

The Veripeditus AR Game Framework

Enabling everyone to freely create Augmented Reality Games

Eike Tim Jesinghaus, Dominik George
FOSDEM 2017 ■ 2017-02-04



Veripeditus

The Free
AR Game Framework
for Everyone

Kangoo



Talk

```
from veripeditus import framework as f
class AnyChild(f.NPC):
    spawn_osm = {"building": "school"}
    default_image = "schoolkid"

    def on_talk(self):
        return self.say(
            "I can make AR Games :!)"
```

Today's talk

Who we are

About Veripeditus

How to create games

Framework development and design

State and future

You and us and Veripeditus

Eike

- ▶ Eike Tim Jesinghaus
- ▶ 15 year-old student from Germany
- ▶ Python programmer since age 11
- ▶ Python tutor at Teckids e.V.

Nik

- ▶ Dominik George
- ▶ 26 years old
- ▶ Dayjob: Software developer at tarent solutions GmbH
- ▶ Head of Teckids e.V., the FOSS youth organisation in Germany

Today's talk

Who we are

About Veripeditus

How to create games

Framework development and design

State and future

You and us and Veripeditus

Today's talk

Who we are

About Veripeditus

How to create games

Framework development and design

State and future

You and us and Veripeditus

What is Veripeditus?

- ▶ A free software AR game creation framework
- ▶ Allows easy creation of augmented reality games
- ▶ Run your own server with different worlds and different games in them
- ▶ Play the games using any mobile browser

How it got started

- ▶ We were looking for a new fun project
- ▶ AR gaming seemed to become popular (cf. that monster catching thingy)
- ▶ We wanted to create something like that without the privacy hassle
- ▶ Many different ideas - why not make it a framework?

New goals after some time

- ▶ The framework got more and more powerful
- ▶ Developing games with it got much easier than expected
- ▶ So now:
 - ▶ Make it as accessible as possible
 - ▶ Enable use in various fields besides just-for-fun gaming
 - ▶ More about that later!

Today's talk

Who we are

About Veripeditus

How to create games

Framework development and design

State and future

You and us and Veripeditus

Before we discuss more, let's look at how to make a game!

Basic workflow for making a game

1. Plan any game objects and their features (position, names, images, . . .
2. Creating a game is easy - plan game objects and their features, the plot and other logic of the game and implement it in pure python class definitions.
3. Draft a game plot, activities, other logic
4. Put everything together in object/class definitions in pure Python

A first tiny game

```
1  from veripeditus import framework as f
2
3  class Player(f.Player):
4      pass
5
6  class AnyChild(f.NPC):
7      spawn_osm = {"building": "school"}
8      default_image = "schoolkid"
9
10     def on_talk(self):
11         return self.say("I can make AR games :)!")
```

Quiz time: What have we got here?

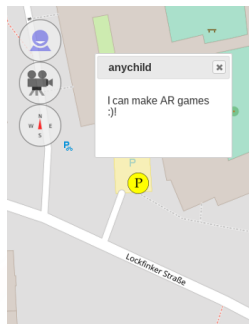
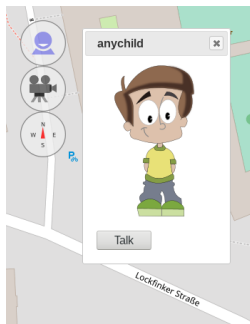
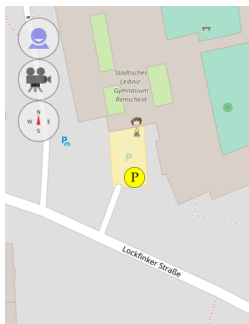
Quiz time: What have we got here?

Answers:

- ▶ A Player object with no special behaviour
- ▶ An NPC (non-player character) named AnyChild
 - ▶ Spawning at every building tagged a school on OpenStreetMap
 - ▶ Having the image of a schoolkid
 - ▶ Saying "I can make AR games :)" when asked

Isn't that cool?

First tiny game, in pictures



First tiny game, in pictures



Today's talk

Who we are

About Veripeditus

How to create games

Framework development and design

State and future

You and us and Veripeditus

Framework development and design goals

- ▶ Make game creation as easy and accessible as possible
- ▶ Still allow adding arbitrarily complex code
 - ▶ No DSL - game cartridges are plain and honest Python packages
 - ▶ Don't get in the way while still being helpful
- ▶ Provide all basic objects and functions to derive from
 - ▶ Player, Items, NPCs, Locations, ...
 - ▶ Dialogues, item management, inventory management, interaction, ...
- ▶ Tight and simple integration with OpenStreetMap

Technologies used

- ▶ Backend
 - ▶ Python 3
 - ▶ Flask web framework with Flask-Restless, and others
 - ▶ SQLAlchemy and OSMAchemy
- ▶ Frontend
 - ▶ HTML 5, JavaScript, CSS 3
 - ▶ JQuery and JQuery-UI
 - ▶ Leaflet
- ▶ Simple RESTful HTTP API

Today's talk

Who we are

About Veripeditus

How to create games

Framework development and design

State and future

You and us and Veripeditus

What's already there

- ▶ Framework features and objects
 - ▶ Items and NPCs
 - ▶ Spawning at lat/lon or OSM objects
 - ▶ Interaction (collecting items and talking to NPCs)
- ▶ Backend technology with advanced API
- ▶ (Rudimentary) reference web application for gameplay
 - ▶ Map view
 - ▶ Camera / 3D view
- ▶ Advanced debugging/testing mode for easier development

Current uses of Veripeditus

- ▶ Teckids e.V. school project at Ganztagsrealschule Odenthal
 - ▶ Public school in Germany
 - ▶ 20 students between 12 and 15 years
 - ▶ Weekly coding lessons
 - ▶ Drafting, describing and planning of AR games
 - ▶ Actively contributing feature and change requests

Many different use fields

- ▶ Gaming, of course. . .
 - ▶ Create just-for-fun outdoor games with AR aspects
- ▶ Educational use
 - ▶ Coding lessons
 - ▶ Basic coding
 - ▶ Object-oriented modelling
 - ▶ Databases, APIs, and much more
 - ▶ Interdisciplinary use
 - ▶ Educational games for history, arts, . . .
 - ▶ Fun at field trips
- ▶ Tourism and attractions
 - ▶ Interactive stories in open air museums, . . .

Spontaneous question: Any quick ideas here?

Coming up / Roadmap

- ▶ Improvements to web application
- ▶ Interactive / live game code editor
- ▶ More detailed and diverse game object interactions
- ▶ WebGL 3D models
- ▶ Sound support
- ▶ HUD defined by games
- ▶ 3D interaction with game objects (aiming, etc.)
- ▶ ...

Scheduled date for public beta release: 2017-03-11!

Today's talk

Who we are

About Veripeditus

How to create games

Framework development and design

State and future

You and us and Veripeditus

What we can do for you

- ▶ Make Veripeditus a framework for YOU to use
- ▶ Help you implement game ideas in your use fields

What you can do for us to help us help you

- ▶ Report your ideas, feature requests and bugs
- ▶ Help us if you are (or want to be) a JavaScript or CSS guru
- ▶ Test Veripeditus on many mobile devices
- ▶ Tell us stories from active usage in different fields

Where to find us

- ▶ GitHub:
<https://github.com/Veripeditus/veripeditus-server>
 - ▶ Please find up-to-date contact and testing information there!
 - ▶ Go mad reporting ideas, questions, requests and bugs
- ▶ Twitter: @VeripeditusTeam
 - ▶ Please follow us to receive updates!
- ▶ E-mail: team@veripeditus.org

Some numbers

- ▶ Code size: >2 kloc (>3.5 kloc including comments)
- ▶ Commits: exactly 1000 ☺!
- ▶ Age: First birthday was on 2017-01-19
- ▶ Active contributors: 3
- ▶ Tasks in issue tracker: 33 open, 48 closed

FOSDEM game

Go to <https://nightly.veripeditus.org> with any mobile browser.

- ▶ Small testing game on the campus
- ▶ Game may be slow and may not work on all devices
 - ▶ Camera view only works on Firefox on Android
 - ▶ Map view should work on most browsers
- ▶ Please ping us via e-mail or find us somewhere if you have bugs to look at
- ▶ Find the full code at:
<https://github.com/Veripeditus/game-fosdem17>

Start by finding the guy at the entrance of building K and listen carefully ☺!

Thank you!

Time for questions and discussion...

We also want to thank...

Python ▪ Flask ▪ Flask-Restless ▪ SQLAlchemy ▪ jQuery ▪ Leaflet ▪
OpenStreetMap ▪ mirabilos ▪ Debian ▪ and others we forgot to mention!

Invitation:

OSMAIchemy workshop ▪ Tomorrow, 12:30 ▪ Geospatial
Devroom

Proudly made with L^AT_EX beamer and without fancy office products ▪ Sources at:

<https://github.com/Veripeditus/veripeditus-server/tree/master/doc/presentations>