**Design and Implementation of Smart Mailbox**

**System using Arduino Yun**

Faculty of Computing & Electronic Engineering   
Department of Engineering and Design  
Institute of Technology Sligo, Ireland

Cillian Mc Meekin: [Cillian.mcmeeking@mail.itsligo.ie](mailto:Cillian.mcmeeking@mail.itsligo.ie)Davey Feeney: [Davey.feeney@mail.itsligo.ie](mailto:Davey.feeney@mail.itsligo.ie)Jessica Henry: [Jessica.henry@mail.itsligo.ie](mailto:Jessica.henry@mail.itsligo.ie)Paul Mc Gonigle: Paul.mcgonigle@mail.itsligo.ie

­­

Trello: https://trello.com/b/otrPkRB9/iot  
GitHub: <https://github.com/Box4U/IoT-Project-Code>  
YouTube: https://www.youtube.com/watch?v=E8e-XA-72NU

1. **Summary**

Box4U is a smart mailbox we are creating using an Arduino Yun. The device consists of a sensor that will detect when there is something in the mailbox. When the sensor is activated the owner of the mailbox will receive an alert to let them know post is there. This box will be mainly of use for people who might be immobilized or disabled so that they won’t waste precious energy on trips to their post-box, or for people who might have very long driveway, so they will not have to go down for nothing. This project is a very good addition to any smart home and is a must have for anyone looking to build up a range of smart technologies in their home. The mailbox will need a Wi-Fi connection and it will also need a constant source of power, we also had to make sure that the Wi-Fi could reach through the metal of a post-box. The post-box also had to accurately go off once the post had arrived, these problems were all thought if while we were working our way towards having a functional and practical solution to our problem.

Many people when checking their mailbox are faced with the empty mailbox, this wear becomes greater when the person is waiting for an important letter and outside is in a bad weather.

We have decided to try and solve this problem with our project, a smart mailbox, the BOX4U. Our team will attempt to maximize efficiency of the physical mailbox by finding a way to tell the owner that their mail has arrived, instead of the owner having to check the box multiple times a day, or having to ask a neighbor to check the box whilst the owner is absent or away.

1. **Literature Review**

American James H. Baggarly was a pioneer in associating internet of things in logistics and postal services, in 29 June 1999 he patented electronic mailbox with keyboard alarm system. Since then, the concept of the problem and analysis of the current state has been studied and developed and with each study a proposal for improvement is created in order to become increasingly useful in people's lives.

Another important character in the history of the evolution of smart mailboxes was inventor Orion A. Roberson that in 2002 patented what was named the Mail check method and system. A system for determining the presence of contents in a mailbox from a remote location regardless of distance comprising:

* a mailbox sensor for detecting the presence of contents in a mailbox, evaluation of mailbox content, said sensor being positioned in the mailbox;
* a mailbox control unit in communication with said mailbox sensor for storing information on the status of a mailbox, the status being the presence of contents in the mailbox; and
* a remote access device to enable a user to contact said mailbox control unit from a remote location to the mailbox to obtain the status of the mailbox.
  1. **Project Objectives**

The goal of the project is to create a mailbox system alerting the user when a new correspondence / letter arrives.

With the internet of things (IoT) resources, the sensor detects that the new letter triggers an alert for the registered user's application, sent it will send an alert to the application on the user's cell phone. When the mailbox is opened, it is understood that the user has removed the incoming correspondence, making it possible to restart the correspondence history.

The IoT is used to communicate via wi-fi, to send the alert to the application on the user's cell phone and e-mail.

**2.2 Wi-Fi Getting to Arduino**

A very big part of our project solution was how to get Wi-Fi to our Arduino device. The entire project rests on the Arduino being able to connect to the users Wi-Fi so it can work and transfer messages to the user’s device. To connect the Arduino, we had to follow the appropriate steps to get the Arduino an Ip address using the Wi-Fi status sketch to see if it was connected and if it had its own independent Ip address. With the Arduino now connected to we are able to run API’s and make the sensors send Data to another device. Another problem we worried about was when the sensors was installed in the mailbox, that the metal could potentially stop the Wi-Fi signal from getting into the Arduino inside the mailbox, but after testing it was discovered that this wasn’t a problem and the Wi-Fi was able to reach the Arduino with no problem inside the Arduino.

**2.3 Blynk – User interface**

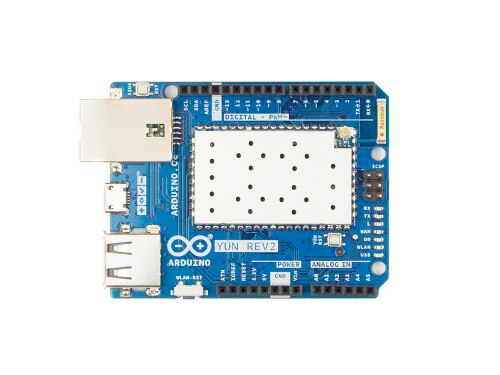
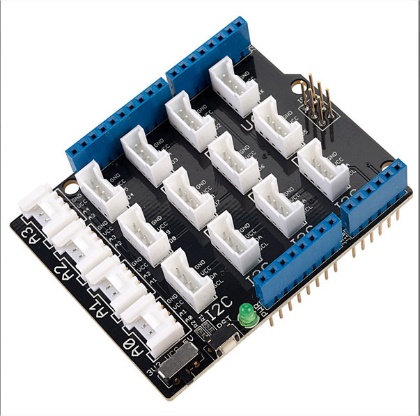
For the integration with the mobile interface, an existing software on the market was used.

Blynk is an internet of things platform which allows controlling electronic devices remotely using its iOS and android apps. It provides dashboard by which user can create

graphic interface using different widgets. Blynk can also store and display sensor data and provides libraries for most of the popular hardware platforms like Arduino and Raspberry.

**2.4 Arduino UNO**

Arduino is an open-source platform used for constructing and programming of electronics. (O`Reilly Media, Inc, 2009). It can receive and send information to most devices, and even through the internet to command the specific electronic device. It uses a hardware called Arduino Uno circuit board and is programmed using the programming language C++.



Grove Kit Seeed - Orangepip Grove Base Shield V2

Arduino YUN REV 2

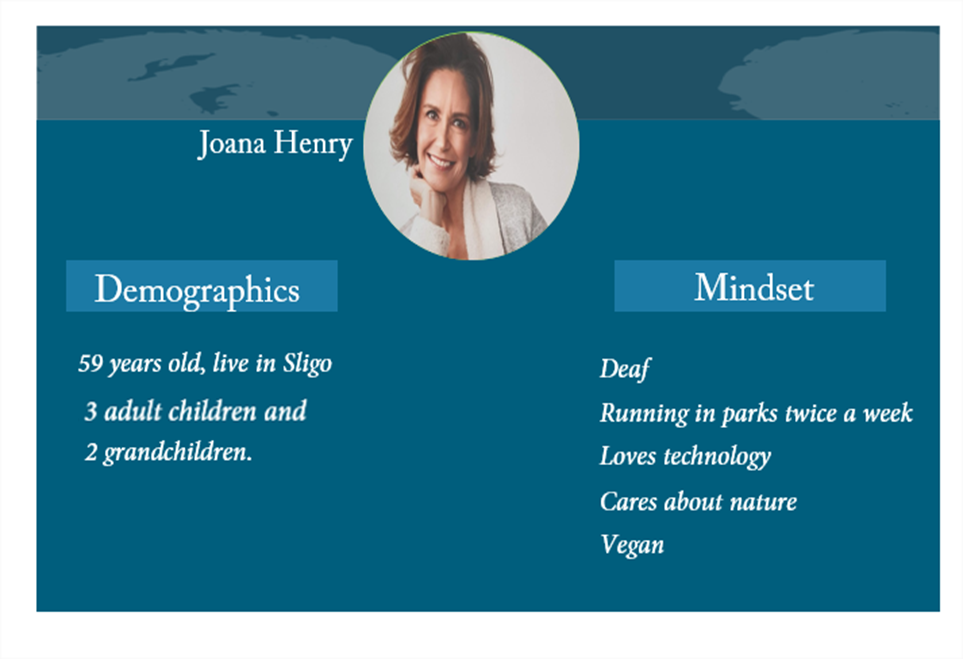
**3. System Design and Implementation**

This chapter describes the design of the software component of this research project and discusses the various design decisions that have been implemented to

test.

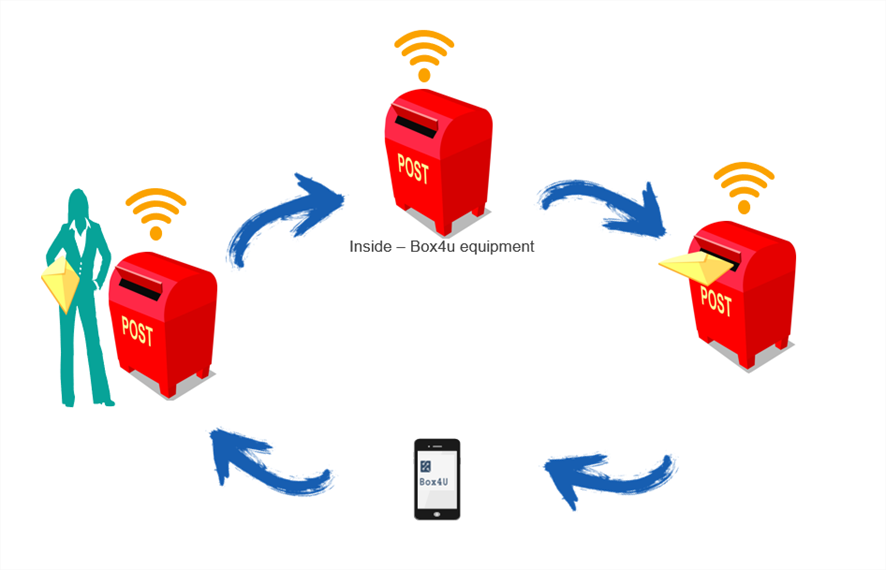
**31.1. Persona**

Joana Henry is the perfect candidate for our Box4U, she is 59 years old, deaf, loves running in the park and love technology.  
This smart solution would help Joana out due to her impairment and her being very busy with her exercise and children.

****

**3.1.2 System Diagram**

The Software Design of Smart Mailbox consist of a program in Arduino environment, connected with wi-fi a simple user interface mobile solution. The function of the Arduino ensures and maintains the correct functioning of the components. From then on, it is possible to connect to wi-fi, it also has to be connected with a database and API to send email and notification.

****

**3.2 System Requirements**

This section reiterates the problem this research project is aiming to solve. The system requirements include functional and nonfunctional requirements. Functional requirements regard what the system does while nonfunctional requirements outlines the manner in which the functional requirements will be accomplished**.**

**3.2.2 Functional Requirements**

Run in real time  
Detect the entry of correspondence in the mailbox   
Must send the notification to user when have a new correspondence   
Must connect to an external device   
Must to connect with wi-fi   
Must keep the user up to date on the status of the box

Run for long period of time

**3.2.3 Nonfunctional Requirements**

Easy to set up  
Maintainable  
Precise  
Economic  
Environmental  
Scalability  
Safe

**3.3 Project Constraints**

One of the main constraints dealt with in the project is the development of the mobile application interface.

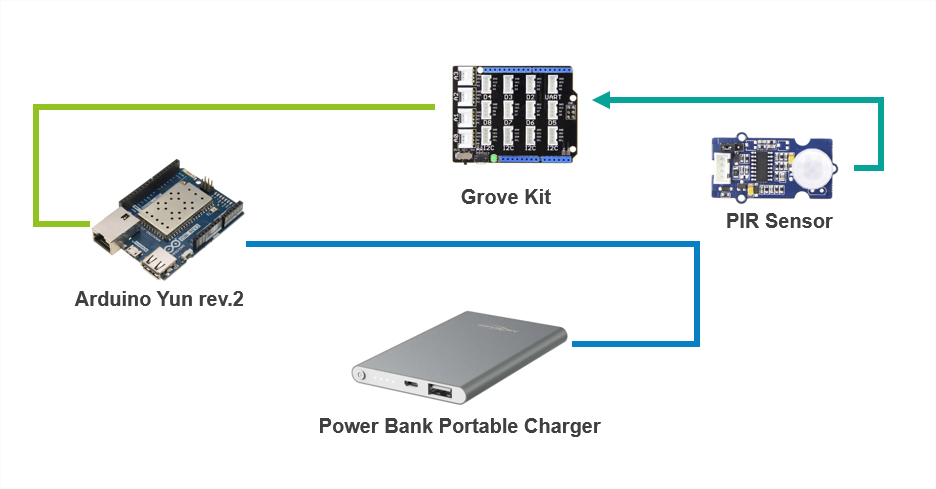
**3.3.1 Technologies**

This section is included the programming language C++, software libraries, GitHub repositories and resources that were used in the prototype implementation.

In addition to the Arduino YUN REV 2 board and the grove Seeed Kit, additional hardware and software were needed for use in the project. Access to a computer and cell phone was also necessary to make use of Arduino IDE software and create interactions.

Below we describe the necessary components and the main functions.

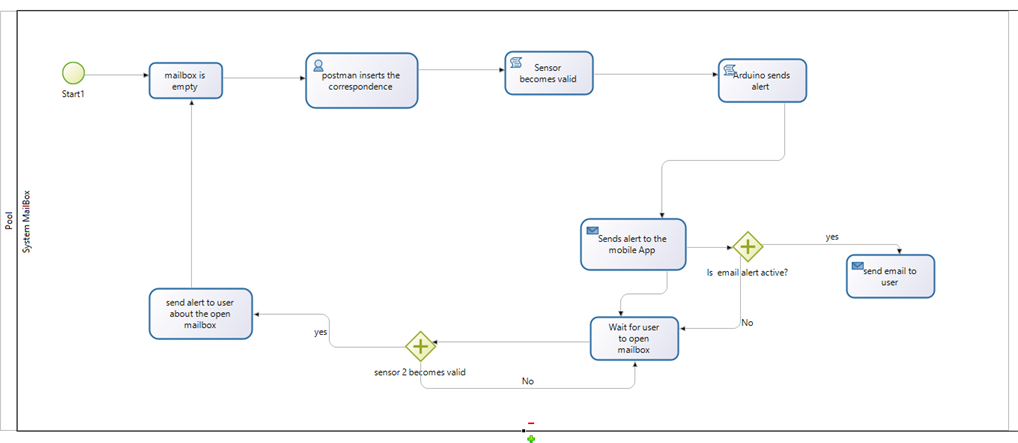
**3.3.2 Components**



**3.3.4 About Version Control**

Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later. For the examples in this book, you will use software source code as the files being version controlled, though in reality you can do this with nearly any type of file on a computer.

1. **System Model Process**



* + 1. Library for the Arduino IDE and APIs

**4.1 Design Solution**

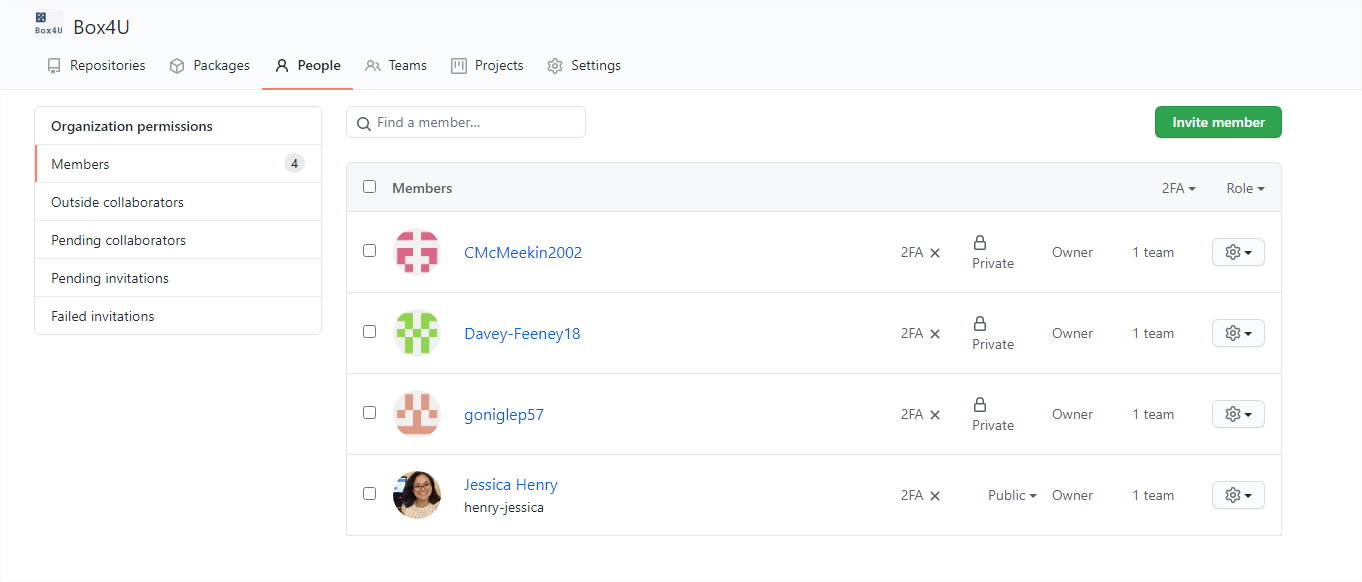
Presentation of the proposed model. First version of the project. The proposal of this project is the elaboration of a prototype of a box of residential mail, which aims to notify the user of a residence through notification on the cell phone, when a new correspondence is inserted in the mailbox. post office. The system also informs about the opening of the box of mail being understood as soon as the correspondence was collected.

Prototype version one bellow:

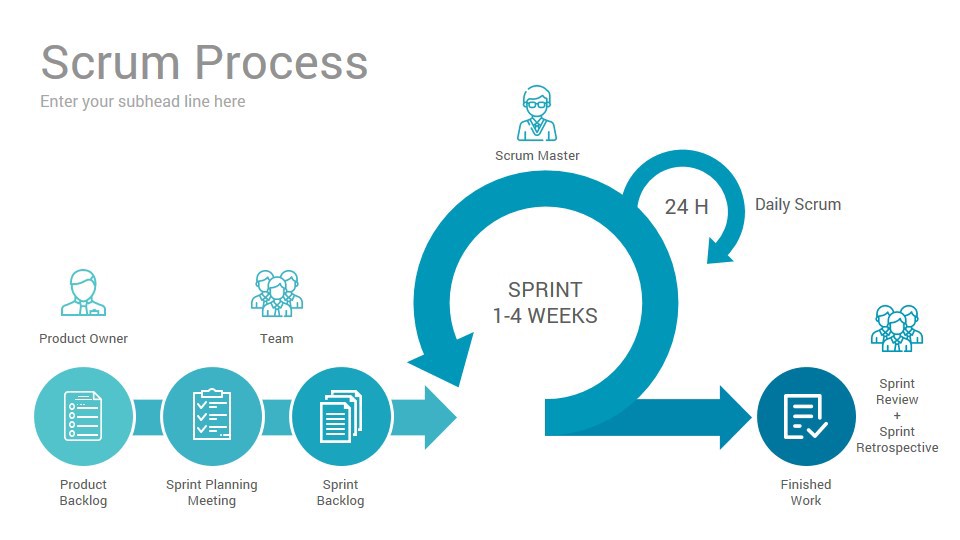
****

**4.2 GitHub**

For versioning the system development, we will use GitHub. A private organization was created and all team members were inserted. It was possible to follow the development of the system through GitHub.

GitHub: https://github.com/Box4U/IoT-Project-Code

**4.3 System Implementation**

Scrum Agile is framework efficiently and effectively. It can be used for any type of project, but it was founded mainly on software development. Scrum can be divides larger projects into small. At the end of each iteration, something of value is produced. The product that is produced during each iteration must be a usable product to obtain feedback from users or interested parties.

For the development of Box4U, the Scrum Framework was chosen because we are working with a relatively small team where we did not want to follow cascade orders and thus leave team members idle waiting for one tariff to be completed to start another one, just as it happens in the cascade method. The model was flexible and fit well in the team.

In our first team meeting was held, even before deciding on the idea to be developed, we arrived at the first question: What methods of managing the project are we going to use?

The decision was to perform Scrum Agile because it is in the process to focus on delivering the business value in the short time, using the sprints was efficient way to get results and solutions after each sprint.

The core roles: Scrum master, product owner and scrum team (development team).

**Scrum Master:** He was the facilitator of the scrum development process. Starting our daily meeting in snapchat with the team. Also, the scrum master makes certain that scrum rules was being applied as intended. The closing and opening meetings of sprints were organized by the scrum master.

**Product Owner:** Focused on the end customer, the P.O followed the solution design process. Determining the specification of the box, the audience being served, the creation of the persona, the changes in the product, in the development of Box4U the PO managed the Backlogs.

**Scrum Team** – Development: The scrum team focused on solve programs and produce deliverable product. The development team can organize itself, so it is possible to perform the sprints in a dynamic and flexible way

When the task has been defined and a time is inserted for the activity, it makes the interactions of the activities more dynamic and less worrying. As the tasks were still something to be studied, during the execution of the tasks, the development became clear and a large and complicated task became something smaller and simpler to solve.

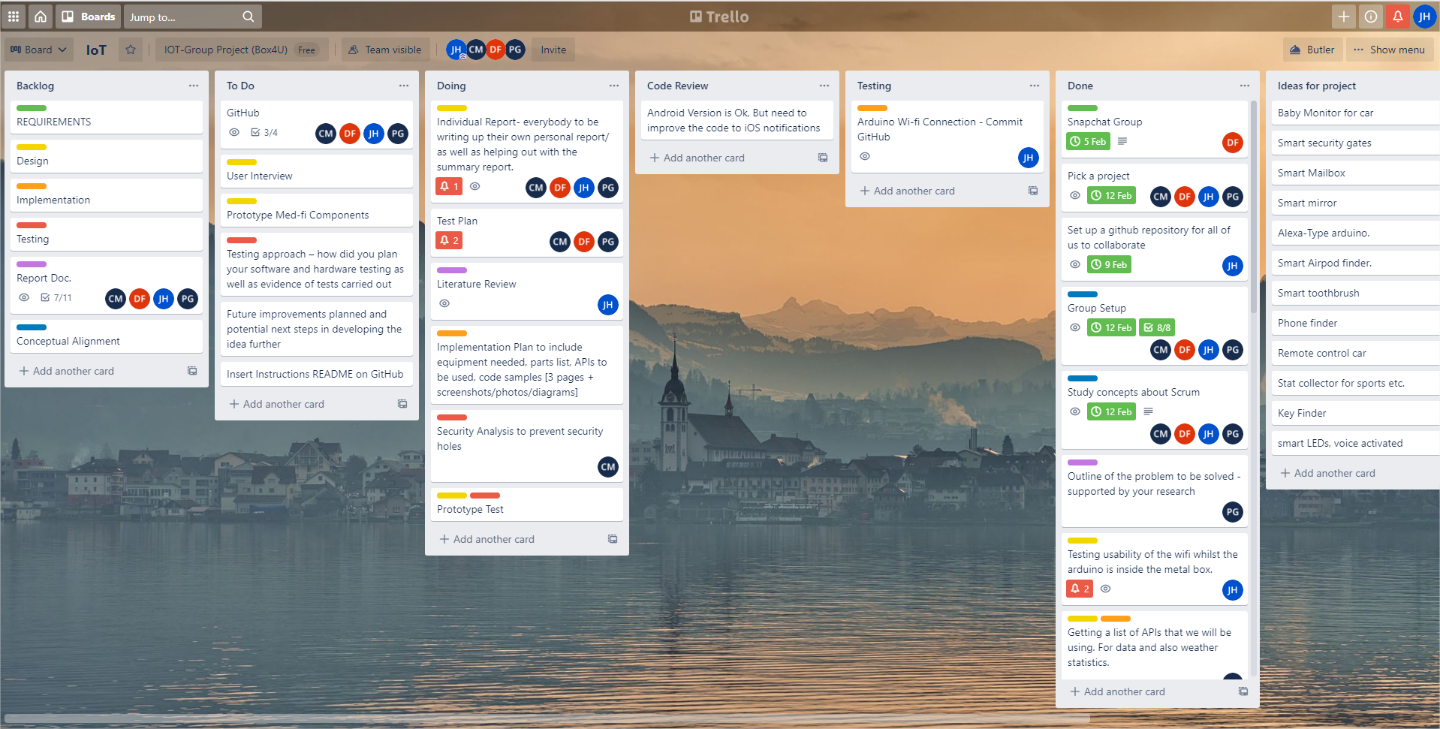
**Box4U Scrum**

For the project the Sprints life was 7 days, the meetings were held daily through the Snapchat group. We use Kanban.

Kanban is a visual system for managing software development work. For this we used Trello to organize the interactions backlog sprints, what needs to be done, what we are doing and what has been completed.

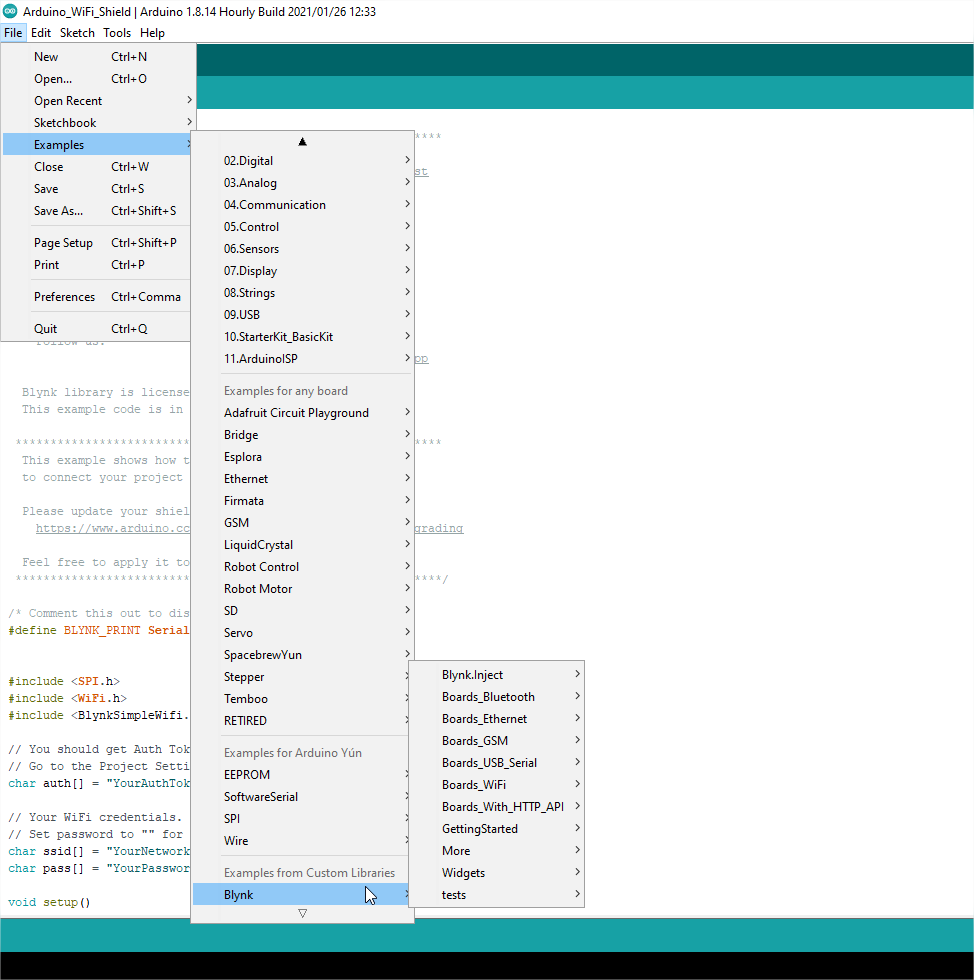
**4.4 Trello & Kanban**

Our Trello was organized as follows: Each colour refers to a backlog. and the sprints follow these colours, so the team can follow in a more organized way what is pending for which backlog.



**4.5 Implementation Code**

Para implementação foi necessário incluir a biblioteca Blynk, assim conseguir realizar a integração.



**4.6 Coding**



line 7: Connect with sensor in port D2 grove kit.

Line 8: The systems starts false, because the mailbox starts empty.

Line 9: A variable bool was created because it will control the status of the mailbox.

In the main if the sensor detect something it get the Boolean mail\_status true and print the messages. After 5 seconds it send the notification to blynk.run at the beginning of the interaction.

**5 Testing Plan**

1: Detect the entry of correspondence in the mailbox

**PLAN:** Have the sensor itself constantly updating (every second), and if there is a change in movement, or there is movement detected, then the sensor should send a signal to the Arduino, and then to a spreadsheet API.

2: Must send the notification to user when have a new correspondence

**PLAN:** Test out the first plan, and if all is a success then step two will be two check if a notification is sent to the ‘testers’ phone, if not then we will have to make adjustments to the code.

3 Must to connect with wi-fi   
**PLAN (3,4):** Using the wifiStatus sketch from the example sketches and check the Arduino ‘monitor’ to see if the Arduino has an IP-Address, if it does then that means the Arduino is connected to an external device(wi-fi Router), if not then we will have to make adjustments to our project.

5: Must keep the user up to date on the status of the box-  
PLAN: We need to connect the Arduino to an external application on the users phone, so that the user can ping the Arduino whenever he/she wants .

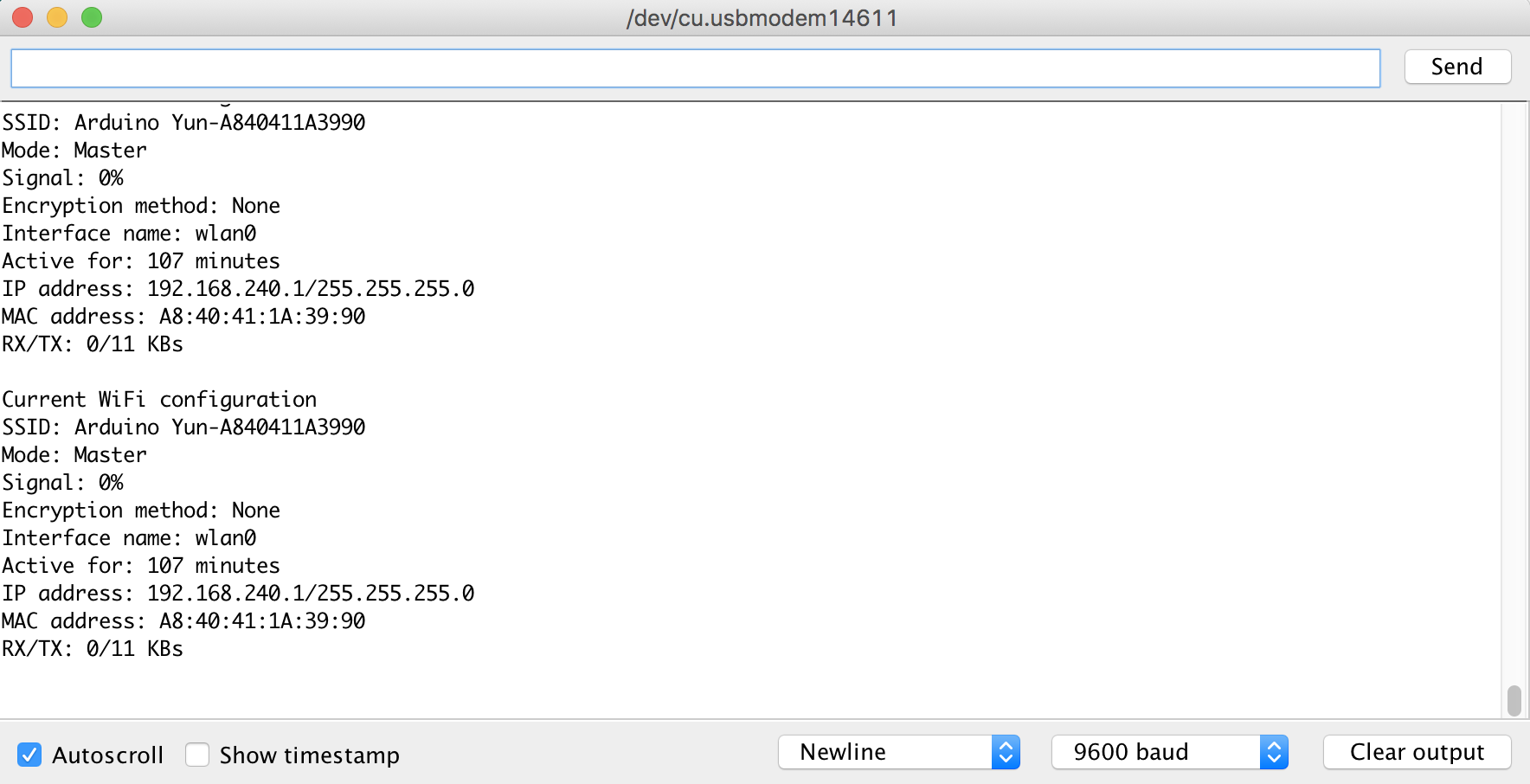
6: Run for long period of time--

PLAN: We can check this out after we have the project built, we will first be checking with the battery to see how many hours the project can stay alive for. If this is insufficient then we will have to replace the battery with solar panels.

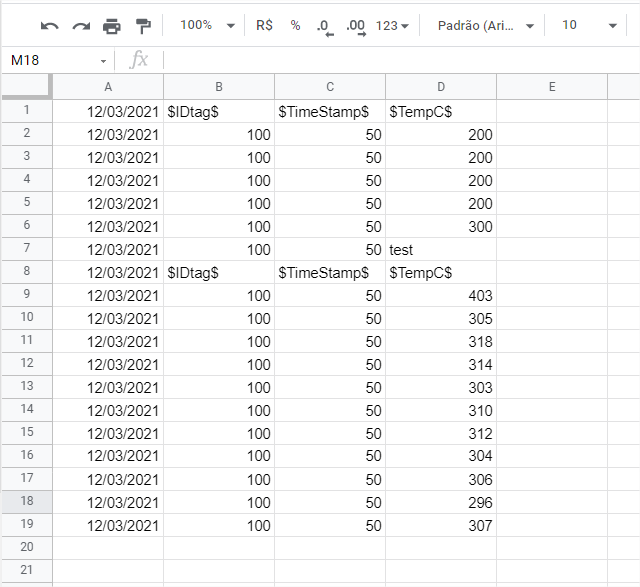
7: Run in real time-  
PLAN: We will code the Arduino to check every constantly for a change in movement. We will first test this out with PushingBox, to update the spreadsheet every second, if this works well then it will show that it is running in real time.

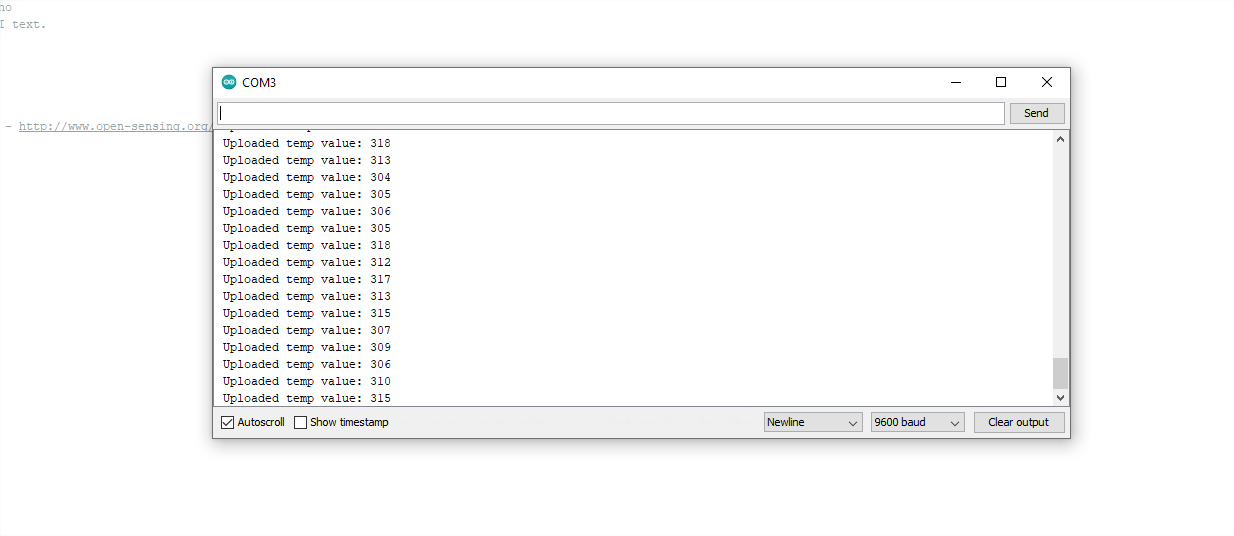
**5.1 Tests Results**

**Testing Wifi connection output**



**5.1.1 – Testing Pushingbox output**





**5.5.2 Testing Blynk**



**6 Security Analysis**

A security breach is a very possible problem, especially when using a device like an Arduino Yun. Our device in particular be more than likely be installed in an outdoor area; this makes it more likely for anyone to access the hardware. Possible security breaches with the Arduino Yun devices include but are not limited to, plugging into the exposed port, theft of device (as it is outside), and resetting all the data that is installed on the device. These are only some of the possible compromises in using an Arduino Yun. Maybe one of the simplest security breaches, theft, is a very real threat with our system as it is outside. This means that anyone could just take it at any time meaning we would lose all of the progress that has been put into that device, we have thought about this and hope that the locked steel mailbox would prevent us from these circumstances. Another security threat, a manual plug-in, is also a very possible threat as the device as it has an exposed USB port for connecting to a PC, if someone were to plug in to our device it would make it possible for them to manipulate code or even put their own code on it and render the device useless for what we intended it to do. This could also lead to the perpetrator having access to an API that would notify them when something went into our box, this could give them full access to our details. Supplemental to this, someone could also reset the Arduino device by holding the button which is on the side. This would erase all data from the Arduino and disconnect it from Wi-Fi also rendering the device useless.

**7 Future improvements**

The improvement proposal for this project will be a challenge in changing the equipment, finding a way to replace our Arduino with smaller and compact components. It will also be suggested a way to protect the equipment against rain, creating a waterproof cover. The use of RFID for tracking letters was discussed at a meeting, but it has not yet been decided whether this will be increased.

Our first idea was focused on making the system work, meeting the requirements stipulated at the beginning of the project. Now our improvement will be to refactor the code so that we don't use the Blynk notification only, we need to implement the sending of email. The sending of email would be a nicer touch to our project as users could keep track of when the sensor was activated thus having a record of times and dates mail was delivered. Another way of notifying a user that w discussed was the use of SMS. This method of notification would be a very ideal way of letting our user know there is mail as the user’s device doesn’t need to have an active internet connection like it would when using blynk or an email notification. This would be very useful if our user was out of the house and had some priority mail coming in. The use of SMS would be harder to implement as we would have to purchase a sim card, and we would also have to keep adding credit to that sim card so overtime the costs would add up, even with this downside we would greatly consider adding it to our project in the future. As of now our main priority would be getting an API that sends emails to our user.

Another future improvement that was discussed in our meetings was the addition of a weather feature, this would help our user know what the weather was like when their package was delivered in case of damaged goods or just simply curiosity. There was two different API’s that were discussed that would provide us with the data that was needed, these were an Accuweather API, and a Yahoo weather API.

On top of software improvements, we also have a plan to improve the hardware of our project, as stated above we plan to use a protective cover on our device. To do this we must use a suitable material such as Perspex or another type of water and windproof plastic. This would ensure that none of the sensitive electronic parts of the Arduino would be exposed to any moisture, second to this, from a security sense, the Perspex covering would also prevent any perpetrators from getting their hands on the device. We would also like to make the device smaller, this is so the device is suitable for more smaller ranges of post-box, giving us more of a customer base and a more appealing product

**8 References**

Andrea, I., Chrysostomou, C., Hadjichristofi, G., 2015. Internet of Things: Security vulnerabilities and challenges, in: 2015 IEEE Symposium on Computers and Communication (ISCC). Presented at the 2015 IEEE Symposium on Computers and Communication (ISCC), pp. 180–187. <https://doi.org/10.1109/ISCC.2015.7405513>

Angelopoulou, A., Mykoniatis, K., Carlson, K., Kim, S.-J., 2014. UCF Smart Mailbox: Reinforcing Communications in the Neighborhoods, in: Stephanidis, C. (Ed.), HCI International 2014 - Posters’ Extended Abstracts, Communications in Computer and Information Science. Springer International Publishing, Cham, pp. 393–398. <https://doi.org/10.1007/978-3-319-07857-1_69>

Khan, T., 2020. A Solar-Powered IoT Connected Physical Mailbox Interfaced with Smart Devices. IoT 1, 128–144. <https://doi.org/10.3390/iot1010008>

Kumar, S., Dubey, P., 2013. SOFTWARE DEVELOPMENT LIFE CYCLE (SDLC) ANALYTICAL COMPARISON AND SURVEY ON TRADITIONAL AND AGILE METHODOLOGY. ABHINAV NATIONAL MONTHLY REFEREED JOURNAL OF RESEARCH IN SCIENCE & TECHNOLOGY 2, 22–30.

Le Vinh, T., Bouzefrane, S., Farinone, J.-M., Attar, A., Kennedy, B.P., 2015. Middleware to Integrate Mobile Devices, Sensors and Cloud Computing. Procedia Computer Science, The 6th International Conference on Ambient Systems, Networks and Technologies (ANT-2015), the 5th International Conference on Sustainable Energy Information Technology (SEIT-2015) 52, 234–243. <https://doi.org/10.1016/j.procs.2015.05.061>

Malhão, S., Dionísio, R., Torres, P., 2019. Industrial IoT Smartbox for the Shop Floor, in: 2019 5th Experiment International Conference (Exp.at’19). Presented at the 2019 5th Experiment International Conference (exp.at’19), pp. 258–259. <https://doi.org/10.1109/EXPAT.2019.8876562>

Muhammad, A., Rehman, N. ur, 2016. Intelligent mailbox with centralized parallel processing, in: 2016 17th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD). Presented at the 2016 17th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), pp. 255–259. <https://doi.org/10.1109/SNPD.2016.7515910>

Parsons, D., Thorn, R., Inkila, M., Mac Callum, K., 2018. Using Trello to Support Agile and Lean Learning with Scrum and Kanban in Teacher Professional Development. pp. 720–724. <https://doi.org/10.1109/TALE.2018.8615399>

Pătru, I., Carabaş, M., Bărbulescu, M., Gheorghe, L., 2016. Smart home IoT system, in: 2016 15th RoEduNet Conference: Networking in Education and Research. Presented at the 2016 15th RoEduNet Conference: Networking in Education and Research, pp. 1–6. <https://doi.org/10.1109/RoEduNet.2016.7753232>

Risteska Stojkoska, B.L., Trivodaliev, K.V., 2017. A review of Internet of Things for smart home: Challenges and solutions. Journal of Cleaner Production 140, 1454–1464. <https://doi.org/10.1016/j.jclepro.2016.10.006>

Spaho, E., Sula, A., Matsuo, K., Barolli, L., Xhafa, F., Iwashige, J., 2013. An Integrated System of Robot, SmartBox and RFID as an Approach for Internet of Things, in: 2013 Eighth International Conference on Broadband and Wireless Computing, Communication and Applications. Presented at the 2013 Eighth International Conference on Broadband and Wireless Computing, Communication and Applications, pp. 589–594. <https://doi.org/10.1109/BWCCA.2013.103>

Srivastava, A., Bhardwaj, S., Saraswat, S., 2017. SCRUM model for agile methodology, in: 2017 International Conference on Computing, Communication and Automation (ICCCA). Presented at the 2017 International Conference on Computing, Communication and Automation (ICCCA), pp. 864–869. <https://doi.org/10.1109/CCAA.2017.8229928>

Turská, S., Madleňáková, L., 2019a. Concept of Smart Postal Mailbox. Transportation Research Procedia 40, 1199–1207. <https://doi.org/10.1016/j.trpro.2019.07.167>