IT-Component lender

Microcontroller & Microprocessor Project

Sonjoy Dey
dept. of CSE
Student ID: 011202074
United International University
sdey202074@bscse.uiu.ac.bd

Al-Momen Reyad
dept. of CSE
Student ID: 011203011
United International University
areyad203011@bscse.uiu.ac.bd

Taimur Rahman
dept. of CSE
Student ID: 011221427
United International University
trahman221427@bscse.uiu.ac.bd

Md. Shabbir Hossain Sourav
dept. of CSE
Student ID: 011203023
United International University
msourav203023@bscse.uiu.ac.bd

Anas Mohammad Ishfaqul Muktadir Osmani
dept. of CSE
Student ID: 011203004
United International University
aosmani203004@bscse.uiu.ac.bd

***Abstract*—A service of hardware support that allows students to borrow different electronics and electric components is common in most engineering institutions. However, they are usually maintained by a group of employee which has some drawbacks. Thus, here we are proposing an automated lending system that require a minimum man power to run. This machine can let student borrow components and return them back when they desire without the need of any human interaction. Our system is built using an Arduino MEGA and a windows laptop. Fundamentally, there are two major operations; borrowing and returning components. For borrowing, there is a keypad and LCD display with which the user can select an item and it's quantity. As for the return process, there is a marked platform for the return item to be kept. Then, using a deep learning model the item is recognized and transferred to it's designated storage. Besides, to ensure security of who can access the system an RFID module is used. This will only grant permission of use to those who have their student ID card.**

***Index Terms*—hardware component, borrow, LCD, keypad, arduino, deep learning model**

I. Introduction

[Problem statement] Usually there is only one room for hardware support services which means it is not mobile. Besides, someone needs to be standby to provide the service which can be tiresome. In order to keep lending records a physical record book is kept and this is done manually by the service provider. Another problem is the confirmation taken by the service provider when an item is returned is simply a tick mark on the actual hard note which means there can be an ambiguity in whether or not an item is returned or not. Also, the process of finding out a certain lending record can be difficult if the item is borrowed a long time ago. Thus, it can also be time consuming. Additionally, students might not be able to convey clearly what they want to the service provider vocally. In our proposed project we tried to find solutions to mitigate these above issues related to lending components.

[Literature review] First of all, since our lending process is done by a machine similar to a vending machine it can be located and moved to various places. On top of that, no one needs to continuously standby as this is an automated system. We also do not need a physical note book to manage the records of the items taken and given as we would maintain a database. For those who face difficulty in making others understand which components they need, they can easily go through the item menu to find their desired items.

II. Project overview

- **Security mechanism:** only students can access the system with their student ID card.
- **Borrow components:** different items can be borrowed from the system by giving the item number and quantity as input.
- **Return components:** items can be returned to the system after they are borrowed.

III. Workflow

To start the system a card is punched on the RFID scanner. The ID is transferred to the Arduino and then to the python, where it is matched with the registered IDs in the database. If the ID is authorized only then the system is allowed to be used. Otherwise, the access to the system is denied. An authorized user will be presented with three option displayed in the LCD. Then,the keypad button 1,2 and 3 can be pressed to borrow, return and exit respectively.

For the borrowing part, first the user is asked to choose an item and then it's quantity. When the system gets both of this inputs it will check if it has enough items in stock or not. If so, it will show a message of how many and which item it is giving in the LCD and give the requested number of items that can be then collected from a collection tray on the lower part

Fig. 1. Project view from user side

of the machine. Otherwise, it shows a quantity-exceed-stock message in the LCD.

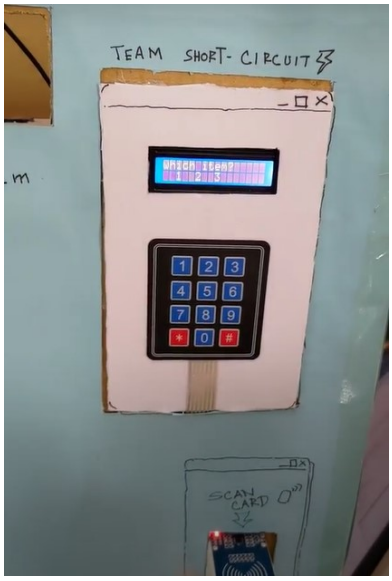

Fig. 2. User Interface of the system

As for the return process, the item to be returned is placed on the top platform's camera view zone. Then, button * is pressed for confirming to start returning the component. The webcam above the platform captures an image of the item. This image is send to a deep learning model which returns a prediction of the captured item. After that, a sequence of servo motor rotation transfers the component from this platform to it's designated slot in the conveyor belt.

Fig. 3. Returning a transistor

IV. Component List

A. Arduino mega

It is used to interface with all the actuators and sensors. Also, to communicate with database server via python in the laptop.

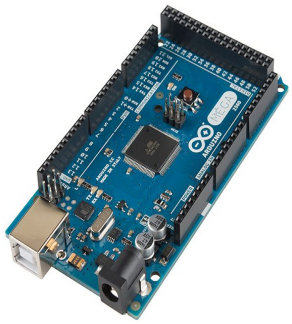

Fig. 4. Arduino MEGA

B. RFID Scanner

RFID is used to detect the student ID. If anyone uses any unregistered ID, it will detect it as invalid.

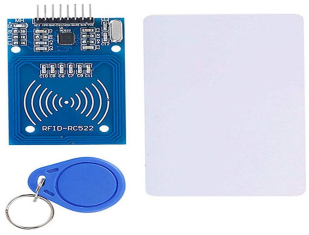

Fig. 5. RFID module

C. LCD Display

In the LCD display, students can see the instruction and will give input accordingly.

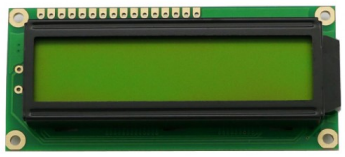

Fig. 6. LCD display

D. Stepper motor

The Stepper motor is used to rotate the conveyor belt.

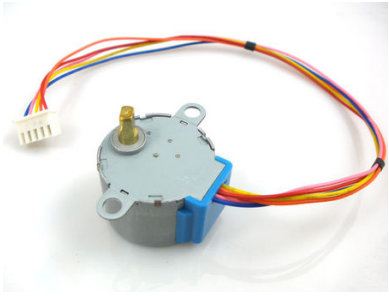

Fig. 7. Stepper motor

E. Servo motor

Servo motors are used in the return part.The 1st one is used to open the gate.The 2nd and 3rd one is used for creating the return path.

Fig. 8. Servo motor

F. Conveyor belt

The conveyor belt stored the product. When students try to get a product, it will rotate forward to give that product and in backward direction during returning time.

G. Web cam

In the return part, the webcam is used to capture and detect the returning component.

Fig. 9. Logitech webcam

H. DC 5 volts adapter

It is used to power the servo and stepper motors.

Fig. 10. 5 volts DC adapter

I. Keypad

It is used to take item, quantity and general instruction input from the user.

Fig. 11. 3x4 keypad

V. Software Environment

A. Arduino IDE

We have used the stable version 1.8.19 of the Arduino IDE to write program for the Arduino MEGA.

B. Jupyter notebook

It is used to run the python code for image classification and communicating with Arduino and database in the windows laptop. Jupyter nootbook version 6 is used in this project.

C. XAMPP MySQL

As for the database server we have used the XAMPP MySQL server version 10.4.24-MariaDB.

VI. Deep Learning Model

In order to detect a component when returning to the system we have trained a CNN(Convolutional Neural Network) model. We have created a custom dataset using the webcam. For each type of item, 101 images were captured. So, in total there was 303 images of which 10% was used to validate the model and 90% for training the model.

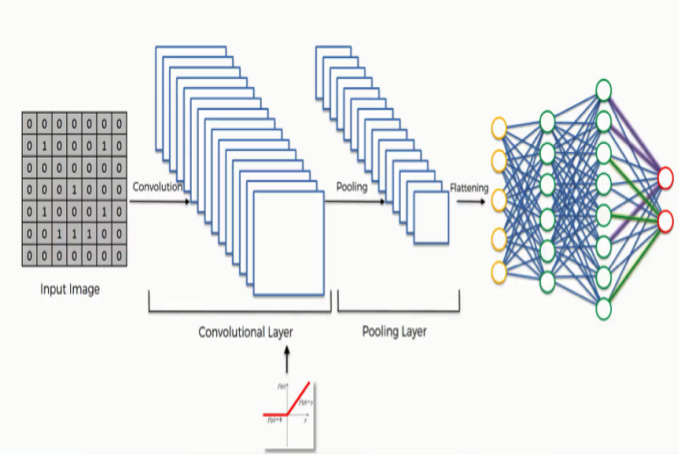

Fig. 12. CNN model

VII. System Flow Chart

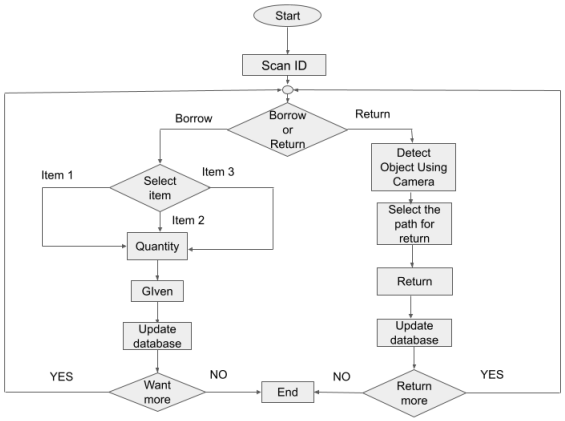

Fig. 13. Simplified flowchart of the system

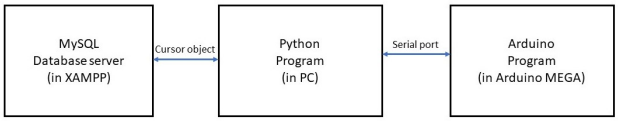

Fig. 14. Network view of the system

VIII. Future Work and Scope

For future work, we are planning to have the system be scalable and more secure. For scalability, it should be possible to include more conveyor belts for more IT components, this would include both making the software accommodate more cases, as well as modular hardware units. A limitation of our current prototype is only working with three types of components, with a maximum of three units per item. A more production grade version of our project should be able to include a lot more.

As for security, we should make it so that if an item is not returned within a defined grace period, then an SMS would be sent to whoever borrowed the component. Our database already stores order information and whether or not an item has been returned or not, so an update to our code would suffice. More optimisations can be made to the code so it can run faster, and to the AI model so images can be recognised faster.

References

[1] https://www.youtube.com/watch?v=CEz1EeDlpbs
[2] https://www.youtube.com/watch?v=8YzTE8jBEU0
[3] https://www.youtube.com/watch?v=h7yO6FPNjho
[4] https://www.youtube.com/watch?v=m4JOdTV2o2E

N.B: Source code for this project can be found in this github repository. Link: https://github.com/AnasIshfaque/IT-Component-lender