Voor de test op Amazon AWS is deze handleiding (verwijzing) gevolgd. Er zijn wel een aantal zaken aangepast geweest om een realistischere opstelling te benaderen. De handleiding begint met het aanmaken van twee S3 Storage Buckets. In dit geval is er gekozen om maar een S3 Storage Bucket aan te maken. Deze zal dienen als een output voor bestanden. Voor de input werd een GitHub repositorie gebruikt. Vervolgens wordt in de handleiding de applicatie gemaakt. Voor dit experiment is er voor de drie Cloud platformen dezelfde applicatie gebruikt. Figuur (verwijzing) toont de hoofdklasse van deze applicatie en figuur (verwijzing) toont de testklasse van deze applicatie. Vervolgens wordt er in de handleiding een YAML-bestand aangemaakt waarin de verschillende stappen voor het compileren en testen van de applicatie gedefinieerd worden.

Dit YAML-bestand heeft op Amazon AWS de naam ‘buildspec.yml’. Figuur (verwijzing) toont dit bestand. Over het aanpassen van dit YAML-bestand is er maar weinig informatie te vinden. Ook is het niet helemaal duidelijk hoe er precies een andere virtuele machine gekozen moet worden. Tijdens het uitvoeren van de configuratie is er gebleken dat de documentatie tot verwarring kan lijden. Dit YAML-bestand bevat altijd vier fases waarin een reeks commando’s uitgevoerd kunnen worden. De eerste fase is de ‘install‘ fase. Hier wordt er gedefinieerd wat voor omgeving gebruikt moet worden. In dit geval gebruikt de handleiding ‘corretto11’. Dit is een Linux gebaseerde machine. De tweede fase is de ‘pre-build’ fase. Hierin kunnen er commando’s uitgevoerd worden om de machine voor te bereiden, repositories te kopiëren, enz. In dit geval staat er een echo-commando als plaatsvulling. De derde stap is de ‘build’ stap. Hierin worden de commando’s geschreven die ervoor moeten zorgen dat de applicatie gecompileerd wordt. In een vierde fase, de ‘post-build’ fase, is er de mogelijkheid om nog commando’s uit te voeren na het compileren van de applicatie. Dit is in dit geval ook ongebruikt. Voor een Java applicatie wordt alles door middel van een pom.xml uitgevoerd. Figuur (verwijzing) toont dit bestand. Het enige dat dus moet gedefinieerd worden in de ‘buildspec.yml’ is het uitvoeren van een herstel op de Java-bestanden.

version: 0.2

phases:

install:

runtime-versions:

java: corretto11

pre\_build:

commands:

- echo Nothing to do in the pre\_build phase...

build:

commands:

- echo Build started on `date`

- mvn install

post\_build:

commands:

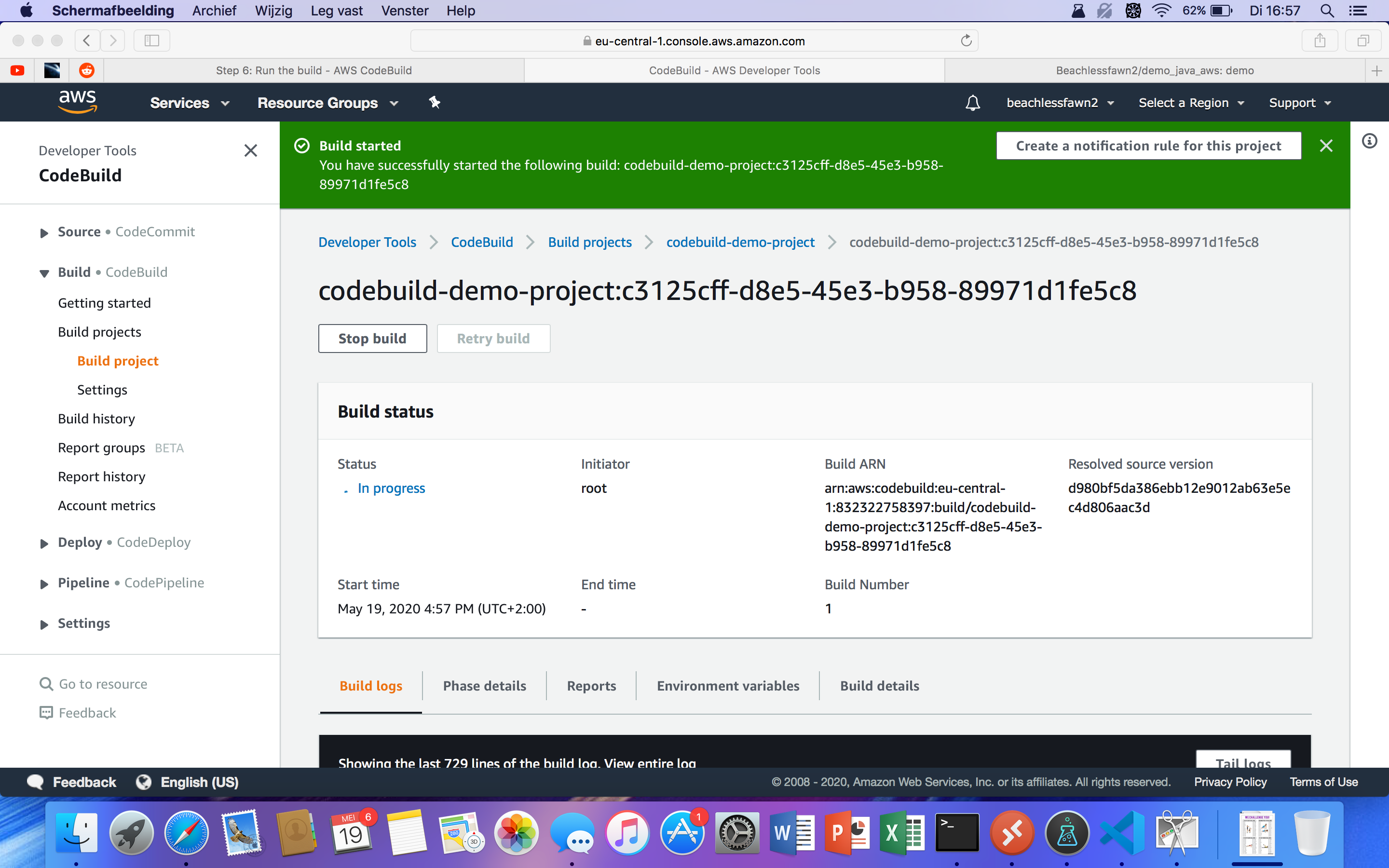
- echo Build completed on `date`

artifacts:

files:

- target/messageUtil-1.0.jar

In een volgende stap wordt in de handleiding de pijpleiding aangemaakt op Amazon AWS. Dit werd volledig gedaan via de online console van Amazon AWS. Een verschil in deze configuratie is het selecteren van een bron voor de broncode. Deze bevind zich nu op GitHub. Figuur (verwijzing) toont de gewijzigde configuratie op AWS. Voor alle ander opties is de handleiding gevolgd. Ook de volgende stappen vanuit de handleiding zijn onveranderd gevolgd. Zo werd er als resultaat in de Bucket een Jar-bestand achtergelaten. Deze was zeer gemakkelijk te downloaden om daarna uit te voeren.



Het volgen van deze handleiding ging niet zonder slag of stoot. Het was in het begin zeer moeilijk om te begrijpen wat ‘buildspec.yml’ nu allemaal kan. Ook heeft Amazon AWS een hele reeks van producten dat het soms moeilijk maakt om te selecteren wat er nu juist nodig is. De tijd die in beslag genomen is in vergelijking met de andere platformen, om de handleiding van het begin tot het einde te volgen wordt getoond in tabel (verwijzing). Hier is ook het opstellen van de applicatie meegerekend en ook het uitvoeren van het resulterende Jar-bestand.

2:75u