

Deployment

Contents

1. Azure.....	2
1.1. Visual Studio Account erstellen.....	2
1.2. Azure Portal öffnen	2
1.3. Ressource „SendGrid“ erstellen	2
1.4. Ressource „SQL-Datenbank“ erstellen	3
1.5. Ressource „Web App“ erstellen (Service)	6
1.6. Ressource „Web App“ erstellen (Application)	8
1.7. Web Apps für Java konfigurieren	9
2. Datenbankschema	11
3. Eclipse	12
3.1. Eclipse einrichten	12
3.2. Serverdaten einrichten.....	13
3.3. Applikationen ausführen	13
3.4. Build erstellen / Deployment	13
4. FTP	15

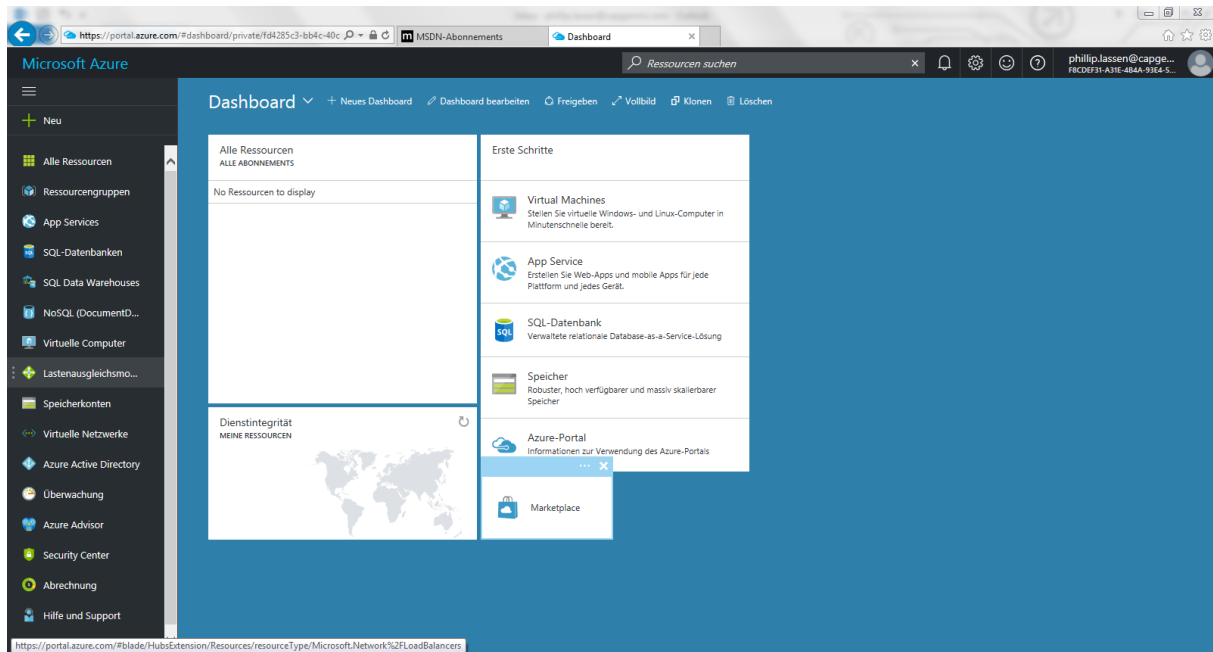
1. Azure

1.1. Visual Studio Account erstellen

- Support kontaktieren

1.2. Azure Portal öffnen

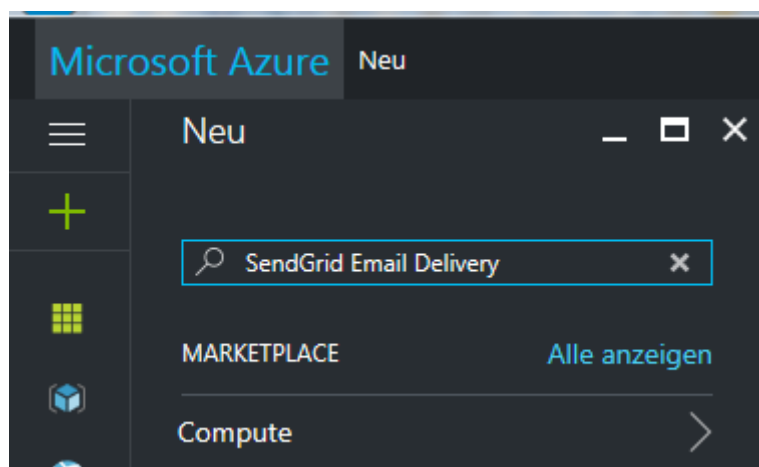
- Azure öffnen (portal.azure.com)



01_azure_home.png

1.3. Ressource „SendGrid“ erstellen

- Klick auf „+ Neu“
- Suche nach „SendGrid Email Delivery“



02_sendgrid_search.png

- “Erstellen”
- Erforderliche Daten laut Screenshot

Create a New SendGrid ...

CREATE

Name

MailService

Password

Confirm Password

Abonnement

Visual Studio Enterprise – MPN

Resource group

Neues Element erstellen

Vorhandenes Element verwenden

PublicRatings

Pricing tier

free

Promotion Code

Contact Information

Completed.

Rechtliche Bedingungen

Rechtliche Bedingungen akzeptiert

An Dashboard anheften

Erstellen

Automatisierungsoptionen

03_sendgrid_create.png

- Die Daten für den Mail-Versand finden sich in „Configurations“ und werden später benötigt

MailService

SendGrid Account

Manage

Delete

Change passw...

Reset Password

Zusammenfassung

Ressourcengruppe

PublicRatings

Status

Running

Standort

USA Mitte

Abonnementname

Visual Studio Enterprise – MPN

Abonnement-ID

2dfd39d0-a8f8-40f3-88f5-bfc4bf93dfeb

Alle Einstellungen

Pricing Information

Pricing tier:

FREE

F1

25,000 emails/month

Settings

Filtereinstellungen

SUPPORT + PROBLEMBEHANDLUNG

Aktivitätsprotokoll

ALLGEMEIN

Properties

Configurations

Contact Information

RESSOURCENVERWALTUNG

Tags

Sperren

Benutzer

Automatisierungsskript

Configurations

MailService

USERNAME

azure_f064328235032fd614f05be4abaa73

PASSWORD

Your Password

SMTP SERVER

smtp.sendgrid.net

04_sendgrid_settings.png

1.4. Ressource „SQL-Datenbank“ erstellen

- Klick auf „+ neu“
- Suche nach „SQL-Datenbank“

SQL-Datenbank		
Ergebnisse		
NAME	HERAUSGEBER	KATEGORIE
SQL-Datenbank	Microsoft	Databases
Web App und SQL	Microsoft	Web und mobil
SQL Server (logischer Server)	Microsoft	
Ressourcengruppe	Microsoft	VM Extensions

05_sql_database_search.png

- „Erstellen“
- Erforderliche Daten laut Screenshot. Name unbedingt „PublicRatingsService“.
- „Auswählen“ und „Erstellen“

SQL-Datenbank

* Datenbankname

PublicRatingsService

* Abonnement

Visual Studio Enterprise – MPN

* Ressourcengruppe

Neues Element erstellen

Vorhandenes Element verwenden

PublicRatings

* Quelle auswählen

Leere Datenbank

* Server

yourdomain (Westeuropa)

Möchten Sie einen elastischen SQL-Pool verwenden?

Ja

Nicht jetzt

* Tarif

Basic

* Sortierung

SQL_Latin1_General_CP1_CI_AS

☒ An Dashboard anheften

Erstellen

Automatisierungsoptionen

Server

+ Neuen Server erstellen

Es wurden keine Server gefunden.

Neuer Server

* Servername

yourdomain

.database.windows.net

* Serveradministratoranmeldung

User

* Kennwort

.....

* Kennwort bestätigen

.....

* Standort

Westeuropa

V12-Server erstellen (aktuelles Update)

Ja

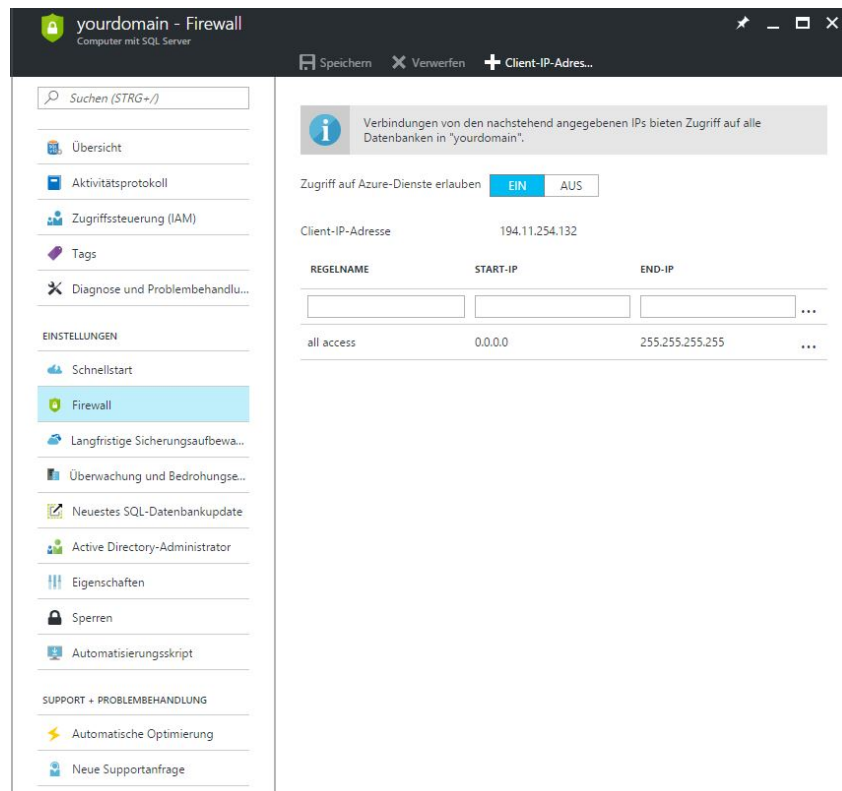
Nein

☒ Azure-Diensten Zugriff auf den Server erlauben

Auswählen

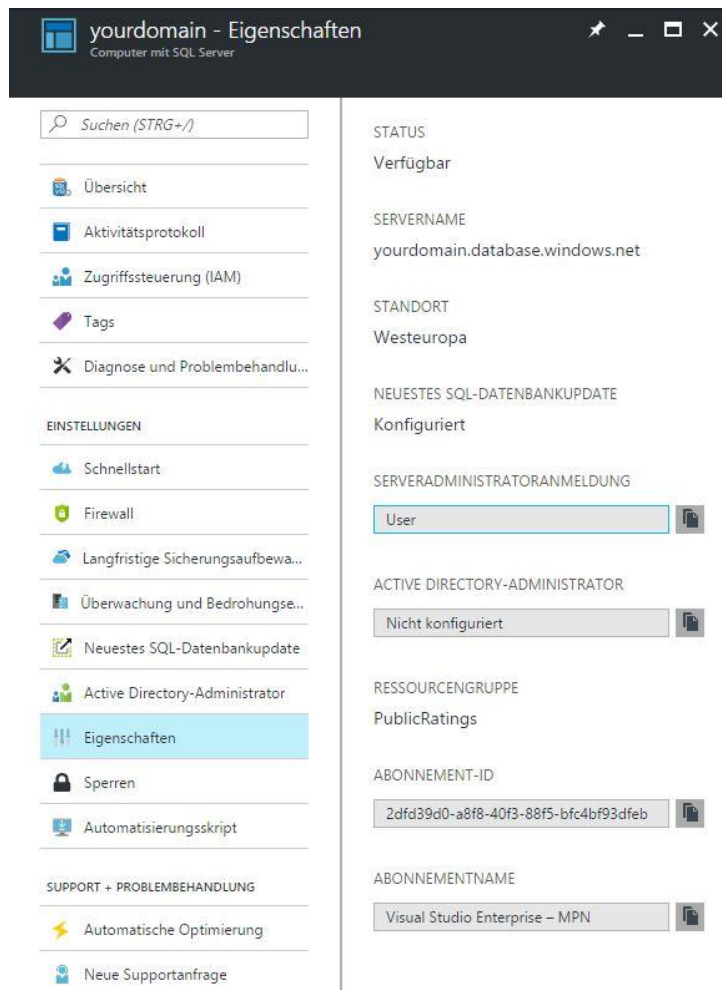
06_sql_database_create.png

- Nach Erstellung auf „Alle Ressourcen“ „SQL-Server“ auswählen und „Firewall“ auswählen. Hier lassen sich die IPs der Clients festlegen, die Zugriff erhalten. Um den Zugriff nicht zu beschränken, alle IPs von 0.0.0.0 bis 255.255.255 freischalten (siehe Screenshot).



07_sql_database_firewall.png

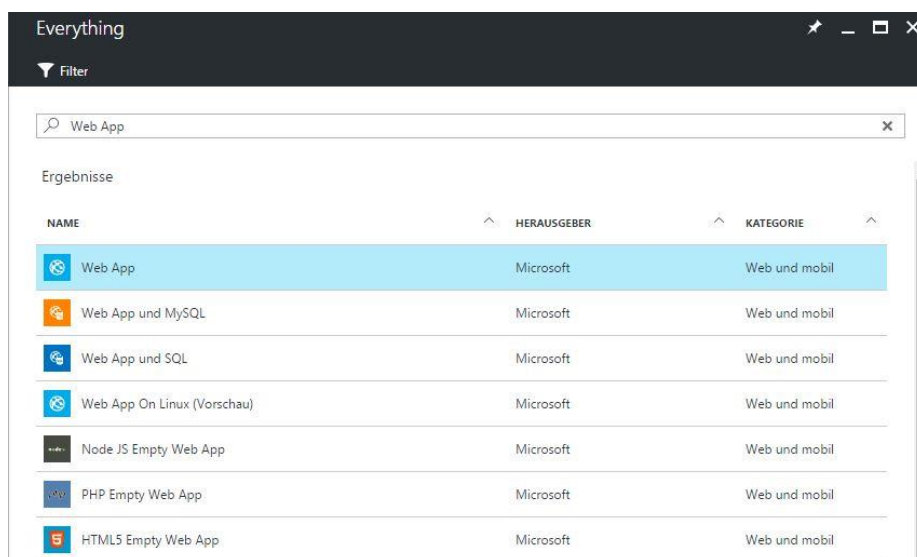
- „Speichern“ klicken
- Die Verbindungsdaten von Datenbank und SQL-Server finden sich jeweils unter „Eigenschaften“ und werden später benötigt.



08_sql_database_settings.png

1.5. Ressource „Web App“ erstellen (Service)

- Klick auf „+ neu“
- Suche nach „Web App“



09_service_search.png

- „Erstellen“
- Erforderliche Daten laut Screenshot

Web App

* App-Name
YourServiceDomain ✓
.azurewebsites.net

* Abonnement
Visual Studio Enterprise – MPN

* Ressourcengruppe
☐ Neues Element erstellen
☒ Vorhandenes Element verwenden
 PublicRatings

* App Service-Plan/Standort
ServicePlan(West Europe)

Application Insights ☐ Ein ☒ Aus

☒ An Dashboard anheften

Erstellen Automatisierungsoptionen

App Service-Plan
Wählen Sie einen Plan für die Web-App aus.

Ein App Service-Plan ist der Container für Ihre App. Die App Service-Einstellungen legen den Standort, die Features, die Kosten und die Computeressourcen fest, die Ihrer App zugeordnet sind.

+ Neu erstellen

ServicePlan(B1) (Neu)
West Europe Neuer Plan

10_service_create.png

- „Auswählen“ und „Erstellen“
- Unter „Anwendungseinstellungen“ „Java 8“ auswählen (siehe Screenshot)

YourServiceDomain - Anwendungseinstellungen
App Service

Suchen (STRG+V)

Übersicht
Aktivitätsprotokoll
Zugriffssteuerung (IAM)
Tags
Diagnose und Problembehan...

APP-BEREITSTELLUNG
Schnellstart
Anmeldeinformationen für Be...
Bereitstellungsslots
Bereitstellungsoptionen
Continuous Delivery (Vorsch...

EINSTELLUNGEN
Anwendungseinstellungen
Authentifizierung/Autonsieru...
Sicherungen
Benutzerdefinierte Domänen

Allgemeine Einstellungen

Die 64-Bit- und die AlwaysOn-Optionen können in den Tarifen "Basic" und höher aktiviert werden. AutoSwap kann nur für den Tarif "Standard" und höher aktiviert werden.

.NET Framework-Version v4.6
PHP-Version 5.6
Java-Version Java 8
Java-Nebenversion Aktuellste
Webcontainer Aktuellste Tomcat 8.5
Python-Version Aus
Plattform 32-Bit 64-Bit
Websockets Aus Ein
Immer aktiv Aus Ein
Verwaltete Pipelineversion Integriert Klassisch

Sie können die Leistung Ihrer zustandslosen Anwendungen verbessern, indem Sie Affinitätscookies deaktivieren. Für zustandsbehaftete Anwendungen sollten Affinitätscookies aktiviert bleiben, um eine bessere Kompatibilität zu gewährleisten. Klicken Sie hier, um weitere Informationen zu erhalten.

11_service_settings.png

- Unter „Anmeldeinformationen für Bereitstellung“ die FTP-Daten festlegen (siehe Screenshot). Sie werden später benötigt. Die FTP-URL und der Nutzernamen sind danach unter „Übersicht“ zu finden.

Suchen (STRG+ /)

Diagnose und Problembehan...

APP-BEREITSTELLUNG

Schnellstart

Anmeldeinformationen für Be...

Bereitstellungsslots

Bereitstellungsoptionen

Continuous Delivery (Vorsch...

Neuer Name und neues Kennwort

Git und FTP können zur Authentifizierung nicht das Konto verwenden, mit dem Sie angemeldet sind. Erstellen Sie einen neuen Benutzernamen und ein neues Kennwort, um diese Technologien zu nutzen.

Verwenden Sie diesen Benutzernamen und dieses Kennwort für die Bereitstellung in beliebigen Apps für alle Ihrem Microsoft Azure-Konto zugeordneten Abonnements.

* FTP/Bereitstellungsbenutzername ⓘ

YourUser ✓

* Kennwort ⓘ

..... ✓

* Kennwort bestätigen ⓘ

..... ✓

12_service_ftp.png

1.6. Ressource „Web App“ erstellen (Application)

- Klick auf „+ neu“
- Suche nach „Web App“
- „Erstellen“
- Erforderliche Daten laut Screenshot

13_application_create.png

- „Auswählen“ und „Erstellen“
- Unter „Anwendungseinstellungen“ „Java 8“ auswählen
- Unter „Anmeldeinformationen für Bereitstellung“ die FTP-Daten. Sie werden später benötigt. Die FTP-URL und der Nutzernamen sind danach unter „Übersicht“ zu finden.

1.7. Web Apps für Java konfigurieren

- In Azure die Ressource auswählen und unter „Entwicklungstools“ auf „Erweiterte Tools“ -> „Gehe Zu“
- Im Kudu-Portal auf „Debug-Console“ -> „CMD“
- Hier in das Verzeichnis „site/wwwroot/“ eine neue Datei „web.config“ anlegen („+“) und bearbeiten
- Folgenden Code einfügen (vorher Namen der .war-Datei eintragen) und speichern:

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
  <system.webServer>
    <handlers>
      <add name="httpPlatformHandler" path="*" verb="*"
modules="httpPlatformHandler" resourceType="Unspecified" />
    </handlers>
    <httpPlatform processPath="%JAVA_HOME%\bin\java.exe"
```

```
arguments="-Djava.net.preferIPv4Stack=true -
Dserver.port=%HTTP_PLATFORM_PORT% -jar
"%HOME%\site\wwwroot\webapps\PublicRatingsService-1.war";">
</httpPlatform>
</system.webServer>
</configuration>
```

Quelle: <https://docs.microsoft.com/en-us/azure/app-service-web/web-sites-java-custom-upload#springboot>

2. Datenbankschema

- „Microsoft SQL Server Management Studio **2016 (EN)**“ herunterladen, installieren und ausführen
- Mit den Verbindungsdaten eine Verbindung zur Datenbank erstellen (Screenshot)



14_ssms_connect.png

- SQL-File „resources/database.sql“ öffnen
- Darauf achten, dass die Datenbank „PubliRatingsService“ ausgewählt ist



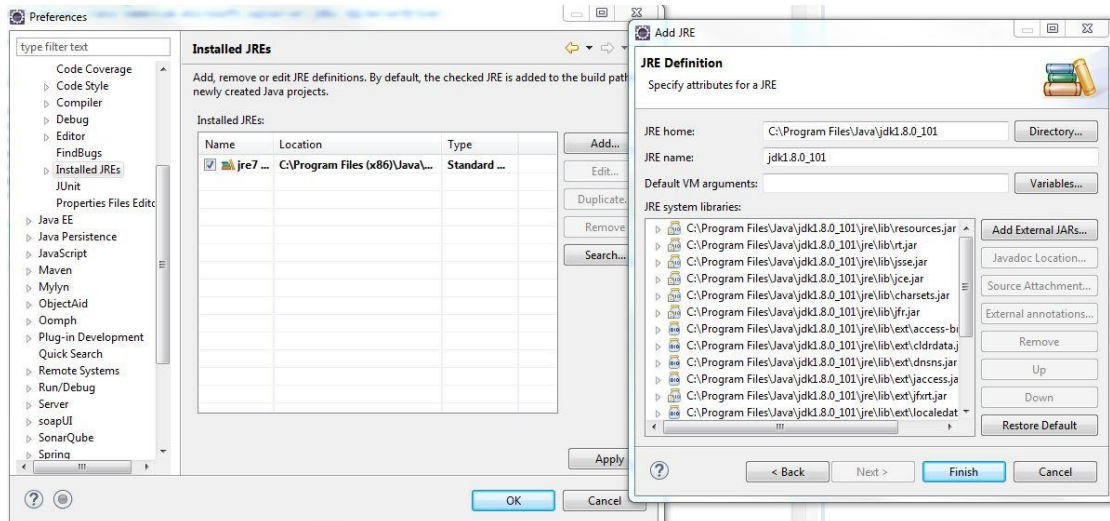
15_ssms_database.png

- Ausführen („Execute“)

3. Eclipse

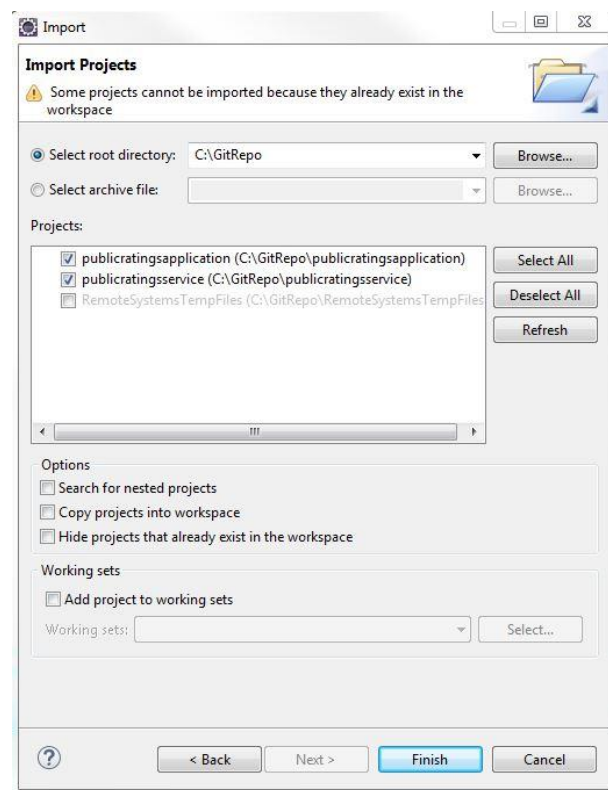
3.1. Eclipse einrichten

- Das Git Repository (<https://seu.sdm.de/pu/smartcity/git/repository.git>) clonen. Wichtig: Der Git Root-Ordner ist gleichbedeutend mit einem Eclipse-Workspace.
- Eclipse und Maven installieren und Eclipse starten. Als Workspace den Git-Ordner auswählen.
- Unter „Window“ -> „Preferences“ -> „Java“ -> „Installed JREs“ das installierte JDK 1.8 als einzige Runtime des Workspace einrichten (Screenshot)



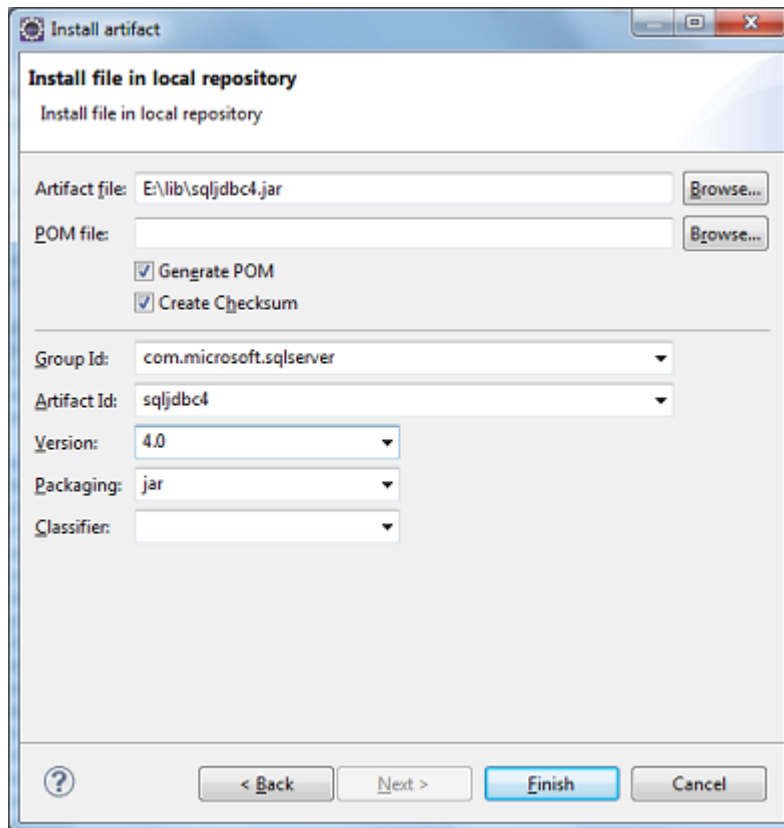
16_eclipse_jre.png

- „File“ -> „Import...“ -> „Existing Projects into Workspace“ -> Git-Ordner als root directory auswählen -> „Fertig“



17_eclipse_import.png

- „File“ -> „Import...“ -> „Maven“ -> “Install or deploy an artifact to a Maven repository”.



18_eclipse_jdbc.png

- Rechtsklick auf eines der Projekte -> „Maven“ -> „Update Project“ -> beide auswählen und „Ok“

3.2. Serverdaten einrichten

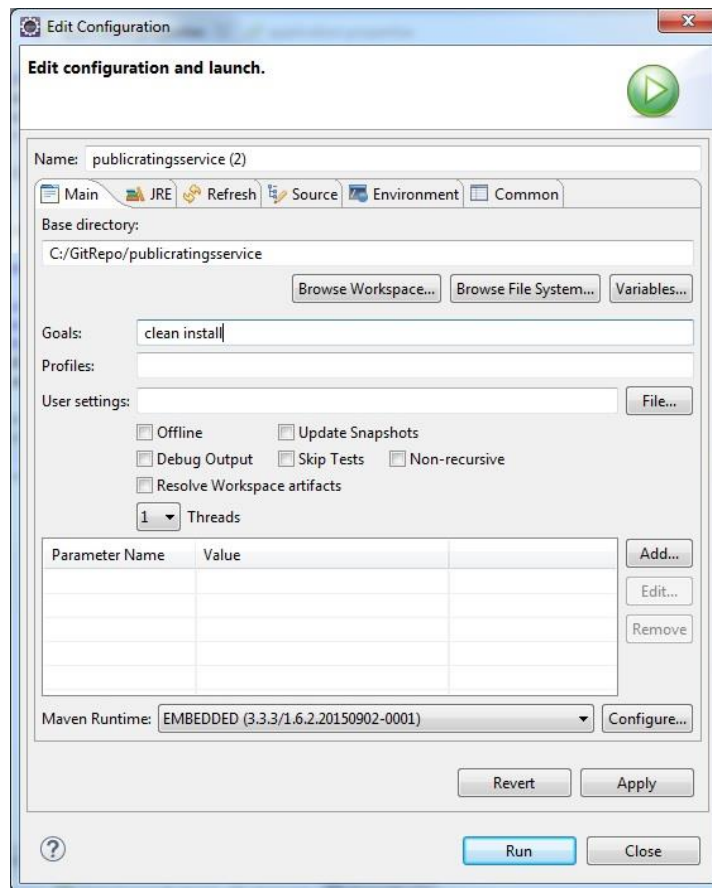
- Die Einstellungen des Service vornehmen in **publicratingservice/Java Resources/src/main/resources/application.properties**
- Die Einstellungen der Application vornehmen in **publicratingsapplication/Java Resources/src/main/resources/application.properties**

3.3. Applikationen ausführen

- Um eine Web App auszuführen, jeweils Rechtsklick -> „Run as..“ -> „Spring Boot App“

3.4. Build erstellen / Deployment

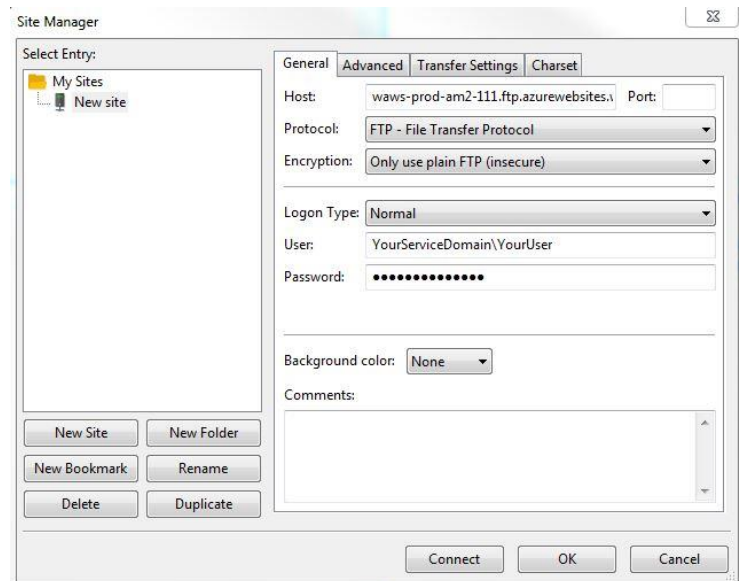
- Projekt Rechtsklick -> Run As... -> 6 Maven Build ... (Screenshot) -> „Run“



19_eclipse_build.png

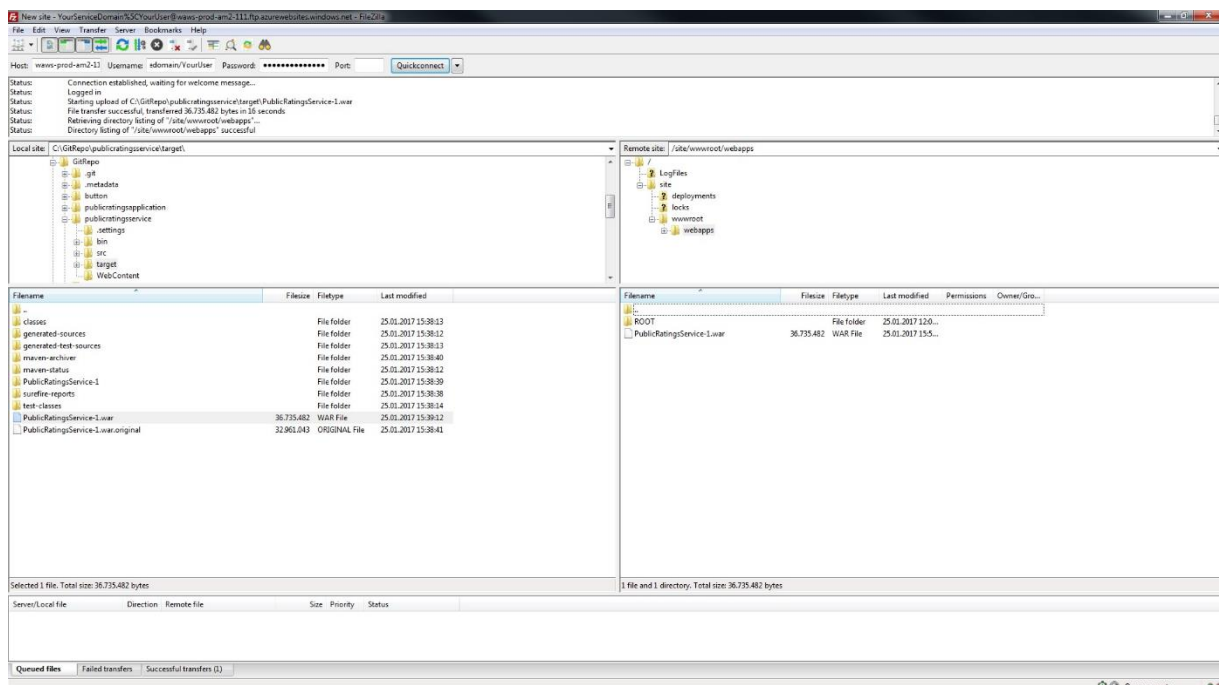
4. FTP

- FTP-Settings laut Azure und Screenshot



21_filezilla_settings.png

- Maven erstellt durch den Build ein „war“-File im „target“-Unterverzeichnis des Projektordners, welches über FTP auf den Server geladen wird. Verzeichnis auf dem Server: „/site/wwwroot/webapps“

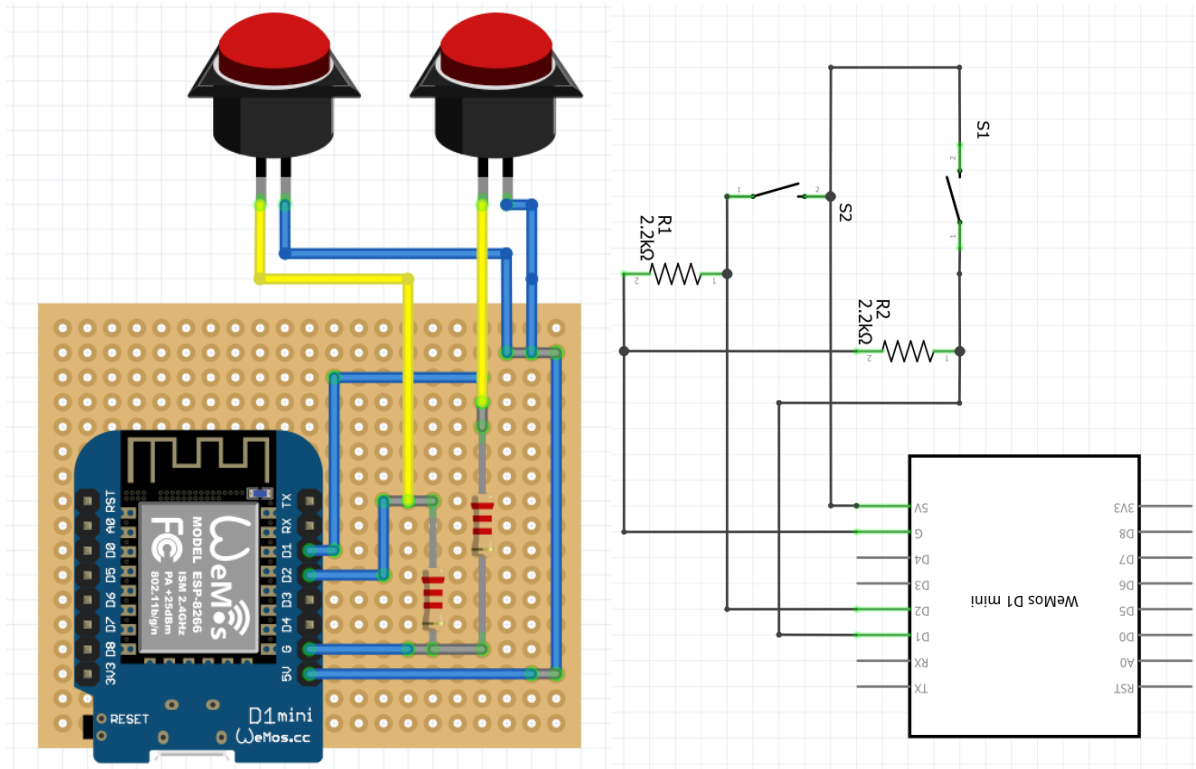


20_filezilla.png

5. Wemos D1 Mini

5.1. Buttons

- Den Wemos D1 Mini laut Schaltplänen mit den Knöpfen verkabeln / löten



wemos_1.png / wemos_2.png

5.2. Arduino IDE

- Download: <https://www.arduino.cc/en/Main/Software>
- Installation
- Wemos D1 Mini anschließen
- Datei öffnen: /button/Touch/TouchNeu.ino
- Einstellungen im Code:
 - Wifi-SSID
 - Wifi-Passwort
 - Service-URL
- Einstellungen in der IDE
 - Tools -> Upload Speed -> 115200
 - Tools -> Board -> Wemos D1 R2 & mini
 - Tools -> Port einstellen
- Upload