The DIY Resilient Smart Garden Kit

Birgit Penzenstadler, Jason Plojo, Marinela Sanchez, Ruben Marin, Lam Tran, Jayden Khakurel
CSU Long Beach
Long Beach, USA
[birgit.penzenstadler|jayden.khakurel]@csulb.edu
[jason.plojo|marinela.sanchez|ruben.marin|lam.tran]@student.csulb.edu

ABSTRACT

Our climate is changing rapidly, and the effects of that have shown themselves through the droughts in California. In an effort to become more resilient and contribute to saving water and other resources, people become more interested in growing their own food, but do not have sufficient gardening experience and education on conserving water. Previous work has attempted to develop resilient smart gardens that support the user in automated watering using simple embedded boards. However, none of these solutions proved to be scalable nor are they easy to replicate for people at home. We developed a smart resilient garden kit with IoT devices that is easy to rebuild and scales (by series connections). We use an Arduino board and a number of sensors connected to a planter. Data gets stored in a local database and is accessible via a web app and a web page. The resilient smart garden provides a learning environment that helps bridging between computer science and this sustainability-centric application domain.

In this paper, we report on a the blueprint for an educational blueprint for multidisciplinary smart garden projects, our experiences with self-guided implementation and reflection meetings, and our lessons learned. By learning about water conservation using automation on a small scale, students develop a sense for engineering solutions regarding resource limitations early on. By extending such small projects, they can prepare for developing large-scale solutions for those challenges.

CCS CONCEPTS

• Social and professional topics → Sustainability; **• Software and its engineering → Requirements analysis**;

KEYWORDS

Sustainability, software engineering, pedagogy

ACM Reference Format:
Birgit Penzenstadler, Jason Plojo, Marinela Sanchez, Ruben Marin, Lam Tran, Jayden Khakurel. 2018. The DIY Resilient Smart Garden Kit. In *Proceedings of Intl. Workshop on LIMITS (LIMITS'18)*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

LIMITS'18, May 2018, Toronto, Canada

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 MOTIVATION AND INTRODUCTION

According to William Rees, "Resilience science is based on the simple premise that change is inevitable and that attempts to resist change or control it in any strict sense are doomed to failure" [33, p. 5]. He adds that to achieve that, development strategies have to abandon efficiency and maximization as primary goals in favor of social equity and ecological stability [33]. The global effects of climate change can be observed in phenomena such as frequent flooding, droughts, and heat waves. Such effects directly and indirectly impact the: i) environment (i.e., changes in soil fertility and the growth patterns of plants and local landscapes [8, 10, 34]; ii) economy (i.e., the need for financial aid due to drought [22]); and iii) society (i.e., increased food and water prices due to greater demand [31]. For example, parallels can currently be seen in California, where the occurrence of droughts and intense heat waves has increased [20]. To become more resilient to future climate conditions [1], create a greener society, save money, and improve food quality [35], many individuals are practicing home horticulture. Home horticulture is the nonprofessional cultivation of plants for recreation, personal health, cost savings, and environmental and social benefits [10, 27, 29]. Though mitigating climate change is a motivation to learn home horticulture, individuals still need to reduce their outdoor water consumption. Research indicates that individuals consume more water during outdoor activities such as gardening than during indoor activities due to a lack of knowledge about water usage [15]. Domene et al. [11] argue that garden watering could account for up to a third of household water consumption yearly and close to 50% of total consumption in the summer [11]. Caetano et al. [6, pp. 566] state, "too little water will retard plant growth and reduce quality, while too much will leach fertilizers and reduce aeration". Adequate watering dictates the quality of the harvest, which is why we try to facilitate it by an automation that protects the user from overwatering (wasting resources) and protects the plants from drought.

To reduce an individual's outdoor water usage, researchers and practitioners have developed automated watering system [6, 12, 16]. However, there have been no scalable, affordable, or easily replicated solutions for people at home who lack technological skills. Attari et al. [4] report that participants in their study were ready to simply reduce their usage rather than explore efficient solutions because of the additional expense involved in controlling water consumption through an acquired, efficient system. Research also indicates that DIY solutions enhance consumer's desire by increasing their pleasure and sense of individualism while saving them money [18]. Therefore, the challenge is to find an affordable, scalable solution that can reduce the cognitive effort and cost while closing the gaps for gardeners with limited technological and home horticulture experience.

The **objective** of our research was to develop the prototype of an affordable resilient smart garden education kit that teaches students how to embed technology in a gardening environment to monitor and support the natural growth of vegetables.

The **contribution** is to provide a prototype of an educational blueprint for a resilience-oriented, interdisciplinary project course for smart gardens. This gives students the opportunity to explore working with limits in terms of natural, technical, and economic resources.

In this study, we developed an affordable, do-it-yourself (DIY) resilient smart garden to reduce the experiential learning curve of people who have an affinity for technology but lack gardening knowledge. This was conducted in a hands-on summer research project with four undergraduate research assistants. Specifically, we used an Arduino board and sensors connected to a pump to control water consumption through data visualization. Data stored in a local database was easily accessible via an online portal and a responsive web application. This application was implemented to improve the interoperability of apps and to facilitate the gardeners' activities. The project connects multiple disciplines, namely computer science, computer engineering, environmental studies, biology, and permaculture [28]. The impacts of the resilient smart garden system are that it:

- Reduces the cognitive effort and closes the technological gaps for gardeners with limited prior technological experience;
- Reduces the experiential learning curve of people with affinity to technology who have little prior gardening knowledge;
- Helps to bridge between disciplines — computer science and computer engineering and environmental studies and biology.

The rest of this paper is organized as follows: In Section 2, we differentiate the related work, in Section 3 we present the research design and in Section 4 the implementation. In Section 5 we analyze the results, and in Section 6 we offer a discussion. We conclude with a summary and future work in Section 7.

2 BACKGROUND

A closer look at the related work shows that the majority has looked at water conservation projects, resilient smart garden projects and systems used for farming, gardening help systems, Arduino DIY projects and educational kits for gardening.

2.1 Small-scale water conservation projects.

A few small-scale water conservation projects discuss and utilise Arduino boards as a hardware tool with additional sensors. Amberg [2] and Maleficarum [23] offer instructions on water-saving systems using Arduino tools that are intended to be built at home by a hobbyist. Maleficarum [23] presents a tutorial for a water conservation system project that uses water from a washing machine to recycle to a toilet tank, which can be completed within 10-20 hours using an Arduino Uno board, a water sensor and a water pump. Amberg [2] describes the setup, programming and installation of a water-saving device on a tap using Arduino and a flow sensor to measure how much water is used. By lighting up red LEDs after one liter of flow, the system helps reduce unnecessary water waste. The project requires 5-10 hours of effort. Lee and Gallardo [37] created the project Vinduino, which aimed to reduce agricultural water consumption in a vineyard. This project uses an Arduino Uno, low-cost moisture sensors, and a solar module — and went on to win a hackathon competition. Palmer [30] developed a simple clock-based irrigation controller that aimed to allow residents with limited IT background to set up a cheap and simple way to monitor and reduce the water they used for irrigation. The project consisted of a moisture sensor, a real-time clock module, an Arduino Uno R3 and an Arduino Shield.

All these projects use Arduino boards and maker ideas to install water-saving interventions. The proposed Resilient Smart Garden project draws upon these maker principles and applies them to save irrigation water for a small vegetable garden.

2.2 Other RSGs and systems used in farming.

The Resilient Smart Garden is not the only project designed to bring gardening and technology together to make gardening easier.

Connected Garden [13] was implemented by The University of Central Florida. They set up an outdoor garden and used sensors controlled by an Arduino board to collect data. The project relied on a variety of sensors to send information to servers. The main focus of this project was to collect data on both the natural environment and interactions with the garden.

The Guarduino project [39] carried out by students at Poornima Institute of Engineering Technology in India is most similar in design to the Resilient Smart Garden. The Guarduino uses a variety of analog and digital sensors including light, temperature, and homemade moisture sensors that are all connected to an Arduino.

The Automated Aquaponics Design Report [7] describes a sustainability project that monitors both the condition of a garden as well as a fish tank. As explained in the article aquaponics works off the principal that the garden and fish will create an ecosystem where the waste from the fish will provide nutrients to the plants. It is a system that replicates the relationship between fish, bacteria, and plants — a relationship that exists in the natural world [7]. While the system monitors two different elements, the project still has elements that are similar to the Resilient Smart Garden.

OpenAg [26], also known as Open Agriculture, is an open source project developed by MIT that was created to make gardening easier and provide a more controlled environment to care for a garden bed. Out of all the systems found OpenAg is the most developed and requires a lot more hardware. Their product is defined as a personal food computer. The garden bed is kept in a closed environment that is controlled by a raspberry pi instead of an Arduino.

2.3 Gardening help / data collection.

There are a few commercial-off-the-shelf "COTS" systems that are available in the market to help gardeners grow plants. The search led to identify Edyn Smart Garden System [14] and GreenIQ Smart Garden Hub [16].

The Edyn Garden helps gardeners to monitor the environment condition using Edyn Garden Sensor. This sensor can track humidity, temperature, light, moisture, soil nutrition, then compare those data to provide plant database and soil science, and generate the garden guidance to help gardener. Edyn garden system consist of setting up mobile apps on both operating system (i.e. Android and IOS), which is able to send garden data in the real-time to the gardeners such as the condition of the garden and the guidance on how to adjust the garden better through WiFi network.

The GreenIQ Smart Garden Hub is a system that helps gardeners grow plants from anywhere and anytime by wisely managing a garden's irrigation system and lighting scheduling using Internet Cloud and mobile technologies. All GreenIQ models are watersense certified by ICC-ES (can save up to 50% on a water bill). The system allows users to control and schedule the garden's irrigation and lighting from anywhere and anytime by using desktop or mobile apps. It connects to WiFi networks, collects the information from the nearest weather station and then calculates the water needed for the garden.

Both tools facilitate the gardening and irrigation but are not targeted towards educational use.

2.4 Other DIY Arduino projects.

Electronic DIY projects are more accessible with easily programmable single board microcontrollers. Arduino is an open source hardware platform that can be programmed using Arduino IDE which supports official Arduino hardware like the Arduino UNO and third party hardware such as the nodeMCU. Daniels [9] offers instructions to make an outdoor automatic garden watering device using an Arduino UNO that measures the soil moisture levels. The project is placed inside an enclosure that has an LCD screen that displays the current moisture levels and is powered by a 12V battery. Aqib [3] presents an advanced automatic watering garden tutorial that will store moisture, temperature, humidity, heat index, pressure, and value status into a database. The Arduino UNO is powered by a 12V battery and communicates with a server locally using an Arduino Ethernet Shield. Hamza [17] provides information on making a temperature data logger using a hardware clock. The data is stored locally on an SD and does not communicate with a server. Iseman [21] demonstrates an automatic watering garden using DIY moisture sensors. Two nails are attached to a wire and connected to the Arduino to detect the soil moisture level by putting a low current through the soil via one nail and detecting the resistance via the other. The more water in the soil, the less resistance there is (and vice versa). The temperature, humidity, and moisture data is sent through a serial port, but not stored into a database. The Arduino UNO must be connected to a computer to display the data through the serial port and to power it.

All of the DIY Arduino projects have similar approaches to implementing an automatically watering garden. The Resilient Smart Garden shares some characteristics to minimize water usage while maintaining a sustainable environment for the plants, but goes beyond the pure DIY implementation.

2.5 Other educational garden kits.

The Bee Smart kit [32] is an educational miniature garden for use in a school or home environment. The educational kit is geared toward children from grades 3-6 to learn about gardening systems in a easy to setup kit. The goal is connecting them to plants, pollinators, food, and gardens by (potentially) creating habitat for pollinators. The kit itself is very limited with regards to gardening as it is marketed as a simple educational kit with minimal effort by the user.

The Tower Gardens school kit [36] is the educational version of an aeroponic system that uses water, liquid nutrients and a soilless growing medium to quickly and efficiently grow produce. The kit is pre-built and requires minimal effort to set up and begin growing. The kit is built around a portable grow light that lights the vegetation from the inside and requires no setup from the user with regards to hardware. The DIY part involves planting the seeds with ease into the base of the kit and allow it to grow. The user is only responsible for maintaining the pH and water levels as it does not require sunlight, since it was built with the indoor garden in mind. The downside to this educational kit is the significant price tag, for a basic kit it costs $45.25 per month, a steep price for an educational system.

Heitlinger et al. [19] argue that research into ubiquitous computing for sustainability must move its focus beyond designing for individual consumer behaviors. Their work on participation, community, citizenship and collective action in London's urban grassroots food-growing communities proposed the Talking Plants Sale prototype, to support the values of the farm.

3 RESEARCH DESIGN

3.1 Context: Undergraduate Summer Research.

The Resilient Smart Garden is an autonomous controlled garden that efficiently and effectively grows plants. With the Resilient Smart Garden, a hobby gardener will be able to use less water while growing their vegetables with greater ease. While the idea of a small "smart" personal garden may seem to have limited scope, if this type of technology is adopted in large enough numbers, it can be begin to have a huge impact. The scope can quickly go beyond an interested hobbyist, to a local farmer's market, to small commercial sellers and may even influence large-scale producers. If the method is simple enough to integrate in a variety of systems, economically viable and is efficient in its use of resources, then the product will no longer be viewed as a small enthusiasts weekend project but as an actual alternative to traditional farming methods. If we can at least begin to start the conversation of smart technology and sustainable farming in an educational setting, then this project will be a major success.

The initial prototype of the system for a single plant is extended for a small garden with several plants. This system serves as an educational tool for connecting software engineering to systems thinking and sustainability. The system under consideration is a project developed using an Arduino board and permaculture principles. These principles allow for sustainable long-term garden cultivation with maximum harvest based on the natural capacity of the soil enhanced by well-planned companion planting [28]. The system vision is to connect a growing bed via sensors to an Arduino board such that we can measure moisture, humidity, and temperature and log that data. This enables determining the minimally feasible amount of watering, which is an environmentally sustainable measure in drought-prone Southern California. This system enables people with little background in gardening to successfully grow vegetables in the most sustainable and resource-conserving way. Along with the development of the extended version of the system, the research assistants develop a documentation intended for non-technical hobbyists to be able to follow and set up their own Resilient Smart Garden, see Fig. 1.

3.2 Research design & methodology.

The Resilient Smart Garden in its current prototypical form can manage one plant by using one humidity sensor and one temperature sensor. The envisioned extended version manages 10 plants of different kinds using multiple temperature and humidity sensors. According to companion planting, we select a design that groups together plants that benefit from growing together, for example a tomato plant and a basil plant. The root systems of those two plants use different layers of soil, and the basil keeps potential pests (e.g. snails) at bay with its strong fragrance. The ten plants are planted in two planter boxes each monitored by one Arduino board. In addition to the embedded part, a web app provides the front-end user interface where the gardener interacts with the setup of the garden. This includes a companion-gardening design feature that helps the inexperienced gardener to put together the plants (as in the above tomato-basil example). Imagine a zoom-in in Google Maps with a layover of plants that can be dragged and dropped on a garden design canvas. The user receives a list of proposed plants that grow well in that specific area plus feedback on whether a plant grows well next to the plant it is being dropped on the map. Furthermore, users can review the graphics of the measured data over time and adjust the automatic watering. They can see their garden's history and compare data over time and they can opt in to make their data visible to other gardener's using the same web app so they can learn from or be inspired by other designs. This way the platform can grow into an educational tool that gets better over time as more users provide their data.

3.3 Preliminary/foundational work in Spring.

This work is based on a prototype developed for an individual plant in a senior design capstone project from the Spring semester 2017. A first prototype of the Resilient Smart Garden system had already been built as senior project by a student team in the Fall semester of 2016 and was expanded upon by a different team in Spring 2017. Four of the students who worked on the preliminary work continued as research assistants over the summer to fully implement the prototype.

3.4 Educational strategy.

The students were allowed to work largely self-directed based on their previous experience from the first semester. They reported back weekly and we held reflective meetings to enhance their own analysis skills and learn from how the project unfolded [25].

3.5 Data Collection.

The data for this report is comprised of the requirements specification, project documentation, the garden data (sensor measurements as well as manual measuring and observation), and the project diary written by the research assistants.

4 IMPLEMENTATION: THE KIT

Day zero of the project started with the students and a set of small rubber ducks taking possession of the Research Lab. The students based the implementation on the requirements specification they had elaborated in the Spring.

4.1 Gardenware

The watering system was one the important parts in Resilient Smart Garden. This system involved a water pump that pumped water from a bucket through vinyl tubing; these tubes had small holes all along them in order for the water to pushed towards the plants, the design tried to replicate a drip irrigation system, but with high water pressure and small holes plants were more or less sprayed when watered. The water pump was controlled by the Arduino board, and it was only activated when the soil's moisture was at low levels. Two planters, two water pumps, four temperature and humidity sensors, and two moisture sensors were used in this system. They were placed at convenient places to collect the most exact data. For the vegetables, we chose five companion plants: Lettuce, basil, onion, carrot, and tomato. The resources needed for watering system and garden included: Vinyl tube, Zip ties, Plug (Expo Marker Caps), Planters (24" in length, with a width of 7.88" and a height of 7.25"), Soil, 5 Gallon Plastic Bucket, RGB Led Trip, and a Glue Gun. Garden and Watering System Setup:

(1) For each planter, make two rectangular holes (0.5cm h x 1.5cm w) on the 2 opposite hole to attach the humidity and temperature sensors.
(2) Drill a smaller hole along the other 2 walls of planter.
(3) Put 5 feet vinyl tube and arrange it along the interior side of the planter. Use zip ties to tie it to the planter and then use the plugs to seal one end of each vinyl tube. Then us the pins to poke small holes along the vinyl tube to let the water spray towards the plants through these holes.
(4) Attach the water pump to the other end of each vinyl tube and place the water pumps connected to the vinyl tube into the bucket.
(5) Place the plants in the planters, fill the planter with soil and place the moisture sensor directly into the soil in the middle of the planter. Finally, put led strip on the wall of the planters.

4.2 Hardware

The hardware consists of a temperature/humidity sensor, moisture sensor, water pumps, and a project enclosure made from a cardboard box that contains a power strip with USB type A plugs, power terminal block, 12V 1A power supply, Arduino UNO, nodeMCU WiFi module, logic level converter, and power relay to activate. The Arduino gathers the readings from the temperature/humidity sensor and moisture sensor. The nodeMCU gets the sensor readings from the Arduino UNO to send to a server to store the data. The power relays is controlled by the Arduino UNO and activates the water pump when the moisture sensor readings is below a certain value.

The **project enclosure** is made from the Amazon shipping box the electronics came in. The cardboard box has a width of 7.25 inches, length of 10 inches, and height of 3.25 inches. Two of the cardboard box flaps at the bottom with the longest length will be cut to repurpose it as a mount for the electronics. There are 4 electronic mounts in total: Two for the pair of Arduino UNO and power relay,

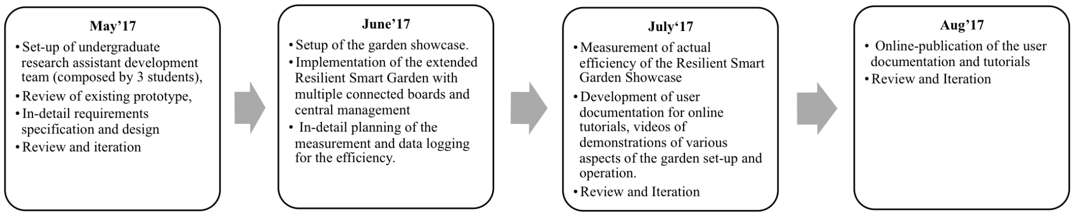

Figure 1: Timeline of the research project

one for the power terminal block, and the last for the nodeMCU. Holes were made on the box flaps to match the mounting holes in the electronics, which are fastened into place using a screw, standoff, and nut. The bottom of the project enclosure is duct taped along the edges both inside and outside to seal it from potential water. Rubber feet were also added to the bottom to further prevent water from entering the project enclosure by raising it from the surface.

The temperature/humidity **sensor**, moisture sensor, and power relay will be wired to the Arduino UNO. The temperature/humidity sensor is digital and needs 3 wires for VCC, GND, and DATA. VCC is connected to the Arduino UNO's 5V pin, GND to GND, and DATA to an available digital pin. The moisture sensor is analog and needs 3 wires for VCC, GND, and OUT. VCC is connected to the Arduino UNO's 5V pin, GND to GND, and OUT to an available analog pin. The power relay needs 3 wires for VCC, GND, and SIGNAL. VCC is connected to the Arduino UNO's 5V pin, GND to GND, and SIGNAL to an available digital pin.

The **nodeMCU** communicates with the server to send garden sensor readings that is retrieved from the Arduino UNO through I2C protocol. I2C communication requires 2 pins for serial data (SDA) and serial clock (SCL). There are dedicated pins on the microcontrollers for I2C communication: pin D2 for SDA and pin D1 for SCL on the nodeMCU and pin A4 for SDA and pin A5 for SCL on the Arduino. The nodeMCU operates at 3.3V and the Arduino operates at 5V; therefore, a logic level converter is needed for communication between the nodeMCU and Arduino. The logic level converter needs a power source from the nodeMCU and the Arduino to use as a reference.

The **power relay** activates the water pumps which is controlled by the Arduino UNO. The water pumps are off by default when the power relay is not activated. The water pumps are powered by a 12V 1A power supply that are controlled by the power relay. A circuit is made from the power supply, the water pump, and the power relay. The positive wire from the power supply connects to the common(C) pin in the power relay. The water pump positive power wire is connected to the normally open(NO) pin on the power relay. The negative power wire from the water pump is connected to the negative power wire from the power supply.

The **electronics** are powered from the power strip that includes 2 power plugs and 3 USB type A plugs. The pair of Arduino UNOs is powered through the USB type A plugs using a USB type B to USB type A cable. The nodeMCU is powered by the last USB type A plug using a USB Micro-B to USB type A cable. The pair of power relays share power from the 12V 1A power supply that is plugged into the power strip. The power is distributed using a terminal power block that has at least 3 rows: one from the power supply and two for the pair of power relays.

The **Arduino UNO and nodeMCU** uses the Arduino IDE development environment to implement code and upload sketches onto the microcontrollers. The ESP8266 package needs to be installed into the Arduino IDE using Boards Manager in order to developer, compile, and upload sketches onto the nodeMCU. The ESP8266 package can be accessed on the Arduino IDE by including a link to the package in the Preferences menu. All libraries used for WiFi communication on the nodeMCU is included with the ESP8266 package. The ArduinoJson library is used on the nodeMCU to format the sensor readings in JSON format to send it to the server. The ArduinoJson library can be obtained through the Arduino IDE Manage Libraries.

Figure 2: Garden with LED strip and irrigation

4.3 Software

The software side of the project mostly involved setting up a server written in nodeJS for handling http requests from the Arduino board. The use of node allowed for an easy set-up for collection of data from the Arduino board. The server is currently hosted on AWS, using a lightsail instance for deployment. The server.js file is ran by the AWS and the code listens for the requests. However, during the research period, it was impossible to use AWS due to security issues from the university's network. To get around this issue, a local network was set-up and the server was ran indefinitely on a

Figure 3: Garden with LED strip, ripe tomatoes, and research assistants

dedicated machine running a basic Linux distribution. The server used several external libraries to implement the features that are required by the project. The external libraries mainly used for the purposes of research are 'body-parser' and 'levelDB'. Bodyparser allows for the http request to read the contents of the body sent within the request, which then stores these contents into a local database set up by levelDB. These two libraries allowed for a simple implementation of the project. For the simple purposes of data collection, only one http request was used, a POST request using the '/data' endpoint that allowed the Arduino board to send data readings in a JSON format. The server would then read the JSON and parse the data into the .log file created by the local database. The process can be followed easily if broken down into steps: 1. Arduino makes an http request using a localhost:3001/data endpoint and sends the json data with a header and a body 2. The server checks the header to ensure that the credentials match the expected response 3. The server parses the data stored in a JSON file and prints it onto a .log file created by the database.

5 EVALUATION & ANALYSIS

This section provides a qualitative analysis of the implementation of the project as it occurred in summer 2017, an overview of our data, and the observations we made.[1]

Project diary. The students wrote a project diary from the start of the project until the end of the summer when the official data collection was done. The insights from the journal are mainly a documentation of the steps, how much work they could get done every time they met, and some observations about the complications and unforeseen circumstances. A few noteworthy entries are listed in Tab. 1.

Water savings. Surprisingly, we found that the soil remained really moist with very little watering, especially near the middle of the planter. The data would say that no or only little watering took place during the past week, but the soil was more than humid. We are not sure whether the ventilation in the lab is super low or whether the fact that it was inside is already leading to that outcome. The building our lab is in has AC and there are vents in

Table 1: Excerpt from project diary

Day Zero: Ducks everywhere	Got a set of rubber ducks for the lab for rubber duck debugging [5]. They have names and make people smile when they enter the lab.
May 22, Shopping	Worked on shopping lists for both garden and hardware components. Finding planters and soil was relatively easy. Finding the prices for seedlings online was more tricky for the cost estimation, as many of them weren't listed.
Memorial Day, 4 Hour Remote Work	Included some soldering and WiFi module research. The students were so motivated for the project to even put in work on a holiday.
June 5, Irrigation system	We did not have power tools, a power drill specifically, on hand. We had to manually make holes using scissors, screwdriver, and wire cutters. We made use of whatever we had available at any given time.
June 7, WiFi problems	Attempting to get the garden system to communicate with the server has a snag with the campus WiFi. The school is using WPA2 Enterprise encryption and PEAP protocol for user and password authentication. The current stable library for the WiFi module does not support this feature. Temporary solution: Tethering over phone. We now found a permanent solution using a library.
June 19 Pests detected	Only 1 of the 4 lettuces initially planted in the planters is alive, and currently struggling. We found a few pests: vine lice were found on the chives in planter 2, a cocooned caterpillar (Shield) was found on the leaves of the lettuce in planter 1, and a caterpillar (Little Jerkwad) ate a good portion of the leaves of the carrots in planter 1.
June 21, Pest control	Vinegar, get the damn pests away. That worked only partially, and one plant died supposedly because of too much vinegar sprayed.
June 22, First flood	Opening the door revealed a big flood on the floor and an empty water bucket. My first reaction is to turn off the power strip.
June 27, Partial solution	Online research suggests that having capacity constantly powered in a moist environment will enhance the corrosion on the capacitor. We minimized the time the sensors are powered.
July 5, Project enclosure	It turns out, the temperature/humidity sensor was mounted upside down, which flipped the polarity causing a short. Remounting the temperature/humidity sensor is unfortunately not an easy option for us, so we flipped the wiring on the Arduino and breadboard instead.
July 19, Press coverage	In the afternoon, a video team from IEEE Computer Society came in and interviewed us while taking B-Roll shots of the garden and the system. The project got more attention than we expected.
July 31, Harvest	Tomatoes in planter 2 are beginning to split open. We picked them. At this point the plant also had too little room and started to deteriorate.

our lab, but due to having a north facing room, it may not require much help by the AC to keep the temperature in the building-wide range. As we didn't have a hose to connect the irrigation to, we implemented a bucket solution. Filling up the bucket was a slightly arduous task due to its weight when full. Due to this surprising observation, we have no insights on potential water savings with our system. This has to be revisited with the outdoor prototype in Summer 2018.

Garden observations. In our observation spreadsheet, we documented Height, Width (spread), Color, Critter status (we added that column once the first ones appeared as we did not foresee that), Appearance, Fruit (if), and Scent. The tomato plant grew over four feet high (it actually grew to 5 feet but the top snapped during measuring on July 12 and regrew little) and brought quite a bit of fruit by July 26 (see Fig. 3). Unfortunately, then the roots couldn't go further, so the plant started deteriorating and eventually died. Lettuce never quite caught on. It wilted for a while, it turned yellowish, which may be a sign of over-watering, then it started regrowing for a bit, apparently it likes to get its leaves wet, but eventually deteriorated. The chives got vine lice after a few weeks and also turned yellowish, and they were never an intense green, which is why we got the LED UV lights, but they didn't improve the situation too much. The carrot plants lasted better, until one of the two got eaten by a caterpillar. The second one in the other planter kept growing, and eventually when we pulled out the harvest, the carrots had grown around the corner because they had reached the bottom of the planter. The basil stayed happiest of all plants, and kept growing. Basil is generally doing better with shade than the

Figure 4: Sensor readings for one planter over a two hour period (created using https://plot.ly/)

other vegetables we chose, and it doesn't root as deep, which is why we think some herbs are better suited for future indoor experiments than vegetables. The most important lesson from the deterioration of most plants after a few weeks is to next time use a bigger planter.

Data. In Fig. 4, we show a little extract of the sensor readings from our database. It shows the sensor readings of temperature, humidity, and moisture over an interval of two hours. In order to gain more insight from this data we need a comparative study, which is one of the steps in future work (planned for Fall 2018).

Affordability. The setup costs for the hardware and the gardening supplies summed up to $192 in total. We did not spend any money on software and relied exclusively on open source for developing our own software.

Continuation. This project was continued after the summer as second part of the senior design project that the students had already been working on during the Spring semester. The students are currently finalizing their software products supporting a more versatile use of the resilient smart garden implementation.

6 DISCUSSION

6.1 Computing within Limits.

This project is intended to provide students with the opportunity to work with computing within limits - in several dimensions.

- Natural limits: We only had two small planters due to little space and used a simple water bucket solution with a pump for the automated watering. Outdoors, this would be using grey water. In addition, we used means like companion planting to increase harvest.
- Economic limits: The constraints of a small budget made us stick to the bare essentials - we were able to keep the costs at $192, of which $25 were spent on garden ware that can mostly be reused, about $25 on seedlings and soil, and the rest on the computational and electronics hardware setup. We are confident this can be further reduced in the future, as we had to start from scratch with no reusable old parts of any kind.
- Technical limits: The sensors and boards were very basic and the software had to work with the limited availability of computing power.

Bridging Disciplines. In many topic areas relevant to limits, we have to bridge disciplines. While we had only computer science students working on this project, the topic would have lent itself well to an interdisciplinary team.

Companion Planting. The method mentioned above that we used in the project is companion planting. It is a technique that helps get the highest possible yield out of a limited amount of land by using synergies between plants. For example, the first one works as nitrogen fixer for the second one that, in turn, provides shade for the first one.

Non-computational Solution Alternatives. There is justified skepticism of approaches for decreasing consumption (of water, or other resources) that simultaneously employ active means (sensors, actuators, software, artificial lighting, etc.) while omitting much simpler, tried-and-true passive means (tenting/greenhouses, ollas, increased-organic-matter soils, etc.). The research project at hand uses active means because it was a feasible way to expose computer science students to a hands-on agricultural topic.

6.2 Benefits

Fun. The students had fun doing this project, so it was a research experience that keeps their motivation and interest up for further research. It was adventurous (as the excerpts of the diary show), they "made friends" with caterpillars, and had a high level of engagement. One student noted how easy it is to become a perceived expert in a domain, in that case particularly the gardening domain. Knowing the most out of a team of four, she had been declared the "expert" early on during the project. The unexpected "catastrophes" like the flooding of the lab floor due to the corroded sensor as well as the uninvited guests in form of various little pests on the plants certainly added to the entertainment factor of the research, and are lessons learned in terms of unforeseen side effects and risks.

Connection of several disciplines. The project provided a balance between disciplines and every team member, some with non-garden experience and some with less technological expertise, were able to significantly contribute. It was an opportunity for multidisciplinary work - the familiar computing domain and the unfamiliar application domain, which is something students will often encounter in their future work environments where software developers have to familiarize themselves with new application domains on a regular basis.

Hands-on, self-guided work. Students developed a strong connection to the project.

- Sense of ownership: Building it themselves from scratch.
- Sense of contribution: Everyone's contribution is visible.
- Sense of knowledge: In-depth knowledge and understanding the very foundation of electronics, hardware and software and the experience of putting them together.
- Sense of solution: Contributing to solving the California drought problems.
- Sense of control: DIY implies it is possible to deviate from it and modify to needs.

Scalability. It is possible to add another Arduino Board anytime, and therefore we can add as many temperature/humidity sensors as we need. The embedded system of the Resilient Smart Garden was designed with scalability in mind. A single planter can have as many sensors as needed to increase the accuracy of the readings as long as the Arduino has enough available pins. The sensor readings will be averaged to produce a value that will be sent to the server. The Arduinos and the nodeMCU communicate through I2C, meaning up to 128 Arduinos can theoretically be connected to a single garden. All extensions of hardware can easily be incorporated into the software by adding a few lines of code. The logic for the sensor readings and i2c communication will be scaled according to the amount of hardware that is integrated to the garden system. The amount of water pumps will depend on the amount of Arduinos implemented in the garden, as each planter is designated one water pump. The water pumps will scale along with the Arduinos but requires planning before implementing the power delivery system. The power supply will depend on the amount of water pumps that will be used and the amount of amps it will need. The terminal block that will distribute the power to the water pumps must also have enough rows for each water pump and the power supply. Once it has been planned out, the positive and negative wires from the water pump can easily be placed in a free row in the power terminal strip. Another set of power supply and terminal block may be added if there is an available power plug on the power strip in the case that further unplanned expansion is needed.

Flexibility. We are able to modify to certain needs, for example there are also other Arduinos that we can use for a same or similar setup. It doesn't have to be same brand of moisture sensor. We can use actual drip irrigation instead of crafting one out of vinyl tube.

Variety. We used one plant per companion plant group (figuring out "friends" and "enemies") and there are many other options for which plants can go into that size of planter. One task currently under work is developing a companion planting database that supports adequate choices for planting vegetables together in a confined space.

Wireless. Having wireless communication for flexible placement (not tethered to Ethernet cable) makes the system's deployment more versatile. A WiFi connection provides flexible placement of the garden system. Data collection of moisture, temperature, and humidity sensor readings from other Arduino DIY projects are done locally through an Ethernet cable which restricts the placement of the garden system. The placement of the garden will have to consider the location of the router, which could result in a lengthy Ethernet cable that would need to be routed. An ethernet cable to connect the garden and the server is not an ideal situation when placed in an outdoor environment. Having a WiFi connection will eliminate the limitation of being tethered to a router via Ethernet cable. It may seem counterintuitive to have a WiFi connection on the garden system when it is tethered to a power plug, but there is more availability of power plugs in a typical household in comparison to routers. The garden system is powered through a power strip that provides enough USB plugs for the Arduino and nodeMCU. The power strip needs to be plugged into the wall to deliver power to the garden system. Having a WiFi connection means having one less wire to consider when planning out the placement of the garden system. The quality of the connection will depend on the location of the router, but it can be easily extended with a WiFi range extender if needed.

Project Enclosure. Most of the DIY projects do not have a project enclosure. We provide documentation to build a simple project enclosure using common tools. In current work, we are developing a more weather resistant enclosure.

6.3 Limitations

Indoor setup. The lab we were using to perform the experiment came with a few limitations. We had a north facing room and therefore little light. We tried to mitigate this by installing LED UV lights, but we are not convinced of their actual efficiency[1]. Furthermore, the planters we chose fit the table designated for the project, but they turned out to be too small in the long run for the plants to grow effectively. Moisture level readings suggests that indoor environment has an influence on how frequent the garden is water. Moisture levels remained the same for a week without any watering. Plant symptoms suggests that the plants were over watered, even though water was rarely delivered.

Power source. Our system was powered from the wall, which can be restricting. In our current solution, we are limited to powering the system with a wall plug and an extension cord. This restricts us to deploying the system in an area that can be supplied by extension cord. However, in the next iteration of the system, we are planning to take it outdoors and use a solar energy source.

Maintainability. For the longer term use, we had to recalibrate the sensors a few times. One of our water pumps also burned out as a result of the bucket being emptied, and the system wanting to reach it's appropriate moisture level but no water ever reached the garden, so the system never shut off the pump allowing it run indefinitely in an empty bucket. We also had to refill the bucket multiple times The Resilient Smart Garden will automatically provide enough water to keep the plants alive, but it will require maintenance for proper functionality. The *automatic watering system* depends on the moisture sensor and it needs to be functioning correctly. The moisture sensors will eventually fail from corrosion/oxidation due to the moist environment, enhanced by the power that is being delivered to the moisture sensors. A failing moisture sensor gives the garden false readings which can flood the garden. The life span of the moisture sensor can be extended by limited the power duration, but the moisture sensors needs to be replaced (cannot be repaired). Some research and advice from peers with more expertise on sensors suggested that we look into alternating currents, which would mean having different voltages running along each prong of the sensors at different times, to prevent corrosion over time, and looking into gold-plated sensors. The *moisture sensors* need to have accurate readings for the automatic watering system to function correctly. Each moisture sensor will have different maximum values due to the analog output. Each moisture sensor will need to be individually calibrated to ensure accurate readings. The calibration process involves putting the moisture sensor in moist soil and getting the maximum raw value that is outputted. The calibration process requires judgment from the user and it may be prone to error if not properly performed. The *water bucket* must have water to ensure that the plants are receiving water and the water pumps will not burn out. The Resilient Smart Garden was intended for outdoor usage, but this research was conducted indoors. As a result, the water source for the garden system is a bucket that requires

[1]We think that they may actually have been counterfeit, as later on one student took the planters home and outside, where the plants went back to growing.

the user to check and make sure that there is water. There must be water in the bucket for water delivery system to function as intended. The plants will be put at risk if there is no water in the bucket. The water pump is also at risk since it will continue to run until the moisture levels are back to normal values. The water keeps the water pumps cool and will burn out without it.

Cost. We used low cost components all the way, which in part led to components failing early, specifically the moisture sensors. We have to run a series of tests to find out where it is more feasible in the long run to invest a few dollars more into a specific component and what the most resilient hardware setup for the system is.

Insights. We didn't have a comparative study set-up, which is why our insights are not as extensive as they could have been in a comparative setting. However, the exact numbers on how much potential water savings we could have had in a comparative setting are also strongly influenced by the particular environment of the room and building we were hosted in. The follow-up study outdoors will have a comparative set-up.

7 CONCLUSIONS

With a rapidly changing climate, we notice a significant effect on the environment, most noticeable in the drought-prone state of California. Every year, California faces severe drought during the summer, sometimes resulting in a state of emergency. Conserving water is crucial but it is difficult due to California's agricultural production. This results in a societal effort to become more sustainable, motivating individuals to take an interest in growing their own food. Despite the good intentions, few people have gardening experience and even fewer are educated in water conservation. Others have attempted to remedy this issue with automated systems, however they do not account for scalability or ease of replication for the average consumer. By developing a smart resilient garden using basic IoT devices we can promote a sustainable system that anyone can build and use with minimal difficulty. An open source system will allow the average user to integrate a smart garden into their homes and promote sustainability and collaboration in community gardens and open source development for future enhancements.

The **objective** of the Resilient Smart Garden is to minimize water usage while maintaining a sustainable environment through automation. The Resilient Smart Garden is built using commercial off-the-shelf parts, making it accessible to anyone. There are no strict build guidelines, providing flexible integration. Expansion requires little effort as the hardware is designed to easily scale.

The **contribution** of this paper is a **prototype of an educational blueprint for a multidisciplinary project that supports community resilience**.

Future Work

The Resilient Smart Garden helps fight the drought by providing knowledge and material to answer the call to action. On the foundation provided in this paper, we plan several strands of future work as follows.

Outdoors. The first step in future work is to repeat the experiment outdoors with the equipment in an enclosure as the current version does not have a project enclosure. Options we are currently exploring are to use a ready-made box, to 3D print one in the on-campus 3D printing lab, or to repurpose some other materials.

Innovation challenge. The students are currently participating in a team in the campus-wide Innovation Challenge to propose an easily replicable Resilient Smart Garden Education Kit with an online portal that educates the user beyond set-up tutorials on how to gain the best possible yield from a tiny to small-sized garden.

User intention. We are planning an empirical analysis on factor affecting the user's behavioral intention to use DIY kit for smart gardening. For example, factors can be self-efficacy, usefulness, attitude towards usage, benefit, etc. One hypothesis could be Attitude will have a significant positive effect on Intention to Use DIY kit for smart gardening. Future studies will be testing these hypotheses.

Scalability. As mentioned throughout the report the Resilient Smart Garden can easily be scaled up to include multiple gardens by adding Arduinos for each garden. Another opportunity to scale up the project in the future lies in expanding the Resilient Smart Garden past the garden and into agriculture by adapting the system to work with different types of irrigation systems [38]. One of the objectives of this project was to facilitate raising a garden in areas dealing with droughts, like Southern California. As mentioned by the California Ag Water Stewardship Initiative [38], water quality is no longer the only concern when it comes to agriculture. With the growing population within California water becomes a more limited resource, so the concern is to use sufficient water that crops get the necessary amount while minimizing the amount of water used.

While the project at hand had constraints that make it hard to draw conclusions on the water savings, we are still interested in further investigating the potential for recharging groundwater and supporting non-crop ecosystems.

Greywater solution. Our long-term vision is a Resilient Smart Garden Showcase that uses greywater (e.g., collected rainwater) to water the plants and thereby makes the system even more environmentally sustainable. Given our setting in Southern California that would include a more sophisticated collection system that includes the collection of morning dew [24]. Using renewable energy by connecting the system to solar panels is planned for the next iteration.

Education kit. A future iteration of the DIY kit with evaluated user instructions could become an educational toolkit for use in schools as well as ICTD work for developing countries.[2]

Comparative study. We are planning to conduct a comparative study that evaluated the effectiveness of active means to lower resource consumption (sensors and water pump) with passive means like increasing the soil quality with organic matter.

ACKNOWLEDGMENTS

The authors would like to thank Margo McCall for reporting on the project and for her gardening advice. The work is supported by a CSULB 2017 Recreational Scholarly and Creative Activities Faculty Small Grant.

[2] A research proposal by principal investigators joining forces between several departments and colleges is currently under review.

REFERENCES

[1] W. Neil Adger, Nigel W. Arnell, and Emma L. Tompkins. 2005. Successful adaptation to climate change across scales. (2005), 77–86 pages. https://doi.org/10.1016/j.gloenvcha.2004.12.005 arXiv:1307.3552

[2] Amberg Thomas. 2016. Augmented Water. (2016). http://www.instructables.com/id/Augmented-Water/

[3] Muhammad Aqib. 2017. Advanced Garduino with Data Logging to Database. (2017). https://www.hackster.io/muhammad-aqib/advanced-garduino-with-data-logging-to-database-9adf69

[4] S. Z. Attari. 2014. Perceptions of water use. *Proceedings of the National Academy of Sciences* 111, 14 (2014), 5129–5134. https://doi.org/10.1073/pnas.1316402111

[5] Jeff Atwood. 2012. Rubber Duck Problem Solving. (2012). https://rubberduckdebugging.com/https://blog.codinghorror.com/rubber-duck-problem-solving/

[6] Filipe Caetano, Rui Pitarma, and Pedro Reis. 2015. Advanced system for garden irrigation management. In *Advances in Intelligent Systems and Computing*, Vol. 353. 565–574. https://doi.org/10.1007/978-3-319-16486-1_55

[7] Carson Miller. 2015. *Automated Aquaponics Design Report.* Technical Report. http://muse.union.edu/carsonmiller/files/2014/08/ECE-499-Final-Paper.pdf

[8] Susan Crate and Mark Nuttal. 2012. Anthropology and Climate Change: From Encounters to Actions; Political Theory and Global Climate Change. *Ethics, Policy & Environment* 15, 1 (2012), 125–129. https://doi.org/10.1080/21550085.2012.672696

[9] D. V. Daniels. 2016. Automatic Garden Watering Device - Arduino. (2016).

[10] G R Dixon, R H Collier, and I Bhattacharya. 2014. An assessment of the effects of climate change on horticulture. In *Horticulture: Plants for People and Places.* Vol. 2. 817–857. https://doi.org/10.1007/978-94-017-8581-5_10

[11] Elena Domene, David Saurí, and Marc Parés. 2005. Urbanization and Sustainable Resource Use: The Case of Garden Watering in the Metropolitan Region of Barcelona. *Urban Geography* 26, 6 (2005), 520–535. https://doi.org/10.2747/0272-3638.26.6.520

[12] Xiao Juan Duan. 2014. Research on IOT-Based Smart Garden Project. *Computer-Aided Design, Manufacturing, Modeling and Simulation IV* 608 (2014), 321–325. https://doi.org/10.4028/www.scientific.net/AMM.608-609.321

[13] E2i Creative Studio. 2013. Connected Garden. (2013). http://e2i.ist.ucf.edu/project/8

[14] Edyn. 10 October 2017. Edyn. (10 October 2017). https://edyn.com/

[15] Liangxin Fan, Fei Wang, Guobin Liu, Xiaomei Yang, and Wei Qin. 2014. Public Perception of Water Consumption and Its Effects on Water Conservation Behavior. *Water* 6, 6 (2014), 1771–1784. https://doi.org/10.3390/w6061771

[16] GreenIq. 2017. GreenIq. (2017). http://greeniq.co/

[17] Hamza, A. 2017. How to Make an Arduino Temperature Data Logger. (2017). https://diyhacking.com/make-arduino-temperature-data-logger/

[18] Siobhan Hatton-Jones and Min Teah. 2015. Case analysis of the do-it-yourself industry. *Asia Pacific Journal of Marketing and Logistics* 27, 5 (2015), 826–838. https://doi.org/10.1108/APJML-09-2015-0135

[19] Sara Heitlinger, Nick Bryan-Kinns, and Janis Jefferies. 2013. UbiComp for Grassroots Urban Food-growing Communities. In *Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication (UbiComp '13 Adjunct)*. ACM, New York, NY, USA, 589–594. https://doi.org/10.1145/2494091.2497617

[20] Richard Howitt, Josué Medellín-azuara, and Duncan Macewan. 2014. Economic Analysis of the 2014 Drought for California Agriculture. *Center for Watershed Sciences. University of California, Davis, California* (2014), 20p. http://watershed.ucdavis.edu

[21] Iseman, L. 2012. Garduino: Geek Gardening with Arduino. (2012). https://makezine.com/projects/make-18/garduino-geek-gardening/

[22] Legislative Analyst's Office. 2016. *The 2016-17 Budget: The State's Drought Response.* Technical Report. Legislative Analyst's Office. http://www.lao.ca.gov/reports/2016/3343/drought-response-020516.pdf

[23] Maleficarum Maleus (*pseudonym*). 2016. Water Saving System Using Arduino. (2016). http://www.instructables.com/id/Water-Saving-System-Using-Arduino/

[24] Jonathan Margolis. 2012. Growing food in the desert: is this the solution to the world's food crisis. (2012). https://ourworld.unu.edu/en/growing-food-in-the-desert-is-this-the-solution-to-the-worlds-food-crisis

[25] Maira Marques, Sergio Ochoa, Maria Cecilia Bastarrica, and Francisco Gutierrez. accepted for publication in 2017. Enhancing the Student Learning Experience in Software Engineering Project Courses. *IEEE Transactions on Education* (accepted for publication in 2017).

[26] Massachusetts Institute of Technology. 2017. Personal Food Computers. (2017). https://wiki.openag.media.mit.edu/personal

[27] Mary H. Meyer, Douglas Needham, John Dole, Brain Trader, Jennifer Fox, Marnie Conley, Michael Neff, and Jean Shaw. 2016. Importance of horticulture and perception as a career. *HortTechnology* 26, 2 (2016), 114–120.

[28] Bill Mollison and David Holmgren. 1978. *Permaculture.* Lesmurdie Progress Association.

[29] National Institute of Food and Agriculture. 1893. USDA Definition of Specialty Crop. (1893). https://nifa.usda.gov/sites/default/files/resources/definition

[30] Palmer Andrew. 2015. Smart Sprinkler: Water Saving Irrigation Controller for Residential Use. (2015). https://people.ece.cornell.edu/land/courses/eceprojectsland/STUDENTPROJ/2014to2015/ajp294/smart

[31] Camille Parmesan. 2006. Ecological and Evolutionary Responses to Recent Climate Change. *Annual Review of Ecology, Evolution, and Systematics* 37, 1 (dec 2006), 637–669. https://doi.org/10.1146/annurev.ecolsys.37.091305.110100

[32] Pollinator Partnership. 2017. Bee Smart Kit for elementary school kids. http://pollinator.org/bee-smart, last accessed on Feb 8th, 2018. (October 2017).

[33] William E Rees. 2010. Thinking "resilience". *The post carbon reader: managing the 21st century's sustainability crises. Watershed Media, Post Carbon Institute, Santa Rosa* (2010).

[34] J. Sardans and J. Peñuelas. 2005. Drought decreases soil enzyme activity in a Mediterranean Quercus ilex L. forest. *Soil Biology and Biochemistry* 37, 3 (2005), 455–461. https://doi.org/10.1016/j.soilbio.2004.08.004

[35] The National Gardening Association. 2014. National Gardenting Association Special Report: Garden to Table: A 5 Year Look at Food Gardening in America. (2014), 24. https://garden.org/special/pdf/2014-NGA-Garden-to-Table.pdf

[36] Tower Garden. 2017. 6 Reasons Tower Garden Is a Great Teaching Tool. (2017). http://www.towergarden.com/grow/school-gardens

[37] Reinier van der Lee and Gallardo Patricio. 2015. Vinduino, a wine grower's water saving project. (2015). https://hackaday.io/project/6444-vinduino-a-wine-growers-water-saving-project

[38] Kaomine Vang and David Zoldoske. [n. d.]. Irrigation Management. ([n. d.]). http://agwaterstewards.org/practices/irrigation

[39] Ankit Vashista, Harsh Rathore, and Gaurav Jain. 2016. Automatic Gardening System using Arduino. *SSRG International Journal of Electronics and Communication Engineering* 3, 8 (2016), 119–120.