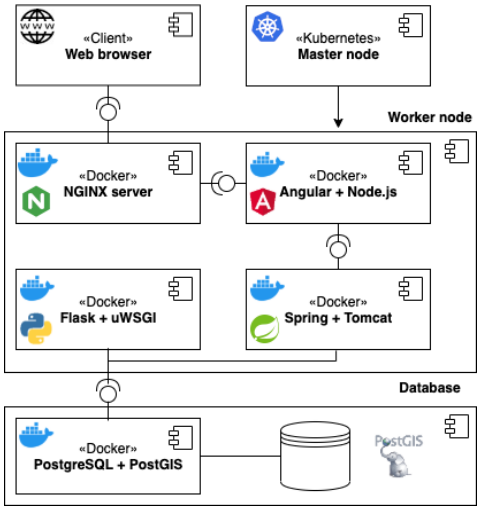
Technical Specification Double-side Page

1. **TECHNICAL SCOPE:** The mock-up solution is suitable and correctly addresses the challenge/theme selected over the REACH dataset/s. The Big Data solution architecture proposed is adequate to tackle the data management issues associated to the solution in mind. "To what extent does the applications handle the data provided?"

The front-end is based on a **GIS platform**, where users can just click on their field and immediately get the **recommendation which variety to plant and what yield they can expect**, the two primary requirements set by the Data Provider. On the backend, the GPS location is scraped, weather/soil/elevation data is retrieved and based on these inputs, the yield of all soybean varieties is predicted. The one with the highest yield is then recommended to the user. In the Experiment phase we will add other statistics, satellite-based info, risk analysis, and other auxiliary data-modules to the platform (login, user profile, field drawing...). The system will be operational over a large area (country/regional level) at the 10m resolution (the highest resolution of a data source), where dense time-series need to be processed. For this reason, the system architecture is optimised for handling spatial data and quick execution of the AI algorithms.



Regarding the system architecture, every **Python** component is containerised with a separate **Docker**, while **Nginx** serves as a proxy server for accessing the system components. The front-end is implemented using **Angular** framework and **Node.js** and in the back-end, **REST** API application is implemented using **SpringBoot** running on a **Tomcat** Web server. In order to optimise the workflow, algorithms developed in **Python** and **PySpark** are exposed as a **REST** service using **Flask** running on **uWSGI** Web server. In order to orchestrate the **Docker** containers within the Working nodes of the cluster, we are using **Kubernetes**. Last but not least, **PostgreSQL** is used as the database with **PostGIS** extension for spatial data.

2. **SELECTION OF ALGORITHMS AND TOOLS:** The indicated Data Science approach, i.e. algorithms chosen, and Big Data architecture approach, i.e. tools chosen may successfully accomplish the required data governance, processing and analysis. A clear understanding of the used REACH dataset/s is demonstrated.

The primary goal of AI in this project is to model the growth of soybean varieties and predict their yield at a particular location. Every location in agriculture is defined by the weather, soil type and terrain, which are the input features, while the output feature is the yield (ground truth from the Data Provider). Every sample from the Delta's dataset contains the precise location that serves for downloading the raw weather/soil/elevation data. The system is composed of the following components:

Feature engineering – C3S weather data comes at 1h resolution and the agricultural season typically lasts for around 6 months. This quantity of data would be disproportionate to the number of ground truth samples so the number of input features for ML needs to be reduced. For this reason, we are using a smaller set of features that are based on agronomic knowledge and that well define weather conditions in critical phases of plant growth. Similar holds for SoilGrids that provides more than 200 soil features, so PCA, LDA, autoencoders and other dimensionality reduction techniques will be employed.

Yield prediction – models are trained on weather, satellite and soil data, with Delta's field records as ground truth. So far, XGBoost has proven to have superior performance over Random Forest, SVM and Deep Neural Networks (DNN) in general yield prediction. Here however, the existence of multiple soybean varieties means that there is a categorical feature (variety), that needs to be tackled either with encoding (one-hot / embeddings), or with algorithms that intrinsically deal with this (catBoost, histGradientBoosting, DNNs...). All models will be evaluated using leave-one-year-out cross-validation, in order to avoid data leakage (mutually correlated samples from the same year should not appear in both the training and the test set).

Big Data analytics - in order to scale-up the models successfully over a large area, we will use PySpark as the Big Data framework. PySpark allows for execution of Python scripts in batches or in parallel, which has the potential to lower the algorithm runtime, especially due to complex data structures present in the dataset (with various temporal and spatial resolutions, size and amount of data).

3. **TECHNICAL SCALABILITY AND FLEXIBILITY OF THE SOLUTION:** The solution can truly cope with humongous and increasing datasets, potentially from diverse data providers, and is flexible it to adapt to other related domains.

In order to produce a scalable solution, we are using free global data sources. C3S (**weather** data source) operates on a worldwide level, ESA's Sentinel 2 **satellites** have global coverage at a 10m resolution, while SoilGrids maps are issued by the world **soil** association (ISRIC) and come at a 250m resolution. The only local data that is required is **field data**, needed for calibration of yield prediction algorithms in local growing conditions. For this, usually a few hundred local yield samples are required. For this reason, we are collaborating with various agricultural companies (K+S, MK), research institutes (BioSense, NS Seme) and seed breeders (Delta, Syngenta) who are providing us with local data.

After the successful implementation with Delta, our aim is to cover other crops (maize, wheat etc.) and serve companies around the world. For this, we would need to recalibrate our models for new geographies. This could be done using completely **new datasets**, but in order to scale most efficiently we will use the innovative **transfer learning** approach (with scaling, shifting, surrogate data generation, reinforcement learning, ML models coupling etc.), and enable quick and non-demanding deployment in new markets. Transfer learning here allows for using the trained model in different regions taking into account the different planting dates, season lengths etc., but without having the need for a large new dataset. For model transfer, only few data points are needed, making the models more easily applicable in new settings.

4. **DATA GOVERNANCE AND LEGAL COMPLIANCE:** Data sharing challenges, data governance and legal compliance, must be observed. Selected methods and technologies to access and manage data assets have to be described. The security level of the solution, i.e. how authentication, authorization policies encryption or other approaches are used to keep data secure, are well explained. The proposed solution is compliant with the current data legislations concerning security and privacy (e.g. GDPR).

For crop growth modelling we are using **open-source** datasets from Copernicus and ISRIC that are available for commercial use. Regarding the field data, an **NDA** has been signed with Delta to secure that the company's data is not shared with any 3rd party. These datasets do not include any personal information, only data about agricultural operations, fertilisers/pesticides used and yields. All data will be safely stored at Cropt's cloud-based servers (Hetzner) and all Cropt's data analysts and software developers have signed the **confidentiality agreement** that prevents them from sharing the classified information with a third party. Furthemore, raw data processing will be conducted only within the production system allowing only authorized users to access and process the secured data. Security is enabled through protected access to the administration interface with SSL encryption via https protocol and transferring data (up, and downloads) via FTPS/SFTP. Conformity to the related regulatory requirements for full quality and performance will be pursued, namely the **GDPR** to ensure that all users' information and data is kept private, and the rules for the transfer of data outside the EU. Based on the client's requirements, we will consider **ISO 27017** and **ISO 27001** for security of cloud-based information. We will closely monitor different and fragmented national safety rules across the EU and other markets. This will help us determine if the profitability makes sense for the investments in a particular go to market strategy or opening new routes to market.

5. **QUALITY ASSURANCE AND RISK MANAGEMENT:** Feasible and credible quality process followed for the final product generation. The potential risks in all the phases of the project (design of the solution, development, testing, deployment...) are identified and convincing mitigation plans put in place.

Leave-one-year-out **cross-validation** of Big Data models will reveal the accuracy of yield prediction. This is the most rigorous validation technique which gives the lower boundary for the accuracy, while the 10-fold cross-validation across the years gives a more optimistic assessment (the upper boundary). However, besides ML performance estimation we are heavily relying on **client feedback**. Good performance metrics do not have to mean that the client will be satisfied. For this reason, we are producing intuitive results, such as regional maps and charts with predicted yields in different weather scenarios which agronomists and seed breeders can understand well. In line with **agile** principles, we will have regular meetings with the Data Provider and immediately update the system according to their feedback.

The main project deployment risk is **Covid-19** (probability: 30%, impact 5%). We are preparing to work from home and remotely access the servers / computers, while the team meetings will be held regularly over Zoom. We will assure that we have workforce redundancy, so that every person could be substituted by another one in case someone is affected. The main design-related risk is that **deep learning** may not give satisfactory results for yield prediction (p:50%, i:5%), especially if data proves to be too scarce for complex networks. In this case, we will use conventional ML algorithms such as XGBoost, catBoost, Random Forest etc. The risk that may come up during testing is that AI models are too **time demanding** (p:20%, i:5%). In this case we will parallelise the workflow and use code speed-up techniques (e.g. cython) to allow for faster execution, and to prevent this, we will rely on PySpark from the start.