JoHanna Kirchmaier

Anja Bergmann

Tabea Halmschlager

HATman

Project Work 1

ITM14

Content

[Previously on HATman (allgemeine Zusammenfassung, noch überarbeiten, bla) 3](#_Toc452920356)

[Introduction 4](#_Toc452920357)

[Approach 5](#_Toc452920358)

[Cocos2D 5](#_Toc452920359)

[Arianne 5](#_Toc452920360)

[Conclusion 6](#_Toc452920361)

# Previously on HATman (allgemeine Zusammenfassung, noch überarbeiten, bla)

The purpose of our project was to show if it is possible to use a higher programming language (Python, to be specific) to write a simple multiplayer game with 2D graphic. We planned to re-implement the classic game Pacman. In our version – called HATman – it should be possible to play not only the HATman itself, but also the ghosts.

Our first attempt was to use the framework Cocos2D. We managed to make the game playable, but had problems when it came to implementing the multiplayer mode. We found a lot of frameworks that provided the functionality we needed, but unfortunately they were all written in Python 2 and therefore not 100% compatible with Python 3, which we choose because we want to work with the latest versions.

This is why we discarded our first attempt and started all over. This time we intended to use the framework Arianne.

VERSUCH Nr. 2 MARUAUROA

*Nach dieser Anleitung gemacht:* [*https://stendhalgame.org/wiki/InitialStepsWithMarauroa*](https://stendhalgame.org/wiki/InitialStepsWithMarauroa)

<https://github.com/arianne/marauroa> downloaden

Ant build:

* ant dist

Projekt in Eclipse-Workspace imporiteren

Datenbank vorbereiten:

* mysql –user root –password
* Password: toor
* Create database marauroa
* Grant all on marauroa.\* to hatman@localhost identified by ‚keines‘
* Exit

Server.ini erstellen (siehe <https://stendhalgame.org/wiki/Configuration_file_server.ini>)

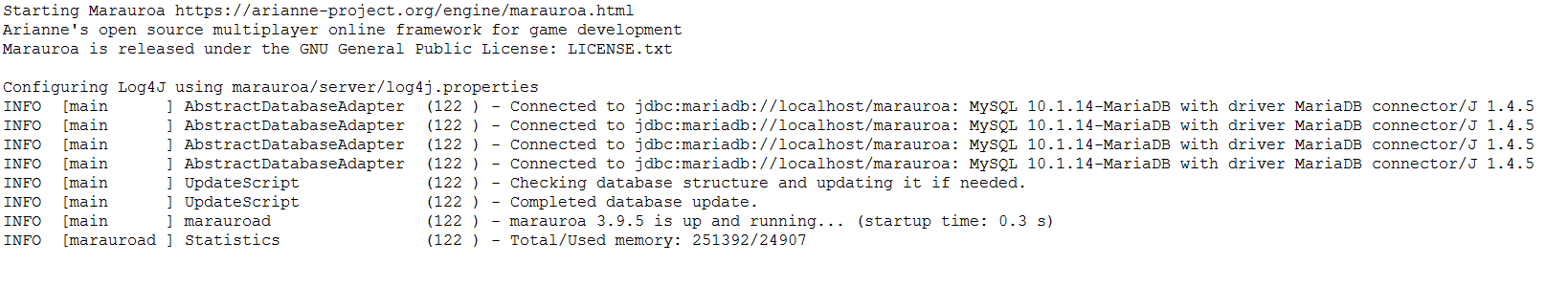
Marauroa.ini erstellen (siehe <https://stendhalgame.org/wiki/How_to_write_games_using_Arianne_in_Python#How_to_write_a_game_using_Arianne>)

MariaDB-Connector downloaden <https://downloads.mariadb.org/connector-java/>

MariaDB-Connector in den Libs-Folder tun (jar-File)

In Eclipse Rechtsklick auf das Projekt > Properties > Java Build Path > Libraries > Add Jars > MariaDB-Connector-Jar hinzufügen > Ok

Src > marauroa.server.mararoad.java starten; Output sollte ungefähr so aussehen:



Wenn nicht: Pro-Tipp: Fehlermeldungen **lesen**!

//Todo: Allgemeines Zeug über Cocos2D schreiben

//Todo: Allgemeines Zeug über Arianne schreiben

//Todo: Sturkturierung

//Todo: Screenshots einfügen

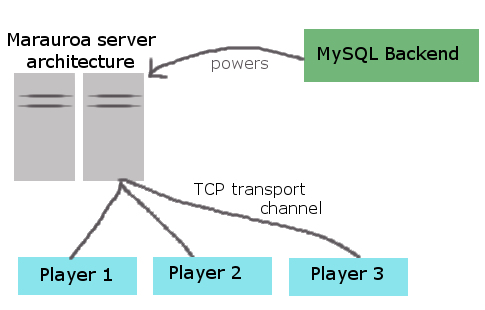
//Todo: Anjas gefailtes Englisch verbessern

# Introduction

# Approach

## Cocos2D

## Arianne



Arianne is an open source, multi-player online framework and engine to develop turn based and real time games. Its design provides a portable, robust and multithreaded server architecture, which is called “Marauroa”. You can use it to easily create your games on. Python is used for writing game rules, but you have to use Java for the server. Additionally, a MySQL backend powers the server and the server communicates with players using a TCP transport channel. The reference clients are coded using Java and C in order to achieve maximum portability. Moreover, Arianne’s server is totally client agnostic, which makes it very modifiable.

Since 1999, Arianne has been in development and its key concept has always been KISS: Keep it simple, stupid. It has evolved from a tiny application written in pseudo-C++ to a powerful, expandable server framework running on the Java platform and a portable client framework written in bare C to allow total portability of Arianne’s clients.

**Marauroa**

Marauroa is based on a kind of philosophy that is called Action/Perception. Each turn one or more clients can ask the server to do any action on their behalf using *action* requests. A *perception* is then sent to the clients in order to explain to them what they perceive of the world around them, and thus the result of their action. Like already stated above, Marauroa is totally game agnostic and makes very little assumptions about what you are trying to do with it. Because of this, it allows great freedom to create whatever type of game you want.

# Conclusion