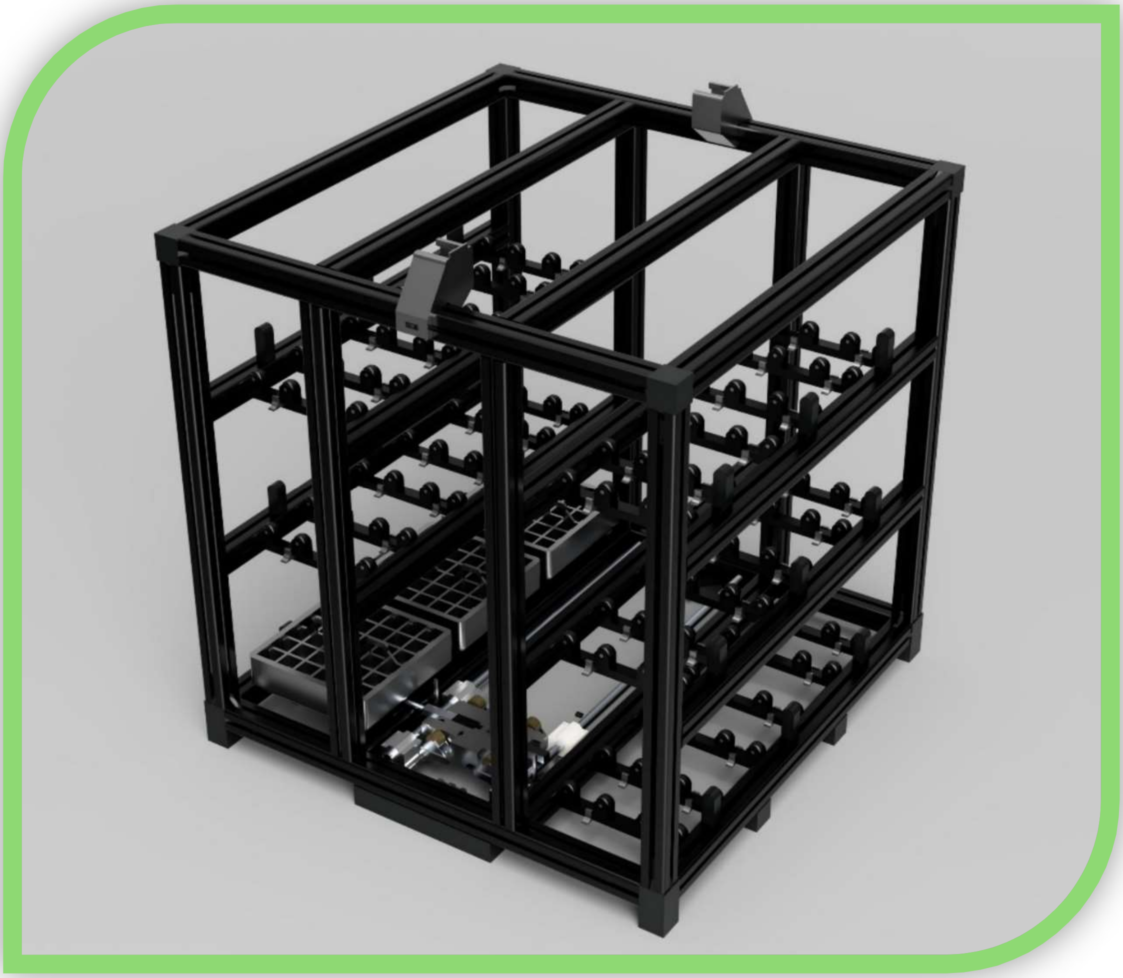


Farmatic

A fully automated, Modular, Vertical Farm

By Jakob Gutersohn and Valéry Piot

Coach: Julian Levkov

Table of contents:

Our Team

We are a small team of three: Jakob Gutersohn, Valéry Piot and our coach Julian Levkov. On our way to this Olympiad countless other people have supported us, especially as our project has been growing. We would like to thank all those people, as they made this project possible.

As our project is diverse and many different systems are involved, we decided to split up the responsibilities. We made sure to have a seamless integration between our hardware and Software. This was of big importance, especially in the testing phase. So, Jakob was responsible for the Hardware and Valéry for the Software. Our Coach Julian helped us to set and achieve our goals. He motivated us in difficult moments, for example when we had a lot of exams. He motivated us for the project and for learning.

Jakob Gutersohn

Jakob Gutersohn, from Bern, is currently in his first year at Gymnasium Kirchenfeld. In his free time, he enjoys working on technical projects. For this project, he took responsibility for the hardware, leveraging his significant experience in 3D design and prototyping. Jakob made the PCB design and designed most physical parts of the Farm including the lift system.

Valéry Piot

Valéry Piot, also from Bern, is in the same class as Jakob. In addition to his various technical projects, he enjoys playing tennis in his free time. Thanks to his expertise in software development, Valéry took responsibility for writing most of the code and developing the corresponding app for Android and Windows devices. Furthermore, he led the implementation of our AI model into the software.

Julian Levkov

Julian Levkov is our coach. He is currently studying Cybersecurity at the EPFL in Lausanne. We first made contact over our English teacher, which advised us to team up with him and choose him as our coach. In this project he helped us with setting realistic goals and showing us the challenges of the Project, so we can prepare better. Thanks to him we've reached our goals on time.

Project Idea

Space has always been of interest to us. That's why we decided to choose the subcategory **"Robots assist life in space"** with a focus on **"robots build living habitats and find resources."** However, our robot is also compliant with other subtopics, as it features an AI and helps to safe resources and space in cities. Furthermore, our robot supports different SDG'

Initially we didn't have any specific ideas, so we decided to have a look at what is essential if ever humans lived on Mars. We quickly figured out that food still is an unsolved problem. The transport of food is expensive, and it currently seems impossible to plant anything into the Martian soil. Therefore, an artificial farm is needed.

Multiple space agencies have already figured out how to build Mars habitats which protect the inside from the dangerous radiation and the harsh weather. So, we already knew there is a safe environment for a farm, **but we asked ourselves how we could use the space as efficiently as possible and how a potential farm would be brought to mars.** The only solution for these issues is to go vertical and to go modular.

Astronauts are perhaps the busiest people in the universe, they follow a tight schedule of experiments, research and regeneration. So, they won't have much time to build and maintain a farm, to provide food for themselves. Because of this and the other mentioned reasons we decided on our robotics project: **A fully modular, automated vertical farm**.

We began with informing ourselves about the topic and the possibilities. We also had a look at the need for this robot. Jakob had the chance of **talking to Prof. Dr. Thomas Zurbuchen**, the former Nasa research director who now is leading ETH Zurich Space. Thomas Zurbuchen explained that vertical farms will be the future of space farming and that an automated vertical farm would be a great project. Especially with facing the high food costs for astronauts on other planets.

Furthermore, we decided to partner with an actual vertical farm to get a better understanding of vertical farming, especially as there are many different sub-systems involved. We wrote to Yasai, a vertical farming startup based in Zurich, and they agreed on giving us a tour of their farm. Yasai showed us all the different subsystems involved in detail and enhanced our understanding on how to make vertical farming as efficient and as user friendly as possible.

Our Vertical Farm now is fully modular, with about 11 different modules involved. It can be assembled in about 30 Minutes. Summarized, our Farm is built with a focus on modularity, automation and cost efficiency, to make it as accessible as possible. To enhance the Automation of our farm we implemented an Artificial Intelligence.

Our Robot Solution

The initial idea was to build a fully automated, modular vertical farm, with a focus on space. After the Regional Competition we decided to implement an Artificial Intelligence to improve our Automation system. We also improved our PCB, and we now have a fully working ESP32 PCB board. Our Farm is mostly automated, with a watering and light system included. If we qualify for the international competition, we will also add our harvesting and seeding mechanism, which currently still is in its prototyping phase.

We believe astronauts, which would be our main target group, but also others could profit a lot from a modular and cheap vertical farm.

About our Idea

Currently, there are many different vertical farms around the world, most of which are automated. For example, Yasai, a start-up from ETH Zurich that we visited. However, there isn't one of them which is modular. These vertical farms often take months to build and come with significant building and designing costs as you must redesign the farm to match with the building you use. This problem does not exist if your farm is fully modular and stackable. In the past there have been efforts to build modular vertical farms, but none of them were automated. This meant we had to fundamentally rethink how vertical farms are designed.

The Framework

System of the main Rods

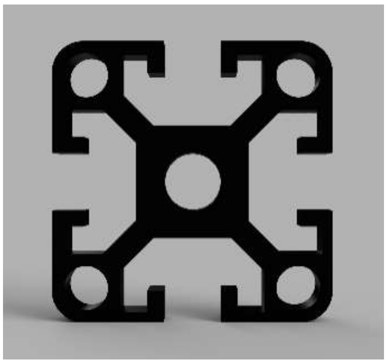
On our way to finding the best framework, we came along the EU standardized Profiles for aluminum. Those profiles are standardized in the industry and are used for different machines and constructions.

We choose the 40 x 40mm profiles but quickly discovered how expensive these profiles are. Furthermore, these profiles don't offer screw holes and other mounting mechanisms we would need for our project.

So, we figured out we needed a customized version of these profiles and custom mounting mechanisms. Another key factor was our budget; these aluminum profiles are way too expensive for our small budget.

We quickly agreed on 3D printing these profiles. We knew this would be very complex, but we saw a new challenge and experience in printing these profiles.

The Prototyping and Design of the Profile



The Side of the Final Profile

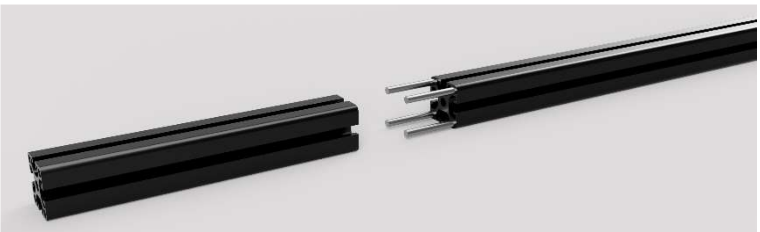
We learnt more about the EU standardized profiles and noticed that these profiles are optimized for aluminum. So, we needed to make a major redesign to achieve good strength and durability on our 3D printed profiles.

First, we made multiple prototypes of different materials and size to consider the strength and ensure an efficient material usage. Additionally, we developed different profiles and gathered more information about the strength of different materials. Another big challenge was the selection of the material. We figured out that we needed about 15kg of 3D printing filament just for the basic framework, however this is still way cheaper than aluminum. So, we compared different materials:

Material Selection

Filament	PLA+	ABS	PETG
Ease of printing	Very easy	Moderate	Difficult
Biodegradable	Yes	No (hazardous)	No
Strength	Medium	High	High
Flexibility	Low	High	Medium

We chose PLA+ as it offers the best printability while offering a sufficient strength for our framework. ABS would have been the best option, but it wasn't available to us for a reasonable price, and we also discovered that it contains hazardous ingredients and additives which could get into the plants. PETG is a bit more expensive and offers worse printability, however it is perfect for absorbing water and resisting fertilizer. So, we decided to use PETG for all the parts exposed to water. As our framework needed 750mm length, but our printer could just print 250mm, we decided to use a compound material system of aluminum and PLA:



Compound system

As seen above 100mm x 6mm aluminum rods get inserted into the 4 x 6mm holes, this system offers good strength. Before the insertion of the aluminum rods, we added a strong two component adhesive to make sure the rods would remain stable.

We now have a system for our 22 needed frame rods, which is optimized for the best price to strength ratio. Overall, these rods needed 734 hours of 3D printing. And about 30 minutes each, to be assembled into their final form.

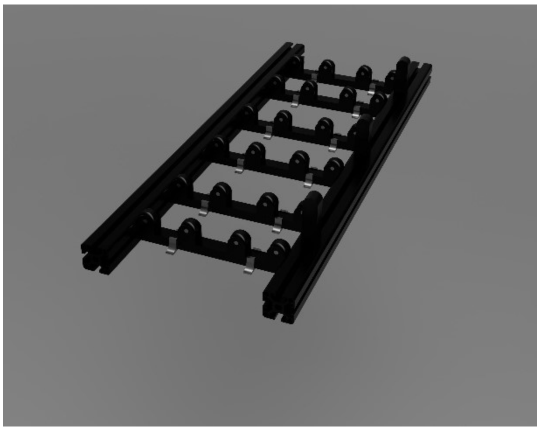
Framework Components

We now began designing the other important structural components needed for the framework, such as the end pieces holding the rods.

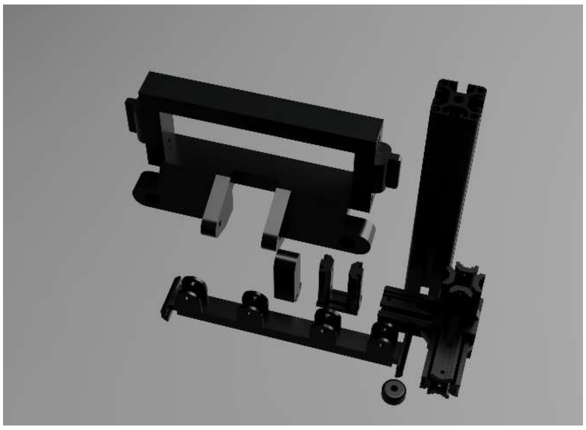
We have designed about 8 different Framework components, which we all had to prototype and develop. We made at least 3 prototypes for each structural component, to optimize the models for mass 3D printing and to ensure a good performance and filament usage.

The screw problem we encountered at the beginning was solved too, as we could use heat inserts which we could melt into the plastic and would then offer a good anchor for the screws.

While printing all the structural components needed, we began designing the main XY system and the different subsystems needed.



An assembled module containing components we developed



Some of our developed components

Main Systems Development

This included:

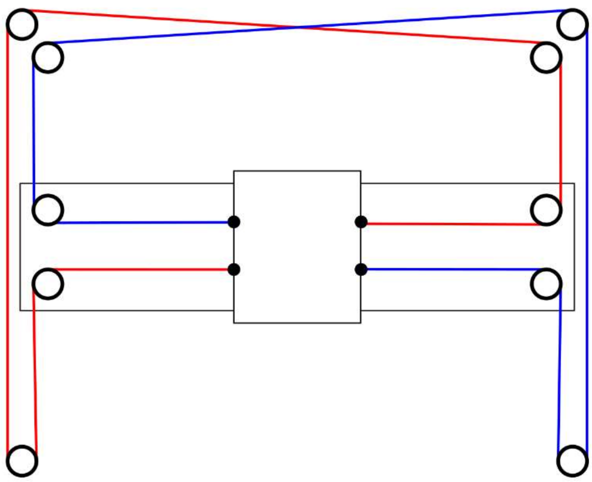
- XY system
  - motor and wheel mount
  - shuttle with pull/push mechanism, x axis
  - lift on y axis
- Water system
- Lighting system
- Harvest and seeding mechanism
- Tray

Water and Light System

The water system was straightforward with just a bracket for the pump and the tubing needed, which was then mounted on the framework. For the lighting system we designed brackets which allowed us to mount the led strips on the rods for the trays, this straightforward method saved us a lot of material as we just needed to print the clip. After this we had to make the lighting system as modular as possible. We decided to use screw terminals which clip the different module wires together. We later decided to apply this system on the whole farm.

The XY System

For the XY axis system we decided to apply the principle of CoreXY as it offers many advantages: Precision, strength, less vibration and fixed motors.



The CoreXY principle

Motor and Wheel Mount

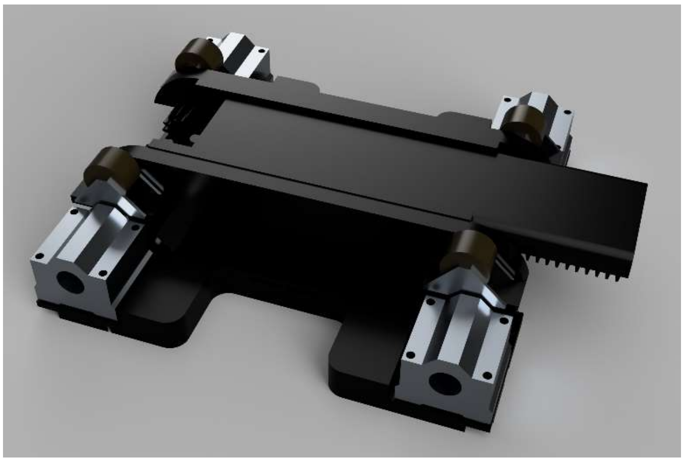
We mounted the motors and the drivers on top of the farm so the motors and its drivers would be safe from any eventual water spills.

While designing the wheel mount we noticed that the farm needed to be higher to reach the bottom level of the farm with the XY-system. Therefore, we quickly redesigned some parts of our framework, ensuring the farm would be slightly above the ground to offer enough space for the wheels. We bought some pulleys which we mounted on an aluminum rod, the pulleys are separated by some spacers. The aluminum rod is embedded into a custom component which is designed to mount the pulleys on the frame.

Shuttle

The shuttle was an exceptional challenge for us, as it needs to fit a variety of systems on a very tight space and as the tolerances are extremely low for the trays to be taken out exactly.

We used linear ball bearings to ensure the shuttle would drive as smoothly as possible on the aluminum rods of the y axis. Initially we used linear actuators for the Shuttle to take out the Trays, however this system had a variety of problems. After the regional championship we completely redesigned our shuttle and are now using a gear-based system. This system is powered by two motors which drive out a slider to a certain point. Then the servos mounted at the end of the slider lock onto the underside of the tray, and it can be taken out. We also designed a system for the slider to not drive too far. The Shuttle also houses the mounting system for the Belt. We used the exact height level of each Belt. This is necessary because the CoreXY mechanism needs different belt heights to function properly. We also outsourced the electrical components to the main housing of the PCB, to use less cable. The major redesign made the shuttle a lot more reliable and overall simpler.



The redesigned Shuttle

Lift on the Y Axis

The lift on the y axis carries the two rods on which the shuttle drives. The lift uses the profile of the framework as a linear guide rail. At the end of the two rods are the custom designed cross beams. They feature 8 small wheels on each side and guide parts which go in the profile of the framework to act as a linear guide rail. The cross beams are designed to hold the aluminum rods of the x axis as low as possible, so there is a quarter “left out” when you look at the cross beams. As this whole component isn’t printable like that, we split it up into different parts. We then applied the same compound method as we already did with the main framework rods, to hold the component together.

The Tray

The Tray features a hole in the Bottom so the Shuttle mechanism can lock onto the tray. In the tray itself is a special net in which the substrate with the plants is embedded. These nets are meant to be taken out by the seeding mechanism. The whole tray is printed in PETG, which ensures that the tray doesn’t leak any water.

Harvesting Mechanism and seeding Mechanism

We haven’t had enough time to fully design these mechanisms yet. We are building them as an external module behind the water station at the bottom of the farm. The tray will drive out of the farm on the module and the net with the substrates will be taken out by a mechanism. The Net will be turned around by linear actuators and the substrates containing the plants will fall out. After this the net will be refilled with substrates and then put back into its tray. As a seeding mechanism we want there to be a nozzle which lets out a defined number of seeds into every hole. This mechanism will also feature a linear axis system.

The Electronics

The farm system is powered by an ESP32 board, chosen for its Wi-Fi capability and sufficient processing power to efficiently run our software. Additionally, the ESP32 offers a straightforward programming process and the option to build a custom PCB.

For AI integration, the ESP32 communicates via UART with a Raspberry Pi 5, which hosts the AI system. The Raspberry Pi 5 processes data and sends values to the ESP32, where they are interpreted for further system operations.

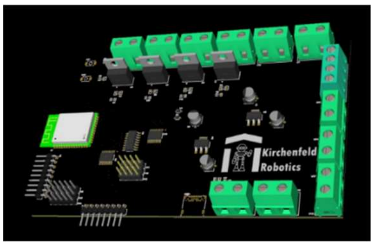
The CoreXY system is powered by two Nema23 stepper motors controlled by two tb6600 motor drivers which are connected to the ESP32. The pump, and the motors of the shuttle are controlled by two L298N driver boards which provide 12V output. The LED system is controlled by our custom PCB, featuring a MOSFET. Most of the electronics are embedded in our custom case to ensure good cooling.

Our custom PCB

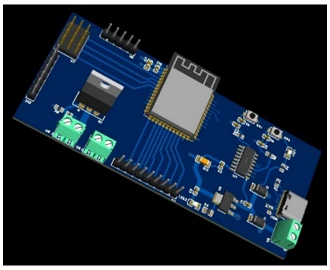
We have successfully designed our custom ESP32 PCB in just two PCB iterations. The PCB is optimized to control the Farm and features these highlights:

- MOSFET for 12V power control (used for lighting system)
- Two independent power sources; 5V and USB-C
- Servo headers, 5V Power and 3.3V PWM
- UART to serial conversion to receive a Serial output and debug more easily (115200 rate)
- Pins and Screw Terminals labelled with corresponding cable

Our Farm is theoretically fully independent and not reliant on a custom PCB. We noticed that it is very time consuming and error prone to connect every cable to an unlabeled not customized PCB. That’s why we decided to design our own PCB.



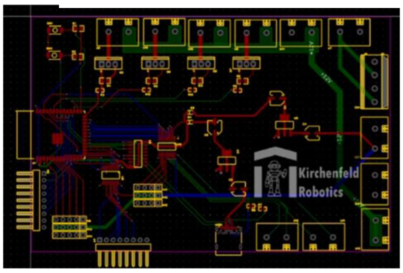
3D Render of the first iteration



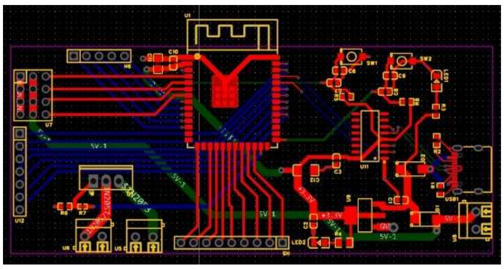
3D Render of the second (working) iteration

Improvement second iteration

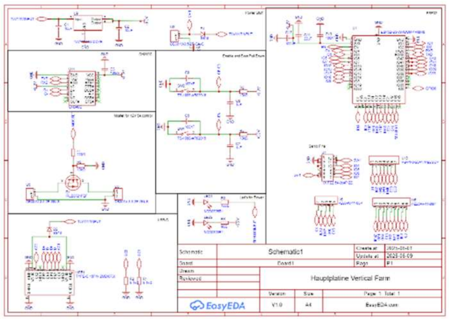
- less space
- less electronical interferences
- Design optimization (cooling, GND plane)



Layout of our first iteration



Layout of our second working iteration



Schematic of the second Iteration

In the first iteration we had a Problem with electrical noise. We fixed this issue with decoupling capacitors and made the whole PCB more efficient by using more Pins and less screw terminals. We also improved cost, by assembling the PCB ourselves.

Software

The Software was an important part of the farm. We decided to develop an application for all platforms. Therefore, we searched for a cross-platform application. We decided on a Software called Flutter, which features cross-compatibility for Android, Ios, Mac, Windows and Web. This software works with the programming language Dart, a language like C#.

The ESP32 runs CPP locally, and acts as an API for our app. For this to work, we added an Integration which allows the communication between CPP and our Flutter application in the JSON format. We had to learn a lot on how to use Flutter and on integrating it into our local running CPP.

After we developed our Flutter program, we began programming the different scripts used for controlling the different subsystems, such as the motors. An example is the XY system for which we made a script containing the location of each box, which is executed when triggered by the Flutter app.

```
void handleBoxmitteMitteNordenUp() {···
}

void handleBoxmitteMitteNordenDown() {···
}

void handleBoxmitteMitteNordenStatus() {···
}
```

The handle function

Calibration System (Homing)

As a calibration system we chose to add four buttons at the end of each axis. If the calibration script is triggered, the motor will drive on each axis until the button is hit. While driving in the other direction, the steps needed to reach the next button are counted. These Steps are then used to calculate the location of the box. The Home location of the Shuttle is at the water station; this means this location is used as an initial point for the XY system. This process of measuring and calculating the location in the XY system is called “Homing”.

Tray Motion

The farm operates across three levels, simplifying tray location calculations. By dividing the homing system's measured value by two, we can accurately determine each tray’s position.

Once the shuttle reaches the correct location, the slider extends by setting the L298N to the appropriate HIGH and LOW values. The L298N then supplies 12V to the shuttle's two motors, which engage the gear and push the slider outward.

At full extension, the servo at the slider's end rotates 90 degrees, locking onto the tray. The slider then retracts, pulling the tray onto the shuttle. Since the slider only reaches halfway across the tray, it must travel to the opposite side to fully load it on the shuttle. The servo then rotates back down, and the Slider retracts into the shuttle. The Shuttle is then ready to move to its next location with the Tray loaded on it. Upon arrival, the same process repeats in reverse to unload the tray.

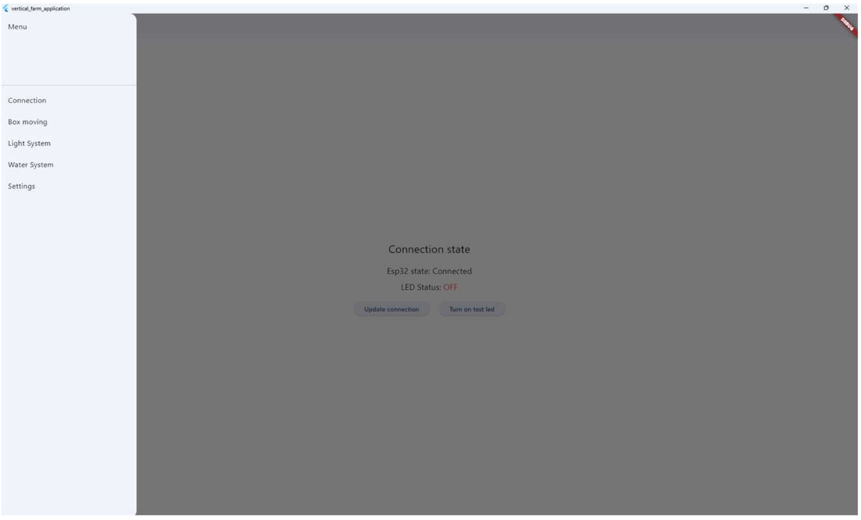
Diagram of the code

Flutter explained

Flutter is open-source UI software development kit created by Google. It can be used to develop cross platform software from a single code editor. In this environment it is possible to develop apps for Android, iOS, macOS, Windows and Web. But in the beginning, we had a focus on the Windows app as it needed to be working first. But as flutter is a cross-platform development tool, we developed the software to work for Android and iOS or other devices too.

The Application

Our windows application features a simple interface with a Navigation Bar at the left, which we have customized while still using the standard background. We have designed the Pages on the right side completely on our own. In the process of programming our own app the official flutter documentation helped us a lot, as it features a good documentation of all the steps.



Flutter Interface

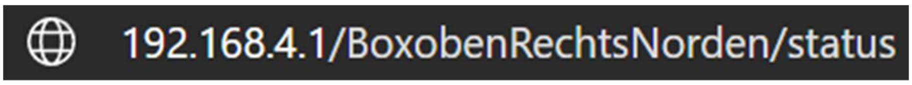
For the Main Pages, which we designed on our own, we had to learn a lot about Flutter. We dove deep into the software to make a good looking and functional interface.

We faced some major issues with the connection of the Flutter interface to the backend, as there are very few documentations on this topic.

Connection of the end Device to the ESP32

We knew that we wanted to include a wireless option, as the ESP32 features a Wi-Fi module. However, we weren't sure if there would be Wi-Fi access at the robotic competitions, so we decided to create our own Access Point with the ESP32 itself. This way we aren't dependent on a Wi-Fi access.

Once the end device is connected to the AP of the ESP32, it is steerable by two methods. The first method is just by adding the IP address of the ESP32 in the browser, together with the script that should run. There is a direct connection to the API and no Interface in between.



However, if you want to use our custom designed flutter app you can just connect to the AP and open the app on your end device. The app is available for IOS, Android, Windows and Mac, to ensure maximum flexibility. The Flutter interface will then connect to the ESP32. As mentioned before, the Flutter app communicates with the ESP32 over the JSON format.

```dart
Future<void> _toggleled() async { // Turn on off the led System
  setState(() {
    _isLoading = true; // change state on true -> spinning wheel
  });
  try {
    final endpoint = _ledsState ? '/led/off' : '/led/on'; // roots enpoint of the webserver
    final uri = Uri.parse('http://$_ipAddress$endpoint');
    log("Sending request to $uri"); // Debugging
    final response = await http.get(uri).timeout(const Duration(seconds: 5)); // Timeout after not connection after 5 seconds
    log("Toggle response status: ${response.statusCode}"); // The Webserver Text

    if (response.statusCode == 200) { // Request successfully
      await _fetchledStatus(); // Request led state
    } else { // no connection
      setState(() { // Failed
        _response = 'Failed to toggle led: ${response.statusCode}';
        _isLoading = false;
      });
    }
  } catch (e) { // if sth else happened led
    log("Error toggling led", error: e, name: "error");
    setState(() {
      _response = 'Connection error: $e';
      _isLoading = false;
    });
  }
}
```

Example of JSON communication

In this example we see how the Flutter app connects to the ESP32, after 5 seconds it gives a timeout error. This code is specifically used for a Test Led to test the connection.

Below we can see how the app asks for the status of the boxes; this way we can see the status of the box in our interface. The Value of the Status is communicated in JSON.

AI Plant Detection

For our AI plant detection, we considered different AI models. Our criteria were quite strict as the model needed to be optimized for plants, be free and run efficient enough so we could power it by a raspberry Pi 5. Therefore, we decided on using PlantCV. This model is an open-source project by the Donald Danforth Plant Science Centre based on OpenCV, an open computer vision library for Real-time image processing.

Setup

The PlantCV Model runs locally on a raspberry Pi 5 with 8 GB of ram for an optimal performance. The Raspberry Pi communicates with the main Esp32 PCB via serial connection. The Raspberry Pi sends the outputs of the PlantCV Model over the serial connection to the ESP32. The Esp32 then implements these information’s into the existing software.

About PlantCV

Plant CV offers different parameters, however as our time was limited, we decided to just use the size analysis. This parameter analyses the plants size and passes the parameters to the esp32. Later, we want to implement more parameters to enhance the AI system.

These pictures show how a plant is analysed. First the green parts get marked to enhance the visibility. Then a net is then laid around the green parts. The net gets a size estimation based on the plants shape and characteristics.

```
from plantcv import plantcv as pcv

# Set global debug behavior to None (default), "print" (to file),
# or "plot" (Jupyter Notebooks or X11)

pcv.params.debug = "plot"
# Optionally, set a sample label name
pcv.params.sample_label = "plant"

# Characterize object shapes
shape_image = pcv.analyze.size(img=img, labeled_mask=mask, n_labels=1)

# Save returned images with more specific naming
pcv.print_image(shape_image, '/home/malia/setaria_shape_img.png')

# Access data stored out from analyze.size
plant_solidity = pcv.outputs.observations['plant_1']['solidity']['value']
```

The corresponding code from the documentation

Faced Challenges

With our software we faced major challenges. The main challenge was our limited knowledge. Until this point, we haven't done any application development and weren't experienced with the ESP32 chip. Furthermore, we have had no experience with Flutter or the Front or Back-end usage. Additionally, it isn't possible to write such programs with ChatGPT. The code is way too complex because it features an API and JSON communication. Furthermore, ChatGPT doesn't allow for any visual view of the generated code. So, we had to do most of the things on our own. All our knowledge gaps had to be filled by learning, so the software part of this project was perhaps where we have learned the most. This was very time intensive, so we had to invest way more time than initially planned. As a part of this project, we both also took a 10-hour CPP course (Codecademy) to enhance our knowledge of programming the ESP32 and the API.

Social Impact and Innovation

The strengths of our Farm

The invention of our farm could have a huge impact on a variety of communities.

Our Farm is fully modular and assembled in about 30 minutes. It is also very cheap with just <500fr. cost for the whole system. Important too is, that the whole system is scalable. The Farm presented consists of a cube, theoretically you could just stack these cubes vertically and increase the food production. If you already stacked the farm vertically you can also stack it horizontally, however then you would need to increase the length of the aluminum rods for the x axis.

Because of these factors our farm is very appliable for many different groups or communities. This robot will help to fight hunger crisis with cheap food provided quickly and efficiently. Furthermore, it will offer for more flexible food production, for example in cities as mentioned below. Our farm is optimized for other planets but is also very appliable on earth. Until now there have been no efforts on making a fully automated vertical farm with a focus on modularity so it's transportable. We believe this idea has great potential and can be developed further.

Our farm is just a prototype, which means that a real version of this farm would at least double the size to allow for a higher food production. However, our farm proves which features make sense in a vertical farm (custom PCB, AI detection and growth optimization).

Vertical Farming's Impact yet

Vertical Farming has significant benefits:

- Year-round production
- 95–98% less water:
- Farming vertically is 200–400% more efficient.
- no pesticides
- 10 times more plants on the same area if you farm vertically instead of horizontally.

It also has a big impact already:

- In Singapore, vertical farming is part of the 30-by-30 strategy, aiming to produce 30% of the country's nutritional needs locally by 2030.
- Switzerland – *Agroscope* is running a pilot vertical farming project in partnership with YASAI, Fenaco, and ZHAW to study plant production and sustainability.

- Controlled environment agriculture reduces the need for pesticides and antibiotics, minimizing agricultural runoff and pollution.

Vertical farming already is a big business, with the vertical farm market valued at about 6.92 billion USD.

Our Vertical Farm already has those advantages compared to conventional farming, but offers some more which make it advanced:

- Modularity
- Cost efficiency
- Scalability
- User friendliness with custom application

Example 1 – Space

Jakob had the chance to talk to Thomas Zurbuchen, the former Nasa Director of research. With him he talked about vertical farms in space. We concluded that our thoughts that vertical farms in space would be a perfect fit, were right.

Today food for astronauts needs to be transported with separate flights to the space station or the other planet for the astronauts to survive there. These flights are very expensive and inefficient. So, there is a need for farming in space. There have already been experiments on the ISS with vertical farming. However, as astronauts are very busy, it is necessary that they don't need to do all the farming by themselves. So, our farm is perfect as it is fully modular and fully? (you have already said that). On another planet the key component before all the other considerations is the transportability. Our farm is lightweight, fully modular, and easy to assemble. Therefore, our farm is perfect for supporting human life on mars. Once assembled, it provides food to the astronauts completely autonomous. There is no need for expensive and inefficient food transports anymore. Our farm is the perfect solution for this problem, and is designed around it, with a weight of about 20kg and an easy and quick assembly. The astronaut would just need to separate the crops from the substrate.

Example 2 - Cities

Today, vertical farming is mainly popular in cities, as it allows to grow plants directly in city and therefore reduce the transportation ways and allow for much fresher food. A new opportunity in this field would be the temporary use of buildings. Today, there often are business buildings or areas which can’t be used, as the contract is time limited. These temporary usages often need a short timeframe, so the buildings often remain unused for some months. A modular vertical farm would be the perfect fit for this problem as it allows to be mounted quickly and can be reused at another location and the advantages of vertical farming in cities would persist. Furthermore, in many buildings in cities it isn’t allowed to make major changes to the interior, what is necessary for traditional vertical farming. There our vertical farm would be a perfect fit as well, as it doesn’t use any fixed mounting mechanisms at the building itself and works completely autonomous.

Examples for vertical farms in cities are already found all around the world:

- Growing Underground – London, UK
- Pink Farms – São Paulo, Brazil
- SpaceFarms – Tbilisi, Georgia
- Vertical Agri – Svalbard, Norway
- Can-Agri – Pretoria, South Africa
- Plant Factory – Istanbul, Turkey

but with our farm there could be even more!

Our project is an important step towards modular, fully automated vertical farms and to make vertical farms more accessible by making them cheaper. We are excited where this project will lead us and how it will contribute to better vertical farming.

Business & Finance Overview

Our current Finances are listed in this table:

Product	Price	Comment
PCB iteration 1	CHF 240	First iteration
PCB iteration 2	CHF 80	Second generation
Bauhaus mixed items	CHF 160	Screws, cables and other small items
T-Shirts and branding	CHF 65	T-Shirts and Posters
Diver’s electronic components	CHF 290	Esp32, servo, L298N
Filament 20 kg PLA and 10 kg PETG	CHF 370	Jayo filament and taxes
Conrad stepper motor	CHF 60	For the XY Core System
Equipment	CHF 80	Tools which we specially had to buy
Others	CHF 60	Things like solder paste, replacement parts for 3D printer
Total	CHF 1405	

We have invested significant time and money in this project. Fortunately, having access to tools, including a 3D printer, has been invaluable in advancing our work. However, our financial situation remains challenging.

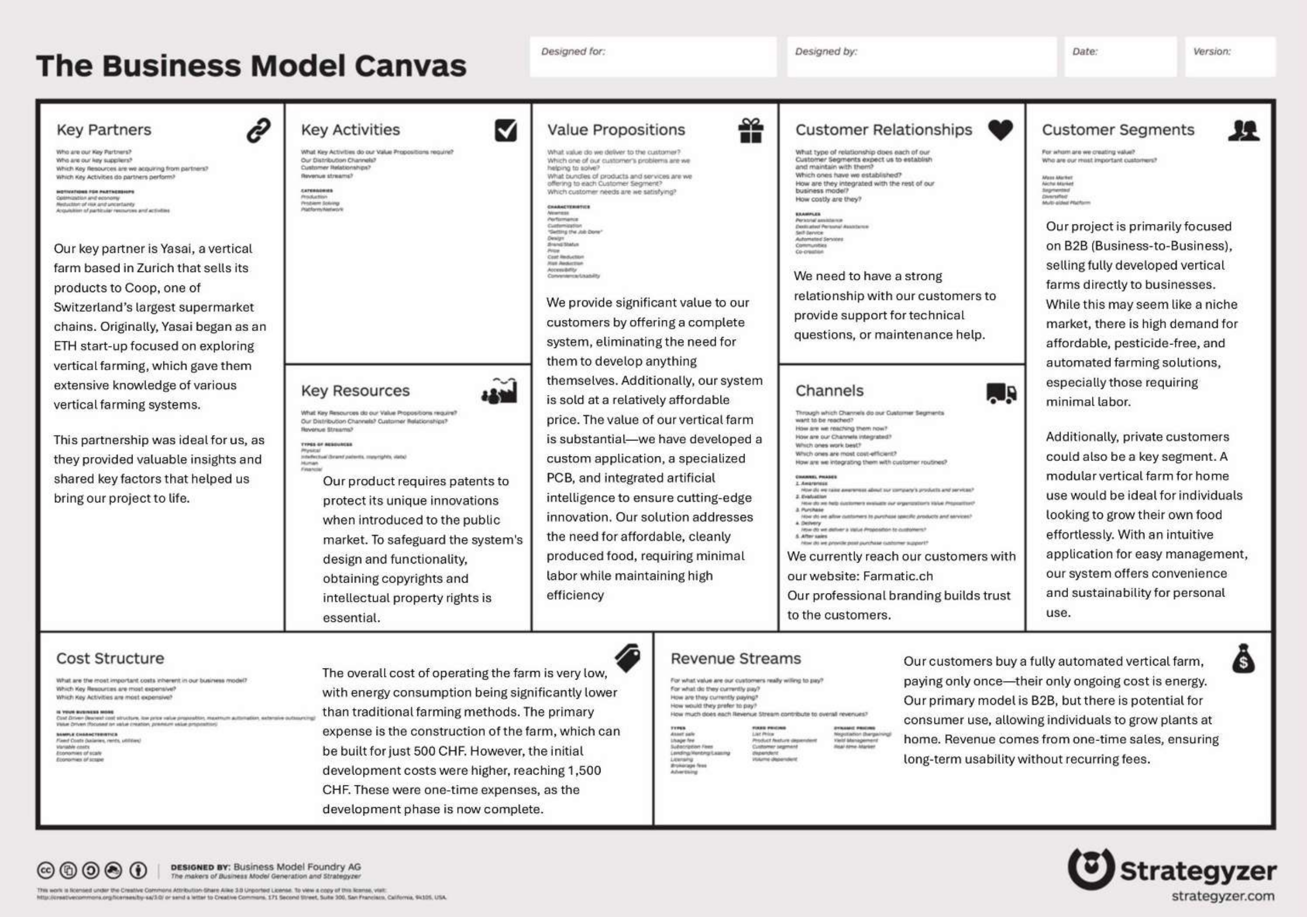
Since 2024, Jakob has worked as a newspaper deliverer, contributing funds to the project. Valéry has received financial support primarily from his parents. To sustain further development, we have sought donations, as we first need to repay our existing expenses before acquiring new materials to enhance our vertical farm.

Currently, we have no financial partners, but we are collaborating with Yasai, a vertical farm in Switzerland. They have generously shared their innovations and provided valuable advice. However, as a start-up, they are not in a financial position to offer funding.

We had hoped for financial support from our school, but unfortunately, we have not received any funding from them, mainly because we hadn’t had the time to apply for funds.

We invested in our branding and image to enhance our chances of securing sponsorship, especially if we qualify for the international final.

In our Business Canva we filled out the parts matching our project and topic:



The Business Model Canvas

Designed for: | Designed by: | Date: | Version:

Key Partners

Our key partner is Yasai, a vertical farm based in Zurich that sells its products to Coop, one of Switzerland's largest supermarket chains. Originally, Yasai began as an ETH start-up focused on exploring vertical farming, which gave them extensive knowledge of various vertical farming systems.

This partnership was ideal for us, as they provided valuable insights and shared key factors that helped us bring our project to life.

Key Activities

Key Resources

Our product requires patents to protect its unique innovations when introduced to the public market. To safeguard the system's design and functionality, obtaining copyrights and intellectual property rights is essential.

Value Propositions

We provide significant value to our customers by offering a complete system, eliminating the need for them to develop anything themselves. Additionally, our system is sold at a relatively affordable price. The value of our vertical farm is substantial—we have developed a custom application, a specialized PCB, and integrated artificial intelligence to ensure cutting-edge innovation. Our solution addresses the need for affordable, cleanly produced food, requiring minimal labor while maintaining high efficiency

Customer Relationships

We need to have a strong relationship with our customers to provide support for technical questions, or maintenance help.

Channels

We currently reach our customers with our website: Farmatic.ch

Our professional branding builds trust to the customers.

Customer Segments

Our project is primarily focused on B2B (Business-to-Business), selling fully developed vertical farms directly to businesses. While this may seem like a niche market, there is high demand for affordable, pesticide-free, and automated farming solutions, especially those requiring minimal labor.

Additionally, private customers could also be a key segment. A modular vertical farm for home use would be ideal for individuals looking to grow their own food effortlessly. With an intuitive application for easy management, our system offers convenience and sustainability for personal use.

Cost Structure

The overall cost of operating the farm is very low, with energy consumption being significantly lower than traditional farming methods. The primary expense is the construction of the farm, which can be built for just 500 CHF. However, the initial development costs were higher, reaching 1,500 CHF. These were one-time expenses, as the development phase is now complete.

Revenue Streams

Our customers buy a fully automated vertical farm, paying only once—their only ongoing cost is energy. Our primary model is B2B, but there is potential for consumer use, allowing individuals to grow plants at home. Revenue comes from one-time sales, ensuring long-term usability without recurring fees.

DESIGNED BY: Business Model Foundry AG
The makers of Business Model Generation and Strategyzer

Strategyzer
strategyzer.com

Sources

Project idea

Modular Vertical Farming

GreenState AG - Modular Farming Solutions

Frontiers - Business Models for Modular Urban-Vertical Farming

Association for Vertical Farming - Drop-N-Gro Modular Farm

Space Farming

Space Agriculture: Growing Food for Mars and Beyond

Wageningen University - The Future of Space Farming

NASA - Space Farming Benefits for Earth

Mars Habitats & UV Protection

Marspedia - Radiation Shielding on Mars

NASA - Protecting Astronauts from Space Radiation

SciTechDaily - Surviving Mars: Radiation Protection

Yasai Vertical Farming Startup

Yasai - Scaling Up Vertical Farming

Startup.ch - Yasai AG Overview

Swiss Food & Nutrition Valley - Yasai's Growth

Thomas Zurbuchen

Thomas Zurbuchen - NASA Science

Thomas Zurbuchen

Thomas Zurbuchen – Wikipedia

Prof. Dr. Thomas Zurbuchen | ETH Zürich

Our robotic solution

Modular Vertical Farming

GreenState AG - Modular Farming Overview

Frontiers - Business Models for Modular Urban-Vertical Farming

Market Research Intellect - Modular Farming & Vertical Agriculture

CoreXY System

Printing Atoms - Best CoreXY 3D Printers

3DSourced - CoreXY 3D Printer Guide

GitHub - Open Source CoreXY Configurations

ESP32 Applications

LibHunt - ESP32 Open-Source Projects

Random Nerd Tutorials - ESP32 Guides

How2Electronics - ESP32 IoT Projects

3D Printing Materials Comparison

Simplify3D - Ultimate 3D Printing Materials Guide

3Dnatives - Compare 3D Printing Materials

Eufymake - 3D Printing Materials Guide

Automated Farming Systems

McKinsey - Trends Driving Farm Automation

Digi International - IoT in Agriculture

Springer - Robotics & Autonomous Systems for Agriculture

Modular Vertical Farming

GreenState AG - Modular Farming Solutions

Frontiers - Business Models for Modular Urban-Vertical Farming

Association for Vertical Farming - Drop-N-Gro Modular Farm

CoreXY System

RepRap - CoreXY Mechanism Overview

CoreXY - Cartesian Motion Platform

Wikipedia - CoreXY System

ESP32 Applications

LibHunt - ESP32 Open-Source Projects

Random Nerd Tutorials - ESP32 Guides

How2Electronics - ESP32 IoT Projects

PCB Design

KiCad EDA - Open-Source PCB Design

3DSourced - Top PCB Design Software

CNCSourced - Free PCB Design Tools

ESP32 Applications

LibHunt - ESP32 Open-Source Projects

Random Nerd Tutorials - ESP32 Guides

How2Electronics - ESP32 IoT Projects

Flutter Software

Flutter Official Documentation

Building a Web Server with ESP32 and Fetching/Sending Data in Flutter | by Jahidul Islam | Medium

Wilson Wilson - Ultimate Flutter Resources

DEV Community - Useful Flutter Websites

SoluteLabs - Top Flutter GitHub Repositories

JSON Communication

Dadroit - JSON Datasets

SQLPey - Public JSON APIs

GitHub - Awesome JSON Libraries

Motor Driver Systems

Components101 - L298N Motor Driver Module

Arduino Project Hub - Using L298N Motor Driver

Last Minute Engineers - L298N Motor Driver Tutorial

Social Impact and Innovation

General Benefits of Vertical Farming

Profolus - Vertical Farming Advantages

Raiz Vertical Farms - Benefits & Advantages

GreenState AG - The Benefits of Vertical Farming

World Economic Forum - Vertical Farming Overview

Economic Impact

Springer - Economics of Vertical Farming

IntechOpen - Vertical Farming Innovations

Agroscope - Vertical Farming Research

Water Efficiency

World Economic Forum - Water Efficiency in Vertical Farming

Petr Kirpeit - Water Consumption in Vertical Farming

Wikifarmer - Sustainability and Environmental Impact

Urban Vertical Farming Examples

Sustainable Urban Delta - 9 Vertical Farms in Cities

MDPI - Urban Vertical Farming

Green City Times - Vertical Farming in Urban Spaces

Space Applications

World Economic Forum - Space AgTech

NASA - Contributions to Vertical Farming

NASA Spinoff - Indoor Farming