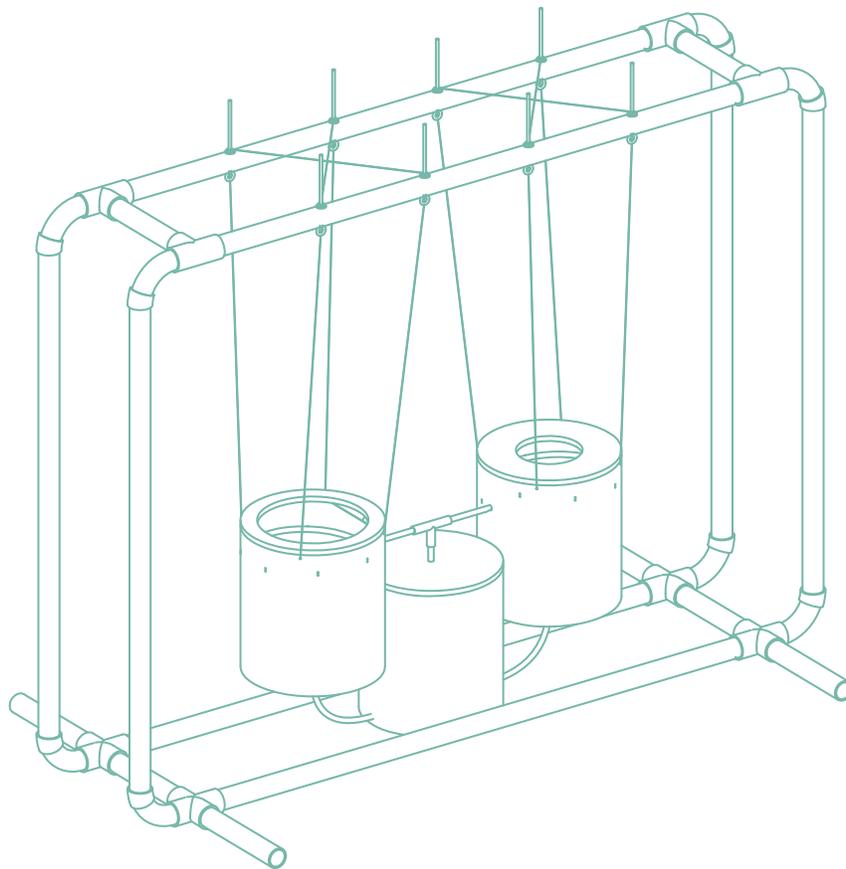


HYDRBASE

UC Berkeley School of Information 2016 Master's Final Project Report



Anubhav Gupta Bryan Morgan Joshua Appleman Juan Shishido Robin Kalia

Advisor: John Chuang

Table of Contents

Acknowledgements	3
Introduction	4
Problem statement.....	4
Solution.....	4
Scope	4
Target users	4
User Research.....	5
Research Questions	5
Methodology	6
Key Findings.....	8
System Design	10
Overview / Goals	10
System Architecture	10
Information Architecture.....	10
Physical setup	16
User Interface.....	17
Information Visualization	17
Measuring success.....	17
Hydrobase Application	19
Dashboard	19
Devices.....	19
Plant profiles.....	19
Grows	19
Notifications	20
Emergency Stop.....	20
Data Export.....	20
Documentation.....	20
Login/Signup Page	20
Splash page.....	20
Computer Vision	21
Future work	22
Permutation Testing	22
Filters for dashboard	22
Crowdsourcing grow profiles.....	22
Other Outcomes	22
Experiments.....	22
Modular physical kit design.....	22

Search box on the plant profile page	23
Reflection.....	24
Appendix.....	25
User research.....	25
User Interface.....	29
Physical setup	38
Grow Progress	41
Electronics hook up diagram	42
Network Diagram	42

Acknowledgements

First, we would like to thank our advisor John Chuang for sharing his expertise, guidance and encouragement during the duration of this project. We sincerely appreciate the interviewees, survey respondents, and usability testers for sharing their time and valuable insights for a cold slice of pizza or nothing at all. Of course, we are grateful for our family, friends and significant others who showed us their support and exhibited patience during our late night hacks. Finally, the I School faculty, administration and community, a big thank you goes out to you for being so incredible. We couldn't have done it without you.

Introduction

Problem statement

Increasingly there are a number of different systems and technologies being developed to support urban agriculture, ranging from Grove Lab's Ecosystem to the MIT OpenAg initiative, each attempting to support effective food production. Yet the majority of these systems suffer from the same central problems, in that they require tight integration between the hardware and software. While this is certainly understandable due to the complicated nature of hardware and software interactions, it creates substantial barriers for entry. In the case of Grove Labs' ecosystem, the vertical integration allows for a relatively easy and intuitive user experience, yet this comes at the cost of preventing users the ability to make their own choices about the plants they are grow and what is done with the data. On the other end of the spectrum, the MIT OpenAg initiative allows users to customize their system in whatever way they see fit, yet requires substantial knowledge of electrical engineering, system design and fabrication techniques. Subsequently, we wanted to focus on developing a system that would allow users the ability to customize their own set up, without requiring they have an advanced degree. We feel by lower the bar for entry into urban agriculture we can more effectively support exploration and greater experimentation.

Solution

The solution we developed was to create an open cloud-based hardware agnostic control and data collection platform for hydroponic growing systems. We also developed a suggested, though not required, hardware system that works with the cloud-platform in order to both demonstrate the capabilities of the system as well as serve as a potential blueprint from which others could build upon.

Scope

While our project had a rather audacious scope at the beginning, the scope has been reduced as we have developed the system as well as learned more about the domain. Specifically, the project focused on supporting situations where accurate plant data collection and environmental maintenance is critical, such as research, commercial or experimentation applications. This is not to say that these are the only situations under which the system could operate, but through our user research we found these to be the most applicable use cases for such a system.

Target users

The target users for our system essentially fall into two different categories but both share a common characteristic of being heavily invested in growing plants.

Technically inclined, botanically disinclined: This individual is comfortable with information system but has little experience working with plants and does not fully understand what they need or how to effectively support them. This user is predominantly focused on exploration and experimentation as they are comfortable with data extraction and analysis but need an effective entry point for hydroponic growing.

Botanically inclined, technically disinclined: This individual understands what plants need and is interested in accurately recording data about the plants but does not have the technical expertise to set up an automated system on their own. This user is predominantly focused on leveraging technology to support their existing practices in either research or commercial applications.

User Research

Research Questions

Our initial research goals were focused truly stemmed from an initial assumption; that individuals who are doing hydroponic urban farming would be interested in automating as much of their growing as possible. From this assumption we began developing our initial goals focused on understanding user needs within the urban farming space. Our initial goals were predominantly focused on identifying current solutions, how users leverage them, and identifying potential opportunities for our system to improve upon existing solutions.

This focus on user needs and opportunities for differentiation subsequently allowed us to focus on three major questions for our research:

What are the key metrics users are interested in?

This question is focused on understanding what the users really care about from an informational standpoint. There are almost an infinite number of different aspects of a plant that could be tracked and recorded and so we wanted to ensure that we were not missing out on information that was critical to the users. Additionally, we wanted to make sure that our solution did not include extraneous or confusing information that the user is unlikely to need or even want.

What are the major pain points for users when using existing systems?

Here we are focused predominantly on developing empathy for the users. Running hydroponic grow systems is difficult and so we wanted to understand common problems users are running into with running and maintaining these systems. This serves the dual purpose of understanding where they are currently struggling and need help, and understanding what potential solutions we could provide via the Hydrobase system.

What role does technology play in their growing process?

This question was focused on understanding user's current technical abilities, experiences and typical tasks. We were attempting to understand if most users leverage technology to support their existing growing set-up or behavior and if so what they were using. This was important because we were using as a way to gauge both interest in technical solutions as well as understand proficiency with technology.

Each question was focused on attempting to understand a different aspect of a potential user's experience, the goal of which was attempting to help us develop different personas for Hydrobase. Yet as we went through the process, we started to see a pattern that we had made a rather substantial assumption regarding the system; that people want to use hydroponic systems like Hydrobase for their growing needs.

We had attempted to make sure we were recruiting individuals who were technologically inclined or had substantial knowledge of urban farming, which we felt meant that they would be a good fit for Hydrobase but as we did our research we began to realize that we needed to take a step back. So in addition to our original goals, we also began to try and understand at a more meta-level why people were growing plants in the first place.

Methodology

Survey

[Our survey](#) allowed us to reach out to a wide audience and helped us identify different personas based on needs and requirements. Since Hydrobase is developing a platform to empower small to medium scale urban growers, feedback from non-growers was not valuable to us and we only accepted participants with some growing experience. Since our target user base of enthusiasts and small-scale professionals is relatively niche, we realized quickly that finding a large number of qualified respondents would be difficult. We ended up with 15 survey responses. Ideally, we would have liked to have more, but nonetheless they gave us plenty of valuable feedback on people's growing needs, which was our goal. Our survey questions exhaustive and most of the questions were open-ended. This left us with a lot of valuable information. However the disadvantage of this survey structure is that it was probably too time consuming for many people to fill out.

Recruiting

To recruit for the survey, we turned to the UC Berkeley I School and Reddit. At the I School we used the localannouncements@ischool.berkeley.edu to request responses. On Reddit we found several relevant subreddits including:

- [IndoorGarden](#) - 13,832 subscribers
- [Hydro](#) - 7,523 subscribers
- [Urbanfarming](#) - 7,353 subscribers
- [H2Grow](#) - 479 subscribers
- [UrbanAgriculture](#) - 167 subscribers

We ended up only posting to the top three because they had the most subscribers. Below is a map showing the locations of the survey respondents.



The cluster in California is mostly I School students. Reddit fortunately provided much more geographic diversity, which was helpful because people in different climates and urban areas typically have different growing habits which we wanted to learn about. The other unexpected benefit of Reddit were the [comments on our post requesting survey responses](#). There people expressed concerns about modularity and pricing which are both important for us to understand.

Question Design

The survey questions were set to start out easy to build up a rapport between the participant and the researcher¹. In the beginning were close-ended questions about what they were growing. These are easy and quick to answer and allow some buildup of momentum. Below is the general flow of topic categories:

- What is the participant growing?
- What motivates the participant to grow?
- What challenges does the participant face?
- What is the role of technology in the participant's process?

We did our best to phrase questions with simple unambiguous words so the users did not have any unnecessary mental strain. The types of questions covered were characteristic, behavioral and attitudinal. All were either freeform answers or Likert Scale.

Interviews

One of the questions on the survey asked if the participant was willing to engage in an interview, and if they were, to provide contact information. We used this information to recruit interview subjects.

Interviews allow us to gain a deeper understanding of user needs. Although we used a question guide, purposely let the conversation meander and take us where it might. As is typical with one-on-one interactions participants seemed to open up on a personal level more than they did on the relatively impersonal survey.

Interview Design

We followed our [interview guide](#) and started by thanking the participant and then proceeded to give them context for the session. They are told we are a graduate student research group developing technology to grow plants that are healthier to consume and have a smaller impact on the environment. We emphasized how important it is for us to learn about real growers' processes, what motivates them and what their pain points are. Similar to the survey, we started out with easy close-ended questions, to build momentum, then proceeded to delve into more personal value based open-ended questions. We then concluded by profusely thanking them for their time and let them know how important their feedback was to our product.

Usability studies

Experiment Structure:

- Justinmind medium Fidelity Prototype of Hydrobase software
- In person usability tests, scoped to be approximately 15 minutes each.
- Focused on a typical / core task with the system; adding a "grow"
- Selected 5 participants, 2 women and 3 men
- All were technically inclined but did not necessarily have a green thumb.

¹ "Question and Questionnaire Design", by Krosnick, J.A., & Presser, S., from Marsden, P.V., & Wright, J.D., Handbook of Survey Research, 2nd Ed. (2010)

- One moderator and one note taker.

[The test](#) was designed to be completed within 15 minutes with clear sub-tasks, such as setting a condition for the grow, and was conducted by two members of the team. One team member moderated the test, using a script to ensure consistency between the different tests, while the other took notes on the test.

We were specifically looking for problems understanding how to:

- Instantiate a new grow
- Give the grow a name
- Associate the grow with a circuit board
- Associate sensors with the specific grow
- Associate sensors with the specific grow
- Set a condition that the system needs to respond to (pH getting below a certain point)

Key Findings

Overall respondents were supportive and enthusiastic about the Hydrobase system and were interested in beta testing future iterations. Below are the key findings from the survey, interviews and usability tests.

User needs

On Motivation

Watching life happen right before my eyes, something that's been possible for billions of years, to unfold in front of me, it's fascinating to me and a humbling experience. [...] And the eating, that's definitely a part of it. -Adrian

There is so much waste that happens in the [agriculture] industry. I do as much as I can without relying on that sort of system. It's a local mindset. -Lauren

Gardening generally made the survey respondents feel relaxed. They feel accomplished when something they've nurtured survives. Also many people, especially in California, are motivated by environmental concerns. This may be due to the recent drought. Several respondents expressed interest in using less water and energy. Furthermore, some find comfort in knowing that if the grocery store closed down, they would be able to survive on the food that they grew themselves.

On pain points

The challenge for me is that there is no knowledge-based growing. Information is scattered. -Libo

Every once in awhile the plant will get yellow leaves and look like it's dying and it's not clear why it's happening. Sometimes I give it a different light source but it's a process of elimination. -Andy

My challenge is Albany [CA] is close to the shore and is pretty overcast. I used to grow in the yard. -Dale

Several people states that testing the conditions of their plants was time consuming and they wish they could offload it through some automation. People also shared that they sometimes get anxious about whether they are correctly taking care

of their plants and that they wish there was a more centralized repository of ideal growing condition knowledge. Experimenting on one's own is difficult because there is a long delay between action and feedback. It is hard to isolate what is causing a plant to wither or thrive.

Feature suggestions

Below are a few opinions users shared about specific system features:

- Collecting data on the plants is important than automation, though both are desired.
- Modularity of the system is key. People want to be able to start small with a growing kit and scale up from there without starting over.
- Growers want to be able to test daylight conditions to optimize the use of artificial light.
- People want to know if they are ahead, behind or on schedule to harvest
- It would be helpful to have tips about plant cleaning and maintenance. Also growers want reliable suggestions for getting rid of pests in a healthy way.

Usability

We ran usability studies with 5 different individuals using the [grows page on an interactive prototype of our software](#) created in Justinmind. The subjects were asked to walk through a typical task with the system; adding a "grow" to the system. A "grow" in our system is a digital representation of a plant that identifies which circuit board, sensors, time based controls as well as conditions for the plant. This is an integral aspect of the system as it essentially sets up all the rules for how Hydrobase will interact with a plant, and subsequently we wanted to ensure our proposed solution was clear and understandable for the users. One of the key questions we were trying to answer was whether or not we had reached an appropriate level of abstraction so that the user could understand what was happening without knowing every little detail of everything was being handled.

For the most part, participants were successful at achieving their task. All but one of them were able to complete the task within the allotted time, and completed all tasks they were given without any prompting or guidance by the moderator. Regarding the individual who did not complete the task within the allotted time, the main reason they did not complete the task was because they felt they had completed everything despite missing some steps. This subsequently forced us to question whether or not the individual fully understood what was being asked of them (essentially a problem with the usability test) or there was perhaps a communication problem as this individual was not a native English speaker. Nonetheless the other usability tests suggested that we were on the right path, though there was still some room for improvement.

The largest problem seemed to be surrounding terminology and ensuring that participants understood labels and how to mentally map the labels to what they wanted to achieve. A specific example of this would be when the participants were asked to conditionally activate an actuator, in this case a pH increase pump. The terminology of an actuator was foreign to many of the participants and so it was difficult to map this term to the label we had used in the UI, "Doing Stuff". While this was not the case for all, with one participant noting that he liked how we had been less technical in the UI, it was a consistent enough of problem that the development of a controlled vocabulary would appear to be helpful for users.

System Design

Overview / Goals

The main goal of the system design is to support interoperability and while this may seem like a trivial task, we identified it early on as one of the most important aspects of making our solution as open as possible. With this central goal in mind, all of our other decisions were focused on facilitating flexibility in every aspect of the system. Subsequently we chose to use open-source hardware, a highly extensible database and micro-application framework to ensure that the system was capable of adapting and evolving over time. This flexibility in the system turned out to be extremely important because even in the short period of time we have been working on this project, the landscape for open agriculture technology has changed substantially.

System Architecture

- Electronic

The electronic system design was focused on using “off the shelf” components for a number of reasons, but the main one being to support user success. We wanted to ensure that users could build the same system themselves if they so desired. With this in mind we did some research on different platforms, their capabilities, availability, cost and also their support community to understand what systems potential users are likely to be familiar with. From this research we chose to use an Arduino Mega as it was capable of working with a number of different peripherals in the Arduino suite of products, offered ample I/O and more than enough RAM for the applications we would be running on it. Additionally, Arduino is a very popular brand of open-source hardware with a strong community supporting it, and subsequently we felt confident suggesting it to potential users.

Once we had the main board selected we began choosing applicable sensors and actuators. Again we wanted to work with open-hardware to ensure users would have a similar experience, there by giving us the best confidence possible that users would be successful. We focused on what we felt would be important information such as air temperature, electrical conductivity, pH, lux as well as important actuators such as water pumps and peristaltic pumps. We specifically chose to keep the sensors and actuators as minimal as possible, again to try and ensure users would be successful without spending a fortune on their system.

Information Architecture

We used mongoDB to store all the data to power the hydrobase application. The database design for hydrobase was pretty complex and we did spend a lot of time working and reworking the database design so that we could optimize access to the database and with short queries that would allow the application to run faster. In the end we split the database into the following collections:

Users

The user collection contained the information about the user. Whenever a user signs up with Hydrobase, a document is created in the User collection for that particular user. Here is a sample documents in the users collection

```
{
  "email": "iotmims16@gmail.com",
  "username": "admin",
  "password": "pbkdf2:sha1:1000$JifYdXt6$258a0e3729d059d292368dfd0ccdb290858285ce",
  "zip": "94703"
}
```

The username assigned to each user is unique and also serves as the channel name on pubnub for all the grows/devices associated to that particular user to communicate on.

We hashed the password at the backend before storing into the database in order to improve security. While signing in this password is converted back to plaintext and then compared with the one entered by the user.

Devices

The devices collection contained the information about the devices. Whenever a device is added to the application, a document is created in the Devices collection. We assigned each device an "id" which was randomly generated GUID. Sample document from the devices collection

```
{
  "actuators": {
    "light_2": "31",
    "light_1": "30",
    "phUpper_pump": "35",
    "nutrient_pump": "33",
    "water_pump": "32",
    "phDowner_pump": "34"
  },
  "type": "Arduino",
  "device_name": "Wario",
  "device_id": "202d31e6-aa8b-4fe6-b479-285362ea6606",
  "kit": "standard",
  "emergency_stop": "false",
  "username": "admin",
  "sensors": [
    "Lux",
    "Water_Temp",
    "Air_Temp",
    "Humidity",
    "pH",
    "EC",
    "TDS",
    "PS"
  ]
}
```

As we can see above the devices document contain device information like name, identifier, kit, emergency stop, user the device belongs to and sensors and actuators associated with the device.

Grows

The grows collection contains information about the different plants that belong to a particular user. Whenever a new grow is added through the application, a document in the grows collection is created. Sample document in the grows collection

```
{
  "username": "admin",
```

```

"experiment": "false",
"grow_name": "Lily",
"actuators": {
  "light_2": "31",
  "light_1": "30",
  "phUpper_pump": "34",
  "phDowner_pump": "35",
  "water_pump": "32",
  "nutrient_pump": "33"
},
"sensors": [
  "Lux",
  "Water_Temp",
  "Air_Temp",
  "Humidity",
  "pH",
  "EC",
  "TDS",
  "PS"
],
"device_id": "68dc6c7e-8c19-499b-b167-5bd4fe37ff3d",
"plant_profile": "capsicum_annuum_'jalapeño'",
"controls": {
  "condition": [
    {
      "value": 6,
      "operator": "<",
      "action": "on",
      "actuator": "phUpper_pump",
      "sensor": "pH",
      "unit": "pH"
    },
    {
      "value": 6,
      "operator": ">",
      "action": "off",
      "actuator": "phUpper_pump",
      "sensor": "pH",
      "unit": "pH"
    },
    {
      "value": 6.8,
      "operator": ">",
      "action": "on",
      "actuator": "phDowner_pump",

```

```

        "sensor": "pH",
        "unit": "pH"
    },
    {
        "value": 6.8,
        "operator": "<",
        "action": "off",
        "actuator": "phDowner_pump",
        "sensor": "pH",
        "unit": "pH"
    },
    {
        "value": 600,
        "operator": "<",
        "action": "on",
        "actuator": "nutrient_pump",
        "sensor": "EC",
        "unit": "EC"
    },
    {
        "value": 700,
        "operator": ">",
        "action": "off",
        "actuator": "nutrient_pump",
        "sensor": "EC",
        "unit": "EC"
    }
],
"time": [
    {
        "action": "toggle",
        "actuator": "light_1",
        "value": 12,
        "unit": "hours"
    },
    {
        "action": "toggle",
        "actuator": "light_2",
        "value": 12,
        "unit": "hours"
    },
    {
        "action": "toggle",
        "actuator": "water_pump",
        "value": 15,

```

```

        "unit": "minutes"
      }
    ]
  },
  "device_name": "Mario"
}
]

```

A document in the grows collection contains information about a particular grow like name, device associated to it, sensors and actuators linked to that grow as well as the various controls both time and condition based for that particular grow.

Plant Profiles

The plant profiles collection contains the information about the various profiles that we sourced from USDA and other reliable sources and that are available to the user to create their grows from. Sample document from plant profiles collection

```

{
  "fruit_nut_seed_product": "Yes",
  "fruit_nut_seed_color": "Green",
  "drought_tolerance": "Low",
  "shade_tolerance": "Intermediate",
  "precipitation_minimum": "Medium ",
  "ph_maximum": 5.8,
  "foliage_color": "Green",
  "height_mature_feet": 8,
  "precipitation_maximum": "High",
  "growth_period": "Spring/Summer",
  "bloom_period": "Autumn",
  "flower_conspicuous": "Yes",
  "ph_minimum": 5.2,
  "growth_rate": "Medium",
  "scientific_name": "Citrullus lanatus",
  "temperature_minimum_f": 45,
  "identifier": "citrullus_lanatus",
  "salinity_tolerance": "Low",
  "lifespan": "Moderate",
  "fruit_nut_seed_conspicuous": "No",
  "fruit_nut_seed_begin": "Autumn ",
  "fruit_nut_seed_end": "Fall",
  "common_name": "Watermelon",
  "fruit_nut_seed_abundance": "Medium"
}

```

Data

The data collection contains the data that is gathered by the various sensors on the devices. Each packet of of data that is published by the device to pubnub is added to the data collection as a document. The timestamp is also appended along with the other information in order to identify the time at which that data was captured.

Sample document from the data collection.

```
{
  "airTemp": 75,
  "hour": 9,
  "min": 11,
  "lux": 45,
  "PS": "0.35",
  "EC": "723.8",
  "humidity": 33,
  "sec": 36,
  "year": 2016,
  "TDS": "390",
  "grow_name": "Lily",
  "waterTemp": 74.1875,
  "month": 5,
  "pH": "6.598",
  "day": 6,
  "device_id": "68dc6c7e-8c19-499b-b167-5bd4fe37ff3d"
}
```

So the document contains all the sensor information along with the timestamp and the device and grow the information is associated with.

CV Data

CV data is published once a day and contains the data that we gathered using computer vision. Sample CV document.

```
{
  "CV_Data": {
    "sample_3": {
      "time_stamp": "2016-05-06 05:03:04.340355",
      "foliage_density": {
        "fd_raw": 17.218166666666665,
        "fd_increase": 0
      },
      "grow_name": "Lily",
      "height": {
        "height_increase": -7,
        "height_raw": 78
      }
    }
  }
}
```

```
}
```

PubNub Communication

We use PubNub, “the global data stream network for IoT, mobile, and web applications,” to send data to and from our sensors. It is what controls the information flow. Devices can both subscribe and publish to channels, which enables “reading” and “writing,” respectively.

To store the sensor data, the application subscribes to the channel and sends what it receives to the database.

In order to automate actuator behavior based on user-specified settings, the device listens for messages in the following format:

```
{  
  "68dc6c7e-8c19-499b-b167-5bd4fe37ff3d": {  
    "30": 255,  
    "31": 255  
  }  
}
```

This includes the device ID as well as pin numbers and associated values. The pin numbers correspond to actuators and the values instruct the device to them turn on or off. In our setup, off is represented by 0 and on is represented by 255.

These values are determined by the user-specified of the default settings. For example, a user can set the lighting schedule to something like 12 hours on and 12 hours off. We call these controls and work with two types: time-based and condition-based controls. Time-based controls only depend on time while condition-based controls depend on data.

Typically, lights and water pumps will be set up using time-based controls. Nutrient and pH pumps, on the other hand, will be set up using condition-based controls. With pH, an action, such as increasing or decreasing the pH, is triggered based on the most recent data. For example, a user may specify that pH not go below 6.5. In this case, if the most recent sensor read is below that value, the pH “upper” is triggered and the pH level of the solution is brought back into an acceptable range.

Physical setup

We optimized the [physical setup](#) for cost, strength and simplicity. For \$500 - \$1000 someone could replicate what we’ve built, the lion’s share of that cost being the sensors. Early on in this project we had discussed using 3D printed parts for the structure but later decided against that since we could accomplish what we wanted for less money and time buying parts from Home Depot and Amazon. In the appendix we show [early concept drawings of our structure](#).

The framing itself is built out of PVC pipe and PVC pipe connectors. Other parts of the system are common plastic buckets, rubber tubing and twine. The documentation on the Hydrobase web application provides detailed instructions on what to buy, how to assemble and even provides a [downloadable 3D model](#) which specifies drill hole locations.

We did have some issues with leaking when the [aeroponic sprayers](#) were on for extended periods of time but were able to resolve the problem for the most part with copious amounts of glue gun glue. In the future we would want to design a modular screw/snap-together grow system that lets growers scale up over time

User Interface

Initially the UI was [designed for mobile devices](#) with the notion that growers may want to track their grows remotely. Although this was still a secondary goal, based on the amount of information that should simultaneously be visualized and the complexity of the tasks being executed, desktop web seemed like the best primary platform. The first [low-fidelity desktop mockup of the web application](#) was created in Balsamiq. The “sketchy” nature of this prototyper made it ideal for evaluating what layout works best and which high level functions are important without getting caught up with color preferences, typefaces and other polishing touches. Furthermore, finally having a graphical representation of what we had been discussing in the abstract for weeks made our conversations more concrete and also gave us a better sense of what scope we could handle for this project.

After agreeing upon an updated high-level organization, we transitioned to an [interactive prototype in Justinmind](#). This prototype was useful for showing off what our web application would eventually be capable of. Additionally, since it was interactive, it was perfect to use in our usability studies. It included:

- An animated dashboard with filters
- A devices page for adding circuit boards (platforms) with associated sensors and actuators
- A grow page where the user could set up new grows, link to specific sensors and actuators, set up automated controls over a specified recurring time period based on a Boolean conditions and add photos

We were able to quickly learn and iterate using the prototypes so by the time we started coding the actual web application, we had a much better sense of what we wanted to build and how it should be organized.

Information Visualization

The information visualization was simultaneously extremely important and extremely boring. The problem with the information visualization was that if everything was working perfectly with the system, there was actually very little change or variation to display to the user via information visualization. Subsequently we developed two different types of visualizations. The visualizations on the “grow” pages became “sanity checks” to ensure that the system was doing what it was supposed to and that the user did not need to intervene. In order to accomplish this, we leveraged the data we had about the different plants to illustrate a “safe zone” within which the plant would be comfortable and displayed it as a range within which the data points should lie.

The real time dashboard on the other hand had a slightly different use case as we wanted to let users understand quickly what was happening at that moment with all of their devices, and so we developed a dashboard that displayed the 10 most recent data points and would continuously update itself as new data came in. This is effective to understanding the current conditions of the system and potentially could aid in either setup or system modification as the user is able to get immediate feedback from the system.

Measuring success

The general factors that we consider in order to measure success are the growth in height of the plant and increment in foliage density. However, these factors are qualitative and we explored using image processing and computer vision techniques to measure these factors quantitatively.

The 2 factors that we thought about to quantitatively define success over a certain period of growth time of the plant are

1. Increase in height

2. Increase in foliage density

For this purpose we have a setup. We use a raspberry pi board with a compatible camera that is fixed in its place. We installed the raspberry pi-camera and OpenCV libraries on the raspbian operating system. The user needs to run our module (`hydrobase_cv.py`) and keep the camera fixed in a stable location. The camera is adjusted to focus on the image whose growth needs to be measured. We take 3 consecutive pictures of the plant every 12 hours when the light is natural and the leaves seem to be in their normal color. For our current setup we used a Jalapeno plant, which we nicknamed 'Lily', and fixed our camera on it. At present, we are only considering green leaved plants for our Computer Vision application, since more than 98% of the plants in our current database have green foliage. Since, we are using growing lamps to grow the plants as a part of our hardware setup, we need to take pictures of the plant in normal surroundings when there is no artificial glow on the plant.

We then run algorithms on the collected images to measure the growth in height and foliage density of the plant over a certain period of time.

Hydrobase Application

Dashboard

The [dashboard page](#) on the application is an aggregate page that allows the user to view all his grows in one go. The dashboard page lists shows graphs for pH, electrical conductivity, temperature, humidity and lux which display the real time data that is coming in from the sensors. Thus the users know the status of their grow in real time. The dashboard also allows the user to compare different grows against each other with the help of graphs.

Devices

In order to effectively managed the different plants our users would be growing, we needed to ensure that we had an accurate representation of the devices (microcontrollers) that were controlling the plant environments. Since we were allowing for interoperability of different devices, we needed to create a way for users to explain their hardware to the system. Subsequently we created a section of the application where users could manage the devices in great detail.

The [devices page](#) allows users to add new devices, identifying a number of different critical aspects of the device such as the type, associated sensors and associated actuators. Additionally, for the actuators the users are able to identify which pins the actuators are connected to, thus allowing Hydrobase to activate them and either turn pumps / lights on or off. Finally for the more technically inclined users, the devices section gives the user keys to the pub/sub system we are using and their specific channel, making it possible for them to connect many different devices to the system without Hydrobase needing to be specifically programmed for the device.

The devices page also is used to help users who are less technically inclined as it offers the option to use our pre-designed kit and gives the user the code required to connect to Hydrobase, send data and receive commands from the system. Again this was done in an explicit attempt to support both the technically inclined as well as the less technically inclined.

Plant profiles

We collected most of our [plant profiles](#) from the US Department of Agriculture data that is available online. We have a collection of more than a thousand plants and we have several parameters that we keep in our database. The parameters that we consider significant are scientific name, pH range, precipitation range, temperature and drought tolerance, growth rate and period, average mature height, lifespan, relevant flower/fruit/nut/seed attributes, and foliage color. There were nearly 100 plant species as well that are quite commonly grown by people but that were missing from the USDA database. We had to scrape the relevant data from them using online resources, such as Google search. We also collected pictures of these plants using Google search that had the license rights for using them without restrictions thereby preventing any copyright infringement claims. All the data that we collected from USDA and various other reliable sources was cleaned and massaged to remove any irregularities and then inserted into the database. Each one of these profiles is available to the user to create a grow from.

Grows

[Grows](#) corresponds to an instance of a plant that the user creates in the system. A grow can either be created from a predefined plant profile which will pick up the default values for various parameters like minimum and maximum pH, electrical conductivity and air temperature from the plant profiles derived from the USDA data. Or a grow can also be customized as per user allowing them to set the ranges for the above parameters manually. This is done by setting a controls that are either time based or condition based. The time based controls perform an actuator action like turn the light on or turn the water on on a timely basis like every 12 hours. On the other hand the condition based controls perform an action only when a particular condition is met for example turn on the pH Upper pump if the pH levels fall below the range specified by the user. This allows the user to create grows with custom profiles which can later be compared to the default profiles provided by USDA. The user can also opt to participate in an experiment while creating a grow. What this option does it allow

the user to set the range of values for various parameters like pH and electrical conductivity within a specified range of the defaults provided by the USDA for that plant. The grows that participate in experiments have a special flag that is set to true so that the system can identify those plants use the data from those plants for testing purposes.

In addition to all these flexibility provided to the user while creating a grow, the application also provides the user the flexibility to change the sensors and actuators linked to a grow as well as add, remove or change the time and condition based controls that have been applied to a particular grow.

Notifications

We implemented a way to notify the user when for any grow that belong to the user either the time based control or condition based control is tripped. Although the system can rectify the condition that was tripped by turning on or off the associated actuator but the user is still notified so that they can dig deeper and find out the root cause for the same.

Emergency Stop

In the application we provided the user a capability to stop a device all-together meaning that all the actions that had to be performed by that device would be temporarily suspended until the user releases the device from the [Emergency stop](#) state. This was not planned in initially but one of the team members wanted to take some close up pictures of the plant roots thereby exposing the water nozzles. This had to be planned very carefully in order to prevent the device spraying water solution while taking those picture. We figured out the users might want to perform certain actions once in awhile that required the devices to be suspended and hence we implemented this feature.

Data Export

During our user research one of the main feedback that we got from the users was that most of the existing hydroponics setups either did not capture and store the data at all or if they did the data was not exposed to user. We provided the user with a way to download all the data for their grows as JSON file that could be loaded into other data analysis and visualization tools.

Documentation

It quickly became clear based on the complexity of our system that good [documentation](#) was a must. Buying the wrong part or missing a single connection could be the difference between the system running smoothly or not at all. For the structure we created a [3D model](#) so people could see clearly what the end product should look like. It also included all the drill hole locations for people to connect aeroponic tubes and hang the buckets. The electronics section provides a detailed diagram and instructions of how to connect the sensors and actuators.

Login/Signup Page

The login and signup pages were simple HTML forms that asked the user for various inputs like username, email, password, name and zipcode. We performed some error checking on the page so that the email was in proper format and password had some minimum requirements. The password was hashed and stored in the database for the purpose of security.

Splash page

We used a simple freely available bootstrap framework to model our splash page. We have the simple pages for signing up, login and product details embedded on the splash page. We have a few pictures of our setup that illustrate how we went about developing our setup. Additionally, we have a documentation page that describes the details our setup in an exhaustive manner.

Computer Vision

Here we explain some of the logic in the algorithms that we use for implementing the image processing and computer vision component of our setup that is used to calculate the growth in foliage density and height of the specific plant under consideration.

We generally take 720p (1280 x 720) resolution images of the plant with the raspberry pi camera. The camera should be set up in such a way so as to make the captured image include the complete plant with a substantial space (≥ 200 pixels) between the top of the image and the top part of the plant to help in measuring effective growth in height of the plant. It is fine if a portion of the bottom region of the plant gets clipped since the increment in height is always measured from the top.

We use a color spectrum clustering technique to identify the regions of the image that qualify as the viable portions of the plant. Using the color spectrum clustering technique, we group pixels in a binary manner as either portions of foliage or non-plant portions. We then use a morphological image processing and connected components based segmentation method to fill out missed regions in the images as part of the foliage that could have been eliminated in the initial color spectrum clustering method. Then in order to measure height of the plant from the top part of the image, we measure the beginning of the foliage region as measured from the top of the image in a robust manner to avoid any false measurements. This value is then compared with the value for the benchmark image to find out the increment in height of the plant.

In order to reduce the impact of noise, we average the values for height and foliage density increments over 3 images that are taken at an interval of 3 seconds after a normal measurement time of 12 hours. We finally wrap these values in a json object that is then published to the PubNub channel specific to our plant.

To collect data, at first the camera will capture the first set of images to set a benchmark. Then it will keep on capturing 3 consecutive images every 12 hours and would compare it to the benchmark. These images are stored in the memory drive running the raspbian operating system. We have a centralized SOA architecture for taking continuous measurements using our hardware and software setup. Here the raspberry pi module works as a client. It does algorithmic processing to calculate the quantitative results followed by publishing them to the same PubNub channel that is used for publishing the other solution parameters specific to that plant, such as the pH and EC measurements.

The specific json object format in which the data is published to the PubNub channel is mentioned below.

```
"CV_Data": {
  "sample_23": {
    "time_stamp": "2016-04-27 04:08:19.160730",
    "foliage_density": {
      "fd_raw": 13.523666666666665,
      "fd_increase": 2.9573200000000006
    },
    "height": {
      "height_increase": 5,
      "height_raw": 87
    }
  }
}
"grow_name": "Lily"
```

Future work

Permutation Testing

In order to determine which grow profiles yield the best results, we have started to develop a permutation-testing-based method for assessing each profile's efficacy. A benefit of permutation testing is that it does not rely on assumptions about how the data are distributed or about independence between observations. While there are several outcomes that users might care about, such as number of fruits or taste, we focus on biomass. For this, we leverage the CV data. It's important to note that, once implemented, the comparisons will only be made between the same type of plants.

A challenge with comparing grow profiles is in the number of possible combinations that are possible. For example, one user might specify particular lighting and water cycles in conjunction with specific upper- and lower-bound pH levels while another might specify completely different settings. To alleviate this issue, we reduce the number of choices that can be made for each actuator. We have yet to implement a way to differentiate between (i.e., label) various profiles.

Filters for dashboard

Understanding that some users might have multiple grows and that information overload is counterproductive, we plan to implement filters for our dashboard. This will allow users to toggle the visibility of grows or sensor data.

Crowdsourcing grow profiles

One of the challenges with agriculture is that machine readable data is not readily available. There is also a lot of misinformation about the type of conditions that produce the best results. Once our platform scales, we will have access to profile and outcomes data from large numbers of users, which will allow us to determine the settings that are most effective for plant grows. Similar to how large communities of people work together to make reliable Wikipedia articles, we hope to create the most comprehensive and reliable repository of plant recipes using the crowd.

Other Outcomes

Because a lot of what users care about is qualitative, such as fruit taste, we would like to incorporate user feedback on those attributes. This would allow us to determine which profiles are most suitable for optimizing those outcomes.

Experiments

A valuable feature, especially for researchers, that we would like to implement is the ability to run experiments. Users might want to compare nutrient solutions, for example, or compare lighting schedules on their effects on specific outcomes.

A valuable experiment, brought to our attention by an owner of a local hydroponics store, is to assess nutrient uptake. Many nutrient solutions claim to be highly effective. However, there isn't much data to support those claims. A simple approach to this is to measure the EC at the time the nutrient solution is added and compare it to the levels after the plants have had a chance to absorb it.

Modular physical kit design

As we have stated a number of times, a main focus of our system was to allow for exploration and experimentation in either the plants or the hardware and we feel that modular physical kits could play a critical role in this. We did choose our system to ensure people had everything they need but not more, yet as we did our user research we realized how substantial a barrier the "unknown" was. Modular physical kits would allow users to "get their feet wet" before committing to a more expensive system, thus mitigating to a degree this barrier of the unknown.

Additionally, our conceptualization of these physical kits is that they would be decomposable, allowing users to dismantle them and rebuild them in whatever fashion was most applicable to their needs. This serves a dual purpose of ensuring users are not being “locked in” to different products, thus further decreasing the uncertainty barrier, as well as allowing users to develop systems we could never have imagined. Unfortunately, we were unable to accomplish much more in the way of kit development outside of our initial system, but this is an area a number of the team members are eager to explore.

Search box on the plant profile page

We have about 400 plant profiles that we have sourced from various data sources and carefully curated them into plant profiles from which a user can set up a particular grow. We would like to implement a search text box that would allow the user to search for a grow based on the common name and pull up a plant profile to create a grow from.

Reflection

The entire process of building both the software application as well as the hardware has been incredibly educational, certainly the most informative of our time at the School of Information. We have been able to incorporate not only our technical skills but also our information organization skills, communication skills as well as socio-technical skills in an attempt to develop a different approach from what we are currently seeing. This project truly felt like the culmination of our education, an effective summary of what we have learned.

Yet none the less the project has raised more questions than it has answered. We certainly feel that technology similar to ours could be incredibly helpful and impactful in the future but it seems to be such early days that its exact form still appears quite uncertain. In this way, we felt it was best for us to explore a potential solution, as opposed to being so bold as to say that we have developed the ultimate solution. This more light hearted interpretation allowed us to meet with a wide range of individuals from local entrepreneurs to professors of botany, and truly emphasized the notion that a purely product driven perspective is not always the best approach for understanding a problem and its potential solutions. Subsequently we are extremely pleased to have had the opportunity to explore and work hard on a problem we feel is worthwhile.

From a hardware perspective, the increasing availability of “off the shelf” open-hardware will only make it easier to build these systems, increasing the pace of exploration and discovery. For this reason we feel it is critical to incorporate data capture and analysis systems to ensure that the collected data is not lost and so that knowledge and information can be effectively shared with a larger audience. Nonetheless there are strong economic forces at play that have an incentive to leverage lock-in, meaning that without strong advocates in the open-source community the information could be difficult or impossible to access for the greater public. Subsequently we would like to continue to work with open-source advocates and organizations to help develop platforms that facilitate exploration, experimentation and discovery.

We feel that using image processing and computer vision techniques to get further quantitative results about the growth of the plants is a good way to automate the performance measurement of any hydroponics setup. At present, we are focusing on height and foliage density growth over a certain period of time. In the future we would like to detect anomalies in the foliage to alert the user regarding possibility of an infection, nutrients deficit, or incorrect setup that could be possibly damaging the plants. For example, detecting increment in brown spots in green foliage might signal the spread of an infection or deficiency in nutrients.

Our sole purpose of integrating IOT measurements with computer vision and image processing is to automate the process of taking care of your plants in a hydroponics setup in a seamless manner. Our value proposition to the user is reduced hardware cost, coupled with an intelligent setup platform that provides better automation than any other solution out there, and this translates to a better user experience with substantial user time savings.

Interested in automating or optimizing your hydroponics setup?



We'd love to talk to you!

We're a group five of UC Berkeley graduate students developing a user-friendly hydroponic system that's both automatically controlled and cross-platform compatible. We're looking for small to medium scale growers to interview for 45 – 60 minutes to better understand our users' needs. Our goal is to advance urban agriculture and agricultural experimentation, so you'd be contributing to a system that enables growers to work more locally and more efficiently.

Please e-mail japple@berkeley.edu if you are willing to contribute an hour of your time.

hydroponics automation
japple@berkeley.edu

Survey

Growing System User Needs

Hydrobase is a flexible and visual platform for growing plants that are healthy for you and sustainable for the environment. With intuitive controls and analytics, our aim is to make the urban hydroponics growing process smarter and more fun! Right now we're focused on learning more about our prospective users so we can better serve their needs. We appreciate you carving out the time to provide thoughtful answers to this survey.

* Required

Name

Your answer

E-mail *

Your answer

Location

City, State

Your answer

What most interests you about hydroponics?

e.g. it uses less water than growing in soil, it uses less energy than traditional farming and distribution, high nutrient levels in plants, consistency of plant output, growing year round, advancing knowledge of ideal growing conditions, technology is fun

Your answer

What kind of plants are you interested in growing?

e.g. tomatoes, basil, kale, orchids

Your answer

For what purpose are you growing?

e.g. personal use; sell to grocery stores; use in my restaurant

Your answer

How much do you want to grow?

e.g. enough spinach for 3 salads per month; 3 plants; as much as my apartment can hold

Your answer

Are you currently growing plants? If so, what are you growing and what is your setup?

Your answer

How much space do you have to grow indoors?

Provide length, width, height estimates if possible

Your answer

If you have indoor space that could be used for growing, does it have access to sunlight?

Your answer

How much space do you have to grow outdoors?

Provide length, width, height estimates if possible

Your answer

How much gardening experience do you have?

Your answer

How does gardening make you feel? Why?

e.g. frustrated, relaxed, happy...

Your answer

Hydrobase is a web-based application that connects to a physical hydroponics (growing plants without soil) system. Would you be more interested in connecting this platform to a standard kit or linking it to your own customized system?

Your answer

Rate These by level of importance

	Not important	****	Somewhat important	***	Very important
I want to use automated controls to give light to my plants	<input type="radio"/>				
I want to use automated controls to provide a nutrient solution to my plants	<input type="radio"/>				
I want to use automated controls to provide carbon dioxide to my plants	<input type="radio"/>				
I want to use automated controls to maintain the pH of my nutrient solution	<input type="radio"/>				
I want to use automated controls to maintain the ratio of nutrients to water	<input type="radio"/>				
I want to be able to collect data on pH	<input type="radio"/>				
I want to be able to collect data on humidity	<input type="radio"/>				
I want to be able to collect data on light levels	<input type="radio"/>				
I want to be able to collect data on temperature	<input type="radio"/>				
I want to be able to collect data on solution nutrient levels	<input type="radio"/>				
I want to be able to collect data on bacteria levels	<input type="radio"/>				
I want to be able to compare the success of similar plants under different growing conditions for optimization purposes.	<input type="radio"/>				
I want this system to be modular so I can gradually scale it.	<input type="radio"/>				
I want to be able to browse what other people are growing and the conditions they grew in.	<input type="radio"/>				
I want to be able to browse a library of standard plant profiles (e.g. for chili peppers or lettuce) that apply optimal settings to my growing controls	<input type="radio"/>				

Are there features that have not been mentioned that you think this system should have?

Your answer

How would you describe your level of comfort with technology?

Your answer

How much experience do you have working with microcontrollers like Arduino or Raspberry Pi?

Your answer

How much would you be willing to pay for a physical hydroponics growing kit that allows you to automatically control lights, pH, nutrient levels, water pumps for multiple plants?

Don't worry you don't need to commit to this..)

Your answer

Assuming most of the features mentioned in this survey were enabled, how much would you be willing to pay for a monthly subscription to the Hydrobase control and analytics platform?

Your answer

Would you be willing to be interviewed about your growing needs?

If so, provide a phone number, or we could do a Google Hangouts call.

Your answer

Would you be willing to be a beta tester of our software?

Don't worry, even if you say yes now, you can opt out at any time.

Your answer

Other comments?

Your answer

Growing needs interview guide

Note to interviewer: if interviewee responded to [our survey](#), be sure to familiarize yourself with their responses before the interview and adapt your questions accordingly. Make it seem like we value their previous input.

Intro:

Hi, I'm <your name> Thanks so much for taking the time to do this interview. How's everything going today?

I thought I'd start by giving an introduction to our project to give some context to the questions I'm going to be asking. After that, we'll spend most of the time talking about you and your growing process. Does that sound good?

It will help me better focus on the conversation if I don't need to focus too intensely on taking notes. Would you mind if I recorded our conversation?

<Great, thank you> or <no worries, that shouldn't be a problem>

<start recording>

OK, I've started the recording. Thanks for agreeing to that <interviewee name>. So I'm <your name> and I'm working with a team of UC Berkeley graduate students on developing web-based platform for analyzing sensor data and controlling plant growth on a small to medium scale. It's called Hydrobase. Our aim is create technology that enables people to grow plants that are healthier to consume and have a smaller impact on the environment.

As we develop this system, we want to be sure we are addressing *actual* user needs, and are not just making up imaginary problems, which is where you come in. This interview serves to help us understand a real grower's process and how it could be improved upon. We want to learn about your pain points and also what motivates you. With that, I'll jump into the questions. Sound good?

Question guide (remember to go with the flow and ask follow up questions)

- What city are you based out of?
- Tell me about your grow setup.
- How much time each week do you spend gardening?
- Why do you grow? What are your primary goals?
- What challenges do you face while growing?
- Tell me about a recent positive experience you had while growing.
- Tell me about a recent negative experience you had while growing.
- What role, if any, does technology *currently* play in your growing process?
- If you could have one technology, and it doesn't need to exist yet, added to your growing process, what would it be and why?
- Do you ever run experiments with your grows? If so, tell me about that.
- How much money have you invested in your current setup?
- How do you measure success with your grows? Yield? Taste? They didn't die?
- How do you track growth? E.g. harvest time, when to water, pH levels?
- How do you do record keeping around your grows?
- Do you do any data analysis currently?
- Would you be willing to be a beta tester?
- How do you find out the ideal growing conditions for your plants? E.g. water, pH, light, nutrients
- How important is the aesthetic quality of your growing setup?

Conclusion

Those are all the questions I had. Is there any other information you think my team should consider as we develop this technology?

Is there any questions you have for me?

Thank so much for taking the time to do this interview. You were super helpful and we appreciate it! Feel free to follow up with any questions or comments by email. Take care, goodbye.

Grow setup usability test

Introduction:

I'm <moderator person> and I'll be acting as the moderator for this session and will be silent for the most part. If you truly get stuck during this process, let me know and I will give you some direction. <scribe person> will be taking notes. <recorder person> will be making a video recording of the session. This is not a timed exercise.

We are developing an analytics and automation platform to help people grow food more effectively. It's called Hydrobase. Our system has two parts, a physical setup and a web application. Today, we are asking for your help testing the latter. However since both of these components work together in tandem, we ask that you please use a bit of your imagination.

Imagine you want to grow an avocado plant and a chili pepper plant. You've bought the materials for the Hydrobase growing kit and followed our documentation to successfully assemble the structure and connect the sensors/actuators. There are 2 buckets for growing plants, one reservoir of nutrient solution and a "brain" containing a variety of useful electronics.

Now that you have the physical setup taken care of, you need to connect it to our web application so you can collect data on your plants and control their nourishment.

In this usability exercise, you will be using a prototype mockup of our web application to set up a "grow," connect it to the physical system and initiate an automated control. What we call a "grow" is a digital representation of a plant you are growing. You can collect data on a grow and you can setup automated actions to feed your grow.

A disclaimer: This is not a test of your intelligence. If there is something unclear about the software, that's an opportunity for improvement, not anything wrong with you. So just be honest and we ask that you please speak aloud as you go through our guide.

Instructions:

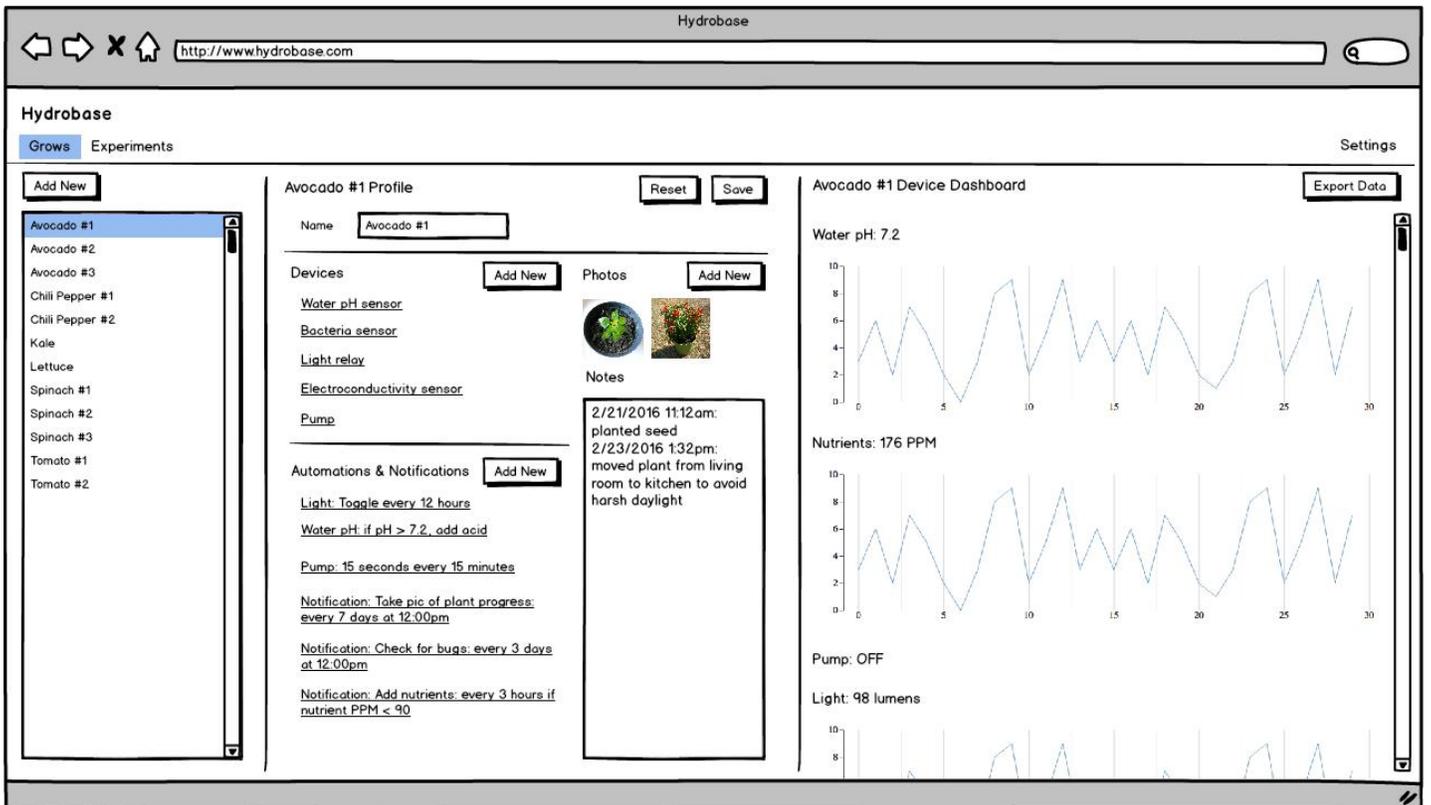
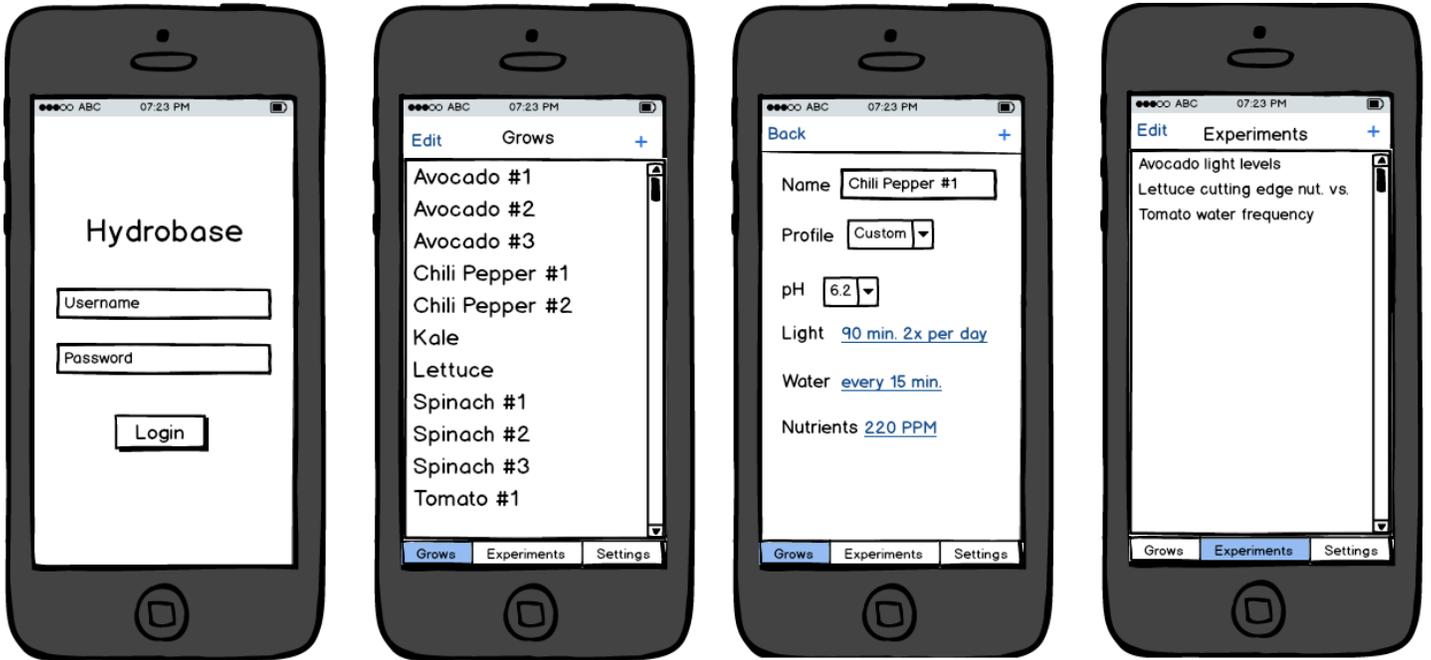
1. Open this link in a new tab:
<https://www.justinmind.com/usernote/tests/13167258/13167266/19415956/index.html#/screens/eb9d7895-8a8b-46ea-ae7-29b3e0de344e>
2. Create a new grow
3. Give the grow a name
4. Connect the following devices from the Arduino #1 circuit board:
 - a. Sensors: EC #1, Humidity #1, Light #1, pH #1, Temp #1
 - b. Actuators: Light #1, Nutrients, pH Decrease #1, pH Increase #1, Water Pump #1
5. Set up a control that indefinitely checks every 5 minutes if the pH is below 6.8, and if it is, it turns on the pH increase pump for 30 seconds

Conclusion

This concludes the exercise. Thanks so much for helping us out! Do you have any questions?

User Interface

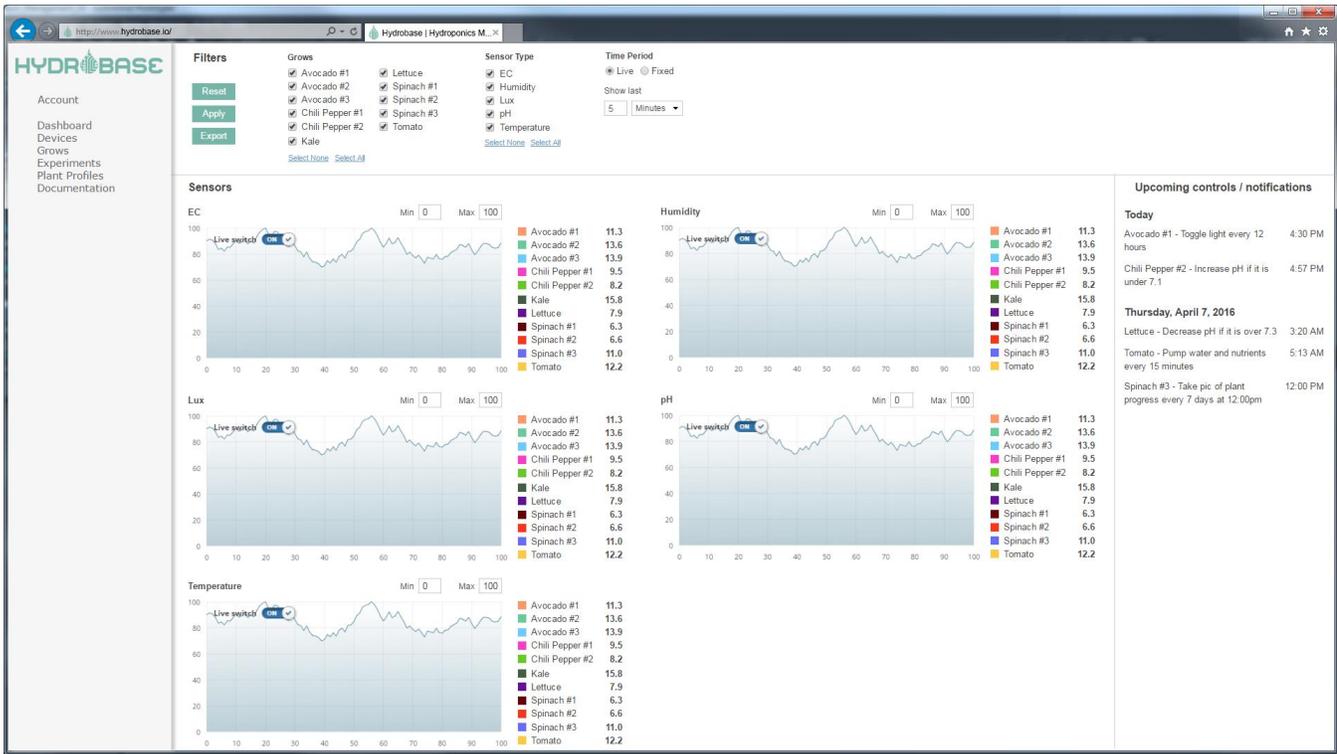
Low fidelity mockups



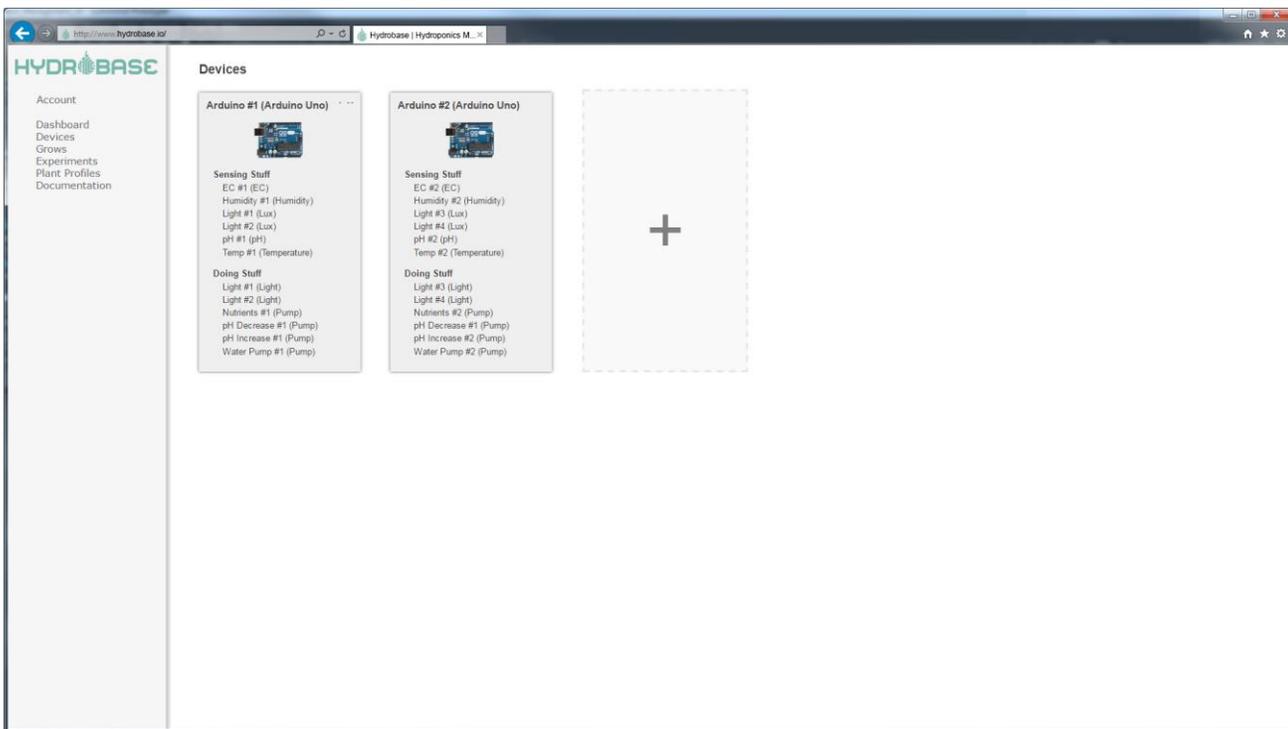
Interactive Prototype

An interactive version of this prototype can be found [here](#).

Dashboard with filters



Devices



Grows

The screenshot displays the Hydrobase web application interface. On the left, a sidebar lists navigation options: Account, Dashboard, Devices, Grows, Experiments, Plant Profiles, and Documentation. The main content area is titled 'Avocado #1 Profile' and includes a 'Delete Cancel Save' header. Below this, there are sections for 'Sensing (Arduino #1)' (listing EC #1, Humidity #1, Light #1, and Temp #1) and 'Doing (Arduino #1)' (listing Light #1, Nutrients (Pump), pH Decrease #1, pH Increase #1, and Water Pump #1). A 'Controls / Notifications' section lists tasks such as 'Toggle light every 12 hours' and 'Pump water and nutrients every 15 minutes'. A 'Photos + Videos' section shows a photo of a grow tent setup. On the right, three sensor graphs are displayed: 'Avocado #1 Sensors' showing EC #1 (65.3), Humidity #1 (22.9%), and Light #1 (70.2 Lumens). Each graph has a 'Live switch' control set to 'ON'.

Control setup with condition

Control / Notification Setup

Name:

What to do

Activate control Send notification

Activate for

When to do it

Once Repeating

Start

Repeat every

End

Never

After occurrences

On

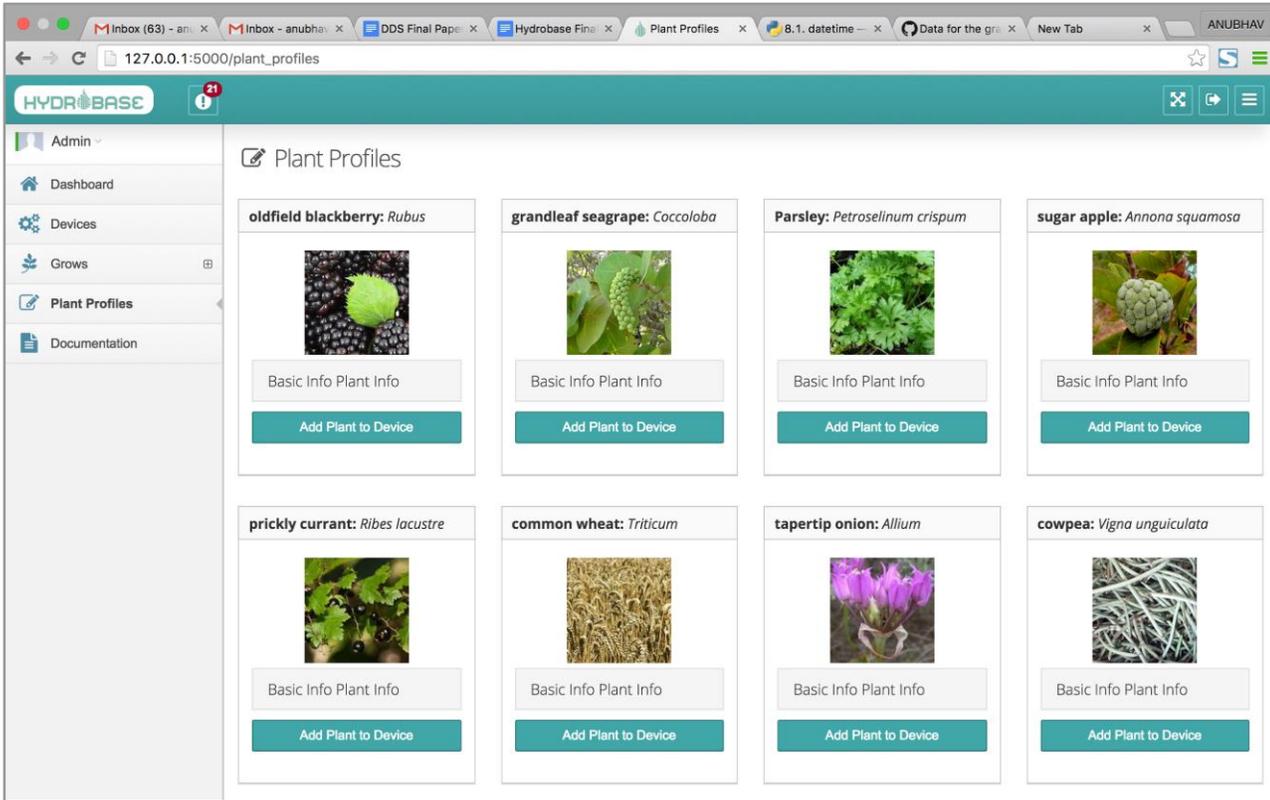
Under which conditions to do it (optional)

If of these conditions are

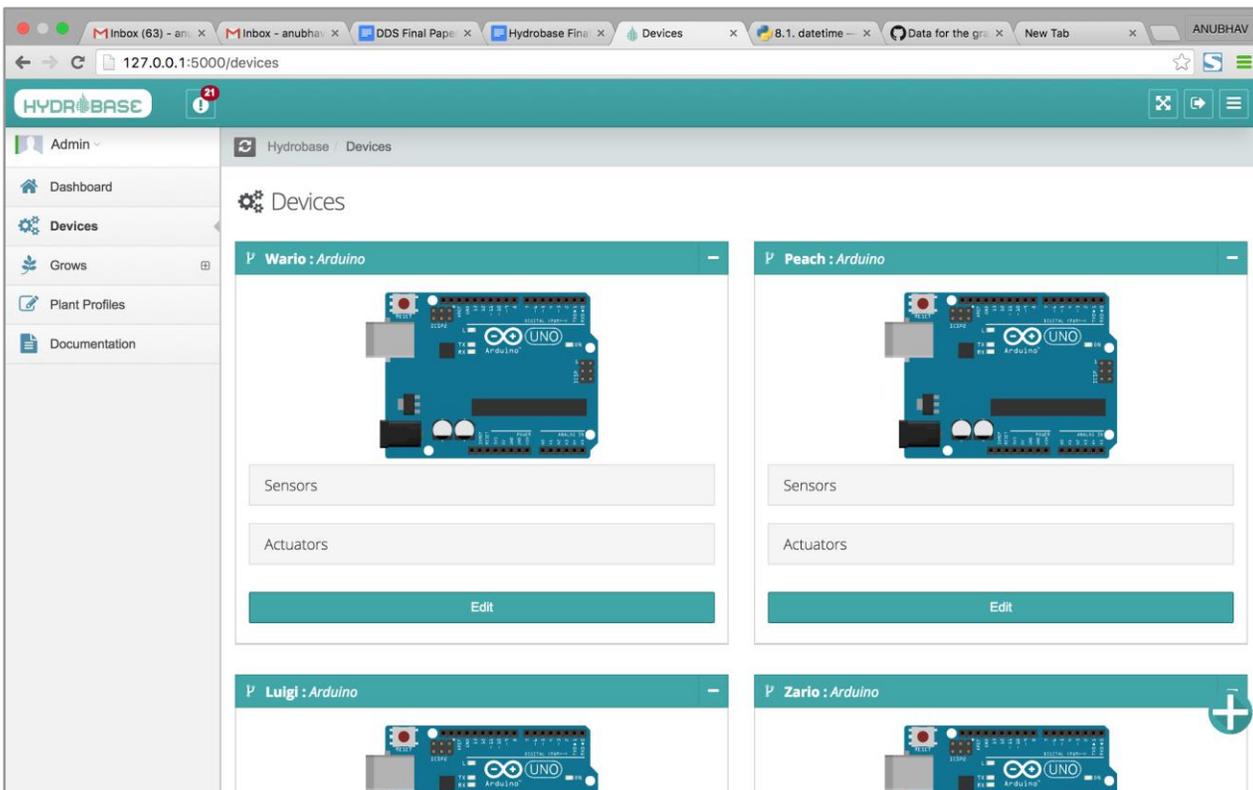
If is

Web Application

Plant Profiles



Device management



Edit a device and emergency stop

The screenshot shows the 'Emergency Stop' configuration interface for a device named 'Wario'. The interface includes a sidebar with navigation options like 'Dashboard', 'Devices', 'Grows', 'Plant Profiles', and 'Documentation'. The main content area is titled 'Emergency Stop' and contains the following sections:

- Device Name:** A text input field containing 'Wario'.
- Wario Set Up:** A section header.
- Sensors:** A list of sensors with checkboxes, all of which are checked: Lux, Water Temp, Air Temp, Humidity, pH, EC, TDS, and PS.
- Actuators:** A grid of actuators with input fields for their IDs:
 - Light 2: 31
 - Light 1: 30
 - PhUpper pump: 35
 - PhDowner pump: 34
 - Water pump: 32
 - Nutrient pump: 33
- Publish Key:** A text input field containing a long alphanumeric string: "pub-c-5bef0e0e-e177-4a94-b06f-ce3c1355e5cd".
- Subscribe Key:** A text input field, currently empty.

Edit a grow

The screenshot shows the 'Edit a grow' interface for a grow named 'Lily: capsicum_annuum_jalapeno'. The interface includes a sidebar with navigation options like 'Dashboard', 'Devices', 'Grows', 'Plant Profiles', and 'Documentation'. The main content area is titled 'Lily: capsicum_annuum_jalapeno' and features a photo of soil in a cup. Below the photo, there are two sections for configuring actions:

- Time Based Actions:** A list of actions with toggle switches and frequency settings:
 - Light 1: toggle Every 12 hours
 - Light 2: toggle Every 12 hours
 - Water Pump: toggle Every 15 minutes
- Condition Based Actions:** A list of actions with condition settings:
 - phUpper pump on if: pH < 6.0
 - phUpper pump off if: pH > 6.0
 - phDowner pump on if: pH > 6.8
 - phDowner pump off if: pH < 6.8
 - nutrient pump on if: EC < 600
 - nutrient pump off if: EC > 700

On the right side of the page, there is a line graph showing 'pH Value' over time. The y-axis ranges from 2 to 10, and the x-axis shows times from 22:00 to 23:30. The graph shows a fluctuating pH value with a peak near 11 and a trough near 3. A 'Show all' button is visible in the top right corner of the graph area.

Add automated control to grow

Set Grow Controls

What To Do On A Schedule (Controls):

Time Based Activity 1:

Actuator: Water Pump

Every: 10 Minutes

05/11/2016 05/24/2016

What To Do If...(Conditions):

If This Is To This Then Do

pH Greater Than 7.5 pH Downer C

If This Is To This Then Do

pH Less Than 6.5 pH Upper Or

Add Control Add Condition

Permutation testing

Create Grow From Plant Profile

Grow Name: Lily

Do You Want To Help Us With An Experiment?

Yes No

Grow Set Up

pH Range

.25 Below Default () .25 Above Default ()

EC Range

100 Below Default () 100 Above Default ()

Lights

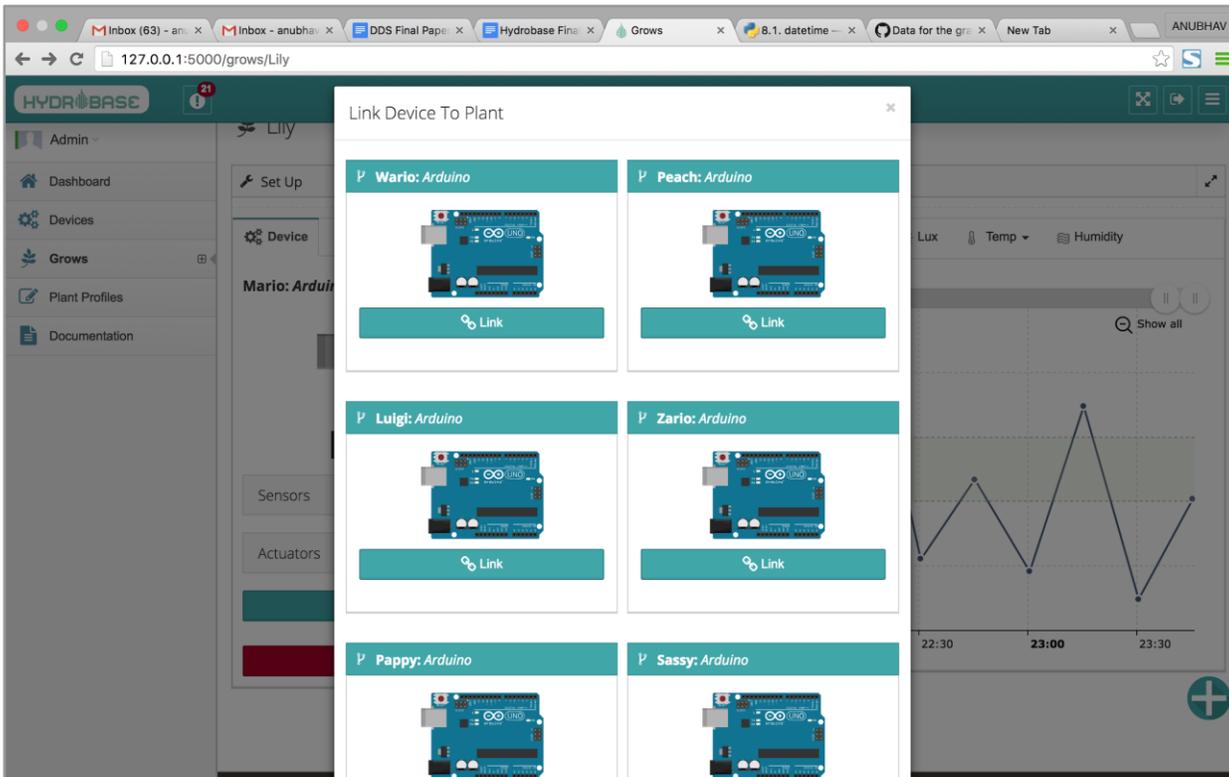
24H On 18H On 6H Off 12H On 12H Off 6H On 18H Off

Water Cycle

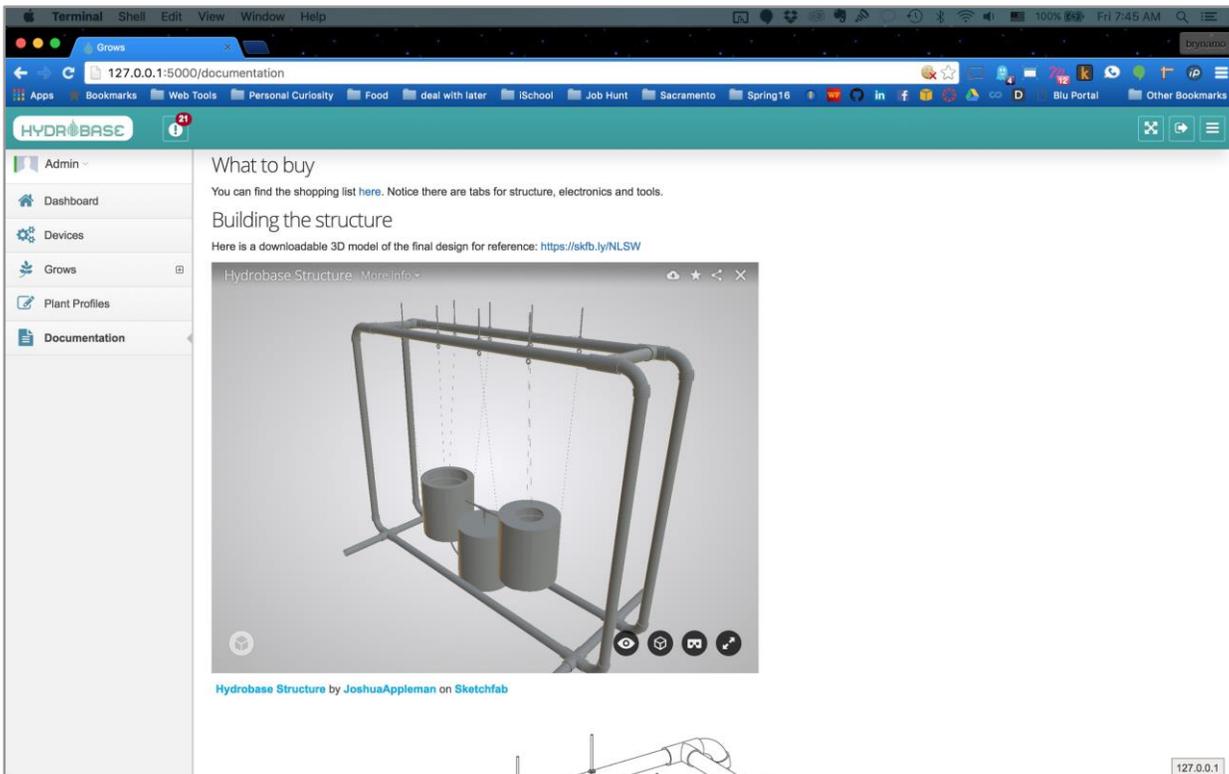
5Min On 5Min Off 15Min On 15Min Off 30Min on 30Min off

Submit Cancel

Link device to grow



Documentation 3D Model



Documentation drill holes

Terminal Shell Edit View Window Help

Grows

127.0.0.1:5000/documentation

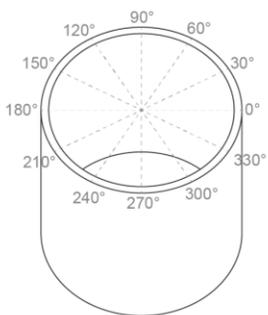
HYDRABASE

Admin

- Dashboard
- Devices
- Grows
- Plant Profiles
- Documentation

Note

Of the 3 buckets, 1 will be your nutrient solution reservoir, 2 will be for plants. When specifying drill hole locations on the bucket sides, we'll use an angle coordinate (see diagram below) and a distance from top or bottom.



Drilling holes in reservoir bucket and lid

- In the center of the plain bucket lid, drill a 1/2" hole
- Drill 1/2" holes in bucket side at 0° and 180°. Make it so the bottom of the hole is just about touching the bottom of the inside of the bucket.

Drilling holes in plant buckets

- Drill 1" hole centered in the bottom of the bucket
- Drill 1/2" hole in bucket side at 0°, 3.5" from top lip of bucket.
- Drill 1/8" holes in bucket side at 45°, 135°, 225° and 315°, .375" from top lip of bucket
- Drill pairs of 1/8" holes spiraling around bucket side for zip ties, one on top of the other with 1/4" between.
 - At 45°, with pair center 3.5" from top lip of bucket
 - At 90°, with pair center 3.25" from top lip of bucket
 - At 135°, with pair center 3" from top lip of bucket

127.0.0.1

Documentation attach funnel and tubing

Terminal Shell Edit View Window Help

Grows

127.0.0.1:5000/documentation

HYDRABASE

Admin

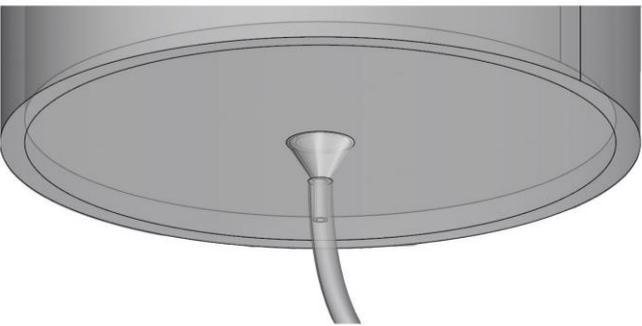
- Dashboard
- Devices
- Grows
- Plant Profiles
- Documentation

- 1 - tubing T connector
- 1 - Bucket basket lid (10" wide)
- 1 - Bucket basket lid (6" wide)

Instructions

Use glue gun to attach top of funnels to bottom of plant buckets.

Once dried, use glue gun to attach 1' pieces of tubing to bottom of funnels.



Put the reservoir bucket on the ground between the two hanging plant buckets. Rotate the reservoir so the holes on the sides are under the adjacent plant bucket. On both sides, take the tubing attached to the funnel and push about 1/2" of it into the hole in the side of the reservoir. Use glue gun, then waterproof caulk to seal hole.

Connect the slack tubing from the top of the reservoir and the sides of the plant buckets using the T connector.

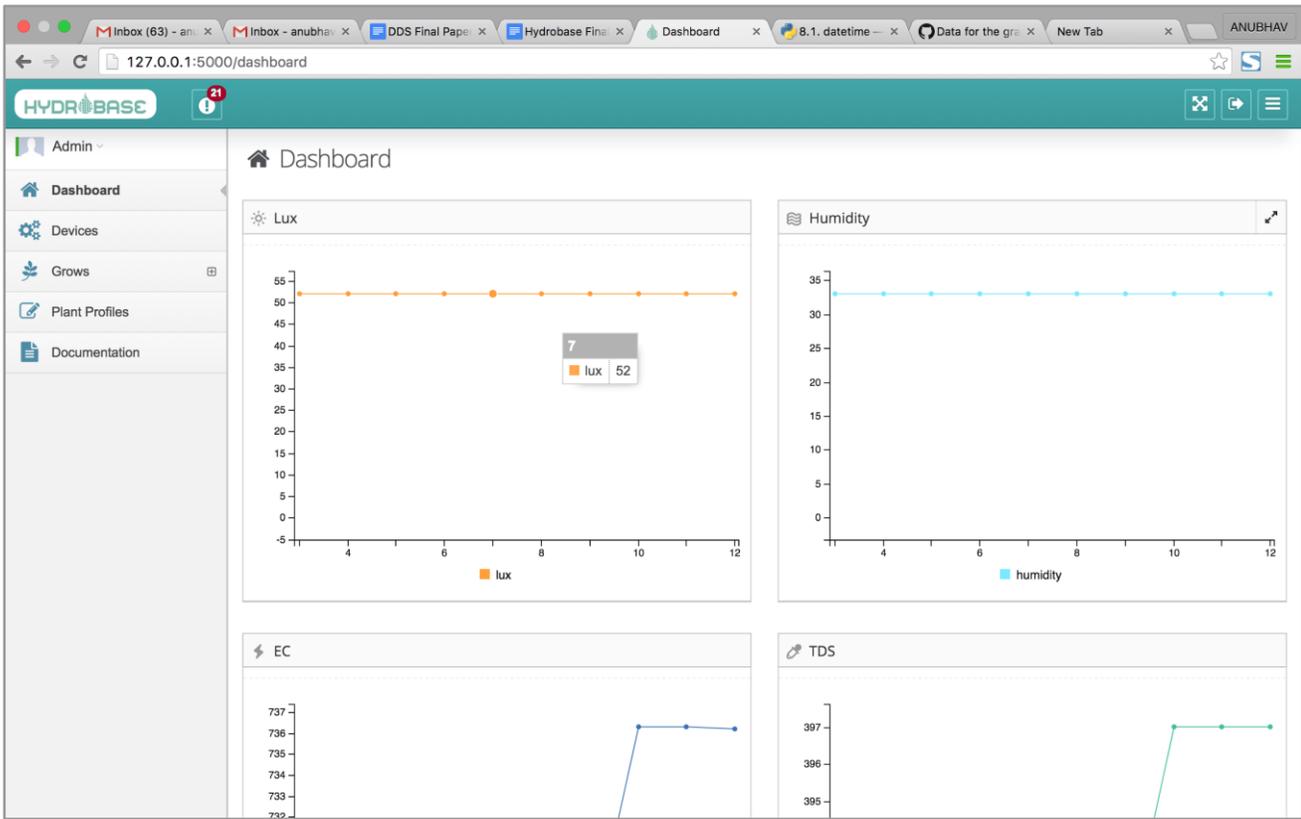
Place basket lids (6" and 10") on plant buckets. Close up all unsealed holes with glue gun.

Final assembly:



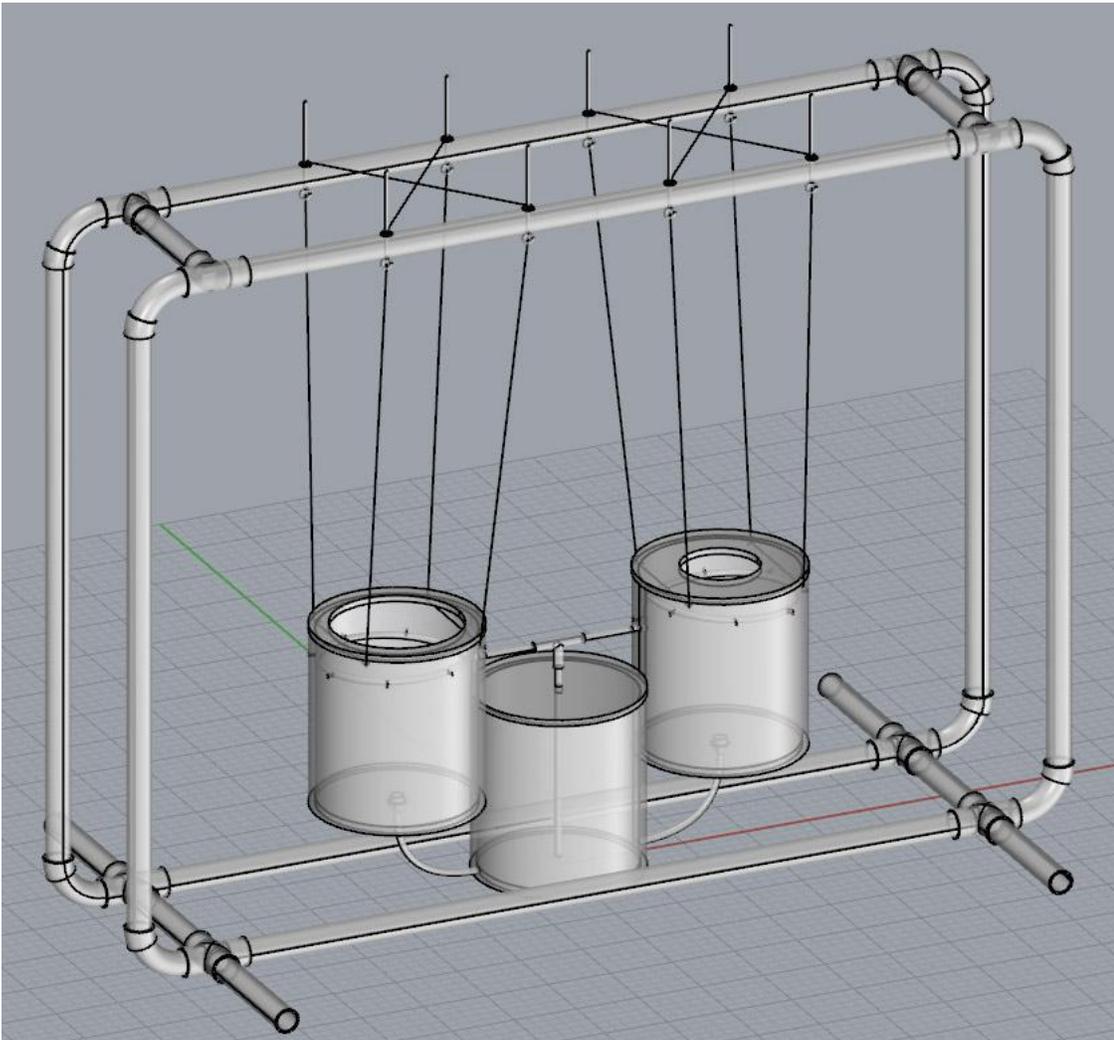
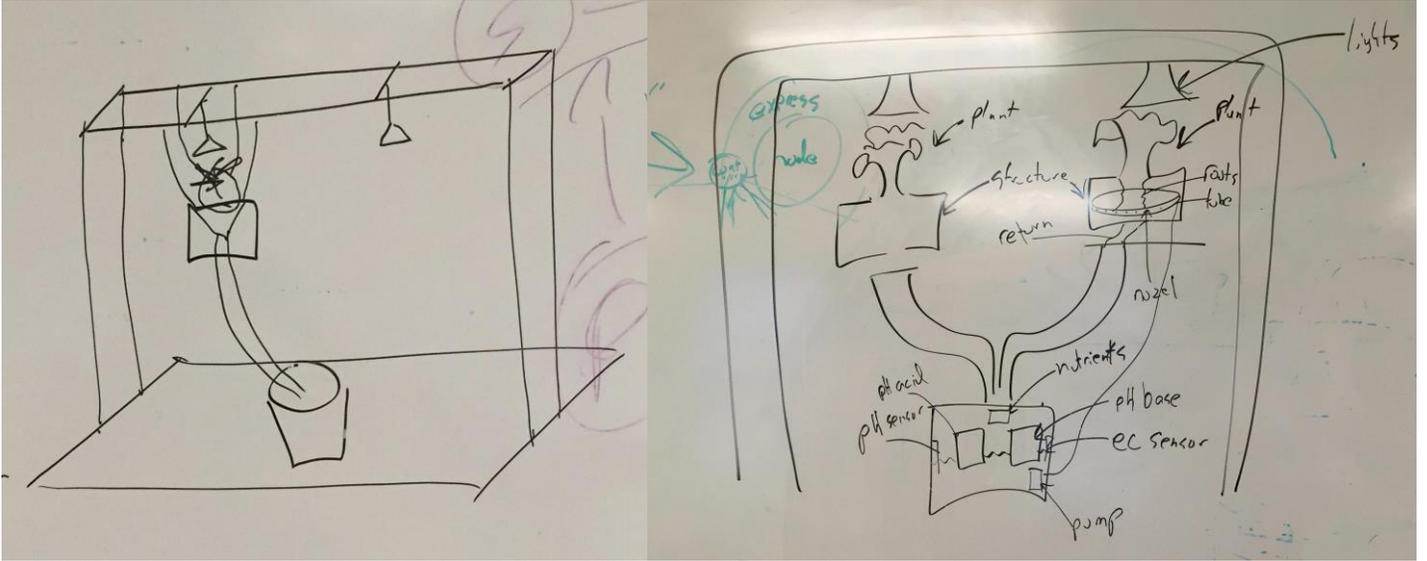
127.0.0.1

Sensor visualization dashboard

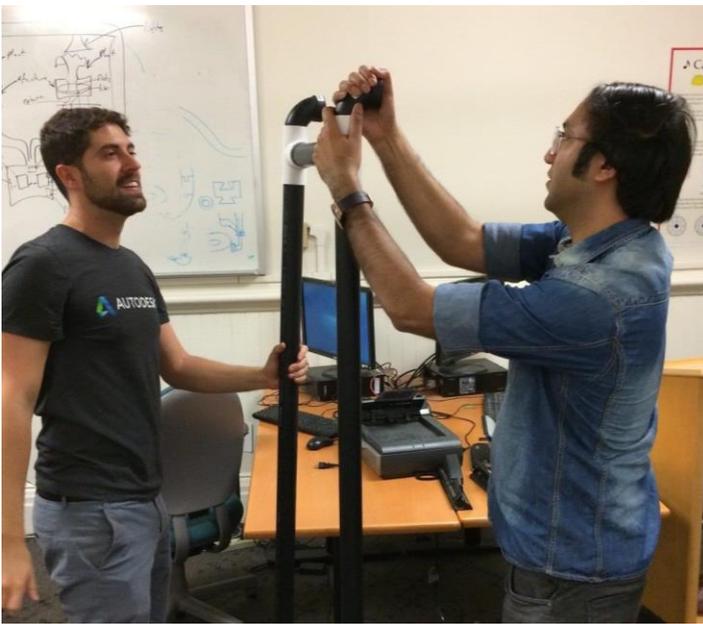


Physical setup

Drawings and 3D model of structure. Downloadable model found [here](#).



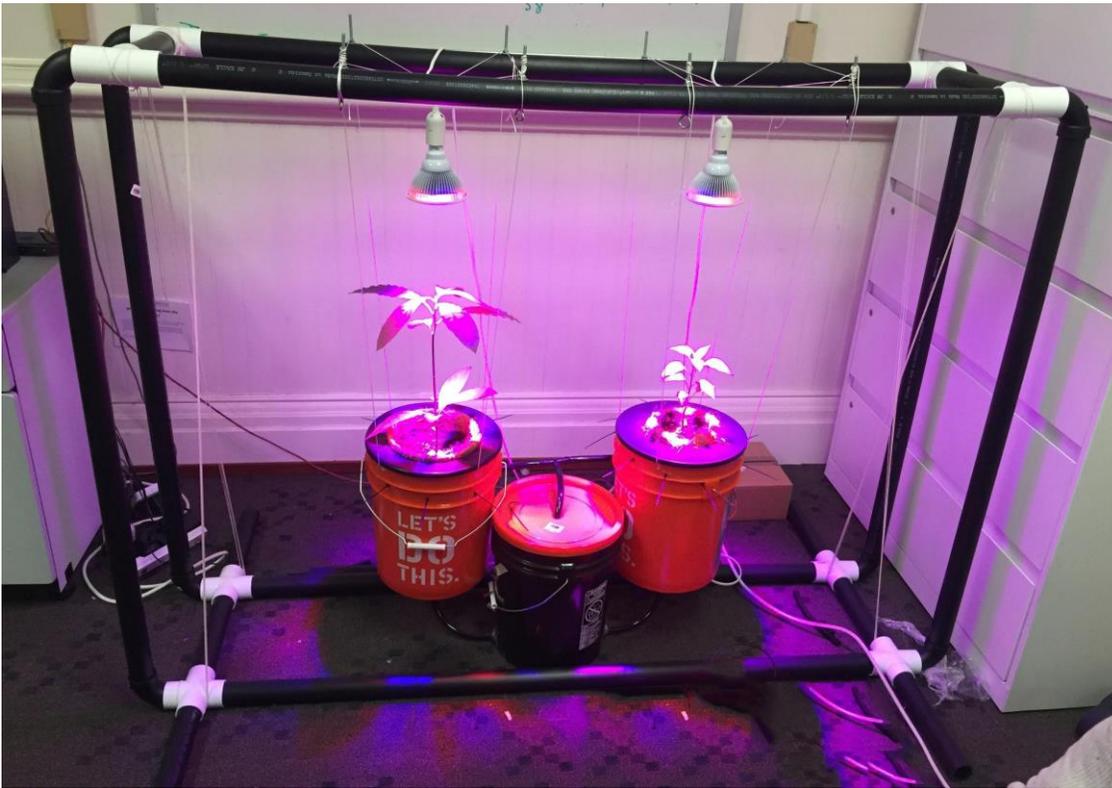
Building framing and connecting tubing



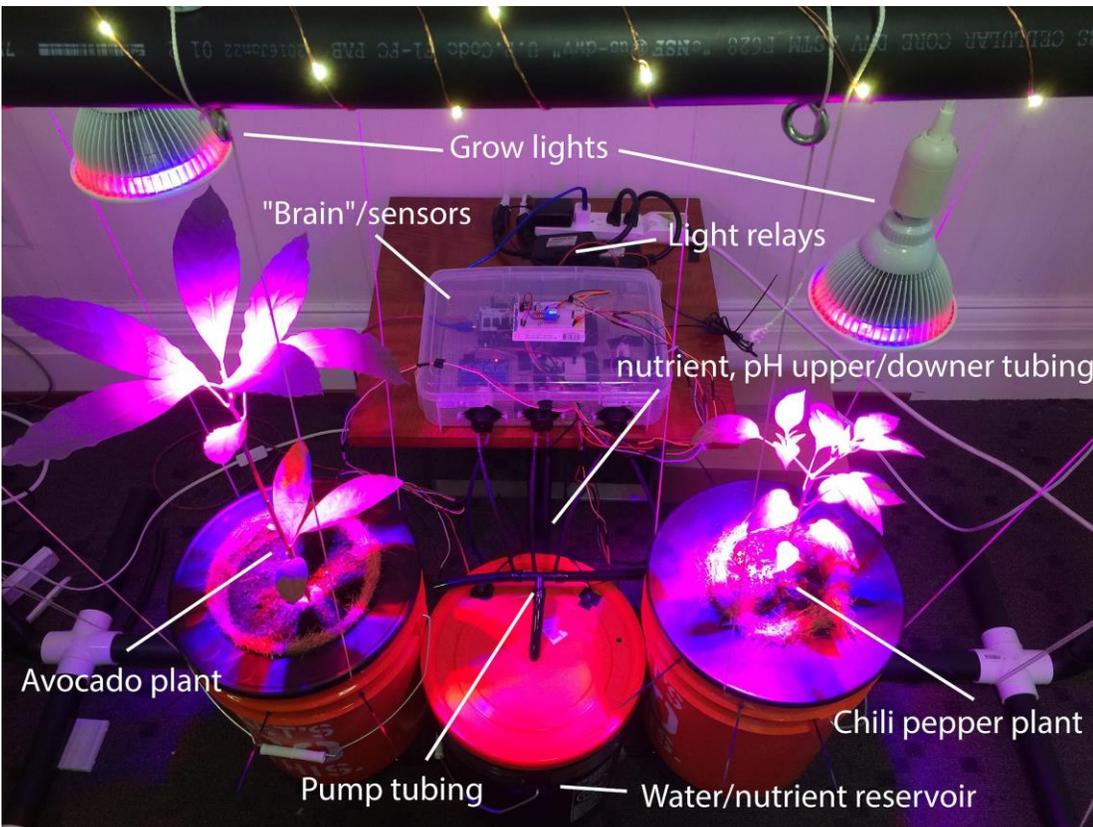
Inside of bucket with aeroponic spray nozzles and beautiful white plant roots



Finished structure



Annotated structure and electronic setup

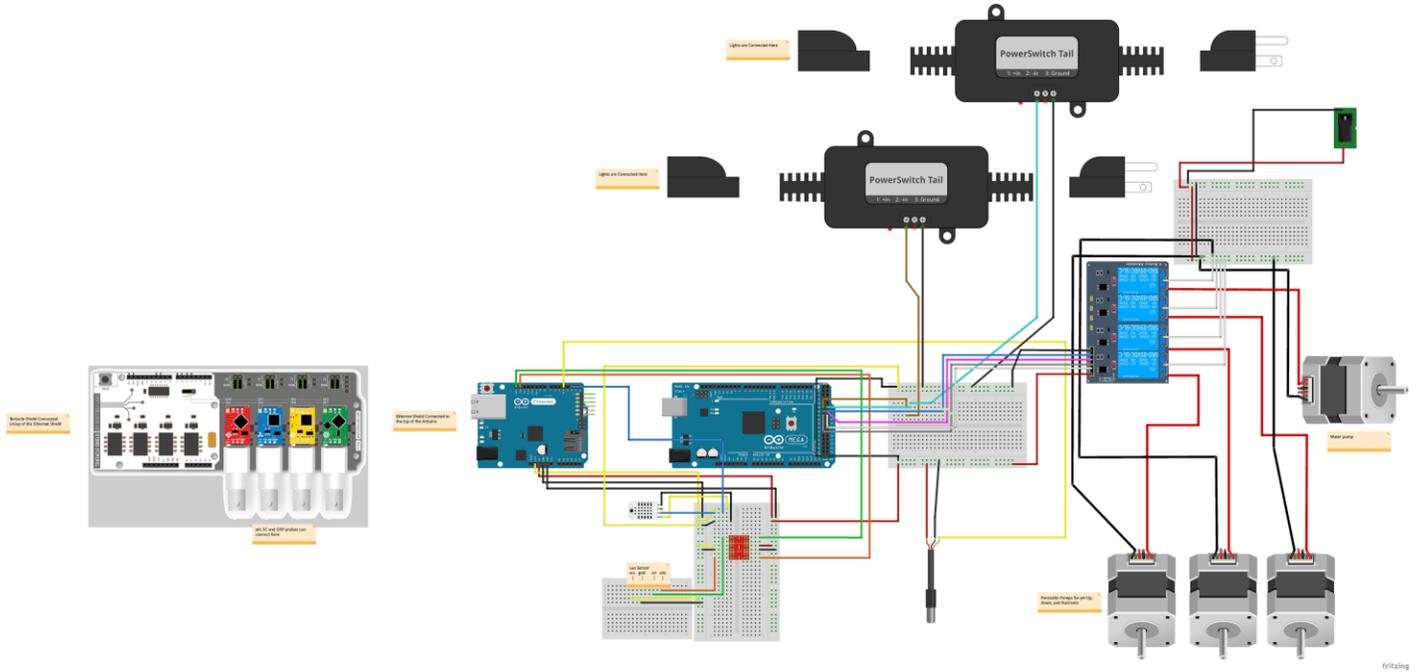


Grow Progress



Electronics hook up diagram

Full resolution version here: <https://goo.gl/orxVtB>



Network Diagram

