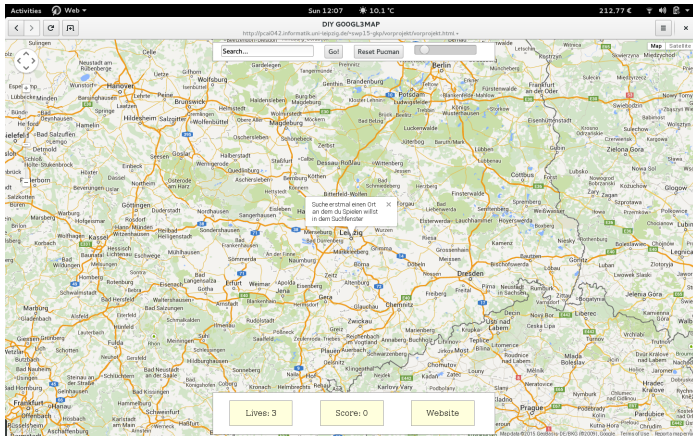


Kartenbasiertes Multiplayerspiel

- Retrospektive
- Perspektive
 - Phaser
 - Struktur- und Entwurfsprinzipien
 - Überblick Implementierungsphase
 - Qualitätssicherung

Produktübersicht -Retrospektiv



- Zugriff über Spiele-Website auf das Programm
- Suchfeld
- Reset und Lautstärke
- Anzeige der Highscore und der verbleibenden Leben



PHASER FEATURES

WEBGL & CANVAS

PRELOADER

PHYSICS

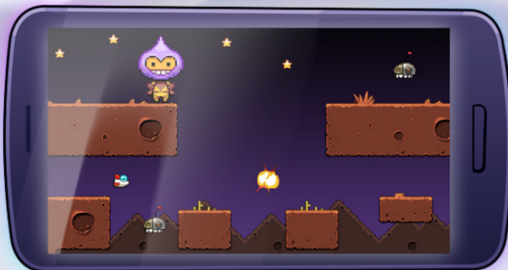
SPRITES

GROUPS

ANIMATION

PARTICLES

CAMERA



INPUT

SOUND

TILEMAPS

DEVICE SCALING

PLUGIN SYSTEM

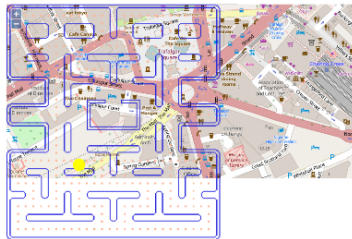
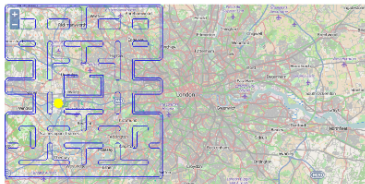
MOBILE BROWSER

DEVELOPER SUPPORT

BATTLE TESTED

Persepektive





Proof of Concept:

Spiel läuft als Overlay auf beliebiger Karte

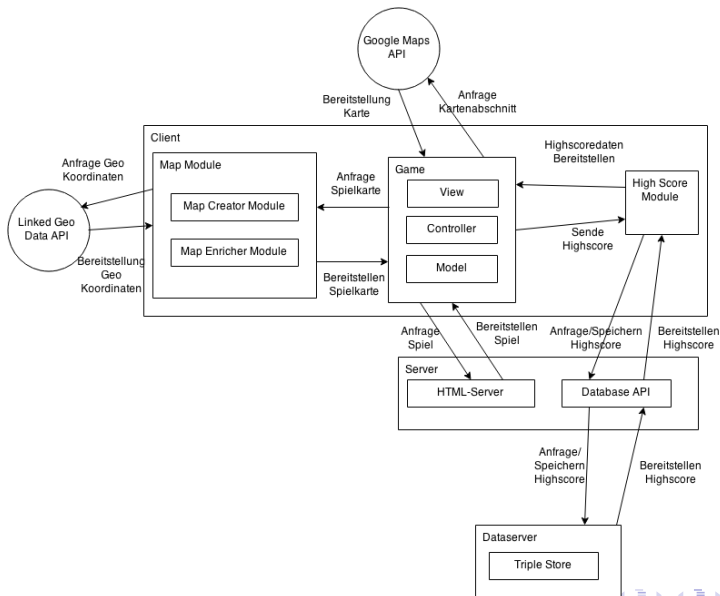
Pacman ist animiert

Pacman bewegt sich

Kollisionsabfrage momentan noch Gridbasiert.

Mit Mausrad wird Karte sowohl als Spiel gezoomt

Persepektive - Struktur- und Entwurfsprinzipien



- Spieldateien
- Verarbeitung von Highscoreanfragen
- Database API - Weiterleitung von Manipulationen zwischen dem Dataserver und dem Client

angelehnt an das MVC Prinzip, nutzt Phaser

- model
 - Enthält die darzustellenden Daten.
 - Unabhängig von den anderen Schichten.
- view
 - Darstellung der benötigten Daten aus dem Modell.
 - Entgegennahme von Benutzerinteraktionen
- controller
 - Verwaltet die Präsentation
 - und Benutzeraktionen.

- Highscore Modul
 - Verwaltet die Highscores der Spieler.
 - Anfragen an den Server zum Triplestore.
 -
- Linked GeoData API
 - Extrahiert aus OpenStreetMap die Geo-Daten für das Erstellen des Spiel-Levels.
- Map Module
 - Map Creator Modul
 - Bildet den Graphen des Levels aus den Straßenzügen.
 - Map Enricher Modul
 - Bindet semantische Daten in die Kartenerstellung ein.
 - Geo-Daten die von der GeoData API.

Übernimmt das Anzeigen der Karte sowie das Suchen des eingegebenen Ortes.

Die integrierten Steuerelemente sind bei der Suche nach einem geeigneten Spielort verfügbar, danach werden die Elemente deaktiviert um einen flüssigen Spielbetrieb zu gewährleisten.

kurzer Überblick über die Implementierungsphase

- ab Woche 1
 - Leveldarstellung
 - Vorprojekt in Phaser
- ab Woche 2
 - Mapmodul
- ab Woche 3
 - Semantische Daten
- ab Woche 4
 - Geister
 - Powerups
- ab Woche 5
 - Alles war auf der Strecke blieb.
 - Debuggen

Produktqualität	Relevant	Nicht relevant
Funktionalität	✓	
Zuverlässigkeit		✓
Benutzbarkeit	✓	
Effizienz		✓
Änderbarkeit	✓	
Übertragbarkeit	✓	

- Code in Java Script
- -> Code Conventions für Java Script
- <http://javascript.crockford.com/code.html>
- inspiriert durch die Sun document Code Conventions für Java
- beinhaltet auch Quelltextdokumentation

Beispielcode

```
/**
 * Sets the filters for the displayObject.
 * * IMPORTANT: This is a WebGL only feature and will be ignored by the canvas renderer.
 * To remove filters simply set this property to 'null'
 * @property filters
 * @type Array(Filter)
 */
Object.defineProperty(PIXI.DisplayObject.prototype, 'filters', {
  get: function() {
    return this._filters;
  },
  set: function(value) {
    if (value){
      // now put all the passes in one place..
      var passes = [];
      for (var i = 0; i < value.length; i++){
        var filterPasses = value[i].passes;
        for (var j = 0; j < filterPasses.length; j++){
          passes.push(filterPasses[j]);
        }
      }
      // TODO change this as it is legacy
      this._filterBlock = {target: this, filterPasses: passes};
    }
    this._filters = value;
  }
});
```

F R A G E N ?