**GENERAL DESCRIPTION**

“Removing Human Error” this is the main problem that the proponents of this project wanted to solve. Many of the maintenance crew on Urban green seldomly forget to open some of the water systems in the Gutters in their Hydroponic system when they are doing maintenance on it, and with that even if it’s just 1 water valve that is left closed, a whole layer of plants is in verge of dying in that process and that would be a hundred of plants that will not be sold for that reason only. With that, Urban Greens CEO Ralph Becker, assigned his Engineering interns to think for a solution that would help Urban Greens in Reducing Human error.

The Engineering interns have proposed a Smart Valve system with Blynk Application for monitoring and control, that is an Arduino based automation which the proponents think that would solve the problem. The System is an Automated System incorporated with IOT. The proponents of the system Uses Arduino UNO as the brain for their operation on Automation that will be use in order for the system to work. The system is connected to Blynk application so that any of the of the users who has access on the application, or the system can monitor or control what they want to do on the system. The proponents use C language on programming the system, with this any of the user who have small knowledge on the C language programming can modify the system on how they want it to work.

**OBJECTIVES**

The main Objective of this project is to “Reduce the Human Error” so that Urban green can achieve Automation on their systems and to avoid the mortality rates of plants on the process of growing them. As for the other objectives, these are the objectives that the proponents of this projects wanted to achieve:

1. To connect the System to Blynk server to achieve IOT.
2. To achieve automation, monitoring and control using the Blynk application while the system is working.
3. To achieve efficiency, effectiveness of the project to the Hydroponic system, and to cost less on the production as much as possible.

**REQUIRED MATERIALS**

|  |  |
| --- | --- |
| **4 layer Smart Valve System** | **12 layers Smart Valve System** |
| Arduino Uno | Arduino ATmega |
| Breadboard or protoboard – 2 6x8cm | Breadboard or protoboard – 2 or 4 6x8cm |
| 8 channel Relay | 16 channel Relay |
| Normally OPEN Plastic Solenoid Valves – 4 pcs | Normally OPEN Plastic Solenoid Valves -12 pcs |
| Tact Switch – 4 pcs | Tact Switch – 12 pcs |
| Resistors 220Ω - 4 pcs | Resistors 220Ω - 12 pcs |
| LED lights red – 4pcs | LED lights red – 12 pcs |
| Junction box 255x200x80mm | Junction box 255x200x80mm  (Change if necessary) |
| LED Lamp 12 Dc - 1pc | LED Lamp 12 Dc – 2pcs |
| Heat Shrinks – 1 set | Heat Shrinks – 1 set |
| Terminal Lugs | Terminal Lugs |
| Jump wires - male to female | Jump wires - male to female |
| Standoff Screws | Standoff Screws |
| Regular plug | Regular plug |
| Power supply 12V 10A | Power supply 12V 15A |
| Diode 1n4001 – 4 pcs | Diode 1n4001 – 12 pcs |
| ESP- 01s | ESP- 01s |
| ESP link | ESP link |

**FEATURES**

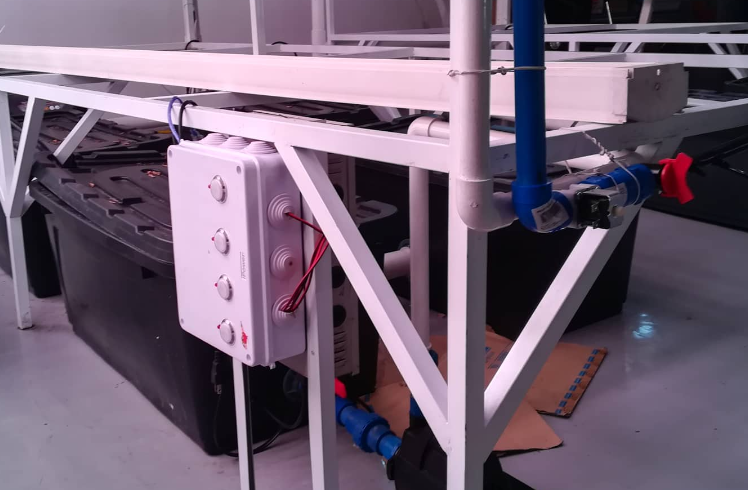
The Main feature of this system that it is a Smart Automation, where you can Control and Monitor the Smart Valves.

Chart

Description automatically generated with medium confidence

As of now the User interface on the Blynk application look like this, since the system is still in 4 layers. As stated earlier, it is a Smart System that can monitor and control the Smart valves. As you can see, we have 4 layers and they also have 4 identifications or states if the valves are Open or Close. Those identifications can also serve as a button, so once a user taps on the open on the Leyer 4, it will indicate there that you will be closing the water valve on layer 4, same goes to the other layers. As for the BULB STATE, it will be the indicator for the Users to know if one of the LAYERS are OPEN or CLOSE. So once 1 of the LAYERS are CLOSE the BULB STATE will light up and once every LAYERS have been open the BULB STATE should Light off.

**DESIGN**

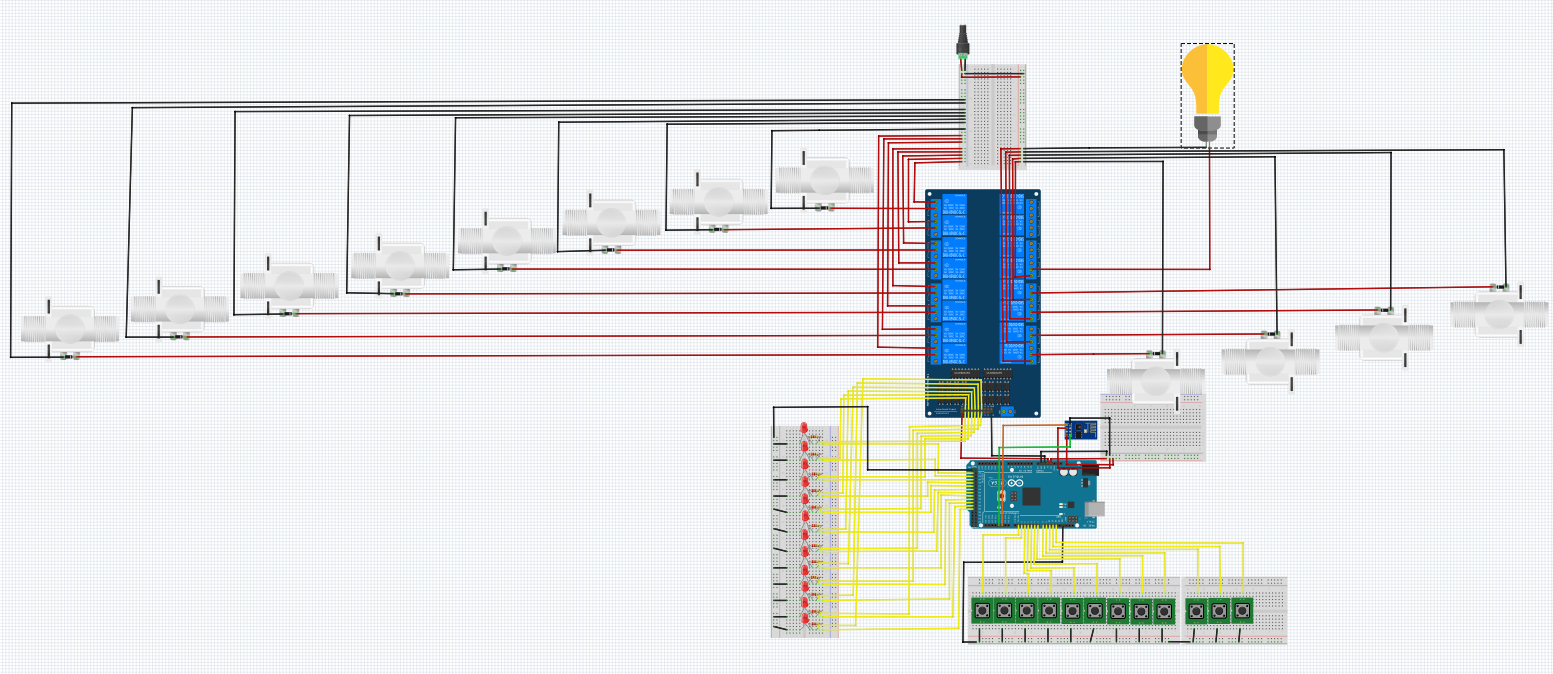


**Wiring Diagram**

Diagram

Description automatically generated

­­4 LAYERS



12 LAYERS