Automation of Beer Dispensers: A Cyber-Physical System Solution

Lorenzo Piarulli
Sapienza University of Rome
Dept. of Computer Science

Luca Di Pietro
Polytechnic University of Milan
Dept. of Computer Engineering

Luís Paiva
University of Porto
FEUP

José A. Gaspar
University of Porto
FEUP

Lucas Sousa
University of Porto
FEUP

Alexandre Correia
University of Porto
FEUP

Miguel Teixeira
University of Porto
FEUP

Jaume Sansano
University of Porto
FEUP

Emanuel Gestosa
University of Porto
FEUP

Rui Pinto
University of Porto
FEUP, SYSTEC, ARISE

Gil Gonçalves
University of Porto
FEUP, SYSTEC, ARISE

Who Am I?

Lorenzo Piarulli

Master's student at Sapienza University of Rome
Erasmus student at the University of Porto

I have a bachelor's degree in Computer and Automation Engineering.

I am part of the HLC research group at Sapienza, led by Prof. Daniele De Sensi, specializing in HPC, parallel programming, accelerators, and HPC networks.

My work focuses on accelerating intensive tasks on emerging architectures and benchmarking HPC systems.

I collaborate with the OEHI organization on the analysis of collectives in HPC systems in partnership with HUAWEI.

I was a volunteer at the SC24 conference in Atlanta, USA.

piarulli.1884766@studenti.uniroma1.it

The Outline

1. Introduction and Motivation

2. Proposed System Architecture Overview

3. Proposed Solution:

   - ➡ Communication
   - ➡ Tank Controller Hardware
   - ➡ Cloud API
   - ➡ Security
   - ➡ Application

4. Conclusions

5. Future Works

Introduction and Motivation: Starting Problems

Tanks are vital in industrial and commercial operations. IoT automation enhances efficiency, reduces risks, and optimizes resource use through real-time monitoring, predictive maintenance, and precise control, minimizing waste and improving performance.

1 Manual Pouring

Traditional beer dispensing methods are prone to human error, resulting in inconsistent pours or wasted beer.

2 Time-Consuming

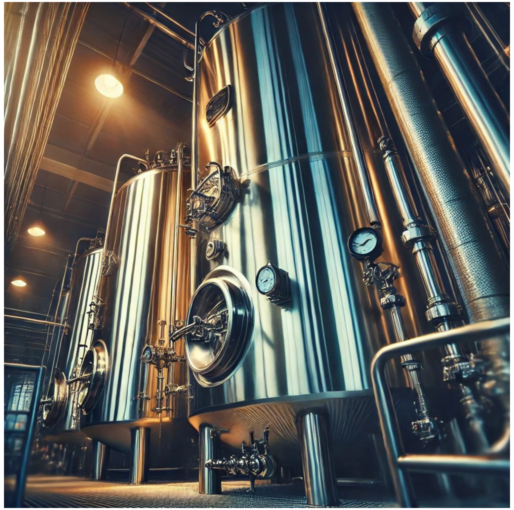
Manual pouring can be slow, especially during peak hours, leading to long wait times for customers.

3 Labor Dependency

Relying on staff creates a bottleneck during busy times, as they cannot assist multiple customers simultaneously.

4 Uncertainty of Inventory

Staff often have no real-time visibility of beer levels in the tanks



Introduction and Motivation: Project Goals

Our project is based on a commercial solution used in beer vending systems. The selected objectives aim to address key challenges in beer dispensing while providing a scalable approach applicable to liquid tank storage in various industries.

Automate Dispensing

Traditional beer dispensing methods are prone to human error, resulting in inconsistent pours or wasted beer.

Increase Efficiency

Reduce pouring time and streamline the dispensing process, improving customer satisfaction.

Enhance Pouring

Allow customers to activate pouring, allowing for more pours and less wait time.

Check Temperature

Ensure ideal temperatures by monitoring the beer's temperature and keep it cool.

Managing Inventory

Provide a reliable way to check the remaining beer in the tank.

Provide Easy UX

A companion app allows for easier use by the customers and better control by the staff.

System Architecture: Overview of the System

Each **shop** has a **central Raspberry Pi**, connecting multiple tank-attached **Arduino-based systems**. Data from sensors and actuators is transmitted to the **Edge server** and then forwarded to the **Cloud** for remote monitoring.

The system has been divided in multiple layers to enhance security, usability and maintenance

Cloud Layer

Hosting a cloud server that allows for the storage and processing of dispenser data. Supports the companion app.

Fog Layer

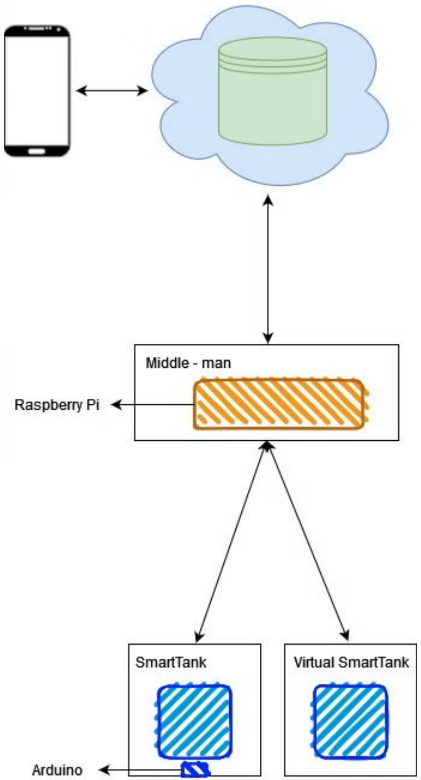
Manages communication between the machines in the edge layer and the cloud layer, simplifying an establishment's networking. Managed by a Raspberry Pi.

Edge Layer

Manages and coordinates the operation of each dispenser. Managed by each machine's Arduino, sensors and actuators.

App Layer

Application that allows user to buy beer, activate the dispenser and see real time and historical data.



Proposed Solution: Communication Architecture

Why Cloud?

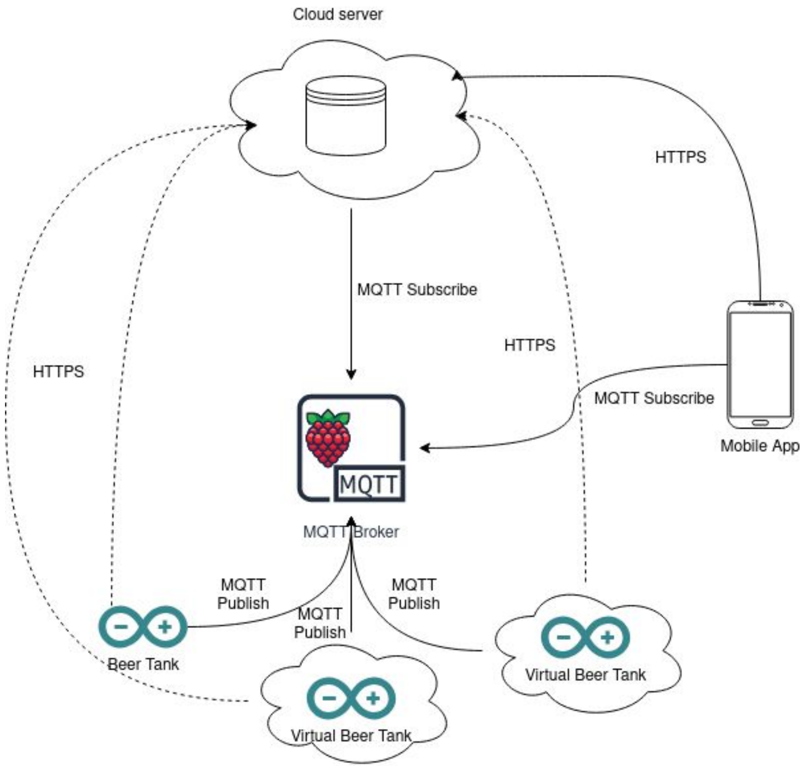
- Enables long-term data storage and analytics
- Provides scalability and flexibility in data management

Why Edge?

- Each store manages its own tanks with a local controller
- Ensures secure and efficient communication
- Reduces latency by processing data closer to the source

System Architecture

- **Star Topology**: MQTT broker at the center for efficient communication
- **Edge-to-Cloud Communication**: Edge devices can validate beer tokens via HTTPS
- **Mobile App Integration**: Primarily connects to the cloud but also subscribes to the MQTT broker for real-time data



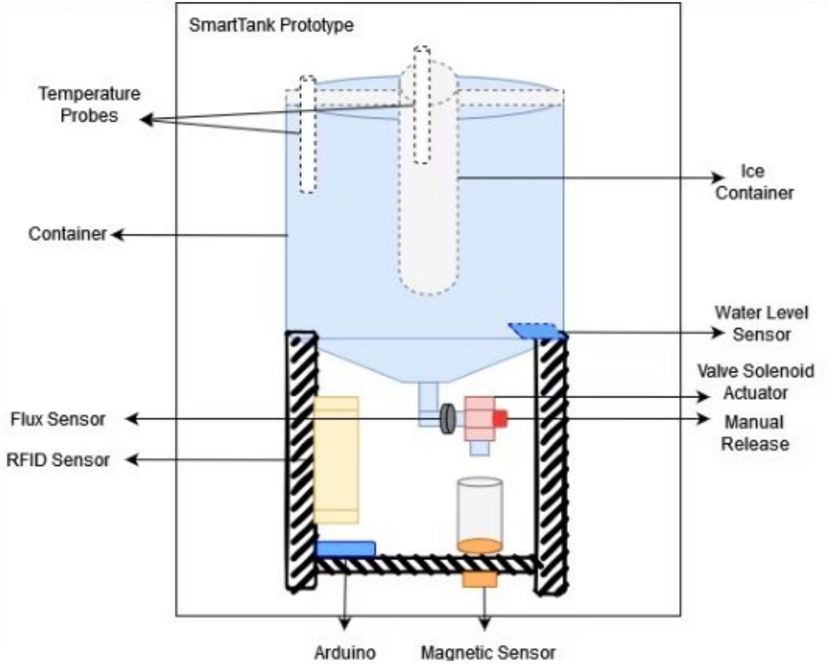
Proposed Solution: Tank Controller Hardware

Each **beer dispensing tank** is controlled by an **Arduino Nano WiFi**, responsible for monitoring sensors, activating actuators, and sending data

The **modular design** allows for easy replacement of valves, sensors, and upgrades.

System Routine:

1. **RFID Token Detection** – User scans an RFID-equipped cup.
2. **Token Validation** – Arduino requests verification from the central system.
3. **Dispensing Activation** – Upon approval, Arduino triggers the **relay** to open the **solenoid valve**.
4. **Liquid Flow Control** – **Flux sensor** monitors the volume in real-time.
5. **Completion & Reset** – Once the correct amount is poured, the system **closes the valve** and **resets flow status**.



Proposed Solution: Tank Controller Hardware

Key Components

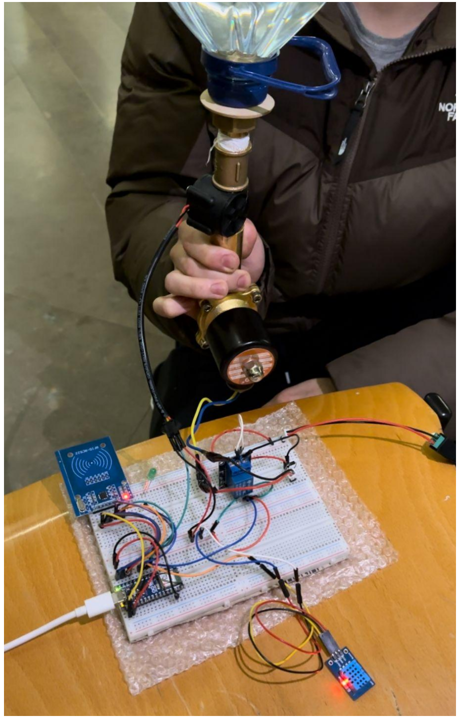
- **RFID Reader** – Detects user authentication tokens.
- **DHT11 Sensor** – Monitors temperature and humidity.
- **Flux Sensor** – Measures liquid volume dispensed.
- **Solenoid Valve** – Controls liquid flow.
- **Relay Module** – Activates the solenoid valve.
- **Power Supply (12V Charger, Diode, Resistors, Cables)** – Provides stable energy to components.

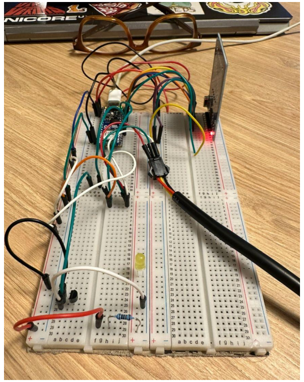
Firmware & Software Control

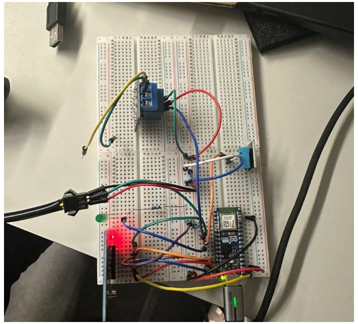
- Developed in **C/C++**, utilizing **interrupt-based programming**

Arduino Nano	1
RFID Reader	1
RFID Tag	2
Temperature Sensor	1
Relay	1
Flux Sensor	1
Solenoid Valve	1
Tank	1
Junctions	4
Diode	1

3.3 V - 12 V







Proposed Solution: Cloud API

The cloud infrastructure is designed for long-term storage and advanced analytics, enabling users to choose from different providers while addressing privacy and security concerns.

How is built?

- Built with Node.js, Express.js, and SQLite.
- Deployed on Render.com with automated CI/CD pipeline.
- JWT-based authentication with dual-token system.

What provides?

- Tracks real-time tank telemetry (temperature, levels).
- Supports dynamic endpoints with contextual validation.
- Provides analytics for tank and establishment performance.

Proposed Solution: Security

Ensuring robust security is critical, especially since both the cloud server and the Raspberry Pi are exposed to the internet. Even in a proof-of-concept stage, strong security measures must be implemented to prevent vulnerabilities and unauthorized access.

Key Security Measures:

- **Encrypted Communication**: Always use **HTTPS** instead of HTTP.
- **Secure MQTT Connections**: Use **MQTT over TLS** to encrypt data transmission.
- **Authentication & Access Control**:
  - Disable anonymous connections to the MQTT broker.
  - Require strong, randomly generated passwords for MQTT clients.
- **Remote Administration**:
  - The MQTT broker runs on the Raspberry Pi, which requires remote management.
  - **SSH authentication is enforced using SSH keys only**, preventing password-based attacks.

Proposed Solution: Application for Customers

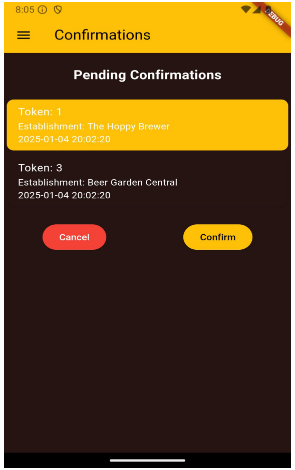
The application supports two types of users, each with access to specific features based on their role.

User Types:

- **Customer**
- Staff

Customers:

- Buy beer tokens from establishments
- Generate and validate tokens
- Spend beer token to activate dispenser through NFC



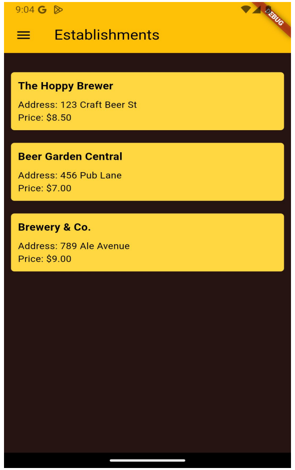
8:05

Confirmations

Pending Confirmations

Token: 1
Establishment: The Hoppy Brewer
2025-01-04 20:02:20

Token: 3
Establishment: Beer Garden Central
2025-01-04 20:02:20



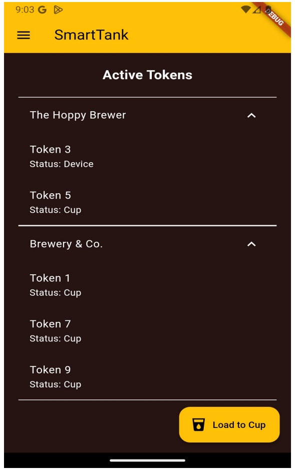
9:04

Establishments

The Hoppy Brewer
Address: 123 Craft Beer St
Price: $8.50

Beer Garden Central
Address: 456 Pub Lane
Price: $7.00

Brewery & Co.
Address: 789 Ale Avenue
Price: $9.00



9:03

SmartTank

Active Tokens

The Hoppy Brewer

Token 3
Status: Device

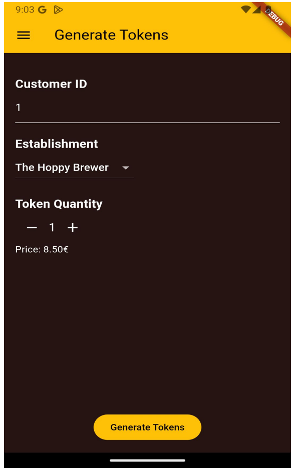
Token 5
Status: Cup

Brewery & Co.

Token 1
Status: Cup

Token 7
Status: Cup

Token 9
Status: Cup



9:03

Generate Tokens

Customer ID
1

Establishment
The Hoppy Brewer

Token Quantity
− 1 +

Price: 8.50€

Proposed Solution: Application for Staff

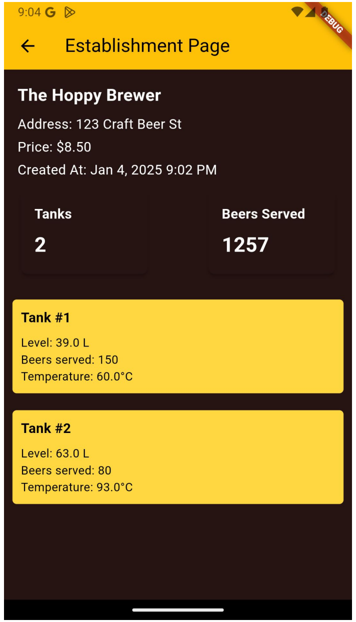
The application supports two types of users, each with access to specific features based on their role.

User Types:

- Customer
- **Staff**

Staff:

- View statistics such as **temperature** and **beer level**
- Assign **beer tokens** to customers



9:04 G

Establishment Page

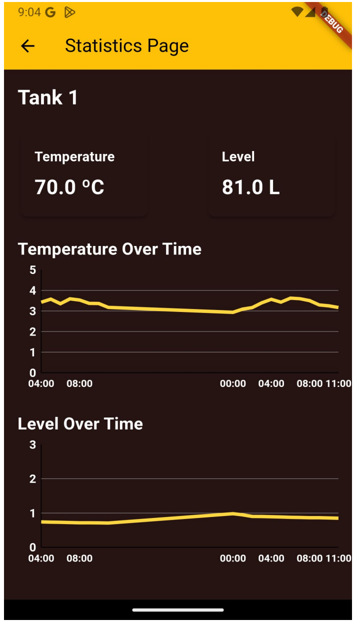
The Hoppy Brewer

Address: 123 Craft Beer St
Price: $8.50
Created At: Jan 4, 2025 9:02 PM

Tanks	Beers Served
2	1257

Tank #1
Level: 39.0 L
Beers served: 150
Temperature: 60.0°C

Tank #2
Level: 63.0 L
Beers served: 80
Temperature: 93.0°C



9:04 G

Statistics Page

Tank 1

Temperature	Level
70.0 °C	81.0 L

Temperature Over Time

Level Over Time

Conclusions

Efficient Customer Experience

- Reduces waiting times and improves service quality
- Automates repetitive tasks, enhancing staff work and entertaining customers

Modular Real-Time Technology

- Cloud integration enables real-time monitoring and insights
- Modular and scalable design for new sensor integration or application for other purposes (like industrial tanks)

Security

IoT solutions suffer of security problems, but

- NFC-based access control enhances security
- Double check validation to overcome NFC corruption

Limitations & Challenges

- Communication latency due to cloud reliance
- Hardware constraints (single-threaded processing, pressure-dependent valve)
- Potential adoption barriers for users unfamiliar with NFC technology

Future Works

System Enhancement and Generalization

- Explore **advanced Edge computing** and **decentralized communication** to reduce cloud dependence
- Improve **real-time responsiveness** with **multithreaded microcontrollers**
- Integrate **machine learning** for **predictive maintenance** and **consumer analytics**
- Develop a **more versatile solution** adaptable not only for **commercial beer dispensing** but also for **industrial and agricultural applications**

Beer Application Expansion

- Implement **automated reordering** from suppliers to ensure **seamless operations**
- Support **different cup sizes** to enhance **user flexibility**
- Improve **usability** based on **user feedback** for better **accessibility and adoption**

And now, let’s enjoy a **NON-ALCOHOLIC** beer together! 🍻

Thank you!

Any questions?

For any information contact me!

piarulli.1884766@studenti.uniroma1.it

Viale Regina Elena, 295, Building E Room 310