instinctools

Product Engineering For an AgTech Manufacturer

How developing a Pictureline web app empowered an agricultural startup to bring the idea of real-time crop assessment and farming support to life and flourish, along with 10,000+ SQUIDS users.

Business challenge

There are industries, such as agriculture, that alongside tough environmental challenges, face a lack of resources and extreme qualified labor shortages.

Our client is an example of an innovative agritech startup that is solving modern farming – **launch offers automated AI-powered crop health assessment, yield forecasting, and farming recommendations specializing in medical cannabis' production.**

The team's mission is to empower farmers with data-driven insights to optimize their operations and boost crop yields.

Before switching out to us the client had already had:

- **Blue/UV image** that identify stack speed
- **V-color - Magic** harmonic collecting data from the plants
- **IR - Nitro** - for analyzing the data

The client's goal was to streamline the plant care process in several aspects:

- Replace manual labor with **automation** to save a sizable portion of the company's budget
- Substitute a lack of scientists for **intelligent sensors**
- Spot problem plants in the early stages of mildew, botrytis, and other diseases to prevent crop loss and **standardize** their production process
- Detect pests early (aphids, broad mites, thrips, etc.) and **hyperspectral imaging analysis** on one of the client's CANN plants powered by the UV color TIFF, level maps, and automated recommendations for pest control in farmers
- Monitor plant health and growth with **detailed real-time analysis to make analytics** instead of guesswork to optimize and increase yields

However, **there was no software that would resolve the data from the AI-powered robot and visualize it in a detailed way** for the end users. Farmers. The client needed a **SaaS** solution and a custom web application that would deliver the purpose of our project.

How does the client's solution work?

Instinctools' dedicated team had to craft:

Frontend

capable of displaying images of farms with 2,000+ plants.

Backend

able to process up to 2,000 requests from the frontend in a second.

Visualization

of the map itself as the data collected by Hugh (smart robot) is analyzed by Data and is a unique and intuitive portal.

User-friendly visual design

for the web app.

Solution

PoC and prototype

We assembled a **core team consisting of a business analyst, product designer, and software architect** to more effectively deal with the client's requirements and extensive first-time project.

Puzzling over the map of plants

Identifying the best way to visualize a farm's map and finding the optimal way to display the information was the most demanding technical challenge of the entire project of this white project.

We had to kindly ask them not to overcomplicate, so **we decided to start with a simple map with tags only, where a cell equals a plant.**

As the initial implementation of the project, we chose **Vitpr – a JavaScript library** for bespoke data visualization, as a whole.

Identifying the most appropriate framework for the back end

According to our client's requirements, the **Flask web back-end had to handle up to 1000 requests from the client's side in a second.** Flask's lightweight nature and **asynchronous framework** seemed most appropriate, specifically for fulfilling high-performing web applications. Flask's API was also the best fit.

Looking for the best way to transfer the original farm map structure to the app

At the stage, the best positive question – what is the best way for our frontend and backend developers to transfer the original farm map structure to the app? Our team considered several options and chose **Leaflet** – a JavaScript library for mobile-friendly interactive maps. Leaflet's simplicity, extensibility, and performance allowed us to create a custom farm map structure.

Finding the optimal way to store and process the incoming subsections cause 80 KB a user. We considered two options: **Protobuf** and **Apache Arrow**. The client's goal is Protobuf for left users.

- Fast to code
- Fully compatible with the declarative language JSON-based schema which allows to create a **validator** and **generate** code for the client and server **validation** on gets field-wise
- Compatible with Client-Server communication scenarios
- Protobuf's binary format is more compact than JSON, consequently, Environment's Gzipping
- Binary log size and speed are better than JSON
- Leaf's file structure needed for the root

Protobuf

- Data compatible with the declarative language JSON-based schema which allows to create a **validator** and **generate** code for the client and server **validation** on gets field-wise
- Compatible with Client-Server communication scenarios
- Protobuf's binary format is more compact than JSON, consequently, Environment's Gzipping
- Binary log size and speed are better than JSON
- Leaf's file structure needed for the root

Apache Arrow

- Fast to code
- Fully compatible with the declarative language JSON-based schema which allows to create a **validator** and **generate** code for the client and server **validation** on gets field-wise
- Compatible with Client-Server communication scenarios
- Binary log size and speed are better than JSON
- Leaf's file structure needed for the root

What **user functionality** was provided in the prototype?

Ability to log in and sign in with Google and Microsoft credentials

Digital twin of a greenhouse

Access to key data on each plant

Information visualization on the plants' health status and their condition based on the analysis of a whole greenhouse, updated once a day

MVP

The app's **prototype was successfully tested on a Canadian farmer's greenhouse** where Pictureline's feedback provided future improvements and updates to the MVP development stage. Instinctools' team upgraded the UX flows once and expected the solution functionality for the client's satisfaction.

The scale and view of the map and the ability to search it

As having a map level was over mandatory for companies that cultivate greenhouses, we suggested **reducing the map's scale and adding the ability to search it.** Our team also added a feature that allows users to access the Greenhouse's GUID quickly and easily.

As you can see on the mock-up, we've **added grey** and **green** colors to the map's cells, which allowed farmers to see which cells/plants it is to cover and as Hugh's location, which helps to do a GUI log in the Greenhouse. Farmers can also see the Plant's Name/Tree grey indicating a crash plant.

Enhanced UX

Whereas in the prototype users could only see information on a specific plant, the **overview is visible on the scale of an entire farm.**

Broader amount of data collected about plants

Work using **hearing of mildew, botrytis, and pests** – the client's goal was to **provide farmers with a tool** for **determining hyperspectral imaging** (HSI). The client's goal was to provide farmers with a tool that would be able to determine the pests and mildew.

Hyperspectral camera's output matrix is 1000 x 1000 pixels. Only one was a closer with the following data:

- Leaf number of the Plant ID
- TIFF index
- CSV index
- Mildew percentage
- Botrytis percentage
- Pest presence

Select mode on the map with the ability to see statistics for the chosen area

While in the prototype, it was possible to see statistics for a specific plant, we added a feature that allows the user to select any area on the map and check up on the minute data about it. This functionality works in the same way for all charts.

More detailed filtering with the option to set check-boxes on several tracking layers simultaneously

With multiple filters enabled, **utilizing the filtering** was one of the project's toughest challenges. Filtering, the user's ability to see the data they need, was a crucial feature for farmers. Our team's solution was to **make multiple layers** and **combine** them with a "match" value and combinations each.

They also suggested **adding a drop-down menu** and **a combination of parameters from the different map layers.** The client's team also suggested **adding a "Reset" button** for quick navigation. We also added a **zoom-out button** and **hover maps overlay at the** farm's map.

Here the map has two layers.

Plants

Environment

Real-time data updates

As the greenhouse's environment is the greenhouse, we had a hope that one day **Hugh** would be **a robot that would be able to provide real-time data** and **increase productivity of everyone in employ.** Therefore, we ensured that Hugh's the smart sensor's output would be updated in real-time. Our team's solution is a prototype could provide an **up-to-date** data using the **Cron** job.

MVP evolution

The MVP version of the app was successfully tested on the Canadian breeder's farm. Pictureline's feedback provided future improvements and updates to the MVP development stage, which expanded its features and upgraded or improved its UX flows.

Ability to view the greenhouse history

Initially, our provided recommendations were based on a "short-term" data gathering and, as an individual plant to a whole farm, **Hugh's data** was only available for a short period of time. Only data from previous periods, Hindsight, gives better plan the following season.

Leaving notes on the map

Each cell's data needed to be kept and people mentioned that would be required to work with the data. Our solution's convenient With such an approach, if an employee discovers a problem, he/she can leave a note and inform the relevant people about the issue, which helps to improve operations starting from either shifts to native developers through sales.

Different roles within the app

There were no different roles in the app's employee system. Hence the client's goal was to provide greenhouse's big data access to the users. Therefore, we suggested adding **three roles** – **Admin, Farmer, and Viewer** – each with different levels of access to features and data of the reports.

- **Greenhouse managers** can only check the greenhouse's data and manage the plant's data, and add/remove farmers.
- **Farmers** can only access the greenhouse's and all farms' data, but they are not able to "mix" reports.

Key features

Innovation

Spex.ai is a trailblazer for intelligent and sustainable farming, specifically in cannabis monitoring and analysis.

Profitability

The software unleashes two profit-boosting powerhouses: automated plant health assessment reducing the number of plants that need to be disposed of due to non-safe THC levels, and early disease detection.

Scalability

The solution's adaptability can be used both for medical and recreational cannabis and 2,000+ plants in huge farms with tens of thousands of plants.

Business value

- Thanks the solution, the farmers can focus on collecting the data of healthy cannabis plants in greenhouses.
- 100% automated plant monitoring and analysis.
- Increasing current revenue in a range of hundreds greenhouses.
- Increasing profit and cash-outlayed at the end of a harvest.
- Minimized risk of plant production because of the revealed THC level.
- Decision-making backed up with up-to-date actual information and historical data.

Client's testimonial

"We're pleased with Instinctools' work. Their business analysts are exceptional. They serve as the epitomizes between technology and business, representing both sides effectively.

I'm also impressed by Instinctools' transparency and open communication. The team is dependable when it comes to managing time and finances, consistently staying within the designated budget. And actually the "Instinctools team deliver before the deadline, which is fantastic.

Nadine Walther
Co-Founder & CEO, Spex.AI Limited

Do you have a **similar project idea**?