

# Projects I've Done So Far

## Year: 2019

### 1. Restaurant POS

- A Point of Sale system for restaurants including an **online menu** for ordering, live updates to the kitchen and waiters using **web sockets**, inventory management, reception view and an online integrated **payment system**.

[https://github.com/lurayy/restaurant\\_module](https://github.com/lurayy/restaurant_module)

tag: Django, Python, html/css, js

### 2. Auto-watch server for watching league videos

- An automation script using python and pygubu for GUI. The script acts like a bot that watches the VOD of League of Legends games and collects rewards. It uses requests lib to create a session and uses a proxy to login into the website using multiple accounts at the same time.

tag: automation

### 3. Abettor [College project for Software Engineering, 6th semester, BSc. CSIT]

- College information management system
- Feature Includes
  - a. College message board
  - b. Staff and Student management
  - c. Library management
  - d. Exam Result generation
  - e. Online bill payment capabilities using Khalti
  - f. Login using QR.

<https://github.com/lurayy/Abettor>

tag: Django, html/css/scss, js

### 4. AI Playing Flappy Bird [Seventh Semester Project]

- An AI model that can learn to play games using its own experience.
- achieved using Q learning a type of reinforcement learning algorithm
- Uses a clone of the flappy bird as an environment to train and demonstrate the agent's capability.

Paper:

[https://github.com/lurayy/AI\\_playing\\_flappy\\_bird/blob/master/Documentation.pdf](https://github.com/lurayy/AI_playing_flappy_bird/blob/master/Documentation.pdf)

[https://github.com/lurayy/AI\\_playing\\_flappy\\_bird/](https://github.com/lurayy/AI_playing_flappy_bird/)

tag: automation, python

## **Year: 2020**

### 1. Mandala Inventory:

- An enterprise resource planning web application that includes inventory management, point of sales, staff management, ability to handle payment, gift cards as well as simple accounting and more.

[https://github.com/lurayy/simple\\_inventory](https://github.com/lurayy/simple_inventory) [Deprecitated Version]

tag: Django, React-js

### 2. Ecwid App For Discount And Offer Management:

- A discount management web application that uses webhooks to discount information to an integrated site.
- discount can be based on item, bill amount, coupon, general discount or location.
- Currently used in: <https://produitsducap.com/>

tag: Django, react-js

### 3. Reading App: [Incomplete]

- A mobile app for reading, buying and publishing ebooks.
- the pdfs are broken into pieces and transferred after being encrypted.
- decrypts the pieces in saved RAM rather than on disk preventing unwanted access.

tag: Django, mobile app

## **Year: 2021**

### 1. Image-Compression:

- automation tool to compress images using multiple algorithms.
- effectively compressed files by 80% without any obvious loss in quality.

<https://github.com/lurayy/Image-Compression>

tag: python, crunch, pillow, pyguetzli

### 2. AutoDeploy:

- An in-house tool created to automate the deployment process of new web applications to an EC2 server.
  - The main idea is to automatically clone projects from GitHub and create the required Nginx file, uwsgi file and service file to run the app
  - This only handles the initial part, for CI/CD I used GitHub action to trigger a git pull script in the server.
- tag: Django, CI/CD, automation*

### 3. Booksmandala Repository:

- acts as a collection and point of data for multiple ERPs that need to connect and provide data to <https://booksmandala.com/>
  - processes and handles invoices and relays them to the proper ERP.
- tag: system design, Django*

### 4. WebScrapper using selenium:

- scrapper using selenium instead of using requests and beautiful soup.
- [https://github.com/lurayy/web\\_scrapper\\_aava](https://github.com/lurayy/web_scrapper_aava)
- tag: python, scrapper*

### 5. Easyhome: [Incomplete]

- A home automation solution that merges multiple third-party smart home applications with an architectural ability to add more services in the future.
  - uses a custom light version of Django stripped down of middleware as the backend framework
  - uses micro-service architecture and **gRPC** to communicate with other services
- Core Component in the app architecture:

#### a. The House:

- Dashboard to check and deploy add-on apps.
- This is not an add-on app. This is more like a developer tool to speed up the deployment process and manage the deployed apps on the docker swarm.
- This is the app that controls the docker and automates the add-on process for other apps.
- Handles Nginx configuration for add-ons apps (running in docker containers).

#### b. Maid Module:

- Handles login for all the add-on apps.
- Handles all the user-related data and user preferences.

- This acts as the main hub to which all the add-ons will take to in the future.
- c. Garage Module:
- connects to third-party garage door API
  - handles status garage door functionality with monitoring.
- tag: Django, micro-service architecture, docker, docker swarm

#### 6. Bizmandala v1:

- provides ERP v1 in the SaaS model.
  - Uses docker containers to achieve complete isolation of data for every user's instance of ERP
  - deploys kills and removes ERP container based on the user's subscription
  - Has payment subscription model
- <https://bizmandala.com/>
- tag: Django, docker, automation, DevOps

## **Year: 2022**

1. Degenhouse server:
  - A simple REST server for persistent data storage and off-chain calculation.

tag: blockchain, Django, REST
2. ENS-Domain-Check:
  - Uses selenium to check if an ENS domain is available or not.

tag: automation, selenium, blockchain
3. Vedalaya:
  - A company portfolio website with e-commerce and custom content management ability.

<https://www.vedalayagroup.com/>

tag: Django, Reactjs, e-commerce
4. Book verification Script:
  - Automation solution to verify book details and images using google search.
  - Is used to update any mistakes in book details and images and is used to process 60,000 books from <https://booksmandala.com/>

tag: automation, selenium

5. IRD sync service:

- A simple relay application coded using express js.
- is used to relay billing information from multiple ERPs to Inland Revenue Department servers

tag: *node-js, express-js*

6. book-image-selector using selenium

- automation solution for updating book images on <https://booksmandala.com/>
- Clicked image is used to replace the image of the current book

tag: *python, selenium, automation*

7. Gweistation:

- Ethereum Gas Tracker.- A utility app.
- Gas price free of Ethereum
- Has Mainnet Block Explorer
- Has an account spy feature (push notification is sent every time the saved address makes a transfer)
- Also has a discord bot for gas price updates and push notifications

<https://gweistation.io/>

tag: *blockchain, Django, Ethereum*

8. Lineage-name-checker

- A automation script to check if a name is available or not at

<https://linageenameregistrar.com/>

tag: *python, automation*

9. Mandala ERP v2

- Migrated from the Request-Response model to Event Driven Architecture using Rabbitmq as message broker and Redis as celery backend.
- Added custom celery consumers to consume events from Rabbitmq queues.
- Converted into a stateless application
- Old isolation was achieved between different ERPs instances. The old approach was to use docker and deploy a new container if a new ERP is needed. The whole process was automated and controlled by Bizmandala and was a subscription model.
- Now isolation is achieved within the database using isolated schema. A single database keeps every ERP's data. Each ERP's data is in a separate schema within the single database. Schema is selected during the request process instead of when the app starts. The sub-domain is mapped to a particular schema.

*tag: system design, Django, multi-tenancy*

10. Booksmandala Repository v2:

- Migrated from the Request-Response model to Event Driven Architecture using Rabbitmq as message broker and Redis as celery backend.

*tag: system design, Django*

11. Bizmandala v2: [work in progress]

- Migrated from the Request-Response model to Event Driven Architecture using Rabbitmq as message broker and Redis as celery backend.
- Converted from using docker to using events to trigger new ERP instance creation.

<https://bizmandala.com/>

*tag: system design, Django*

12. RVM cold:

- A logistic solution that handles the collection, storage and transport of items from multiple sources to multiple destinations giving access and real-time information to multiple types of users with multiple types of permissions.

<https://rvmcold.com/>

*tag: Django, react-js*