Project Report

Farm n Food

Jenish Patel (B00897765)

Lavita Pereira (B00879443)

Meet Patel (B00899516)

Rutvik Patel (B00897762)

Vinay Patil (B00911203)

Dalhousie University

Subject

CSCI5308 - Adv Topics in Software Develop (Sec 1) - 2022 Winter

Professor

Dr. Tushar Sharma

TA: Narendran Krishnakumar

Farm n Food

Digital solution for agriculture industry

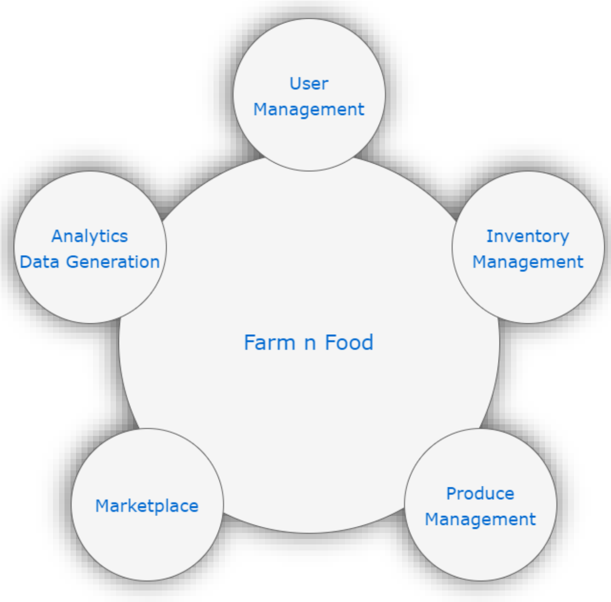
Objective :

An application for farmers and breeders that helps them manage and record their inventory, enhance the efficiency, productivity, and quality of their products, and bridge the gap between producer (farmer) and consumer (local people, restaurant owners)

Goals:

- Farmer can visualize feedback on the allocation of their resources and investigating
- Provide information to the local citizen about local food availability and freshness.
- Connect local farmers with local citizen
- A database records and tracks farming and breeding activities.

Business Functions:



Dependencies that are important to run this project are as follows:

1. **Django**
   Django is a Python-based free and open-source web framework that follows the model–template–views architectural pattern and we have used this framework to create our Farmfood project[1].

2. **Flake8**
   This is a pycodestyle(Gitlab Recommened Library) based python library used for checking our code base against coding style, programming errors, and to check cyclomatic complexity. The important thing about this library is that it runs with wrapping of pyflakes, pycodestyle and Ned Batchelder's McCabe script[2].

3. **Mysqlclient**
   We used the following tool to establish the connection with the server. This allowed us to transmit commands and deliver queries to the server, as well as administer our server-based database[8].

4. **PyJWT**
   The authentication functions in our projects were managed by the JSON web Tokens that helped us to encode and decode the strings that we used in the Email verification and data management process[6].

5. **WhiteNoise**
   WhiteNoise allowed our web app to serve its own static files with just a few lines of configuration, making it a self-contained unit that can be deployed anywhere without relying on any other external service. Even helped us to provide static configuration on the heroku server without any difficulties. We have put it to our work in our production environment[4].

6. **Gunicorn**
   Gunicorns handles many instances of our web application, ensuring that they are healthy and restarting them as needed, as well as distributing incoming requests among them and communicating with the web server. Furthermore, Gunicorn is a lightning quick runner when it comes to it[3].

7. **Pillow**
   Python Pillow module is built on top of PIL (Python Image Library). Pillow is a Python Imaging Library (PIL) that adds image opening, manipulation, and saving functionality to Python[7].

Build/Deployment instructions

1) Setting up Heroku:

Create an account on Heroku.

Create a new app by clicking on New > Create new application.

Get the unique API Key under account settings.

Configure database variables in Heroku by selecting application name → settings →Config Vars

2) Configure MySQL database by setting variables NAME, USER, PASSWORD, and HOST. These can be set in the project settings at Settings → CI/CD → Variables.

3) Set Heroku variables API_KEY, HOST_NAME, and HOST in CI/CD variables.

4) Enable GitLab runners.

5) Include all the dependencies in a requirements.txt file

6) Create a Procfile. Heroku web applications require a Procfile to tell Heroku to run a Gunicorn server.

7) Create a **.gitlab-ci.yml** file. The contents of the file are as follows:

The following commands should be run before each job's script commands

apt -y update: The apt command is responsible for installation, removal, and updating of software in the system.

apt -y install apt-utils:

apt -y install net-tools python3.8 python3-pip mysql-client libmysqlclient-dev

apt -y upgrade: upgrades packages to their latest versions and installs new packages if they are required as dependencies

pip3 install -r build_req.txt: Install packages from build_req.txt file

Build Stage:

python3 manage.py makemigrations: Generates SQL commands for preinstalled apps and the farmfoodapp.

python3 manage.py makemigrations farmfoodapp: Create migration scripts under migrations subdirectory of your farmfoodapp

python3 manage.py migrate: Executes SQL commands in the database file and creates tables.

python3 manage.py check: (Checks for Configuration and Setup Erros before running the Server)

Test Stage:

We are using inbuilt Django testing module The command used is:

```
python3 manage.py test
```

Code Quality stage:

We are using Flake8, a Python library, for checking the code base against coding style (PEP8), programming errors (like "library imported but unused" and "Undefined name") and to check cyclomatic complexity.

Deployment Stage:

We have deployed the app on Heroku

The image used is ruby:2.6.

Dpl is a deploy tool made for continuous deployment, which allows one to test all commands from the local terminal. The command used is:

```
gem install dpl
```

wget is a command line tool used to download files using command line interface)

```
wget -qO- https://cli-assets.heroku.com/install-ubuntu.sh | sh
```

To deploy farmfoodapp to Heroku, we need to specify heroku as provider, specify api_key and app. This is done using the below command:

```
dpl --provider=heroku --app=$HEROKU_APPNAME --api-key=$HEROKU_APIKEY
```

Run makemigrations and migrate commands to reflect the models to MySQL database.

```
heroku run --app $HEROKU_APPNAME python manage.py makemigrations farmfoodapp
```

```
heroku run --app $HEROKU_APPNAME python manage.py migrate
```

8) Once the project and **.gitlab-ci.yml** is created, push project to GitLab.

Usage scenario

1) The users must register themselves on the register page giving their details that are going to be stored in the database.

2) Once the user registers, an activation email is sent to the user and the user can only log in only after they have activated their account by clicking on the link. If the user tries to login without activating their account, a message appears asking the user to first activate their account.

3) The user can see all the listings available on the farm food application in various categories like fruits, vegetables, livestock, dairy, and seasonal foods.

4) The user can also register as a farmer, and post products that they want to sell, in the application. The user also has an option to edit or delete the products posted.

5) The farmers can manage their inventory by adding items in various categories like seeds, fertilizers, livestock, and pesticides. The farmer also has an option to edit or delete the inventory items. This is visible only if the user is registered as a farmer.

6) Farmers can manage their costs by using a cost manager to keep track of all their expenses. The categories available are raw materials, equipment, vehicle, and labor charges. The farmer also has an option to edit or delete the expenses.

7) The farmers can also see analytic data supplied by the application which will assist them to comprehend the present market and plan their future move. Farmer can see the Global Product Market, Current User Product, Users to Farmers Ratio, Trend Analysis, and Cost Analysis.

8) The Farmers can publish, edit, and delete their blogs.

Different API made for communication within the web system:

API NAME: Register API
ENDPOINT: https://farm-food13.herokuapp.com/register-api/
AUTH: SESSION AUTH MANDATORY
METHOD: POST, GET
PARAMS: First Name, Last Name, Date of Birth, Email, Phone No, Password

API NAME: Login API
ENDPOINT: https://farm-food13.herokuapp.com/login-api/
AUTH: SESSION AUTH MANDATORY
METHOD: POST, GET
PARAMS: Email, Password

API NAME: Forget Password API
ENDPOINT: https://farm-food13.herokuapp.com/forget-password/
AUTH: SESSION AUTH MANDATORY
METHOD: POST, GET
PARAMS: Email

API NAME: Reset Password API
ENDPOINT: https://farm-food13.herokuapp.com/forget-password/
AUTH: SESSION AUTH MANDATORY
METHOD: POST, GET
PARAMS: Email

API NAME: Onboard Farmer API
ENDPOINT: https://farm-food13.herokuapp.com/onboard-vendor/
AUTH: SESSION AUTH MANDATORY
METHOD: POST, GET
PARAMS: Company Name, Location, Market Name, Address

API NAME: Add Product API
ENDPOINT: https://farm-food13.herokuapp.com/add-product/
AUTH: SESSION AUTH MANDATORY
METHOD: POST, GET
PARAMS: Product Name, Category, Description, Price, Product Image

API NAME: Edit Product API
ENDPOINT: https://farm-food13.herokuapp.com/edit/[Product-ID]
AUTH: SESSION AUTH MANDATORY
METHOD: POST, GET
PARAMS: Product Name, Category, Description, Price, Product Image

API NAME: View Product API (On clicking a product)
ENDPOINT: https://farm-food13.herokuapp.com/product/[Product-ID]
AUTH: SESSION AUTH MANDATORY
METHOD: GET

API NAME: Dashboard API
ENDPOINT: https://farm-food13.herokuapp.com/dashboard/
AUTH: SESSION AUTH MANDATORY
METHOD: GET

API NAME: Add Inventory API
ENDPOINT: https://farm-food13.herokuapp.com/add-inventory/
AUTH: SESSION AUTH MANDATORY
METHOD: POST, GET
PARAMS: Item Name, Category, Description, Quantity, Unit

API NAME: Edit Inventory API
ENDPOINT: https://farm-food13.herokuapp.com/edit-inventory/[Product-ID]
AUTH: SESSION AUTH MANDATORY
METHOD: POST, GET
PARAMS: Item Name, Category, Description, Quantity, Unit

API NAME: View Inventory API
ENDPOINT: https://farm-food13.herokuapp.com/view-inventory/
AUTH: SESSION AUTH MANDATORY
METHOD: POST, GET
PARAMS: Item Name, Category, Description, Quantity, Unit (for POST)

API NAME: Category API
ENDPOINT: https://farm-food13.herokuapp.com/category/vegetables
AUTH: SESSION AUTH MANDATORY
METHOD: GET

API NAME: Search API
ENDPOINT: https://farm-food13.herokuapp.com/search/apple[Search String]
AUTH: SESSION AUTH MANDATORY
METHOD: POST
PARAMS: Search String

API NAME: Charts API
ENDPOINT: https://farm-food13.herokuapp.com/analytics/
AUTH: SESSION AUTH MANDATORY
METHOD: GET

API NAME: Publish Blog API
ENDPOINT: https://farm-food13.herokuapp.com/publish-blog/
AUTH: SESSION AUTH MANDATORY
METHOD: POST, GET
PARAMS: Blog Title, Content (For POST)

API NAME: Edit Blogs API
ENDPOINT: https://farm-food13.herokuapp.com/edit-blog/[Product-ID]
AUTH: SESSION AUTH MANDATORY
METHOD: POST, GET
PARAMS: Blog Title, Content

API NAME: View Blogs API
ENDPOINT: https://farm-food13.herokuapp.com/view-blogs/
AUTH: SESSION AUTH MANDATORY
METHOD: GET

API NAME: Cost Manager API
ENDPOINT: https://farm-food13.herokuapp.com/cost-manager/
AUTH: SESSION AUTH MANDATORY
METHOD: POST, GET
PARAMS: Category, Cost incurred to, Expense

API NAME: Edit Expenses API
ENDPOINT: https://farm-food13.herokuapp.com/edit-cost/[Product-ID]
AUTH: SESSION AUTH MANDATORY
METHOD: POST, GET
PARAMS: Category, Cost incurred to, Expense

API NAME: View Expenses API
ENDPOINT: https://farm-food13.herokuapp.com/view-expenses/
AUTH: SESSION AUTH MANDATORY
METHOD: GET

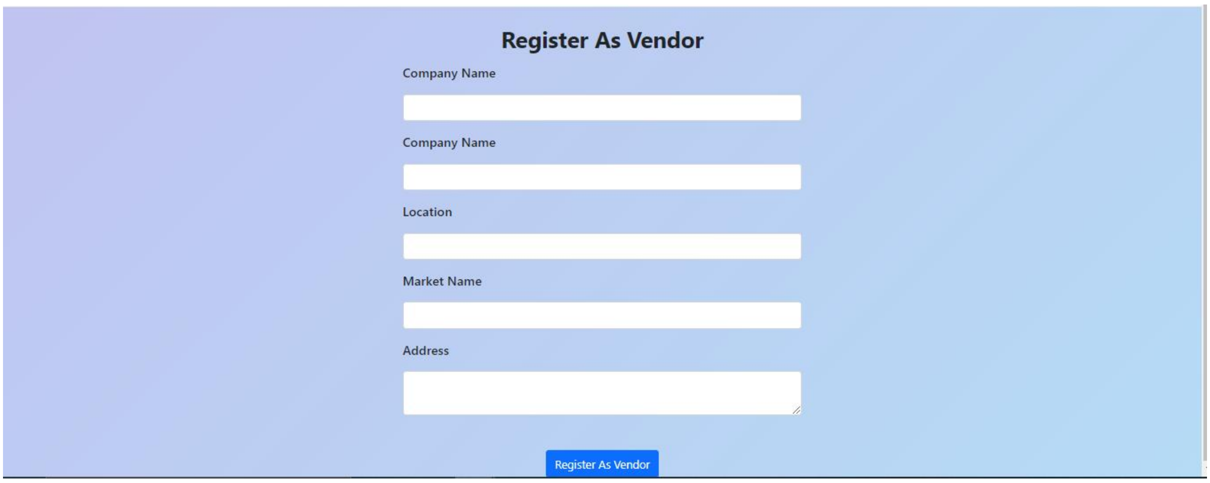
API NAME: View Blog List API
ENDPOINT: https://farm-food13.herokuapp.com/blogs/
AUTH: SESSION AUTH MANDATORY
METHOD: GET

API NAME: View Blog content API
ENDPOINT: https://farm-food13.herokuapp.com/blog/[Product-ID]
AUTH: SESSION AUTH MANDATORY
METHOD: GET

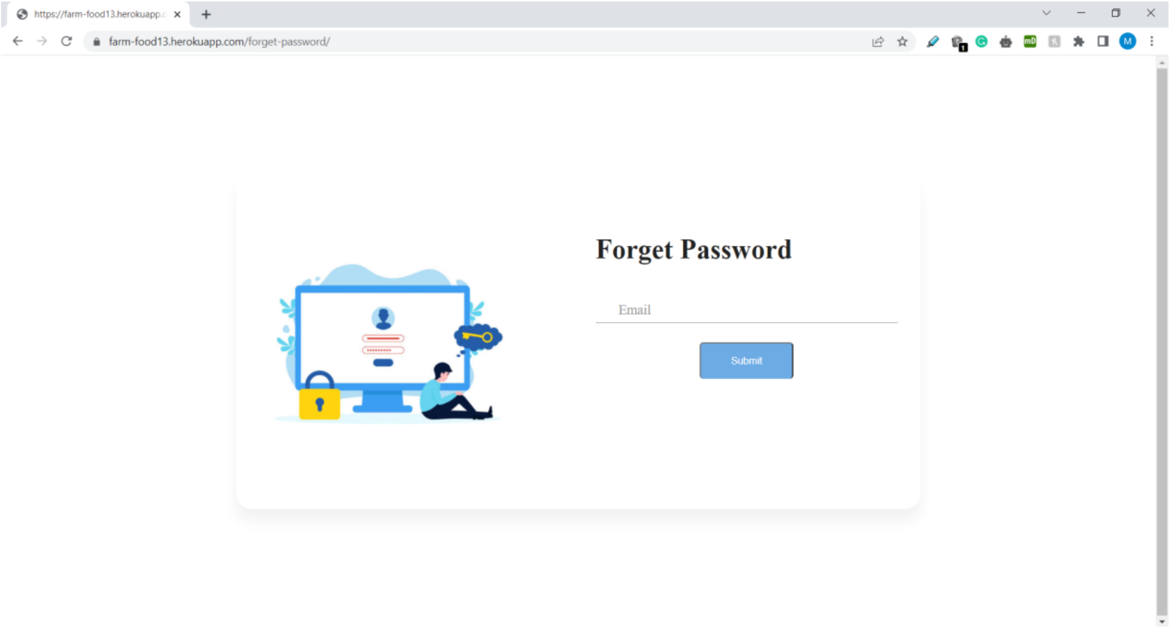
User Interface:

Onboarding

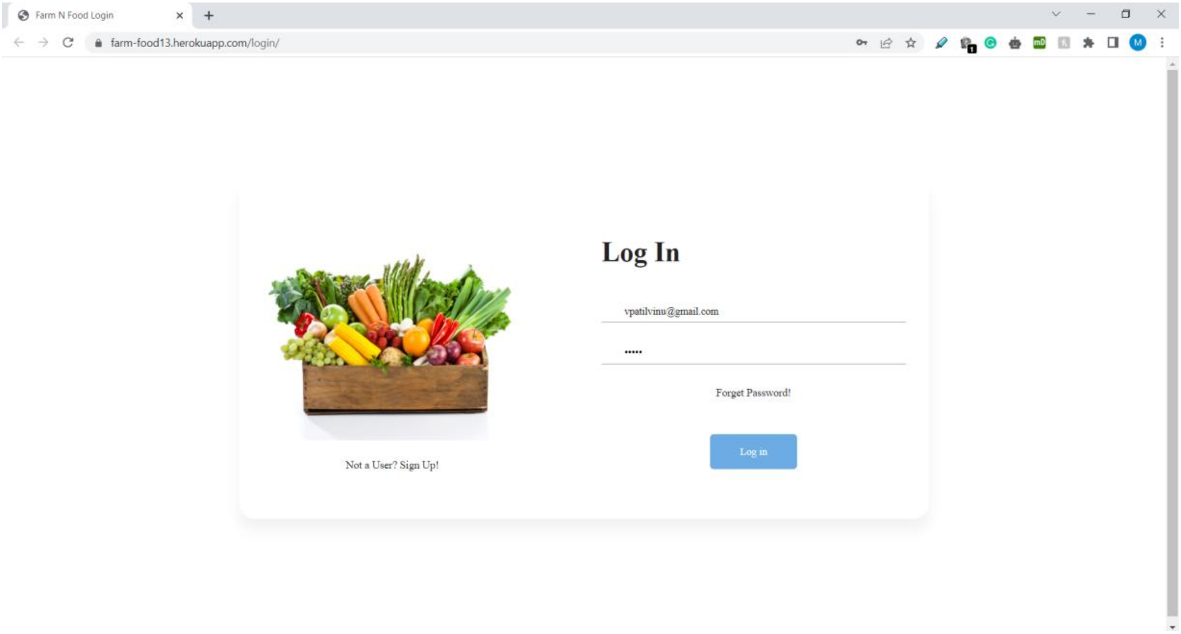
1. Onboard_Farmer.html



2. User_ForgetPassword.html


https://farm-food13.herokuapp.c
farm-food13.herokuapp.com/forget-password/
Forget Password
Email
Submit

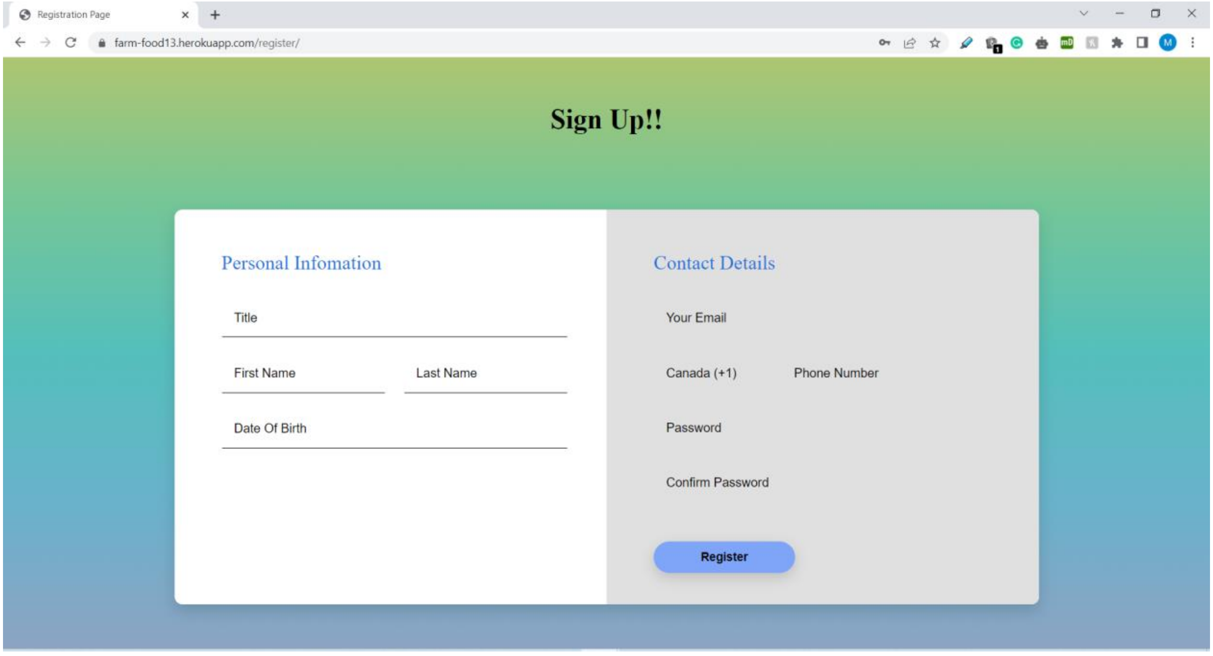
3. User_login.html



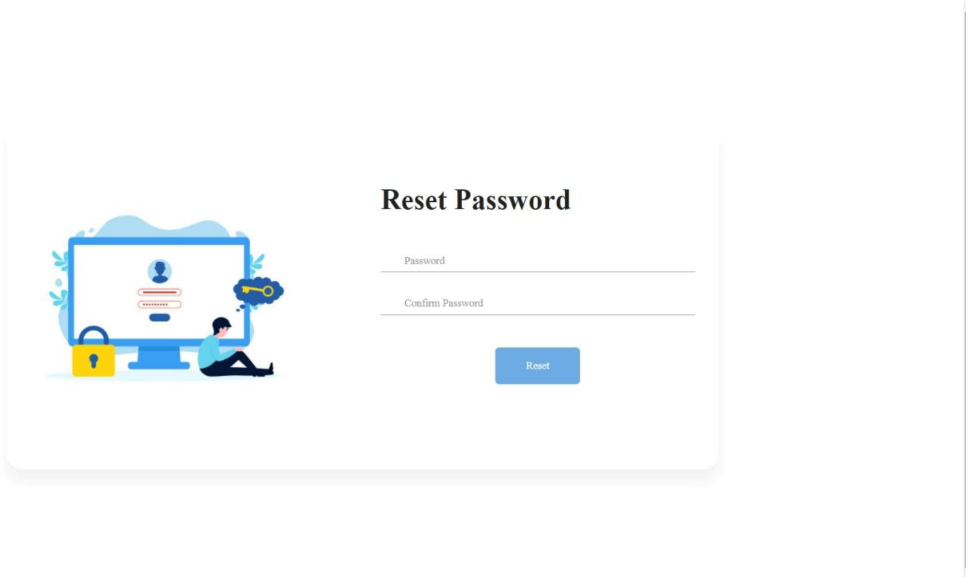
vpatilvinu@gmail.com

•••••

4. User_Registration.html



5. User_ResetPassword,html



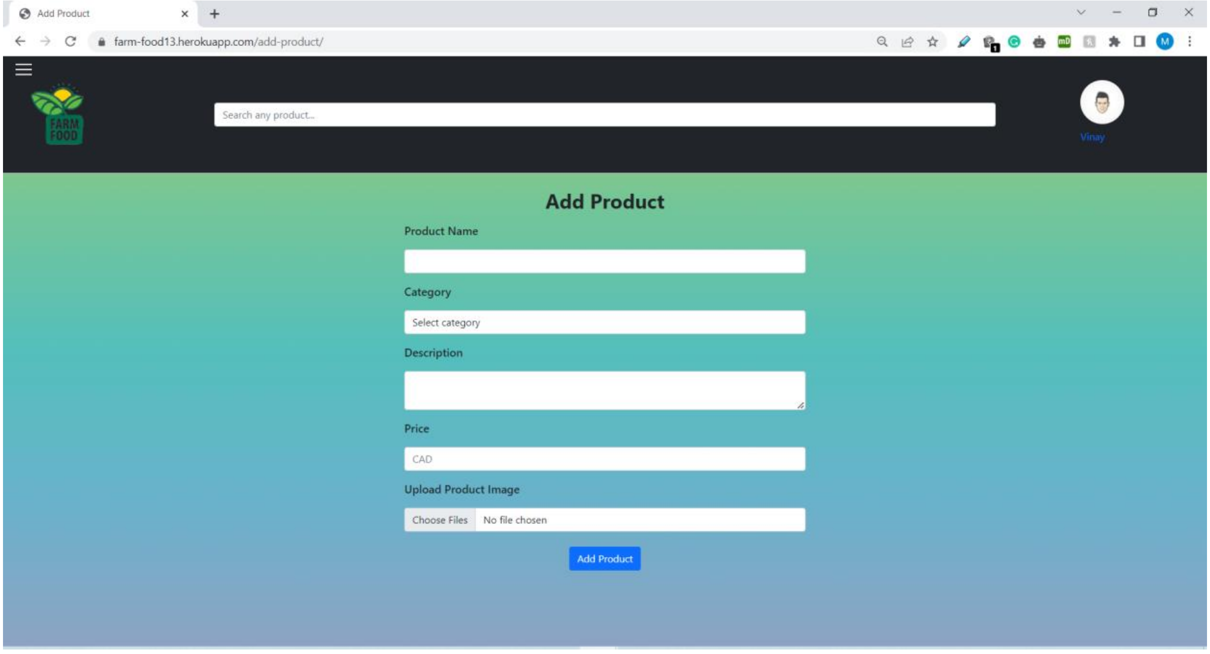
Reset Password

Password

Confirm Password

Products

1. Add_Products.html



Add Product

Product Name

Category

Select category

Description

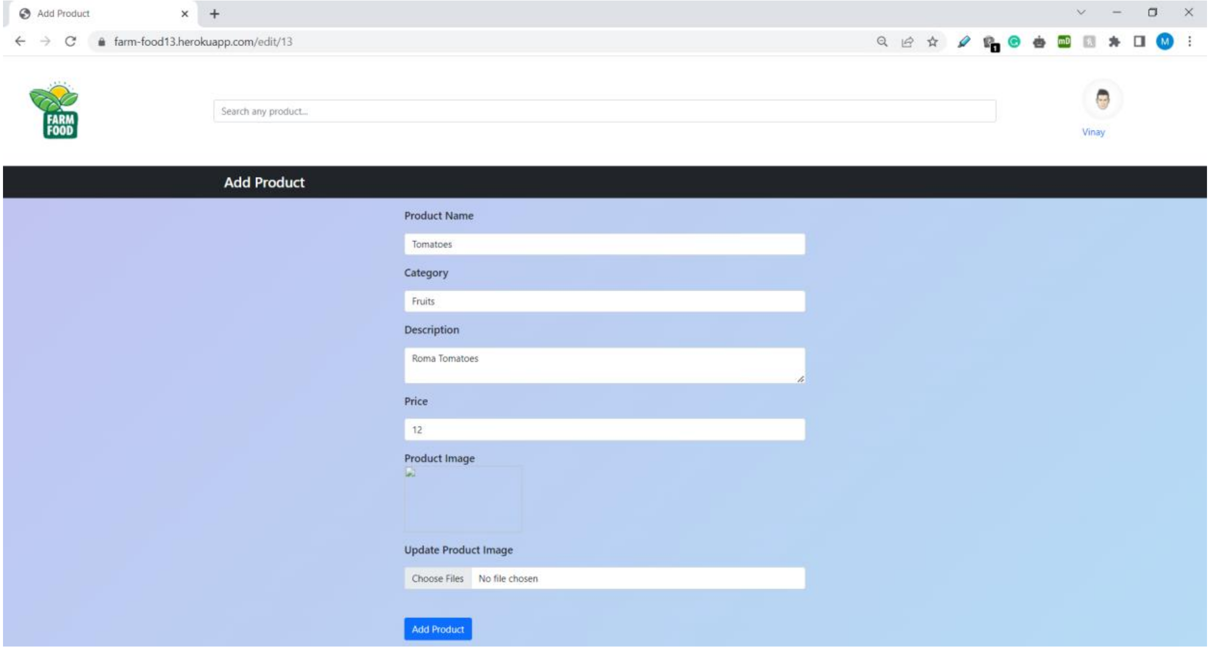
Price

CAD

Upload Product Image

Choose Files No file chosen

2. Edit_Products.html



Add Product

Product Name

Tomatoes

Category

Fruits

Description

Roma Tomatoes

Price

12

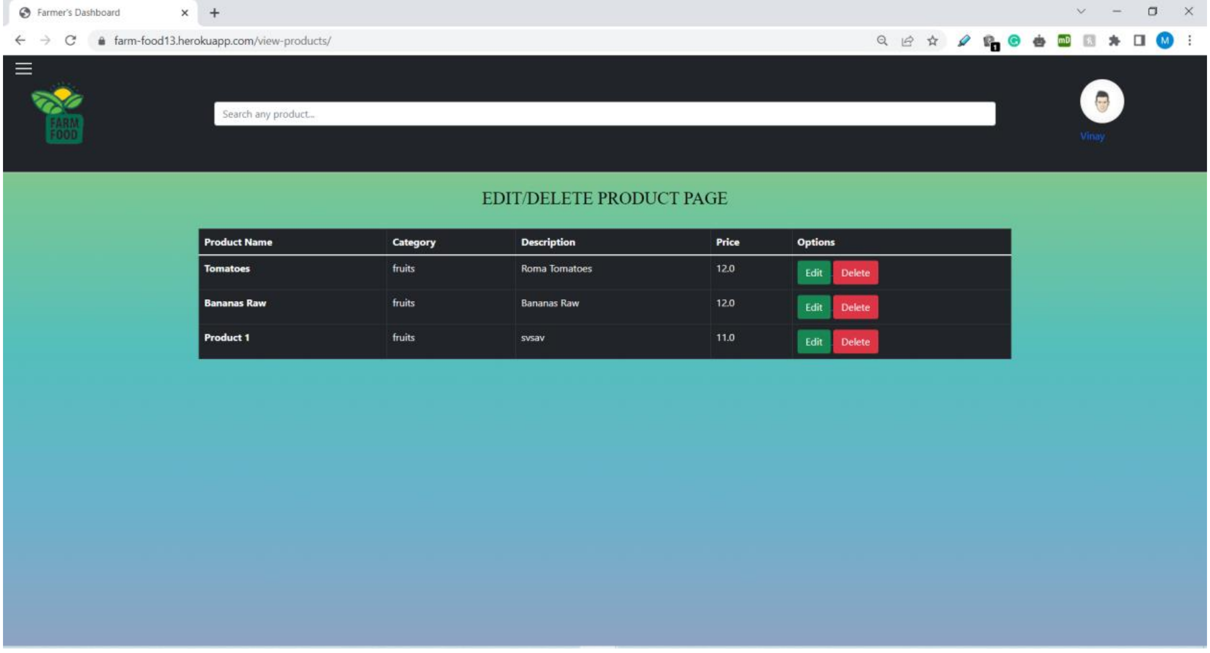
Product Image

Update Product Image

Choose Files No file chosen

Add Product

3. View_Product.html

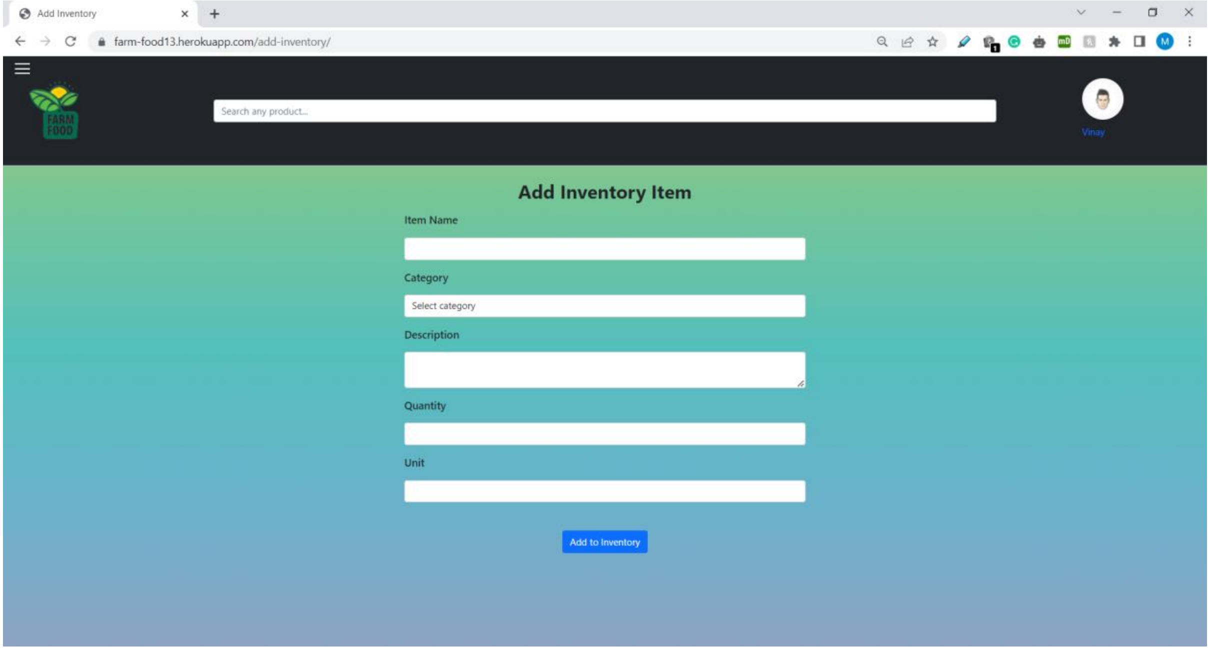


EDIT/DELETE PRODUCT PAGE

Product Name	Category	Description	Price	Options
Tomatoes	fruits	Roma Tomatoes	12.0	Edit Delete
Bananas Raw	fruits	Bananas Raw	12.0	Edit Delete
Product 1	fruits	svsav	11.0	Edit Delete

Inventory

1. Add_Inventory.html



Add Inventory Item

Item Name

Category

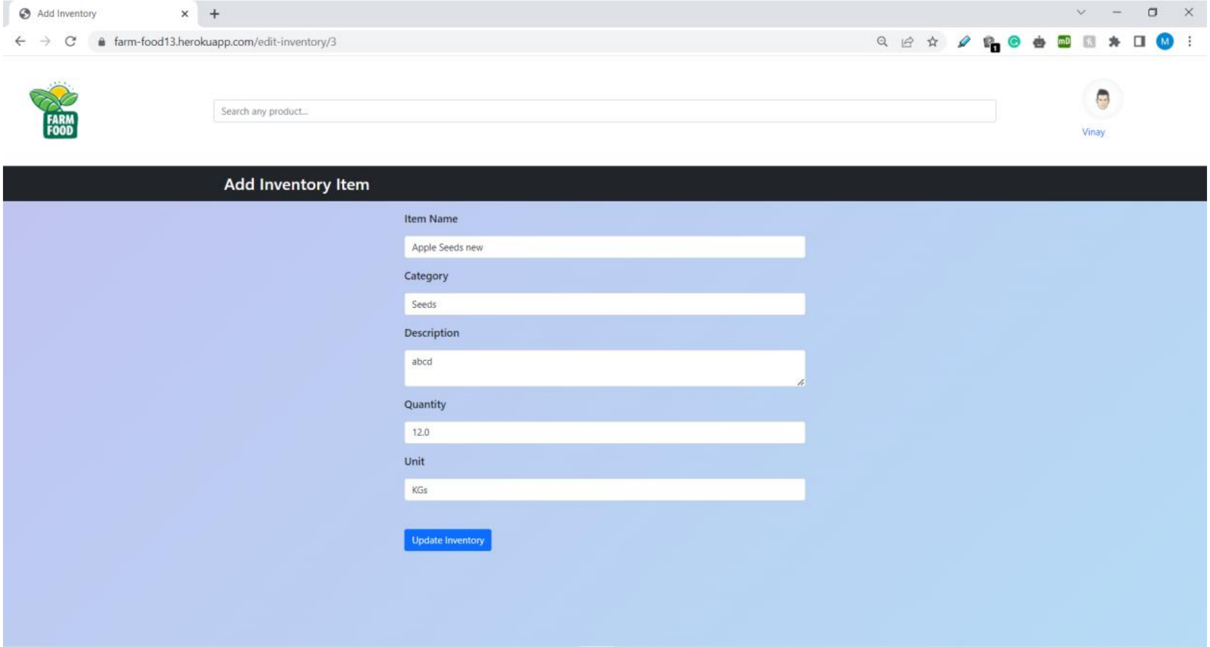
Select category

Description

Quantity

Unit

2. Edit_Inventory.html



Add Inventory Item

Item Name: Apple Seeds new

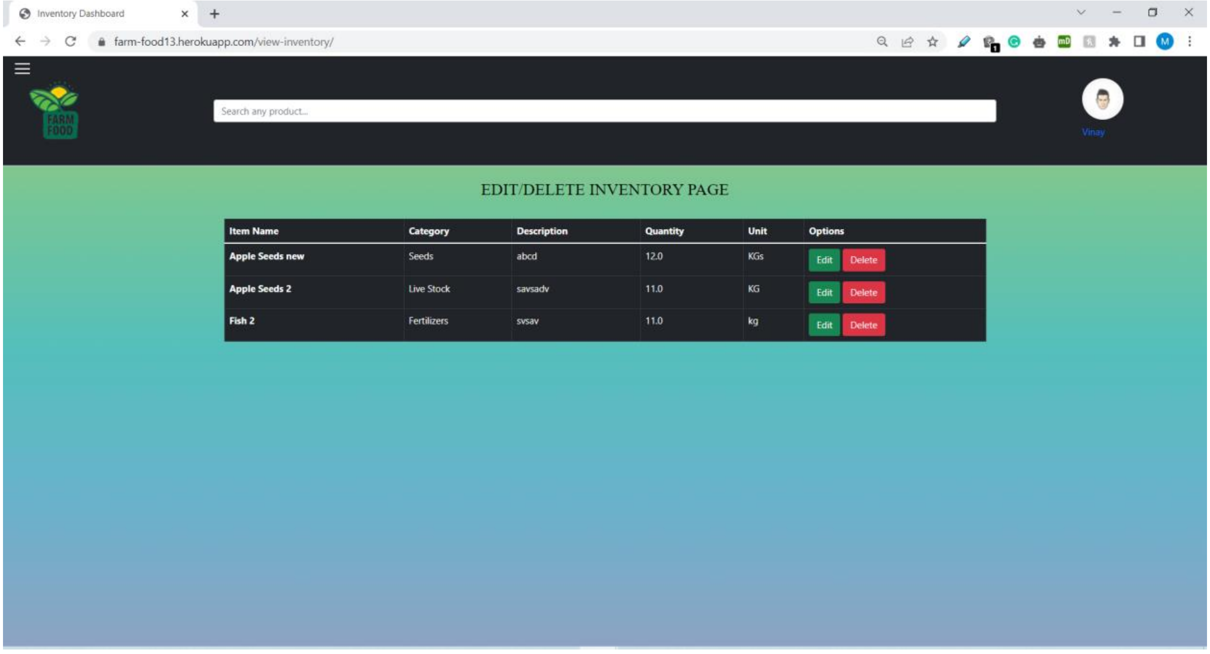
Category: Seeds

Description: abcd

Quantity: 12.0

Unit: KGs

3. View_Inventory_Page.html

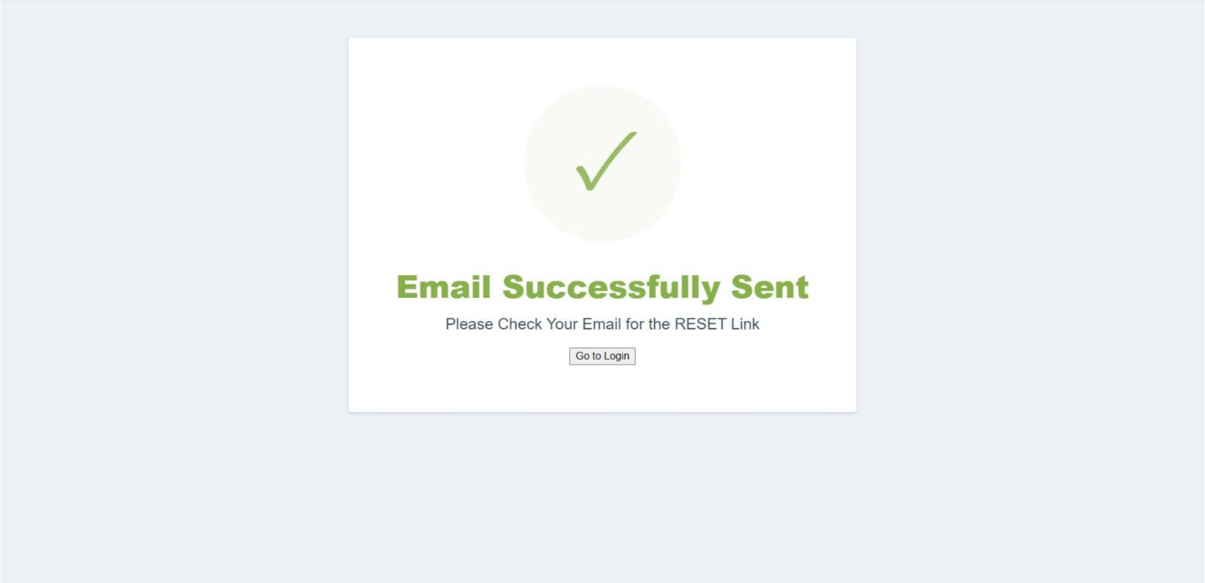


EDIT/DELETE INVENTORY PAGE

Item Name	Category	Description	Quantity	Unit	Options
Apple Seeds new	Seeds	abcd	12.0	KGs	Edit Delete
Apple Seeds 2	Live Stock	savsadv	11.0	KG	Edit Delete
Fish 2	Fertilizers	svsav	11.0	kg	Edit Delete

Success

1. Page_success.html

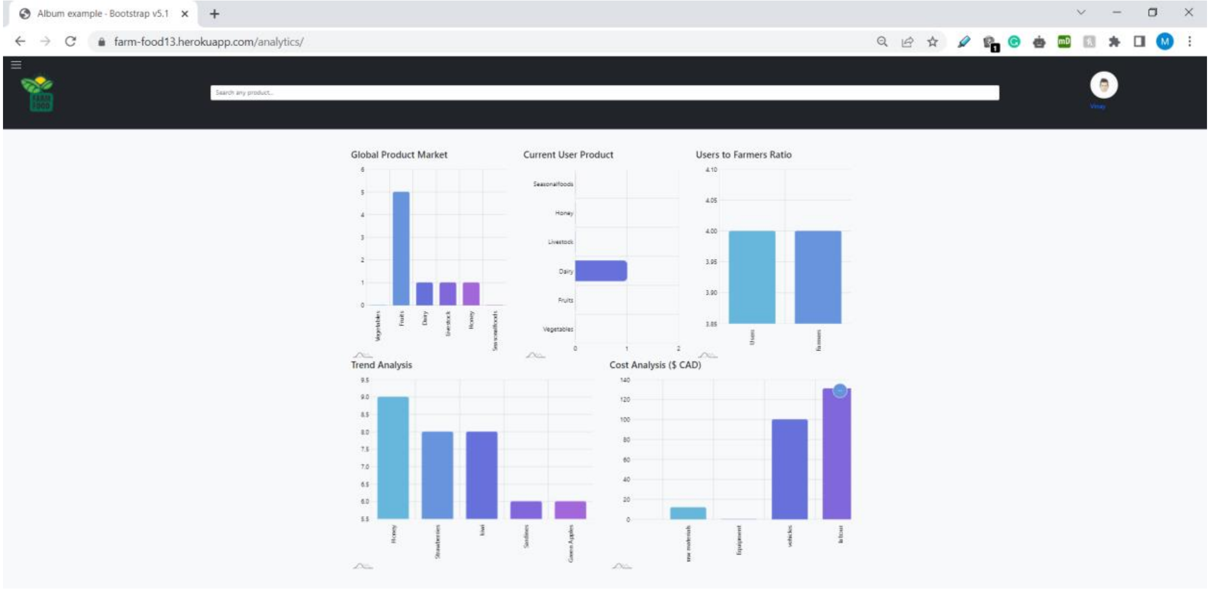


Email Successfully Sent

Please Check Your Email for the RESET Link

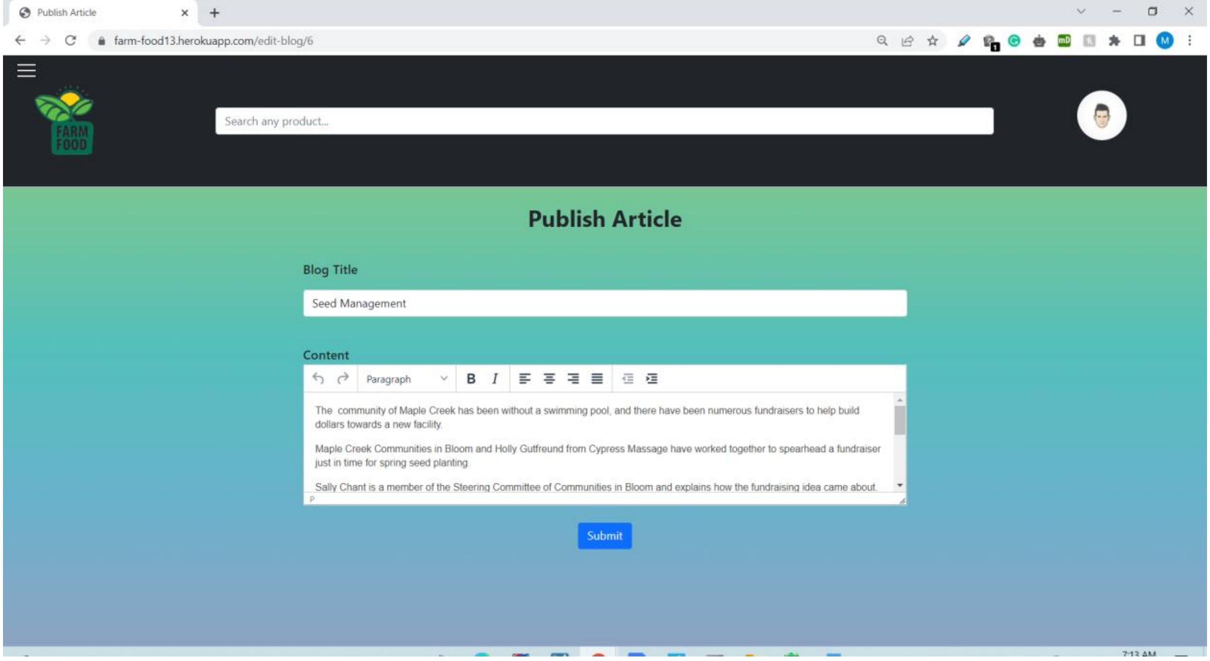
Analytics

1. Charts.html



Blog

1. Edit_Blog.html



Publish Article

Blog Title

Seed Management

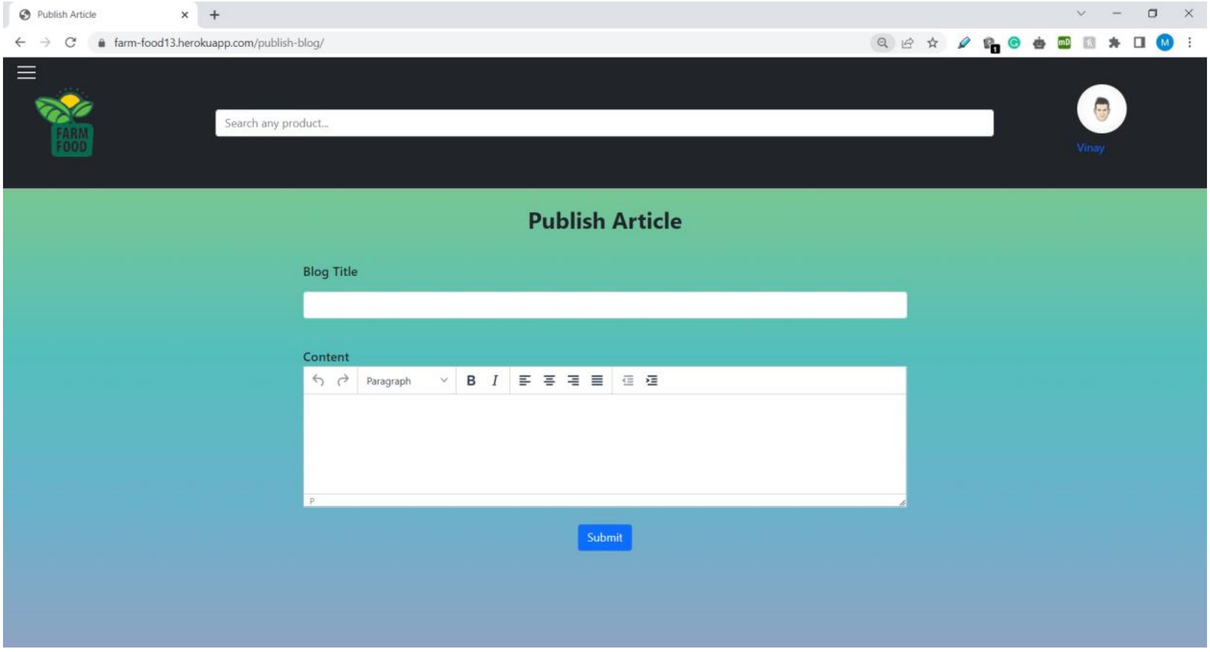
Content

The community of Maple Creek has been without a swimming pool, and there have been numerous fundraisers to help build dollars towards a new facility.

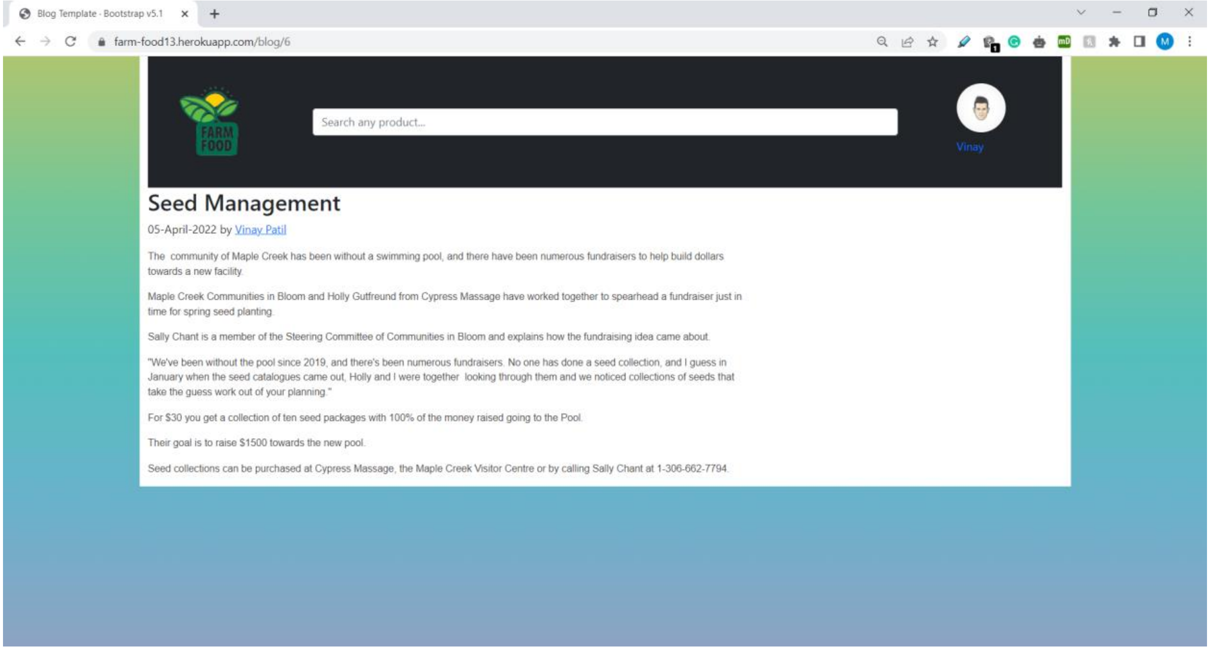
Maple Creek Communities in Bloom and Holly Gutfreund from Cypress Massage have worked together to spearhead a fundraiser just in time for spring seed planting.

Sally Chant is a member of the Steering Committee of Communities in Bloom and explains how the fundraising idea came about.

2. Publish_Article.html



3. Show_blog.html


Blog Template · Bootstrap v5.1
farm-food13.herokuapp.com/blog/6
FARM FOOD
Search any product...
Vinay

Seed Management

05-April-2022 by Vinay Patil

The community of Maple Creek has been without a swimming pool, and there have been numerous fundraisers to help build dollars towards a new facility.

Maple Creek Communities in Bloom and Holly Gutfreund from Cypress Massage have worked together to spearhead a fundraiser just in time for spring seed planting.

Sally Chant is a member of the Steering Committee of Communities in Bloom and explains how the fundraising idea came about.

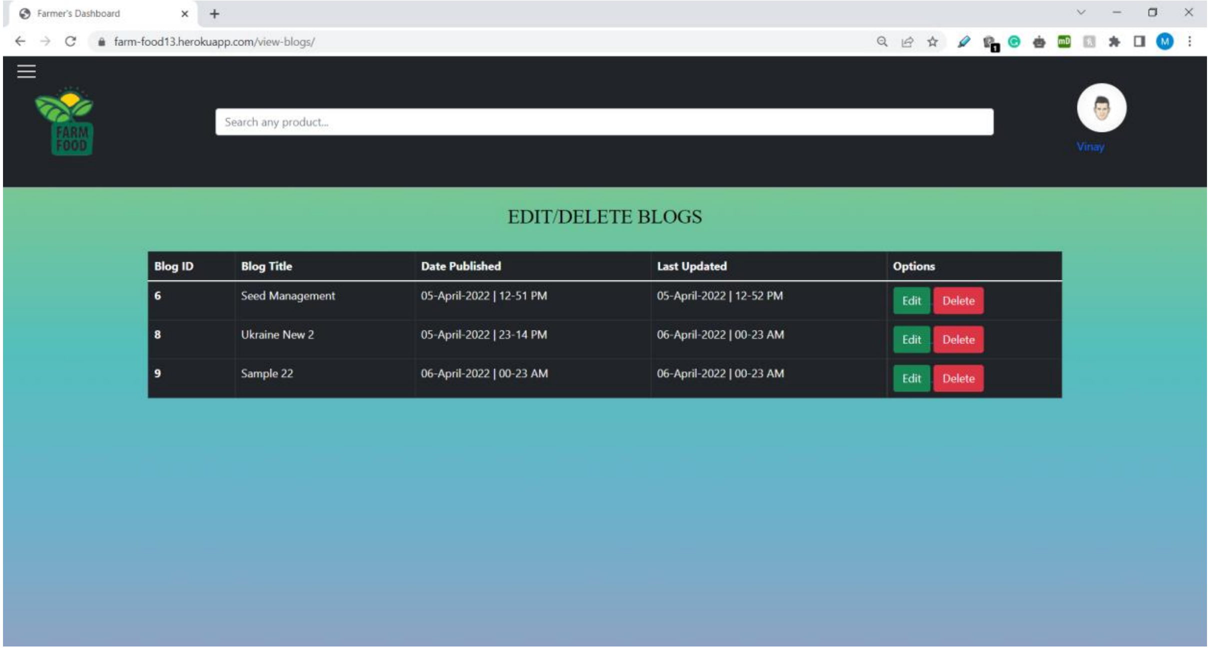
"We've been without the pool since 2019, and there's been numerous fundraisers. No one has done a seed collection, and I guess in January when the seed catalogues came out, Holly and I were together looking through them and we noticed collections of seeds that take the guess work out of your planning."

For $30 you get a collection of ten seed packages with 100% of the money raised going to the Pool.

Their goal is to raise $1500 towards the new pool.

Seed collections can be purchased at Cypress Massage, the Maple Creek Visitor Centre or by calling Sally Chant at 1-306-662-7794.

4. View_Blogs.html

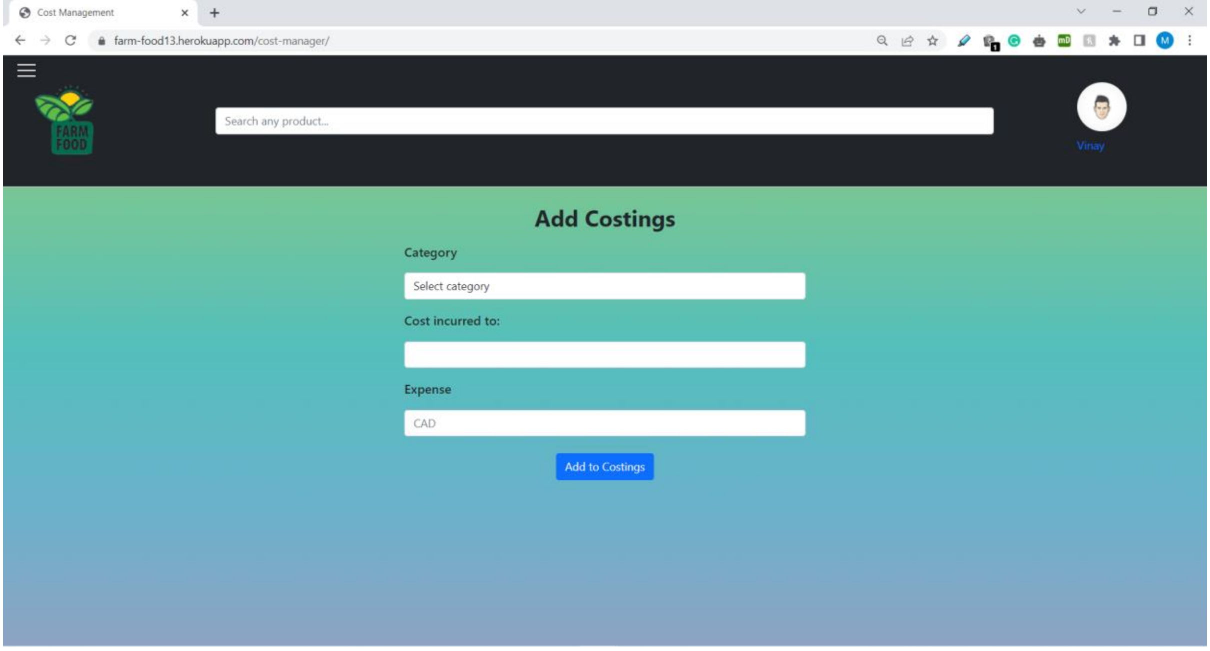


EDIT/DELETE BLOGS

Blog ID	Blog Title	Date Published	Last Updated	Options
6	Seed Management	05-April-2022 | 12-51 PM	05-April-2022 | 12-52 PM	Edit Delete
8	Ukraine New 2	05-April-2022 | 23-14 PM	06-April-2022 | 00-23 AM	Edit Delete
9	Sample 22	06-April-2022 | 00-23 AM	06-April-2022 | 00-23 AM	Edit Delete

Cost

1. Cost_Management.html



Add Costings

Category

Select category

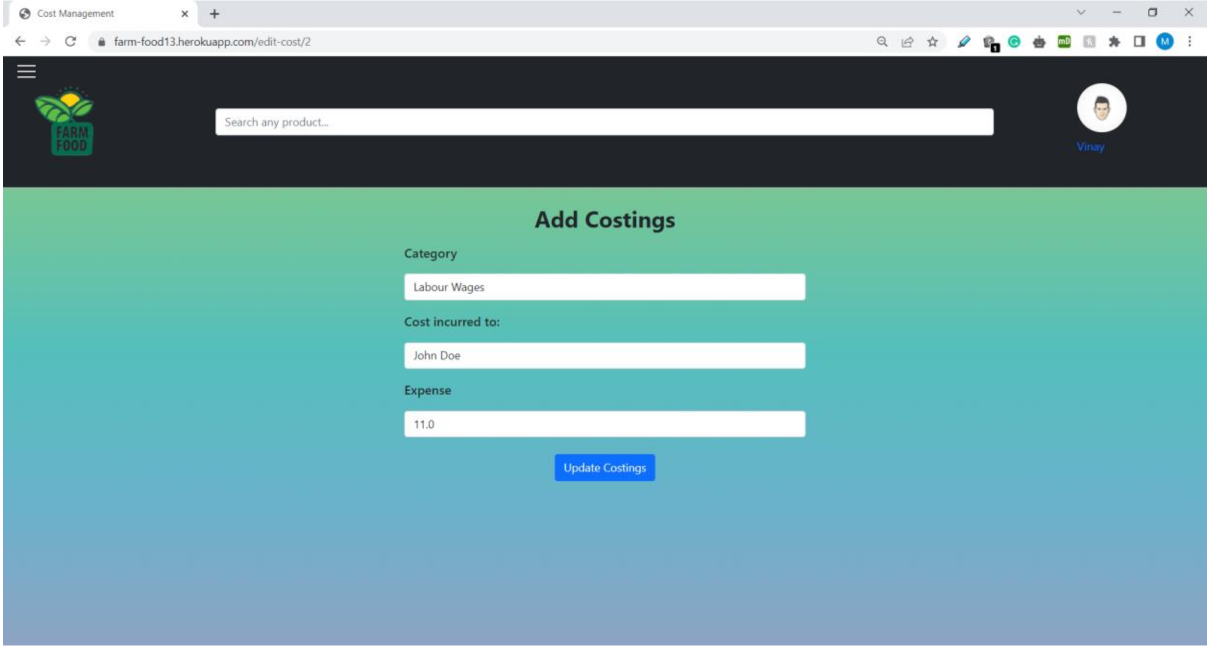
Cost incurred to:

Expense

CAD

Add to Costings

2. Edit_Cost.html



Add Costings

Category

Labour Wages

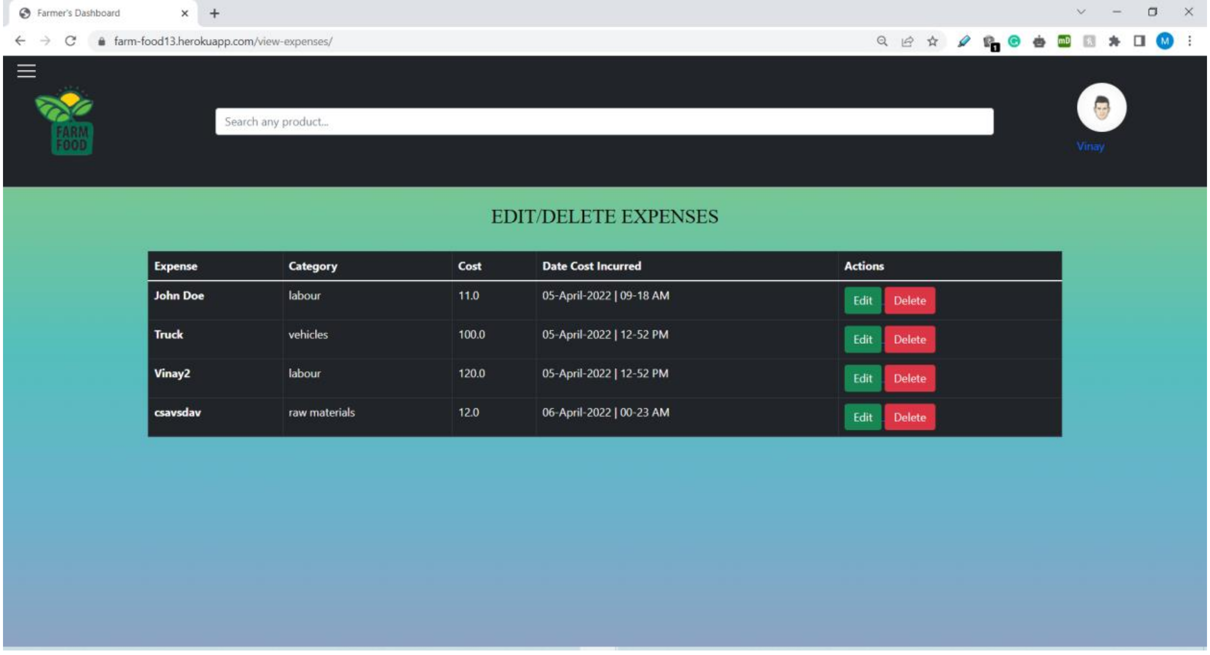
Cost incurred to:

John Doe

Expense

11.0

3. View_Expenses.html

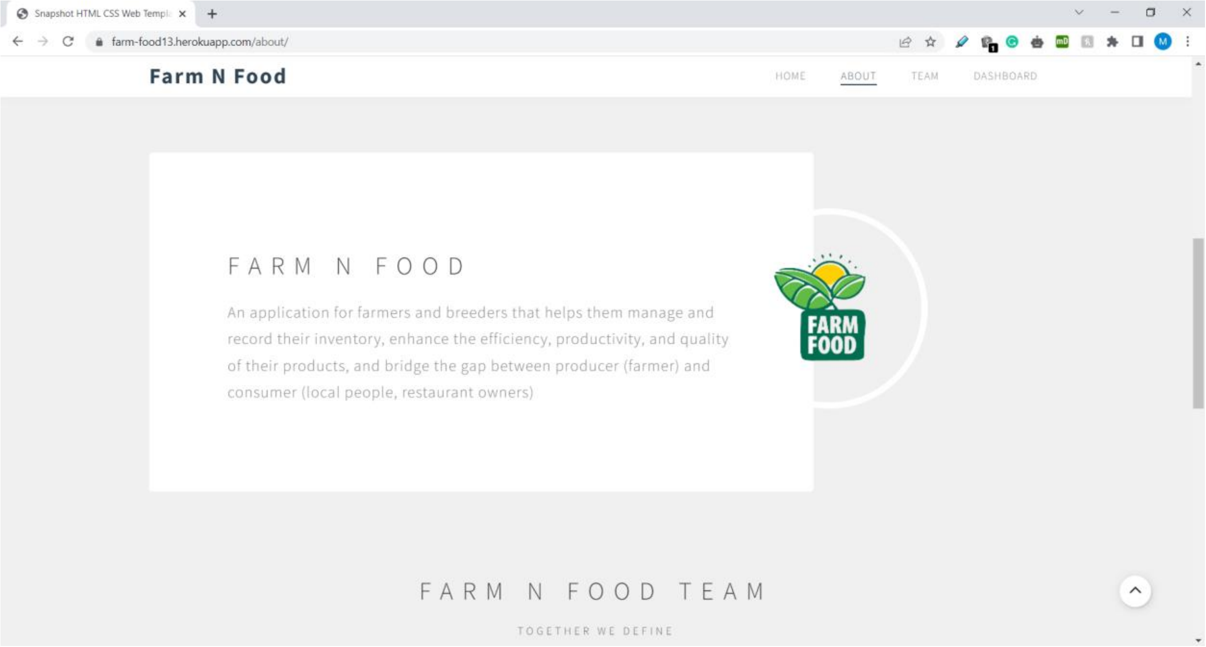


EDIT/DELETE EXPENSES

Expense	Category	Cost	Date Cost Incurred	Actions
John Doe	labour	11.0	05-April-2022 | 09-18 AM	Edit Delete
Truck	vehicles	100.0	05-April-2022 | 12-52 PM	Edit Delete
Vinay2	labour	120.0	05-April-2022 | 12-52 PM	Edit Delete
csavsdav	raw materials	12.0	06-April-2022 | 00-23 AM	Edit Delete

Home

1. About.html



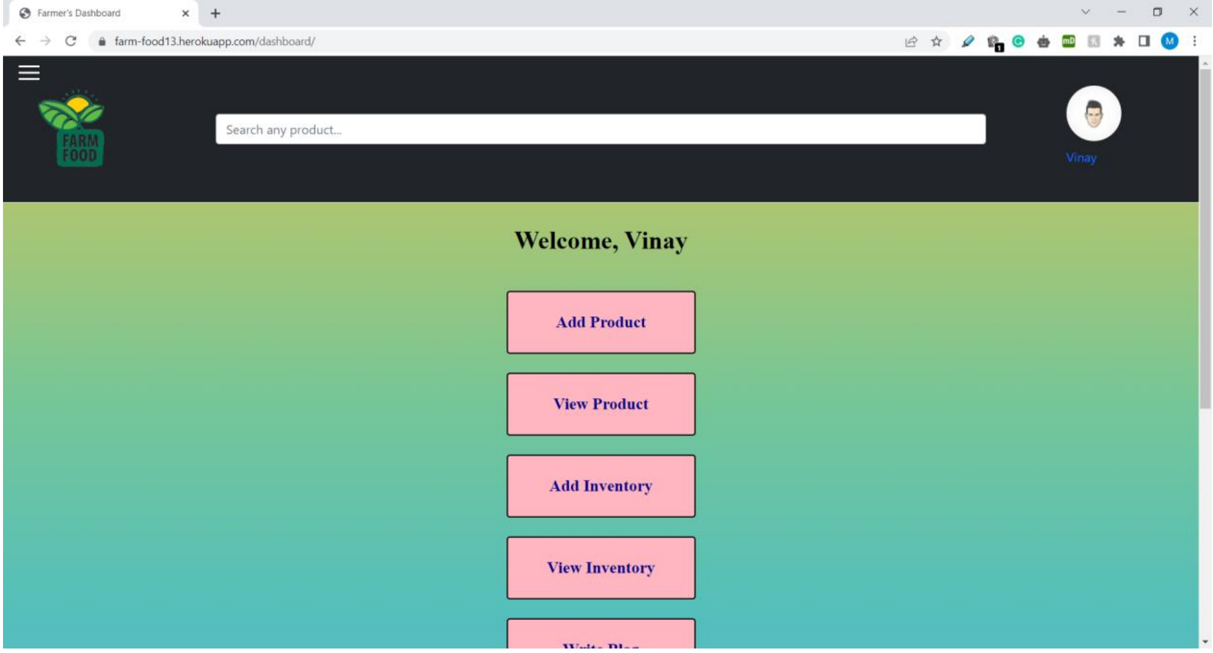
FARM N FOOD

An application for farmers and breeders that helps them manage and record their inventory, enhance the efficiency, productivity, and quality of their products, and bridge the gap between producer (farmer) and consumer (local people, restaurant owners)

FARM N FOOD TEAM

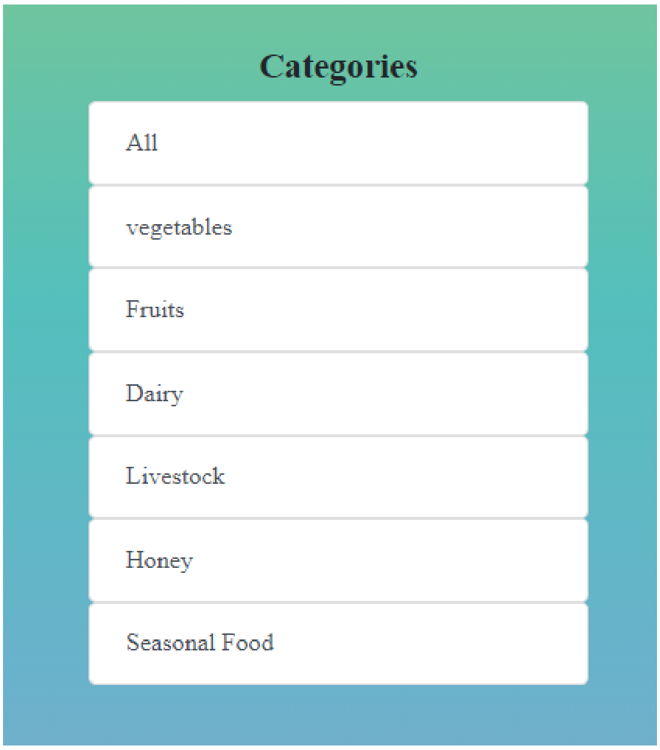
TOGETHER WE DEFINE

2. Farmer_Dashboard.html



Welcome, Vinay

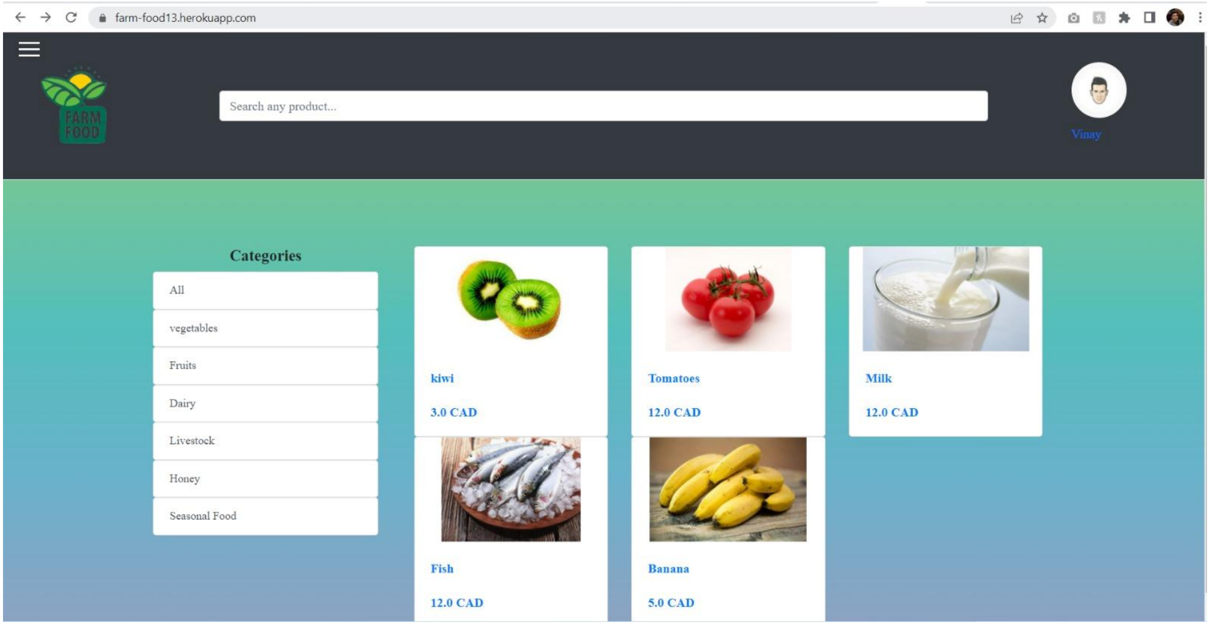
3. Get_Catagory.html



Categories

- All
- vegetables
- Fruits
- Dairy
- Livestock
- Honey
- Seasonal Food

4. HomePage.html



Categories

- All
- vegetables
- Fruits
- Dairy
- Livestock
- Honey
- Seasonal Food

kiwi

3.0 CAD

Tomatoes

12.0 CAD

Milk

12.0 CAD

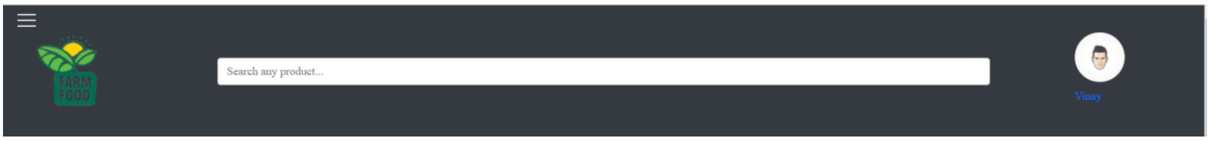
Fish

12.0 CAD

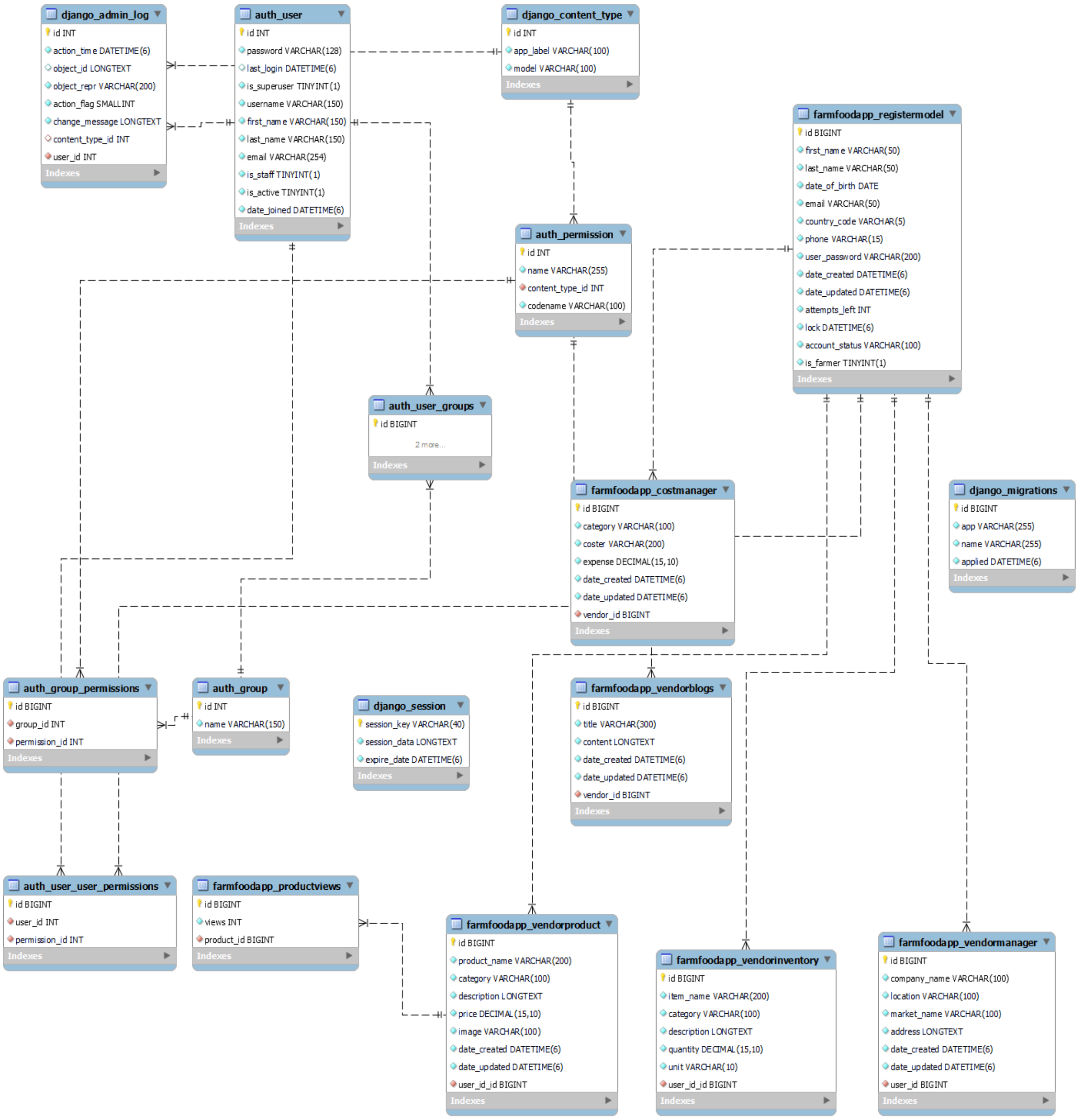
Banana

5.0 CAD

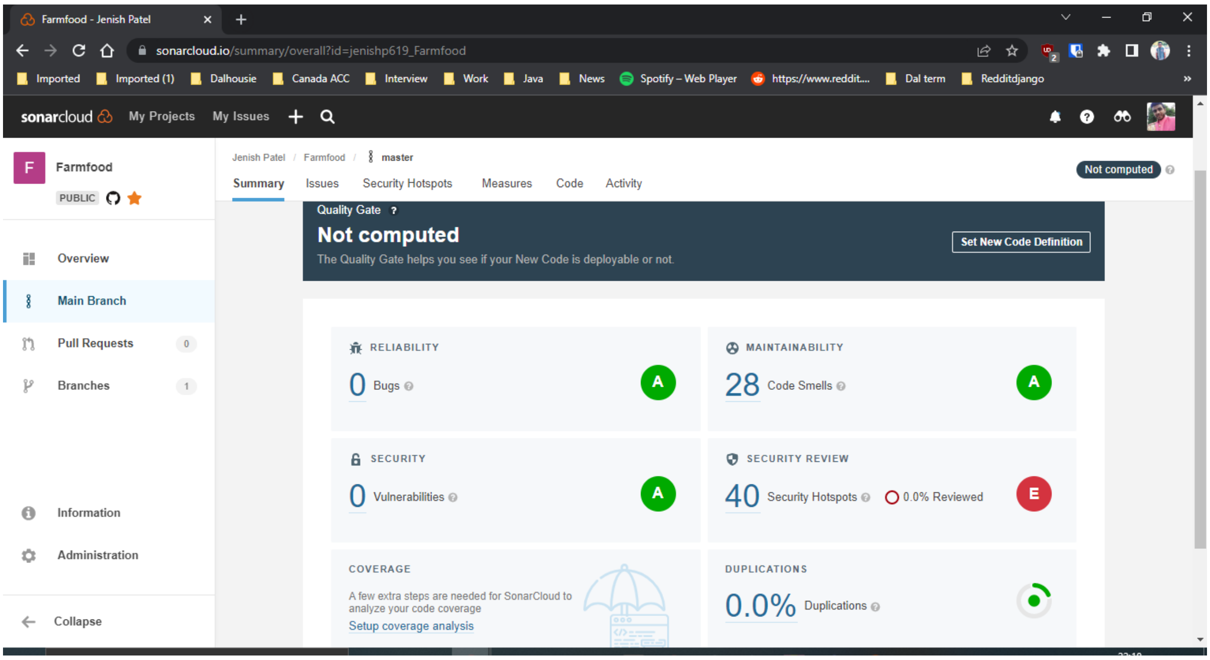
5. Search.html



Database ERD:



Sonarcloud (Code Smells)



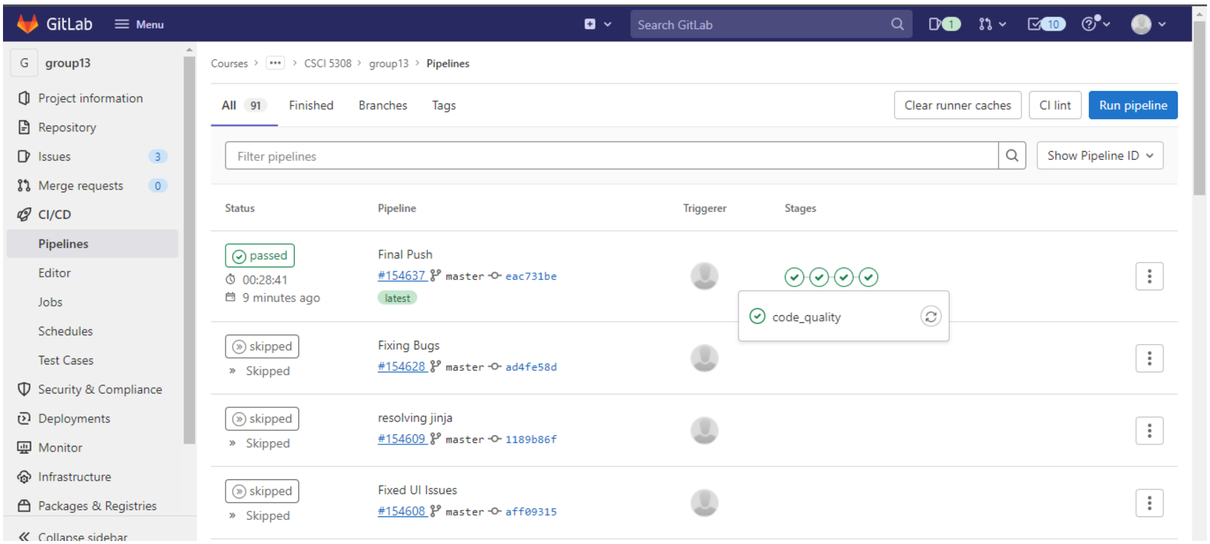
Quality Gate ?

Not computed

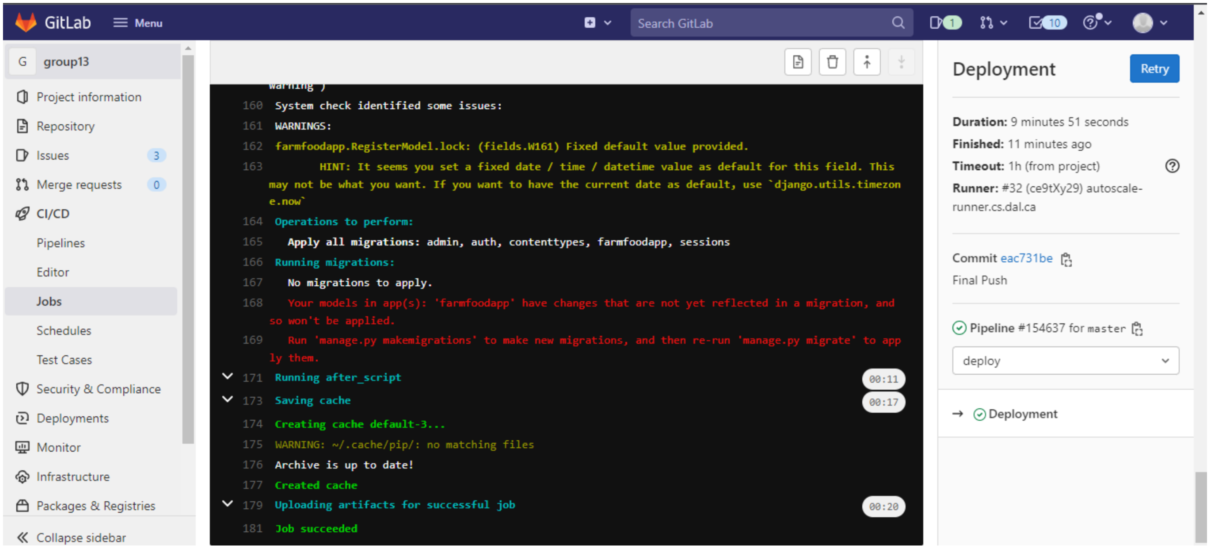
The Quality Gate helps you see if your New Code is deployable or not.

RELIABILITY 0 Bugs — A	MAINTAINABILITY 28 Code Smells — A
SECURITY 0 Vulnerabilities — A	SECURITY REVIEW 40 Security Hotspots — 0.0% Reviewed — E
COVERAGE A few extra steps are needed for SonarCloud to analyze your code coverage Setup coverage analysis	DUPLICATIONS 0.0% Duplications

Pipeline



Status	Pipeline	Triggerer	Stages
passed 00:28:41 9 minutes ago	Final Push #154637 master eac731be latest		✓ ✓ ✓ ✓ ✓ code_quality
skipped Skipped	Fixing Bugs #154628 master ad4fe58d		
skipped Skipped	resolving jinja #154609 master 1189b86f		
skipped Skipped	Fixed UI Issues #154608 master aff09315		



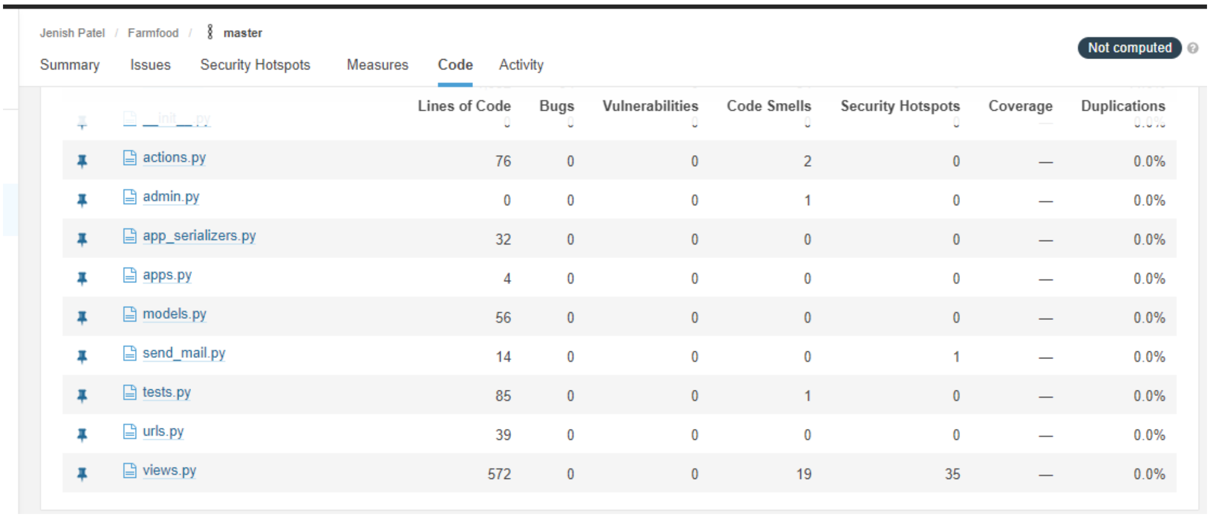
```
warning )
System check identified some issues:
WARNINGS:
farmfoodapp.RegisterModel.lock: (fields.W161) Fixed default value provided.
        HINT: It seems you set a fixed date / time / datetime value as default for this field. This
may not be what you want. If you want to have the current date as default, use `django.utils.timezon
e.now`
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, farmfoodapp, sessions
Running migrations:
  No migrations to apply.
  Your models in app(s): 'farmfoodapp' have changes that are not yet reflected in a migration, and
so won't be applied.
  Run 'manage.py makemigrations' to make new migrations, and then re-run 'manage.py migrate' to app
ly them.
Running after_script                                                                        00:11
Saving cache                                                                                00:17
Creating cache default-3...
WARNING: ~/.cache/pip/: no matching files
Archive is up to date!
Created cache
Uploading artifacts for successful job                                                      00:20
Job succeeded
```

Deployment

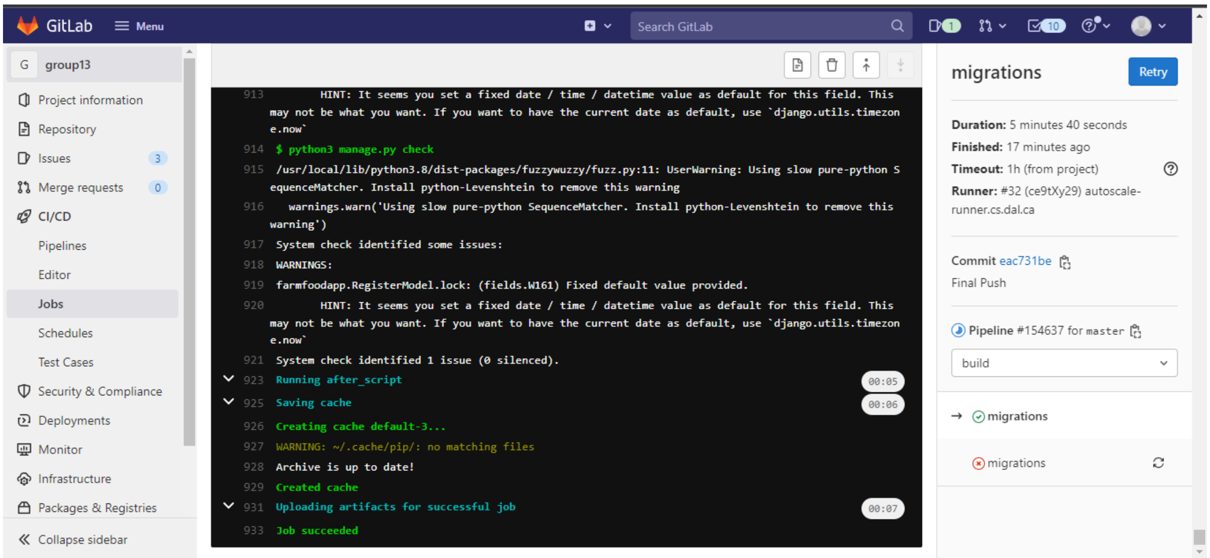
Duration: 9 minutes 51 seconds
Finished: 11 minutes ago
Timeout: 1h (from project)
Runner: #32 (ce9tXy29) autoscale-runner.cs.dal.ca

Commit eac731be
Final Push

✓ Pipeline #154637 for master



	Lines of Code	Bugs	Vulnerabilities	Code Smells	Security Hotspots	Coverage	Duplications
__init__.py	0	0	0	0	0	—	0.0%
actions.py	76	0	0	2	0	—	0.0%
admin.py	0	0	0	1	0	—	0.0%
app_serializers.py	32	0	0	0	0	—	0.0%
apps.py	4	0	0	0	0	—	0.0%
models.py	56	0	0	0	0	—	0.0%
send_mail.py	14	0	0	0	1	—	0.0%
tests.py	85	0	0	1	0	—	0.0%
urls.py	39	0	0	0	0	—	0.0%
views.py	572	0	0	19	35	—	0.0%



```
        HINT: It seems you set a fixed date / time / datetime value as default for this field. This
may not be what you want. If you want to have the current date as default, use `django.utils.timezon
e.now`
$ python3 manage.py check
/usr/local/lib/python3.8/dist-packages/fuzzywuzzy/fuzz.py:11: UserWarning: Using slow pure-python S
equenceMatcher. Install python-Levenshtein to remove this warning
  warnings.warn('Using slow pure-python SequenceMatcher. Install python-Levenshtein to remove this
warning')
System check identified some issues:
WARNINGS:
farmfoodapp.RegisterModel.lock: (fields.W161) Fixed default value provided.
        HINT: It seems you set a fixed date / time / datetime value as default for this field. This
may not be what you want. If you want to have the current date as default, use `django.utils.timezon
e.now`
System check identified 1 issue (0 silenced).
Running after_script                                                                        00:05
Saving cache                                                                                00:06
Creating cache default-3...
WARNING: ~/.cache/pip/: no matching files
Archive is up to date!
Created cache
Uploading artifacts for successful job                                                      00:07
Job succeeded
```

migrations

Duration: 5 minutes 40 seconds
Finished: 17 minutes ago
Timeout: 1h (from project)
Runner: #32 (ce9tXy29) autoscale-runner.cs.dal.ca

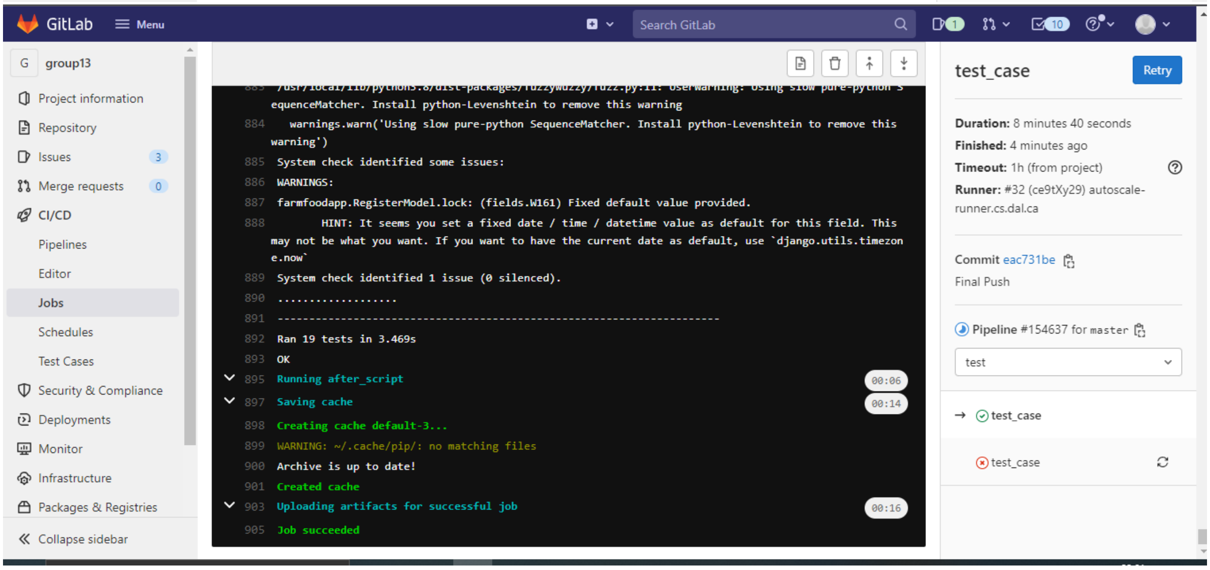
Commit eac731be
Final Push

Pipeline #154637 for master

build

→ migrations

migrations



```
/usr/local/lib/python3.8/dist-packages/fuzzywuzzy/fuzz.py:11: UserWarning: Using slow pure-python S
equenceMatcher. Install python-Levenshtein to remove this warning
  warnings.warn('Using slow pure-python SequenceMatcher. Install python-Levenshtein to remove this
warning')
System check identified some issues:
WARNINGS:
farmfoodapp.RegisterModel.lock: (fields.W161) Fixed default value provided.
        HINT: It seems you set a fixed date / time / datetime value as default for this field. This
may not be what you want. If you want to have the current date as default, use `django.utils.timezon
e.now`
System check identified 1 issue (0 silenced).
...................
----------------------------------------------------------------------
Ran 19 tests in 3.469s
OK
Running after_script                                                                        00:06
Saving cache                                                                                00:14
Creating cache default-3...
WARNING: ~/.cache/pip/: no matching files
Archive is up to date!
Created cache
Uploading artifacts for successful job                                                      00:16
Job succeeded
```

test_case

Duration: 8 minutes 40 seconds
Finished: 4 minutes ago
Timeout: 1h (from project)
Runner: #32 (ce9tXy29) autoscale-runner.cs.dal.ca

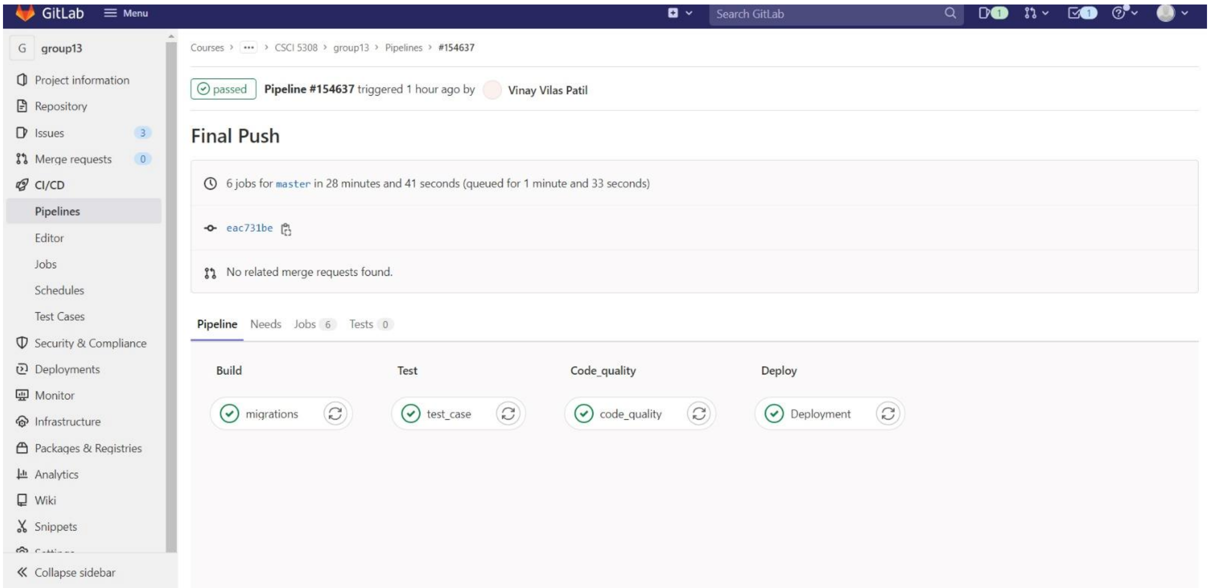
Commit eac731be
Final Push

Pipeline #154637 for master

test

→ test_case

test_case



passed **Pipeline #154637** triggered 1 hour ago by Vinay Vilas Patil

Final Push

6 jobs for master in 28 minutes and 41 seconds (queued for 1 minute and 33 seconds)

eac731be

No related merge requests found.

Build: migrations
Test: test_case
Code_quality: code_quality
Deploy: Deployment

Reference:

[1] "API reference," *Djangoproject.com*. [Online]. Available: https://docs.djangoproject.com/en/4.0/ref/. [Accessed: 05-Apr-2022].

[2] "Flake8," *PyPI*. [Online]. Available: https://pypi.org/project/flake8/. [Accessed: 05-Apr-2022].

[3] "Gunicorn," *PyPI*. [Online]. Available: https://pypi.org/project/gunicorn/. [Accessed: 05-Apr-2022].

[4] "WhiteNoise 6.0.0 documentation," *Evans.io*. [Online]. Available: http://whitenoise.evans.io/en/stable/. [Accessed: 05-Apr-2022].

[5] "FuzzyWuzzy python library," *GeeksforGeeks*, 24-Dec-2017. [Online]. Available: https://www.geeksforgeeks.org/fuzzywuzzy-python-library/. [Accessed: 05-Apr-2022].

[6] "Welcome to PyJWT — PyJWT 2.3.0 documentation," *Readthedocs.io*. [Online]. Available: https://pyjwt.readthedocs.io/en/latest/. [Accessed: 05-Apr-2022].

[7] "Pillow," *Readthedocs.io*. [Online]. Available: https://pillow.readthedocs.io/en/stable/. [Accessed: 05-Apr-2022].

[8] "Mysqlclient," *PyPI*. [Online]. Available: https://pypi.org/project/mysqlclient/. [Accessed: 05-Apr-2022].

[9] G. Maciel, "How to deploy a Django app on heroku - geek culture - medium," *Geek Culture*, 22-Jan-2021. [Online]. Available: https://medium.com/geekculture/how-to-deploy-a-django-app-on-heroku-4d696b458272. [Accessed: 05-Apr-2022].

[10] templatemo, "572+ free HTML CSS templates by," *templatemo*, 22-Mar-2022. [Online]. Available: https://templatemo.com/. [Accessed: 05-Apr-2022].

[11] T. Christie, "Serializers - Django REST framework," *Django-rest-framework.org*. [Online]. Available: https://www.django-rest-framework.org/api-guide/serializers/. [Accessed: 05-Apr-2022].

[12] "DSU farmers' market," *DSU Farmers' Market*. [Online]. Available: http://www.dsumarket.ca/. [Accessed: 05-Apr-2022].

[13] T. Christie, "Home - Django REST framework," *Django-rest-framework.org*. [Online]. Available: https://www.django-rest-framework.org/. [Accessed: 05-Apr-2022].

[14] M. Otto, J. Thornton, and Bootstrap contributors, "Introduction," *Getbootstrap.com*. [Online]. Available: https://getbootstrap.com/docs/5.0/getting-started/introduction/. [Accessed: 05-Apr-2022].