GitHub Actions: Build and Push CSHub Play Environment Images

This doc contains a GitHub Actions workflow that automates the process of building Docker images for the CSHub project (frontend and backend environments), pushing them to Docker Hub, and pulling the updated images to a remote AWS EC2 instance.

Workflow File

`.github/workflows/CSHub-push-env.yml`

Trigger

- **Manual trigger only**: The workflow runs when manually triggered via the GitHub Actions interface (`workflow_dispatch`).

What It Does

1. **Checkout Code**
    a. Clones the repository into the GitHub Actions runner.
2. **Set Up Docker Build**
    a. Enables advanced Docker build features.
3. **Login to Docker Hub**
    a. Authenticates with Docker Hub using secrets (`DOCKER_USERNAME` and `DOCKER_PASSWORD`).
4. **Build and Push Docker Images**
    a. Builds the backend environment Docker image from `./backend/Dockerfile.backend_env`.
    b. Builds the frontend environment Docker image from `./frontend/Dockerfile.frontend_env`.
    c. Pushes both images to Docker Hub under the `cmusvsc/smu_cshub` namespace:
        i. `cmusvsc/smu_cshub:cshub-backend-env`
        ii. `cmusvsc/smu_cshub:cshub-frontend-env`
5. **Remote AWS EC2 pull docker CSHub env images via SSH**
    a. SSHs into a remote EC2 instance using credentials stored in secrets (`EC2_HOST`, `EC2_USER`, `EC2_SSH_KEY`).

b. Logs into Docker Hub on the remote EC2 machine.
c. Pulls the latest Docker images for both frontend and backend env images.

Required Secrets (Saved in Github Actions Secrets and Variables)

Secret Name	Description
`DOCKER_USERNAME`	Docker Hub username
`DOCKER_PASSWORD`	Docker Hub password
`EC2_HOST`	Public IP or hostname of the EC2 instance
`EC2_USER`	SSH username for the EC2 instance
`EC2_SSH_KEY`	Private SSH key for EC2 access

Notes

- Make sure the EC2 instance has Docker installed and your SSH key is added to its authorized keys.
- Ensure that the Docker images are properly tagged and maintained to avoid unexpected behavior.