

June 4, 2023. I wrote the Prompt below. Every subsequent word was written by ChatGPT and is reproduced here with no changes. - Ian Heller

Prompt: Write an API in Python to connect White Cup Solutions analytics module to Epicor ERP

Sure! Here's an example of how you can create an API in Python to connect White Cup Solutions' analytics module to Epicor ERP. Please note that this is a simplified example, and you may need to adapt it based on your specific requirements and the APIs provided by White Cup Solutions and Epicor ERP.

```
import requests

# Define the base URLs for White Cup Solutions and Epicor ERP APIs
white_cup_url = "https://whitecupsolutions-api.com"
epicor_url = "https://epicor-erp-api.com"

# Define your API credentials for White Cup Solutions and Epicor ERP
white_cup_api_key = "your_white_cup_api_key"
epicor_api_key = "your_epicor_api_key"

# Define the endpoint to retrieve analytics data from White Cup Solutions
white_cup_analytics_endpoint = "/analytics"

# Define the endpoint to create/update data in Epicor ERP
epicor_data_endpoint = "/data"

# Create a function to retrieve analytics data from White Cup Solutions
def get_analytics_data():
    headers = {
        "Authorization": f"Bearer {white_cup_api_key}"
    }
    response = requests.get(white_cup_url + white_cup_analytics_endpoint,
headers=headers)
    if response.status_code == 200:
        return response.json()
    else:
        return None

# Create a function to create/update data in Epicor ERP
def sync_data_to_epicor(data):
    headers = {
        "Authorization": f"Bearer {epicor_api_key}",
        "Content-Type": "application/json"
    }
```

```

    response = requests.post(epicor_url + epicor_data_endpoint, headers=headers,
    json=data)
    if response.status_code == 200:
        return response.json()
    else:
        return None

```

```

# Create the main API endpoint to sync analytics data from White Cup Solutions to
Epicor ERP

```

```

def sync_analytics_to_epicor():
    analytics_data = get_analytics_data()
    if analytics_data:
        result = sync_data_to_epicor(analytics_data)
        if result:
            return {"status": "success"}

```

```

    return {"status": "error"}

```

```

# Run the API

```

```

if __name__ == "__main__":
    # You can use a web framework like Flask or Django to run the API on a web server
    # For simplicity, let's assume we're using Flask
    from flask import Flask, jsonify

```

```

    app = Flask(__name__)

```

```

    @app.route("/sync", methods=["POST"])

```

```

    def sync_endpoint():
        result = sync_analytics_to_epicor()
        return jsonify(result)

```

```

    app.run()

```

In this example, we first define the base URLs for the White Cup Solutions and Epicor ERP APIs, along with your API credentials for each service. Then, we create functions to retrieve analytics data from White Cup Solutions and sync that data to Epicor ERP. Finally, we define a main API endpoint ("/sync") using the Flask web framework, which calls the **sync_analytics_to_epicor()** function and returns the result as a JSON response.

Remember to replace the placeholder URLs, API keys, and endpoints with the actual values provided by White Cup Solutions and Epicor ERP. Additionally, you may need to customize the data transformation and mapping logic to match the specific data structures and requirements of your analytics module and Epicor ERP system.