

Imię i nazwisko studenta: Jakub Świłocki  
Nr albumu: 175846  
Poziom kształcenia: Studia pierwszego stopnia  
Forma studiów: stacjonarne  
Kierunek studiów: Informatyka  
Profil: Inteligentne systemy interaktywne

Imię i nazwisko studenta: Paulina Lis  
Nr albumu: 113426  
Poziom kształcenia: Studia pierwszego stopnia  
Forma studiów: stacjonarne  
Kierunek studiów: Informatyka  
Profil: Teleinformatyka

Imię i nazwisko studenta: Jakub Zwolski  
Nr albumu: 175967  
Poziom kształcenia: Studia pierwszego stopnia  
Forma studiów: stacjonarne  
Kierunek studiów: Informatyka  
Profil: Algorytmy i modelowanie systemów

Imię i nazwisko studenta: Maciej Łotysz  
Nr albumu: 175525  
Poziom kształcenia: Studia pierwszego stopnia  
Forma studiów: stacjonarne  
Kierunek studiów: Informatyka  
Profil: Teleinformatyka

## PROJEKT DYPLOMOWY INŻYNIERSKI

Tytuł projektu w języku polskim: Opracowanie prototypowej platformy i urządzeń do zautomatyzowanej uprawy hydroponicznej

Tytuł projektu w języku angielskim: Development of a prototype platform and equipment for automated hydroponic cultivation

Opiekun pracy: dr inż. Andrzej Sobecki

Data ostatecznego zatwierdzenia raportu podobieństw w JSA: 13.12.2021

## **Streszczenie**

Głównym celem projektu inżynierskiego jest zapoznanie z dostępnymi na rynku rozwiązaniami dającymi możliwość prowadzenia upraw roślin metodą hydroponiczną w domowych warunkach oraz opracowaniu prototypu wraz z oprogramowaniem do zautomatyzowania tego rodzaju hodowli. Zakres pracy dotyczy zarówno budowy fizycznego, działającego prototypu, jak i opracowaniu oprogramowania zarządzającego całym procesem hodowli i monitorującym postępy w rozwoju roślin. Najważniejszym celem tego projektu jest opracowanie takiego prototypu, który zapewni jak największą automatyzację hodowli i jej niezależność od warunków zewnętrznych. Po zbudowaniu w pełni funkcjonalnego prototypu i wdrożeniu niezbędnego, dla jego działania, oprogramowania prototypowa platforma została poddana testom sprawdzającym poprawność jej działania. Eksperymenty polegały na sprawdzeniu szybkości działania urządzenia w przypadkach, gdy monitorowane przez prototyp czynniki, odbiegały od określonej w oprogramowaniu normy. Wyniki testów pokazują poprawne działanie prototypowej platformy, ale można z nich wyciągnąć wnioski, co można poddać usprawnieniom. Przykładem systemu do poprawienia jest układ chłodzenia. Projekt ten realizowany jest w grupie czteroosobowej, a podział pracy wygląda następująco: Jakub Świślocki jest odpowiedzialny za realizację rozdziałów 1,5,8, Jakub Zwolski jest odpowiedzialny za realizację rozdziałów 2,3,4, Paulina Lis jest odpowiedzialna za realizację rozdziału 6, Maciej Łotysz jest odpowiedzialny za realizację rozdziału 7.

**Słowa kluczowe:** hydroponika, pompa perystaltyczna, uprawa, truskawki, roślina, pompa strzykawkowa

**Dziedzina nauki i techniki, zgodnie z wymogami OECD:** Elektrotechnika, elektronika, inżynieria informatyczna

## **Abstract**

The main goal of the engineering project is to familiarize with the solutions available on the market that enable the cultivation of plants with the hydroponic method at home and the development of a prototype with software for automating this type of breeding. The scope of work concerns both the construction of a physical, working prototype and the development of software that manages the entire breeding process and monitors the progress in plant development. The most important goal of this project is to develop a prototype that will ensure the greatest automation of breeding and its independence from external conditions. After building a fully functional prototype and implementing the software necessary for its operation, the prototype platform was tested to verify its correct operation. The experiments consisted in checking the speed of operation of the device in cases where the factors monitored by the prototype deviated from the standard specified in the software. The test results show the correct operation of the prototype platform, but conclusions can be drawn from them, which can be improved. An example of a system to be improved is the cooling system. This project is carried out in a group of four, and the division of labor is as follows: Jakub Świsłocki is responsible for the implementation of chapters 1,5,8, Jakub Zwolski is responsible for the implementation of chapters 2,3,4, Paulina Lis is responsible for the implementation of chapter 6, Maciej Łotysz is responsible for the implementation of Chapter 7.

**Keywords:** hydroponics, peristaltic pump, cultivation, strawberries, plant, syringe pump

## **SPIS TREŚCI**

1. WPROWADZENIE .....	7
2. PRZEGLĄD ISTNIEJĄCYCH ROZWIĄZAŃ .....	8
2.1. General Hydroponics .....	8
2.2. Nutriculture .....	9
2.3. Vakplast .....	10
2.4. Green Farm .....	10
2.5. Freight Farms .....	11
2.6. Porównanie gotowych rozwiązań .....	12
2.7. Podsumowanie .....	13
3. ANALIZA POTRZEB I WYMAGAŃ ŚRODOWISKOWYCH .....	14
3.1. Opis wymagań .....	14
3.2. Parametry środowiskowe w hodowli truskawek .....	14
4. PROJEKT MODELU STERUJĄCEGO HODOWLĄ .....	16
4.1. Diagram stanów .....	16
4.1.1. Przygotowanie .....	17
4.1.2. System zarządzający .....	17
4.1.3. Zarządzanie temperaturą .....	18
4.1.4. Zarządzanie pH .....	19
4.1.5. Zarządzanie kamerą .....	20
4.1.6. Zarządzanie światłem .....	21
4.1.7. Zarządzanie wilgotnością .....	22
4.1.8. Logowanie .....	23
4.1.9. Zarządzanie nawozem .....	23
5. PROJEKT URZĄDZENIA .....	25
5.1. Konstrukcja główna .....	25
5.2. Kontrola środowiska powietrznego .....	30
5.2.1. Kontrola temperatury i wilgotności .....	30
5.2.2. Pomiar i sterowanie temperaturą .....	31
5.2.3. Pomiar i sterowanie wilgotnością .....	35
5.3. Kontrola Środowiska wodnego .....	40
5.3.1. Kontrola poziomu pH .....	40
5.3.2. Kontrola poziomu soli mineralnych .....	44
5.4. Kontrola oświetlenia .....	51
5.4.1. Kolor urządzenia .....	53
5.5. Pozostałe elementy budowy .....	57
5.5.1. Kamera .....	57
5.5.2. Napowietrzanie wody .....	58
5.5.3. Zasilanie .....	59
5.6. Elektronika .....	61
5.7. Końcowa prezentacja urządzenia .....	65

6. PREZENTACJA WARSTWY PROGRAMOWEJ .....	69
6.1. PIN-y GPIO .....	69
6.2. Opis użytych bibliotek .....	70
6.3. Skrypty dla poszczególnych komponentów.....	70
6.4. Serwer Raspberry Pi .....	71
6.5. Połączenie z Raspberry Pi.....	74
6.6. Oprogramowanie Arduino.....	75
6.6.1. Biblioteka “Wire.h” .....	75
6.6.2. Opis programu.....	75
6.7. Program nadzorujący .....	77
6.7.1. Elementy instalacji Apache Airflow .....	77
6.7.2. Diagramy DAG.....	78
7. TESTY URZĄDZENIA .....	81
7.1. Opis środowiska .....	81
7.2. Plan eksperymentów .....	81
7.3. Opis pierwszego eksperymentu.....	82
7.4. Opis drugiego eksperymentu .....	83
7.5. Opis trzeciego eksperymentu .....	84
7.6. Opis czwartego eksperymentu .....	85
7.7. Podsumowanie wyników .....	86
7.8. Prezentacja eksperymentalnej hodowli.....	86
8. PODSUMOWANIE .....	89

## **1. WPROWADZENIE**

Autor rozdziału: Jakub Świślocki

Hodowla własnych warzyw czy owoców nie jest prosta. Wymaga sporo czasu i wynikają z niej pewne komplikacje. Hodowla roślin w warunkach domowych (np. parapet w bloku) niesie za sobą problemy takie jak:

1. Konieczność pamiętania o podlewaniu roślin.
2. Umieszczenie ich w nasłonecznionym miejscu.
3. Istnieje ryzyko pojawiения się szkodliwej pleśni na ziemi doniczkowej.

Większość z powyższych problemów nie dotyczy upraw ogrodowych. Niestety nie każdy może pozwolić sobie na posiadanie ogrodu. Ponadto takie rozwiązanie wprowadza nowe utrudnienia takie jak brak możliwości prowadzenia upraw w okresie zimowym. Poza tym w obu powyższych przypadkach uprawom trzeba poświęcać spore ilości czasu, a z powodu braku możliwości regulacji warunków atmosferycznych i składu gleby, efekt takiej pracy może być niezadowalający.

Z powodów wyżej wymienionych zrodziła się idea ułatwienia domowej hodowli małych roślin za pomocą urządzenia, które w pełni zautomatyzuje ten proces. Główne założenia dla urządzenia to niewielkie rozmiary oraz możliwość pracy w różnych warunkach zewnętrznych, z wyjątkiem temperatury, która powinna być bliska temperaturze pokojowej. Jako roślina do eksperymentalnej hodowli wybrana została truskawka z kilku następujących powodów:

1. Jest to dość popularna roślina w uprawach domowych.
2. Nie jest tak łatwa w uprawie jak np.: sałata bądź zioła.
3. Łatwo można wizualnie określić efekt upraw.

W dalszej części tej pracy omówiono cały proces powstawania urządzenia. W rozdziale drugim przedstawiono istniejące rozwiązania i zdefiniowano problem. W kolejnym rozdziale skupiono się na projekcie rozwiązania i przedstawiono szczegółowe założenia oraz wymagania. W rozdziale czwartym opisano założenia dotyczące oprogramowania. W następnych dwóch rozdziałach przedstawiono kolejno konstrukcję urządzenia oraz implementację oprogramowania. W przedostatnim rozdziale opisano badania i eksperymenty przeprowadzona na urządzeniu. Ostatni rozdział stanowi podsumowanie pracy.

## **2. PRZEGŁĄD ISTNIEJĄCYCH ROZWIĄZAŃ**

Autor rozdziału: Jakub Zwolski

Na rynku istnieje już sporo rozwiązań, które umożliwiają uprawę hydroponiczną zwykłemu użytkownikowi. W dalszej części przedstawione zostało kilka z nich. Przy każdym z urządzeń brany pod uwagę był aspekt automatyzacji rozwiązań tak, aby usprawnić oraz polepszyć ich działanie.

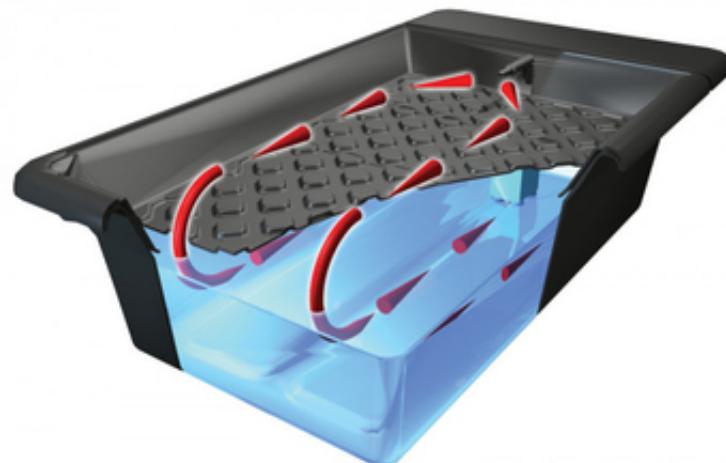
### **2.1. General Hydroponics**



Rys. 2.1. System hydroponiczny GHE Aqua Farm (zdjęcie pochodzi ze strony <https://www.growbox.pl/>)

Firma oferuje kompletny system hydroponiczny wyposażony w pompę napowietrzającą z możliwością zasadzenia od dwóch do sześciu średniej wielkości roślin. Rośliny umieszczone są w donicy z granulatem ceramicznym i włożone do specjalnego pojemnika z wodą.

## 2.2. Nutriculture



Rys. 2.2. NFG Gro-Tanks (zdjęcie pochodzi ze strony nutriculture.co.uk)

Kolejny przykład gotowego rozwiązania to zbiornik z wodą od firmy Nutriculture, który poprzez strumień wody stale dostarcza roślinom napowietrzoną wodę z odżywką. sectionAutoPot



Rys. 2.3. 4Pot System (zdjęcie pochodzi ze strony autopot.co.uk)

Hydroponiczny system opierający się o jeden główny zbiornik, który dostarcza wodę pod do donic z roślinami. Każdy zestaw wyposażony jest także w AQUAvalve. Jest to system pozwalający na odpowiednie dozowanie wody, uzupełnia ją dopiero gdy cała woda w donicy zostanie wykorzystana przez roślinę.

### **2.3. *Vakplast***



Rys. 2.4. Pionowy system hydroponiczny firmy Vakplast (zdjęcie pochodzi ze strony [vakplast.com](http://vakplast.com))

Czeski producent Vakplast w swojej ofercie posiada pionowe systemy wyposażone w półki, w których można umieścić rośliny. Posiada kółka, które pozwalają na łatwe przemieszczanie, jednak przy samej hodowli producent zaleca przykuczenie urządzenia do ściany. Nawadnianie polega na ciągłym przepuszczaniu wody z odżywkami przez rury.

### **2.4. *Green Farm***



Rys. 2.5. Mini szklarnia firmy Green Farm (zdjęcie pochodzi ze strony [green-farm.pl](http://green-farm.pl))

Mini szklarnia to rozwiązanie, które oprócz nawadniania dodatkowo dba o odpowiednie natężenie światła w środku urządzenia. Produkt jak możemy wyczytać ze strony producenta przeznaczony jest głównie do uprawy warzyw liściowych takich jak na przykład sałata masłowa czy szpinak.

## 2.5. *Freight Farms*



Rys. 2.6. Kontener firmy Freight Farms (zdjęcie pochodzi ze strony [freightfarms.com](http://freightfarms.com))

Firma z Bostonu, która jako pierwsza zaoferowała sprzedaż oraz produkcje farm kontenerowych. Farma kontenerowa jest w pełni zautomatyzowana. Składa się z wielopoziomowych półek umieszczonych na bocznych ścianach kontenera. W ofercie dodatkowo zawarte są szkolenia. Można również dowiedzieć się o dedykowanej aplikacji do monitorowania uprawy i informowania producenta o przykładowych niekorzystnych warunkach. Cena za subskrypcję roczną kosztuje w okolicach 5000 zł, lecz można także korzystać z ograniczonej wersji kosztującej około 40 zł za miesiąc. Plusem takiego rozwiązania jest możliwość postawienia kontenera w dowolnym miejscu przy dowolnej zewnętrznej temperaturze.

## 2.6. Porównanie gotowych rozwiązań

Producent	Rozwiązańe	Światło	Sposób wadziania i napowietrzania	Kontrola nawozu i regulacja pH	Regulacja wilgotności i temperatury	Plusy	Minusy
General Hydroponics	Donica ze zbiornikiem	Samodzielny zakup, montaż i kontrola	Pompa napowietrzająca w zestawie nawadnia rośliny kierując w nich stronę stale strumień wody	Samodzielnego monitorowanie i regulacja	Brak	Łatwy do przeniesienia	Brak światła, brak automatycznej dozowania składników odżywcznych, regulacji pH, temperatury i wilgotności
Nutriculture	Zbiornik z wodą	Samodzielny zakup, montaż i kontrola	Pompa napowietrzająca, stale dostarczanie napowietrzonej wody do korzeni rośliny	Samodzielnego monitorowanie i regulacja	Brak	Łatwy do przeniesienia	Brak światła, brak automatycznej dozowania składników odżywcznych, regulacji pH, temperatury i wilgotności
AutoPot	Zestaw donic o pojemności 15 litrów ze zbiornikiem z wodą	Samodzielny zakup, montaż i kontrola	zbiornik z wodą i AQUAvalve, który automatycznie uzupełnia wodę w donicach	Samodzielnego monitorowanie i regulacja	Brak	Łatwy do przeniesienia	Wymagana większa powierzchnia, brak regulacji i monitorowania pH, temperatury i wilgotności
Vakplast	Półki z pojemnikiem na rośliny i zbiornikiem wodnym	Samodzielny zakup, montaż i kontrola	Pompa napowietrzająca, ciągły przepływ wody przez korzenie	Samodzielnego monitorowanie i regulacja	Brak	Dobре zagospodarowanie miejscem	Brak światła, brak automatycznej dozowania składników odżywcznych, regulacji pH, temperatury i wilgotności
Green Farm	Automatyczna mini szklarnia domowa	Wbudowane, zautomatyzowane	Pompa napowietrzająca	Samodzielnego monitorowanie i regulacja	Wbudowany czujnik temperatury	Mozliwość zarządzania osiągnięciem, który odpowiednio uruchamia wentylator	Brak automatyzacji monitorowania i uzupełniania składników odżywcznych, pH, wilgotności, skierowanie do warzyw liściastych
Freight Farms	Zautomatyzowany kontener	Wbudowane, automatyczna obsługa zależności od roślin	System nawadniający	Zautomatyzowane	System automatycznie reguluje temperaturę i wilgotność	W pełni zautomatyzowany system, aplikacja monitorująca	Cena i rozmiar uprawy, długotrwala instalacja, czas dostarczenia

## **2.7. Podsumowanie**

Rozwiązania aktualnie znajdujące się na rynku dbają w głównej mierze o dostarczanie roślinom napowietrzonej wody. W systemach tych użytkownik nadal musi dbać o takie czynniki jak temperatura, natężenie światła, wilgotność, pH czy potrzebne roślinom składniki. Mini szklarnia od firmy Green Farm wpasowuje się w część opisanych problemów jednak nadal brak jej odpowiedniego zarządzania poziomem pH, składnikami odżywczymi czy wilgotnością. Rozwiązanie to też skierowane jest tylko dla warzyw liściastych. Freight Farms i kontener z uprawą hydroponiczną spełniają te wymogi, jednak jego skala jest nieporównywalnie duża do pozostałych. Cena również nie jest przystępna dla zwykłego użytkownika.

Na rynku nie można znaleźć rozwiązania uniwersalnego, które mogłyby być wstanie wyhodować roślinę, która wymaga zmiennych warunków. Rośliny wchodzą w różne fazy i w zależności od rodzaju mogą być wymagane inne warunki do ich hodowli. Systemy dostępne na rynku nie są wstanie ich dostarczyć. Istotną rzeczą jest brak możliwości wprowadzania jakiekolwiek zmian. Dodatkowo użytkownik potrzebuje odpowiedniej wiedzy na temat warunków w jakich dana roślina powinna być uprawiana. Systemy nie są w żaden sposób osłonięte i zewnętrzne środowisko ma ogromny wpływ na hodowlę. Nie każdy użytkownik może pozwolić sobie na wybór odpowiednich warunków zewnętrznych. Wymagane byłoby rozwiązanie, które w jak największym stopniu ograniczałoby warunki w jakich system funkcjonuje.

### **3. ANALIZA POTRZEB I WYMAGAŃ ŚRODOWISKOWYCH**

Autor rozdziału: Jakub Zwolski

Przedstawiony w rozdziale 2.7 problem wymaga znalezienia rozwiązania, które spełni następujące wymagania:

- Kontrola czynników potrzebnych do rozwoju rośliny,
- dostosowanie warunków do danej rośliny, fazy w której się znajduje czy pory dnia,
- użytkownik bez wiedzy fachowej może obsługiwać urządzenie,
- uniezależnienie od warunków zewnętrznych.

#### **3.1. *Opis wymagań***

Warunki hodowli muszą być kontrolowane stale. Potrzebny jest system, który je monitoruje oraz odpowiednio nimi zarządza. Po przekroczeniu któregoś z warunków wymagane jest podjęcie przez niego odpowiedniej akcji. Za kontrolę czynników takich jak temperatura, wilgotność, pH, czy poziom składników odżywczych w wodzie muszą odpowiadać odpowiednie czujniki. Do schłodzenia środowiska wymagany jest system chłodzący. Za zmianę wilgotności w systemie potrzebne są odpowiednie narzędzia do jej zwiększenia lub obniżenia. Do regulacji nawozu i pH wymagana jest pompa potrafiąca zadozować odpowiednią ilość nawozu. Wymagane jest także oświetlenie zamontowane w urządzeniu, które mogłoby odpowiednio dopasowywać poziom natężenia światła do pory dnia czy wymaganego przez rośliny oświetlenia. W celu kontroli zmian w urządzeniu potrzebny był system z zamontowaną kamerą, który byłby w stanie zapisywać zdjęcia. Potrzebny jest także system napowietrzający, aby uniemożliwić rozwój mikroorganizmów w wodzie co mogłoby doprowadzić do procesu gnilnego korzeni roślin.

Do odczytywania odpowiednich warunków przez system wymagane byłyby pliki z danymi, które posiadałyby informacje o warunkach dla danej rośliny. Dzięki takiemu rozwiązaniu użytkownik nie będzie musiał znać potrzebnych wymogów dla danej rośliny. W celu sprawdzenia możliwości takiego systemu hydroponicznego wymagane było wybranie odpowiedniej rośliny. Podjętym w pracy wyborem zostały truskawki, które posiadają różne fazy, są wrażliwe na warunki, a także dają mierzalny wynik w postaci owoców.

W celu sprawdzenia wydajności i możliwości hodowli w różnych warunkach maszyna powinna być uniezależniona od warunków zewnętrznych.

#### **3.2. *Parametry środowiskowe w hodowli truskawek***

Wiedza potrzebna do hodowli truskawek została zebrana ze stron poświęconych takim hodowlom i uprawom hydroponicznych oraz odpowiadającym tematowi prac naukowych. W artykule "Long-day control of flowering in everbearing strawberries" [1] można wyczytać informacje o odpowiednio wysokiej temperaturze potrzebnej do uprawy wynoszącej około 27°C, która sprzyja rozwojowi lepszym jakościowo owocom. Następną rzeczą, o której należy zadbać jest odpowiednia ilość składników mineralnych dostarczanych do roślin. W tym temacie można się posłużyć informacjami od producenta nawozu hydroponicznego oraz regulatorów pH. Do ustalenia konkretnej wartości służy kalkulator ze strony producenta, którego to nawozy są wykorzystywane w

prototype.

	Weeks	Light pro day (hours)	Aqua Vega ml per 25.6 liters	Aqua Flores ml per 25.6 liters	RHIZO TONIC ml per 25.6 liters	CANNA ZYM ml per 25.6 liters	PK 13/14 ml per 25.6 liters	CANNA BOOST ml per 25.6 liters	PPM (Eutech)+	PPM (Eutech) Total
<b>Growth</b>										
Start / rooting (3 – 5 days) Make substrate wet	<1	18	46(A) 46(B)	-	102	-	-	-	576	576
Vegetative phase I Plant develops in volume	0-3 <sup>a</sup>	18	56(A) 56(B)	-	51	64	-	-	704	704
Vegetative phase II Up to growth stagnation after fructification or appearance of the formation of flowers	2-4 <sup>b</sup>	12	71(A) 71(B)	-	51	64	-	51 - 102 <sup>e</sup>	896	896
<b>Flowering</b>										
Generative Period I Flowers or fruits develop in length. Growth in height achieved	2-3	12	-	86(A) 86(B)	13	64	-	51 - 102	1024	1024
Generative Period II Development of the volume (breadth) of flowers or fruit	1	12	-	86(A) 86(B)	13	64	38	51 - 102	1152	1152
Generative Period III Development of the mass (weight) of flowers or fruit	2-3	12	-	64(A) 64(B)	13	64	-	51 - 102	768	768
Generative Period IV Flowers or fruit ripening process	1-2	10-12 <sup>c</sup>	-	-	-	64 - 128 <sup>d</sup>	-	51 - 102	-	-
<b>Generative Phase</b>										

Rys. 3.1. Porcje nawozu przy odpowiednich fazach (Tabela pochodzi ze strony: cannagardening.com)

Z (rys.3.1) można wyczytać, że w pierwszym okresie poziom nawozu składników w wodzie powinien być na poziomie około 750 ppm, a w późniejszym na poziomie wyższym niż 1000 ppm. W broszurze "Wszystko, co chcielibyście wiedzieć o systemach recyrkulacji" [2] od tego samego producenta można wyczytać o odpowiednim pH, które powinno się zawierać w od około 4.8 do 6.3, gdyż wtedy w wodzie dostępnych jest najwięcej potrzebnych składników odżywcznych (rys.3.2).



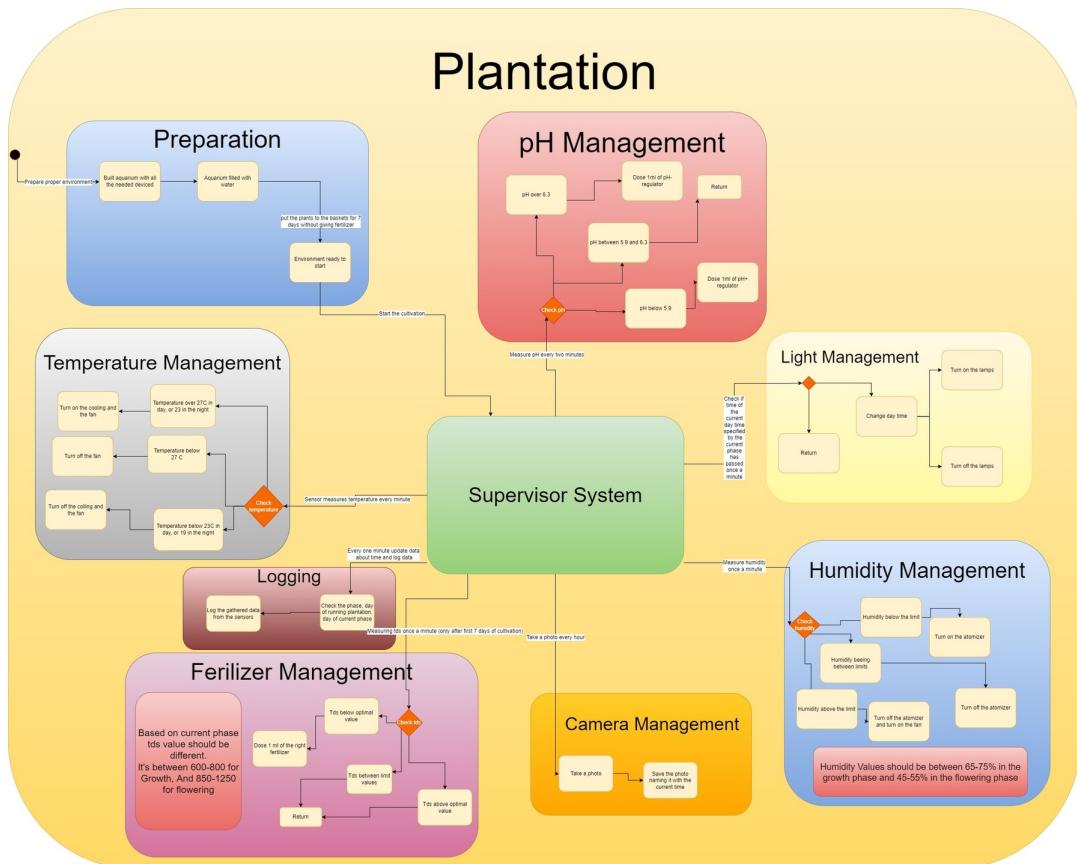
Rys. 3.2. Dostępność składników odżywcznych a pH (Wykres pochodzi z broszury aqua.pdf)

Można również przeczytać o problemach związanych z niedoborami tych składników, a przede wszystkim o odpowiednim ich rozpoznaniu. Kluczowe w uprawie jest także zadbanie o odpowiednią ilość światła. Na ten temat można się wiele dowiedzieć z artykułu "Essentials for growing hydroponic strawberries successfully – Hort Americas" [3], który mówi o  $20 \text{ mol m}^{-2} \text{d}^{-1}$ . Według artykułu How To Build Your Own Hydroponics Grow Room – Dialing In Your Temperature and Humidity (part 3) [4] wilgotność powietrza powinna maleć wraz ze wzrostem i rozwojem rośliny i z 70% dojść do około 40%.

## 4. PROJEKT MODELU STERUJĄCEGO HODOWŁĄ

Autor rozdziału: Jakub Zwolski

### 4.1. Diagram stanów

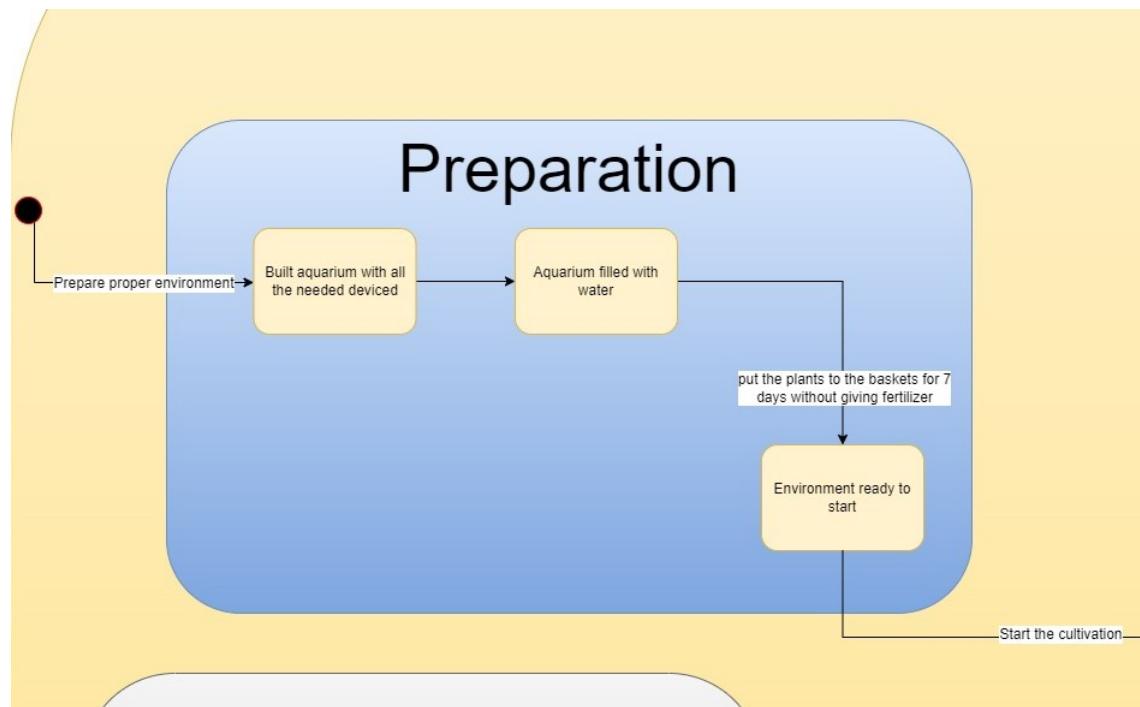


Rys. 4.1. Diagram stanów

W celu zaplanowania pracy nad systemem został przygotowany diagram stanów, który przedstawia logikę zarządzania oraz podaje konkretne informacje odnośnie warunków do odpowiedniej uprawy. Dane diagramu zostały przygotowane na podstawie odpowiednich warunków do rozwoju truskawek. Sekcja przygotowania środowiska została omówiona w rozdziale 4.1.1. W rozdziale 4.1.2 zostały opisane czynności wykonywane przez system zarządzający. Decyzję jakie podejmuje system w celu zarządzania temperaturą omówione zostały w rozdziale 4.1.3. Wykonywanie akcji dotyczących kontroli i dostosowania pH znajdujące się na czerwonym diagramie "pH Management" zawarte zostały w rozdziale 4.1.4. Podsystem do zbierania i zapisywania danych z kamery na diagramie nazwany został jako "Camera Management", a jego dokładniejszy opis znajduje się w rozdziale 4.1.5. Na diagramie o nazwie "Light Management" zawarty został proces zarządzania światłem, jego opis natomiast zawarty jest w rozdziale 4.1.6. "Humidity Management" to nazwa diagramu przedstawiającego proces zarządzania wilgotnością. 4.1.7 to rozdział, który w sposób dokładniejszy je opisuje. Zbieranie danych znajduje się na diagramie "Logging", opis akcji

zawarty został w rozdziale 4.1.8. Rozdział 4.1.9 dotyczy dawkowania nawozu i dokładniej opisuje akcje wykonywane opisane na diagramie "Fertilizer Management".

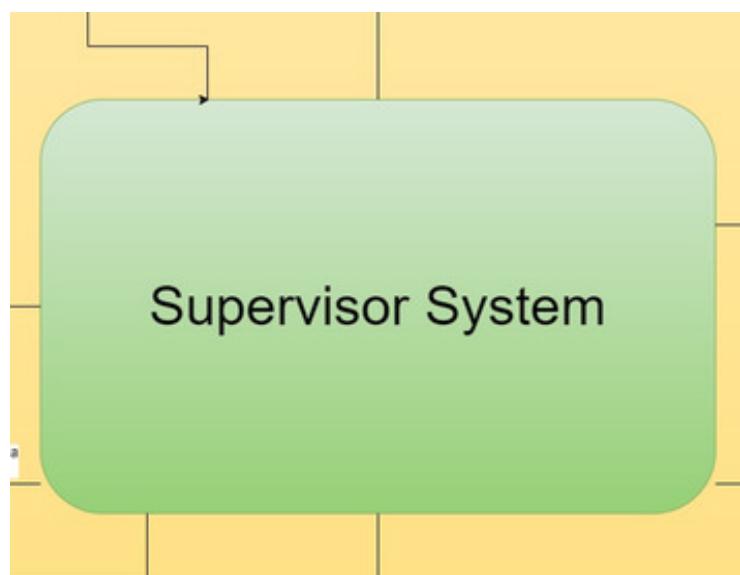
#### 4.1.1. Przygotowanie



Rys. 4.2. Diagram przygotowań

Pierwszym etapem jest przygotowanie odpowiedniego środowiska i napełnienie zbiornika wodą. Na diagramie została zawarta także uwaga o konieczności pozostawienia roślin w wodzie bez składników odżywczych na pierwsze 7 dni w celu przyswojenia ich do nowego środowiska. Etap ten jest jednorazowy dla każdego użytkownika. Po rozpoczęciu uprawy pierwszą fazą jest faza odpoczynku rośliny podczas której nie jest dostarczany nawóz do wody.

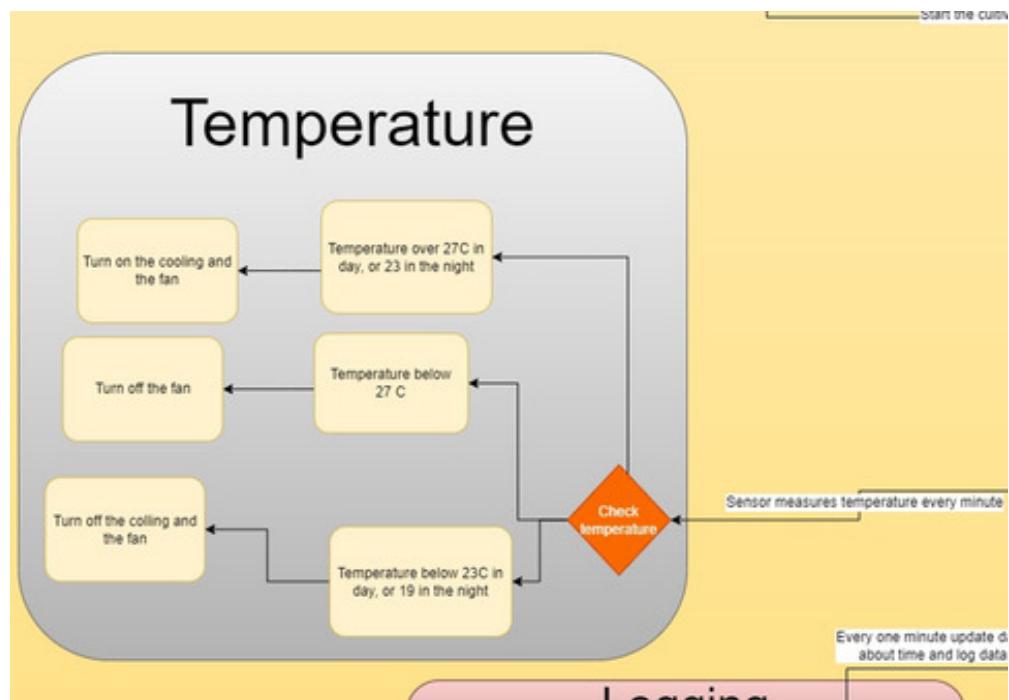
#### 4.1.2. System zarządzający



Rys. 4.3. Diagram systemu zarządzającego

Na diagramie znajduje się system nadzorujący, który co określony czas wywołuje odpowiednie akcje przedstawione na pozostałych diagramach.

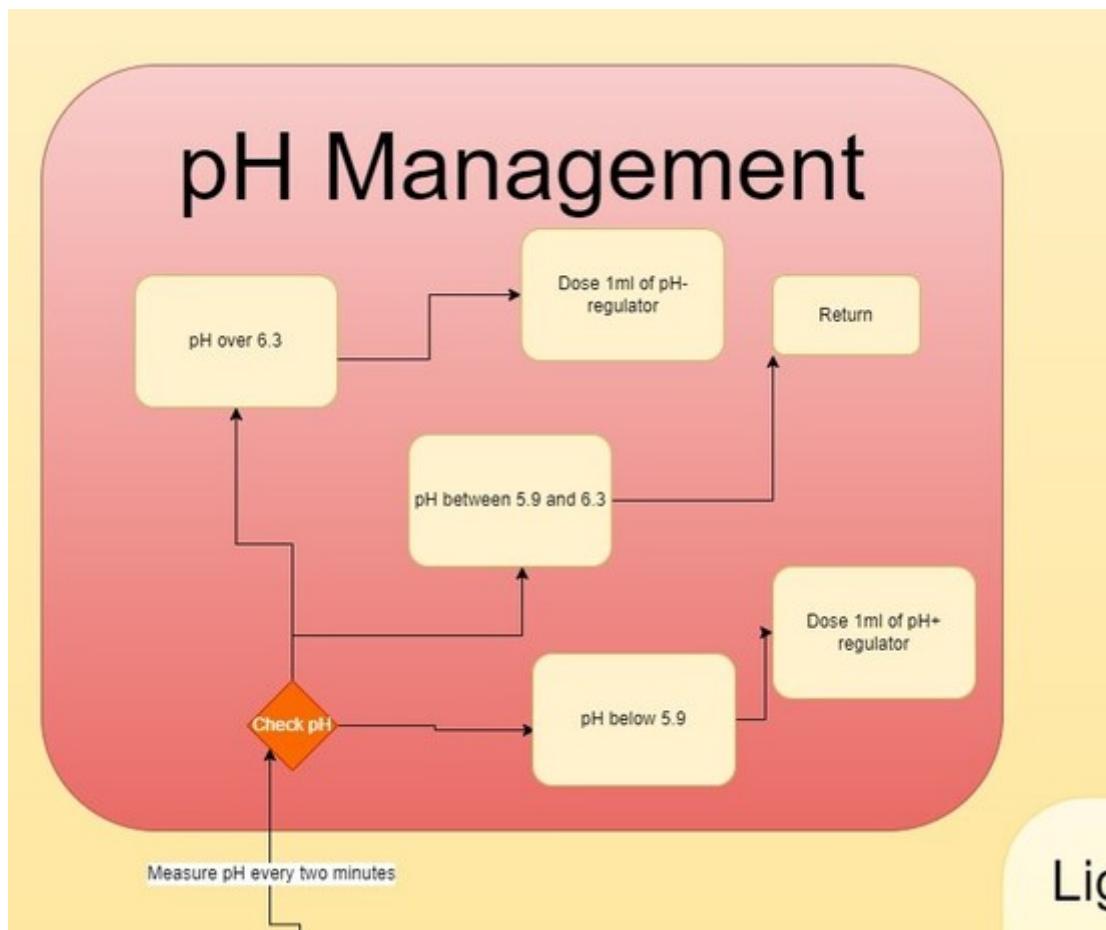
#### 4.1.3. Zarządzanie temperaturą



Rys. 4.4. Diagram modułu temperatury

Wywołanie pomiaru oraz zarządzanie temperaturą, odbywa się co minutę. System wysyła żądanie do czujnika o zmierzenie temperatury, a następnie podejmuje odpowiednie kroki. W przypadku zbyt wysokiej temperatury jest to wyłączenie chłodzenia i wentylatora. Jeżeli temperatura spadnie poniżej maksymalnej granicy wyłączone jest chłodzenie i wentylator.

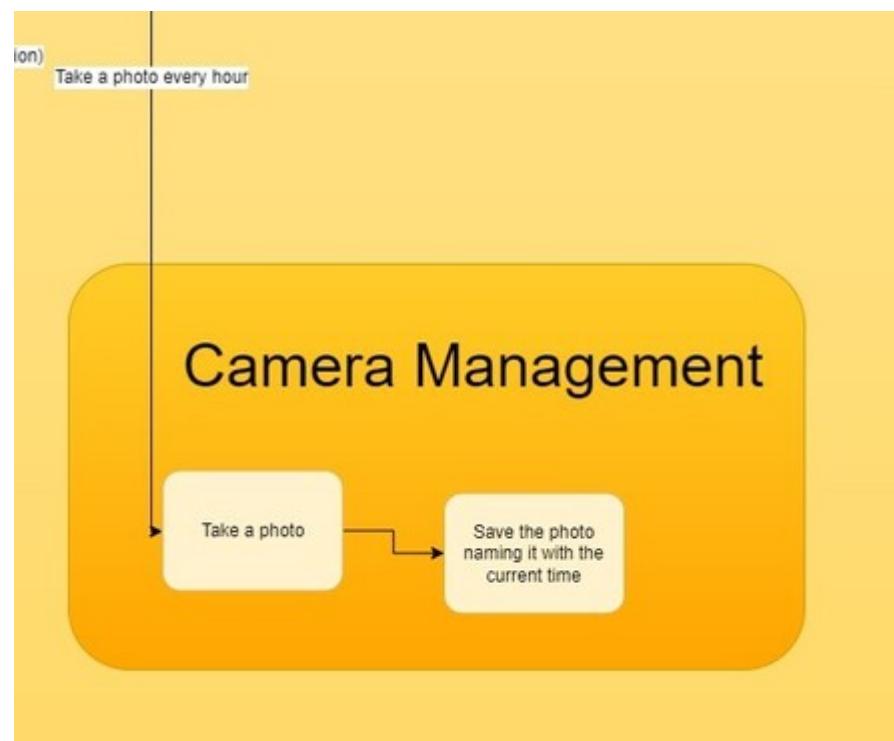
#### 4.1.4. Zarządzanie pH



Rys. 4.5. Diagram obsługi pH

Akcje przedstawione na diagramie wywoływanie są co dwie minuty. W pierwszej kolejności system wysyła żądanie do czujnika o zmierzenie poziomu pH. Następnie gdy któryś z limitu jest przekroczony do wody dodawany jest 1 ml odpowiedniego regulatora pH. Czas akcji różni się od pozostałych akcji, aby mieć pewność, że roztwór zostanie rozprowadzony po całym zbiorniku.

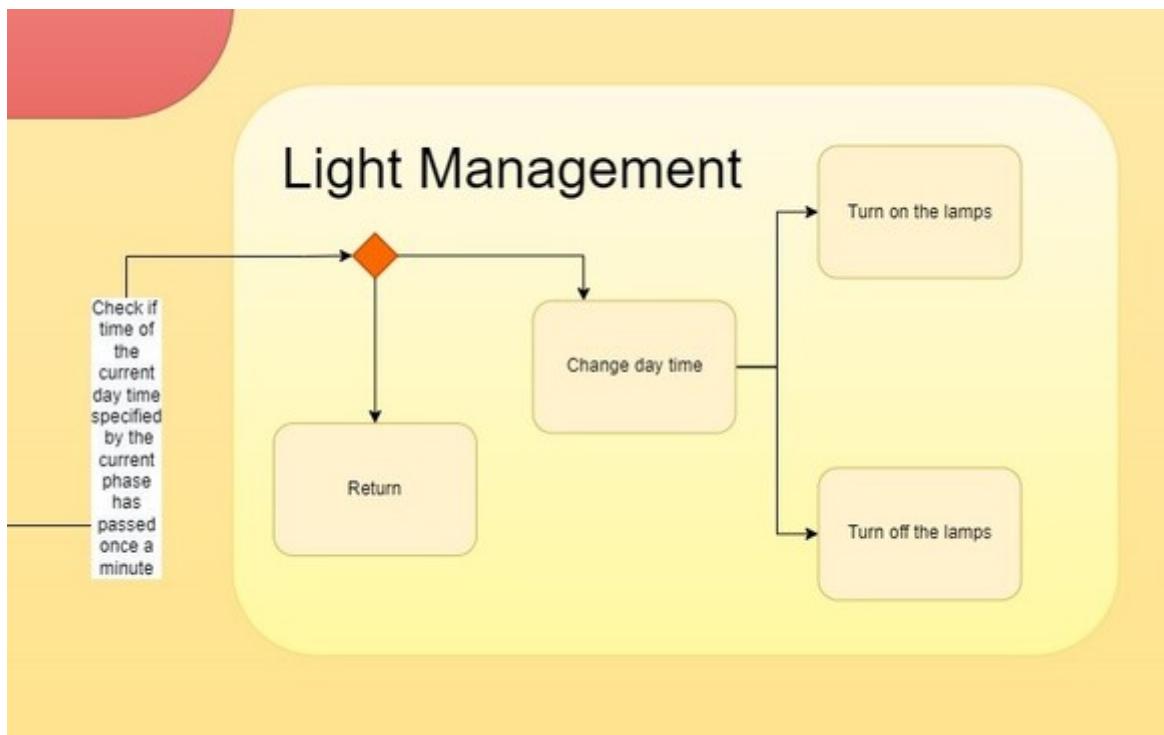
#### 4.1.5. Zarządzanie kamerą



Rys. 4.6. Diagram obsługi kamery

System co godzinę wysyła żądanie do Raspberry Pi o zrobienie i zapisanie zdjęcia. Raspberry wykonuje zdjęcia, a następnie odpowiednio je nazywa używając aktualnej godziny i daty. Na końcu następuje zapis zdjęcia do folderu.

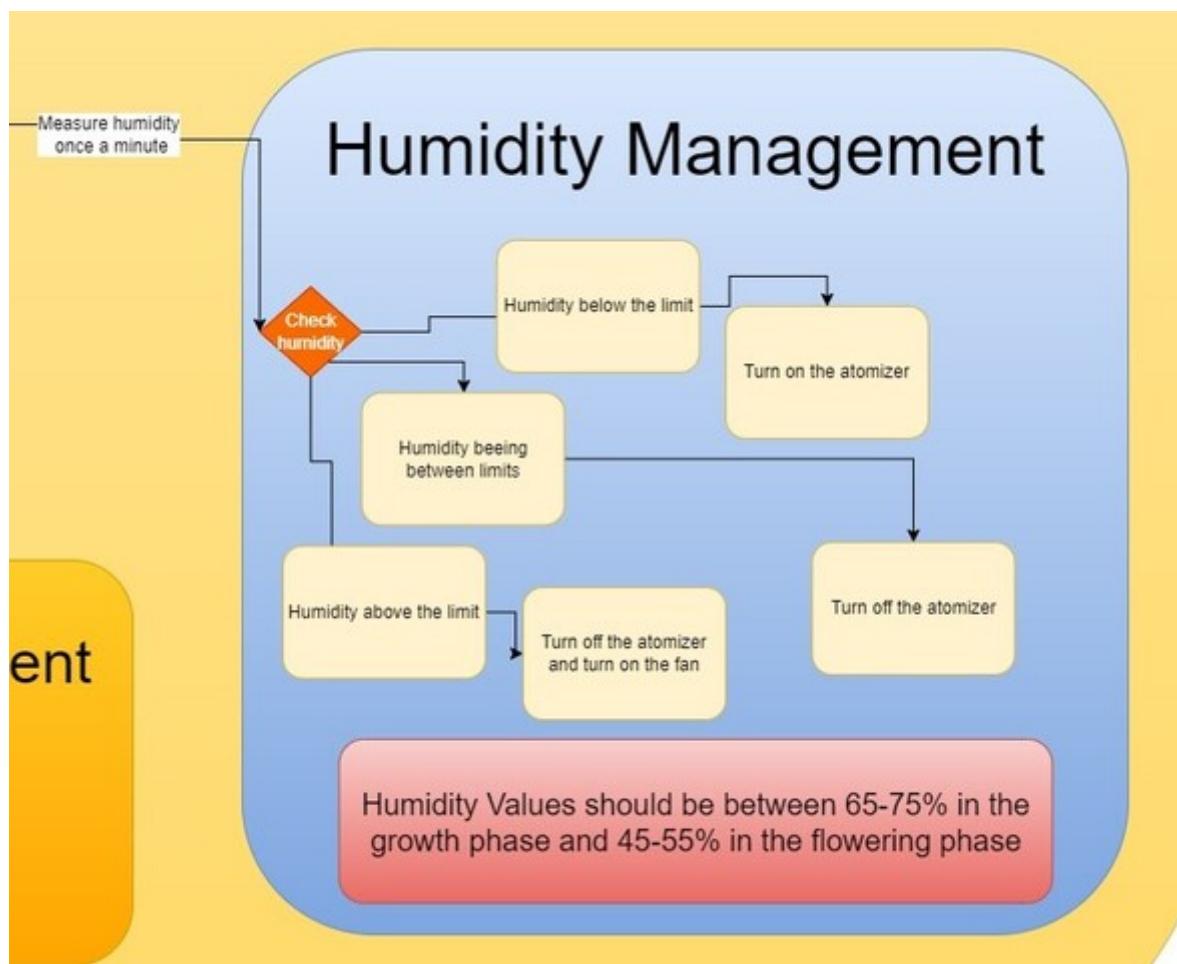
#### 4.1.6. Zarządzanie światłem



Rys. 4.7. Diagram obsługi światła

Akcję związane z modułem światła wykonywane są co minutę. Na początku pobierana jest dana faza i aktualny czas. Następnie sprawdzane jest, czy czas na daną porę dnia upłynął. W zależności czy wcześniej była pora dnia lub nocy wyłączone lub włączane zostają światła.

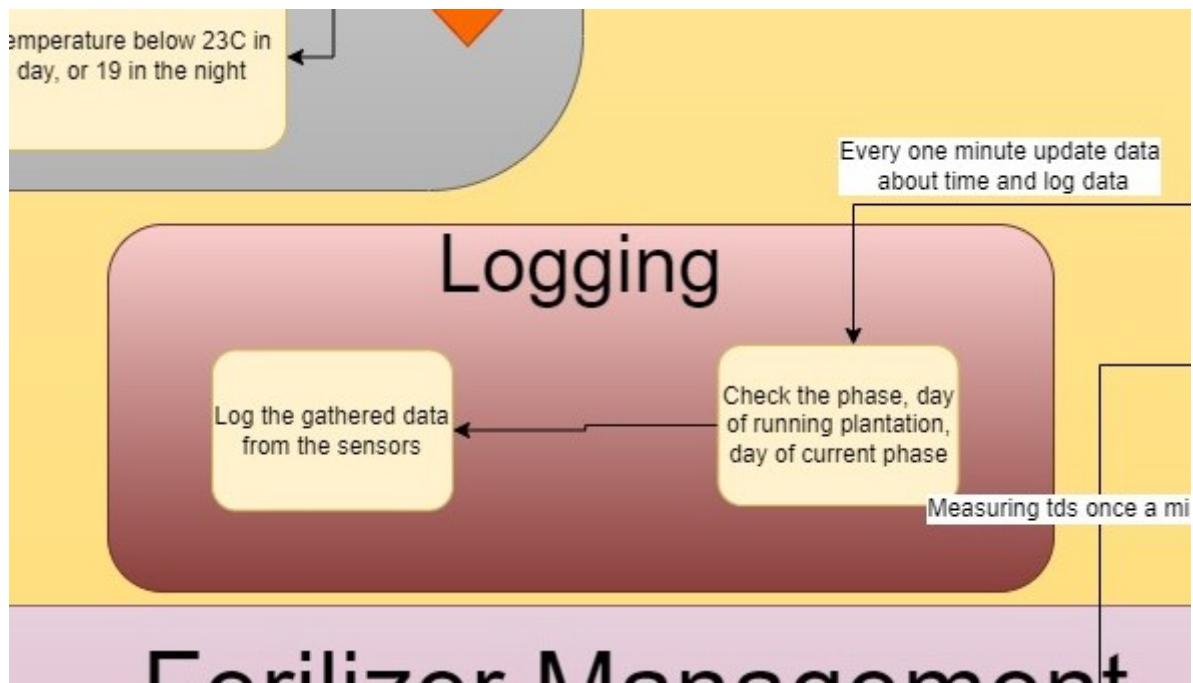
#### 4.1.7. Zarządzanie wilgotnością



Rys. 4.8. Diagram obsługi wilgotności

Moduł uruchamiany jest co minutę. Raspberry Pi po otrzymaniu żądania o sprawdzenie wilgotności wywołuje akcję na czujniku i wysyła wynik z powrotem. Przy przekroczeniu limitu poziomu wilgotności system odpowiednio wysyła żądanie ołączenia atomizera. Przy wartościach poniżej normy lub wartościach mieszczących się w normie wysyła żądanie o jego wyłączenie. Przy wartościach poniżej normy dodatkowo wysyłana jest prośba o zatrzymanie pracy wentylatora. Na czerwonym diagramie widocznym na Rys. 4.8 można wyczytać odpowiednie wartości dla truskawek przy danej fazie rozwoju.

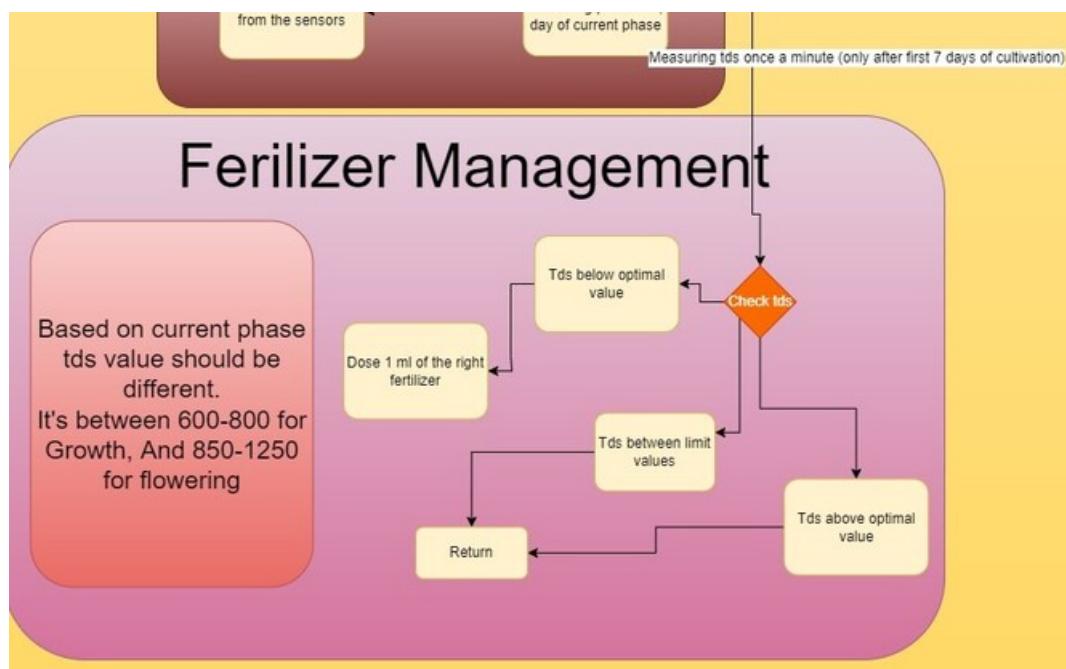
#### 4.1.8. Logowanie



Rys. 4.9. Diagram obsługi logowania

System co minutę wysyła żądanie do Raspberry PI o zalogowanie zebranych danych. System podejmuje akcję zmiany fazy, dnia lub dnia fazy, gdy ta jest wymagana.

#### 4.1.9. Zarządzanie nawozem



Rys. 4.10. Diagram obsługi nawozu

W pierwszej kolejności generowany jest sygnał sterujący czujnikiem TDS i odczytywany jest poziom zawartości nawozu w wodzie. Gdy poziom będzie poniżej normy system generuje sygnał sterujący pompą perystaltyczną. Pompa zadozuje 1 ml, odpowiedniego w zależności od

fazy rozwoju, nawozu. Wartości przygotowane są pod czujnik TDS mierzący ilość soli w wodzie i różnią się od siebie w zależności od aktualnej fazy.

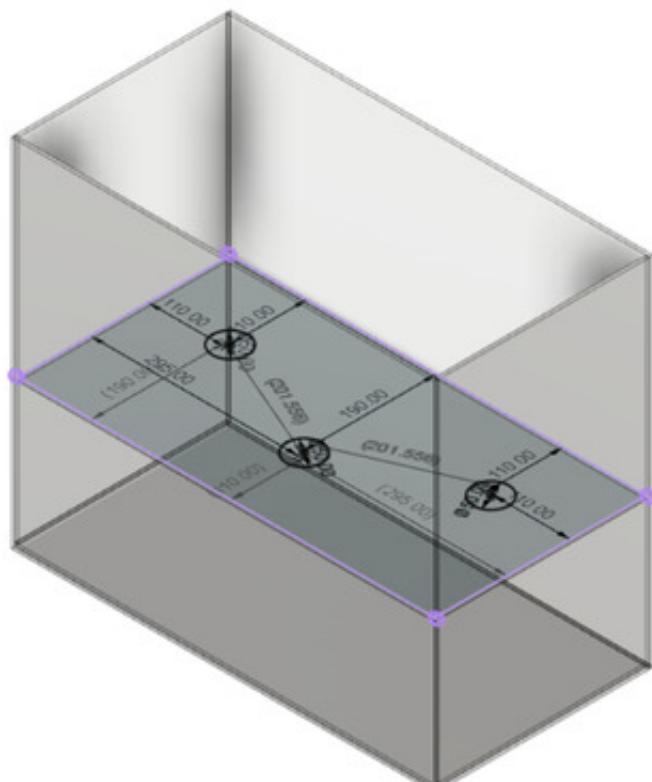
## 5. PROJEKT URZĄDZENIA

Autor rozdziału: Jakub Świsłocki

W poniższym rozdziale opisano elementy konstrukcji prototypu. W pierwszej kolejności (sekcja 5.1) zaprezentowano konstrukcję obudowy urządzenia wraz z tacą i koszykami do mocowania roślin. Następnie przedstawiono budowę pokrywy wraz z montażem źródeł światła. W kolejnej sekcji (5.2) opisano montaż czujników do pomiaru warunków środowiska powietrznego (temperatura i wilgotność) oraz mechanizmów do kontroli tych warunków. Sekcję 5.3 poświęcono na opis montażu czujników kontrolujących środowisko wodne (pH, TDS) i ich połączenie z jednostką sterującą. Opisano również zastosowane mechanizmy do kontroli tych wartości i ich montaż w prototypowym urządzeniu. Kolejna sekcja (5.4) poświęcona została sposobom kontroli i sterowania oświetleniem. W sekcji 5.5 opisane zostały pozostałe elementy budowy: kamera, pompa napowietrzająca oraz zasilanie. Sekcja 5.6 przedstawia schemat elektryczny urządzenia. Ostatnia sekcja tego rozdziału poświęcona została na przedstawienie zdjęć gotowej konstrukcji prototypu i opis elementów znajdujących się na nich.

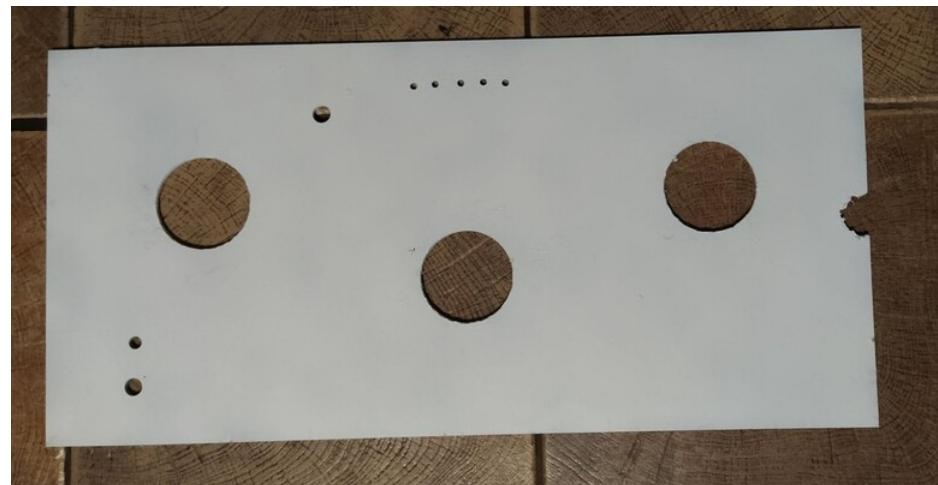
### 5.1. Konstrukcja główna

Podstawą urządzenia jest "akwarium" wykonane ze szkła akrylowego (potocznie: pleksi). Ma ono formę prostopadłościanu o wymiarach 60x30x50cm (długość, szerokość, wysokość) z uchylną płytą górną.

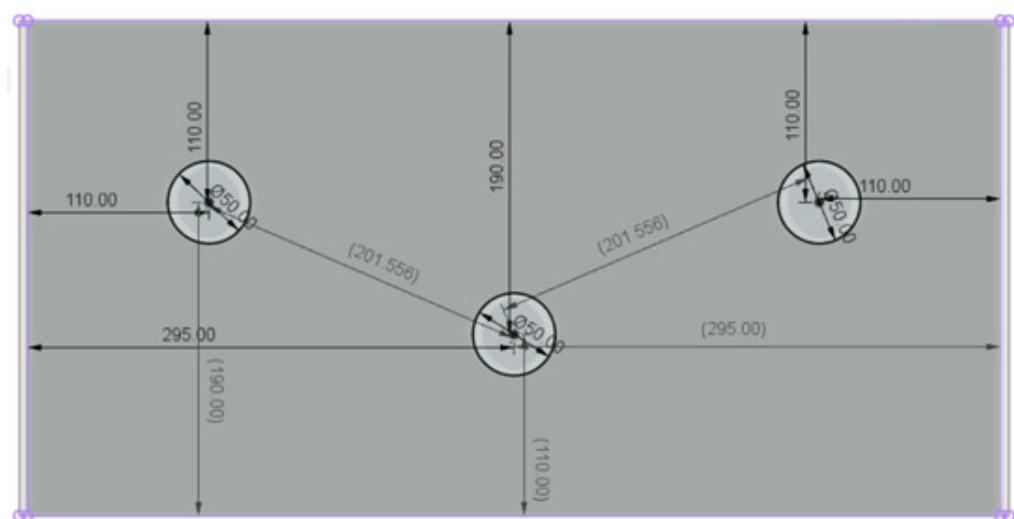


Rys. 5.1. Model "akwarium"

Na ilustracji 5.1 widać model akwarium bez pokrywy. Płyta znajdująca się w środku urządzenia rozdziela wnętrze na dwie części. W części dolnej znajduje się woda natomiast część górna przeznaczona jest dla zielonej części rośliny.

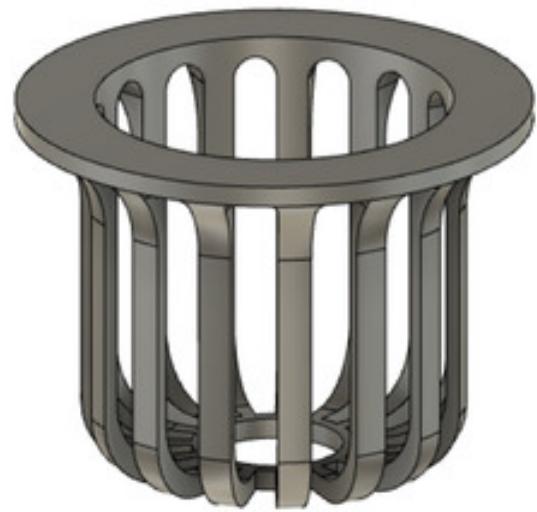


Rys. 5.2. Płyta środkowa "akwarium"



Rys. 5.3. Projekt płyty środkowej

Jak wskazują ilustracje 5.2 i 5.3 płyta rozdzielająca posiada otwory w których umieszczamy "koszyczki" (rys.5.4).



Rys. 5.4. Model koszyczka na krzak truskawki

Koszyczki umieszczone są tak aby korzenie rośliny zanurzone były w wodzie natomiast zielona część rośliny znajdowała się w górnej części urządzenia (rys.5.6). Z racji braku gleby, roślina ustabilizowana jest poprzez małe kamyczki którymi wypełniony jest koszyczek. Doniczki umożliwiają roślinie wypuszczanie korzeni do środowiska wodnego przez otwory.

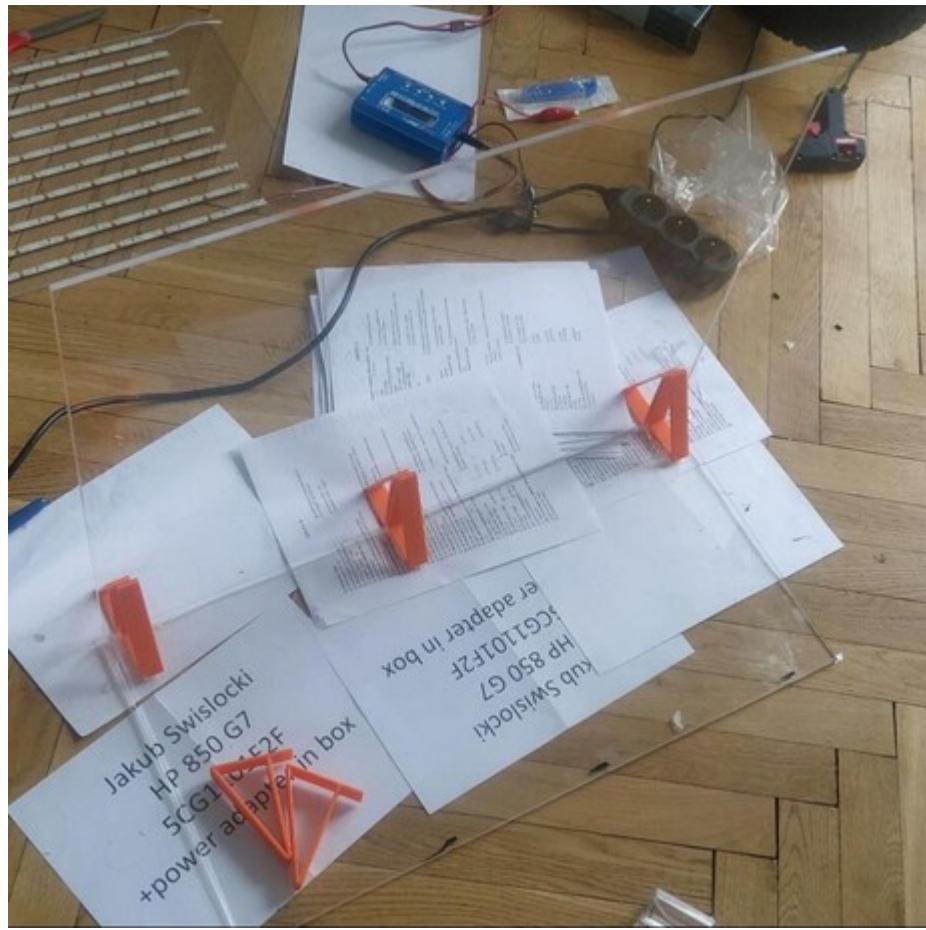


Rys. 5.5. Krzak w koszyczku widziany z góry



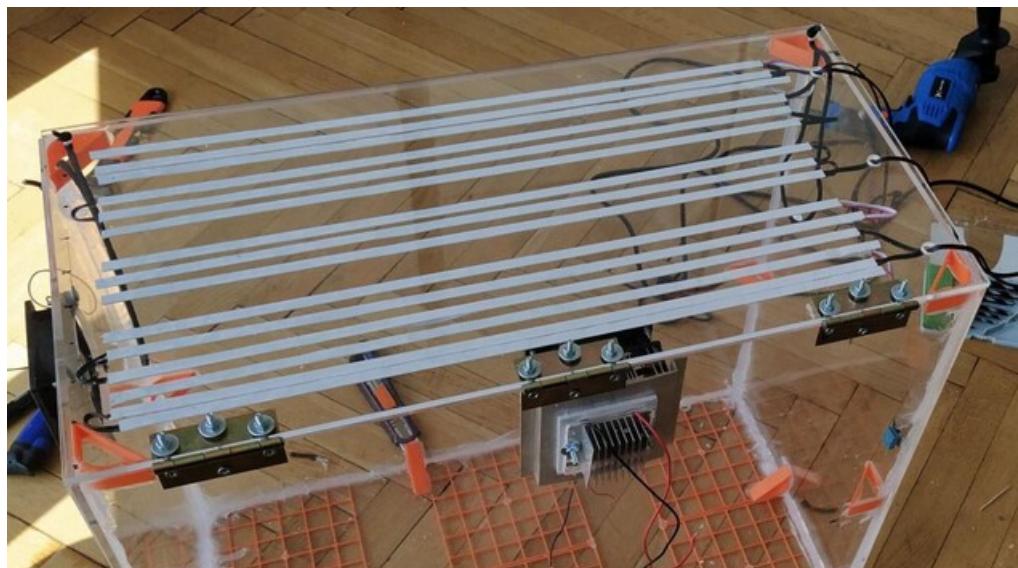
Rys. 5.6. Krzak w koszyczku widziany z boku (koszyczek przechodzi przez płytę środkową)

Aby zapewnić szczelność i wytrzymałość konstrukcji, płyty szkła akrylowego sklejone zostały żywicą epoksydową i uszczelnione silikonem akwariowym.



Rys. 5.7. Pierwsza ściana przyklejona do płyty dolnej

Żywica epoksydowa zapewnia wytrzymałość konstrukcji, a silikon akwariowy zapobiega wyciekom wody. Ponadto silikon akwariowy nie emituje do wody szkodliwych substancji przez co nie zaszkodzi roślinom rosnącym w wodzie. Warstwa silikonu chroni też przed kontaktem wody z żywicą epoksydową. Górną pokrywą przymocowana została na metalowych zawiasach co umożliwia jej otwieranie (rys.5.8 i rys.5.9).



Rys. 5.8. Pokrywa górska przymocowana do urządzenia

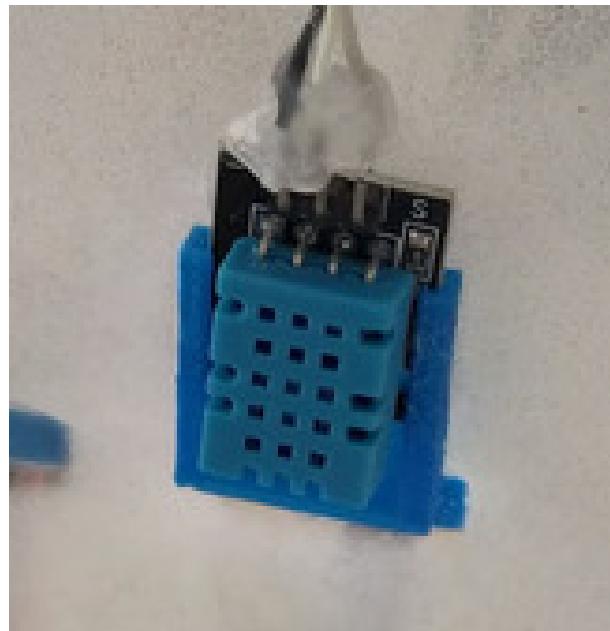


Rys. 5.9. Otwarta pokrywa górną

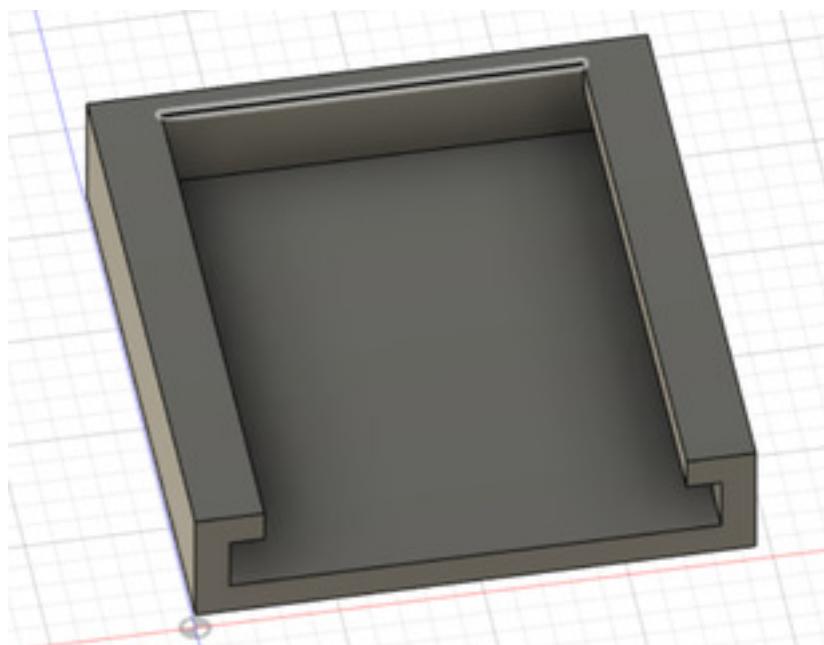
## 5.2. Kontrola środowiska powietrznego

### 5.2.1. Kontrola temperatury i wilgotności

Temperaturę i wilgotność w urządzeniu monitorują 2 czujniki DHT11 (rys. 5.10). Są one rozmieszczone w przeciwnieństwach końcach urządzenia aby można było uśrednić odczyty. Zamocowane zostały one przy użyciu podstawek wydrukowanych za pomocą technologii 3D (rys. 5.11).



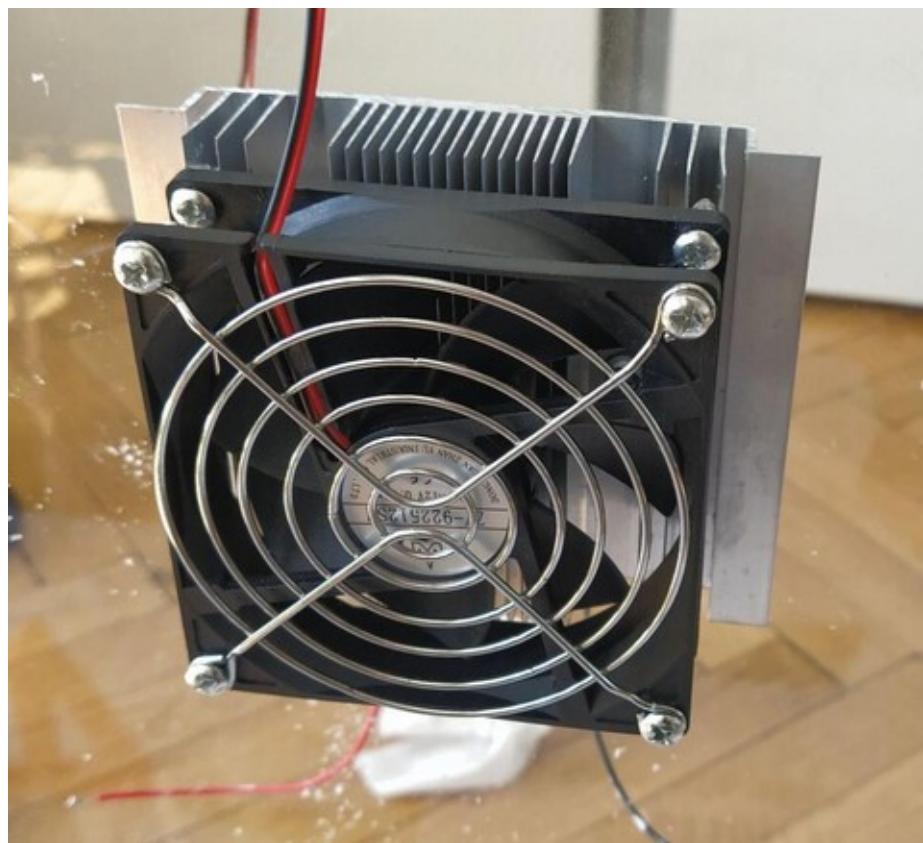
Rys. 5.10. Czujnik DHT11 zamontowany w podstawce



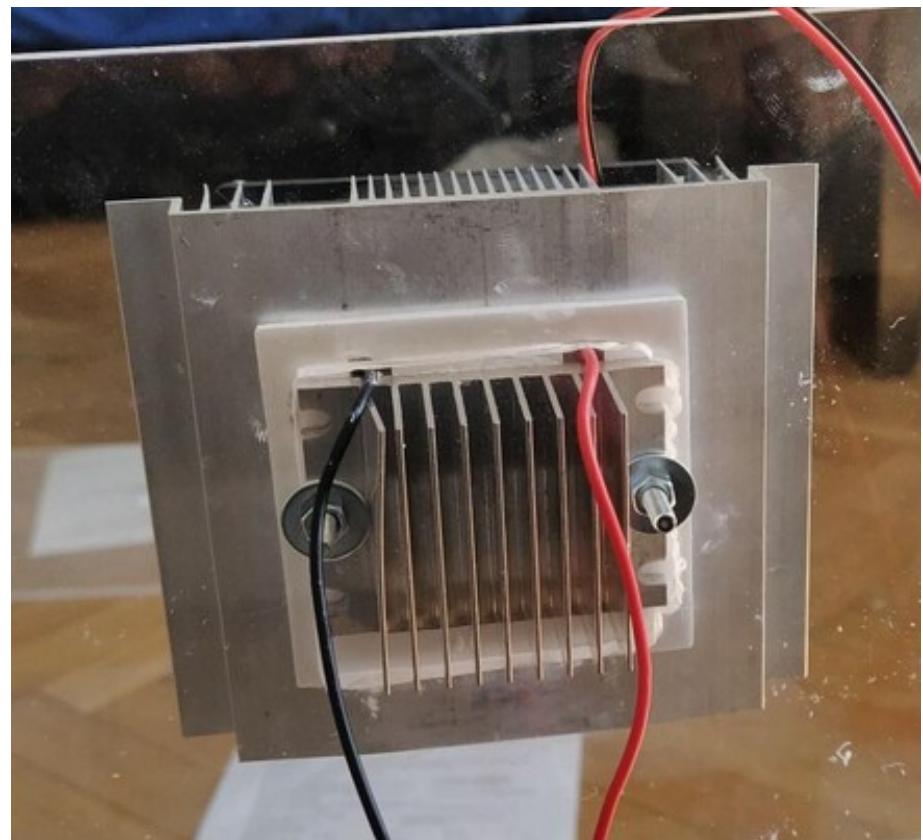
Rys. 5.11. Podstawka czujnika DHT11

#### 5.2.2. Pomiar i sterowanie temperaturą

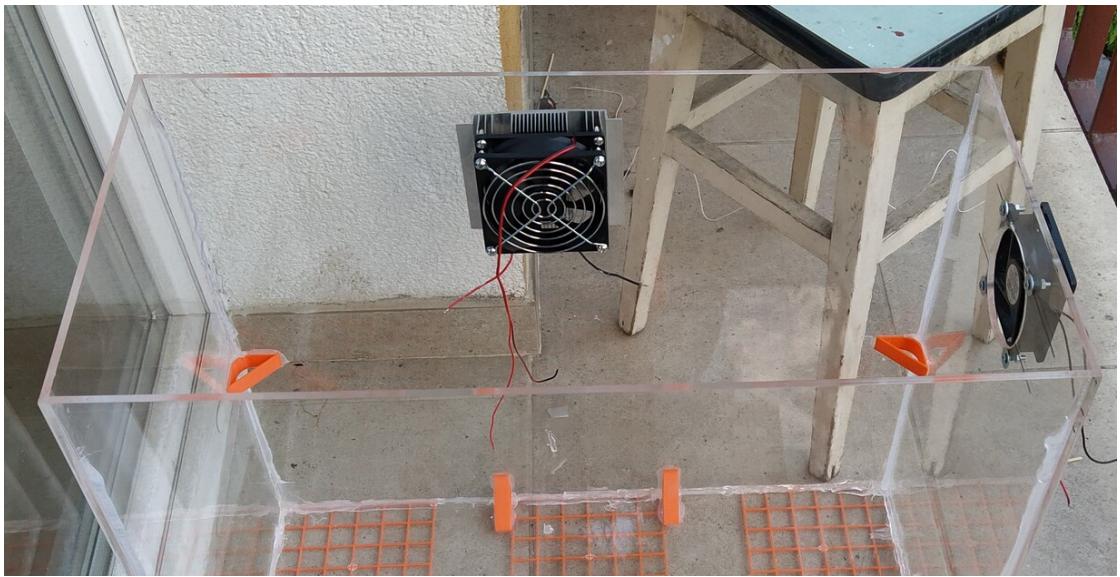
Do obniżania temperatury wewnętrz urządzczenia służy moduł chłodzenia składający się z ogniw Peltiera umieszczonego pomiędzy dwoma radiatorami (rys. 5.15). Strona gorąca ogniw wraz z radiatorem i wentylatorem wystawione są na zewnątrz (rys. 5.13) urządzenia natomiast strona zimna umieszczona jest wewnątrz (rys. 5.12).



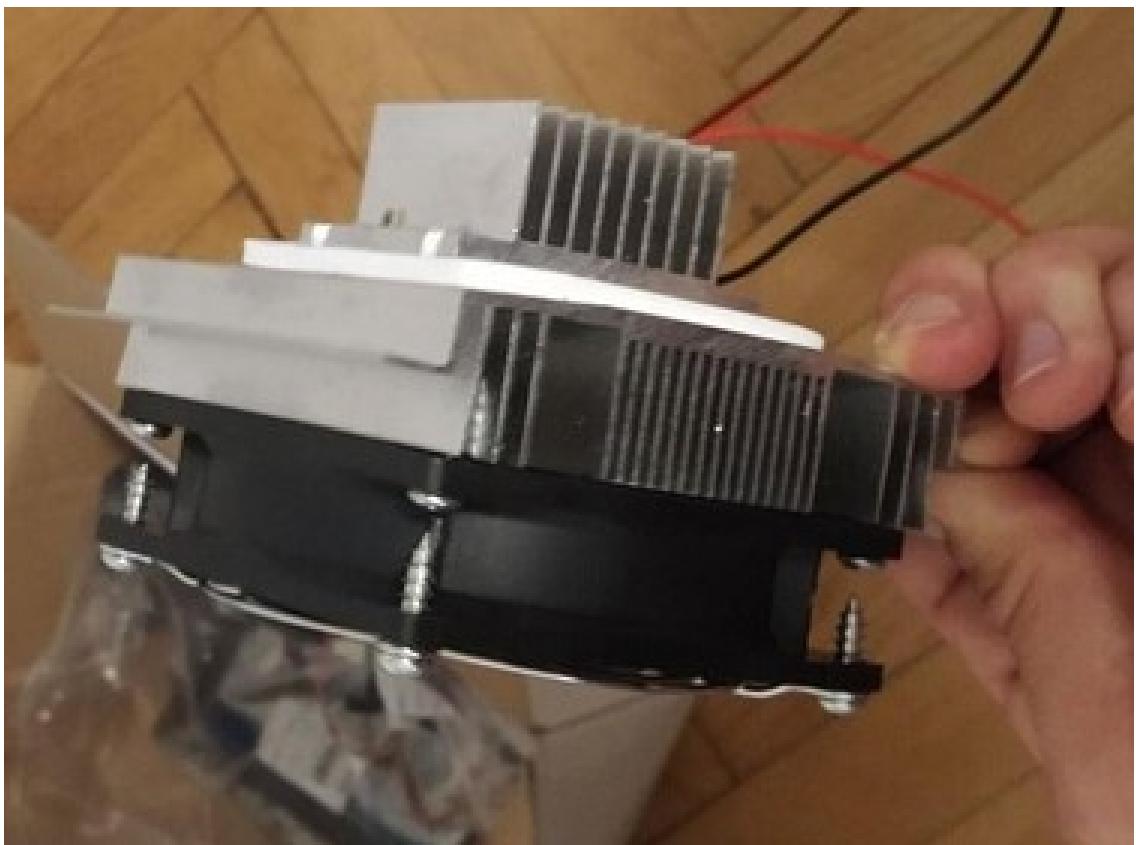
Rys. 5.12. Moduł chłodzenia (strona zimna)



Rys. 5.13. Moduł chłodzenia (strona gorąca)

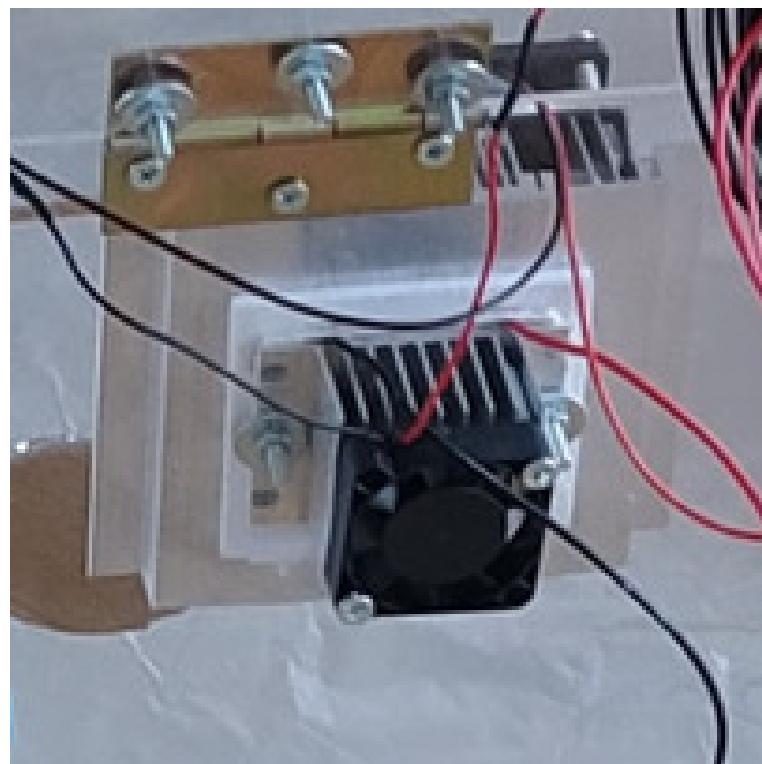


Rys. 5.14. Moduł chłodzenia zamontowany w ścianie urządzenia

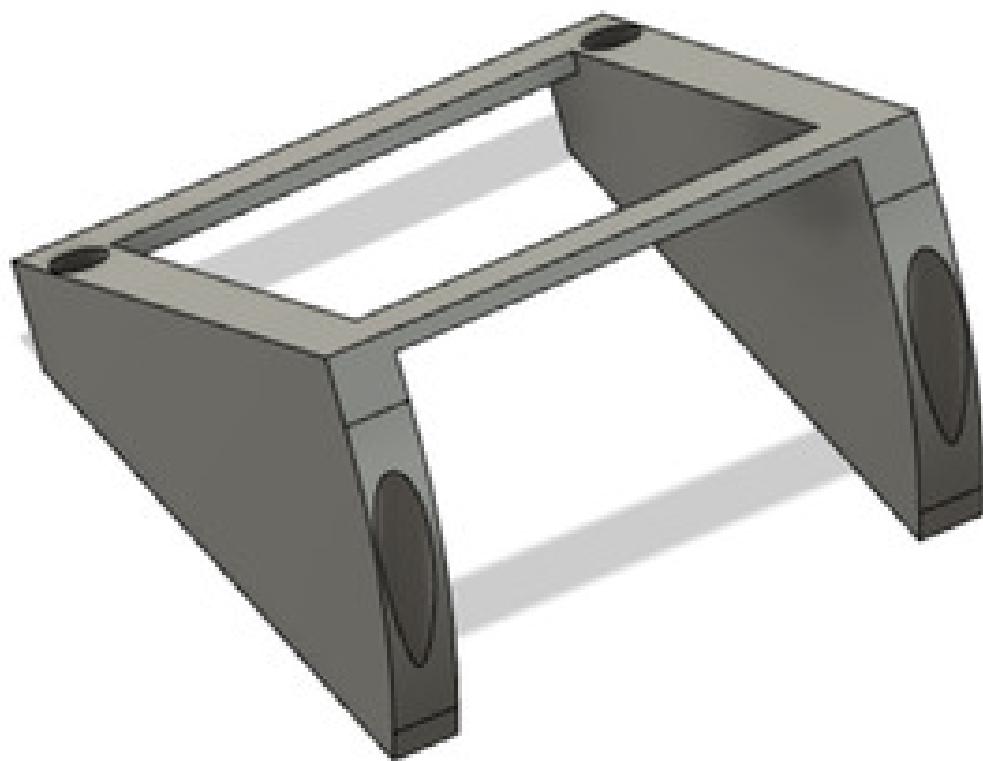


Rys. 5.15. Moduł chłodzenia

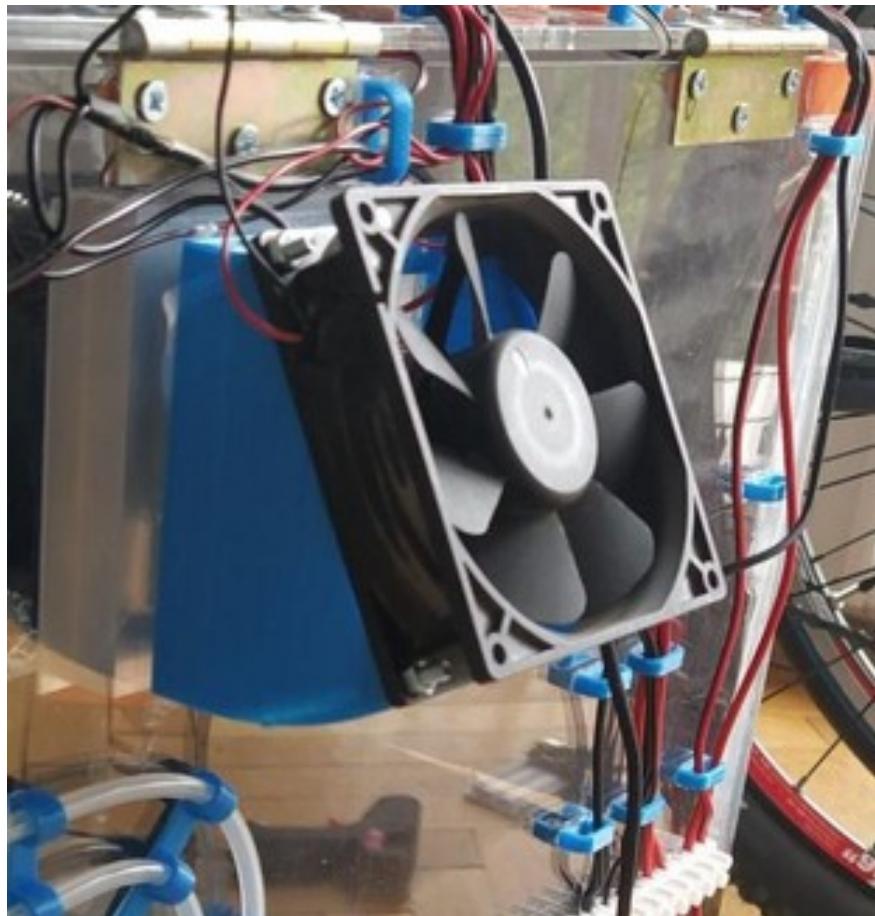
W pierwszej wersji układu chłodzenia stronę gorącą chłodził mały wentylator (rys.5.16). Niestety okazało się to niewystarczające, dlatego w kolejnej wersji zastosowany został większy wentylator (rys.5.18) oraz specjalna podstawka kątowa (rys.5.17), która zwiększyła przepływ powietrza przez radiator.



Rys. 5.16. Strona gorąca modułu chłodzenia chłodzona małym wentylatorem



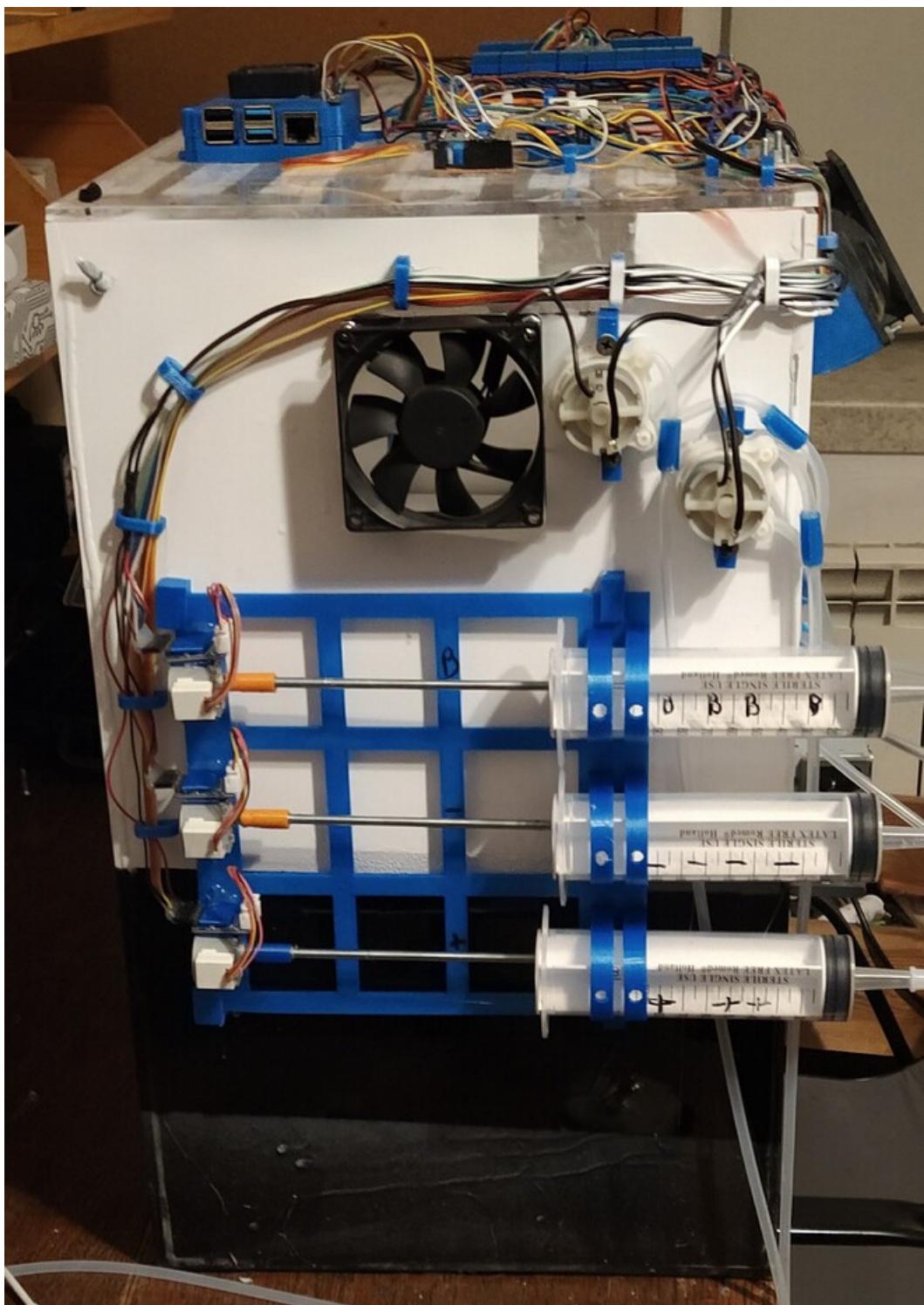
Rys. 5.17. Podstawka kontowa pod większy wentylator



Rys. 5.18. Strona gorąca modułu chłodzenia z większym wentylatorem na podstawce

#### 5.2.3. *Pomiar i sterowanie wilgotnością*

Aby zachować odpowiedni poziom wilgotności powietrza wewnętrz urządzenia potrzebna była możliwość zarówno zwiększenia jak i obniżenia tego poziomu. Do zmniejszenia poziomu wilgotności służy zwykły wentylator który wdmuchuje powietrze zewnętrzne do wewnętrz urządzenia. Jest on zamontowany w jednej ze ścian "akwarium" (rys. 5.19).



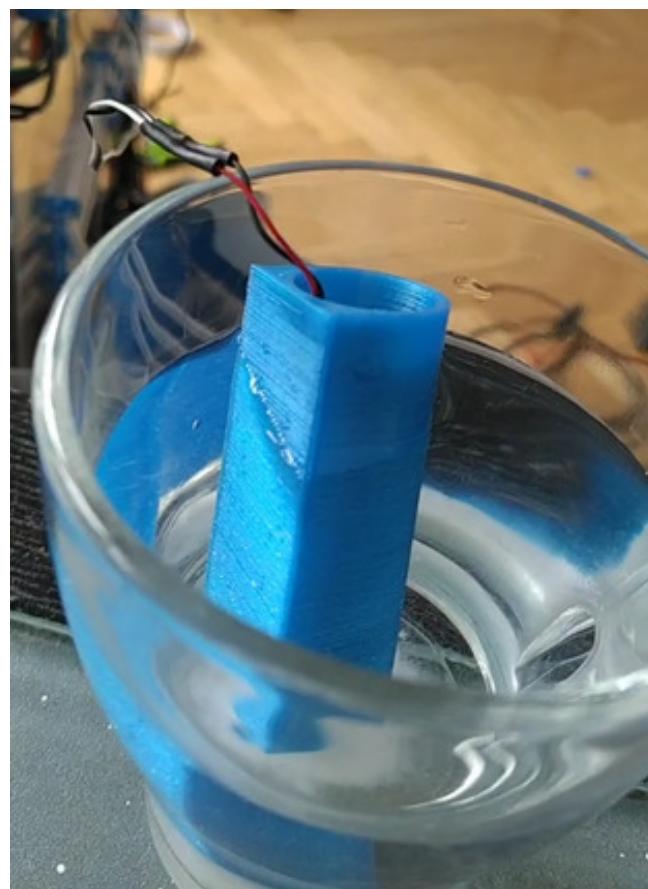
Rys. 5.19. Wentylator w ścianie bocznej

Do zwiększenia wilgotności zastosowany został ultradźwiękowy atomizer piezoelektryczny rys.5.20. Rozpyla on w powietrzu mikro kropelki wody, które następnie szybko odparowują. Składa się on z membrany piezoelektrycznej oraz generatora, który tworzy przemienne napięcie o częstotliwości wprawiającej membranę w drgania. Aby zapewnić stały kontakt membrany z wodą, przy nie zawsze stałym poziomie wody, zastosowana została rurka (rys.5.21 i rys.5.22) z watą bawełnianą w, której górnej części umieszczona została membrana. Natomiast dolna jej część

zanurzona została w zbiorniku dolnym urządzenia.



Rys. 5.20. Generator zasilający atomizer piezoelektryczny

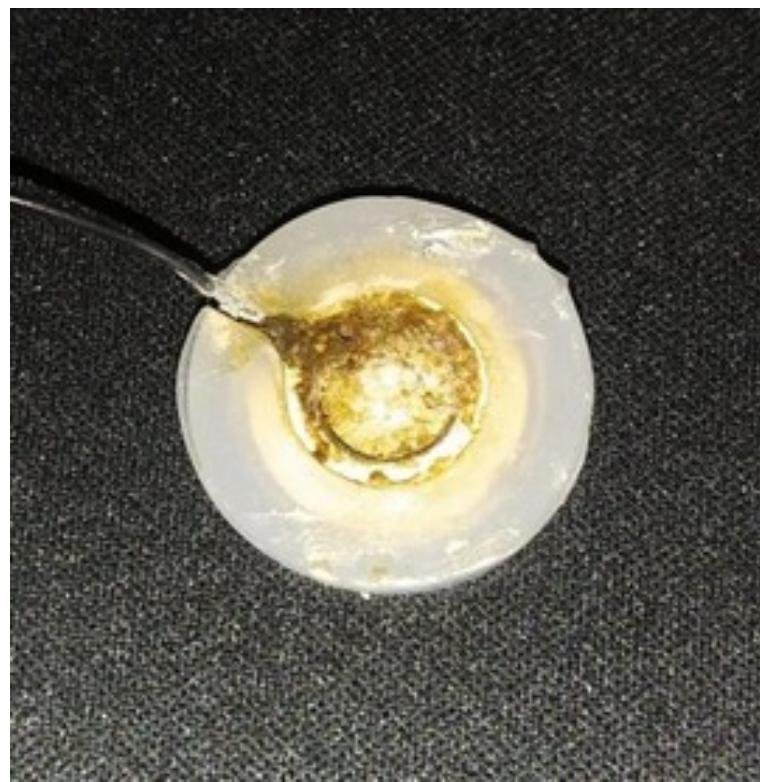


Rys. 5.21. Rurka z wata i atomizerem (próbne uruchomienie w szklance z wodą)



Rys. 5.22. Model rurki atomizera

Niestety okazało się, że przez dużą zawartość soli mineralnych w wodzie w dolnej części urządzenia, na membranie osadzać zaczęła się warstwa soli rys.5.23, która uniemożliwiła jej pracę. Membrana wraz z rurką przeniesione zostały do oddzielnego zbiorniczka z czystą wodą rys.5.24.



Rys. 5.23. Atomizer pokryty osadem



Rys. 5.24. Atomizer z rurką przeniesiony do pojemnika z czystą wodą

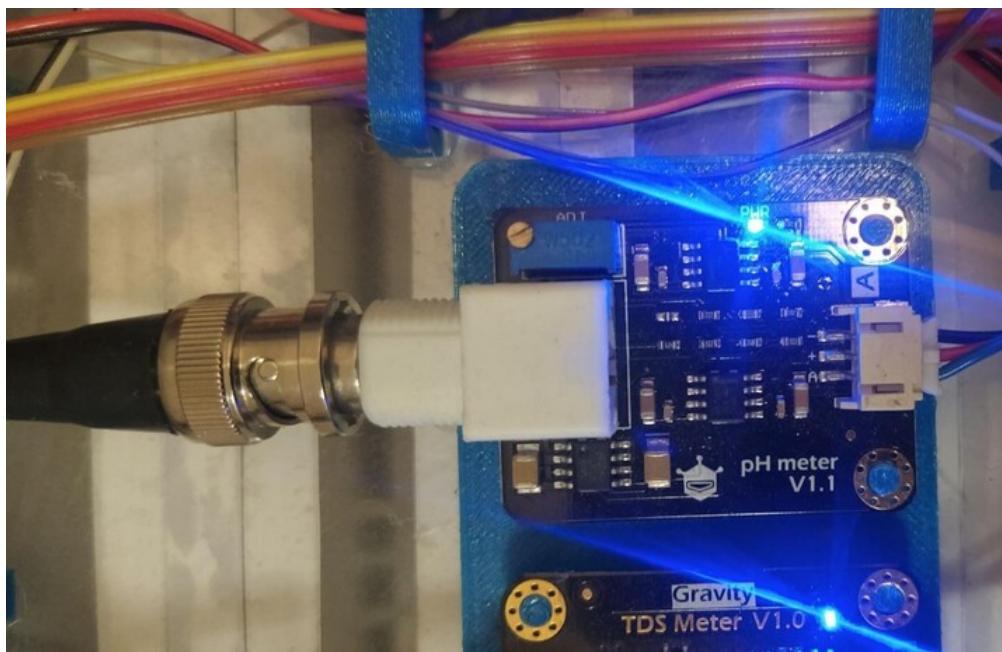
### **5.3. Kontrola Środowiska wodnego**

#### *5.3.1. Kontrola poziomu pH*

Poziom pH kontrolowany jest za pomocą analogowego czujnika pH. Składa się on z sondy pomiarowej rys. 5.25 oraz modułu (rys. 5.26), który wskazania sondy przekształca na sygnał analogowy. Sonda zamocowana jest w płycie średkowej tak aby dolna część sondy zanurzona była w wodzie (rys. 5.27). Z uwagi na zanieczyszczenia wody kranowej, destylowanej oraz wałań pH wszystkich innych powszechnie dostępnych płynów, do kalibracji czujnika używany jest specjalny odczynnik o pH = 7.



Rys. 5.25. Sonda czujnika pH



Rys. 5.26. Moduł czujnika pH



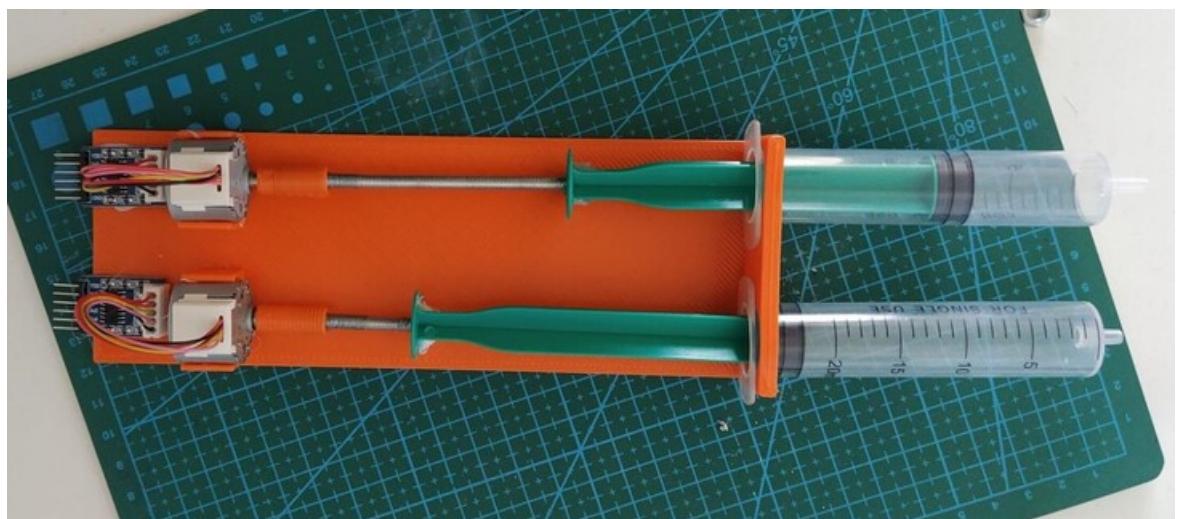
Rys. 5.27. Sondy czujników pH i TDS przechodzące przez płytę środkową

Do sterowania poziomem pH wody używane są odczynniki pH+ i pH-, które dozowane są za pomocą pomp strzykawkowych. Pompy strzykawkowe pozwalają na precyzyjne dozowanie niewielkich ilości cieczy. Pompa strzykawkowa składa się z:

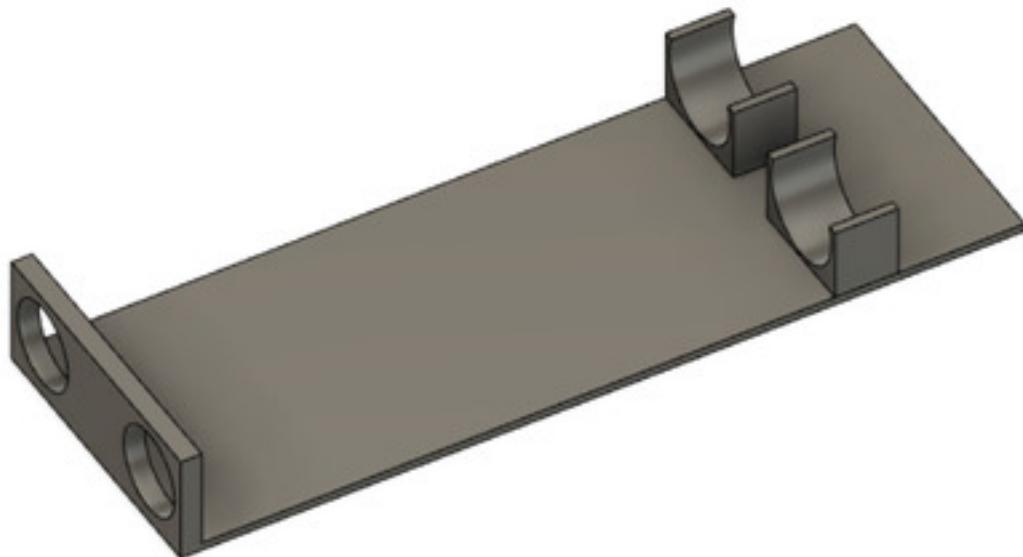
1. cylindra w którym znajduje się płyn
2. tłoka który tłoczy ciecz
3. silnika krokowego napędzającego tłok
4. sterownika silnika krokkowego
5. obudowy

Silnik napędza tłok poprzez przekładnię śrubową. Silnik obraca gwintowany pręt po którym porusza się nakrętka przymocowana do tłoka.

Pierwsza konstrukcja zestawu pomp zakładała dwie pompy (pH+ oraz pH-) i opierała się na strzykawkach o pojemności 20ml.

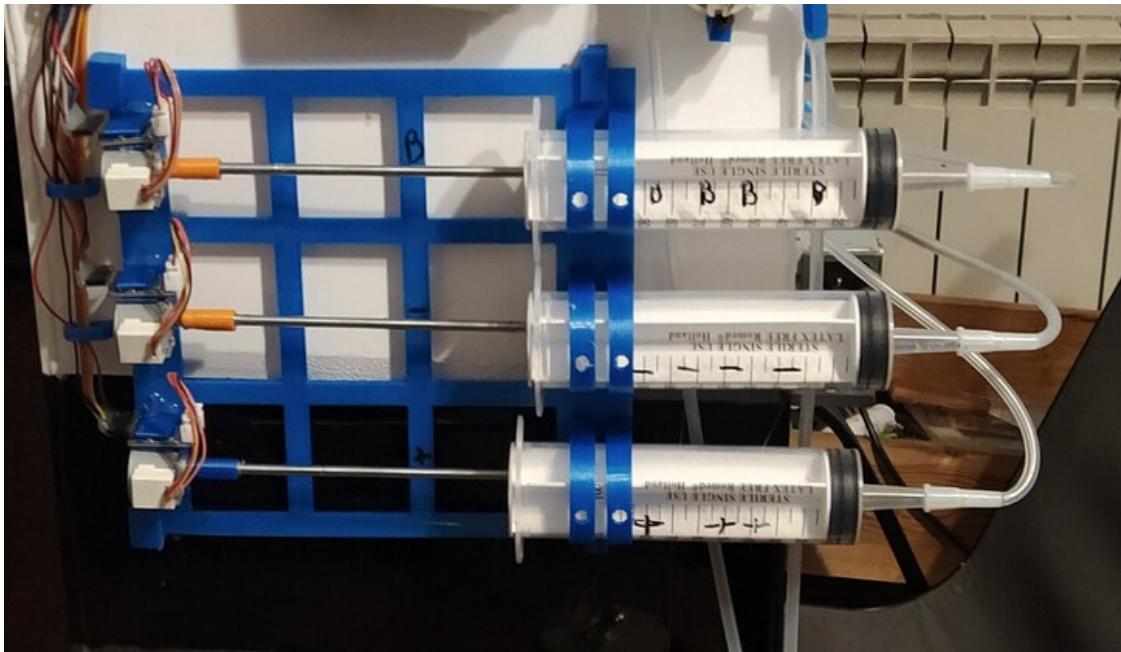


Rys. 5.28. Moduł pomp (wersja pierwsza)

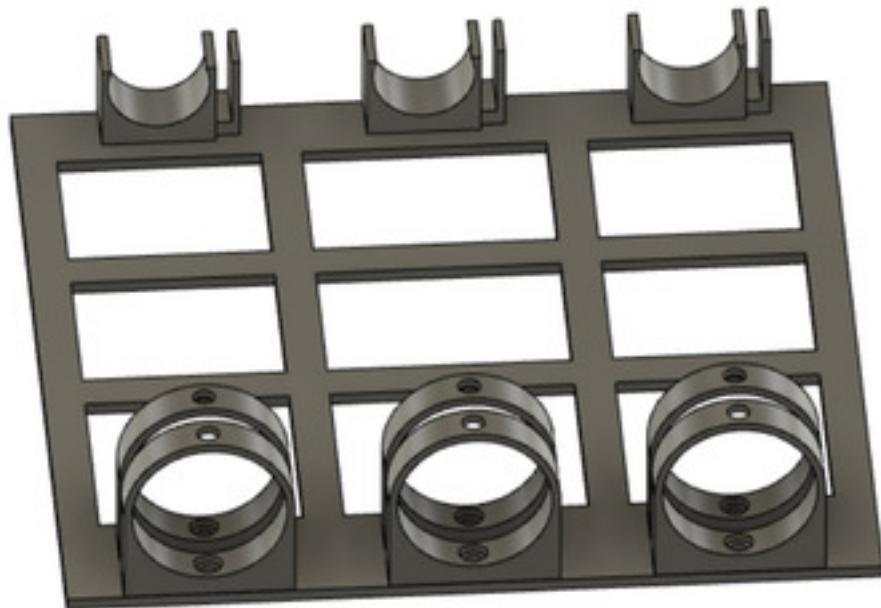


Rys. 5.29. model uchwytu pomp (wersja pierwsza)

Druga wersja konstrukcji zawierała już 3 pompy, ponieważ została dodana pompa dozująca środek wzmacniający owocowanie. Konstrukcję modułu pomp stanowi rama (rys. 5.31), do której przymocowane są wszystkie części pomp. Rama umożliwia również szybkie wyjęcie całego modułu pomp w razie potrzeby (wystarczy odpiąć rurki od strzykawek oraz złącza od sterowników silników krokowych). Konstrukcja nowych pomp została oparta o strzykawki o pojemności 100ml. Dzięki zwiększeniu pojemności strzykawek zmniejszyła się potrzeba uzupełniania płynów podczas hodowli.



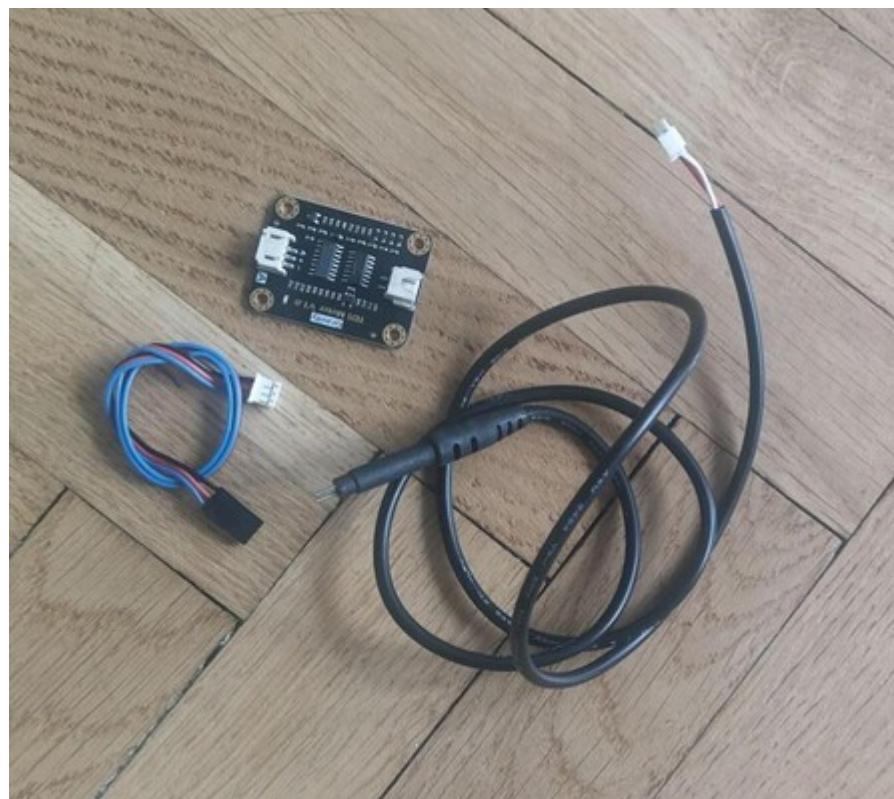
Rys. 5.30. Moduł pomp (wersja druga)



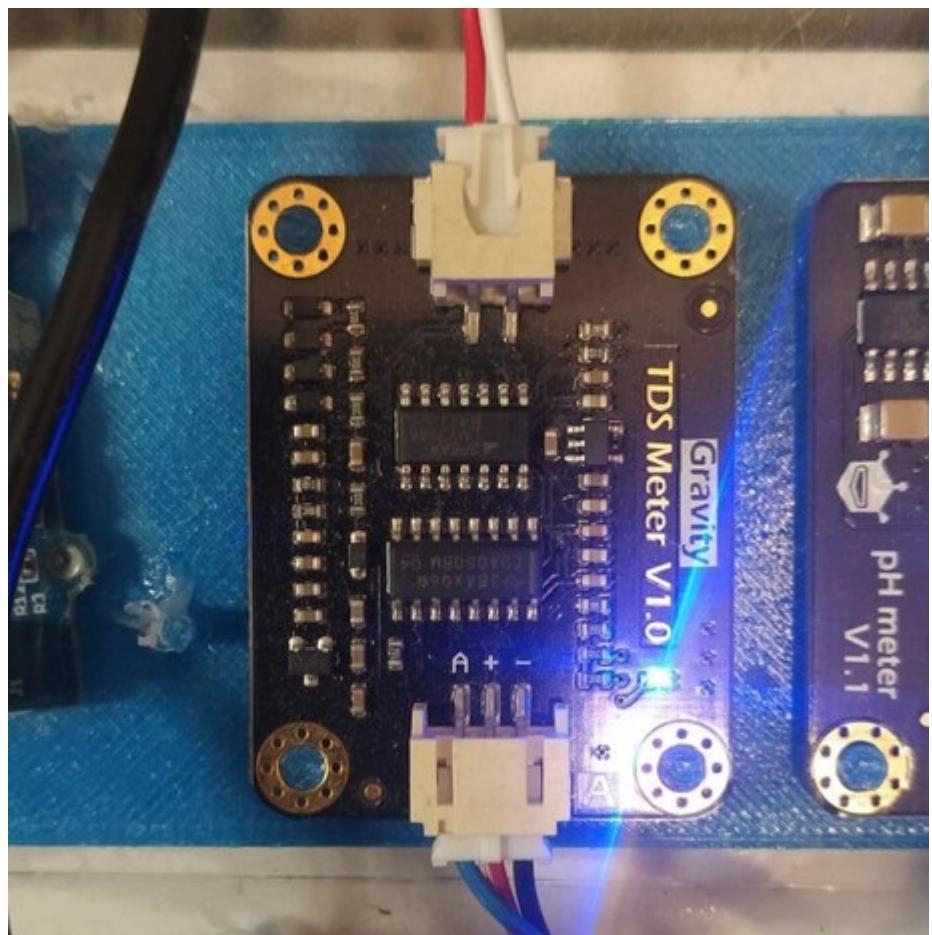
Rys. 5.31. model uchwytu pomp (wersja druga)

### *5.3.2. Kontrola poziomu soli mineralnych*

Do pomiaru zawartości soli mineralnych w wodzie wykorzystany został analogowy czujnik czystości wody firmy DFRobot (rys. 5.32). Czujnik podobnie jak czujnik pH składa się z sondy pomiарowej (rys. 5.34) i modułu przeliczającego wartości odczytane z sondy na wartości analogowe między 0V a 5V (rys. 5.33). Sonda zanurzona jest w wodzie obok sondy pH.



Rys. 5.32. Wszystkie elementy czujnika TDS



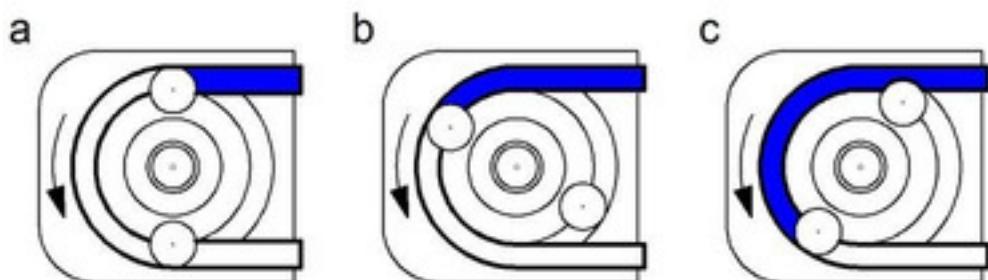
Rys. 5.33. Moduł czujnika TDS)



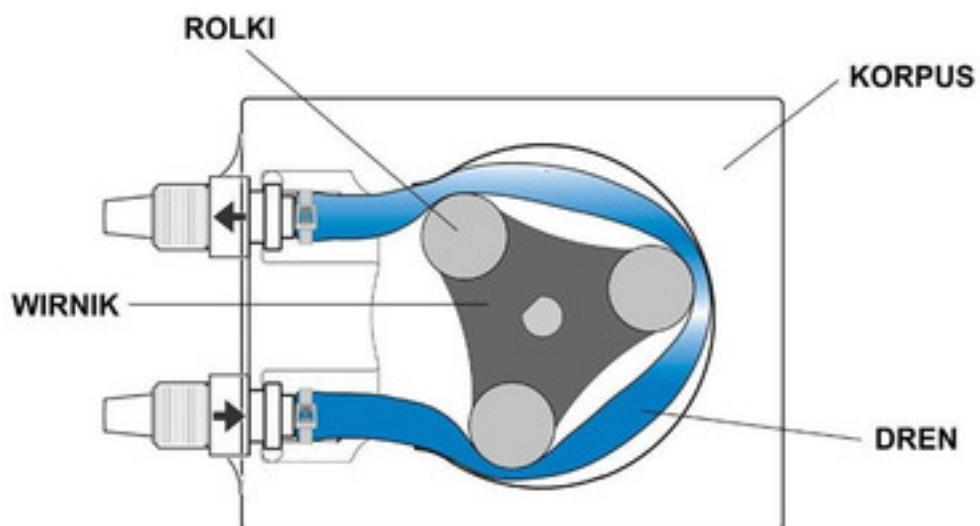
Rys. 5.34. Sonda czujnika TDS

Miarą czystości wody jest TDS (Total Dissolved Solids), a jednostką PPM (Parts Per Milion), która oznacza ilość jednostek masy na milion jednostek masy całkowitej. Wykorzystując miarę czystości wody można określić stężenie nawozów w wodzie, ponieważ są to sole mineralne. Poziom soli mineralnych trzeba utrzymywać na pewnym poziomie (w zależności od fazy rozwoju rośliny). Poziom zmniejsza się samoistnie, ponieważ rośliny wchłaniają nawozy zawarte w wodzie.

Aby zwiększyć ilość soli mineralnych w wodzie należy dodać do niej nawozu. Nawóz dozowany jest za pomocą pomp perystaltycznych. "W pompie perystaltycznej ciecz przepychana jest przez elastyczny przewód za pomocą karbowanego paska, karbowanego wałka, toczących się rolek lub wirnika z krzywkami przesuwającego się po przewodzie." "Pompa perystaltyczna – Wikipedia, wolna encyklopedia" [5]

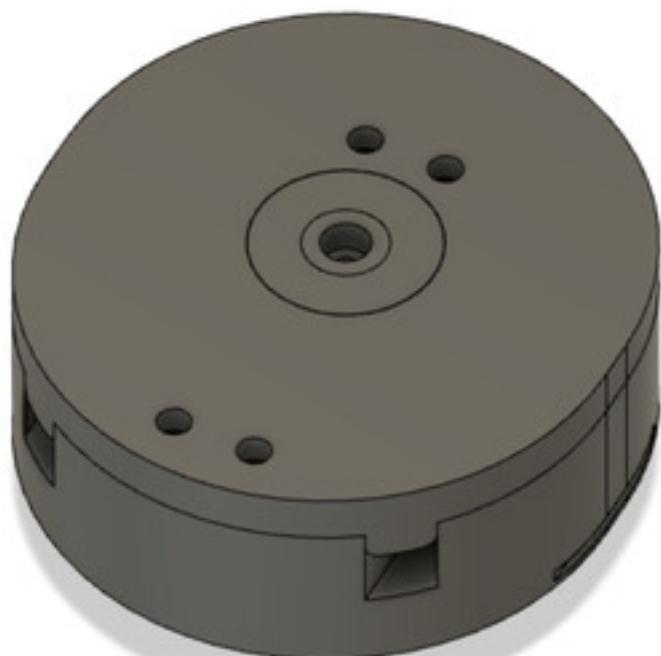


Rys. 5.35. Zasada pracy pompy perystaltycznej (zdjęcie pochodzi ze strony: abc-rc.pl)

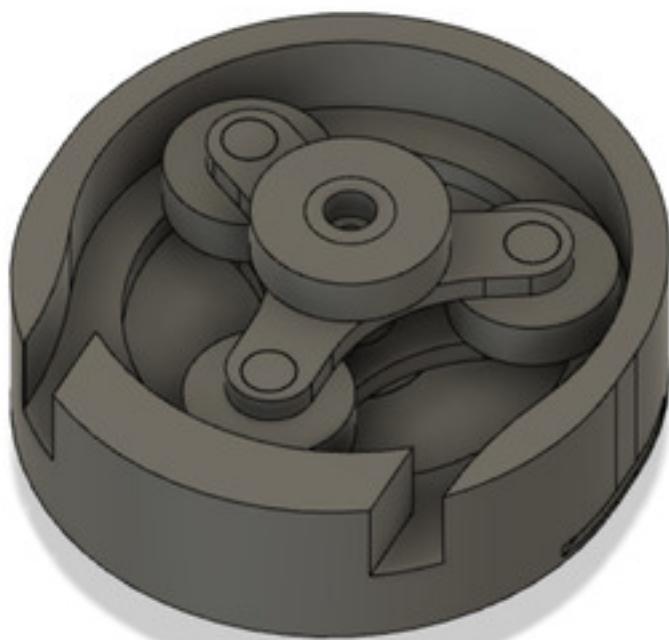


Rys. 5.36. Konstrukcja pompy perystaltycznej (zdjęcie pochodzi ze strony: abc-rc.pl)

Ten rodzaj pomp pozwala na dość dużą dokładność dozowania (trocę mniejszą niż w przypadku pompy strzykawkowej), a zarazem może czerpać płyn z zewnętrznego pojemnika. Pozwala to, w odróżnieniu od pomp strzykawkowych, uniezależnić się od pojemności pompy (strzykawki). Pierwsza wersja pomp perystaltycznych została wydrukowana na drukarce 3D. Konstrukcja przedstawiona jest poniżej.



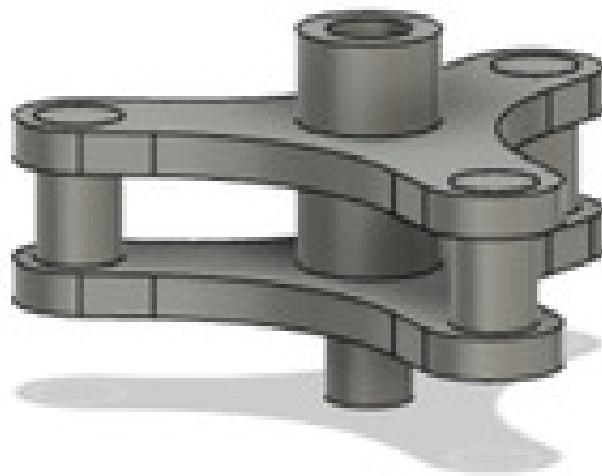
Rys. 5.37. Model pompy perystaltycznej (cały)



Rys. 5.38. Model pompy perystaltycznej (bez pokrywy)



Rys. 5.39. Model pompy perystaltycznej (element wirujący z rolkami)



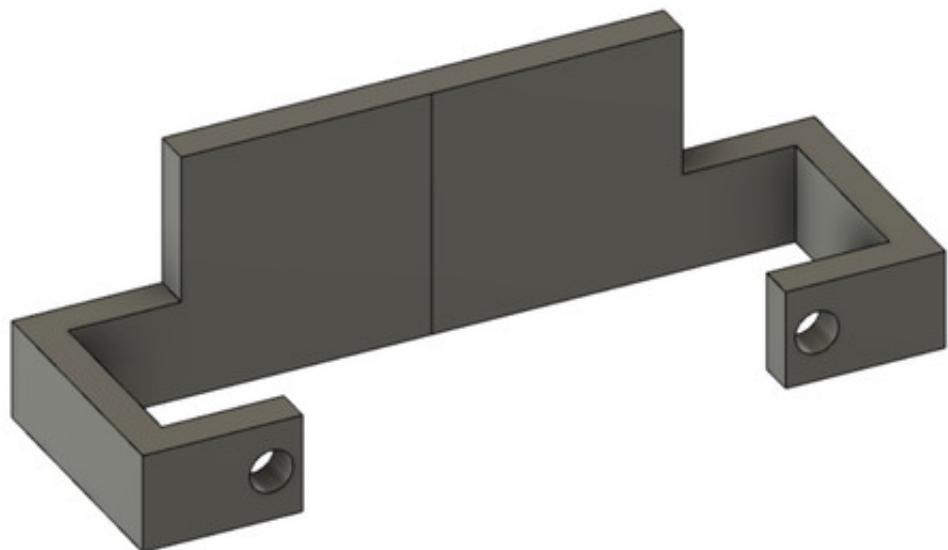
Rys. 5.40. Model pompy perystaltycznej (element wirujący bez rolek)

Pompy wyżej opisane oprócz elementów drukowanych na drukarce 3D składały się z silikonowego węzyka przez, który tłoczony był plyn, stalowych łożysk, które pełniły funkcję rolek oraz silnika ze stalową przekładnią. W tym przypadku rolę silnika z przekładnią pełnił seromechanizm Tower Pro MG-995 z usuniętą elektroniką, potencjometrem oraz blokadą obrotu powyżej 180°.

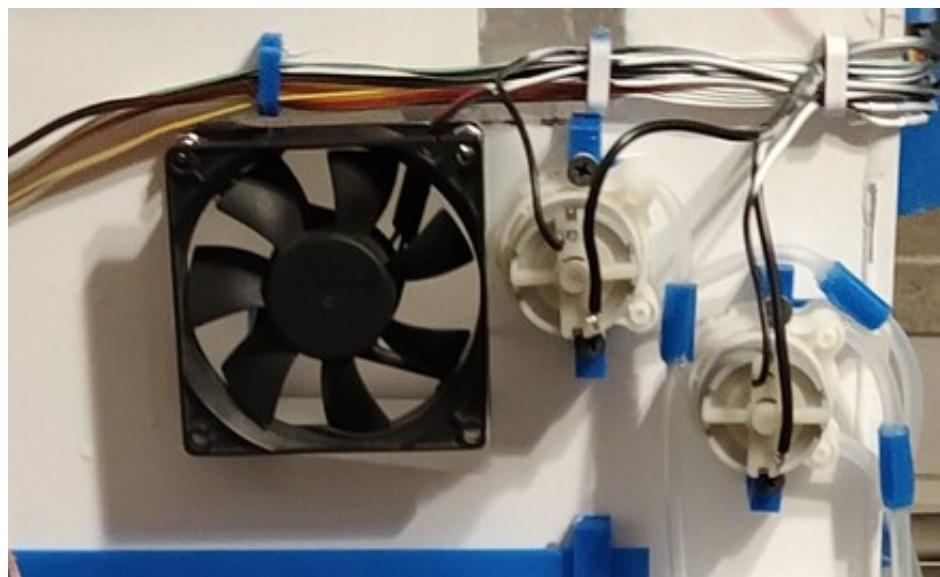


Rys. 5.41. Pompa perystaltyczna (jedna z pierwszych wersji)

Niestety okazało się, że takie pompy czasem się zacinają, a to wymaga fizycznej interwencji. Dlatego ostatecznie zostały one zastąpione gotowymi pompami perystaltycznymi. Zamocowane zostały za pomocą podstawek wydrukowanych z wykorzystaniem technologii 3D. Podstawka jest przyklejona na stałe do urządzenia, a pompa przykręcana do podstawki. Umożliwia to odkręcenie pompy w razie potrzeby, a jednocześnie niweluje potrzebę wierczenia kolejnych otworów w "akwarium".



Rys. 5.42. Model podstawki pompy perystaltycznej



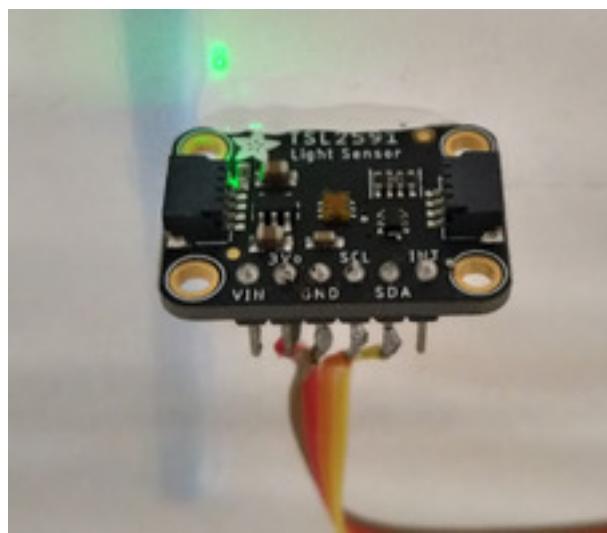
Rys. 5.43. Pompy perystaltyczne zamontowane na urządzeniu



Rys. 5.44. Pompy perystaltyczne z podstawkami

#### 5.4. Kontrola oświetlenia

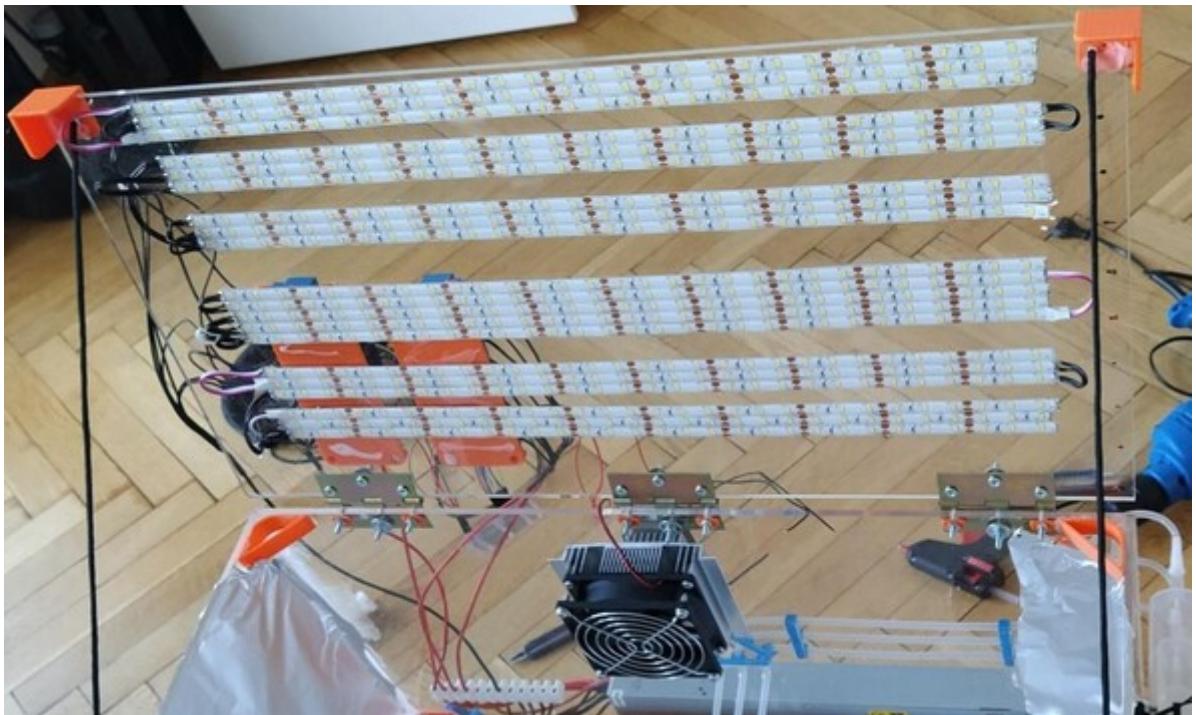
Do pomiaru oświetlenia służy czujnik TSL 2591 (rys. 5.45). Miarą, która jest kontrolowana, jest tu strumień natężenia światła, a jego jednostką jest lux.



Rys. 5.45. Czujnik natężenia światła (TSL 2591)

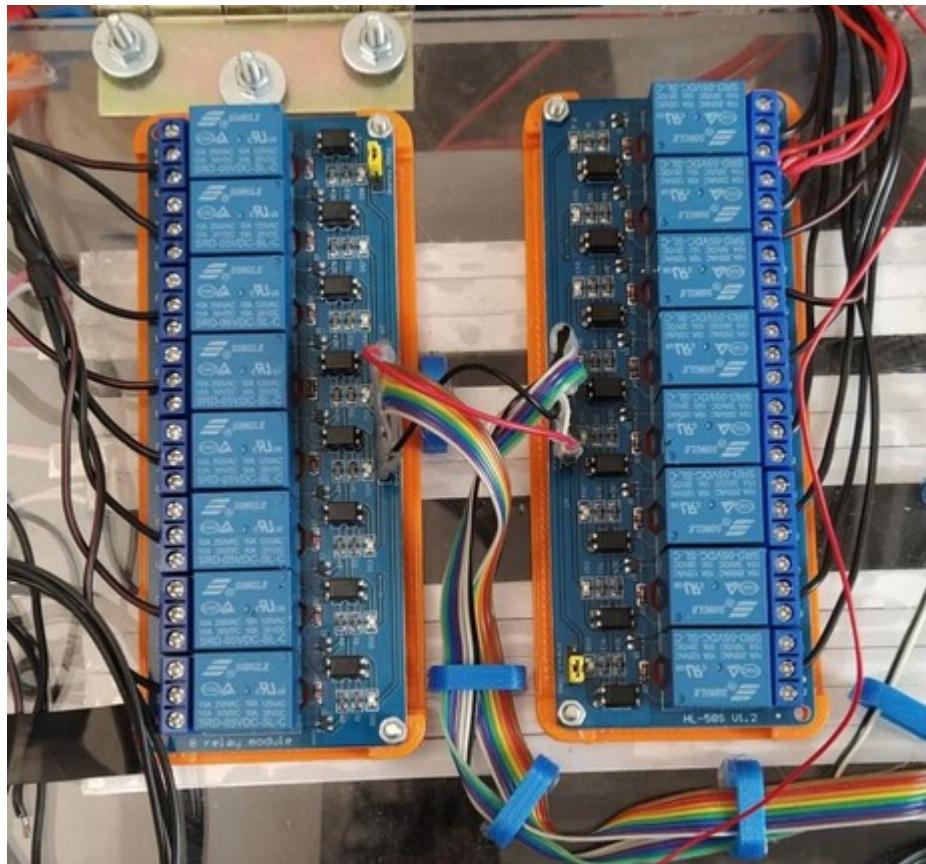
Źródłem światła są paski diod LED. Dokładniej mówiąc jest ich 20, każdy o długości 50 cm i zawierający 30 diod. Paski są podzielone na 6 segmentów rys.5.46. Każdy z nich zasilany jest

przez oddzielny przekaźnik o maksymalnym obciążeniu 20 A. Dzięki temu każdy segment może być oddzielnie włączony.



Rys. 5.46. Paski LED na pokrywie urządzenia

Na pokrywie urządzenia znajdują się 2 moduły przekaźników po 8 przekaźników każdy. Sześć z tych przekaźników steruje oświetleniem (1 na każdy segment).



Rys. 5.47. Moduły przekaźników na pokrywie urządzenia

#### 5.4.1. Kolor urządzenia

Kolor urządzenia nie jest przypadkowy. W celu zminimalizowania rozproszenia światła diod na zewnątrz górna część musiała zostać pokryta czymś, co będzie odbijać światło, a przez to energia świetlna zostanie zatrzymana wewnętrz. Dolna część "akwarium" (zbiornik wodny) powinna otrzymywać jak najmniejsze ilości światła by zapobiec rozwijaniu się w wodzie glonów. Została ona pokryta czarną farbą. Czarna powłoka na zbiorniku wodnym zapobiega przedostawaniu się promieni świetlnych do wewnętrz, a małe ilości energii świetlnej, które wedrą się mimo wszystko do środka szybko zostają przez farbę pochłonięte. Górną część urządzenia na początku pokryta została folią aluminiową, ponieważ doskonale odbija ona światło. Niestety okazało się, że minusem folii aluminiowej jest jej niska wytrzymałość. W folii zaczęły pojawiać się dziury, a ponadto pojawiły się również obawy o potencjalne zwarcia, które folia aluminiowa mogła by wywołać z powodu swojej przewodności.



Rys. 5.48. Wygląd urządzenia (wersja pierwsza)



Rys. 5.49. Płyta środkowa (wersja pierwsza)

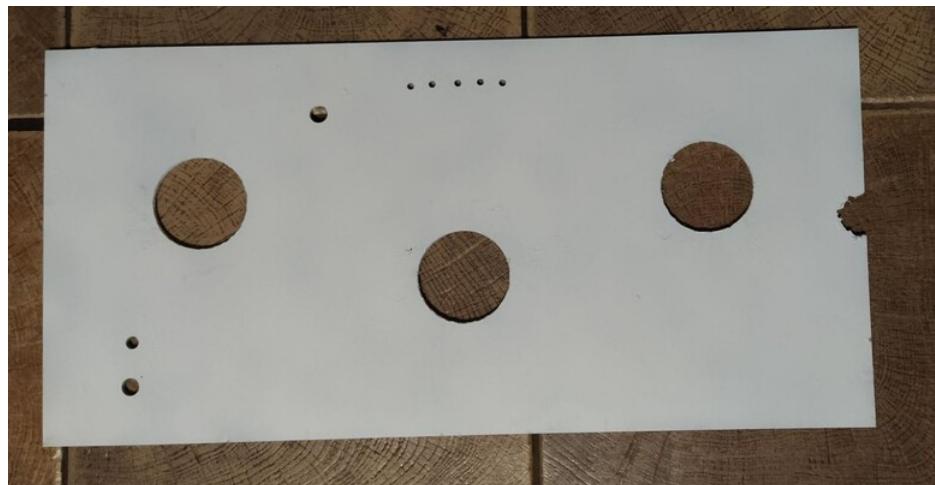


Rys. 5.50. Uszkodzenie folii

Zważywszy na powyższe powody folia została zastąpiona białą farbą. Nowe rozwiążanie wciąż zatrzymuje większość energii świetlnej wewnętrz urządzienia, a przy okazji jest pozbawione dwóch głównych wad folii aluminiowej, czyli przewodności i niskiej wytrzymałości.



Rys. 5.51. Urządzenie chwilę po malowaniu (wersja druga)

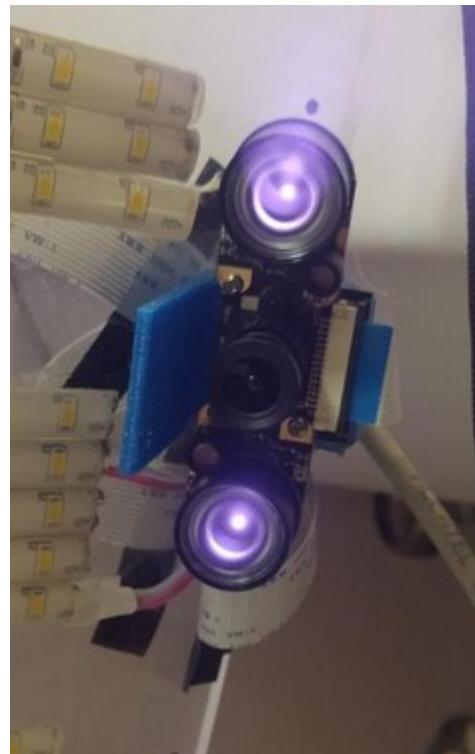


Rys. 5.52. Płyta środkowa (wersja druga)

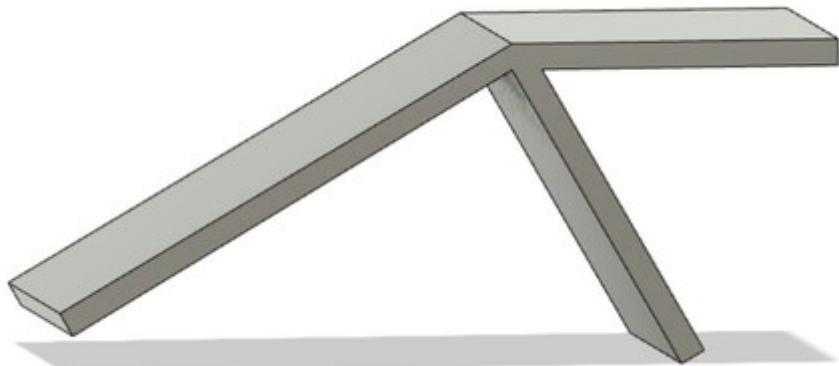
## 5.5. Pozostałe elementy budowy

### 5.5.1. Kamera

Dodatkowym źródłem danych o przebiegu hodowli jest kamera (rys. 5.53). Dzięki niej użytkownik może obserwować sytuację panującą wewnętrz urządzienia w dowolnej chwili. Jest to kamera dedykowana do komputera Raspberry Pi. Posiada ona dodatkowo dwie diody podczerwieni z reflektorami. Dzięki temu obserwację jak i zbieranie danych z kamery można prowadzić zarówno podczas fazy dziennej i nocnej cyklu dobowego.



Rys. 5.53. Kamera zamontowana na podstawce (widać świecące diody podczerwieni)



Rys. 5.54. Model podstawki kamery

#### 5.5.2. Napowietrzanie wody

Aby zapobiec rozwojowi mikroorganizmów w zbiorniku wodnym, które mogłyby doprowadzić np. do gnicia korzeni roślin, woda musi być napowietrzona. Wykorzystany został do tego akwariowy napowietrzacz wody. Jest on cały czas włączony, aby zapewnić ciągłe napowietrzenie oraz cyrkulację wody.



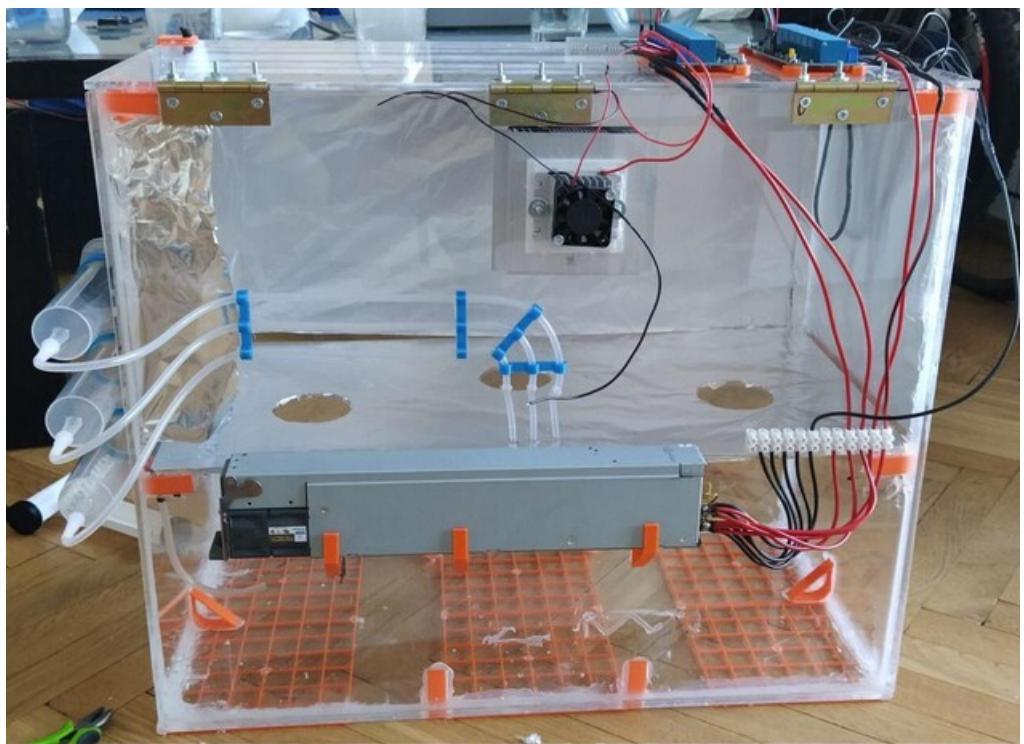
Rys. 5.55. Napowietrzacz wody (zdjęcie pochodzi ze strony allegro.pl)

### 5.5.3. Zasilanie

Jako źródło zasilania wykorzystany został zasilacz serwerowy o mocy 500W. Posiada on dwa napięcia wyjściowe:

1. 12V zasilające:
  - a) oświetlenie
  - b) wentylację
  - c) chłodzenie
2. 5V zasilające:
  - a) Raspberry Pi
  - b) Arduino
  - c) wszystkie czujniki
  - d) wszystkie pompy
  - e) logiczną część modułów przekaźników

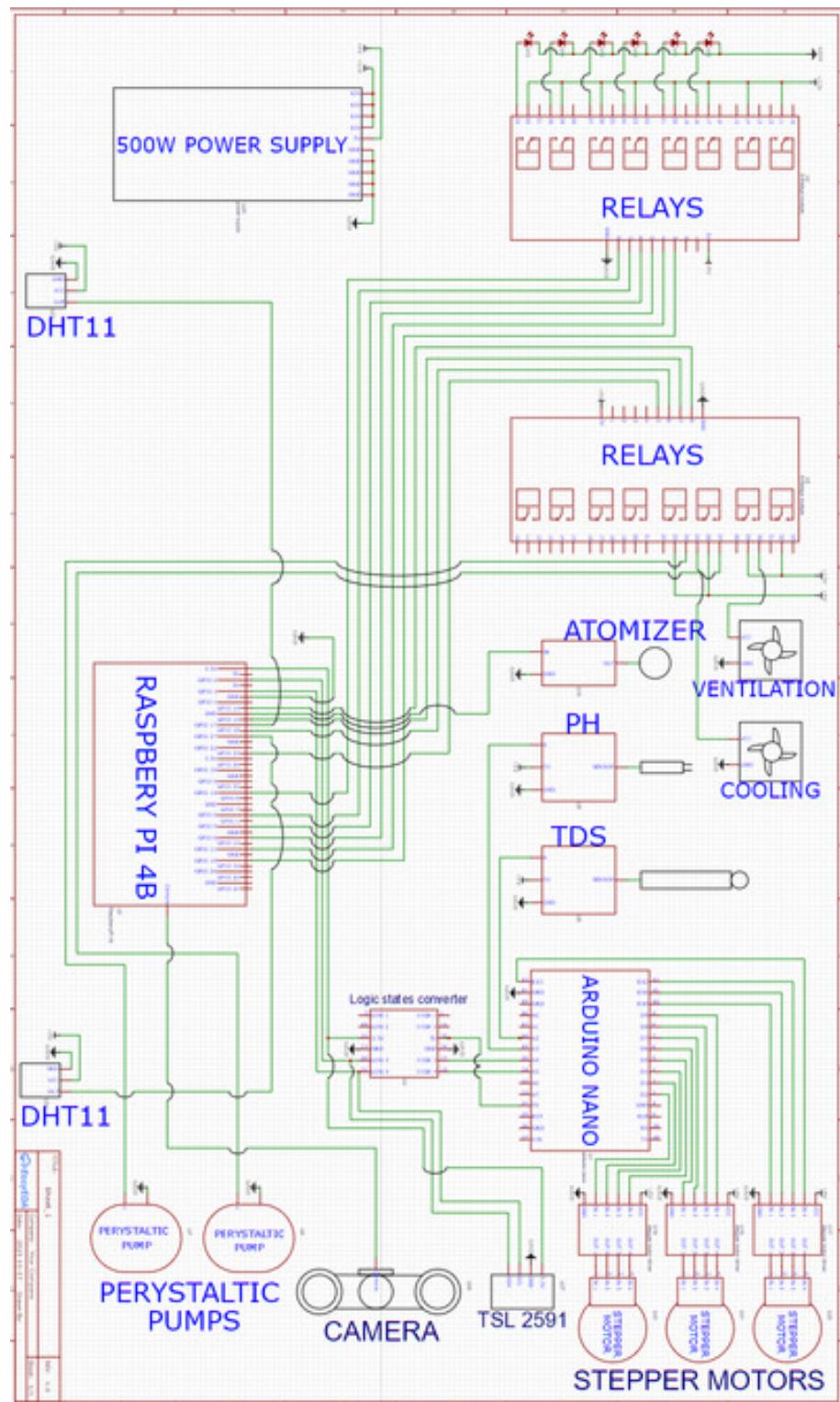
Zasilacz zamontowany jest na tylnej ścianie urządzenia za pomocą podstawek wydrukowanych z wykorzystaniem technologii 3D. Umiejscowiony jest on powyżej tafli wody, aby zmniejszyć ryzyko zwarcia w razie wycieku (rys. 5.56).



Rys. 5.56. Zasilacz zamontowany na urządzeniu

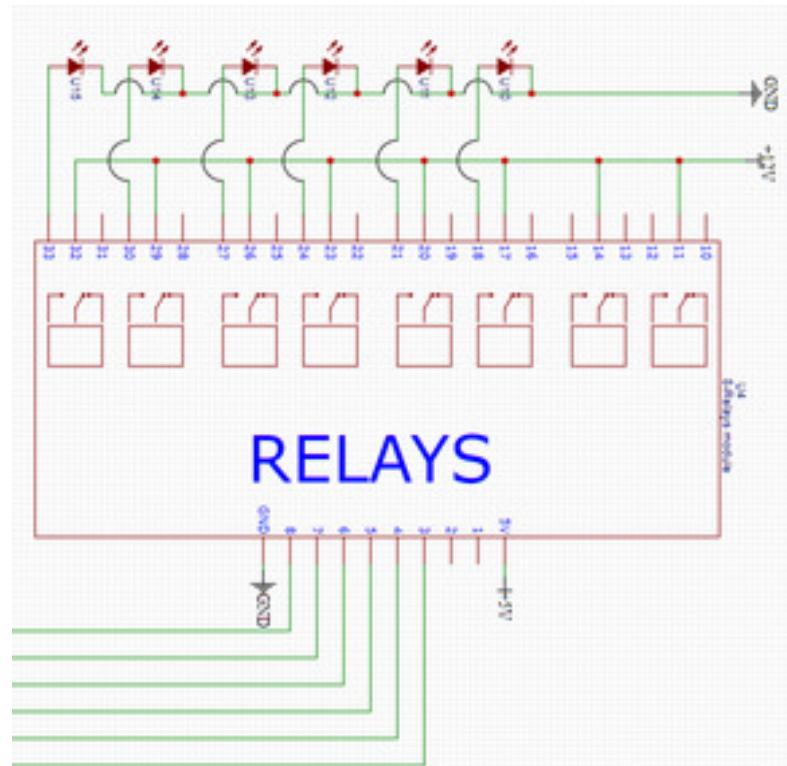
## 5.6. Elektronika

W tej sekcji przedstawiono schemat elektryczny urządzenia.



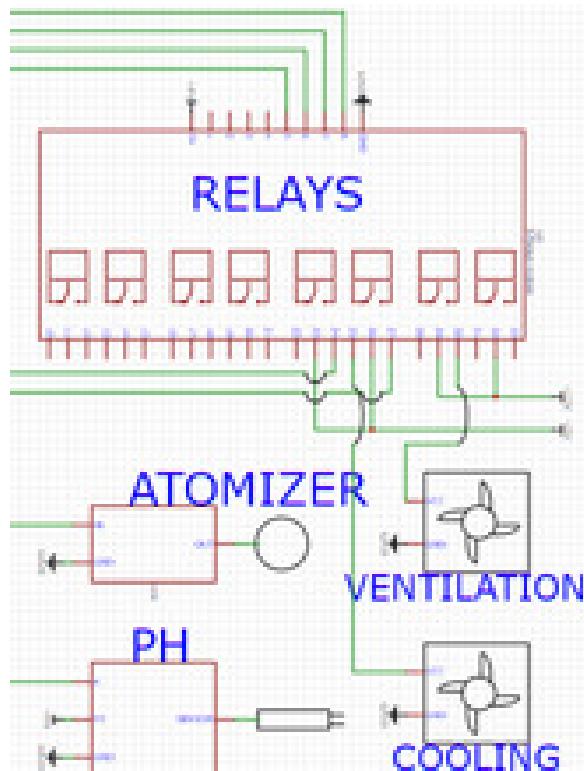
Rys. 5.57. Pełny schemat elektryczny urządzenia

Ilustracja 5.57 przedstawia pełen schemat.



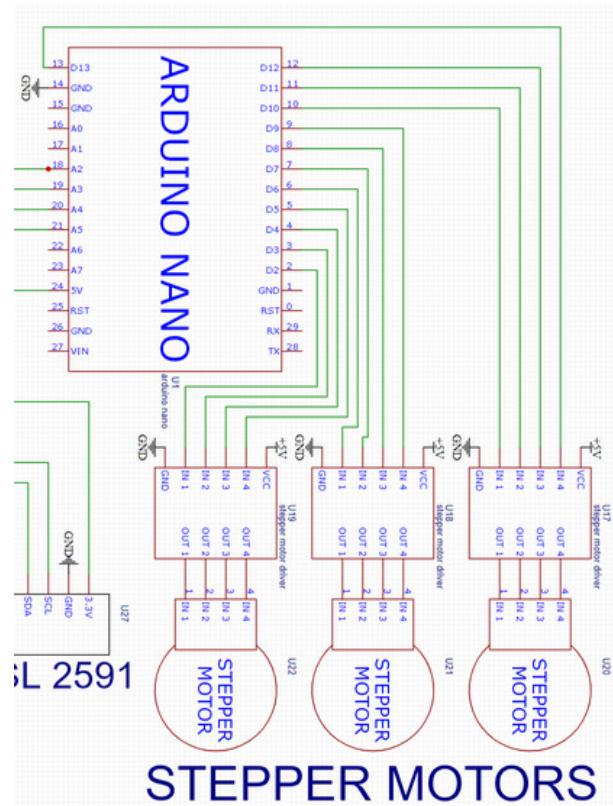
Rys. 5.58. Zbliżenie na moduł przekaźników sterujących światłami (każdy symbol diody oznacza cały segment pasków LED)

Ilustracja 5.58 ukazuje moduł przekaźników sterujący światłami. Dla uproszczenia schematu każdy z segmentów pasków LED reprezentowany jest przez pojedynczy symbol diody.



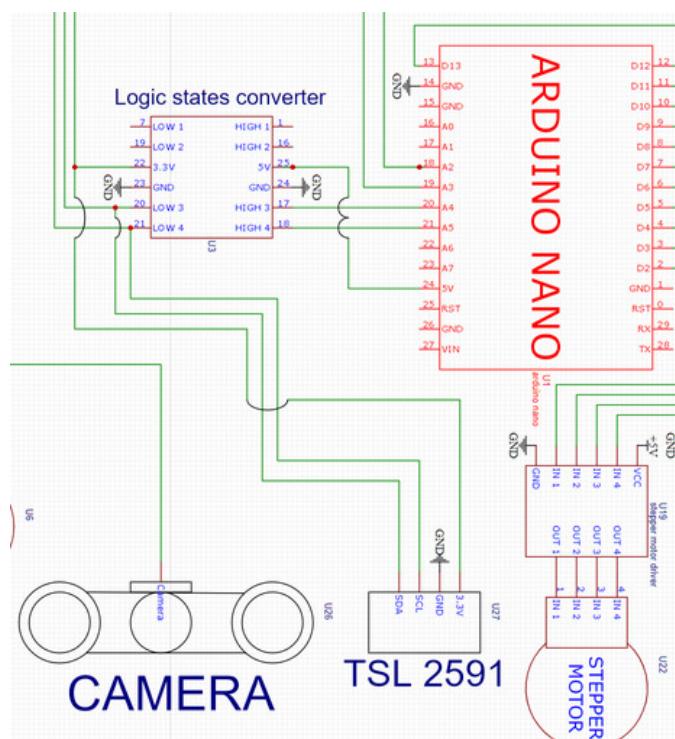
Rys. 5.59. Zbliżenie na drugi moduł przekaźników

Ilustracja 5.59 przedstawia drugi z modułów przekaźników. Steruje on układem chłodzenia, wentylacją oraz pompami perystaltycznymi.



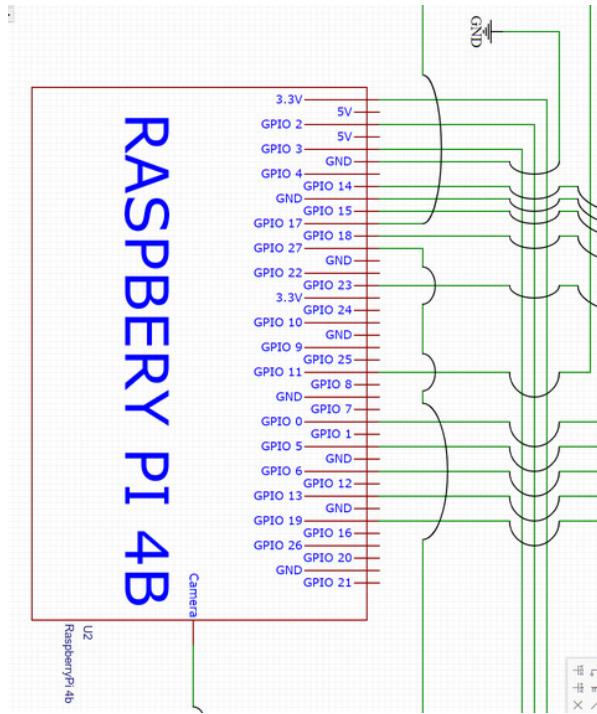
Rys. 5.60. Zbliżenie na Arduino i pompy strzykawkowe

Kolejny element schematu (rys. 5.60) prezentuje połączenie pomp perystaltycznych do mikro kontrolera Arduino.



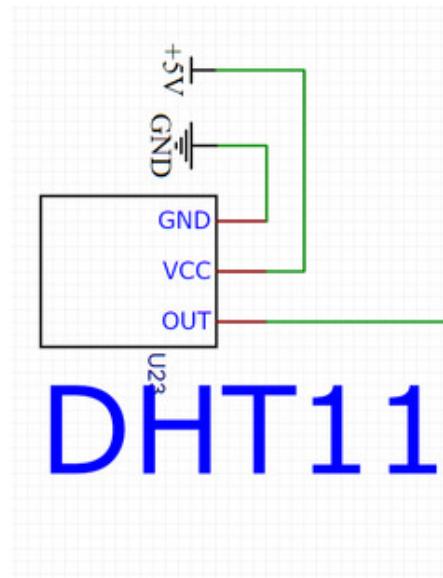
Rys. 5.61. Zbliżenie na konwerter stanów logicznych, kamerę i czujnik natężenia światła (TSL2591)

Ilustracja 5.61 uwidocznia sposób podłączenia Arduino oraz czujnika TSL 2591 do konwertera stanów logicznych. Konwerter ten umożliwia komunikację I2C między podzespołami pracującymi pod różnymi napięciami. Procesor mikrokontrolera Arduino pracuje pod napięciem 5V natomiast procesor Raspberry Pi jak i czujnik TSL 2591 operują napięciem 3.3V.



Rys. 5.62. Zbliżenie na Raspberry Pi

Ilustracja 5.62 przedstawia piny komputera Raspberry Pi oraz doprowadzone do nich połączenia.

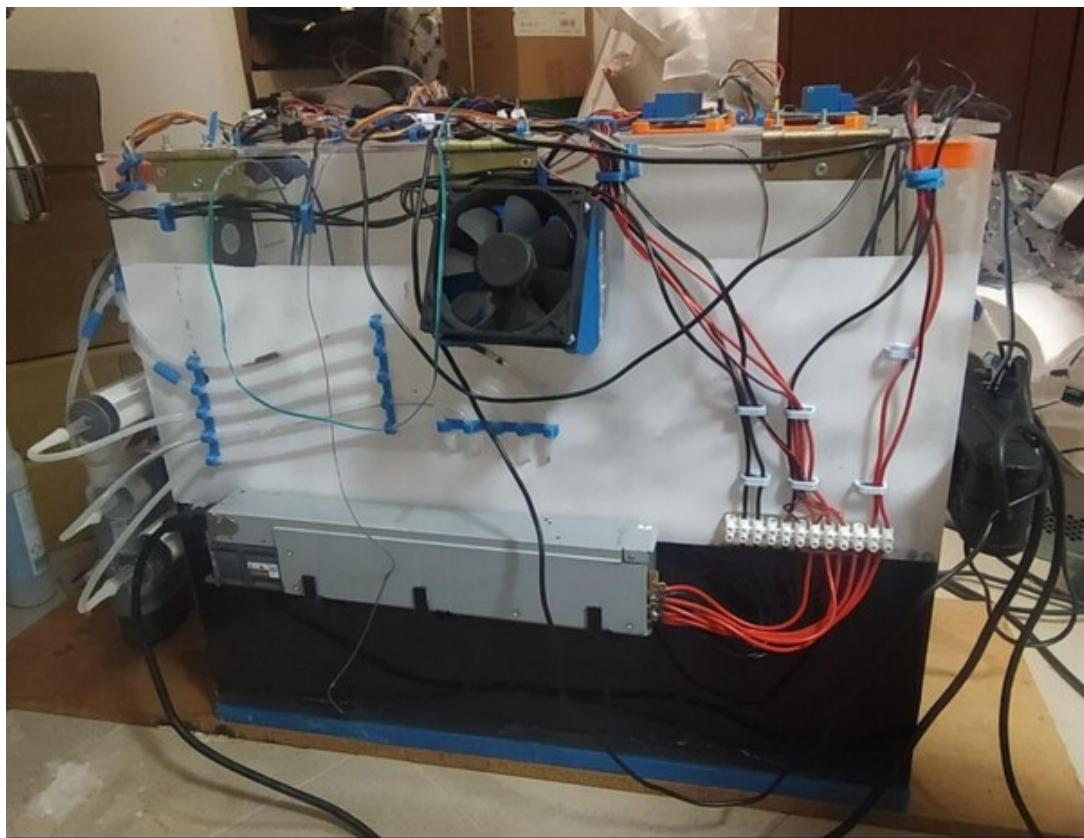


Rys. 5.63. Zbliżenie na czujnik DHT11

Poniższy fragment schematu (rys. 5.63) przedstawia sposób połączenia czujnika temperatury i wilgotności (DHT11).

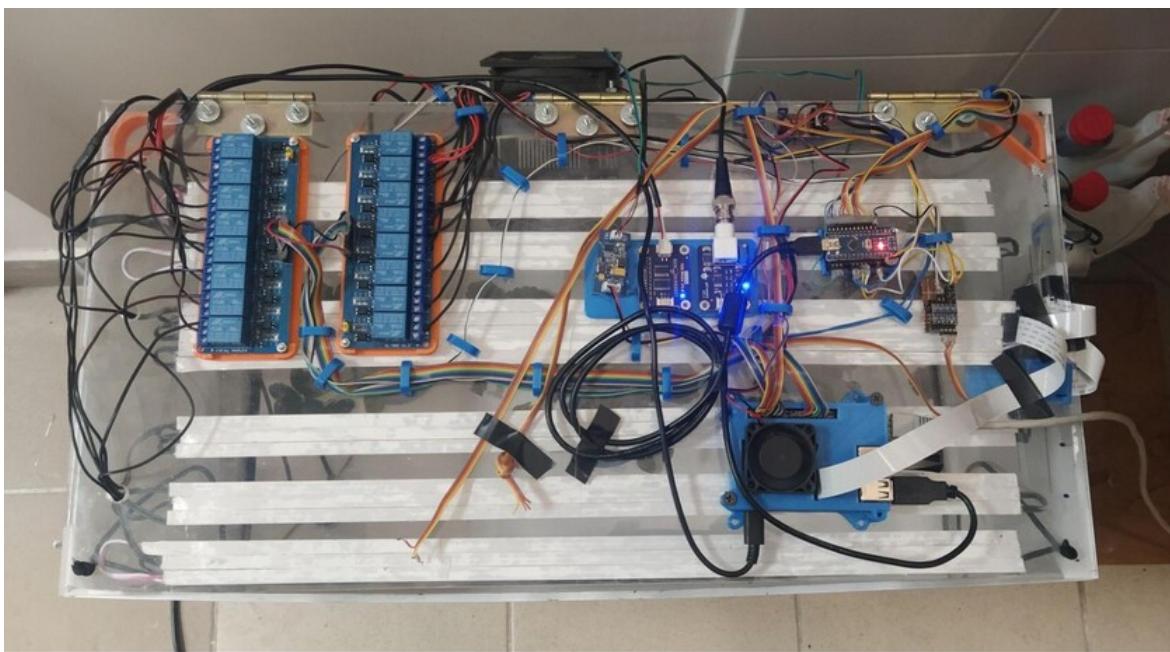
## **5.7. Końcowa prezentacja urządzenia**

Ta sekcja poświęcona została prezentacji prototypu.



Rys. 5.64. Tylna ściana urządzenia

Na zdjęciu 5.64 widać tylną ścianę urządzenia. Znajduje się na niej zasilacz, wentylator gorącej strony układu chłodzenia oraz rurki przez które nawozy i odczynniki pH dozowane są do wody.



Rys. 5.65. Pokrywa urządzenia

Zdjęcie 5.65 przedstawia pokrywę prototypu na której umieszczona jest większa część elementów elektronicznych. Na spodniej części pokrywy przyjmocowane są paski LED.



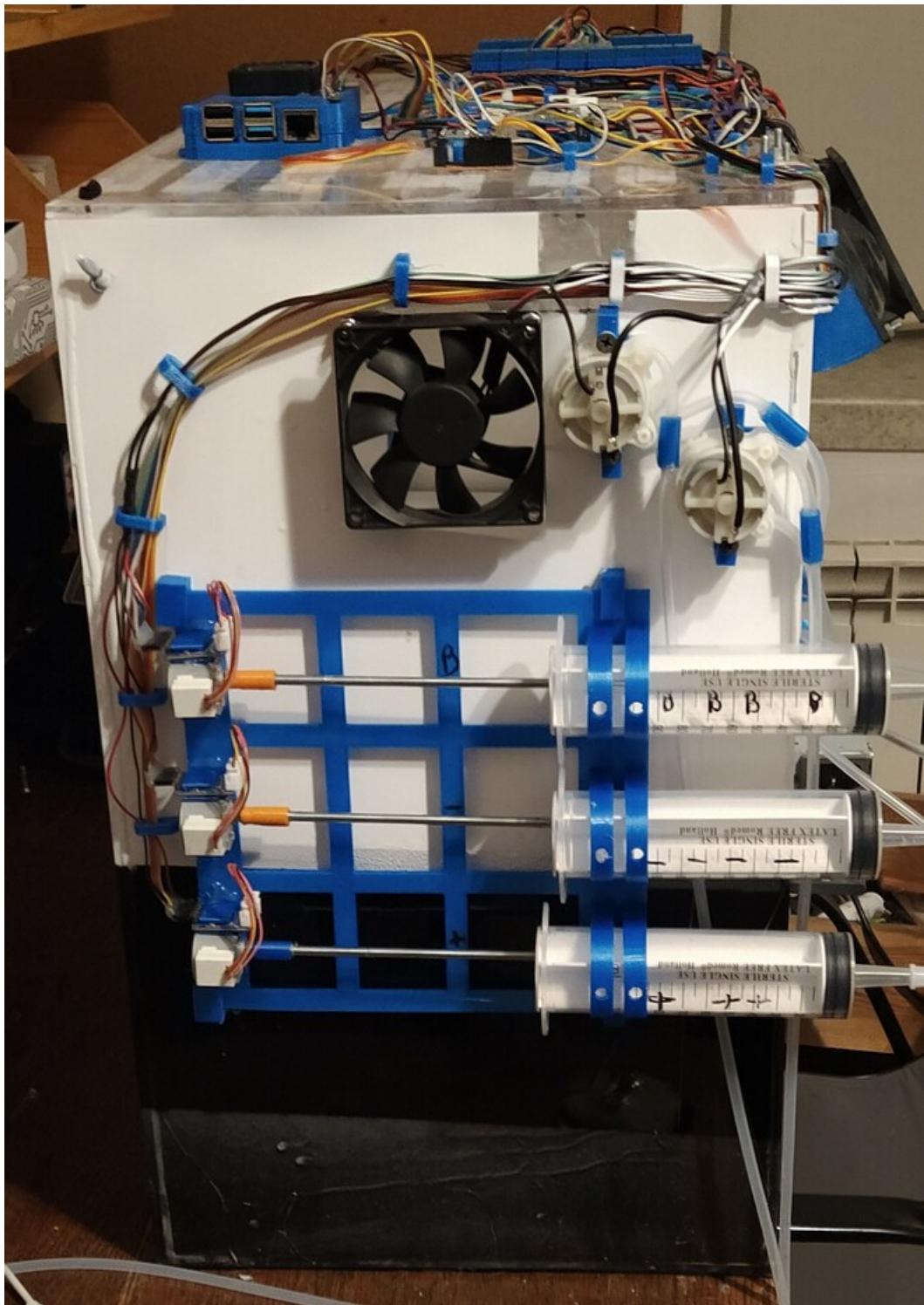
Rys. 5.66. Przód i góra urządzenia

Na zdjęciu 5.66 widać przednią ścianę urządzenia. Najlepiej uwidocznia ona warstwę czarnej farby pochłaniającej światło oraz farbę białą która zatrzymuje światło wewnętrz urządzienia.



Rys. 5.67. Lewa ściana urządzenia

Zdjęcie 5.67 przedstawia lewą ścianę urządzenia oraz zamontowany na niej rozgałęziacz, do którego podłączony jest zasilacz główny, zasilacz Raspberry Pi oraz pompa napowietrzająca.



Rys. 5.68. Prawa ściana urządzenia

Na zdjęciu 5.68 widnieje prawa ściana urządzenia wraz z trzema pompami strzykawkowymi, dwiema pompami perystaltycznymi i wentylatorem.

## 6. PREZENTACJA WARSTWY PROGRAMOWEJ

Autor rozdziału: Paulina Lis

Pierwszym etapem pracy nad oprogramowaniem było ustalenie architektury sprzętowej projektowanego rozwiązania, tak aby zdefiniować możliwości sprzętowe i rozpocząć prace nad oprogramowaniem. W tabeli 6.1 zaprezentowano wykaz wejść/wyjść z warstwy sprzętowej. Do komunikacji z warstwą sprzętową wykorzystano magistrale I2C, GPIO, a do komunikacji z nimi biblioteki RPi.GPIO i smbus po stronie Raspberry Pi oraz Wire.h po stronie Arduino.

### 6.1. PIN-y GPIO

Numer PIN-u GPIO	Sensor/Aktuator	Opis zastosowania
0, 5, 6, 11, 13, 19	Przekaźniki segmentów LED	Regulacja poziomu naświetlenia
4	Atomizer	Zwiększenie wilgotności powietrza
14	Przekaźnik układu chłodzenia	Obniżenie temperatury
15	Przekaźnik wentylatora	Zmniejszenie poziomu wilgotności
17, 27	Czujniki DHT11	Odczytywanie temperatury i wilgotności
18, 23	Przekaźniki pomp perystaltycznych	Dozowanie nawozu

Tabela 6.1. Wykaz wejść/wyjść warstwy sprzętowej

Natomiast w tabela6.2 przedstawiono adresy urządzeń połączonych z Raspberry Pi za pomocą magistrali I2C.

Adres urządzenia	Sensor/Aktuator	Opis zastosowania
0x29	Czujnik TSL2591	Pomiar natężenia światła
0x07	Arduino	- Dozowanie odczynników pH+/pH- - Odczyt poziomu pH wody - Odczyt TDS wody

Tabela 6.2. Urządzenia połączone z Raspberry Pi za pomocą magistrali I2C

## 6.2. Opis użytych bibliotek

Przy pracy nad oprogramowaniem niezbędne było zgłębienie wiedzy z zakresu działania poszczególnych komponentów oraz sprawdzenie istniejących już rozwiązań, takich jak gotowe biblioteki, zapoznanie się ze sposobem ich działania oraz rozważenie zasadności zastosowania ich w pracy.

- adafruit-circuitpython-dht - biblioteka oparta o CircuitPython, który wspiera czujniki DHT11 służący do pomiaru temperatury i wilgotności oraz czujnik TSL2591 używany do pomiaru poziomu naświetlenia. CircuitPython to otwarta wersja Python dla mikrokontrolerów. Zaimplementowanie wbudowanego modułu "board" umożliwia dostęp do pinów.
- RPi.GPIO - biblioteka do sterowania pinami GPIO na Raspberry Pi. Umożliwia dwa tryby odwoływać się do pinów:
  - GPIO.BOARD - polega na odwoywaniu się bezpośrednim do numeru pinu w złączu. Tryb ten jest uniewersalny dla wszystkich modeli Raspberry Pi.
  - GPIO.BCM - umożliwia odwoływanie się do pinów GPIO, które są bezpośrednio podłączone do układu SoC na Raspberry Pi. W napisanych skryptach wykorzystano ten tryb.
- PiCamera - biblioteka udostępnia interfejs do modułu kamery Raspberry Pi.
- smbus - podzbiór protokołu I2C. Służy do komunikacji między układami.
- http.server - moduł, który definiuje klasy do implementacji serwerów HTTP.
- psutil - biblioteka służąca do uzyskiwania szczegółów odnośnie systemu oraz narzędzi procesowych. Można dzięki niej monitorować wykorzystanie zasobów takich jak pamięć, procesor, czujniki. Wykorzystuje się ją też zarządzania uruchomionymi procesami.
- json - pakiet, który pozwala na pracę z danymi JSON. JSON jest to składnia do przechowywania i wymiany danych.

## 6.3. Skrypty dla poszczególnych komponentów

W celu umożliwienia użytkownikowi odczytywania wartości z czujników, a także na potrzeby monitorowania oraz dostosowania warunków w akwariu, napisano skrypty listing 6.1 dla czujników DHT11 oraz listing 6.2 dla TDS oraz pH.

```
1 import time
2
3 import adafruit_dht
4 import board
5
6 # Initial the dht device, with data pin connected to pin17
7 dht_device = adafruit_dht.DHT11(board.D17)
8 while(True):
9     try:
10         temperature = dht_device.temperature
11         print(f"Temperature: {temperature}")
12         dht_device.exit()
13         break
14     except RuntimeError as error:
15         print(error.args[0])
16         time.sleep(1.5)
17         continue
18
19     except Exception as error:
```

```

21     dht_device.exit()
22     raise error

```

Listing 6.1: Skrypt do odczytywania wartości temperatury przy użyciu czujnika DHT11

```

1 from smbus import SMBus
2 import time
3
4 addr = 0x7 #arduino nano adress
5 bus =SMBus(1)
6
7 tds=bus.read_byte(addr)
8 print(tds)
9 bus.write_byte(addr,5)
10 time.sleep(1)
11 ph=bus.read_byte(addr)
12 print(ph)

```

Listing 6.2: Skrypt do odczytywania wartości TDS oraz pH

W celu sprawdzenia działania systemu z początkowymi założeniami, skrypty dla wszystkich komponentów zaimplementowano w pętli wraz z licznymi warunkami. W następnej kolejności została dodana możliwość zbierania danych z poszczególnych czujników w celu monitorowania pracy akwarium.

Po przeprowadzeniu testów napisanego rozwiązania, kod został zoptymalizowany poprzez podział na moduły, klasy oraz ich metody.

#### 6.4. Serwer Raspberry Pi

Serwer Raspberry Pi zawiera dwie klasy: Hydroponics oraz HydroponicsHandler.

Na początku klasy Hydroponics przypisane zostały niezbędne stałe wartości oraz lista zawierająca numery PIN-ów GPIO wykorzystywanych przez przekaźnik segmentów LED (listing 6.3).

```

1 class Hydroponics:
2     cooling_pin = 14
3     fan_pin = 15
4     atomizer_pin = 4
5     fertilizer_ml_per_second = 1.83
6     loop_delay = 10
7     fertilizer_delay = 60
8     ph_delay = 120
9     exceptions_attempts_count = 10
10    ph_plus_pump_num = 2
11    ph_minus_pump_num = 1
12    booster_pump_num = 3
13    fertilizer_a_pump_pin = 18
14    fertilizer_b_pump_pin = 23
15
16    lights_list = [0, 5, 6, 11, 13, 19]

```

Listing 6.3: Klasa Hydroponics

Następnie w części `__init__` (listing 6.4) zainicjalizowano niezbędne klasy oraz ustawiono stan PIN-ów GPIO.

```

1 def __init__(self):
2     self.sensor_light = LightSensor()
3     self.sensor_dht = DhtSensor()
4     self.tds_sensor = TdsSensor()
5     self.ph_sensor = PhSensor()
6     self.cooling = Module(self.cooling_pin)
7     self.fan = Module(self.fan_pin)
8     self.atomizer = Module(self.atomizer_pin, on_state='HIGH')
9     self.light_module = LightModule(self.lights_list)
10    self.logger = Logger()
11    last_log = self.logger.getLastLog()
12    if not last_log:
13        print('log file not found')
14        self.nextDay()
15        self.waterSetup()
16    else:
17        print('log file found')
18        self.day_of_phase = last_log['day_of_phase']
19        self.phase = last_log['phase']
20        self.fertilizer_pump_a = PeristalticPump(self.fertilizer_a_pump_pin)
21        self.fertilizer_pump_b = PeristalticPump(self.fertilizer_b_pump_pin)
22        self.ph_plus_pump = SyringePump(self.ph_plus_pump_num)
23        self.ph_minus_pump = SyringePump(self.ph_minus_pump_num)
24        self.sensors_indications = {
25            "ph": "None",
26            "tds": "None",
27            "light": "None",
28            "temperature": "None",
29            "humidity": "None"
30        }
31
32    GPIO.setmode(GPIO.BCM)
33    GPIO.setwarnings(False)
34
35    # Relay (unallocated)
36    GPIO.setup(9, GPIO.OUT)
37    GPIO.setup(10, GPIO.OUT)
38
39    # Off
40    GPIO.output(9, GPIO.HIGH)
41    GPIO.output(10, GPIO.HIGH)

```

Listing 6.4: Klasa Hydroponics-init

W klasie HydroponicsHandler zostały zdefiniowane dwie metody `do_GET` oraz `do_POST`. Korzystając z metody `do_GET` (listing 6.5) użytkownik może pozyskać informacje o danych rejestrówanych przez poszczególne sensory (odczyt wartości temperatury, wilgotności, pH, sumy substancji rozpuszczonych w wodzie, poziom naświetlenia). Ułatwia to również prowadzenie monitoringu hodowli z użyciem dodatkowych, zewnętrznych narzędzi.

```

1 def do_GET(self):
2     self.send_response(200)
3     self.send_header('content-type', 'text/html')
4     self.end_headers()
5     divided_path = self.path.split("/")
6     if divided_path[1] == "temperature":
7         temperature = hydroponics.sensor_dht.readTemperature()
8         self.wfile.write(str(temperature).encode())
9         hydroponics.sensors_indications['temperature'] = temperature
10    elif divided_path[1] == "humidity":
11        humidity = hydroponics.sensor_dht.readHumidity()
12        self.wfile.write(str(humidity).encode())
13        hydroponics.sensors_indications['humidity'] = humidity
14    elif divided_path[1] == "ph":

```

```

15     ph = hydroponics.ph_sensor.read()
16     self.wfile.write(str(ph).encode())
17     hydroponics.sensors_indications['ph'] = ph
18 elif divided_path[1] == "tds":
19     tds = hydroponics.tds_sensor.read()
20     self.wfile.write(str(tds).encode())
21     hydroponics.sensors_indications['tds'] = tds
22 elif divided_path[1] == "light":
23     light = hydroponics.sensor_light.read()
24     self.wfile.write(str(light).encode())
25     hydroponics.sensors_indications['light'] = light

```

Listing 6.5: Metoda do\_GET

Metoda do\_POST (listing 6.6) umożliwia zażądanie akcji od elementów wykonawczych takich jak pompy, układ chłodzenia czy wentylacja. Możemy zmienić fazę wzrostu w jakiej znajdują się rośliny, dostosować temperaturę, wilgotność, poziom naświetlenia lub podać potrzebne substancje odżywcze.

```

1 def do_POST(self):
2     self.send_response(200)
3     self.send_header('content-type', 'text/html')
4     self.end_headers()
5     divided_path = self.path.split("/")
6     content_length = int(self.headers['Content-Length'])
7     post_data = self.rfile.read(content_length)
8     data = post_data.decode('utf-8')
9     print(data)
10
11    if divided_path[1] == "takePhoto":
12        hydroponics.logger.takePhoto()
13        self.wfile.write("OK".encode())
14
15    elif divided_path[1].startswith("logData"):
16        if hydroponics.logger.log['phase'] != data and data != '':
17            hydroponics.logger.changePhase(data)
18        current_time = datetime.now()
19        if hydroponics.logger.getTimer().hour > current_time.hour:
20            hydroponics.logger.nextDay()
21        hydroponics.logger.logging(sensors_indications=hydroponics.sensors_indications,
22                                     print_only=False)
22        self.wfile.write("OK".encode())
23
24    elif divided_path[1].startswith("changeDayPhase"):
25        if data == 'day':
26            hydroponics.logger.day()
27        elif data == 'night':
28            hydroponics.logger.night()
29
30    elif divided_path[1].startswith("dose"):
31        if data == 'ph-':
32            hydroponics.ph_minus_pump.dosing(1)
33            self.wfile.write("OK".encode())
34        elif data == 'ph+':
35            hydroponics.ph_plus_pump.dosing(1)
36            self.wfile.write("OK".encode())
37        elif data == 'fertilizer':
38            hydroponics.fertilizer_pump_a.dosing(1)
39            hydroponics.fertilizer_pump_b.dosing(1)
40            self.wfile.write("OK".encode())
41
42    elif divided_path[1].startswith("temperature"):
43        if data == 'decrease':
44            hydroponics.cooling.switch(True)
45            hydroponics.fan.switch(True)
46            self.wfile.write("OK".encode())
47        elif data == 'increase':
48            hydroponics.cooling.switch(False)
49            hydroponics.fan.switch(False)
50            self.wfile.write("OK".encode())

```

```

51     elif data == 'remain':
52         hydroponics.fan.switch(False)
53         self.wfile.write("OK".encode())
54
55     elif divided_path[1].startswith('humidity'):
56         if data == 'decrease':
57             hydroponics.fan.switch(True)
58             hydroponics.atomizer.switch(False)
59             self.wfile.write("OK".encode())
60         elif data == 'increase':
61             hydroponics.atomizer.switch(True)
62             self.wfile.write("OK".encode())
63     elif data == 'remain':
64         hydroponics.atomizer.switch(False)
65         self.wfile.write("OK".encode())
66
67     elif divided_path[1].startswith("light"):
68         if data == 'on':
69             hydroponics.light_module.switch('ON')
70             self.wfile.write("OK".encode())
71         elif data == 'off':
72             hydroponics.light_module.switch('OFF')
73             self.wfile.write("OK".encode())

```

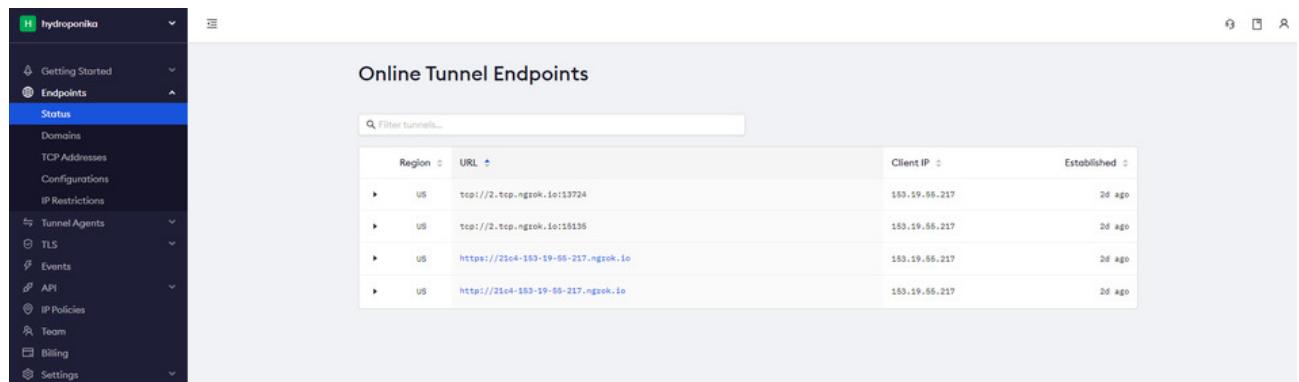
Listing 6.6: Metoda do\_POST

Pomiary temperatury i wilgotności dokonywane są za pomocą dwóch czujników DHT11 zamontowanych po dwóch stronach akwarium. Następnie wyciągana jest średnia z przekazanych pomiarów. Dzięki temu zmierzone wartości dokładniej oddają warunki panujące w całym akwarium.

Pomiary z użyciem czujnika TDS i pH polegają na dokonaniu pomiaru dwadzieścia razy. Następnie wyniki zostają posortowane oraz odrzucane zostają skrajne wartości - pięć najmniejszych i największych wartości, by z pozostałych dziesięciu wyliczyć średnią.

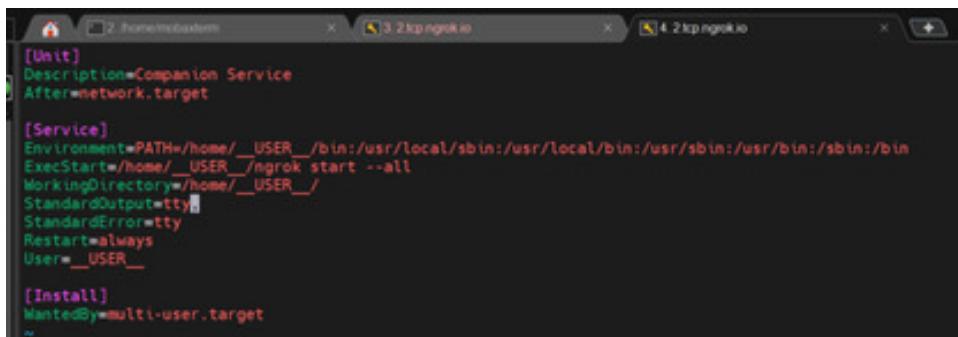
## 6.5. Połaczenie z Raspberry Pi

W początkowej fazie pracy nad prototypem, Raspberry Pi podłączona była do domowej sieci internetowej. Wiązało się to z istotnym ograniczeniem takim jak brak możliwości nadania statycznego IP. Po każdym restarcie maszyny, przypisywany był nowy adres IP, którego nie można było poznać z sieci zewnętrznej. Niezbędne było umożliwienie swobodnego dostępu do pracy z Raspberry Pi. Wykorzystane zostało narzędzie - Ngrok, które udostępnia lokalne serwery do publicznego internetu za pośrednictwem bezpiecznych tuneli.



Rys. 6.1. Widok z dashboard w aplikacji webowej Ngrok

Po każdym restarcie Raspberry Pi, aplikacja Ngrok uruchamiała się automatycznie w tle dzięki skorzystaniu z serwisów. Takie rozwiązanie umożliwiło swobodny dostęp do łączenia się Raspberry Pi niezależnie od tego do jakiej sieci urządzenie było podpięte. Korzystając z połączeń SSH można było łączyć się z maszyną z różnych komputerów jednocześnie, co było niezbędne do swobodnej, nieograniczonej pracy nad prototypem.



```
[Unit]
Description=Companion Service
After=network.target

[Service]
Environment=PATH=/home/_USER/_bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin
ExecStart=/home/_USER/_ngrok start --all
WorkingDirectory=/home/_USER_/
StandardOutput=tty
StandardError=tty
Restart=always
User=_USER_

[Install]
WantedBy=multi-user.target
```

Rys. 6.2. Plik konfiguracyjny .service

W późniejszej fazie pracy nad projektem akwarium wraz z Raspberry Pi zostało przeniesione na teren Politechniki Gdańskiej. Nawiązywanie połączenia z urządzeniem możliwe było dzięki wpięciu Raspberry Pi do uczelnianej prywatnej sieci wirtualnej. Takie rozwiązańiem skróciło proces nawiązywania połączenia o konieczność logowania do aplikacji Ngrok w celu sprawdzenia adres IP przypisanego do Raspberry Pi.

## 6.6. Oprogramowanie Arduino

Oprogramowanie mikrokontrolera Arduino ma na celu sterowanie pompami oraz odczytywanie wskazań czujników analogowych i przesyłanie ich do Raspberry Pi. Komunikacja z Raspberry Pi odbywa się poprzez magistralę I2C za pomocą biblioteki "Wire.h". Arduino jest w tej komunikacji urządzeniem podrzędnym (slave) a Raspberry Pi urządzeniem nadziednym (master).

### 6.6.1. Biblioteka "Wire.h"

Umożliwia utworzenie dwóch funkcji przerwań: onReceive i onRequest. Funkcja przypisana jako onReceive zostanie niezwłocznie wywołana w momencie, kiedy do Arduino zostaną wysłane dane. Funkcja oznaczona jako onRequest jest wywoływana kiedy do urządzenia przyjdzie żądanie o wysłanie danych.

### 6.6.2. Opis programu

Program tego mikrokontrolera podzielony jest na kilka głównych funkcji:

1. Klasa pompy: W celu uproszczenia sterowania została napisana klasa pompy. Zawiera ona takie metody jak:
  - a) SyringePump(int pins[4]): Inicjuje nową pompę. Jako argument przyjmuję tablicę PINów do których podpięty jest sterownik danej pompy.

- b) dosing(int dose) (listing 6.7): Odpowiada za dozowanie odpowiedniej ilości substancji (ilość podawana jest w argumencie jako zmienna całkowita i oznacza ilość obrotów silnika do wykonania). W pętli podawane jest napięcie na kolejne PIN-y podłączone do sterownika silnika krokowego, który na skutek tego sygnału obraca silnikiem pompę, ten z kolei popcha tłok.

```

1 void SyringePump :: dosing ( int dose ) {
2     while ( dose > 0 ){
3         for ( int i = 0 ; i < this -> pinNum ; i ++ ) {
4             digitalWrite ( this -> pins [ i ] , HIGH ) ;
5             delay ( this -> stepDelay ) ;
6             digitalWrite ( this -> pins [ i ] , LOW ) ;
7         }
8         dose -- ;
9     }
10 }
11

```

Listing 6.7: Syringe Pump

- c) efile(int dose): Ta metoda działa podobnie do powyższej. Różni się jedynie odwrotną kolejnością aktywowania PIN-ów. Skutkuje to obrotem silnika w przeciwnym kierunku co powoduje cofanie się tłoka pompy. Umożliwia to napełnianie strzykawek.
2. Klasa kolejki dozowania: Funkcje przerwań powinny być wykonywane jak najszybciej. Dozowanie substancji zajmuje jednak dużo czasu, dlatego nie można by wywołać metody dozującej w funkcji przerwania. Utworzono kolejkę do której funkcja przerwania zapisuje kolejne żądania dozowania odpowiednich substancji. Kiedy kolejka nie jest pusta, główna pętla programu pobiera z niej element i wywołuje metodę odpowiadającą za dozowanie bądź napełnianie jednej z pomp. Klasa ta składa się z następujących komponentów.
    - a) pola:
      - i. int pump : numer pompy
      - ii. int dose : doza jaka powinna być zaaplikowana (podana w obrotach silnika krokkowego)
      - iii. QueueElement\* next : wskaźnik na kolejny element kolejki
    - b) metody:
      - i. int get\_next(int tab[2]) : zdejmuje pierwszy element kolejki, wpisuje do tablicy po danej w argumencie numer pompy i dozę, ustawia kolejny element jako pierwszy.
      - ii. void add(int pump,int dose): dodaje nowy element do kolejki
      - iii. bool is\_empty() : sprawdza czy kolejka jest pusta
  3. Setup (listing 6.8): Wykonuje się przy starcie programu. Inicjowana jest tutaj komunikacja z Raspberry Pi
    - a) za pomocą metody Wire.begin() inicjujemy komunikację I2C i nadajemy adres naszemu urządzeniu (Arduino).
    - b) za pomocą metod Wire.onRequest() oraz Wire.onReceive() przypisujemy odpowiednie funkcje, które będą używane w komunikacji.

```

1 void setup () {
2     Serial . begin ( 9600 ) ;
3     Wire . begin ( 0x7 ) ;
4     Wire . onReceive ( receiveEvent ) ;
5     Wire . onRequest ( requestEvent ) ;
6     pinMode ( TdsSensorPin , INPUT ) ;
7 }
8

```

Listing 6.8: Inicjalizacja komunikacji I2C i przypisanie odpowiednich funkcji

4. Funkcja void receiveEvent(int byte\_count): oznaczona jest jako onReceive czyli zostaje wywoływana kiedy Arduino otrzyma dane. Pierwszy bajt przesyłany do Arduino to kod operacji do wykonania. Operacje możliwe do wykonania to:
  - a) zmiana aktywnego czujnika na czujnik pH : jako czujnik aktywny ustawiany jest czujnik pH
  - b) zmiana aktywnego czujnika na czujnik TDS : jako czujnik aktywny ustawiany jest czujnik TDS
  - c) nowe zlecenie dozowania: kolejne dwa bajty oznaczają numer pompy i dozę substancji (doza podana w ml). Doza wymnażana jest przez współczynnik rotetes\_per\_ml, który mówi ile obrotów powinien wykonać silnik pompy aby za dozować 1 ml substancji. Do kolejki dozowania dodawany jest nowy element.
5. Funkcja void requestEvent(): oznaczona jest jako onRequest czyli wywoływana jest kiedy Arduino otrzyma żądanie o dane. Odczytywane jest wskazanie czujnika ustawionego jako aktywny. Wynik odsyłany jest do Raspberry Pi.
6. Loop: funkcja wykonująca się w pętli przez cały czas działania programu. Zdejmuje ona kolejne elementy z kolejki dozowania i wywołuje metody dozowania odpowiednich pomp.

## **6.7. Program nadzorujący**

Decyzyjna część kodu została przeniesiona na platformę Apache Airflow, dzięki temu rozwiązanie stało się przejrzystsze oraz łatwiejsze w utrzymaniu, dając wiele możliwości na rozwijanie w przyszłości np. zparametryzowanie urządzenia do uprawy innych gatunków roślin lub dodanie obsługi bazy danych.

Apache Airflow jest narzędziem typu open source służącym do zarządzania przepływem pracy. Praca z Airflow opiera się na tworzeniu diagramów - DAG (Directed Acyclic Graph). DAG określa zależności między zadaniami oraz kolejność ich wykonywania oraz ewentualne próby ponownego ich wykonywania. Zadania opisują co należy zrobić np. pobrać dane, przeprowadzić analizę lub wykonać określoną akcję.

### *6.7.1. Elementy instalacji Apache Airflow*

Podstawowa konfiguracja Apache Airflow składa się z takich elementów jak:

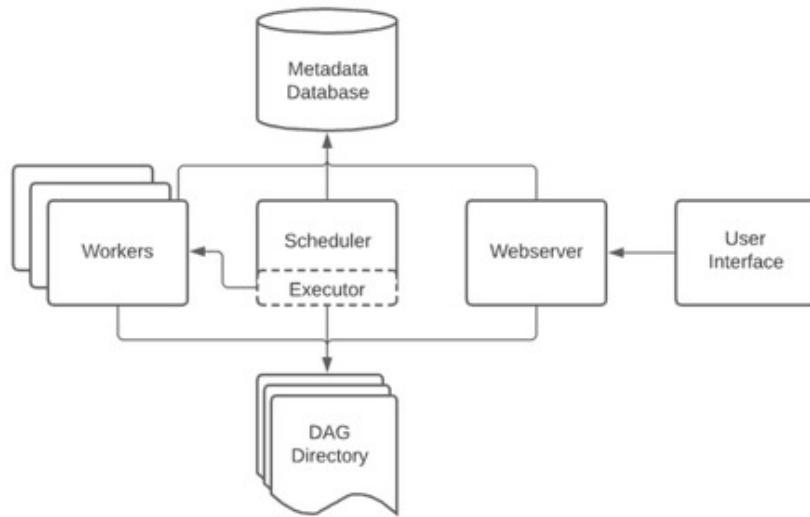
Harmonogram - zajmuje się obsługą wyzwalania zaplanowanych przepływów pracy oraz przesyaniem zadań, które mają zostać uruchomione, do wykonawcy.

Wykonawca - obsługuje uruchomione zadania. W domyślnej instalacji Airflow uruchamiane jest to wewnętrz harmonogramu.

Serwer sieciowy - interfejs użytkownika służący do sprawdzania, wyzwalania i debugowania zachowań diagramów oraz zadań.

Folder plików DAG - odczytywany przez harmonogram i wykonawcę.

Baza danych - używana przez harmonogram, wykonawcę oraz serwer.

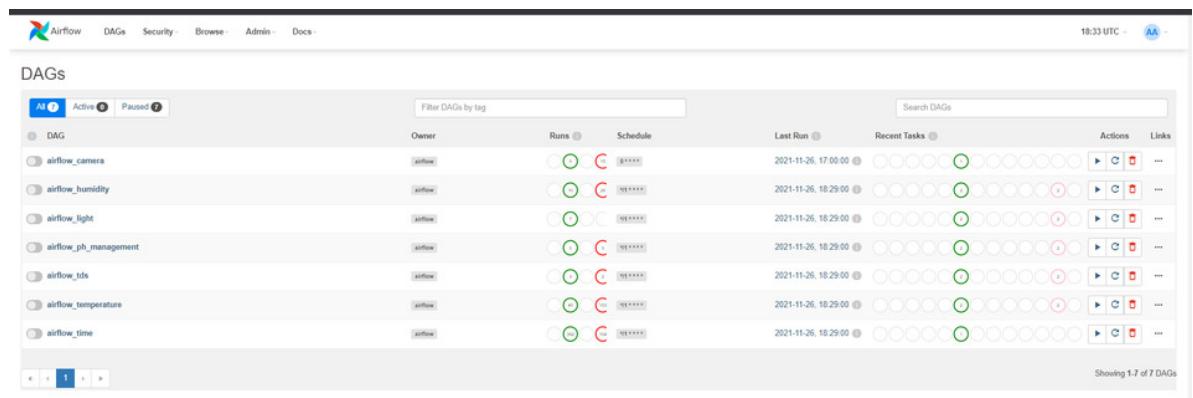


Rys. 6.3. Schemat elementów Airflow (Obraz z airflow.apache.org)

Prezentowana warstwa programowa nie korzysta z dostarczonej wraz z Apache Airflow bazy danych. Do gromadzenia danych służy klasa Loger, która zapisuje dane po stronie Raspberry Pi do plików JSON.

#### 6.7.2. Diagramy DAG

Utworzono diagramy DAG dla zarządzania pracą kamery, sterowania wilgotnością, poziomem naświetlenia, sterowaniem poziomu pH oraz TDS, a także temperaturą oraz diagram airflow\_time, który odpowiada za ustawienie czasu, dnia, dnia fazy w której znajduje się roślina, oraz wysłania żądania do zalogowania ostatnio odczytanych wartości z czujników.



Rys. 6.4. Utworzone diagramy DAG. Widok z platformy Apache Airflow

	Key	Val	Description	Is Encrypted
□	codes	{"to_low": 1, "to_high": 2, "correct": 0}		False
□	daily_light_cycle	{"resting": {"ON": 4, "OFF": 20}, "flowering": {"ON": 6, "OFF": 18}, "growth": {"ON": 3, "OFF": 21}}		False
□	day	22		False
□	day_of_phase	7		False
□	day_of_the_month	26		False
□	indication_limits	{"resting": {"days": 7, "ph": {"standard": 6.1, "hysteresis": 0.2}, "light": {"standard": 0, "hysteresis": 1000}, "lightF": {"standard": 1, "hysteresis": 1}, "temperature": {"day": {"standard": 26, "hysteresis": 3}, "night": {"standard": 24, "hysteresis": 3}}, "humidity": {"standard": 70, "hysteresis": 5}}, "growth": {"days": 14, "ph": {"standard": 6.1, "hysteresis": 0.2}, "light": {"standard": 700, "hysteresis": 100}, "lightF": {"standard": 1, "hysteresis": 1}, "temperature": {"day": {"standard": 26, "hysteresis": 3}, "night": {"standard": 24, "hysteresis": 3}}, "humidity": {"standard": 70, "hysteresis": 5}}, "flowering": {"days": 14, "ph": {"standard": 6.1, "hysteresis": 0.2}, "light": {"standard": 56, "hysteresis": 6.1}, "lightF": {"standard": 1, "hysteresis": 1}, "temperature": {"day": {"standard": 1050, "hysteresis": 200}, "night": {"standard": 1, "hysteresis": 1}, "temperature": {"day": {"standard": 26, "hysteresis": 3}, "night": {"standard": 24, "hysteresis": 3}}}, "humidity": {"standard": 76, "hysteresis": 5}}}}		False
□	modules	{"Temperature": 1, "Humidity": 1, "PH": 1, "TDS": 1, "water_level": 0, "light": 1, "air": 1}		False
□	phase	flowering		False
□	phase_duration	{"flowering": 56, "growth": 14, "resting": 7}		False
□	RPI_IP	http://172.20.2.140:8000		False
□	sensor_indications	{"ph": "None", "hds": "None", "light": "None", "temperature": "None", "humidity": "None"}		False
□	time_of_day	night		False

Rys. 6.5. Reprezentacja stałych wartości

Zadania związane z użyciem kamery uruchamiane są o każdej pełnej godzinie, natomiast wszystkie pozostałe co minutę.

Diagram do sterowania temperaturą w akwarium (listing 6.9):

```

1  from datetime import timedelta
2  from airflow import DAG
3  from airflow.operators.bash import BashOperator
4  from airflow.operators.dummy import DummyOperator
5  from airflow.operators.python import PythonOperator
6  from airflow.operators.python import BranchPythonOperator
7  from airflow.models import Variable
8  from airflow.utils.dates import days_ago
9  from airflow.providers.postgres.operators.postgres import PostgresOperator
10 import requests
11 from time import sleep
12 import glob
13 import json
14
15 address = Variable.get("RPI_IP", deserialize_json=False)
16 variables = Variable.get("indication_limits", deserialize_json=True)
17 ideal_temperatere = variables[Variable.get("phase")][ "temperature"][Variable.get("time_of_day")][ "standard"]
18 hysteresis = variables[Variable.get("phase")][ "temperature"][Variable.get("time_of_day")][ "hysteresis"]
19
20
21 args = {
22     'owner': 'airflow',
23 }
24
25 def measure_temp():
26     r = requests.get(address + '/temperature')
27     return float(r.text)
28
29 def check_temp():
30     temperature = measure_temp()
31     if temperature > ideal_temperatere + hysteresis:
32         return 'lower_temperature'
33     elif temperature < ideal_temperatere:
34         return 'raise_temperature'
35     return 'everything_is_ok'
36
37 def raise_the_temperature():
38     r = requests.post(address + '/temperature', data = 'increase')
39     return
40
41 def lower_the_temperature():
42     r = requests.post(address + '/temperature', data = 'decrease')
43     return
44

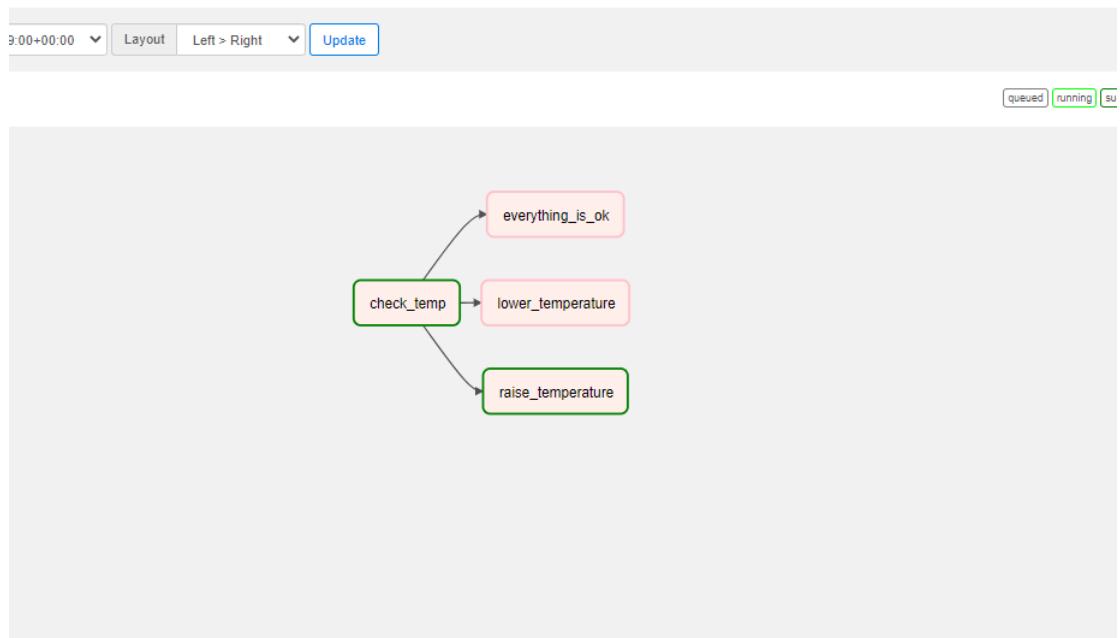
```

```

45 def ok_temperature():
46     r = requests.post(address + '/temperature', data = 'remain')
47     return
48
49 with DAG(
50     dag_id='airflow_temperature',
51     default_args=args,
52     catchup=False,
53     schedule_interval='*/1 * * * *',
54     max_active_runs=1,
55     start_date=days_ago(2),
56     dagrun_timeout=timedelta(hours=10),
57     params={"example_key": "example_value"} ,
58 ) as dag:
59
60     check_temp = BranchPythonOperator(
61         task_id='check_temp',
62         python_callable=check_temp
63     )
64
65     lower_temperature = PythonOperator(
66         task_id='lower_temperature',
67         python_callable=lower_the_temperature
68     )
69     raise_temperature = PythonOperator(
70         task_id='raise_temperature',
71         python_callable=raise_the_temperature
72     )
73
74     everything_is_ok = PythonOperator(
75         task_id='everything_is_ok',
76         python_callable=ok_temperature
77     )
78
79     check_temp >> lower_temperature
80     check_temp >> raise_temperature
81     check_temp >> everything_is_ok
82
83 if __name__ == "__main__":
84     dag.cli()

```

Listing 6.9: Diagram związany ze sterowaniem temperaturą



Rys. 6.6. Reprezentacja diagramu temperatury z poziomu widoku platformy Airflow

## **7. TESTY URZĄDZENIA**

Autor rozdziału: Maciej Łotysz

Istotną częścią tego projektu było sprawienie, aby urządzenie było w stanie poprawnie zdiagnozować i szybko reagować na sytuacje, w których badane czynniki wykraczały poza określone i przyjęte standardy. W celu uniknięcia nadmiernej ingerencji w cykl rozwoju roślin i sztucznego tworzenia sytuacji, w których odczyty odbiegałyby od normy, dlatego w zakres eksperymentów będą najczęściej wchodziły anomalie wynikające z naturalnych zmian dla rozwoju truskawek, takich jak zmiana cyklu dziennego na cykl nocny. Jedynym wyjątkiem, będzie test dla pomiaru wilgotności, w którym warunki testowe zostały wymuszone.

### **7.1. *Opis środowiska***

Prototyp zbudowany został, aby pomieścić trzy krzaczki truskawek, a samo urządzenie umieszczone jest w pokoju technicznym na piątym piętrze wydziału ETI Politechniki Gdańskiej. Jest to ciemne (bez dostępu do światła dziennego) pomieszczenie z temperaturą pokojową. Po zamontowaniu akwarium w nowej lokalizacji zostało ono napełnione wodą kranową. Dodatkowo do urządzenia zadano następujące preparaty:

- nawóz na fazę wzrostu (nawóz zostanie podmieniony na nawóz na fazę kwitnienia dopiero przy zmianie fazy rozwoju truskawek),
- regulator kwasowości pH-,
- regulator kwasowości pH+.

Warunki środowiska w jakim zostało uruchomione urządzenie:

- temperatura=25 °C,
- wilgotność=40%,
- brak okien oraz innych źródeł światła.

Do urządzenia została wlana woda o następujących parametrach:

- pH=6.64,
- poziom zanieczyszczenia wody=342ppm.

### **7.2. *Plan eksperymentów***

Głównym celem pracy inżynierskiej jest stworzenie optymalnego, hydroponicznego środowiska rozwoju dla dowolnej rośliny, którą można hodować w postaci sadzonki. Jako owoc wybrana została truskawka, która zostanie oddany ekspermentowi w prototypowej platformie do zautomatyzowanej uprawy hydroponicznej. Eksperiment odbywa się praktycznie w pełni zautomatyzowany. Jedynie niewielka część zadań wymaga ingerencji zewnętrznej człowieka. Do tych czynności można zaliczyć:

- napełnienie urządzenia wodą przed posadzeniem roślin,
- podłączenie urządzenia do prądu,
- podpięcie do odpowiednich dozowników nawozów i regulatorów kwasowości,
- wymiana nawozów po zmianie fazy wzrostu na fazę kwitnienia.

Poza wymienionymi czynnościami cały proces odbywa się w sposób w pełni zautomatyzowany. Urządzenie monitoruje wszystkie potrzebne czynniki i warunki panujące wewnątrz urządzenia oraz samodzielnie podejmuje decyzje, o ich ewentualnej korekcji w przypadkuodejścia, któregoś z nich od określonej, na podstawie fachowej literatury, normy. Do elementów monitorowanych przez naszą platformę zalicza się:

- poziom wilgotności,
- temperatura powietrza,
- zawartość nawozu w wodzie,
- poziom pH wody,
- natężenie światła.

Eksperymenty polegały na sprawdzeniu poprawności i szybkości reakcji urządzenia na odczyty wartości wychodzących poza określone normy. Z tego właśnie powodu obiektem testów stały się czynniki, które monitoruje prototyp. Konkretniej mówiąc eksperimentom poddane zostaną regulacje następujących elementów:

- poziom wilgotności w urządzeniu,
- temperatura wewnątrz urządzenia,
- poziom nawozu w wodzie,
- poziom pH wody.

### **7.3. *Opis pierwszego eksperimentu***

Jedyny z przeprowadzonych eksperimentów, który ingerował w pracę urządzenia, polegał na otworzeniu klapy przez krótką chwilę, co spowodowało znaczne obniżenie poziomu wilgotności w akwarium. Jak widać na załączonym wykresie 7.1, urządzenie zostało otwarte 25.11.2021 roku o godzinie 12:52:04. Początkowo system nie reaguje na zmianę poziomu wilgotności, ponieważ ciągle znajduje się on w zakresie wartości dopuszczalnych. Dopiero o 12:54:03 dokonuje on pierwszego pomiaru, który wykracza poza określoną dziedzinę (w momencie odczytu wynosił dokładnie 46.5%) iłącza on atomizer, którego zadaniem jest przywrócenie pomiarów wilgotności na odpowiedni poziom. Udało mu się to po około 4 sekundach, ponieważ jak widać na wykresie 7.1 o godzinie 12:58:10, odczytana wartość poziomu wilgotności znajduje się już w określonych normach (odczytana wartość to 58% wilgotności). Następnie przez kolejne 14 minut widzimy, że poziom ten ustabilizował się i oscyluje w granicach optimum. Test ten zakończył się powodzeniem i dodatkowo dał bardzo pozytywne wyniki, ponieważ poradził sobie z prawie dziesięcioprocentowym odchyleniem od normy i około piętnasto procentowym odejściem od optimum w zaledwie 4 sekundy, co jest czasem tak krótkim, że anomalia taka nie powinna w żaden sposób zakłócić prawidłowego cyklu rozwojowego roślin.



Rys. 7.1. Test kontroli poziomu wilgotności

#### 7.4. Opis drugiego eksperymentu

Kolejnym eksperymentem jest sprawdzenie reakcji na nagłą zmianę zakresu temperatury, wynikającą ze zmiany cyklu pracy z dziennego na cykl nocny. Jest to w pełni naturalny proces, który odbywa się codziennie i jest w pełni integralną częścią rozwoju roślin. Jako przykład posłuży tu taki proces z dnia 25.11.2021 r. Zmiana cyklu następuje o godzinie osiemnastej. Stąd też można zauważyć na wykresie 7.2 nagłą zmianę wartości temperatury optymalnej, maksymalnej i minimalnej o tej godzinie. Pierwszy odczyt, w którym temperatura przekracza dopuszczalny próg następuje o 18:01:09, to właśnie wtedy podjęta zostaje decyzja o włączeniu ogniw Peltiera, które odpowiada w urządzeniu za chłodzenie. Osiagnięcie poziomu maksymalnej dopuszczalnej temperatury następuje już po niecałej sekundzie, ponieważ już o 18:02:07. Następnie system chłodzący pozostaje jeszcze włączony, aż do momentu stabilizacji temperatury w obrębie optymalnego poziomu. Wyłączenie ogniw Peltiera następuje po około 20 sekundach od jego włączenia, gdy temperatura osiągnie poziom optymalny dla cyklu nocnego. Jest to kolejny eksperyment, który zakończył się powodzeniem. Obniżenie temperatury o 2 °C nastąpiło po jednej sekundzie, natomiast cały proces obniżenia temperatury do poziomu optimum zajął około 20 sekund, co również jest wynikiem bardziej niż zadowalającym. Po raz kolejny przeprowadzony test pokazał szybkość i poprawność reakcji urządzenia.



Rys. 7.2. Test kontroli temperatury

### 7.5. Opis trzeciego eksperymentu

Następnie urządzenie zostało sprawdzone pod kątem dostosowania poziomu nawozu w wodzie i jego reakcji na nagłe zwiększenie wymaganego poziomu substancji odżywcznych. Zmiana ta wynika z faktu, iż przy fazie kwitnienia truskawki potrzebują znacznie większej dawki substancji odżywcznych, a co za tym idzie nawozy, do prawidłowego rozwoju. Na wykresie 7.3 możemy zaobserwować zmianę z fazy wzrostu na fazę kwitnienia, która nastąpiła w nocy z 18 na 19 listopada (zmiana ta zachodzi dokładnie o północy). Widać, że na początku poziom nawozu w wodzie wynosi około 850 PPM i utrzymuje się on w takich okolicach przez około 50 minut, dopiero po tym czasie zaczyna powoli rosnąć. Ma to związek z faktem, iż nawóz rozprowadzany jest w wodzie, proces jego dyfuzji w cieczach zachodzi zdecydowanie wolniej, dlatego też cały proces zwiększenia poziomu nawozu w wodzie jest znacznie dłuższy niż w przypadku regulacji czynników przedstawionych wcześniej. Pomimo włączenia pomp dozujących nawóz już o północy, dopiero po kolejnej godzinie poziom nawozu osiąga minimalny, wymagany standard, a dopiero po kolejnych 30 minutach dochodzi do optymalnej wartości. Następnie na wykresie 7.3 widzimy jeszcze kolejne pół godziny, z którego można wywnioskować, że poziom nawozu się ustabilizował i cyrkuluje w okolicach optimum. Analizując wyniki tego konkretnego eksperymentu nie można zapomnieć o środowisku w jakim ten test został przeprowadzony oraz o fakcie, iż nastąpił tutaj znaczący wzrost wartości poziomu substancji odżywcznych (około 20%). Uwzględniające właśnie te czynniki, można uznać ten eksperyment za udany i w pełni satysfakcjonujący.



Rys. 7.3. Test kontroli poziomu nawozu

## 7.6. Opis czwartego eksperymentu

Ostatnim przeprowadzonym eksperymentem jest ten sprawdzający reakcję urządzenia na podniesienie się poziomu pH wody. Dane z tego testu pochodzą z dnia 13.11.2021 r. Na wykresie widać, że około godziny 21:04 następuje delikatny wzrost poziomu pH w wodzie, a o 21:09 dokonany jest pierwszy pomiar pH wychodzący poza górną granicę. To właśnie po tym zdarzeniu następuje włączenie pompy dozującej regulator kwasowości zakwaszający wodę w urządzeniu. Pierwszy odczyt ponownie będący w granicach pożądanych odnotowujemy już po 3 minutach. Przez okres kolejnych 6 minut zaobserwować można proces stopniowego zmniejszania pH i jego stabilizacji w pobliżu wartości optymalnej. Pomimo faktu, iż dozowanie regulatora kwasowości zachodzi w środowisku wodnym, tak samo jak w przypadku eksperymentu dotyczącego poziomu nawozu, to w tym eksperymencie satysfakcyjujące wyniki otrzymywane są znacznie szybciej. Ma to związek z faktem, że w przypadku regulacji pH mamy do czynienia z regulacją wartości o znacznie mniejszy procent - tutaj jest to około 3%, nie 20% jak było to w przypadku regulacji zawartości nawozu w wodzie. Podsumowując ten test można stwierdzić, że po raz kolejny urządzenie zostało prawidłowo dostosowane do szybkiej i sprawnej reakcji na odbiegające od określonych norm wartości odczytywanych przez nie czynników.



Rys. 7.4. Test kontroli poziomu pH wody

## 7.7. Podsumowanie wyników

Każdy z przeprowadzonych powyżej eksperymentów zakończył się powodzeniem. Co więcej można stwierdzić, że pod względem reakcji urządzenia na występujące anomalie, działa ono bez zarzutów. Warto też zwrócić uwagę na szybkość radzenia sobie z powstałymi problemami. W każdym przypadku, przywrócenie wartości optymalnych, następował w czasie tak krótkim, że anomalie te nie mogły w żadnym stopniu wpłynąć na prawidłowy rozwój truskawek. Podsumowując, eksperymenty udowadniają prawidłowe działanie całej platformy i jej zdolność do automatycznej i zdolność do utrzymania założonych wcześniej warunków środowiska.

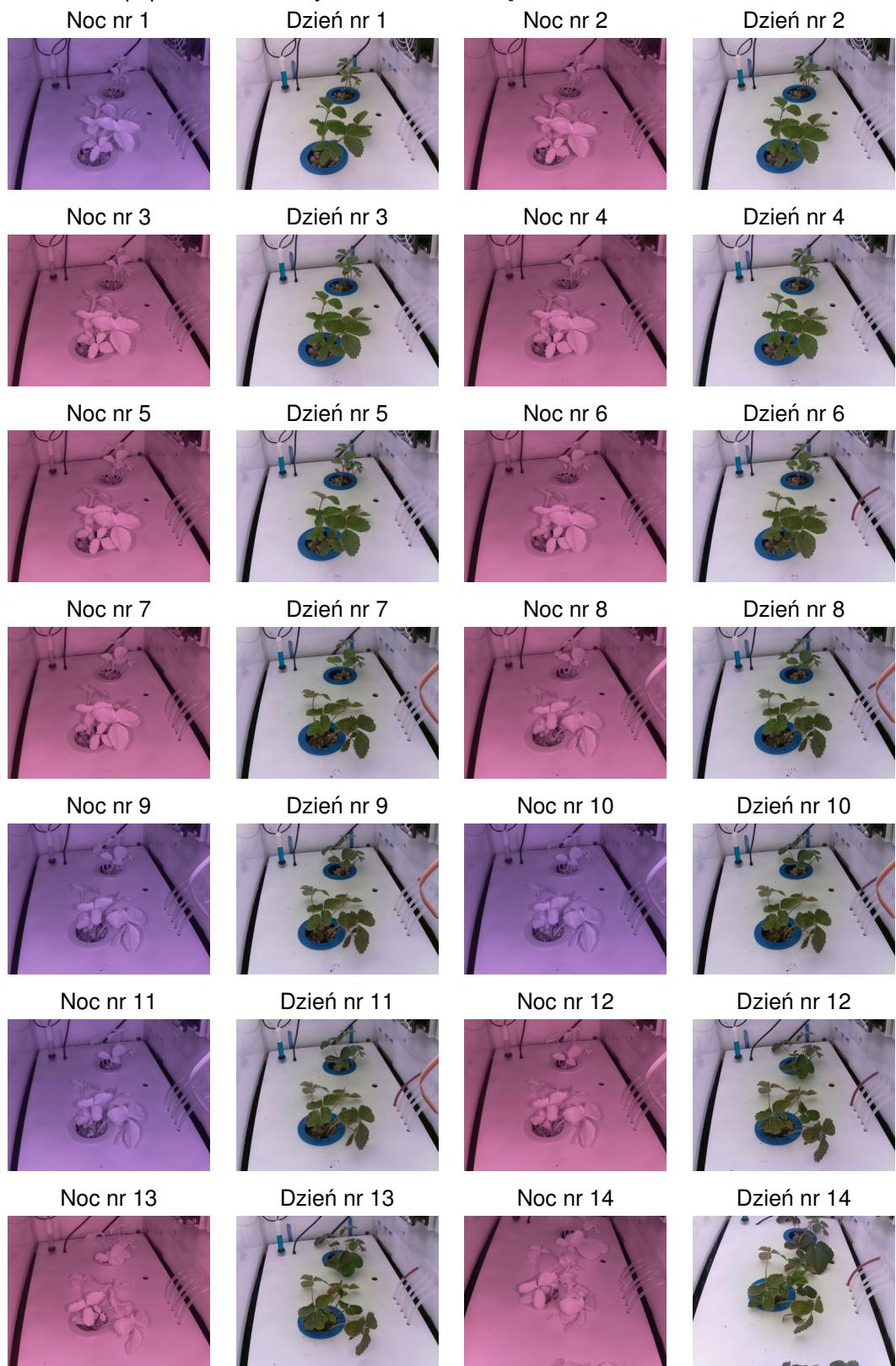
## 7.8. Prezentacja eksperimentalnej hodowli

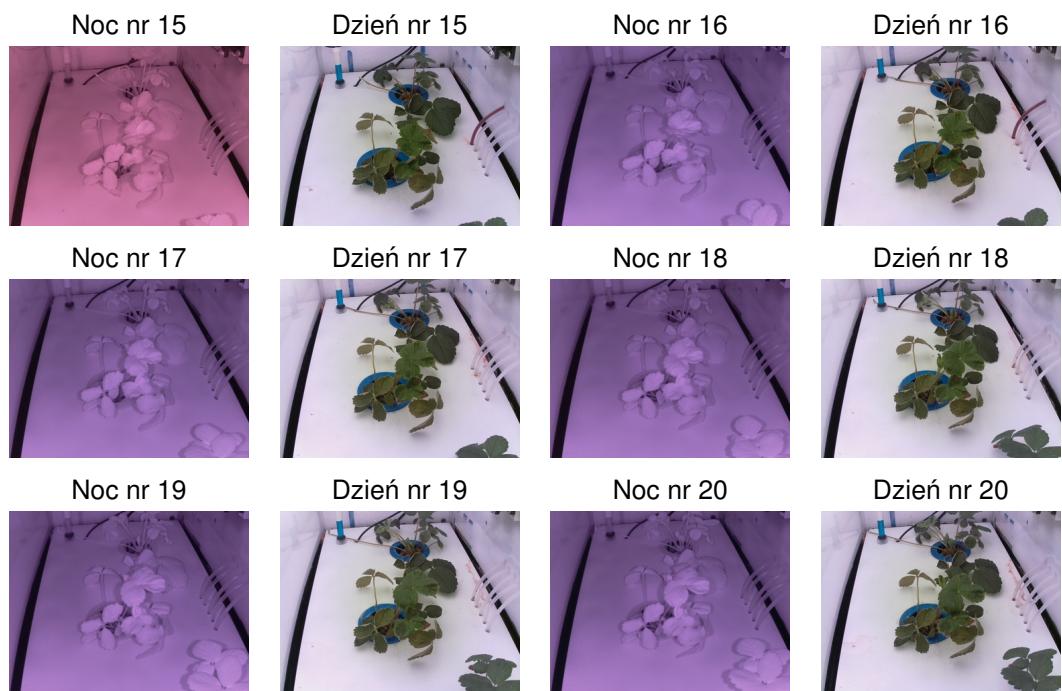
Bardzo ważnym wyznacznikiem jakości dla przygotowanego prototypu jest rozwój roślin w czasie eksperimentalnej hodowli. Wykorzystując mechanizmy monitoringu pracy urządzenia można pokazać poprawność jego działania.



Na powyższych zdjęciach można zaobserwować jak bardzo krzaki truskawek rozwinięły się przez 20 dni czasu trwania hodowli. Dodatkowo warto dodać, że rozwój roślin przebiega zgodnie z rozwojem teoretycznym opisany w literaturze o tematyce truskawek. Zaś na poniższych fotografiach widać stopniowy rozwój roślin z dnia na dzień, co jeszcze lepiej pozwala na analizę i

stwierdzenie poprawności rozwoju truskawek w urządzeniu.





## **8. PODSUMOWANIE**

Autor rozdziału: Jakub Świłocki

Celem niniejszego projektu inżynierskiego było stworzenie prototypu urządzenia wraz z modelem decyzyjnym umożliwiającym uruchomienie automatycznej hodowli roślinnej. W trakcie projektu zdecydowano się na uprawę truskawek ze względu na specyficzne warunki środowiskowe, które wymagają ciągłej kontroli i korygowania. Praca miała charakter teoretyczno-eksperymentalny ze względu na równolegle prowadzone prace nad opracowaniem modelu decyzyjnego oraz prototypu urządzenia do prowadzenia hodowli.

Eksperymenty, których przebieg przedstawiono w rozdziale 7, potwierdziły, że urządzenie opisane w tej pracy rozwiązuje problemy zdefiniowane w rozdziale 2. Wyniki eksperymentów pokazały, że założenia dotyczące pomiaru i kontroli wilgotności, poziomu pH wody, zawartości soli odżywczych w wodzie są realizowane bez zarzutu. Urządzenie utrzymywało również temperaturę w założonym przedziale, ale na wykresie 7.2 można zauważyć, że w cyklu dziennym temperatura utrzymuje się blisko górnej granicy przedziału. Oznacza to, że układ chłodzenia jest jednym z elementów który można usprawnić w przyszłości. Opracowane rozwiązanie można rozwinać również o nowe funkcjonalności takie jak analizę zdjęć w celu automatycznej detekcji chorób roślin bądź automatycznego wykrywania fazy rozwoju rośliny.

## WYKAZ

## LITERATURY

- [1] A Sønsteby and OM Heide. Long-day control of flowering in everbearing strawberries. *The Journal of Horticultural Science and Biotechnology*, 82(6):875–884, 2007.
- [2] Canna. Wszystko, co chcielibyście wiedzieć o systemach recyrkulacji.  
<https://grower.com.pl/public/assets/BT/Canna%20Niedobory/info%20aqua.pdf>.
- [3] Hort Americas. Essentials for growing hydroponic strawberries successfully hortamericas.  
<https://hortamericas.com/blog/news/essentials-for-growing-hydroponic-strawberries-successfully/>, 2019. [Online; odwiedzona 29 listopada 2021].
- [4] The Advanced Nutrients Team. How To Build Your Own Hydroponics Grow Room – Dialing In Your Temperature and Humidity (part 3) advancednutrients.  
<https://www.advancednutrients.com/articles/grow-room-temperature-humidity/>, 2017. [Online; odwiedzona 29 listopada 2021].
- [5] Wikipedia. Pompa perystaltyczna — Wikipedia, the free encyclopedia. <http://pl.wikipedia.org/w/index.php?title=Pompa%20perystaltyczna&oldid=64597932>, 2021. [Online; odwiedzona 29 listopada 2021].
- [6] Dylan Herrada. Adafruit TSL2591 Library. [github](https://github.com/adafruit/Adafruit_TSL2591_Library).  
[https://github.com/adafruit/Adafruit\\_TSL2591\\_Library](https://github.com/adafruit/Adafruit_TSL2591_Library), 2021. [Online; odwiedzona 29 listopada 2021].
- [7] Dylan Herrada. DHT sensor library. [github](https://github.com/adafruit/DHT-sensor-library).  
<https://github.com/adafruit/DHT-sensor-library>, 2021. [Online; odwiedzona 22 listopada 2021].
- [8] lady ada. Adafruit TSL2591 High Dynamic Range Digital Light Sensor.  
<https://learn.adafruit.com/adafruit-tsl2591>, 2021. [Online; odwiedzona 22 listopada 2021].
- [9] Malinowe PI. DHT11 - czujnik temperatury i wilgotności. <https://malinowepi.pl/post/80178679087/dht11-czujnik-temperatury-i-wilgotnosci-uklad>, 2014. [Online; odwiedzona 22 listopada 2021].
- [10] DFRobot. Gravity Analog TDD Sensor Meter For Arduino SKU SEN0244 DFRobot.  
[https://wiki.dfrobot.com/Gravity\\_\\_Analog\\_TDS\\_Sensor\\_\\_\\_Meter\\_For\\_Arduino\\_SKU\\_\\_-\\_SEN0244](https://wiki.dfrobot.com/Gravity__Analog_TDS_Sensor___Meter_For_Arduino_SKU__-_SEN0244), 2017. [Online; odwiedzona 22 listopada 2021].
- [11] Adafruit. TSL2591. [https://cdn-learn.adafruit.com/assets/assets/000/078/658/original/TSL2591\\_DS000338\\_6-00.pdf?1564168468](https://cdn-learn.adafruit.com/assets/assets/000/078/658/original/TSL2591_DS000338_6-00.pdf?1564168468), 2018. [Online; odwiedzona 22 listopada 2021].
- [12] JJ Slabbert. Raspberry PI Multiple I2C Devices.  
<https://www.instructables.com/Raspberry-PI-Multiple-I2c-Devices/>, 2018. [Online; odwiedzona 22 listopada 2021].

- [13] DFRobot. PH meter SKU SEN0161-DFRobot.  
[https://wiki.dfrobot.com/PH\\_meter\\_SKU\\_\\_SEN0161\\_](https://wiki.dfrobot.com/PH_meter_SKU__SEN0161_), 2018. [Online; odwiedzona 22 listopada 2021].
- [14] System Hydroponiczny GHE Aquafarm.  
<https://www.growbox.pl/pl/products/system-hydroponiczny-ghe-aquafarm-639>. [Online; odwiedzona 29 listopada 2021].
- [15] Nutriculture. NFG Gro-Tanks. <https://nutriculture.co.uk/product/nft-gro-tanks/>. [Online; odwiedzona 29 listopada 2021].
- [16] AutoPot. 4Pot System.  
<https://autopot.co.uk/all-products/1pot-systems/4pot-system/>. [Online; odwiedzona 29 listopada 2021].
- [17] Vakplast. Pionowy system hydroponiczny.  
<https://www.vakplast.com/home/vertical-hydroponic-system-1sm>. [Online; odwiedzona 29 listopada 2021].
- [18] Green Farm. Mini szklarnia. <https://green-farm.pl/>. [Online; odwiedzona 29 listopada 2021].
- [19] Freight Farms. Kontener firmy Freight Farms.  
<https://www.freightfarms.com/greenery-s>. [Online; odwiedzona 29 listopada 2021].
- [20] Pompa napowietrzająca. <https://allegro.pl/oferta/napowietrzacz-pompka-4w-do-wody-akwarium-4001-h-9516172657>. [Online; odwiedzona 20 października 2021].
- [21] Czujnik DHT11. [https://abc-rc.pl/product-pol-9784-Modul-DHT11-Czujnik-temperatury-i-wilgotnosci-Arduino-KEYES-KY-015.html?query\\_id=3](https://abc-rc.pl/product-pol-9784-Modul-DHT11-Czujnik-temperatury-i-wilgotnosci-Arduino-KEYES-KY-015.html?query_id=3). [Online; odwiedzona 20 października 2021].
- [22] Pompa perystaltyczna. <https://abc-rc.pl/product-pol-12527-Mini-Pompa-Perystaltyczna-DC-3-7V-6V-pompka-dozujaca-z-rurka-silikonowa.html?text=pompa+parystaltyczna>. [Online; odwiedzona 20 października 2021].
- [23] Apache Airflow. Architecture Overview.  
<https://airflow.apache.org/docs/apache-airflow/stable/concepts/overview.html>. [Online; odwiedzona 20 października 2021].
- [24] Apache Airflow. Scheduling & Triggers.  
<https://airflow.apache.org/docs/apache-airflow/stable/concepts/overview.html>. [Online; odwiedzona 20 października 2021].
- [25] Prateek Shrivastava Joy Lal Chattaraj and Jorge Villamariona. Understand Apache Airflow's Modular Architecture. <https://www.qubole.com/tech-blog/understand-apache-airflows-modular-architecture/>, 2020. [Ostatnio edytowana: 1 grudnia 2021], [Online; odwiedzona 29 listopada 2021].

**WYKAZ****RYSUNKÓW**

2.1	System hydroponiczny GHE Aqua Farm (zdjęcie pochodzi ze strony <a href="https://www.growbox.pl/">https://www.growbox.pl/</a> )	8
2.2	NFG Gro-Tanks (zdjęcie pochodzi ze strony nutriculture.co.uk)	9
2.3	4Pot System (zdjęcie pochodzi ze strony autopot.co.uk)	9
2.4	Pionowy system hydroponiczny firmy Vakplast (zdjęcie pochodzi ze strony <a href="http://vakplast.com">vakplast.com</a> )	10
2.5	Mini szklarnia firmy Green Farm (zdjęcie pochodzi ze strony <a href="http://green-farm.pl">green-farm.pl</a> )	10
2.6	Kontener firmy Freight Farms (zdjęcie pochodzi ze strony <a href="http://freightfarms.com">freightfarms.com</a> )	11
3.1	Porcje nawozu przy odpowiednich fazach (Tabela pochodzi ze strony: <a href="http://cannagarde-ning.com">cannagarde-ning.com</a> )	15
3.2	Dostępność składników odżywcznych a pH (Wykres pochodzi z broszury aqua.pdf)	15
4.1	Diagram stanów	16
4.2	Diagram przygotowań	17
4.3	Diagram systemu zarządzającego	17
4.4	Diagram modułu temperatury	18
4.5	Diagram obsługi pH	19
4.6	Diagram obsługi kamery	20
4.7	Diagram obsługi światła	21
4.8	Diagram obsługi wilgotności	22
4.9	Diagram obsługi logowania	23
4.10	Diagram obsługi nawozu	23
5.1	Model "akwarium"	25
5.2	Płyta środkowa "akwarium"	26
5.3	Projekt płyty środkowej	26
5.4	Model koszyczka na krzak truskawki	27
5.5	Krzak w koszyczku widziany z góry	27
5.6	Krzak w koszyczku widziany z boku (koszyczek przechodzi przez płytę środkową)	28
5.7	Pierwsza ściana przyklejona do płyty dolnej	29
5.8	Pokrywa górna przymocowana do urządzenia	29
5.9	Otwarta pokrywa górna	30
5.10	Czujnik DHT11 zamontowany w podstawce	31
5.11	Podstawa czujnika DHT11	31
5.12	Moduł chłodzenia (strona zimna)	32
5.13	Moduł chłodzenia (strona gorąca)	32
5.14	Moduł chłodzenia zamontowany w ścianie urządzenia	33
5.15	Moduł chłodzenia	33
5.16	Strona gorąca modułu chłodzenia chłodzona małym wentylatorem	34
5.17	Podstawa kontowa pod większy wentylator	34
5.18	Strona gorąca modułu chłodzenia z większym wentylatorem na podstawce	35
5.19	Wentylator w ścianie bocznej	36

5.20 Generator zasilający atomizer piezoelektryczny . . . . .	37
5.21 Rurka z wątą i atomizerem (próbne uruchomienie w szklance z wodą) . . . . .	37
5.22 Model rurki atomizera . . . . .	38
5.23 Atomizer pokryty osadem . . . . .	39
5.24 Atomizer z rurką przeniesiony do pojemnika z czystą wodą . . . . .	39
5.25 Sonda czujnika pH . . . . .	40
5.26 Moduł czujnika pH . . . . .	41
5.27 Sondy czujników pH i TDS przechodzące przez płytę środkową . . . . .	41
5.28 Moduł pomp (wersja pierwsza) . . . . .	42
5.29 model uchwytu pomp (wersja pierwsza) . . . . .	42
5.30 Moduł pomp (wersja druga) . . . . .	43
5.31 model uchwytu pomp (wersja druga) . . . . .	43
5.32 Wszystkie elementy czujnika TDS . . . . .	44
5.33 Moduł czujnika TDS) . . . . .	45
5.34 Sonda czujnika TDS . . . . .	45
5.35 Zasada pracy pompy perystaltycznej (zdjęcie pochodzi ze strony: abc-rc.pl) . . . . .	46
5.36 Konstrukcja pompy perystaltycznej (zdjęcie pochodzi ze strony: abc-rc.pl) . . . . .	46
5.37 Model pompy perystaltycznej (cały) . . . . .	47
5.38 Model pompy perystaltycznej (bez pokrywy) . . . . .	47
5.39 Model pompy perystaltycznej (element wirujący z rolkami) . . . . .	48
5.40 Model pompy perystaltycznej (element wirujący bez rolek) . . . . .	48
5.41 Pompa perystaltyczna (jedna z pierwszych wersji) . . . . .	49
5.42 Model podstawki pompy perystaltycznej . . . . .	50
5.43 Pompy perystaltyczne zamontowane na urządzeniu . . . . .	50
5.44 Pompy perystaltyczne z podstawkami . . . . .	51
5.45 Czujnik natężenia światła (TSL 2591) . . . . .	51
5.46 Paski LED na pokrywie urządzenia . . . . .	52
5.47 Moduły przekaźników na pokrywie urządzenia . . . . .	53
5.48 Wygląd urządzenia (wersja pierwsza) . . . . .	54
5.49 Płyta środkowa (wersja pierwsza) . . . . .	55
5.50 Uszkodzenie folii . . . . .	56
5.51 Urządzenie chwilę po malowaniu (wersja druga) . . . . .	57
5.52 Płyta środkowa (wersja druga) . . . . .	57
5.53 Kamera zamontowana na podstawce (widać świecące diody podczerwieni) . . . . .	58
5.54 Model podstawki kamery . . . . .	58
5.55 Napowietrzacz wody (zdjęcie pochodzi ze strony allegro.pl) . . . . .	59
5.56 Zasilacz zamontowany na urządzeniu . . . . .	60
5.57 Pełny schemat elektroniczny urządzenia . . . . .	61
5.58 Zbliżenie na moduł przekaźników sterujących światłami (każdy symbol diody oznacza cały segment pasków LED) . . . . .	62
5.59 Zbliżenie na drugi moduł przekaźników . . . . .	62
5.60 Zbliżenie na Arduino i pompy strzykawkowe . . . . .	63
5.61 Zbliżenie na konwerter stanów logicznych, kamerę i czujnik natężenia światła (TSL2591) .	63
5.62 Zbliżenie na Raspberry Pi . . . . .	64
5.63 Zbliżenie na czujnik DHT11 . . . . .	64

5.64 Tylna ściana urządzenia . . . . .	65
5.65 Pokrywa urządzenia . . . . .	66
5.66 Przód i góra urządzenia . . . . .	66
5.67 Lewa ściana urządzenia . . . . .	67
5.68 Prawa ściana urządzenia . . . . .	68
6.1 Widok z dashboard w aplikacji webowej Ngrok . . . . .	74
6.2 Plik konfiguracyjny .service . . . . .	75
6.3 Schemat elementów Airflow (Obraz z airflow.apache.org) . . . . .	78
6.4 Utworzzone diagramy DAG. Widok z platformy Apache Airflow . . . . .	78
6.5 Reprezentacja stałych wartości . . . . .	79
6.6 Reprezentacja diagramu temperatury z poziomu widoku platformy Airflow . . . . .	80
7.1 Test kontroli poziomu wilgotności . . . . .	83
7.2 Test kontroli temperatury . . . . .	84
7.3 Test kontroli poziomu nawozu . . . . .	85
7.4 Test kontroli poziomu pH wody . . . . .	86

**WYKAZ****TABEL**

6.1 Wykaz wejść/wyjść warstwy sprzętowej . . . . .	69
6.2 Urządzenia połączone z Raspberry Pi za pomocą magistrali I2C . . . . .	69