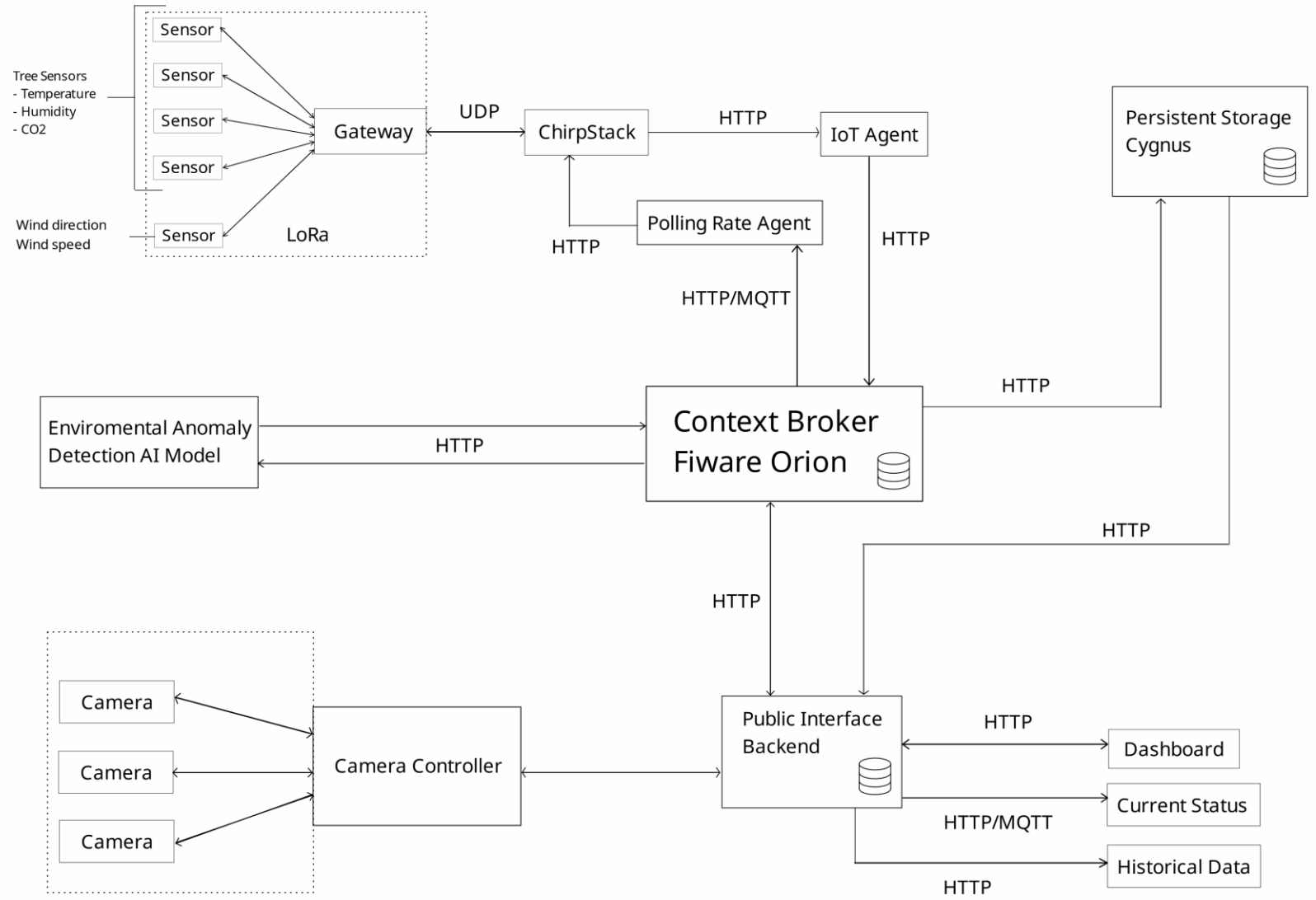


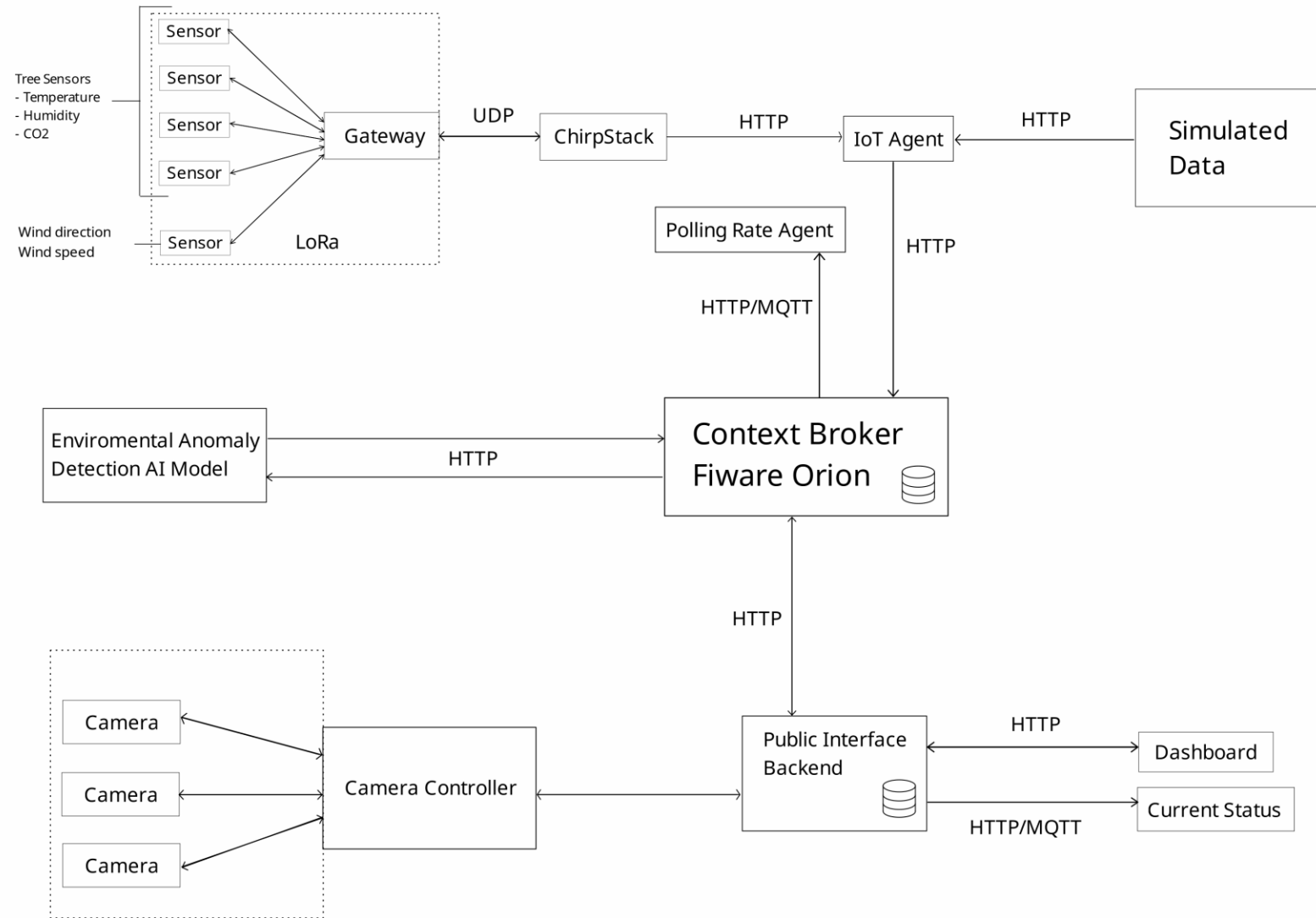
FireGuard Systems Architecture & Implementation

Αλκίνοος Αλυσσανδράκης
Μιχάλης Καΐπης

Ideal Architecture



Architecture
to be
implemented



Data models



Tree Sensor

dateObserved (date-time)
location (GeoJSON)
serialNumber (devEUI)
value (";" separated string)



Wind Sensor

dateObserved (date-time)
location (GeoJSON)
serialNumber (devEUI)
value (";" separated string)

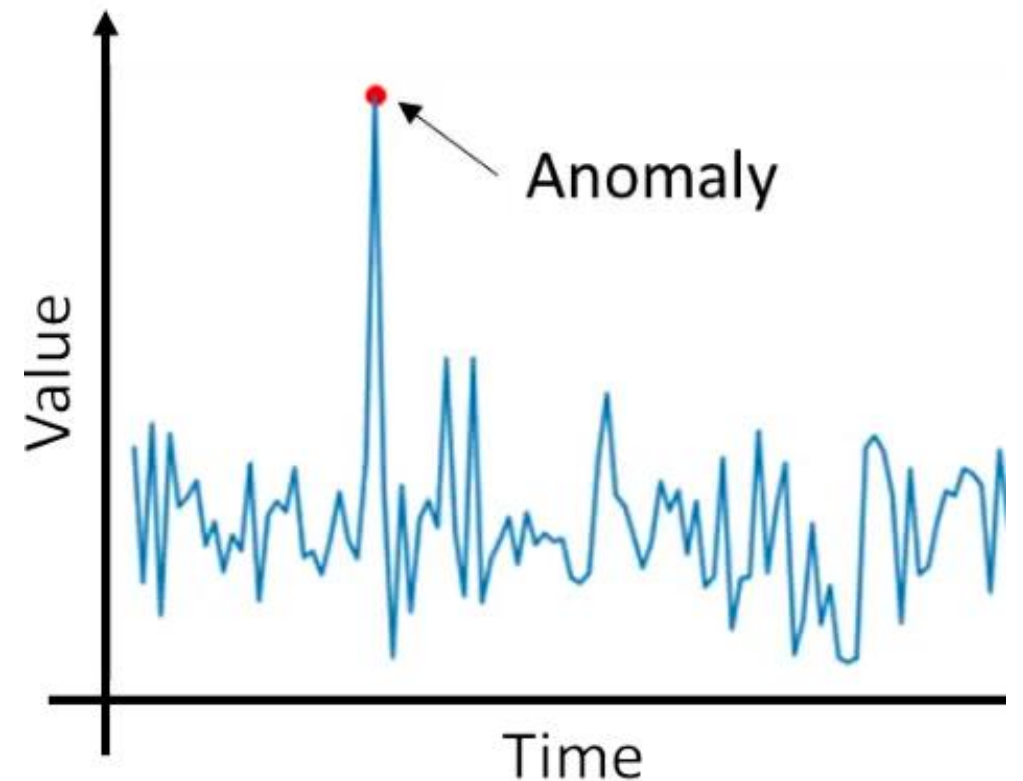


Fire Forest Status

dateCreated (date-time)
fireDetected (bool)
fireDetectedConfidence (number [0-1])
fireRiskIndex (number [0-1])
location (GeoJSON)

Detection AI

In the realm of data analysis, our approach to anomaly detection, a method designed to identify rare occurrences, events, or observations that deviate significantly from the norm within our dataset. These anomalies represent instances that stand apart, not conforming to the expected patterns of normal behavior. By leveraging this method, we aim to enhance our ability to discern potential fire events through the data collected by our advanced sensor network.



AI Training/Prediction Process

Gather Normal Data: { date ,wind speed, co2 , humidity, temperature}

Normalization Process for limiting the values to a certain range of values

Model definition: model type "autoencoder"

Ένας αυτοκωδικοποιητής αποτελείται από δύο κύρια τμήματα:

- **Ενσωματωμένος Κωδικοποιητής (Encoder):** Αυτό το μέρος του μοντέλου δέχεται την είσοδο και τη μετατρέπει σε μια πιο συμπαγή αναπαράσταση, γνωστή ως κωδικοποιημένος χώρος.
- **Ενσωματωμένος Αποκωδικοποιητής (Decoder):** Αυτό το μέρος αναλαμβάνει να ανακτήσει την αρχική είσοδο από τον κωδικοποιημένο χώρο.

Στο τέλος της διαδικασίας, η έξοδος του αποκωδικοποιητή προσπαθεί να είναι όσο το δυνατόν πιο παρόμοια με την αρχική είσοδο.

Training: normal Data -> [encode -> decode] -> almost same data

For the testing we expect to have:

normal data -> [encode -> decode] -> normal data

not normal data -> [encode -> decode] -> huge difference in data

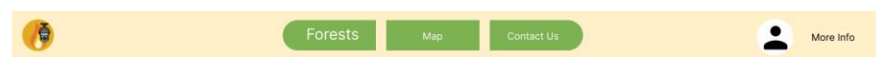
Validation Metrics: recall (ευαισθησία) = $TP / (TP+FN)$, precision (ακρίβεια) = $TP / (TP+FP)$

If we are not ok with the results we retrain the model on different dataset

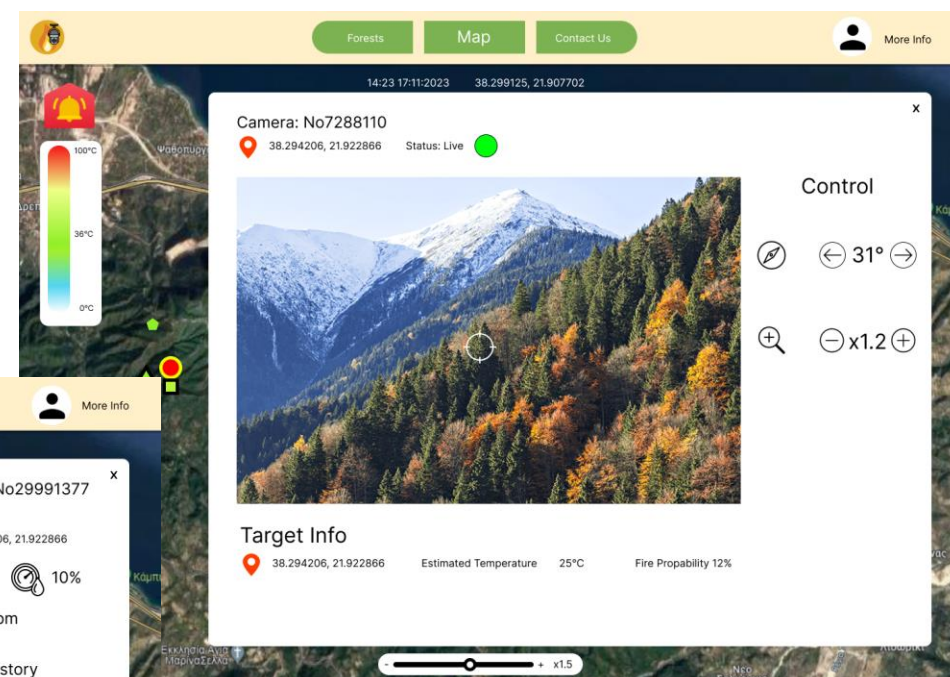
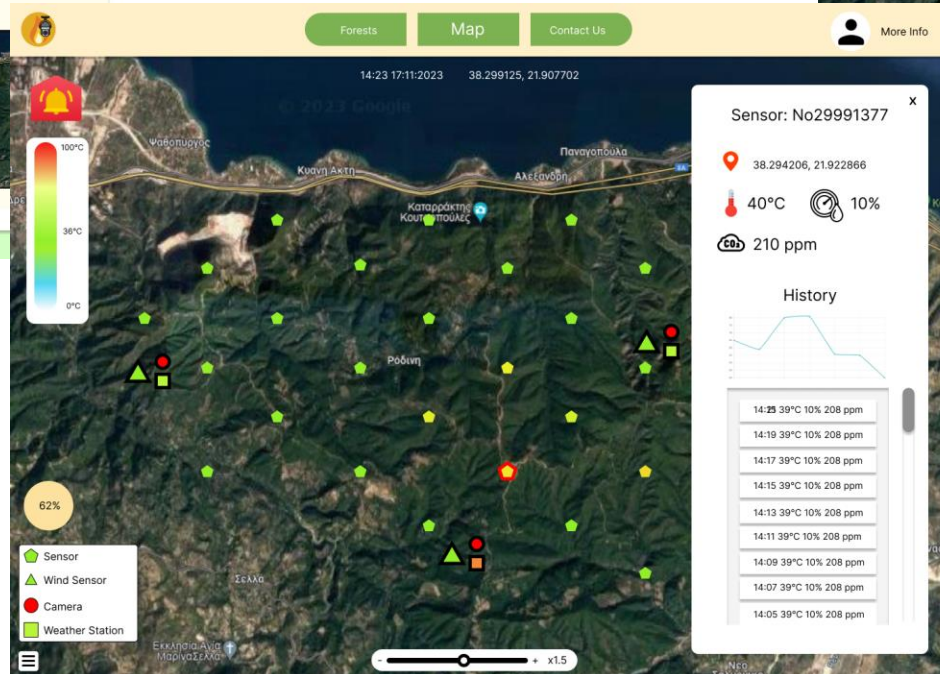
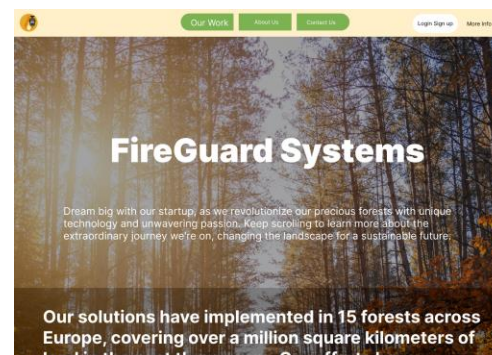
Based on the difference we map it from 0-1 so we have our fire probability estimation for the one sensor at that time. We do this for every sensor data to estimate the fire probability of the forest by :

fire probability of the forest = $1 - \text{Mult}(1 - \text{modelEstimationOfData}(\text{data}) \text{ for data in sensorData})$

Dashboard mockups



Forest Three



Βήματα υλοποίησης

Υλοποίηση αρχιτεκτονικής
(context broker, backend, ...)

Προσομοίωση δεδομένων

Μοντέλο αναγνώρισης
περιβαλλοντικών ανωμαλιών

Διεπαφή διαχείρισης του
συστήματος

