How to Set Up Docker from a GitHub Repository

Overview:
This guide will walk you through the steps to set up and run a Dockerized application cloned from a GitHub repositor

Prerequisites:
- Installed Docker Desktop (https://www.docker.com/products/docker-desktop/)
- Installed Git (https://git-scm.com/)
- GitHub repository containing a Dockerfile or docker-compose.yml

Step-by-Step Guide:

1■■ Clone the Repository from GitHub

Open your terminal or command prompt and navigate to the desired directory:

```
cd path/to/your/folder
```

Clone the GitHub repository:

```
git clone https://github.com/your-username/your-repository.git
```

Navigate to the cloned folder:

```
cd your-repository
```

2■■ Check the Available Docker Files

Inside your project directory, check if one of the following files exists:
- Dockerfile — for building a single container
- docker-compose.yml — for managing multi-container setups

3■■ Build and Run the Docker Container

If the project uses a Dockerfile:
Build the Docker image:

```
docker build -t your-image-name .
```

Run the Docker container:

```
docker run -d -p 8000:8000 --name your-container-name your-image-name
```

Example:

```
docker build -t valid8-backend .
docker run -d -p 8000:8000 --name valid8-backend-container valid8-backend
```

If the project uses a docker-compose.yml:
Run the docker-compose command:

```
docker-compose up --build
```

To run in detached mode:

```
docker-compose up -d --build
```

4■■ Verify Running Containers

Check if your containers are running:

```
docker ps
```

To view logs:

```
docker logs your-container-name
```

5■■ Stop and Remove Containers (if needed)

Stop a running container:

```
docker stop your-container-name
```

Remove a container:

```
docker rm your-container-name
```

Stop and remove all containers with docker-compose:

```
docker-compose down
```

Notes:

- Make sure Docker Desktop is running before executing any Docker commands.
- Update port mappings as necessary depending on your application's settings.
- Use .env files for environment variables if your project supports it.

Example Workflow:

```
git clone https://github.com/KHRISTMAE/Valid8-Attendance-Recognition-System.git
cd Valid8-Attendance-Recognition-System/backend
docker build -t valid8-backend .
docker run -d -p 8000:8000 --name valid8-backend-container valid8-backend
```

Conclusion:

Following these steps, you can successfully set up and run a Dockerized project cloned from any GitHub repository.