

Lab 6

Deploy to AWS

Software Studio
Datalab, CS, NTHU
2023

Gitlab

lab-weathermood-server-file-todo

The screenshot shows the GitLab web interface for a project named 'lab-weathermood-server-file-todo'. The interface is divided into a left sidebar with navigation links and a main content area. The sidebar includes links for Project information, Repository, Issues (0), Merge requests (0), Security & Compliance, Deployments, Monitor, Infrastructure, Packages & Registries, Analytics, Wiki, Snippets, and Settings. The main content area displays the project's name, ID (5308), and statistics (2 Commits, 20 Branches, 0 Tags, 51 KB Files, 51 KB Storage). It also shows the current branch (master) and a list of actions (History, Find file, Web IDE, Clone). A recent commit titled 'Update README.md' by Sheng-ya Chiu is shown, along with a table of recent commits. The README content is displayed below, showing the project's purpose and deployment instructions.

courses > 2023-spring > lab-weathermood-server-file-todo

lab-weathermood-server-file-todo
Project ID: 5308

2 Commits 20 Branches 0 Tags 51 KB Files 51 KB Storage

master lab-weathermood-server-file-todo / +

History Find file Web IDE Clone

Update README.md
Sheng-ya Chiu authored 11 minutes ago

7e4d6d33

Upload File README Auto DevOps enabled Add LICENSE Add CHANGELOG Add CONTRIBUTING

Configure Integrations

Name	Last commit	Last update
README.md	Update README.md	11 minutes ago

README.md

Weathermood Server File-TODO

1 Deploy to AWS

The client side code is in the `weathermood_no_redux` project, `server-file` branch The server side code is in the `weathermood-server_no_redux` project, `file` branch

You can follow the lab tutorial for deploying the code to AWS.

2 Weathermood Server File - TODO

Currently, the function of TODOs are written in client side. In this lab, you can practice to implement these functions in server side like posts.

Hint:

Sign-up for AWS



恭喜您！

感謝您註冊 AWS。

我們正在啟用您的帳戶，需要幾分鐘的時間。當這個程序完成時，您會收到一封電子郵件。

[前往 AWS 管理主控台](#)

[註冊另一個帳戶](#)或[聯絡銷售人員](#)

Identity and Access Management (IAM)

Find IAM service

The screenshot shows the AWS IAM console interface. At the top is a dark navigation bar with the AWS logo, a '服務' (Services) button, a search bar, and a keyboard shortcut '[Option+S]'. Below this is a left-hand navigation pane with the title 'Identity and Access Management (IAM)' and a search bar labeled '搜尋 IAM'. Under the search bar is a '儀表板' (Dashboard) link. A '存取管理' (Access Management) dropdown menu is open, listing '使用者群組' (Groups), '使用者' (Users), '角色' (Roles), '政策' (Policies), '身分供應商' (IdP), and '帳戶設定' (Account Settings). The main content area is titled 'IAM 儀表板' (IAM Dashboard). It features a '安全建議' (Security Recommendations) section with a red '1' badge. The first recommendation is a warning icon and the text '為根使用者新增 MFA' (Add MFA for root user), with a subtext explaining that MFA improves account security. The second recommendation is a green checkmark icon and the text '根使用者沒有作用中的存取金鑰' (Root user has no active access keys), with a subtext explaining that using IAM users instead of the root user's keys improves security. Below this is an 'IAM 資源' (IAM Resources) section with a table showing counts for Groups, Users, Roles, and Policies.

使用者群組	使用者	角色	政策
0	0	2	0

Identity and Access Management (IAM)

For security reason, we have to create new user with less permission
Users -> Add user

Identity and Access Management (IAM)

儀表板

▼ 存取管理

- 使用者群組
- 使用者**
- 角色
- 政策
- 身分供應商
- 帳戶設定

IAM > 使用者

使用者 (0) 資訊
IAM 使用者是一種具有長期憑證的身分

新增使用者

依使用者名稱或存取金鑰尋找使用

使用者名稱	密碼期限	作用中金鑰使用期限
-------	------	-----------

Identity and Access Management (IAM)

- Username should be as clear
- “AWS Management Console Access”
- Uncheck “Require password reset” if you want to keep the custom password
- Finish the form then click next

IAM > 使用者 > 建立使用者

步驟 1
指定使用者詳細資訊

步驟 2
設定許可

步驟 3
檢閱和建立

步驟 4
擷取密碼

指定使用者詳細資訊

使用者詳細資訊

使用者名稱

weathermood

使用者名稱最長可達 64 個字元。有效字元：A-Z、a-z、0-9 和 +, ., @, _ (連字號)

☒ 提供使用者對 AWS 管理主控台的存取權 – 選用
如果您向人員提供主控台存取權，則 [最佳實務](#) 以在 IAM Identity Center 管理其存取權。

您是否向人員提供主控台存取權？

☐ 在 Identity Center 指定使用者 – 建議
建議您使用 Identity Center，以向人員提供主控台存取權。透過 Identity Center，您可以集中管理使用者對其 AWS 帳戶和雲端應用程式的存取權。

☒ 我想要建立 IAM 使用者
建議只有當您需要透過存取金鑰、AWS CodeCommit 或 Amazon Keyspaces 的服務特定憑證，或用於緊急帳戶存取的備份憑證，啟用以程式設計方式存取時，才建立 IAM 使用者。

主控台密碼

☐ 自動產生的密碼
您可以在建立使用者之後檢視密碼。

☒ 自訂密碼
輸入使用者的自訂密碼。

☐ 顯示密碼

☐ 使用者必須在下次登入時建立新的密碼 (建議)。
使用者會自動取得 [IAMUserChangePassword](#) 政策，以允許他們變更自己的密碼。

如果您透過 AWS CodeCommit 或 Amazon Keyspaces 的存取金鑰或服務特定憑證，建立以程式設計方式存取，則可在建立此 IAM 使用者之後產生存取金鑰或憑證。 [進一步了解](#)

取消 下一步

Identity and Access Management (IAM)

Create group for managing permissions

IAM > 使用者 > 建立使用者

步驟 1
指定使用者詳細資訊

步驟 2
設定許可

步驟 3
檢閱和建立

步驟 4
擷取密碼

設定許可

請在現有群組中新增使用者或建立新使用者。使用群組是依職務管理使用者許可的最佳實務方式。 [進一步了解](#)

許可選項

☒ 在群組中新增使用者
請在現有群組中新增使用者或建立新群組。建議您依群組的職務來管理使用者許可。

☐ 複製許可
複製現有使用者的所有群組成員資格、連接的受管政策及內嵌政策。

☐ 直接連接政策
直接使受管政策與使用者連接。建議的最佳實務作法反而是使政策與群組連接，然後再將使用者新增至適當的群組內。

開始使用群組
建立群組，並選取要連接到群組的政策。建議使用群組來依照工作職能、AWS 服務存取權或自訂許可來管理使用者許可。 [進一步了解](#)

建立群組

▶ **許可範圍 - 選用**

設定許可範圍，以控制此使用者的許可上限。使用此進階功能可將許可管理委派給其他人。 [進一步了解](#)

取消

上一步

下一步

Identity and Access Management (IAM)

Search “AdministratorAccess-AWSElasticBeanstalk,” name your group, then create

建立使用者群組

建立使用者群組，並選取要連接到群組的政策。建議使用群組來依照工作職能、AWS 服務存取權或自訂許可來管理使用者許可。 [進一步了解](#)

使用者群組名稱
輸入有意義的名稱以識別此群組。

lab_demo

最多 128 個字元。使用英數和 '+,=, @, -, _' 字元。

許可政策 (1/831)

awselasticbeanstalk 15 相符項目

政策名稱	類型	使用形式
<input checked="" type="checkbox"/> AdministratorAccess-AWSElasticBeanstalk	AWS 受管	無
<input type="checkbox"/> AWSElasticBeanstalkCustomPlatformforEC2Role	AWS 受管	無
<input type="checkbox"/> AWSElasticBeanstalkEnhancedHealth	AWS 受管	無

Identity and Access Management (IAM)

Add your user to the group you just created

許可選項

☒ 在群組中新增使用者
請在現有群組中新增使用者或建立新群組。建議您依群組的職務來管理使用者許可。

☐ 複製許可
複製現有使用者的所有群組成員資格、連接的受管政策及內嵌政策。

☐ 直接連接政策
直接使受管政策與使用者連接。建議的最佳實務作法反而是使政策與群組連接，然後再將使用者新增至適當的群組內。

使用者群組 (1/1)

☒

群組名稱 [↗](#)

☒

lab_demo

▲

使用者

▼

☒

已連接的政策 [↗](#)

▼

☒

已建立

▼

0	AdministratorAccess-AWSElasticBeanstalk	2023-04-19 (現在)
---	---	-----------------

▶ 許可範圍 - 選用

設定許可範圍，以控制此使用者的許可上限。使用此進階功能可將許可管理委派給其他人。 [進一步了解](#) [↗](#)

取消

上一步

下一步

Identity and Access Management (IAM)

Tag is optional, for you to organize, track, and control access for users.
Create User.

檢閱和建立

檢閱您的選擇。建立使用者之後，您可以檢視和下載自動產生的密碼 (如果已啟用)。

使用者詳細資訊

使用者名稱
weathermood

主控台密碼類型
Custom password

需要重設密碼
否

許可摘要

< 1 >

名稱 



類型



使用形式



lab_demo

群組

許可群組

標籤 - 選用

標籤是您可以新增至 AWS 資源的鍵值對，以協助識別、組織或搜尋資源。選擇您要與此使用者關聯的任何標籤。

沒有與該資源相關聯的標籤。

新增標籤

您最多可以再新增 50 個標籤。

取消

上一步

建立使用者

Identity and Access Management (IAM)

You could download csv

擷取密碼

您可以查看和下載以下使用者的密碼，或透過電子郵件傳送登入 AWS 管理主控台的使用者指示。這是您檢視和下載此密碼的唯一機會。

主控台登入詳細資訊

電子郵件登入指示 [↗](#)

主控台登入 URL

使用者名稱

 weathermood

主控台密碼

 ***** [顯示](#)

下載 .csv 檔案

[返回使用者清單](#)

Clone projects

1. Client side code **weathermood_no_redux** project
 - Checkout branch **server-file** branch
2. Server side code **weathermood-server_no_redux** project
 - Checkout branch **file**
3. **npm install** both first to get all the packages

AWS EB CLI

- awsebcli: AWS Elastic Beanstalk Command Line Interface
- Under weathermood-server_no_redux
-> eb init

```
(base) sasaya@sasayadeMacBook-Pro weathermood-server_no_redux % eb init
```

```
Select a default region
```

```
1) us-east-1 : US East (N. Virginia)
2) us-west-1 : US West (N. California)
3) us-west-2 : US West (Oregon)
4) eu-west-1 : EU (Ireland)
5) eu-central-1 : EU (Frankfurt)
6) ap-south-1 : Asia Pacific (Mumbai)
7) ap-southeast-1 : Asia Pacific (Singapore)
8) ap-southeast-2 : Asia Pacific (Sydney)
9) ap-northeast-1 : Asia Pacific (Tokyo)
10) ap-northeast-2 : Asia Pacific (Seoul)
11) sa-east-1 : South America (Sao Paulo)
12) cn-north-1 : China (Beijing)
13) cn-northwest-1 : China (Ningxia)
14) us-east-2 : US East (Ohio)
15) ca-central-1 : Canada (Central)
16) eu-west-2 : EU (London)
17) eu-west-3 : EU (Paris)
18) eu-north-1 : EU (Stockholm)
19) eu-south-1 : EU (Milano)
20) ap-east-1 : Asia Pacific (Hong Kong)
21) me-south-1 : Middle East (Bahrain)
22) af-south-1 : Africa (Cape Town)
23) ap-southeast-3 : Asia Pacific (Jakarta)
24) ap-northeast-3 : Asia Pacific (Osaka)
(default is 3):
```

```
Select an application to use
```

```
1) weathermood-server_no_redux
2) [ Create new Application ]
(default is 1):
```

```
It appears you are using Docker. Is this correct?
(Y/n): y
```

```
Select a platform branch.
```

```
1) Docker running on 64bit Amazon Linux 2
2) ECS running on 64bit Amazon Linux 2
(default is 1):
```

```
Do you wish to continue with CodeCommit? (Y/n): n
```

```
Do you want to set up SSH for your instances?
```

```
(Y/n): n
```

AWS EB CLI

- Under weathermood-server_no_redux
- eb create
- Enter “weathermood-{groupId}” for DNS CNAME prefix
e.g., weathermood-1

```
(base) sasaya@sasayadeMacBook-Pro:~/weathermood-server_no_redux % eb create --single
Enter Environment Name
(default is weathermood-server-no-redux-dev):
Enter DNS CNAME prefix
(default is weathermood-server-no-redux-dev):

Would you like to enable Spot Fleet requests for this environment? (y/N): n
Creating application version archive "app-db12-230419_112614099262".
Uploading weathermood-server_no_redux/app-db12-230419_112614099262.zip to S3. This may take a while.
Upload Complete.
Environment details for: weathermood-server-no-redux-dev
  Application name: weathermood-server_no_redux
  Region: us-west-2
  Deployed Version: app-db12-230419_112614099262
  Environment ID: e-uz2sewkbqx
  Platform: arn:aws:elasticbeanstalk:us-west-2::platform/Docker running on 64bit Amazon Linux 2/3.5.6
  Tier: WebServer-Standard-1.0
  CNAME: weathermood-server-no-redux-dev.us-west-2.elasticbeanstalk.com
  Updated: 2023-04-19 03:26:47.053000+00:00
Printing Status:
2023-04-19 03:26:46 INFO createEnvironment is starting.
2023-04-19 03:26:47 INFO Using elasticbeanstalk-us-west-2-785490623664 as Amazon S3 storage bucket for environment data.
2023-04-19 03:27:07 INFO Created security group named: awseb-e-uz2sewkbqx-stack-AWSEBSecurityGroup-1SHIMI47Z7DF3
2023-04-19 03:27:22 INFO Created EIP: 52.40.193.8
2023-04-19 03:27:58 INFO Waiting for EC2 instances to launch. This may take a few minutes.
2023-04-19 03:28:05 INFO Instance deployment completed successfully.
2023-04-19 03:29:11 INFO Application available at weathermood-server-no-redux-dev.us-west-2.elasticbeanstalk.com.
2023-04-19 03:29:11 INFO Successfully launched environment: weathermood-server-no-redux-dev
```

Elastic Beanstalk

Click the application you just created and get the url

The screenshot shows the AWS Elastic Beanstalk console. The left sidebar has 'Elastic Beanstalk' selected, with sub-links for 'Environments', 'Applications', and 'Change history'. The main content area is titled 'All applications' and contains a search bar and a table of applications.

Application name	Environments	Date created	Last modified	ARN
weathermood-server_no_redux	weathermood-labdemo weathermood-server-no-redux-dev weathermood-server-no-	2023-04-19 11:08:38 UTC+0800	2023-04-19 11:08:38 UTC+0800	arn:aws:2:78549

Below the table, the console shows the details for the 'weathermood-server-no-redux-dev' environment. The environment name is highlighted with a blue box, and the URL 'weathermood-server-no-redux-dev.us-west-2.elasticbeanstalk.com' is also highlighted. A 'Refresh' button is visible in the top right corner of the details section.

Health
A green checkmark icon indicates the environment is healthy.

Running version
app-db12-230419_112614099262
[Upload and deploy](#)

Platform
The Docker logo is shown, indicating the platform used for the application.

Project code

Go to project weathermood_no_redux -> src -> api -> post.js
Paste url to postBaseUrl

```
import axios from 'axios';

// Develop server URL
const postBaseUrl = 'http://localhost:8080/api';

// Staging server URL
// const postBaseUrl = 'http://weathermood-staging.us-west-2.elasticbeanstalk.com/api';

// Production server URL
// const postBaseUrl = {Your URL};
const postBaseUrl = 'http://weathermood-server-no-redux-dev.us-west-2.elasticbeanstalk.com/api';

export function listPosts(searchText = '') {
  let url = `${postBaseUrl}/posts`;
  if (searchText) url += `?searchText=${searchText}`;

  console.log(`Making GET request to: ${url}`);

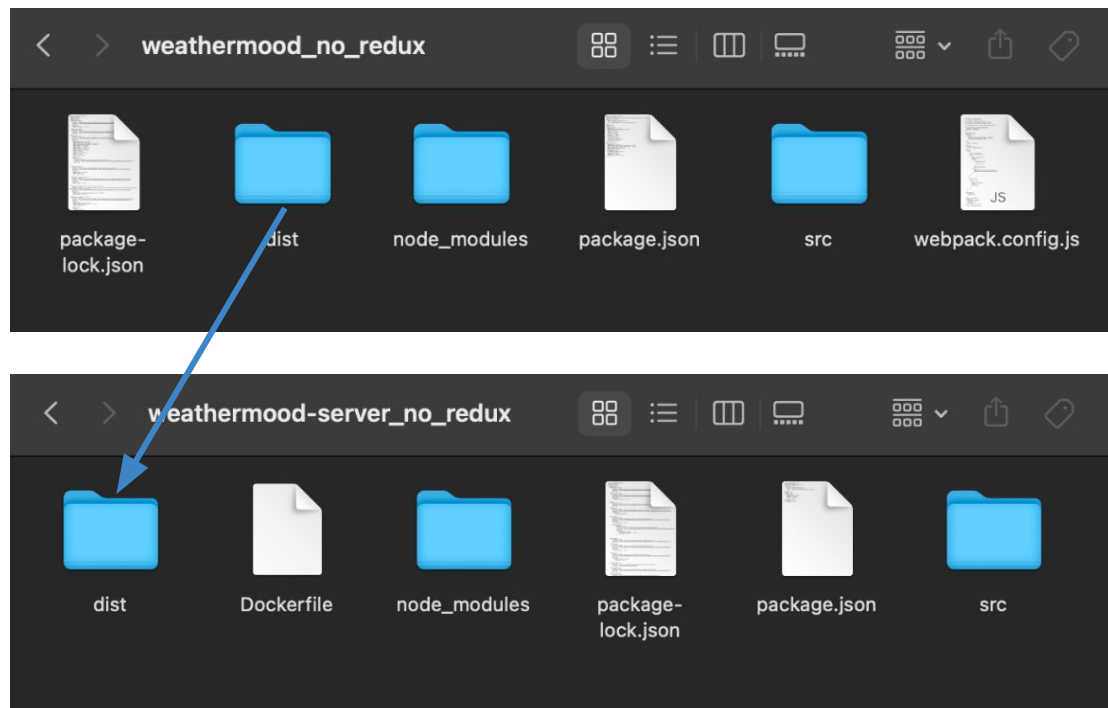
  return axios.get(url).then(function (res) {
    if (res.status !== 200)
      throw new Error(`Unexpected response code: ${res.status}`);

    return res.data;
  });
}
```

Rebuild weathermood_no_redux using **npm run build**

Project code

Copy weathermood_no_redux **dist** folder to
weathermood-server_no_redux



Project code

Under weathermood-server_no_redux -> eb deploy <env>
It will deploy committed change only. Therefore, you have to commit before you deploy.

```
→ weathermood-server git:(file) X git add .
→ weathermood-server git:(file) X git commit -m "for lab demo"
[file 0312b98] for lab demo
4 files changed, 5 insertions(+), 6 deletions(-)
create mode 100644 dist/.DS_Store
rewrite dist/index.bundle.js (62%)
rewrite dist/index.bundle.js.map (60%)
→ weathermood-server git:(file) eb deploy
```

```
→ weathermood-server git:(file) eb deploy
Creating application version archive "app-c335-190427_202417".
Uploading weathermood-server/app-c335-190427_202417.zip to S3. This may take a while.
Upload Complete.
2019-04-27 12:24:30 INFO Environment update is starting.
2019-04-27 12:24:33 INFO Deploying new version to instance(s).
2019-04-27 12:24:40 INFO Successfully pulled node:6.10
2019-04-27 12:24:40 INFO Successfully built aws_beanstalk/staging-app
2019-04-27 12:24:49 INFO Docker container 66d894332056 is running aws_beanstalk/current-app.
2019-04-27 12:24:56 INFO New application version was deployed to running EC2 instances.
2019-04-27 12:24:56 INFO Environment update completed successfully.

Alert: An update to the EB CLI is available. Run "pip install --upgrade awsebcli" to get the latest version.
```

