



Galactic Eatery - Project Proposal

→ Writeups of the design:

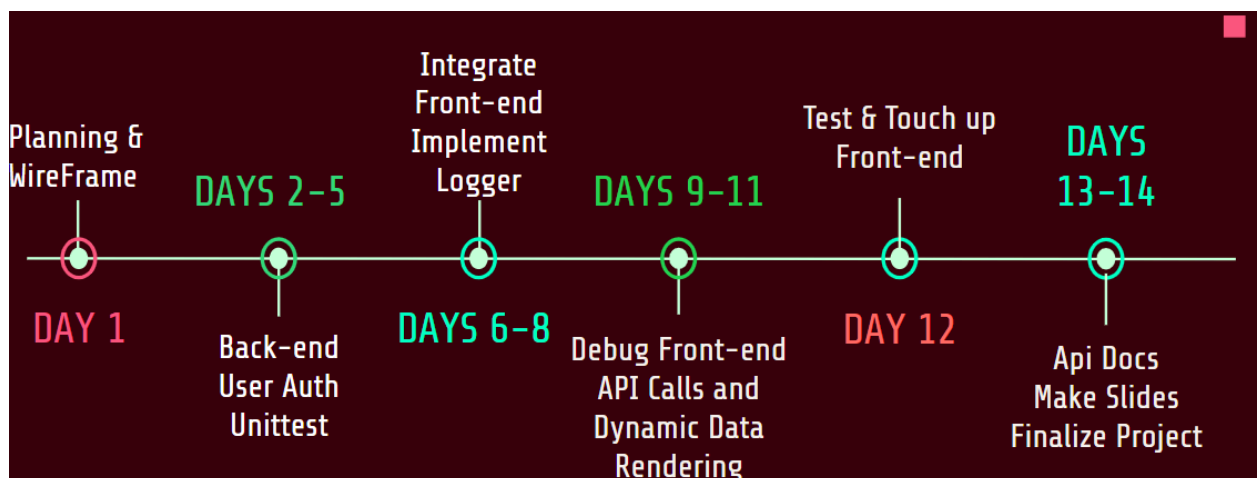
◆ What is it your are going to build and how does it fit the theme?

The theme of the project is the Galactic Market. We plan to create a single-page platform where users can access a galactic menu, enabling them to select dishes from various planets. Additionally, users will have the freedom to choose their favorite dishes from this content, which will be sourced directly from the database. The website will dynamically load this information based on the user's chosen location during their search.

◆ What tools will you use?

◆ How long do you expect to spend on the development of each part?

We have 2 weeks to develop this project.



→ API Endpoints documentation:

User Registration

URL: /register

Method: POST

Request Body:

- username (string, required): The username of the user.
- password (string, required): The password of the user.

Response:

- 201 Created: User registered successfully.
- 400 Bad Request: Username already exists.
- 500 Internal Server Error: Error occurred during registration.

Example using cURL:

```
curl -X POST \
  http://localhost:5000/register \
  -H 'Content-Type: application/json' \
  -d '{
    "username": "YOUR_USERNAME",
    "password": "YOUR_PASSWORD"
  }'
```

User Login

URL: /login Method: POST Request Body:

- username (string, required): The username of the user.
- password (string, required): The password of the user.

Response:

- 200 OK: Login successful. Returns a JWT token.
- 401 Unauthorized: Invalid username or password.
- 500 Internal Server Error: Error occurred during login.

Example using cURL:

```
curl -X POST \
```

```
http://localhost:5000/login \
-H 'Content-Type: application/json' \
-d '{
  "username": "YOUR_USERNAME",
  "password": "YOUR_PASSWORD"
}'
```

Get All Dishes

URL: /dishes Method: GET Response:

- 200 OK: Returns an array of all dishes.
- 500 Internal Server Error: Error occurred while fetching dishes.

Example using cURL:

```
curl -X GET \
http://localhost:5000/dishes
```

Filter Dishes by Name

URL: /dishes/name Method: GET Query Parameters:

- name (string, required): The name or partial name of the dish.

Response:

- 200 OK: Returns an array of dishes matching the provided name.
- 400 Bad Request: Name parameter is missing.
- 500 Internal Server Error: Error occurred while filtering dishes.

Example using cURL:

```
curl -X GET \
'http://localhost:5000/dishes/name?name=quasar-9'
```

Filter Dishes by Planet of Origin

URL: /dishes/:planet_of_origin Method: GET URL Parameters:

- planet_of_origin (string, required): The planet of origin of the dish.

Response:

- 200 OK: Returns an array of dishes from the specified planet of origin.
- 400 Bad Request: Planet of origin parameter is missing.
- 500 Internal Server Error: Error occurred while filtering dishes.

Example using cURL:

```
curl -X GET \
```

```
'http://localhost:5000/dishes/planet_of_origin?planet_of_origin=black%20hole'
```

Create a New Dish

URL: /dishes/create Method: POST Request Body: Dish object (JSON) Response:

- 200 OK: New dish created successfully.
- 500 Internal Server Error: Error occurred while creating the dish.

Example using cURL:

```
curl -X POST \
```

```
http://localhost:5000/dishes/create \
```

```
-H 'Content-Type: application/json' \
```

```
-d '{
```

```
  "name": "Dish Name",
```

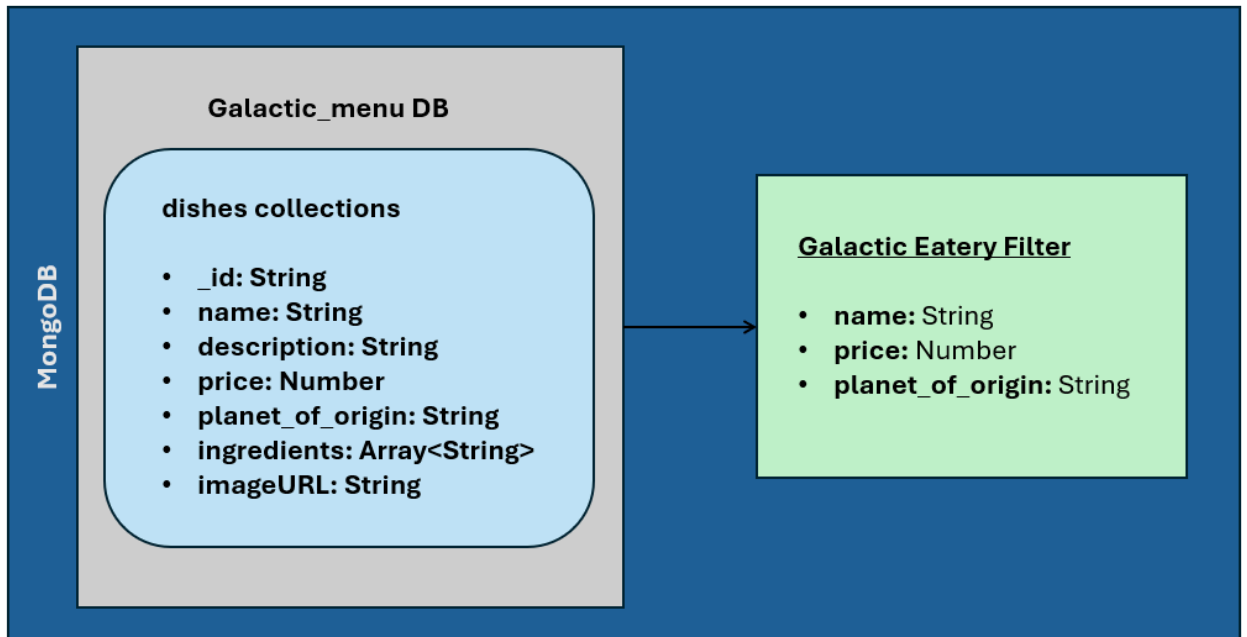
```
  "description": "Dish Description",
```

```
  "price": 9.99,
```

```
  "planet_of_origin": "Earth"
```

```
}'
```

→ Database documentation (UML)



→

→

→

→

→

→

→

→

→

→

→

→

→

→ Wireframes for any implemented UI portions of your application

