Dossier

Inhoudsopgave

[1. Productomschrijving 3](#_Toc197285300)

[2. Technische Documentatie 3](#_Toc197285301)

[2.1 Hardware 3](#_Toc197285302)

[2.2 Software 4](#_Toc197285303)

[2.3 Architectuurdiagram 4](#_Toc197285304)

[2.4 Gegevensuitwisseling 5](#_Toc197285305)

[3. Programmacode 5](#_Toc197285306)

[4. Testrapport 5](#_Toc197285307)

[4.1 Testcases 5](#_Toc197285308)

[4.2 Bevindingen 6](#_Toc197285309)

[5. Installatiehandleiding 6](#_Toc197285310)

[5.1 Benodigdheden 6](#_Toc197285311)

[5.2 Stap-voor-stap installatie 7](#_Toc197285312)

[5.2.1 FireBeetle ESP32 Setup 7](#_Toc197285313)

[5.2.2 Raspberry Pi Setup 8](#_Toc197285314)

[6. Gebruikershandleiding 10](#_Toc197285315)

[6.1 Voor Ploegen 10](#_Toc197285316)

[6.2 Voor Kassamedewerkers 10](#_Toc197285317)

[6.3 Problemen oplossen 11](#_Toc197285318)

# 1. Productomschrijving

Dit project betreft een kassasysteem voor Patro Hoevenen dat gebruikmaakt van RFID-badges om rekeningen te beheren. Het systeem bestaat uit een touchscreen display met een dashboard waarop zowel ploegen als kassamedewerkers hun acties kunnen uitvoeren.

Ploegen krijgen een RFID-badge waarmee ze hun rekening kunnen openen door deze tegen de RFID-scanner te houden. Na het scannen verschijnt er een menu met beschikbare dranken en snacks, elk met een vaste prijs. Wanneer een item wordt geselecteerd, wordt dit toegevoegd aan de rekening van de ploeg.

Kassamedewerkers beschikken over speciale RFID-badges met uitgebreide rechten. Zij kunnen items verwijderen uit rekeningen als er fouten zijn gemaakt. Voor het afrekenen scant de kassamedewerker eerst de badge van de ploeg en vervolgens hun eigen badge. Hierdoor wordt de volledige rekening inclusief het totaalbedrag getoond. Na betaling kunnen de kassamedewerkers alle items van de rekening verwijderen.

Alle bestelde items worden opgeslagen in een MariaDB-database. Voor elke ploeg (badge) wordt een aparte tabel aangemaakt, waardoor het systeem meerdere rekeningen tegelijkertijd kan beheren.

# 2. Technische Documentatie

## 2.1 Hardware

Op basis van de uitgevoerde technische analyse zijn de volgende hardware-componenten geselecteerd:

* **NFC-scanner**: Gravity UART & I2C NFC Module
  + Input Voltage: 3.3V~5.5V
  + Interface: Gravity I2C & UART (PH2.0-4P, Logic Level 0-3.3V)
  + UART Baud Rate: 115200 bps
  + I2C Address: 0x48
  + Maximum Communication Distance: ≥10mm
* **Microcontroller voor NFC**: FireBeetle ESP32
  + Verbonden met de NFC-scanner
  + Communiceert met de Raspberry Pi via WiFi/HTTP
* **Touchscreen display**: 7" Touchscreen (Compatible with Raspberry Pi 5/4B/3B+/3B)
  + Resolution: 800\*480
  + Refresh Frequency: 60Hz
  + Operating Temperature: -20°C~70°C
  + Dimension: 165mm×100mm
* **Besturingssysteem**: Raspberry Pi 4 Model B
  + Gebruikt voor het draaien van de database, webserver en touchscreen interface
* **Status-indicatoren**: Groene en rode LED's
  + Verbonden met de FireBeetle ESP32
  + Voor visuele feedback over de status van scans

## 2.2 Software

* **FireBeetle ESP32**:
  + Arduino IDE code voor het uitlezen van de NFC-scanner
  + DFRobot\_PN532 library voor communicatie met de RFID-scanner
  + HTTP client voor communicatie met de Raspberry Pi
* **Raspberry Pi**:
  + Besturingssysteem: Raspberry Pi OS
  + Database: MariaDB voor het opslaan van gebruikers en bestellingen
  + Webserver: Flask voor het hosten van het dashboard
  + Frontend: HTML/CSS/JavaScript voor de dashboard interface

## 2.3 Architectuurdiagram

Het systeem volgt onderstaande architectuur:

1. De RFID-badge wordt gescand met de NFC-scanner die verbonden is met de FireBeetle ESP32
2. De FireBeetle ESP32 verwerkt de NFC-data en stuurt de UID van de badge via HTTP naar de Raspberry Pi
3. De Raspberry Pi (Flask-applicatie) identificeert de gebruiker en bepaalt de rechten
4. Afhankelijk van het type gebruiker (ploeg of kassa):
   * Voor ploegen: het menu wordt getoond om items te kunnen bestellen
   * Voor kassa: beheermogelijkheden worden getoond om items te verwijderen of rekeningen te beheren
5. Geselecteerde items worden via HTTP verwerkt door de Flask webserver
6. De Flask applicatie slaat de bestellingen op in de MariaDB database
7. De FireBeetle ESP32 geeft via LED's visuele feedback over de status van de scan (groen voor succes, rood voor fouten)

## 2.4 Gegevensuitwisseling

* **NFC-scanner → FireBeetle ESP32**: RFID-gegevens van de gescande badge via UART of I2C
* **FireBeetle ESP32 → Raspberry Pi**: HTTP POST requests met de UID van de badge
* **Raspberry Pi → Touchscreen**: Weergave van het dashboard
* **Flask webserver → MariaDB**: SQL-queries voor het opslaan en ophalen van bestellingen
* **Flask → Browser**: HTTP-responses met HTML, CSS en JavaScript voor het dashboard
* **Gebruiker → Flask**: HTTP-requests voor het toevoegen of verwijderen van items

# 3. Programmacode

De volledige en meest recente programmacode is te vinden in de GitHub repository:

[GitHub - Darryllens/Afstudeerproject](https://github.com/Darryllens/Afstudeerproject)

# 4. Testrapport

## 4.1 Testcases

| **ID** | **Testcase** | **Verwacht resultaat** | **Werkelijk resultaat** | **Status** |
| --- | --- | --- | --- | --- |
| TC01 | Scannen van een geregistreerde ploeg-badge | Menu met beschikbare items wordt getoond | Menu wordt correct getoond | Geslaagd |
| TC02 | Selecteren van een item | Item wordt toegevoegd aan de rekening van de ploeg | Item wordt correct toegevoegd | Geslaagd |
| TC03 | Scannen van een kassa-badge | Beheeropties worden getoond | Beheeropties worden correct getoond | Geslaagd |
| TC04 | Verwijderen van een item door kassamedewerker | Item wordt verwijderd van de rekening | Item wordt correct verwijderd | Geslaagd |
| TC05 | Afrekenen van een rekening | Totaalbedrag wordt getoond en items kunnen worden verwijderd | Totaalbedrag is correct en items kunnen worden verwijderd | Geslaagd |
| TC06 | Scannen van een onbekende badge | Redirectie naar registratiepagina | Redirectie werkt correct | Geslaagd |
| TC07 | Registreren van een nieuwe badge | Nieuwe tabel wordt aangemaakt in de database | Tabel wordt correct aangemaakt | Geslaagd |
| TC08 | Gelijktijdig beheren van meerdere rekeningen | Elke rekening blijft apart zichtbaar en bewerkbaar | Rekeningen blijven gescheiden | Geslaagd |
| TC09 | Connectie tussen ESP32 en Raspberry Pi | ESP32 stuurt UID correct naar Raspberry Pi | Communicatie werkt betrouwbaar | Geslaagd |
| TC10 | LED-indicatie bij scannen | Groen licht bij succes, rood licht bij fout | LED's werken naar behoren | Geslaagd |

## 4.2 Bevindingen

Tijdens het testen zijn de volgende bevindingen gedaan:

* Het systeem werkt naar behoren voor alle basisfunctionaliteiten
* De communicatie tussen de FireBeetle ESP32 en Raspberry Pi is stabiel
* De database-tabellen worden correct aangemaakt en bijgewerkt
* Het systeem is in staat om meerdere rekeningen tegelijkertijd te beheren
* De beveiliging werkt goed met verschillende rechten voor ploegen en kassamedewerkers
* De LED-indicaties geven duidelijke visuele feedback over de status van scans

# 5. Installatiehandleiding

## 5.1 Benodigdheden

* Raspberry Pi 4 Model B
* 7" Touchscreen display
* FireBeetle ESP32
* Gravity UART & I2C NFC Module
* Rode en groene LED's
* Micro SD-kaart (minimaal 16GB)
* Voedingsadapter voor Raspberry Pi
* USB-kabel voor FireBeetle ESP32
* RFID-badges
* WiFi-netwerk

## 5.2 Stap-voor-stap installatie

### 5.2.1 FireBeetle ESP32 Setup

1. **Visual Studio Code installeren en configureren**

* Download en installeer Visual Studio Code van [code.visualstudio.com](https://code.visualstudio.com/)
* Open Visual Studio Code en installeer de PlatformIO IDE extensie:
  + Klik op het Extensions-tabblad (of druk op Ctrl+Shift+X)
  + Zoek naar "PlatformIO IDE"
  + Klik op "Install"
  + Wacht tot de installatie is voltooid en herstart VS Code indien gevraagd

2. **Project aanmaken**

* Open VS Code
* Klik op het PlatformIO-icoon in de zijbalk
* Klik op "New Project"
* Vul de volgende gegevens in:
  + Name: nfc\_scanner
  + Board: FireBeetle-ESP32
  + Framework: Arduino
* Klik op "Finish" en wacht tot het project is aangemaakt

3. **platformio.ini configureren**

* Open het bestand platformio.ini in de projectmap
* Voeg de volgende configuratie toe:

ini

[env:firebeetle32]

platform = espressif32

board = firebeetle32

framework = arduino

monitor\_speed = 115200

lib\_deps =

bblanchon/ArduinoJson@^6.21.4

dfrobot/DFRobot\_PN532@^1.0.0

4. **Hardware aansluiten**

* Sluit de NFC-scanner aan op de FireBeetle ESP32:
  + VCC naar 3.3V
  + GND naar GND
  + SDA naar D21 (SDA)
  + SCL naar D22 (SCL)
* Sluit de LED's aan op de FireBeetle ESP32:
  + Rode LED naar pin D12 (met een weerstand van 220 Ohm)
  + Groene LED naar pin D13 (met een weerstand van 220 Ohm)

5. **Code schrijven**

* Vervang de inhoud van src/main.cpp met de NFC scanner code
* Bewerk de volgende instellingen in de code:

cpp

const char\* ssid = "JouwWiFiNaam";

const char\* password = "JouwWiFiWachtwoord";

const char\* serverAddress = "http://192.168.x.x:5000"; *// IP van je Raspberry Pi*

6. **Code uploaden**

* Verbind de FireBeetle ESP32 met je computer via USB
* Klik op de pijl-knop (Upload) in de onderste statusbalk van VS Code
* Wacht tot de code gecompileerd en geupload is
* Open de Serial Monitor om de output te bekijken (klik op het stopcontact-icoon in de onderste statusbalk)

### 5.2.2 Raspberry Pi Setup

1. **Besturingssysteem installeren**
   * Download Raspberry Pi OS van de officiële website
   * Schrijf het besturingssysteem naar de Micro SD-kaart met Raspberry Pi Imager
   * Plaats de SD-kaart in de Raspberry Pi
2. **Hardware aansluiten**
   * Sluit het touchscreen display aan op de HDMI-poort van de Raspberry Pi
   * Sluit de voedingsadapter aan op de Raspberry Pi
   * Zorg dat de Raspberry Pi verbonden is met hetzelfde WiFi-netwerk als de FireBeetle ESP32
3. **Software installeren**
   * Start de Raspberry Pi op en open een terminal
   * Update het systeem: sudo apt update && sudo apt upgrade -y
   * Installeer MariaDB: sudo apt install mariadb-server -y
   * Beveilig de MariaDB-installatie: sudo mysql\_secure\_installation
   * Installeer Python en benodigde packages:
   * sudo apt install python3-pip python3-venv -y
   * python3 -m venv venv
   * source venv/bin/activate

pip install flask mysql-connector-python

1. **Project downloaden en configureren**
   * git clone <https://github.com/Darryllens/Afstudeerproject.git>
   * Ga naar de projectmap: cd Afstudeerproject
   * Maak de database aan:
   * sudo mysql -u root -p
   * CREATE DATABASE PatroHoevenen;
   * CREATE USER 'Darryll'@'localhost' IDENTIFIED BY 'Darryllens12';
   * GRANT ALL PRIVILEGES ON PatroHoevenen.\* TO 'Darryll'@'localhost';
   * FLUSH PRIVILEGES;

EXIT;

* + Maak de kassamedewerkerstabel aan:
  + sudo mysql -u root -p PatroHoevenen
  + CREATE TABLE kassa (id INT AUTO\_INCREMENT PRIMARY KEY, card\_uid VARCHAR(255) NOT NULL, name VARCHAR(255) NOT NULL);
  + INSERT INTO kassa (card\_uid, name) VALUES ('14E524DB', 'Admin');

EXIT;

1. **Applicatie starten**
   * Start de Flask-applicatie: python app.py
   * De applicatie is nu beschikbaar op het touchscreen
2. **Automatisch starten bij opstarten**
   * Maak een service-bestand aan: sudo nano /etc/systemd/system/kassasysteem.service
   * Voeg de volgende inhoud toe:
   * [Unit]
   * Description=Kassa Systeem Patro Hoevenen
   * After=network.target
   * [Service]
   * User=pi
   * WorkingDirectory=/home/pi/kassa-systeem-patro-hoevenen
   * ExecStart=/home/pi/kassa-systeem-patro-hoevenen/venv/bin/python app.py
   * Restart=always
   * [Install]

WantedBy=multi-user.target

* + Enable de service: sudo systemctl enable kassasysteem.service
  + Start de service: sudo systemctl start kassasysteem.service

# 6. Gebruikershandleiding

## 6.1 Voor Ploegen

**Een rekening openen**

1. Houd je ploegkaart tegen de kaartlezer
2. Wacht tot het groene lichtje brandt
3. Je ziet nu het menu met beschikbare dranken en snacks op het scherm

**Een bestelling plaatsen**

1. Tik op het gewenste item op het touchscreen
2. Het item wordt direct toegevoegd aan je rekening
3. Je ziet een bevestiging op het scherm en het totaalbedrag wordt bijgewerkt

**Je rekening bekijken**

1. Houd je ploegkaart tegen de kaartlezer
2. Je ziet nu al je bestelde items en het totaalbedrag

## 6.2 Voor Kassamedewerkers

**Inloggen als kassamedewerker**

1. Houd je kassamedewerkerkaart tegen de kaartlezer
2. Wacht tot het groene lichtje brandt
3. Je bent nu ingelogd als kassamedewerker

**Een rekening bekijken**

1. Vraag de ploeg om hun kaart tegen de lezer te houden
2. Je ziet nu de rekening van de ploeg op het scherm

**Items verwijderen van een rekening**

1. Bekijk eerst de rekening van de ploeg (zie hierboven)
2. Tik op het "Verwijderen" icoon naast het item dat je wilt verwijderen
3. Het item wordt verwijderd en het totaalbedrag wordt automatisch bijgewerkt

**Een rekening afrekenen**

1. Bekijk eerst de rekening van de ploeg (zie hierboven)
2. Vertel het totaalbedrag aan de ploeg en ontvang de betaling
3. Tik op de knop "Alle Drankjes Verwijderen" om de rekening leeg te maken
4. Bevestig de actie wanneer hierom gevraagd wordt

**Een nieuwe ploeg registreren**

1. Als een onbekende kaart wordt gescand, wordt automatisch het registratiescherm getoond
2. Vul de naam van de ploeg in
3. Tik op "Registreren"
4. De nieuwe ploegkaart is nu geregistreerd en klaar voor gebruik

## 6.3 Problemen oplossen

**Kaart wordt niet herkend**

* Controleer of de kaart dicht genoeg bij de lezer wordt gehouden
* Houd de kaart iets langer tegen de lezer
* Als het rode lichtje brandt, is er een probleem. Neem contact op met de beheerder

**Scherm reageert niet**

* Wacht enkele seconden
* Als het scherm blijft hangen, vraag een kassamedewerker om het systeem opnieuw op te starten

**Geen verbinding tussen kaartlezer en dashboard**

* Controleer of het groene of rode lichtje op de kaartlezer brandt bij het scannen
* Als geen van beide lichtjes brandt, controleer dan of de FireBeetle ESP32 is ingeschakeld
* Controleer of beide apparaten verbonden zijn met hetzelfde WiFi-netwerk
* Herstart indien nodig zowel de FireBeetle ESP32 als de Raspberry Pi

**Verkeerd item toegevoegd**

* Vraag een kassamedewerker om het item te verwijderen
* De kassamedewerker kan items verwijderen met hun speciale kaart