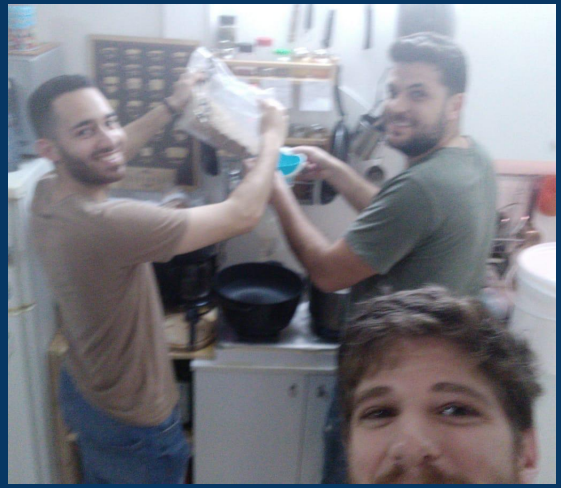
Brew To The Future

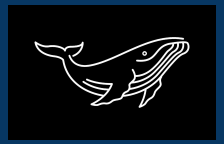
Smart Home Brewery

Eyal Shpater
Matan Hayon
Adi Kapuri

How It All Started



Leviathan Brewery, 2020



Brewing Beer 101

Brew Day
2-3 Hours

Fermentation
2-3 Weeks

Bottling
2 Weeks

About a Month

The Problems We Had

Brewing Day

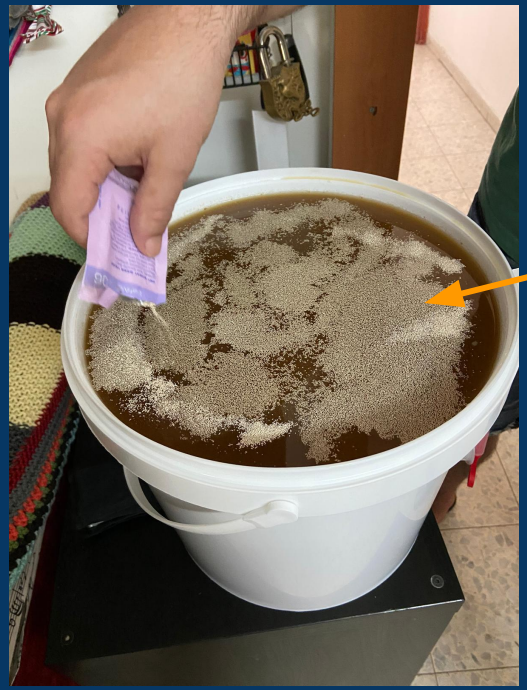
Controlling The Heat Source

Temperature Monitoring

EMA CODING

The Problems We Had

Fermentation



Temperature Monitoring

The Problem

Challenges Faced by Homebrewers:

- Manual tasks lead to errors and inefficiencies.
- Lack of automation affects brew quality and consistency.
- Difficulty in troubleshooting and fine-tuning recipes due to absence of data logging.
- Limited remote monitoring capabilities restrict brewers from overseeing the process effectively.
- Recreating favorite beer recipes becomes challenging.

Our Solution

Smart Home Brewery System:

- Automation of brewing operations
- Smart sensors for monitoring brewing temperature, fermentation, and easy brewing control
- Real-time alerts and insights

Our Solution

The Benefits:

- “Plug & Play” - Simplifys the brewing process while minimizing errors
- Consistency: Achieves consistent results across multiple brewing sessions, enhancing reliability.
- Data-Driven Insights: Provides valuable data analytics to help brewers understand and improve their brewing techniques.
- Community Sharing: Facilitates sharing of recipes and tips within a community of brewers, fostering collaboration.

Main Features

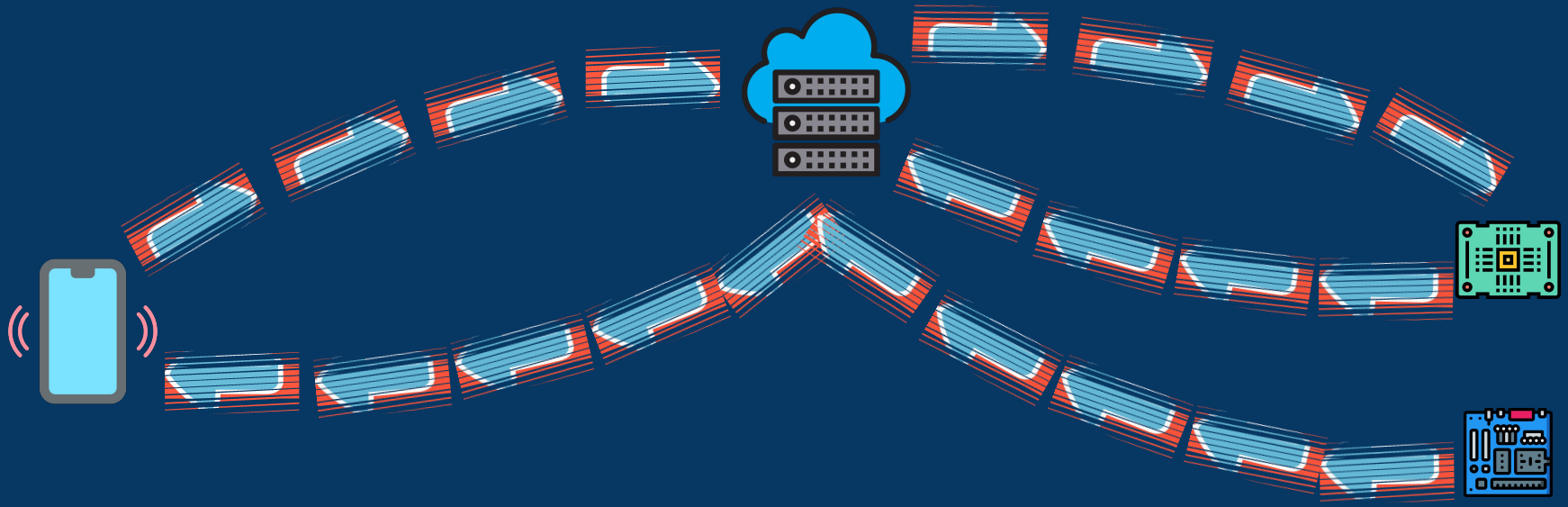
Our project's main features:

- **Brew a recipe:** Offer a user-friendly interface for creating brewing recipes, integrated with an IoT system for automated temperature control and execution. Provide an intuitive control interface displaying recipe stages during the brewing process.

- **Share & Download recipes online:** Enable user profiles for sharing and downloading favorite recipes, while allowing users to contribute, rate, and review recipes to foster a collaborative community.

- **Brewing history:** Display data for each brewing session, along with search functionality to locate sessions by various criteria.

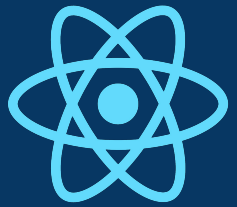
General Architecture



Technology Stack - Frontend

FRONT-END:

- **Development language:** JS
- **FrameWork:** React Native
- **Application:** Mobile
- **Api Communication:** HTTPS

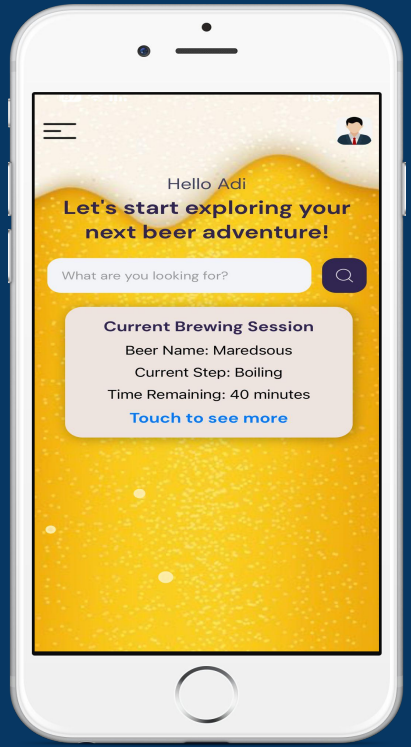


React Native

Screens - Frontend

‘Welcome’ screen

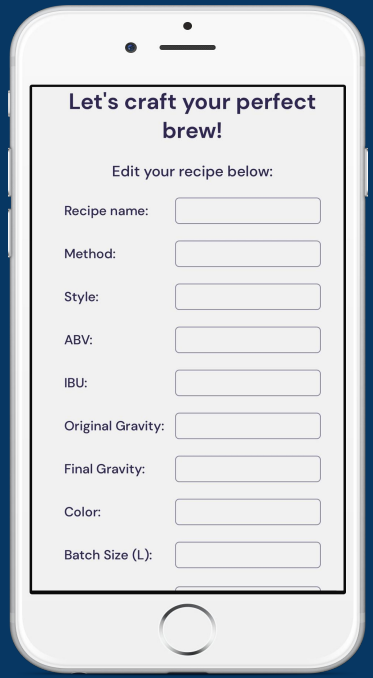
The first screen is the **‘Welcome’** screen, serving as the entry point to the app, featuring a menu that navigates users to the diverse features.



Screens - Frontend

'Create a Recipe' screen

This screen is the **'Create a Recipe'** screen, where users can input desired ingredients and brewing data to add to their personal recipes collection.

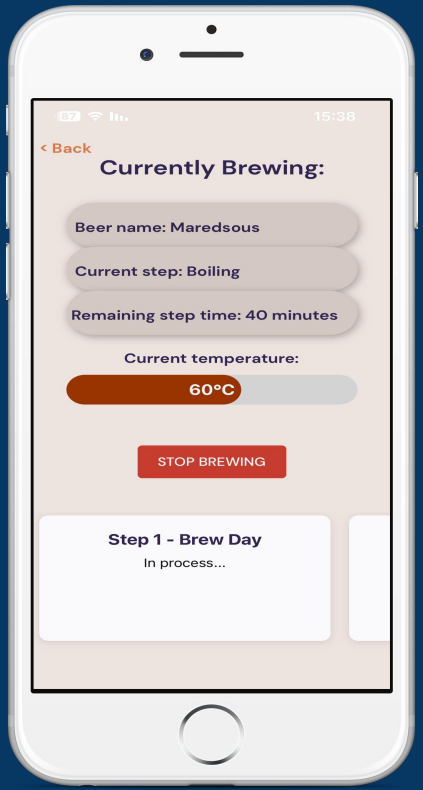


EMA CODING

Screens - Frontend

'Current Brewing' screen

This screen is the **'Current Brewing'** screen, where users can see the current brewing data, such as the current beer temperature, brewing step and time left for current step to end.



Technology Stack - Backend

Development language: Java

FrameWork: Maven, Spring Boot

Database: PostgreSQL

Architecture: Microservices using REST API

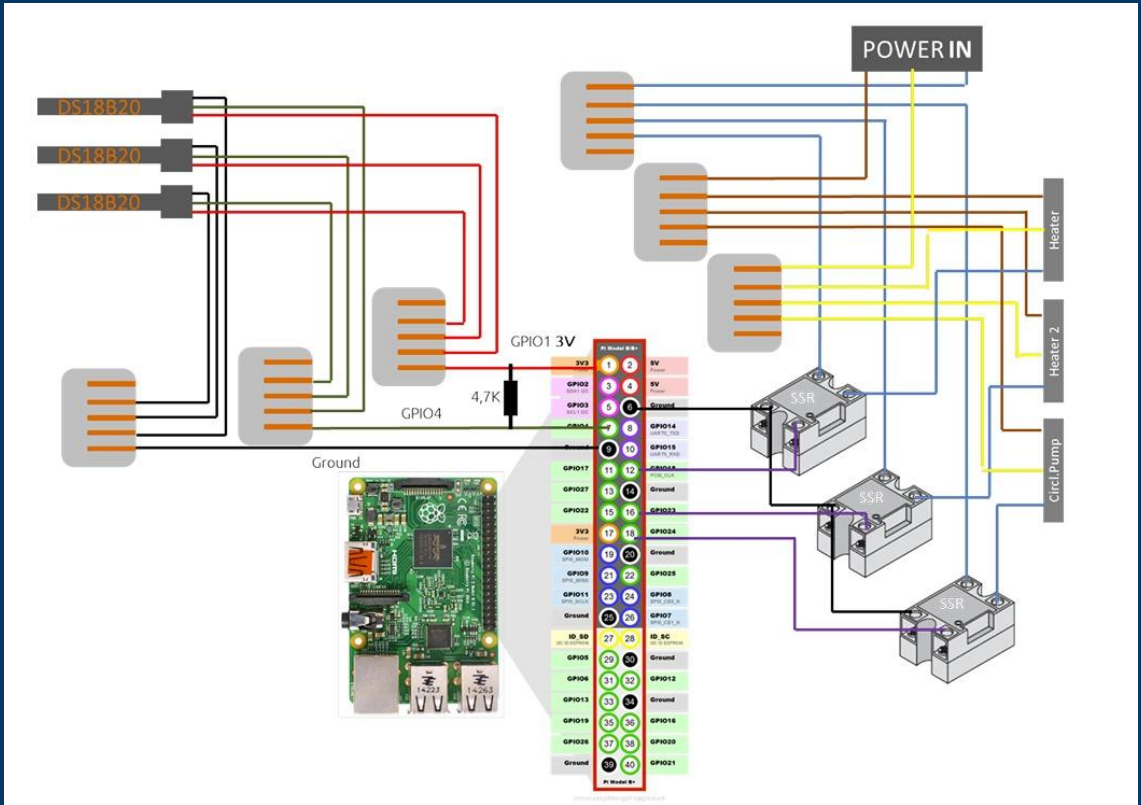
Cloud: AWS

The Brewing System

The main system used to brew beer:

- Main processor: Raspberry Pi 4 - fast, durable to run complex programs
- Temperature sensors: ds18b20
- SSR 3-32VDC -> 24-380VAC
- Heat Sink
- Brewing Kettle

The Brewing System



The Brewing System

Technology:

- Operating System: Linux
- Language: Python
- Wireless communication

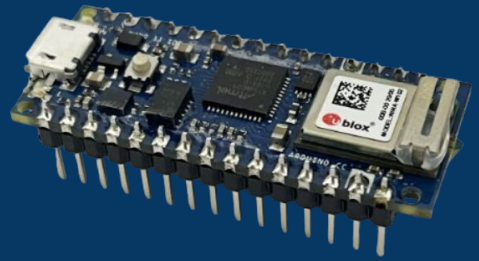
Linux

python

The Fermentation System

Will be used to monitor the fermentation process for 2-3 weeks:

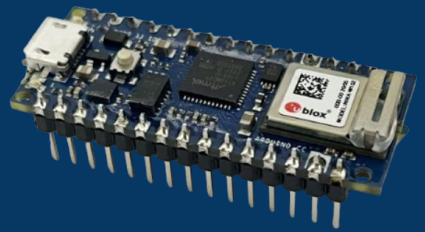
- Main processor: Arduino nano 33 iot
- Environmental sensor BME680: pressure, temperature, humidity



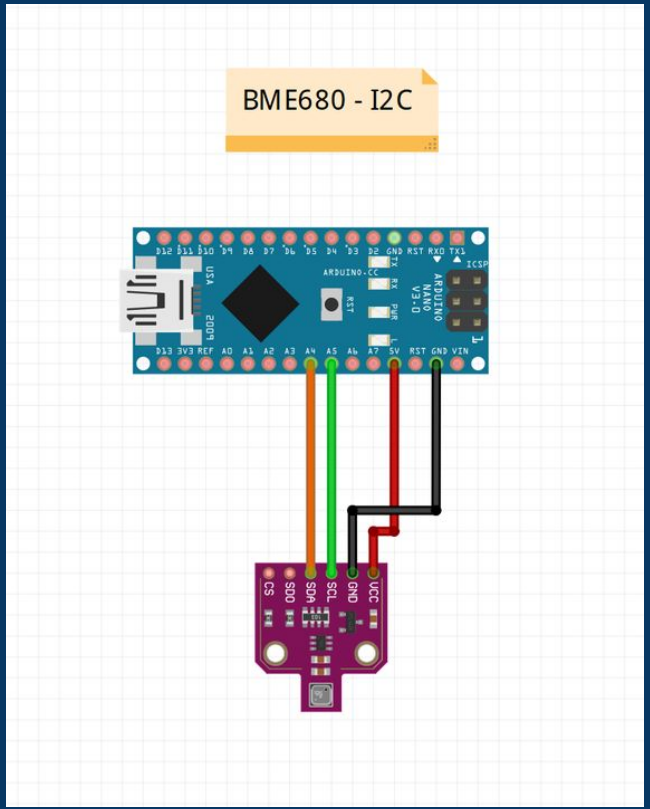
The Fermentation System

Technology:

- Language: C++
- Wireless communication



The Fermentation System

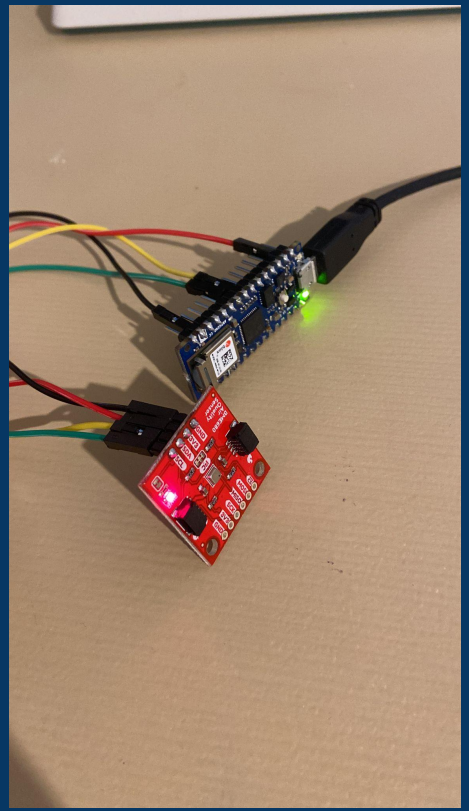


The Fermentation System

Serial Monitor ×

Message (Enter to send message to 'Arduino NANO 33 IoT' on 'COM

```
17:37:22.326 ->
17:37:24.656 -> Temperature = 25.67 *C
17:37:24.697 -> Pressure = 1007.72 hPa
17:37:24.697 -> Humidity = 61.08 %
17:37:24.697 -> Gas = 100.66 KOhms
17:37:24.697 -> Approx. Altitude = 46.14 m
17:37:25.067 ->
17:37:27.398 -> Temperature = 25.68 *C
17:37:27.443 -> Pressure = 1007.71 hPa
17:37:27.443 -> Humidity = 61.13 %
17:37:27.443 -> Gas = 101.09 KOhms
17:37:27.443 -> Approx. Altitude = 46.14 m
17:37:27.802 ->
17:37:30.171 -> Temperature = 25.67 *C
17:37:30.171 -> Pressure = 1007.72 hPa
17:37:30.171 -> Humidity = 61.13 %
17:37:30.171 -> Gas = 101.27 KOhms
17:37:30.171 -> Approx. Altitude = 46.14 m
17:37:30.547 ->
17:37:32.882 -> Temperature = 25.67 *C
17:37:32.919 -> Pressure = 1007.72 hPa
17:37:32.919 -> Humidity = 61.11 %
17:37:32.919 -> Gas = 101.33 KOhms
17:37:32.919 -> Approx. Altitude = 46.14 m
17:37:33.260 ->
17:37:35.662 -> Temperature = 25.66 *C
17:37:35.662 -> Pressure = 1007.72 hPa
17:37:35.662 -> Humidity = 61.05 %
17:37:35.662 -> Gas = 99.11 KOhms
17:37:35.662 -> Approx. Altitude = 46.14 m
17:37:36.035 ->
```



EMA CODING

Thank You!

EMA Tech