FarmClash - API Blueprint for Web-based Idle Game

Camille De Vuyst, Joren Van der Sande, Thomas De Volder,
Faisal Ettarrahi, Ferhat Van Herck, Siebe Mees

April 22, 2024

Contents

1 Introduction

This document outlines the API endpoints for a web-based idle game. Each endpoint requires user authentication and certain routes are restricted to admin users only.

1.1 Base URL

The base URL for all API requests is `https://team3.ua-ppdb.me/api/`.

2 Users

- Endpoint: `/users`

- Method: GET

- Description: Get a look at all the users.

- Access: Requires `admin` user privileges.

- Response: A JSON array of user objects, each containing:

```
[
  {
    "created_at": "Date",
    "email": "email@example.com",
    "password": "pbkdf2:sha256:hasedpassword",
    "username": "username"
  },
  {
    "created_at": "Another date",
    "email": "anotheremail@example.com",
    "password": "pbkdf2:sha256:anotherhashedpassword",
    "username": "anotherusername"
  }
  // Additional user objects as needed
]
```

3 Maps

- Endpoint: `/maps`

- Method: GET

- Description: Get a look at all the maps.

- Access: Requires `admin` user privileges.

- Response: A JSON array of map objects, each containing:

```
[
  {
    "map_id": 1,
    "username_owner": "ownerUsername",
    "width": 100,
    "height": 100,
    "created_at": "Tue, 26 Mar 2024 13:55:30 GMT"
  },
  {
    "map_id": 2,
    "username_owner": "anotherOwnerUsername",
    "width": 150,
    "height": 150,
    "created_at": "Wed, 27 Mar 2024 14:00:00 GMT"
  }
  // Additional map objects as needed
]
```

4 Resources

4.1 General Resource Information

- Endpoint: `/resources`

- Method: GET

- Description: Retrieves a comprehensive list of resources across all users. Designed for administrative use to monitor and manage resources system-wide.

- Access: Admin privileges required.

- Response: A JSON array of objects, each representing a resource. Each object includes the resource type, amount, owner's username, and a unique resource identifier. Example of a generalized response:

```
[
  {
    "resource_id": "unique_identifier",
    "resource_type": "type_of_resource",
    "amount": "quantity_of_resource",
    "owner": "username_of_owner"
  },
  ... // Additional resource objects
]
```

4.2   Specific Resource Information

- Endpoint: /resources/<string:username>

- Method: GET

- Description: Retrieves a list of resources owned by a specific user.

- Access: Admin privileges required.

- Response: A JSON array of objects, each representing a resource owned by the specified user. Each object includes the resource type, amount, and a unique resource identifier. Example of a generalized response:

```
[
  {
    "resource_id": "unique_identifier",
    "resource_type": "type_of_resource",
    "amount": "quantity_of_resource"
```

4

```
  },
  ... // Additional resource objects
]
```

5 Terrain Map

5.1 General Terrain Map Information

- Endpoint: `/terrain-map`

- Method: GET

- Description: Get a look at all the terrain tiles in the map.

- Access: Requires `logged in` user privileges.

- Response: A JSON array of tile objects, each containing:

```
[
  {
    "map_height": 2,
    "map_width": 2,
    "terrain_tiles": [
      [Water1.1, Water1.2],
      [Water2.1, Water2.2]
    ]
  }
]
```

5.2 Specific Terrain Tile Information

- Endpoint: `/terrain-map/<string:friend_username>`

- Method: GET

- Description: Get a look at a user specific terrain tile in the map.

- Access: Requires `logged in` user privileges.

- Response: A JSON object containing the terrain tile information:

```
[
  {
    "map_height": 2,
    "map_width": 2,
    "terrain_tiles": [
      [Water1.1, Water1.2],
      [Water2.1, Water2.2]
    ]
  }
]
```

6 Friendships

- Endpoint: `/friends`

- Method: GET

- Description: Get a list of all friends for the current user.

- Access: Requires `logged in` user privileges.

- Response: A JSON array of relationship objects, each containing:

```
[
  "friend1",
  "friend2",
  "friend3"
]
```

7 Chat

- Endpoint: `/messages/<string:friend_name>`

- Method: GET

- Description: Get a list of all chat messages for the current user.

- Access: Requires `logged in` user privileges.

- Response: A JSON array of chat message objects, each containing:

```
[
  {
    message_id: 1,
    "sender": "senderUsername",
    "receiver": "receiverUsername",
    "message": "Hello, how are you?",
    "created_at": "Tue, 26 Mar 2024 13:55:30 GMT"
  },
  {
    message_id: 2,
    "sender": "receiverUsername",
    "receiver": "senderUsername",
    "message": "I'm good, thanks!",
    "created_at": "Tue, 26 Mar 2024 13:56:30 GMT"
  }
  // Additional chat message objects as needed
]
```

8 Leaderboard

- Endpoint: `/leaderboard`

- Method: GET

- Description: Get a list of top 3 users, 2 friends and yourself sorted by their score.

- Access: Requires `logged in` user privileges.

- Response: A JSON array of user objects, each containing:

```
[
  {
    "place": 1,
    "username": "username",
    "score": 100
  },
  {
    "place": 2,
    "username": "anotherUsername",
    "score": 200
  }
  // Additional user objects as needed
]
```

9 Market

- endpoint: @game_blueprint.route('/update-building-map', methods=['POST'])

- Method: POST

- Description: Handles POST requests to insert JSON data into the database. Expects JSON data in the request body.

- Access: Requires `logged in` user privileges.

- Response: A JSON object with status and message indicating success or failure.

10 Building

- endpoint @game_blueprint.route('/fetch-building-information', methods=['GET'])

- Method: GET

- Description: Handles GET requests to fetch building information for the current user.

- Access: Requires `logged in` user privileges.

- Response: A JSON object containing building information.