Testdocument – Ad Fundum Timer

User Story Overzicht:

1. Automatische Start & Stop van de Timer

User Story:  
*Als deelnemer wil ik dat de timer automatisch start wanneer ik mijn glas optil en stopt wanneer ik het neerzet, zodat mijn drinktijd nauwkeurig wordt geregistreerd.*

Testtypes:

* Unit Test
  + Correcte detectie van vol leeg en geen glas:

#include <Arduino.h>

// Pin waar je FSR op zit

#define FSR\_PIN 36

void setup() {

  Serial.begin(115200);

  pinMode(FSR\_PIN, INPUT);

  Serial.println("FSR-test gestart, lees waarden via de seriële monitor...");

}

void loop() {

  int fsrValue = analogRead(FSR\_PIN);

  Serial.print("FSR-waarde: ");

  Serial.println(fsrValue);

  delay(200);

}

Afbeelding met tekst, schermopname, Lettertype, ontwerp

Door AI gegenereerde inhoud is mogelijk onjuist.Afbeelding met tekst, verbruiksartikelen voor kantoor, overdekt, computermuis

Door AI gegenereerde inhoud is mogelijk onjuist.

Afbeelding met tekst, schermopname, Lettertype, typografie

Door AI gegenereerde inhoud is mogelijk onjuist.

Afbeelding met tekst, overdekt, kabel, verbruiksartikelen voor kantoor

Door AI gegenereerde inhoud is mogelijk onjuist.

Afbeelding met schermopname, tekst, typografie, ontwerp

Door AI gegenereerde inhoud is mogelijk onjuist.

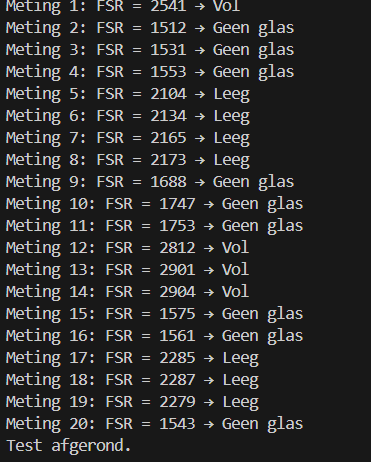
Afbeelding met tekst, kabel, Computerhardware, elektronica

Door AI gegenereerde inhoud is mogelijk onjuist.

* Integratie test:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Voorwaarde** | **Uitleg** | **Verwacht Resultaat** | **Werkelijk Resultaat** | **Tijd** | **Datum** |
| **1. FSR ≥ 2500** | Glas neergezet, sensorwaarde ≥ 2500 (vol glas) | glassFull() = true; state → TAG\_DELAY; NeoPixel begint wit te pulsen | true | 15:00 | 2025-05-29 |
| **2. readUID() vindt kaart** | Direct na vol-detectie NFC-kaart voorlezen | readUID() = “ABC123”; Seriële: “UID: ABC123”; state blijft TAG\_DELAY en NeoPixel schakelt naar blauwe chase | “ABC123” | 15:01 | 2025-05-29 |
| **3. Countdown (< 5 s na tagTime)** | In TAG\_DELAY moet display aftellen van 5 naar 1 met kolom | Display toont “05”, “04”, … “01” (kolon brandt); NeoPixel blijft blauwe chase | Correct | 15:02 | 2025-05-29 |
| **4. Countdown (≥ 5 s na tagTime)** | Na exact 5 s na tagTime | Seriële: “GO”; Display toont “GO”; NeoPixel → groen; state → WAIT\_EMPTY | Pass | 15:07 | 2025-05-29 |
| **5. FSR ≤ 1800 (glas verwijderd)** | Glas opgetild, sensorwaarde ≤ 1800 | Seriële: “Glas verwijderd, start meten…”; NeoPixel → geel; measureStart = millis(); state → MEASURING | Pass | 15:08 | 2025-05-29 |
| **6. Timer loopt (~3 s)** | In MEASURING moet display door­tellen in seconden+centi­seconden | Display toont oplopende tijd, b.v. “00 :03”; NeoPixel blijft geel | Pass | 15:11 | 2025-05-29 |
| **7. FSR ≥ 2000 (glas teruggeplaatst) na ≥ 200 ms** | Glas gezet, sensorwaarde ≥ 2000 en ≥ 200 ms na measureStart | Seriële: “Meting gestopt, drinktijd(ms): 3012”; NeoPixel → rood; Display toont “00 :03 .01”; MQTT-payload verzonden | Pass | 15:14 | 2025-05-29 |
| **8. millis() ≥ doneUntil (5 s na meting)** | 5 s wachten na DONE | Seriële: “Klaar, wacht op nieuwe sessie”; NeoPixel uit; Display uit; state → WAIT\_TAG | Pass | 15:19 | 2025-05-29 |

* Stress-test:  
  Voor deze user story voeren we 20 opeenvolgende FSR-uitlezingen uit (starten pas bij één keer “vol” glas) om te controleren of de sensorwaarden (≥2500 “Vol”, ≥2000 “Leeg”, ≤1800 “Geen glas”) bij herhaling stabiel en foutloos worden herkend zonder haperingen.



* Real-World Test:  
  Ik heb de volledige opstelling in de praktijk getest: zodra het glas vol op de sensor ligt, start de 5-seconden-countdown en gaat de timer direct van “GO” af. Na het weghalen van het glas begint de timer en bij terugplaatsen stopt hij precies. In verschillende tests met meerdere personen en snelheden kwam de getoonde drinktijd steeds overeen met de stopwatch, waarmee is aangetoond dat de setup werkt en de tijd correct wordt geregistreerd.

2. Registratie via de Webpagina

User Story:  
*Als deelnemer wil ik mijn naam en e-mailadres kunnen registreren via een webpagina, zodat ik mijn score per e-mail kan ontvangen.*

Testtypes:

* Unit Test

Test1: geldige registratie

Afbeelding met tekst, schermopname, Lettertype, ontwerp

Door AI gegenereerde inhoud is mogelijk onjuist.Afbeelding met tekst, schermopname, Lettertype, nummer

Door AI gegenereerde inhoud is mogelijk onjuist.

Bij een geldige registratie kom je op een welkom pagina die zegt dat je goed bent ingelogd

Afbeelding met tekst, schermopname, Lettertype, nummer

Door AI gegenereerde inhoud is mogelijk onjuist.Test2: lege naam

Niet mogelijk om geen naam in te geven.

Test 3: ongeldige email

Afbeelding met tekst, schermopname, Lettertype, nummer

Door AI gegenereerde inhoud is mogelijk onjuist.

Niet mogelijk om een ongeldige email in te geven.

* Integratietest:

Test 1: frontend 🡪 backend

Afbeelding met tekst, Lettertype

Door AI gegenereerde inhoud is mogelijk onjuist.

Afbeelding met tekst, Lettertype, schermopname, lijn

Door AI gegenereerde inhoud is mogelijk onjuist.

* Real-world test:

Afbeelding met tekst, Multimediasoftware, software, schermopname

Door AI gegenereerde inhoud is mogelijk onjuist.Configureer de Flask-app op de Raspberry Pi zo dat hij bij elke inkomende request het MAC-adres en alleen het iPad-MAC-adres toestaat. Probeer vanaf de iPad de registratiepagina te openen; deze zou normaal moeten laden. Navigeren vanaf een ander apparaat geeft direct “403 Forbidden.”

3. RFID-Koppeling met Glazen

User Story:  
*Als deelnemer wil ik mijn glas met een RFID-tag koppelen aan mijn account, zodat het systeem mijn tijd correct kan registreren.*

Wat te testen:

* Unit test:

Scannen rfid tag / scannen door 3d print

Afbeelding met tekst, elektronica, kabel, Elektrische bedrading

Door AI gegenereerde inhoud is mogelijk onjuist.Afbeelding met tekst, overdekt, persoon

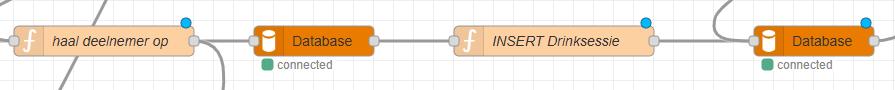
Door AI gegenereerde inhoud is mogelijk onjuist.

Afbeelding met tekst, Lettertype, schermopname, Graphics

Door AI gegenereerde inhoud is mogelijk onjuist.

* Integratietest:

Ik stuur de UID & drinktijd door via mqtt 🡪

Afbeelding met tekst, Lettertype, schermopname

Door AI gegenereerde inhoud is mogelijk onjuist.Data komt binnen op node-red 🡪 via de uid wordt de deelnemer id opgehaald uit de database 🡪 vervolgens wordt de tijd gekoppeld aan de deelnemer id.

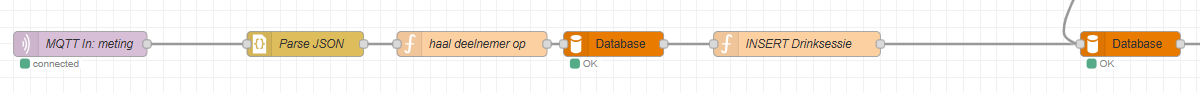
4. Live Scorebord

User Story:  
*Als deelnemer wil ik dat mijn score direct op een live scorebord verschijnt, zodat ik mijn prestaties kan vergelijken met anderen.*

* Integratie test:
  + Van timer naar dashboard:

Afbeelding met tekst, klok, overdekt, tafel

Door AI gegenereerde inhoud is mogelijk onjuist.Meting gebeurd 04:75

 Van timer naar node-red

Van node-red naar de database



Afbeelding met schermopname, Multimediasoftware, Grafische software, software

Door AI gegenereerde inhoud is mogelijk onjuist. Van database naar dashboar

5. Resultaten per E-mail Ontvangen

User Story:  
*Als deelnemer wil ik mijn score per e-mail ontvangen, zodat ik deze later nog kan bekijken en delen.*

Wat te testen:

Integratie test

* Afbeelding met lijn, tekst, schermopname, diagram

  Door AI gegenereerde inhoud is mogelijk onjuist.Automatische verzendactie van de e-mailservice met de juiste drinktijd

Iedere keer als er een nieuwe tijd binnen komt van de timer wordt er een email verstuurd naar de gebruiker van deze tijd.

Afbeelding met tekst, schermopname, Lettertype, software

Door AI gegenereerde inhoud is mogelijk onjuist.

6. LED & Display Feedback

User Story:  
*Als deelnemer wil ik LED-indicatoren en een display zien die aangeven wanneer de timer loopt en stopt, zodat ik direct visuele feedback krijg.*

Wat te testen:

* Unit-tests
  + Verschillende led kleuren en animaties

#include <Arduino.h>

#include <Adafruit\_NeoPixel.h>

#define NEOPIXEL\_PIN D2

#define NUMPIXELS    16   // Zet hier het werkelijke aantal LEDs in je ring

Adafruit\_NeoPixel pixels(NUMPIXELS, NEOPIXEL\_PIN, NEO\_GRB + NEO\_KHZ800);

void setup() {

  pixels.begin();

  pixels.setBrightness(50);  // 0–255

  pixels.show();             // alle LEDs uit

}

void loop() {

  // 1) vul de ring rood

  for (int i = 0; i < NUMPIXELS; i++) {

    pixels.setPixelColor(i, pixels.Color(255, 0, 0));

  }

  pixels.show();

  delay(1000);

  // 2) vul de ring groen

  for (int i = 0; i < NUMPIXELS; i++) {

    pixels.setPixelColor(i, pixels.Color(0, 255, 0));

  }

  pixels.show();

  delay(1000);

  // 3) vul de ring blauw

  for (int i = 0; i < NUMPIXELS; i++) {

    pixels.setPixelColor(i, pixels.Color(0, 0, 255));

  }

  pixels.show();

  delay(1000);

  // 4) “chase” effect: één witte LED langs de ring

  for (int i = 0; i < NUMPIXELS; i++) {

    pixels.clear();

    pixels.setPixelColor(i, pixels.Color(255, 255, 255));

    pixels.show();

    delay(200);

  }

}

Afbeelding met groen, verlichting, Led-lamp, licht

Door AI gegenereerde inhoud is mogelijk onjuist.Afbeelding met verlichting, licht, Majorelleblauw, blauw

Door AI gegenereerde inhoud is mogelijk onjuist.

Afbeelding met verlichting, licht, Led-lamp, rood

Door AI gegenereerde inhoud is mogelijk onjuist.

* + 7-segment display testen

#include <Arduino.h>

#include <Wire.h>

#include <Adafruit\_LEDBackpack.h>

#include <Adafruit\_GFX.h>

// Instantie van het 7-segment display (I2C-adres 0x70)

Adafruit\_7segment display = Adafruit\_7segment();

void setup() {

  Serial.begin(115200);

  Wire.begin();

  display.begin(0x70);

  display.clear();

  display.writeDisplay();

  Serial.println("7-segment test gestart");

}

void loop() {

  // 1) Toon “8888” om alle LEDs te testen

  display.print(8888);

  display.drawColon(false);

  display.writeDisplay();

  Serial.println("Test: 8888");

  delay(2000);

  // 2) Test cijfers 0–9 op digit 0 (de rest uit)

  for (int i = 0; i < 10; i++) {

    display.clear();

    display.writeDigitNum(0, i, false);

    display.writeDisplay();

    Serial.print("Digit0 -> "); Serial.println(i);

    delay(500);

  }

  // 3) Toon “1234” zonder colon

  display.clear();

  display.writeDigitNum(0, 1, false);

  display.writeDigitNum(1, 2, false);

  display.writeDigitNum(3, 3, false);

  display.writeDigitNum(4, 4, false);

  display.drawColon(false);

  display.writeDisplay();

  Serial.println("Test: 1234");

  delay(2000);

  // 4) Toon “12:34” met colon

  display.clear();

  display.writeDigitNum(0, 1, false);

  display.writeDigitNum(1, 2, false);

  display.drawColon(true);

  display.writeDigitNum(3, 3, false);

  display.writeDigitNum(4, 4, false);

  display.writeDisplay();

  Serial.println("Test: 12:34");

  delay(2000);

}

Afbeelding met tekst, klok, elektronica, Elektronisch apparaat

Door AI gegenereerde inhoud is mogelijk onjuist.

Afbeelding met tekst, elektronica, Elektronisch apparaat, overdekt

Door AI gegenereerde inhoud is mogelijk onjuist.Afbeelding met tekst, elektronica, Elektronisch apparaat, overdekt

Door AI gegenereerde inhoud is mogelijk onjuist.

Afbeelding met tekst, klok, elektronica, Elektronisch apparaat

Door AI gegenereerde inhoud is mogelijk onjuist.Afbeelding met tekst, klok, elektronica, Elektronisch apparaat

Door AI gegenereerde inhoud is mogelijk onjuist.

* Integratie test:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Scenario** | **Voorwaarde & Uitleg** | **Verwacht Resultaat (LED & Display)** | **Werkelijk Resultaat** | **Tijd** | **Datum** |
| 1 | `state == WAIT\_TAG` – Geen glas (FSR < 2500). | LED: wit pulserend Display: uit | Alles oke | 20:00 | 2025-06-04 |
| 2 | `state == WAIT\_TAG`, daarna FSR ≥ 2500 én `readUID()` ≠ `` (glas vol en kaart gelezen). | LED: uit (witte puls stopt) Display: uit | Alles oke | 20:02 | 2025-06-04 |
| 3 | `state == TAG\_DELAY` en `glassFull()` blijft true – 5 s countdown en blauwe chase-animatie. | LED: blauwe chase-animatie Display: aftellen: “05”→“04”→… met knipperende colon | Alles oke | 20:05 | 2025-06-04 |
| 4 | `state == TAG\_DELAY`, maar glas valt weg vóór 5 s (FSR < 2500) en blijft zo ≥ 200 ms. | LED: 2 s knipper rood (250 ms aan, 250 ms uit) Display: direct uit | Alles oke | 20:08 | 2025-06-04 |
| 5 | `state == TAG\_DELAY`, en `now – tagTime ≥ 5000` ms (5 s countdown voltooid). | LED: groen Display: “GO” | Alles oke | 20:10 | 2025-06-04 |
| 6 | `state == WAIT\_EMPTY`, en `glassRemoved()` true (glas opgetild, FSR ≤ 1800). | LED: geel Display: timerweergave start | Alles oke | 20:12 | 2025-06-04 |
| 7 | `state == MEASURING`, en `now – measureStart < 200` ms (vroeg in de meting). | LED: geel blijft Display: toont oplopende tijd | Alles oke | 20:13 | 2025-06-04 |
| 8 | `state == MEASURING`, en `now – measureStart ≥ 200` ms én `glassEmpty()` true én `readUID() == currentUID`. | LED: kleur op basis van duration (<5s groen, 5–10s oranje, ≥10s rood) Display: toont pauzetijd | Alles oke | 20:15 | 2025-06-04 |
| 9 | `state == MEASURING`, en `now – measureStart ≥ 200` ms én `glassEmpty()` true, maar `readUID() != currentUID`. | LED: 2 s knipper rood Display: direct uit | Alles oke | 20:17 | 2025-06-04 |
| 10 | `state == DONE`, en `now < doneUntil` (ms < 5 s na stop). | LED: blijft huidige kleur Display: blijft pauzetijd tonen | Alles oke | 20:20 | 2025-06-04 |
| 11 | `state == DONE`, en `now ≥ doneUntil` (≥ 5 s na stop). | LED: uit Display: uit | Alles oke | 20:25 | 2025-06-04 |

7. Dataveiligheid en Privacy

User Story:  
*Als gegevensbeheerder wil ik dat alle gebruikersgegevens veilig worden opgeslagen en beschermd zijn tegen ongeautoriseerde toegang, zodat privacy wordt gewaarborgd.*

Wat te testen:

Veiligheid database:

De database draait lokaal op de raspberry pi en is beveiligd met een wachtwoord .

Afbeelding met tekst, schermopname, Lettertype, ontwerp

Door AI gegenereerde inhoud is mogelijk onjuist.

8. Handmatige Reset & Noodstop

User Story:  
*Als organisator wil ik een handmatige resetknop en een noodstopfunctie hebben, zodat ik kan ingrijpen bij storingen of misbruik.*

Wat te testen:

* Unit test:
  + Werking van de fysieke resetknop

Testen gewoon en door 3d geprint materiaal.

Afbeelding met Elektrische bedrading, kabel, Elektronische engineering, Elektriciteitsnetwerk

Door AI gegenereerde inhoud is mogelijk onjuist.

Afbeelding met Elektrische bedrading, Elektronische engineering, kabel, elektronica

Door AI gegenereerde inhoud is mogelijk onjuist.

* Integratie test:
  + Testen in behuizing en volledige opstelling.

Bij het duwen op de buiten kan van de behuizing wordt de esp gereset.

Afbeelding met tekst, elektronica, kabel, Elektrische bedrading

Door AI gegenereerde inhoud is mogelijk onjuist.

Afbeelding met tekst, schermopname, Lettertype

Door AI gegenereerde inhoud is mogelijk onjuist.

Reset functie:

void checkReset() {

  if (digitalRead(RESET\_PIN) == LOW) return;

  Serial.println("Reset ingedrukt, herstart...");

  ESP.restart();

}

9. Ontwerp en Productie van de Behuizing

User Story:  
*Als technisch beheerder wil ik een stevige en functionele behuizing ontwerpen en produceren, zodat alle hardwarecomponenten goed beschermd zijn en het systeem er professioneel uitziet.*

Wat te testen:

* Unit test:
  + Test prints verschillende vormen drukweerstand

Afbeelding met blauw, overdekt, vloer

Door AI gegenereerde inhoud is mogelijk onjuist.Afbeelding met tekst, verbruiksartikelen voor kantoor, badkamer, gootsteen

Door AI gegenereerde inhoud is mogelijk onjuist.Afbeelding met overdekt, tafel, vloer, ontwerp

Door AI gegenereerde inhoud is mogelijk onjuist.Afbeelding met cirkel, kunst, blauw, vloer

Door AI gegenereerde inhoud is mogelijk onjuist.

* Integratie test:
  + Uitsparingen gemaakt voor hardware:

Afbeelding met schets

Door AI gegenereerde inhoud is mogelijk onjuist.

display

frs

Led ring

Usb-c

Reset knop