*Smart India Hackathon 2022*

# NITK JUMA [GitHub]

## Team


| | |
|---|---|
| Jithedra Puppala | **191cs237** |
| PUTTARAJA | **191it139** |
| Chintamani Masthanaiah | **191cs115** |
| Kruthika. K. Sudhama | **191cs224** |
| VAMSHIKRISHNA M | **191cs261** |
| V N Karthik | **191it154** |

# INDEX



# INTRODUCTION

## Motivation

NITK students who will leave the campus in a month or a week. They can't carry out all their belongings back home. Instead, they want to sell them. Let's say books. The students who are leaving the campus may not require some books and would like to give them to their juniors or freshers at a reasonable price. There are many things other than books. Accessories like Keyboard, Extension Boxes, LAN cables, Printers; Room essentials like mattresses, Buckets, Hangers; Stationery (Drafters); and umbrellas.

It's not easy for them to find a suitable person who wants to buy or sell those things unless they know it through some media. Also, It is not possible for the student to ask every other student if they are willing to buy or sell the things. Also, the students who leave the college in a couple of days want to find the buyers at the earliest possible.

## Proposed Solution

To address this problem, our team came up with the best solution for the intra-college students. A portal is built for the students, where they can request and post things which they would like to buy and sell respectively. Once we find any buyer-seller pair, they can contact each other and get the things transferred.

In detail, the student who wants to buy can request the items from the specific list of things the portal provides. Similarly, the student can post the items one would like to sell. The student has to wait until he finds the person to buy or sell. If there are no other buyers for the item, the student can give those items to the Co-operative Society at a fixed price. Rest is handled by the Co-Operative Society. And if the buyer didn't satisfy with the item or didn't find the item for a long time and bought that new item, the buyer can withdraw his request anytime. Likewise, the seller can also give it to the Co-Operative society at any time.

## Related Work

There are few existing solutions to this problem. But they are not effective. Some online platforms like OLX allow users to post the items. But, it has very little reach for the items like buckets, LAN cables, etc., on such platforms, and this problem can easily be solved if the buyers and sellers are of similar interest like within neighborhood, school, etc.…
In such platforms, usually, we need to wait for the buyer to approach us. This keeps the seller waiting for hours, days, even months. In the problem statement, the student wants to sell them quickly. For which we added a feature for the buyers to request the item forehand.
Even when a buyer is found, and he is at a far distance, it might not be worthy enough to travel to deliver the item. To prevent this, we restrict this solution only to the students inside the college.

# REQUIREMENT ANALYSIS AND SYSTEM SPECIFICATIONS

## Data Requirements

The set of data that is involved in any project is defined using data requirements. In this project, the main data required are the login information to register the application item types to start this application.

## Functional requirements

- Users should be able to sign up as well as sign in with their email accounts and update the account information.

- Users should be able to select the item type on the platform and browse for the best item to buy. Users should be able to publish and post their requests.
- Users should be able to publish the item for sale, providing images and necessary information about the product. Sometimes, the user may sell to the Co-operative Society store for a lesser price.
- Users should be able to update the status of the items once they are sold out or withdrawn.
-

## Non Functional requirements

- Scalability and reliability are taken care of by Django, as it is simple, highly scalable, and flexible.
- Security of the sensitive data transmission of users is taken care of by Django.
- The database is designed with high data integrity.
- The performance of the application will be good with the optimized queries in SQLite.
- The maintainability and manageability of the application will be taken care of by the student administrators because it can be deployed to the college database.

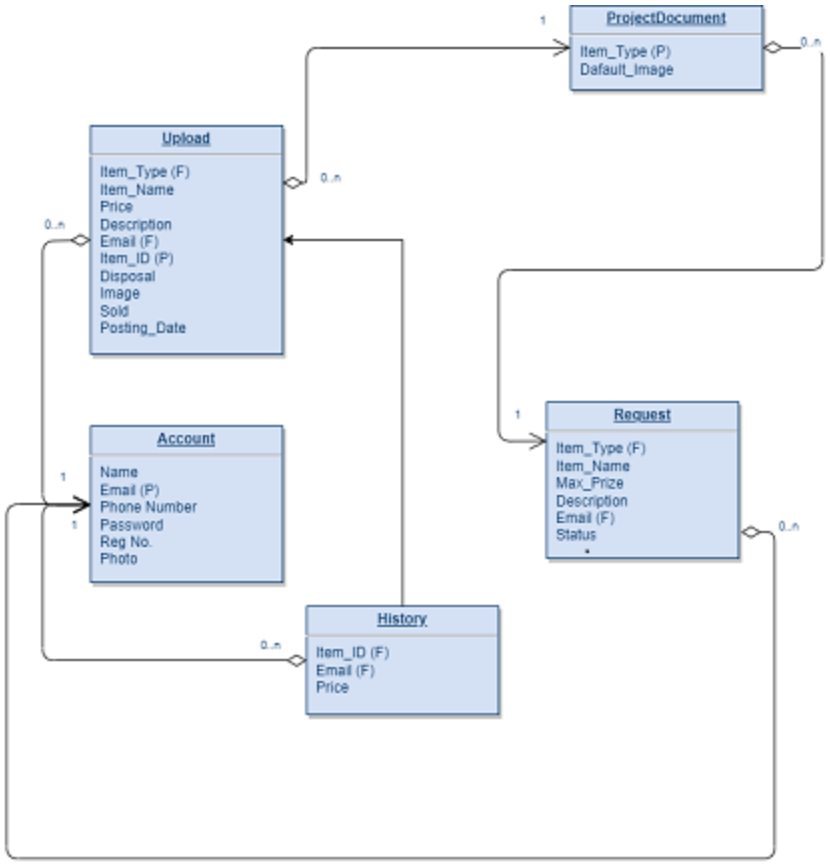
## System Requirements

The application can be installed into a device, system, or any machine in such a way that it should have basic requirements like supporting software and hardware of the device, accessing in-built software, internet permissions, and potential security issues such as viruses or any malware detection**.**

# Database design

The portal uses an SQLite database, which consists of 5 tables. They are named as

1. Account table
2. Request table
3. Item_types table
4. Upload table
5. History table




# Implementation

It's a web-based application using HTML, CSS, and Django. Django uses SQLite as its default database. The interface is made simple for easy navigation. Most of the data taken from the user is clearly illustrated in examples.

# Future Work

This solution can be extended to any person on the campus/college, like for Faculties, Employees. And instead of fixing the prices, we can also have bidding systems for the betterment of users of this application.

It can also extend to other nearby colleges. Things like books can also be sold to the bookstores around the campus.

Chat Box can be implemented in the later version to maintain personal information in secure.

Recommendation system can also be implemented. Notifications system for any new buyers or sellers can be notified.

# Results and Conclusion

From our analysis of the existing solution and its drawbacks, the best solution is put forward and implemented in this portal. The only limitation of this solution is that it is designed for the same college students. It also require user to wait till the users find the right person. But the wait can be reduced if there is a right person.