

## 1. Route: /create-account

- **Request Type:** POST
- **Purpose:** Creates a new user account with a username and password.
- **Request Body:**
  - username (String): The user's chosen username.
  - password (String): The user's chosen password.
- **Response Format:** JSON
- **Success Response Example:**
- **Code:** 201
  - Content:**json
 

```
{
  "message": "Account created successfully for user 'newuser123'."
}
```
- **Error Response Example** (e.g., missing fields):
- **Code:** 400
  - Content:**json
 

```
{
  "error": "Invalid request payload. 'username' and 'password' are required."
}
```
- **Error Response Example** (unexpected error):
- **Code:** 500
  - Content:**json
 

```
{
  "error": "Internal server error."
}
```

### Example Request:json

```
{
  "username": "newuser123",
  "password": "securepassword"
}
```

**Example Response:**

```

{
  "message": "Account created successfully for user 'newuser123'."
}

```

**2. Route: /login**

- **Request Type:** POST
- **Purpose:** Authenticates a user by verifying the provided username and password.
- **Request Body:**
- username (String): The user's username.
- password (String): The user's password.
- **Response Format:** JSON
- **Success Response Example:**
- **Code:** 200

**Content:**json

```

{
  "message": "User 'newuser123' authenticated successfully."
}

```

- **Error Response Example** (e.g., invalid credentials):
- **Code:** 401

**Content:**json

```

{
  "error": "Invalid username or password."
}

```

- **Error Response Example** (unexpected error):
- **Code:** 500

**Content:**json

```

{
  "error": "Internal server error."
}

```

- **Example Request:**json

```
{
  "username": "newuser123",
  "password": "securepassword"
}
```

- **Example Response:**json

```
{
  "message": "User 'newuser123' authenticated successfully."
}
```

### 3. Route: /update-password

- **Request Type:** POST
- **Purpose:** Updates a user's password by verifying the current password and setting a new one.
- **Request Body:**
  - username (String): The username of the user requesting the password change.
  - current\_password (String): The current password of the user to verify their identity.
  - new\_password (String): The new password that the user wants to set.
- **Response Format:** JSON
- **Success Response Example:**
- **Code:** 200
  - Content:**json
 

```
{
            "message": "Password updated successfully for user 'newuser123'."
          }
```
- **Error Response Example** (e.g., missing fields):
- **Code:** 400
  - Content:**json
 

```
{
            "error": "Invalid request payload. 'username', 'current_password', and 'new_password' are required."
          }
```

- **Error Response Example** (invalid authentication):
- **Code:** 401  
**Content:**json
 

```
{
  "error": "Invalid username or password."
}
```
- **Error Response Example** (unexpected error):
- **Code:** 500  
**Content:**json
 

```
{
  "error": "Internal server error."
}
```
- **Example Request:**json
 

```
{
  "username": "newuser123",
  "current_password": "oldpassword",
  "new_password": "newsecurepassword"
}
```
- **Example Response:**json
 

```
{
  "message": "Password updated successfully for user 'newuser123'."
}
```

#### 4. Route: /api/favorites (POST)

- **Request Type:** POST
- **Purpose:** Adds a new favorite location for a user.
- **Request Body:**
  - username (String): The username of the user adding the favorite location.
  - password (String): The password of the user to authenticate.
  - location\_name (String): The name of the location to be added to the favorites.
- **Response Format:** JSON
- **Success Response Example:**
- **Code:** 201  
**Content:**json
 

```
{
  "message": "Favorite location 'Paris' added successfully.",

```

```

    "favorite_location": {
      "id": 1,
      "location_name": "Paris",
      "latitude": 48.8566,
      "longitude": 2.3522
    }
  }
}

```

- **Error Response Example** (e.g., missing location\_name):
- **Code:** 400  
**Content:**json
 

```

{
  "error": "Invalid request payload. 'location_name' is required."
}

```
- **Error Response Example** (authentication failure):
- **Code:** 401  
**Content:**json
 

```

{
  "error": "Invalid username or password."
}

```
- **Error Response Example** (unexpected error):
- **Code:** 500  
**Content:**json
 

```

{
  "error": "Internal server error."
}

```
- **Example Request:**json
 

```

{
  "username": "newuser123",
  "password": "securepassword",
  "location_name": "Paris"
}

```

- **Example Response:**json

```
{
  "message": "Favorite location 'Paris' added successfully.",
  "favorite_location": {
    "id": 1,
    "location_name": "Paris",
    "latitude": 48.8566,
    "longitude": 2.3522
  }
}
```

## 5. Route: /api/favorites (GET)

- **Request Type:** GET
- **Purpose:** Retrieves all favorite locations for a user.
- **Query Parameters:**
- username (String): The username of the user whose favorite locations are being retrieved.
- password (String): The password of the user for authentication.
- **Response Format:** JSON
- **Success Response Example:**
- **Code:** 200

**Content:**json

```
{
  "favorites": [
    {
      "id": 1,
      "location_name": "Paris",
      "latitude": 48.8566,
      "longitude": 2.3522
    },
    {
      "id": 2,
      "location_name": "New York",
      "latitude": 40.7128,
      "longitude": -74.0060
    }
  ]
}
```

- **Error Response Example** (missing query parameters):
- **Code:** 400  
**Content:**json
 

```
{
  "error": "Missing 'username' or 'password' query parameters."
}
```
- **Error Response Example** (authentication failure):
- **Code:** 401  
**Content:**json
 

```
{
  "error": "Invalid username or password."
}
```
- **Error Response Example** (unexpected error):
- **Code:** 500  
**Content:**json
 

```
{
  "error": "Internal server error."
}
```
- **Example Request:**bash  
 GET /api/favorites?username=newuser123&password=securepassword
- **Example Response:**json
 

```
{
  "favorites": [
    {
      "id": 1,
      "location_name": "Paris",
      "latitude": 48.8566,
      "longitude": 2.3522
    }
  ]
}
```

```

    },
    {
      "id": 2,
      "location_name": "New York",
      "latitude": 40.7128,
      "longitude": -74.0060
    }
  ]
}

```

## 6. Route: /api/favorites/weather (GET)

- **Request Type:** GET
- **Purpose:** Retrieves all favorite locations for a user along with their current weather.
- **Query Parameters:**
- username (String): The username of the user whose favorite locations with weather are being retrieved.
- password (String): The password of the user for authentication.
- **Response Format:** JSON
- **Success Response Example:**
- **Code:** 200

**Content:**json

```

{
  "favorites": [
    {
      "id": 1,
      "location_name": "Paris",
      "latitude": 48.8566,
      "longitude": 2.3522,
      "current_weather": {
        "time": "2024-12-10T15:30:00Z"
        "temperature": 22.5
        "humidity": 60
        "windspeed": 5.5
        "precipitation": 0.0
      }
    },
    {
      "id": 2,
      "location_name": "New York",

```



```

        "latitude": 40.7128,
        "longitude": -74.0060,
        "current_weather": {
          "time": "2024-12-10T15:30:00Z"
          "temperature": 30.6
          "humidity": 70
          "windspeed": 6.5
          "precipitation": 1.0
        }
      }
    ]
  }
}

```

- **Error Response Example** (missing query parameters):
- **Code:** 400
  - Content:**json
 

```

{
  "error": "Missing 'username' or 'password' query parameters."
}
          
```
  
- **Error Response Example** (authentication failure):
- **Code:** 401
  - Content:**json
 

```

{
  "error": "Invalid username or password."
}
          
```
  
- **Error Response Example** (unexpected error):
- **Code:** 500
  - Content:**json
 

```

{
  "error": "Internal server error."
}
          
```

- **Example Request:**bash  
GET /api/favorites/weather?username=newuser123&password=securepassword

- **Example Response:**json

```
{
  "favorites": [
    {
      "id": 1,
      "location_name": "Paris",
      "latitude": 48.8566,
      "longitude": 2.3522,
      "current_weather": {
        "time": "2024-12-10T15:30:00Z"
        "temperature": 22.5
        "humidity": 60
        "windspeed": 5.5
        "precipitation": 0.0
      }
    },
    {
      "id": 2,
      "location_name": "New York",
      "latitude": 40.7128,
      "longitude": -74.0060,
      "current_weather": {
        "time": "2024-12-10T15:30:00Z"
        "temperature": 30.6
        "humidity": 70
        "windspeed": 6.5
        "precipitation": 1.0
      }
    }
  ]
}
```

## 7. Route: `/api/favorites/int:favorite_id/weather` (GET)

- **Request Type:** GET
- **Purpose:** Retrieves the current weather for a specific favorite location.
- **Path Parameters:**
- `favorite_id` (int, required): The ID of the favorite location whose weather is being retrieved.
- **Query Parameters:**
- `username` (string, required): The username of the account.
- `password` (string, required): The password of the account.
- **Response Format:** JSON
- **Success Response Example:**
- **Code:** 200

**Content:**json

```
{
  "favorite_location": {
    "id": 1,
    "location_name": "Paris",
    "latitude": 48.8566,
    "longitude": 2.3522,
    "current_weather": {
      "time": "2024-12-10T15:30:00Z"
      "temperature": 22.5
      "humidity": 60
      "windspeed": 5.5
      "precipitation": 0.0
    }
  }
}
```

- **Error Response Example** (missing query parameters):
- **Code:** 400

**Content:**json

```
{
  "error": "Missing 'username' or 'password' query parameters."
}
```

```
}
```

- **Error Response Example** (unauthorized access):

- **Code:** 401

**Content:**json

```
{
  "error": "Invalid username or password."
}
```

- **Error Response Example** (favorite location not found):

- **Code:** 404

**Content:**json

```
{
  "error": "Favorite location with ID '1' not found."
}
```

- **Error Response Example** (internal error):

- **Code:** 500

**Content:**json

```
{
  "error": "Could not fetch weather data."
}
```

- **Example Request:**bash

```
GET /api/favorites/1/weather?username=newuser123&password=securepassword
```

**Example Response:**json

```
{
  "favorite_location": {
    "id": 1,
    "location_name": "Paris",
    "latitude": 48.8566,
    "longitude": 2.3522,
    "current_weather": {
```

```

        "time": "2024-12-10T15:30:00Z"
        "temperature": 22.5
        "humidity": 60
        "windspeed": 5.5
        "precipitation": 0.0
    }
}

```

## 8. Route: `/api/favorites/int:favorite_id/historical` (GET)

- **Request Type:** GET
- **Purpose:** Retrieves historical weather data for a specific favorite location.
- **Path Parameters:**
- `favorite_id` (int, required): The ID of the favorite location whose historical weather data is being retrieved.
- **Query Parameters:**
- `username` (string, required): The username of the account.
- `password` (string, required): The password of the account.
- `start_date` (string, required): The start date for the historical weather data in 'YYYY-MM-DD' format.
- `end_date` (string, required): The end date for the historical weather data in 'YYYY-MM-DD' format.
- **Response Format:** JSON
- **Success Response Example:**
- **Code:** 200

**Content:** json

```

{
  "favorite_location": {
    "id": 1,
    "latitude": 48.8566,
    "longitude": 2.3522
  },
  "historical_weather": [
    {
      "date": "2024-12-01"
      "temperature_2m_max": 15.2
      "temperature_2m_min": 10.5
      "apparent_temperature_max": 16.8
      "apparent_temperature_min": 9.0
      "daylight_duration": 9.8
      "precipitation_sum": 0.1
    }
  ]
}

```

```

    "precipitation_probability_max": 20
    "wind_speed_10m_max": 7.3
    ...
  }
]
}

```

- **Error Response Example** (missing query parameters):
- **Code:** 400  
**Content:**json
 

```

{
  "error": "Missing 'username', 'password', 'start_date', or 'end_date' query parameters."
}

```
- **Error Response Example** (unauthorized access):
- **Code:** 401  
**Content:**json
 

```

{
  "error": "Invalid username or password."
}

```
- **Error Response Example** (favorite location not found):
- **Code:** 404  
**Content:**json
 

```

{
  "error": "Favorite location with ID '1' not found."
}

```
- **Error Response Example** (internal error):
- **Code:** 500  
**Content:**json
 

```

{
  "error": "Could not fetch historical weather data."
}

```

```
}
```

- **Example Request:**bash  
GET /api/favorites/1/historical?  
username=newuser123&password=securepassword&start\_date=2024-12-01&end  
\_date=2024-12-02

**Example Response:**json

```
{
  "favorite_location": {
    "id": 1,
    "latitude": 48.8566,
    "longitude": 2.3522
  },
  "historical_weather": [
    {
      "date": "2024-12-01"
      "temperature_2m_max": 15.2
      "temperature_2m_min": 10.5
      "apparent_temperature_max": 16.8
      "apparent_temperature_min": 9.0
      "daylight_duration": 9.8
      "precipitation_sum": 0.1
      "precipitation_probability_max": 20
      "wind_speed_10m_max": 7.3
    },
    ...
  ]
}
```

## 9. Route: /api/favorites/int:favorite\_id/forecast (GET)

- **Request Type:** GET
- **Purpose:** Retrieves the weather forecast for a specific favorite location.
- **Path Parameters:**
- favorite\_id (int, required): The ID of the favorite location whose weather forecast is being retrieved.
- **Query Parameters:**
- username (string, required): The username of the account.
- password (string, required): The password of the account.
- days (int, optional): The number of days to retrieve the forecast for (default is 7).

- **Response Format:** JSON
- **Success Response Example:**
- **Code:** 200

**Content:**json

```
{
  "favorite_location": [
    "id": 123
    "latitude": 40.7128
    "longitude": -74.0060
  ],
  "historical_weather": [
    {
      "date": "2024-12-01"
      "temperature_2m_max": 15.2
      "temperature_2m_min": 10.5
      "apparent_temperature_max": 16.8
      "apparent_temperature_min": 9.0
      "daylight_duration": 9.8
      "precipitation_sum": 0.1
      "precipitation_probability_max": 20
      "wind_speed_10m_max": 7.3
    },
    ...
  ]
}
```

- **Error Response Example** (missing query parameters):
- **Code:** 400

**Content:**json

```
{
  "error": "Missing 'username' or 'password' query parameters."
}
```

- **Error Response Example** (unauthorized access):
- **Code:** 401

**Content:**json

```
{
  "error": "Invalid username or password."
}
```



```
}
```

- **Error Response Example** (favorite location not found):

- **Code:** 404

**Content:**json

```
{
  "error": "Favorite location with ID '1' not found."
}
```

- **Error Response Example** (internal error):

- **Code:** 500

**Content:**json

```
{
  "error": "Could not fetch weather forecast data."
}
```

- **Example Request:**bash

```
GET /api/favorites/1/forecast?
```

```
username=newuser123&password=securepassword&days=5
```

- **Example Response:**json

```
{
  "favorite_location": [
    "id": 123
    "latitude": 40.7128
    "longitude": -74.0060
  ],
  "historical_weather": [
    {
      "date": "2024-12-01"
      "temperature_2m_max": 15.2
      "temperature_2m_min": 10.5
      "apparent_temperature_max": 16.8
      "apparent_temperature_min": 9.0
      "daylight_duration": 9.8
      "precipitation_sum": 0.1
    }
  ]
}
```

```

        "precipitation_probability_max": 20
        "wind_speed_10m_max": 7.3
    },
    ...
]
}

```

## 10. Route: /api/health

- **Request Type:** GET
- **Purpose:** Verifies if the service is running and healthy.
- **Request Body:** None
- **Response Format:** JSON
- **Success Response Example:**
  - Code: 200
  - Content:
 

```
{ "status": "healthy" }
```

- **Example Request:**  
No request body needed for this endpoint.

- **Example Response:** json
 

```
{
  "status": "healthy"
}
```