Test 1: Blockchain

Test:

Het doel van de test is om te kijken of data goed wordt opgeslagen in de blockchain.

Benodigdheden:

* Draaiende server
* Laptop

Setup:

Login als admin op de webinterface. Upload een bestand en voeg een korte beschrijving toe. Login als consensus en klik op ‘approve’ bij dit block in de pending list. Check of dit block in de blockchain terug te zien is. Download het bestand om te kijken of de inhoud correct is opgeslagen. Bekijk de rauwe blockchain data op hetzelfde ip, port 80/chain. Als dit ook correct is opgeslagen is de test geslaagd.

Code:

<https://github.com/JorisTDP/DIHM>

Edge cases:

* Voeg een foutieve beschrijving toe (zoals het veld van beschrijving leeg laten of gebruik maken van onbekende tekens)
* Upload een bestand dat erg groot is, een rare extensie heeft of een naam met onbekende tekens

Hypothese:

De verwachting is dat de code van het blockchain systeem correct werkt. Upgeloade bestanden worden na goedkeurig correct opgeslagen in de blockchain en op de server.

Uitkomst:

Zoals de hypothese voorspelde, gaat alles goed. Een block wordt geupload naar de pending chain waar alleen een consensus account het kan goedkeuren. Eenmaal goedgekeurd is dit block terug te vinden in de webinterface en als ruwe data bij port 80/chain

Om de edge cases te vermijden is er een maximale bestandsgrootte toegevoegd en een try except statement in de code voor de bestandnaam. Als er iets fout gaat krijgt de gebruiker een foutmelding over de naam of de grootte van het bestand.

Test 2: Opslaan en Restart

Test:

Het doel van de test is om te kijken of data niet verloren raakt na een crash of restart.

Benodigdheden:

* Draaiende server
* Laptop

Setup:

Login als admin op de webinterface. Upload een bestand en voeg een korte beschrijving toe. Login als consensus en klik op ‘approve’ bij dit block in de pending list. Check of dit block in de blockchain terug te zien is. Sluit de server en restart deze. Log opnieuw in en probeer het nieuwe block te downloaden.

Code:

<https://github.com/JorisTDP/DIHM>

Edge cases:

* ?

Hypothese:

De blockchain en de links naar de bestanden worden opgeslagen na het uploaden, dus na een restart zal alles nogsteed in orde zijn.

Uitkomst:

Na een restart zijn alle bestanden nog toegankelijk en de blockchain is in orde.

Test 3: Accounts

Test:

Het doel van de test is om te kijken of de integratie van accounts met verschillende rollen goed is geimplementeerd.

Benodigdheden:

* Draaiende server
* Laptop

Setup:

Login als admin en maak een account voor de uploader, consensus en bekijker rollen. Login op al deze accounts om te zien of je wordt doorgelinked naar de desbetreffende webinterface. (zo kan de bekijker alleen de definitieve blockchain zien, de consensus kan ook de pending blockchain zien, en de uploader kan de upload optie zien). Alle accounts hebben de mogelijkheid om hun wachtwoord te veranderen, aangezien hun account wordt gemaakt door de admin. De admin is ook de enigste die accounts kan verwijderen. Verander het wachtwoord van een account, loguit en probeer opnieuw in te loggen op dit account met het nieuwe wachtwoord. Als dit allemaal lukt is de test geslaagd.

Code:

<https://github.com/JorisTDP/DIHM>

Edge cases:

* Probeer met een bekijkers account bij de andere webpagina’s te komen, zoals /consensus of /upload

Hypothese:

De verwachting is dat alle accounts worden doorgelinked naar de juiste pagina’s. Na het veranderen van het wachtwoord, zal bij het inloggen gebruikt moeten worden.

Uitkomst:

De functies: account toevoegen, wachtwoord veranderen en account verwijderen werken allemaal prima. De verschillende accounts worden ook doorgelinkt naar de correcte webinterface.

Voor de edge cases probeerde ik met een ‘view’ account naar de consensus te komen maar werd direct terug gelinked naar ‘view’ interface waar alleen de blockchain bekeken kan worden. Hetzelfde gebeurde met de andere accounts. Het is niet mogelijk om met een account de interface van een ander account te zien.

Test 4: Hash-check

Test:

Het doel van de test is om te kijken of het systeem kan detecteren of een bestand vervalst is.

Benodigdheden:

* Draaiende server
* Laptop

Setup:

Login als admin op server. Gebruik sudo nano om een bestand in /app/Uploads te bewerken dat in de blockchain staat. Check of dit block in de blockchain terug te zien is en kijk of de kleur van de blockchain verandert. Als de blockchain rood wordt is de test geslaagd.

Code:

<https://github.com/JorisTDP/DIHM>

Hypothese:

De verwachting is dat het hash-check algoritme van de blockchain gelijk ziet dat een hash van een bestand niet meer klopt en de blockchain wordt rood in de webinterface.

Uitkomst:

Na het bestand bewerkt te hebben op de server geeft de web interface aan “Issue in Blockchain” en de hele blockchain staat in het rood.

Test 5: BLE app

Test:

Het doel van de test is om te kijken of je via de app de blockchain kan ophalen van de server en bekijken in de app.

Benodigdheden:

* Draaiende server
* Schip module
* Mobiele telefoon met Bluetooth capabiliteit.

Setup:

Sluit de schip module aan op stroom en doe dit in een gebied met LoRa bereik voor deze test. Klik op scan in de app om voor bluetooth apparaten in de omgeving te kijken. Druk op DIHM-Module om hiermee te verbinden. Wacht totdat de blockchain in de app wordt weergegeven.

Code:

<https://github.com/JorisTDP/DIHM>

Edge cases:

* Er is geen huidige lora verbinding.

Hypothese:

Als er Lora bereik is zal na het verbinden met de schip module van het server de huidige blockchain weergegeven worden. Als er geen huidige verbinding is zal de laatste opgeslagen blockchain laten zien worden.

Uitkomst:

Nadat de module is opgestart en er verbinding is gemaakt via Bluetooth wordt er via LoRa verbinding gemaakt met de webserver. Via het protocol worden de berichten teruggestuurd en nadat deze zijn ontvangen worden deze berichten aan elkaar geplakt en naar de app gestuurd. Hierop is de volledige blockchain te zien. Als met de module verbinding wordt verbroken en de LoRa antenne wordt losgekoppeld zodat er geen verbinding meer is, wordt de vorige opgehaalde blockchain laten zien.

Test 6: Volledige test

Test:

Het doel van de test is om te kijken of de verloop van het hele proces in orde is. Dit om te kijken of het prototype goed gebruikt kan worden voor de daadwerkelijke implementatie. Zo voeren we de vorige 3 testen achtereenvolgend uit zoals het in de werkelijkheid zou gaan gebeuren.

Benodigdheden:

* Draaiende server
* Laptop
* Schip module
* Mobiele telefoon met Bluetooth capabiliteit.

Setup:

Sluit de schip module aan op stroom en doe dit in een gebied met LoRa bereik voor deze test. Verbind de laptop met het internet en ga naar de webinterface. Gebruik een uploader account om een IHM in de blockchain te zetten, voeg ook een beschrijving toe. Gebruik een consensus account om deze dit nieuwe block goed te keuren. Als het goed is zie je dit dan bij de definitieve blockchain staan. Open de app op de mobiele telefoon en klik op scan. Druk op DIHM-Module om hiermee te verbinden. Wacht totdat de blockchain in de app wordt weergegeven. Check of het nieuwe toegevoegde block in de chain terug te vinden is. Als dit zo is, is de test geslaagd.

Code:

<https://github.com/JorisTDP/DIHM>

Hypothese:

Als er Lora bereik is zal na het openen van een bluetooth verbinding met de schip module het nieuwste block en de vorige blocks op de telefoon weergegeven worden. Als er geen huidige verbinding is zal de laatste opgeslagen blockchain laten zien worden, en dus niet het nieuwe toegevoegde block.

Uitkomst:

Nadat een block is toegevoegd aan de webinterface heeft de schip module via LoRa de blockchain inclusief de laatste toevoeging opgehaald. Deze wordt in de app ontvangen en als je de blockchain in de app vergelijkt met de blockchain die wordt verstuurd van de server is deze identiek.

Test 7: Wifi test

Test:

Het doel van de test is om te kijken of we wifi kunnen gebruiken i.p.v LoRa voor het ophalen van de blockchain. Hiervoor is in de app een apart veld toegevoegd waar je kan inloggen met wifi. Eenmaal ingelogd wordt er via de wifi verbinding gemaakt met de blockchain server.

Benodigdheden:

* Draaiende server
* Laptop
* Schip module
* Mobiele telefoon met Bluetooth capabiliteit.

Setup:

Sluit de schip module aan op stroom en doe in de buurt van een wifi netwerk waarvan ook het wachtwoord bekend is. Verbind de laptop met het internet en ga naar de webinterface. Gebruik een uploader account om een IHM in de blockchain te zetten, voeg ook een beschrijving toe. Gebruik een consensus account om deze dit nieuwe block goed te keuren. Als het goed is zie je dit dan bij de definitieve blockchain staan. Open de app op de mobiele telefoon en klik op scan. Druk op DIHM-Module om hiermee te verbinden. Voer de wifi SSID (naam van netwerk) en het wachtwoord in en klik op de knop send. Wacht totdat de blockchain in de app wordt weergegeven. Check of het nieuwe toegevoegde block in de chain terug te vinden is. Als dit zo is, is de test geslaagd.

Code:

<https://github.com/JorisTDP/DIHM>

Edge cases:

* ….

Hypothese:

Als de wifi naam en wachtwoord goed zijn ingevuld zal het nieuwste block en de vorige blocks op de telefoon weergegeven worden. Als er niks wordt weergegeven zullen de wifi gegevens niet goed ingevoerd zijn.

Uitkomst:

Na verbonden te zijn met de module en de wifi gegevens doorgestuurd te hebben, wordt een block aan de blockchain toegevoegd. Door met de module te disconnecten via bluetooth en opnieuw te verbinden wordt de blockchain opnieuw opgehaald. Deze keer staat het nieuwe toegevoegde block netjes achteraan.