Installation unter IIS7.5

# Installation von Helicon Zoo Manager

Mittels Web Platform Installer Helicon Zoo Manager und Python Hosting Package installieren.  
<http://www.helicontech.com/zoo/install.html>

# Installation von Python Template

Mittels Web Platform Installer Python Project installieren.  
<http://www.helicontech.com/zoo/docs/python-wsgi.html>

Den Ordnerinhalt des neu erstellten Ordners bis auf die Datei „web.config“ löschen.

# Installation und Konfiguration des Stadtgedächtnisses

Das Stadtgedächtnis-Repository in den Ordner clonen, den MSSQL-Branch wählen und Submodule updaten.

git clone https://github.com/fraunhoferfokus/mobile-city-memory

git checkout mssql

git submodule init

git submodule update

In der Datei “Mobiles\_Stadtgedaechtnis/settings.py” folgende Settings anpassen:

ADMINS = (

(‚Name‘, ‚E-Mail-Adresse‘), # Name und E-Mail-Adresse eines Admins

)

ALLOWED\_HOSTS = [

‚xxx.xxx.xxx.xxx‘,‘localhost‘,‘xxx.de‘ # Erlaubte http-Hosts

# Django liefert einen 400

# Bad Request, wenn der # aufgerufene Host nicht # in dieser Liste ist

]

DATABASES = {

'default': {

'ENGINE': 'sqlserver\_ado', # MSSQL

'NAME': 'stadtgedaechtnis', # Name der Datenbank

'USER': '', # Benutzername

'PASSWORD': '', # Passwort

'HOST': '127.0.0.1', #

'PORT': '', #

'OPTIONS': {

'provider': 'SQLNCLI10',

}

}

}

MEDIA\_ROOT = “C:/Server/…” # Absoluter Pfad (mit Vorwärts- # Slash!), in dem die # gespeicherten Medien liegen # sollen

STATIC\_ROOT = „C:/Server/…“ # Wie MEDIA\_ROOT für statische

# Dateien

Die Datei “Mobiles\_Stadtgedaechtnis/local\_settings.template.py” in “local\_settings.py” umbenennen und die dortigen Hinweise beachten.

Als AUTHENTICATION\_MODE „moderation“ auswählen.

Im Server-Verzeichnis die Datei „web.config“ wie folgt anpassen:

<?xml version="1.0" encoding="UTF-8"?>

<configuration>

<system.webServer>

<heliconZoo>

<clear />

<application name="python.project">

<environmentVariables>

<add name="VIRTUAL\_ENV" value="%APPL\_PHYSICAL\_PATH%\venv" />

<add name="PATH" value="%APPL\_PHYSICAL\_PATH%\venv\Scripts;%PATH%" />

<add name="PYTHONPATH" value="%APPL\_PHYSICAL\_PATH%\venv\lib\site-packages;%APPL\_PHYSICAL\_PATH%\venv\lib;%APPL\_PHYSICAL\_PATH%;%APPL\_PHYSICAL\_PATH%\Mobiles\_Stadtgedaechtnis" />

<add name="WSGI\_APP" value="Mobiles\_Stadtgedaechtnis.wsgi.application" />

<add name="CONSOLE\_URL" value="console" />

<add name="DEPLOY\_FILE" value="deploy.py" />

<add name="ERROR\_LOG\_DIR" value="log" />

</environmentVariables>

</application>

</heliconZoo>

<handlers>

<remove name="python.project#x86" />

<remove name="python.project#x64" />

<add name="python.project#x64" path="\*" verb="\*" modules="HeliconZoo\_x64" scriptProcessor="python.2.7.wsgi" resourceType="Unspecified" requireAccess="Script" preCondition="bitness64" />

<add name="python.project#x86" path="\*" verb="\*" modules="HeliconZoo\_x86" scriptProcessor="python.2.7.wsgi" resourceType="Unspecified" requireAccess="Script" preCondition="bitness32" />

<!-- Django or WSGI application over fastcgi -->

<!-- HTTP backend (Tornado) over http -->

<!--

<add name="python.project#x86" scriptProcessor="python.2.7.http" path="\*" verb="\*" modules="HeliconZoo\_x86" preCondition="bitness32" resourceType="Unspecified" requireAccess="Script" />

<add name="python.project#x64" scriptProcessor="python.2.7.http" path="\*" verb="\*" modules="HeliconZoo\_x64" preCondition="bitness64" resourceType="Unspecified" requireAccess="Script" />

-->

<!-- HTTP backend (Twisted) over http -->

<!--

<add name="python.project#x86" scriptProcessor="python.2.7.twisted" path="\*" verb="\*" modules="HeliconZoo\_x86" preCondition="bitness32" resourceType="Unspecified" requireAccess="Script" />

<add name="python.project#x64" scriptProcessor="python.2.7.twisted" path="\*" verb="\*" modules="HeliconZoo\_x64" preCondition="bitness64" resourceType="Unspecified" requireAccess="Script" />

-->

</handlers>

</system.webServer>

<system.web>

<authentication mode="Windows" />

<identity impersonate="false" />

</system.web>

</configuration>

Die PyWin32-Version 219 von <http://sourceforge.net/projects/pywin32/files/pywin32/Build%20219/> herunterladen und speichern.

Eine Konsole im Server-Verzeichnis öffnen.

Eine virtuelle Python-Umgebung im Ordner „venv“ erstellen und aktivieren (virtualenv.exe befindet sich im Python-Installationsordner unter \Scripts)

virtualenv venv

venv\Scripts\activate

PyWin32 und das Stadtgedächtnis in der virtuellen Umgebung installieren.

easy\_install PyWin32.exe

python setup.py install

Die statischen Dateien an den gewünschten Ort kopieren.

python manage.py collectstatic