fließendes Spielkonzept





# Weitere Entwicklungswerkzeuge



- GIT-Repository zur Verwaltung aller für das Projekt relevanten Dateien.
- Zur Umsetzung des Test-First-Ansatzes:
  - QUnit: <http://qunitjs.com/>

# Die ersten Schritte

- Teams bilden
- Teamaufgaben (vorläufig) verteilen
- GIT-Repositories und GitHub-Profile anlegen
- Informationen zum Originalspiel sowie den anderen Beispielspielen beschaffen
- Planning Game

# Abgabe der Hausaufgabe

- Heute Abend
- Über UniWorX
- Alle Quellen
- Als ZIP-Archiv (vorname.nachname)
  - Keine Ordnerstruktur



# FRAGEN?

# Abschlusspräsentation



- Freitag
  - Jedes Team präsentiert ihr Spiel
    - Zuerst Folien, dann Live-Demo
    - Insgesamt ca. 15-20 Minuten pro Team

# Präsentationsfolien

- Entwicklungsideen
- Ansätze, grundlegende Ideen
- Verwendete Technologien
- Beschreibung der Spielidee
- Schwierigkeiten/Erfolge bei der Entwicklung
- Anwendung der verschiedenen XP-Techniken
- Lustige Bugs
- Einschätzung zur Eignung der verwendeten Technologie für die Spieleprogrammierung
- Sonstiges (Ideen, Verbesserungsvorschläge)

# Abgabe

- Abgabe einer endgültigen Version
  - Erfüllt alle Anforderungen
  - Alle Quelldateien (HTML, JS, Sounds, Grafiken, Schriften)
  - Publizieren einer Spielwebseite
- Eventuell benötigte 3rd-Party-Software
- Fertigstellung bis Ende nächster Woche