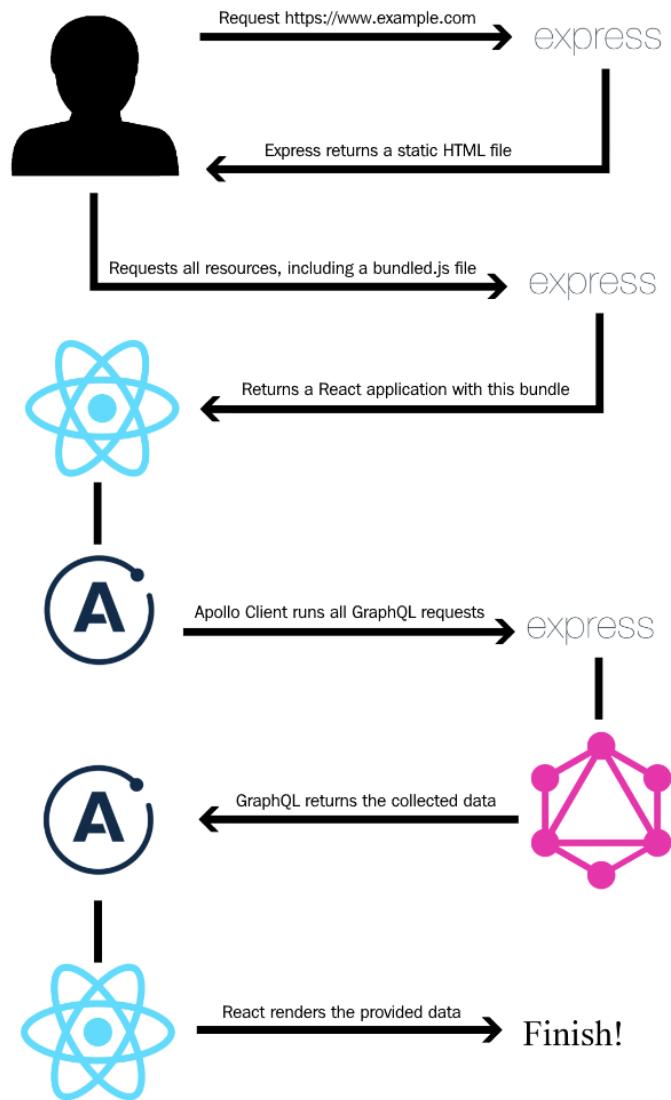


Chapter 1: Preparing Your Development Environment



```
This utility will walk you through creating a package.json file.  
It only covers the most common items, and tries to guess sensible defaults.
```

```
See `npm help json` for definitive documentation on these fields  
and exactly what they do.
```

```
Use `npm install <pkg>` afterwards to install a package and  
save it as a dependency in the package.json file.
```

```
Press ^C at any time to quit.
```

```
package name: (graphbook)
```

```
version: (1.0.0) 0.0.1
```

```
description:
```

```
entry point: (index.js)
```

```
test command:
```

```
git repository:
```

```
keywords:
```

```
author:
```

```
license: (ISC)
```

```
About to write to C:\Users\sebig\Desktop\graphbook\package.json:
```

```
{  
  "name": "graphbook",  
  "version": "0.0.1",  
  "description": "",  
  "main": "index.js",  
  "scripts": {  
    "test": "echo \\\"Error: no test specified\\\" && exit 1"  
  },  
  "author": "",  
  "license": "ISC"  
}
```



Test User
Lorem ipsum



Test User 2
Lorem ipsum



TestUser2

Lorem ipsum 1



TestUser

Lorem ipsum 2

 Components  Profiler

▼ App
 ▼ HelmetWrapper
 ▼ NullComponent **SideEffect**
 NullComponent

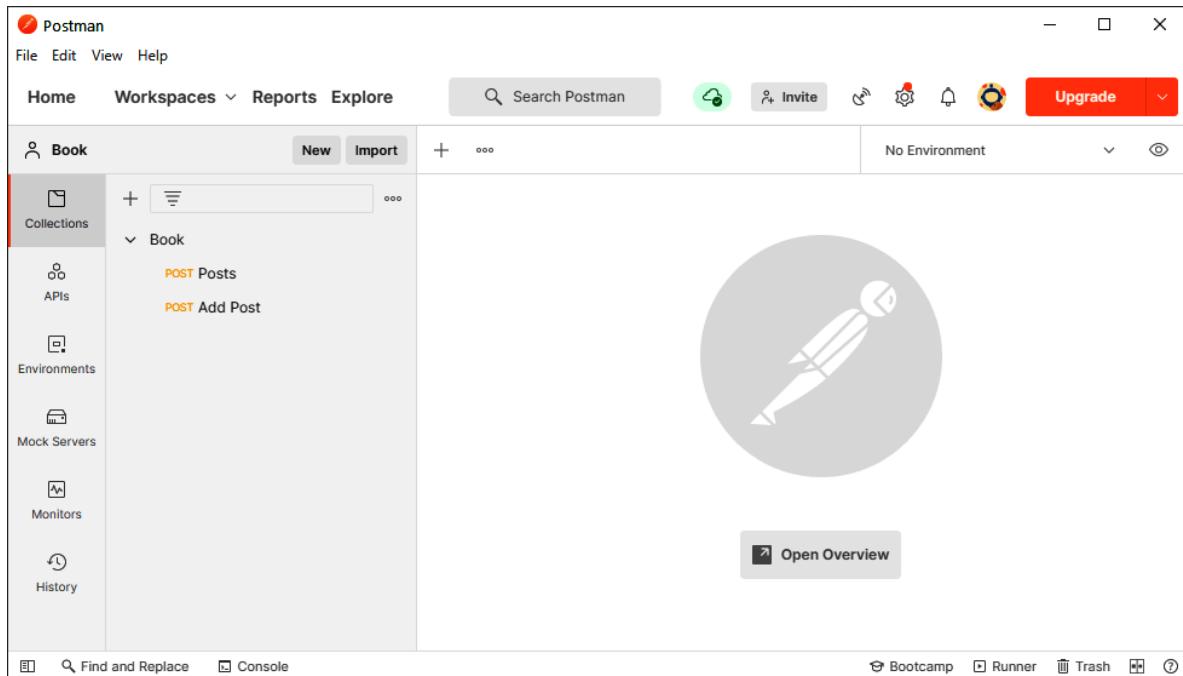
App    

props
 new entry: ""

hooks 
 ▶ State: `[{}, {}]`
 State: ""

rendered by
 `createLegacyRoot()`
 react-dom@17.0.1

Chapter 2: Setting Up GraphQL with Express.js

A screenshot of the Postman application interface showing a POST request to 'http://localhost:8000/graphql'. The 'Body' tab is selected, displaying the following GraphQL mutation:

```
1
2   "operationName": null,
3   "query": "mutation addPost($post: PostInput!, $user: UserInput!) {
4     addPost(post: $post, user: $user) {
5       id
6       text
7       user {
8         username
9         avatar
10      }
11    }
12  }"
13 }
```

The 'Send' button is highlighted in blue. The 'Headers' section contains '(11)' entries. The 'Response' section is currently empty. The bottom navigation bar includes 'Find and Replace', 'Console', 'Bootcamp', 'Runner', 'Trash', and a help icon.

Postman

File Edit View Help

Home Workspaces Reports Explore Search Postman

New Import GET Untitled Request POST Add Post X + ... No Environment

Collections APIs Environments Mock Servers Monitors History

+

Book / Add Post

POST http://localhost:8000/graphql

Params Auth Headers (11) Body ● Pre-req. Tests Settings Cookies

GraphQL No schema C

QUERY

```
1 mutation addPost($post : PostInput!, $user: UserInput!) {
2   addPost(post : $post, user: $user) {
3     id
4     text
5     user {
6       username
7       avatar
8     }
9   }
10 }
```

GRAPHQL VARIABLES ⓘ

```
1 "post": {
2   "text": "You just added a
3   post."
4 },
5 "user": {
6   "avatar": "/uploads/
7   avatar3.png",
8   "username": "Fake User"
9 }
```

Response

Find and Replace Console Bootcamp Runner Trash

The screenshot shows the Postman application interface. On the left, there's a sidebar with icons for Collections, APIs, Environments, Mock Servers, Monitors, and History. The main area shows a 'Book / Add Post' collection with a 'POST Add Post' request. The request is set to 'POST' and points to 'http://localhost:8000/graphql'. The 'Body' tab is selected, showing a GraphQL mutation to add a post. The mutation code is:

```
1 mutation addPost($post : PostInput!, $user: UserInput!) {
2   addPost(post : $post, user: $user) {
3     id
4     text
5     user {
6       username
7       avatar
8     }
9   }
10 }
```

Below the mutation, under 'GRAPHQL VARIABLES ⓘ', is the response JSON:

```
1 "post": {
2   "text": "You just added a
3   post."
4 },
5 "user": {
6   "avatar": "/uploads/
7   avatar3.png",
8   "username": "Fake User"
9 }
```

The 'Response' section below shows a placeholder icon of a rocket launching.

Chapter 3: Connecting to the Database

The screenshot shows the phpMyAdmin configuration interface. On the left is a sidebar with a tree view of databases: Neu, information_schema, mysql, performance_schema, phpmyadmin, and sys. The main area has several tabs: Allgemeine Einstellungen, Anzeige-Einstellungen, Datenbank-Server, Webserver, and phpMyAdmin. The 'Datenbank-Server' tab displays system information:

- Server: Localhost via UNIX socket
- Server-Typ: MySQL
- Server-Version: 5.7.23-0ubuntu0.18.04.1 - (Ubuntu)
- Protokoll-Version: 10
- Benutzer: devuser@localhost
- Server-Zeichensatz: UTF-8 Unicode (utf8)

The 'Webserver' tab shows:

- Apache/2.4.29 (Ubuntu)
- Datenbank-Client Version: libmysql - mysqlnd 5.0.12-dev - 20150407 - \$Id: 38fea24f2847fa7519001be390c98ae0acaef387 \$
- PHP-Erweiterung: mysqli curl mbstring
- PHP-Version: 7.2.7-0ubuntu0.18.04.2

The 'phpMyAdmin' tab lists:

- Versionsinformationen: 4.6.6deb5
- Dokumentation
- Offizielle Homepage
- Mitmachen
- Unterstützung erhalten
- Liste der Änderungen
- Lizenz

Databases

The screenshot shows the phpMyAdmin database list interface. At the top is a 'Create database' button with a question mark icon. Below it are fields for 'Database name' and 'Collation'. A 'Create' button is to the right. The main area is a table listing databases:

	Database	Collation	Action
<input type="checkbox"/>	information_schema	utf8_general_ci	Check privileges
<input type="checkbox"/>	mysql	latin1_swedish_ci	Check privileges
<input type="checkbox"/>	performance_schema	utf8_general_ci	Check privileges
<input type="checkbox"/>	phpmyadmin	latin1_swedish_ci	Check privileges
<input type="checkbox"/>	sys	utf8_general_ci	Check privileges
Total: 5			latin1_swedish_ci

At the bottom are navigation icons: a back arrow, a checkbox labeled 'Check all', and a dropdown labeled 'With selected: Drop'.

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input type="checkbox"/>	1	 id	int(11)			No	None		AUTO_INCREMENT
<input type="checkbox"/>	2	text	text	utf8_general_ci		Yes	NULL		
<input type="checkbox"/>	3	createdAt	datetime			No	None		
<input type="checkbox"/>	4	updatedAt	datetime			No	None		

← T → name

 Edit  Copy  Delete 20210415201849-create-post.js

	id	text	createdAt	updatedAt
<input type="checkbox"/>	1	1 Lorem ipsum 1	2018-08-13 15:28:40	2018-08-13 15:28:40
<input type="checkbox"/>	2	2 Lorem ipsum 2	2018-08-13 15:28:40	2018-08-13 15:28:40

Actions Constraint properties

Column		Foreign key constraint (INNODB)			
Database	Table	Column	Column	Table	Column
 Drop	fk_user_id	ON DELETE	CASCADE	ON UPDATE	CASCADE
 user	userId		graphbook_dev	Users	id
 + Add column					
 Constraint name		ON DELETE	RESTRICT	ON UPDATE	RESTRICT
 graphbook_dev					
 + Add column					

	id	avatar	username	createdAt	updatedAt
<input type="checkbox"/>	1	/uploads/avatar1.png	Test User	2018-08-13 17:04:15	2018-08-13 17:04:15
<input type="checkbox"/>	2	/uploads/avatar2.png	Test User 2	2018-08-13 17:04:15	2018-08-13 17:04:15

	id	text	createdAt	updatedAt	userId
<input type="checkbox"/>	1	1 Lorem ipsum 1	2018-08-14 11:08:28	2018-08-14 11:08:28	1
<input type="checkbox"/>	2	2 Lorem ipsum 2	2018-08-14 11:08:28	2018-08-14 11:08:28	2
<input type="checkbox"/>	3	You just added a post.	2018-08-14 11:08:46	2018-08-14 11:08:46	1

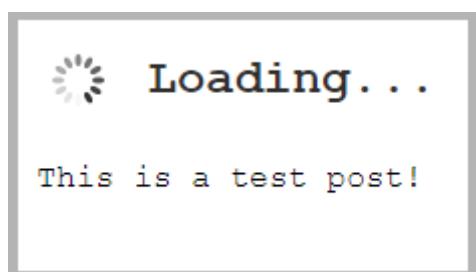
Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> Chats	 Browse  Structure  Search  Insert  Empty  Drop	0	InnoDB	utf8_general_ci	16 Kib	-
<input type="checkbox"/> Posts	 Browse  Structure  Search  Insert  Empty  Drop	3	InnoDB	utf8_general_ci	32 Kib	-
<input type="checkbox"/> SequelizeMeta	 Browse  Structure  Search  Insert  Empty  Drop	5	InnoDB	utf8_unicode_ci	32 Kib	-
<input type="checkbox"/> Users	 Browse  Structure  Search  Insert  Empty  Drop	2	InnoDB	utf8_general_ci	16 Kib	-
<input type="checkbox"/> users_chats	 Browse  Structure  Search  Insert  Empty  Drop	0	InnoDB	utf8_general_ci	48 Kib	-
5 tables	Sum	10	InnoDB	utf8_general_ci	144 Kib	0 B

Actions Constraint properties

Column		Foreign key constraint (INNODB)			
Database	Table	Column	Column	Table	Column
 Drop	users_chats_ibfk_1	ON DELETE	RESTRICT	ON UPDATE	RESTRICT
 user	userId		graphbook_dev	Users	id
 + Add column					
 Drop	users_chats_ibfk_2	ON DELETE	RESTRICT	ON UPDATE	RESTRICT
 chat	chatId		graphbook_dev	Chats	id
 + Add column					

Chapter 4: Hooking Apollo into React

```
▼ {data: {...}, loading: false, networkStatus: 7} ⓘ
  ▼ data:
    ▼ posts: Array(2)
      ▼ 0:
        id: 1
        text: "Lorem ipsum 1"
      ▼ user:
        avatar: "/uploads/avatar1.png"
        username: "TestUser"
        __typename: "User"
        ► __proto__: Object
        __typename: "Post"
        ► __proto__: Object
      ▼ 1:
        id: 2
        text: "Lorem ipsum 2"
      ▼ user:
        avatar: "/uploads/avatar2.png"
        username: "TestUser2"
        __typename: "User"
        ► __proto__: Object
        __typename: "Post"
        ► __proto__: Object
        length: 2
        ► __proto__: Array(0)
        ► __proto__: Object
      loading: false
      networkStatus: 7
    ► __proto__: Object
```



Write your custom post!

Submit

 **TestUser**

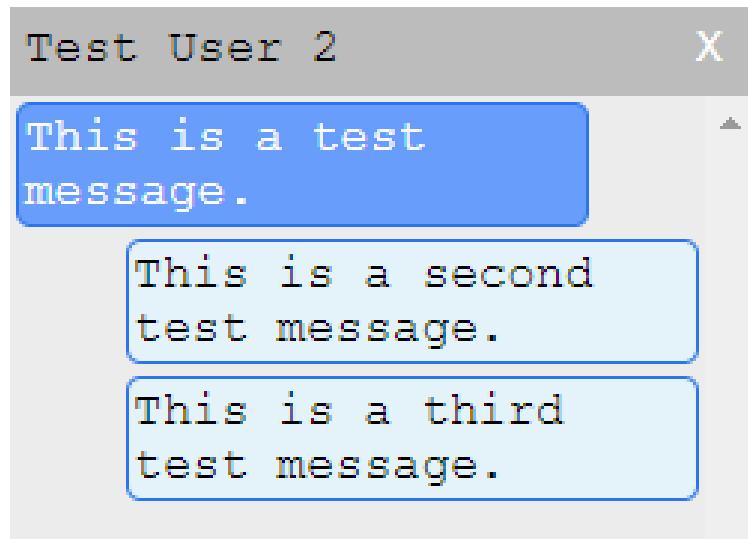
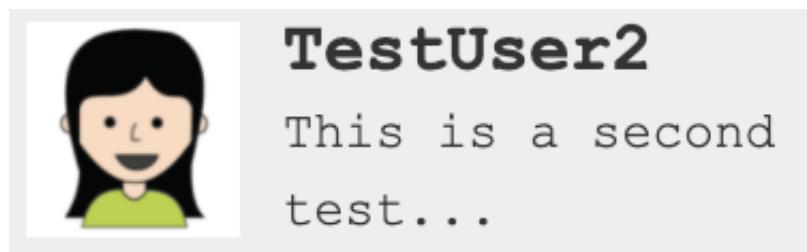
This is a test post!

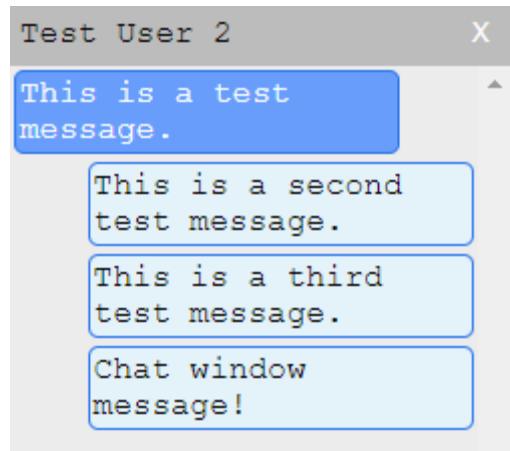
 **TestUser2**

Lorem ipsum 1

 **TestUser**

Lorem ipsum 2





Screenshot of the GraphiQL interface showing a GraphQL query and its results.

GraphQL Query:

```

1  query posts {
2    posts {
3      id
4      text
5      user {
6        avatar
7        username
8      }
9    }
10  }

```

Result Data:

```

{
  "data": {
    "posts": [
      {
        "id": 1,
        "text": "Lorem ipsum 1",
        "user": {
          "avatar": "/uploads/avatar1.png",
          "username": "TestUser",
          "__typename": "User"
        },
        "__typename": "Post"
      },
      {
        "id": 2,
        "text": "Lorem ipsum 2",
        "user": {
          "avatar": "/uploads/avatar2.png",
          "username": "TestUser2",
          "__typename": "User"
        },
        "__typename": "Post"
      }
    ],
    "loading": false,
    "networkStatus": 7
  }
}

```

Screenshot of the GraphiQL interface showing a GraphQL query named 'postsFeed'.

ACTIVE QUERIES (4)

- postsFeed
- chats
- posts
- posts

Query String:

```

query postsFeed($page: Int, $limit: Int) {
  postsFeed(page: $page, limit: $limit) {
    posts {
      id
      text
      user {
        avatar
        username
        __typename
      }
      __typename
    }
    __typename
  }
}

```

Variables:

- root: {} 2 keys
- page: 0
- limit: 10

The screenshot shows the Apollo DevTools interface with the "CACHE" tab selected. The left sidebar lists various cache entries, with "ROOT_QUERY" highlighted in blue. The main panel displays the structure of the "ROOT_QUERY" cache entry.

Cache Entries:

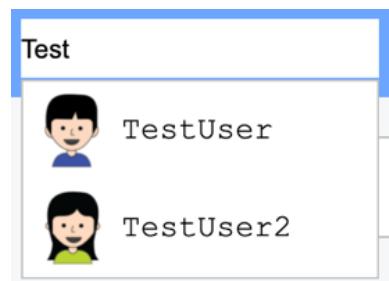
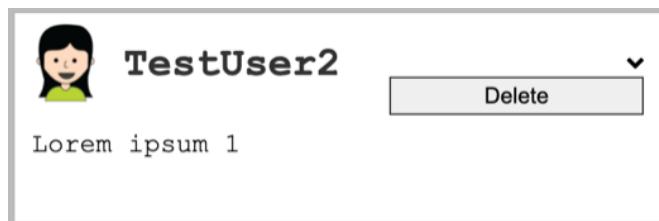
- Chat:1
- Post:1
- Post:2
- Post:27
- Post:28
- Post:29
- Post:30
- Post:31
- Post:32
- Post:33
- Post:34
- Post:35

ROOT_QUERY Cache ID:

```
ROOT_QUERY
  __typename: "Query"
    postsFeed({ "limit": 10, "page": 0 }): []
      2 items
        chats: [] 1 item
```

Chapter 5: Reusable React Components and React Hooks

GraphQL error: connect ETIMEDOUT



Graphbook Style Guide

Filter by name

- Error
- Fontawesome
- Dropdown
- Loading
- AddPostMutation
- DeletePostMutation
- UpdatePostMutation
- Content
- FeedList
- PostForm
- Header
- Post
- PostsFeedQuery

Error

[src/client/components/error.js](#)

[Add examples to this component](#)

Fontawesome

[src/client/components/fontawesome.js](#)

[Add examples to this component](#)

Dropdown

[src/client/components/helpers/dropdown.js](#)

[Add examples to this component](#)

Loading

[src/client/components/loading.js](#)

[Add examples to this component](#)

AddPostMutation

[src/client/components/mutations/addPost.js](#)

[Add examples to this component](#)

Post

[src\client\components\post\index.js](#)

PROPS & METHODS

Prop name	Type	Default	Description
post	object	Required	Object containing the complete post.

Post

[src\client\components\post\index.js](#) 

PROPS & METHODS

Prop name	Type	Default	Description
<code>post</code>	<code>shape</code>	Required	Object containing the complete post. <code>id: number</code> — Required <code>text: string</code> — Required <code>user: shape</code> — Required

[Add examples to this component](#)

Post example:

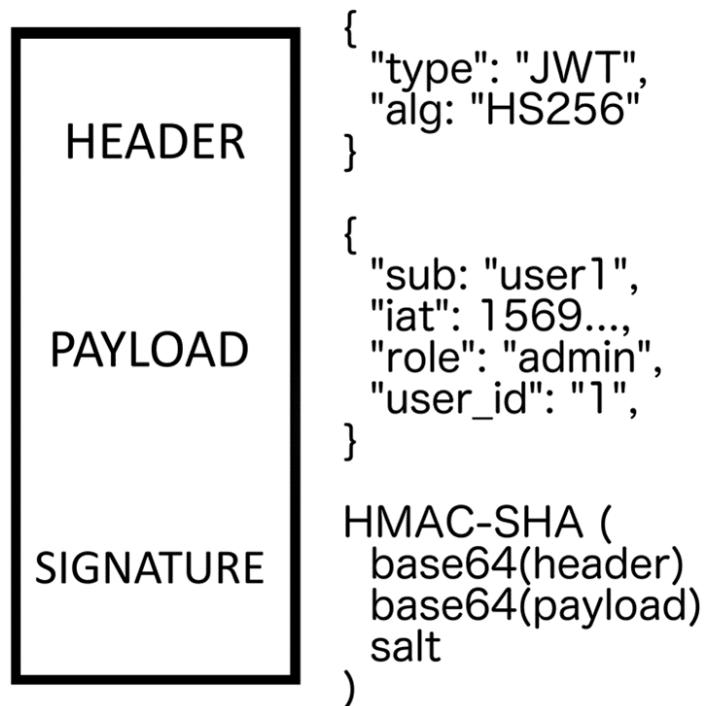


[VIEW CODE](#)



```
const post = {  
  id: 3,  
  text: "This is a test post!",  
  user: {  
    avatar: "/uploads/avatar1.png",  
    username: "Test User"  
  }  
};  
  
<Post key={post.id} post={post} />
```

Chapter 6: Authentication with Apollo and React



Email

Password

Login

GraphQL error: You need to be logged in.

Chapter 7: Handling Image Uploads

Sales Services ▾ [Option+S] Sebastian Grebe ▾ Frankfurt ▾ Support ▾

AWS Management Console

AWS services

- ▶ Recently visited services
- ▶ All services

Build a solution
Get started with simple wizards and automated workflows.

Launch a virtual machine With EC2 2-3 minutes 	Build a web app With Elastic Beanstalk 6 minutes
Build using virtual servers With Lightsail 1-2 minutes 	Register a domain With Route 53 3 minutes

▶ See more

Stay connected to your AWS resources on-the-go

AWS Console Mobile App now supports four additional regions. Download the AWS Console Mobile App to your iOS or Android mobile device. [Learn more](#)

Explore AWS

Amazon DocumentDB (with MongoDB compatibility)
Build high performance apps with a flexible, fully managed JSON database. [Learn more](#)

AWS Training
Free digital courses to help you develop your skills. [Learn more](#)

Free AWS Training
Advance your career with AWS Cloud Practitioner Essentials—a free, six-hour, foundational course. [Learn more](#)

Get Started Faster on ML

Feedback English (US) ▾ © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Services ▾ [Option+S] Sebastian Grebe ▾ Global ▾ Support ▾

Amazon S3

Buckets

- Access Points
- Object Lambda Access Points
- Batch Operations
- Access analyzer for S3

Block Public Access settings for this account

▼ Storage Lens

- Dashboards
- AWS Organizations settings

Feature spotlight

▶ AWS Marketplace for S3

Amazon S3

Account snapshot
Storage lens provides visibility into storage usage and activity trends. [Learn more](#) [View Storage Lens dashboard](#)

Buckets (0) Info		Copy ARN	Empty	Delete	Create bucket
Buckets are containers for data stored in S3. Learn more					
<input type="text" value="Find buckets by name"/> 1					
Name	AWS Region	Access	Creation date		
No buckets You don't have any buckets.					
Create bucket					

Feedback English (US) ▾ © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

AWS Services ▾ Search for services, features, marketplace products, and docs [Option+S] Sebastian Grebe ▾ Global ▾ Support ▾

Amazon S3 > Create bucket

Create bucket Info

Buckets are containers for data stored in S3. [Learn more](#)

General configuration

Bucket name Bucket name must be unique and must not contain spaces or uppercase letters. [See rules for bucket naming](#)

AWS Region

Copy settings from existing bucket - *optional*
Only the bucket settings in the following configuration are copied.

[Choose bucket](#)

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- Block public access to buckets and objects granted through new access control lists (ACLs)**
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- Block public access to buckets and objects granted through any access control lists (ACLs)**

Feedback English (US) © 2008 - 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use Cookie preferences

Block Public Access settings for this bucket

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

Block all public access

Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

Block public access to buckets and objects granted through new access control lists (ACLs)

S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

Block public access to buckets and objects granted through any access control lists (ACLs)

S3 will ignore all ACLs that grant public access to buckets and objects.

Block public access to buckets and objects granted through new public bucket or access point policies

S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

Block public and cross-account access to buckets and objects through any public bucket or access point policies

S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.



Turning off block all public access might result in this bucket and the objects within becoming public

AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

I acknowledge that the current settings might result in this bucket and the objects within becoming public.

You are accessing the security credentials page for your AWS account. The account credentials provide unlimited access to your AWS resources.

To help secure your account, follow an [AWS best practice](#) by creating and using AWS Identity and Access Management (IAM) users with limited permissions.

[Continue to Security Credentials](#)

[Get Started with IAM Users](#)

Don't show me this message again

Your Security Credentials

Use this page to manage the credentials for your AWS account. To manage credentials for AWS Identity and Access Management (IAM) users, use the [IAM Console](#).

To learn more about the types of AWS credentials and how they're used, see [AWS Security Credentials](#) in AWS General Reference.

- ▲ Password
- ▲ Multi-factor authentication (MFA)
- ▼ Access keys (access key ID and secret access key)

Use access keys to make programmatic calls to AWS from the AWS CLI, Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time.

For your protection, you should never share your secret keys with anyone. As a best practice, we recommend frequent key rotation.

If you lose or forget your secret key, you cannot retrieve it. Instead, create a new access key and make the old key inactive. [Learn more](#)

Created	Access Key ID	Last Used	Last Used Region	Last Used Service	Status	Actions
Jul 5th 2021	AKIAXX5AF7VIAE4D4WN	2021-07-10 21:21 UTC+0400	eu-central-1	s3	Active	Make Inactive Delete

[Create New Access Key](#)

Root user access keys provide unrestricted access to your entire AWS account. If you need long-term access keys, we recommend creating a new IAM user with limited permissions and generating access keys for that user instead. [Learn more](#)

- ▲ CloudFront key pairs
- ▲ X.509 certificate
- ▲ Account identifiers

Create Access Key



Your access key (access key ID and secret access key) has been created successfully.

Download your key file now, which contains your new access key ID and secret access key. If you do not download the key file now, you will not be able to retrieve your secret access key again.

To help protect your security, store your secret access key securely and do not share it.

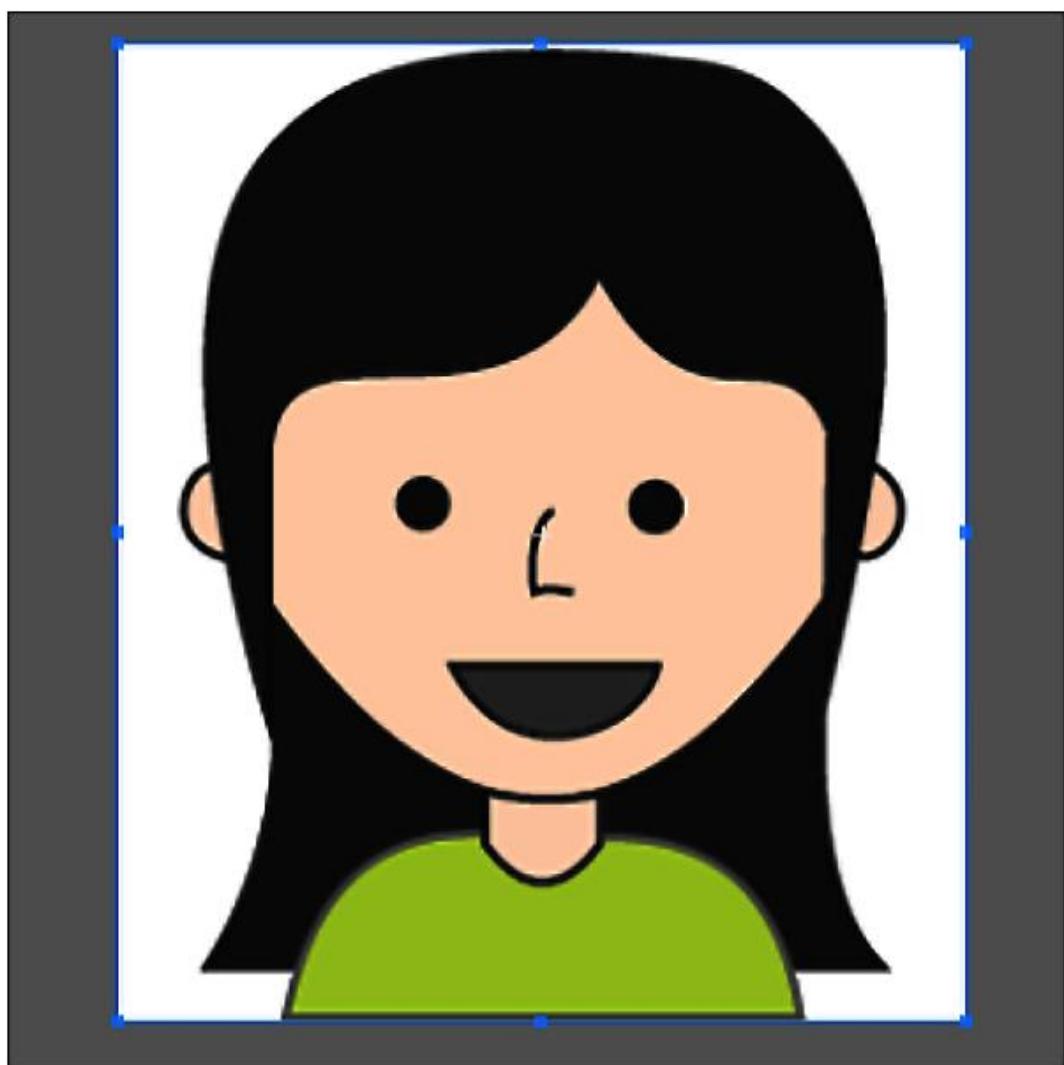
[▼ Hide Access Key](#)

Access Key ID: AKIAXX5AF7VAPKRJTUXW
Secret Access Key: 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

[Download Key File](#) [Close](#)

Drag 'n' drop some files here, or click
to select files

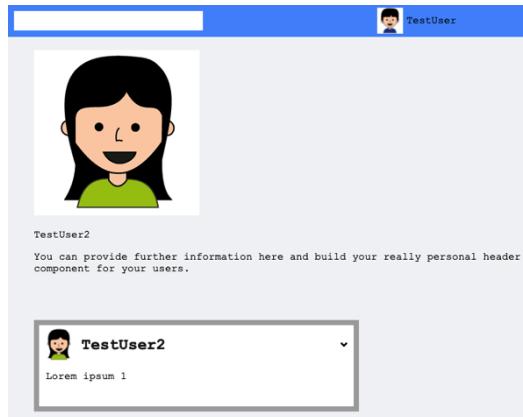
Save



Change image

Save

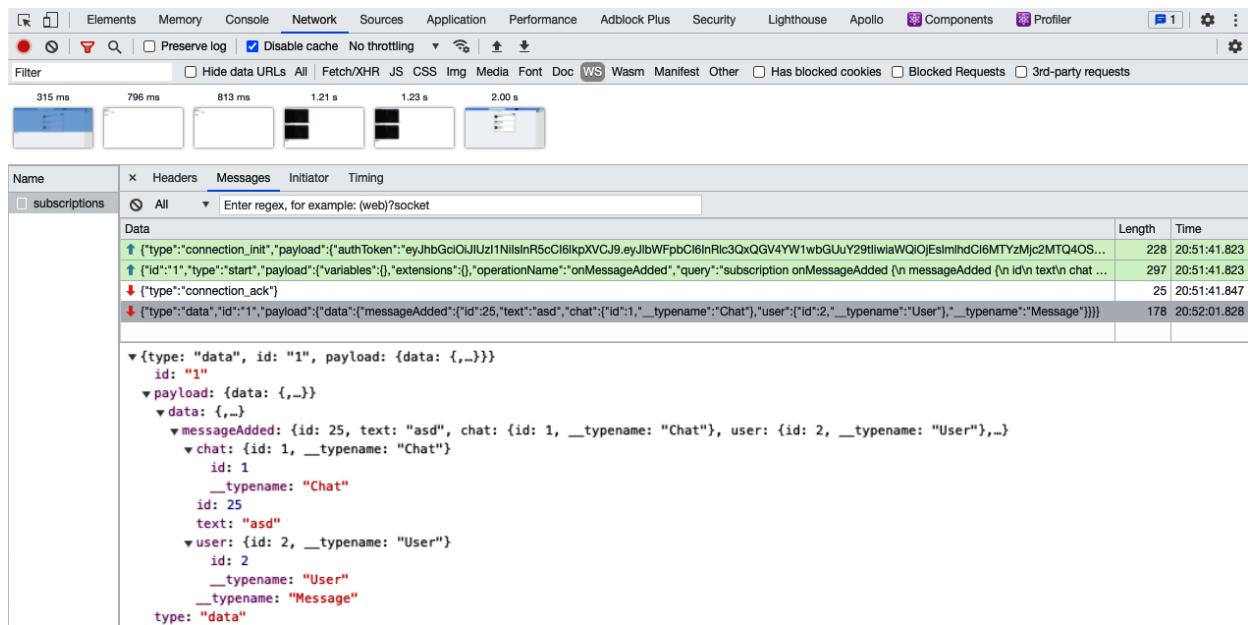
Chapter 8: Routing in React



Chapter 9: Implementing Server-Side Rendering

No Images...

Chapter 10: Real-Time Subscriptions



The screenshot shows the Network tab in the Chrome DevTools. The 'Messages' tab is selected. A log entry for a 'subscriptions' socket connection is expanded, showing a detailed message structure. The message payload includes a 'connection_init' with an auth token, a 'variables' object, and an 'operationName' of 'onMessageAdded'. It also contains a 'messageAdded' event with an ID of 25, text 'asd', a 'Chat' type, and two users: one with ID 1 and another with ID 2.

Name	Headers	Messages	Initiator	Timing
subscriptions	All	Enter regex, for example: (web)?socket		
Data				
↑ ("type": "connection_init", "payload": {"authToken": "eyJhbGciOiJIUzI1NlslnR5cCl6IkpxVCJ9eyJlbWFpbC16InRlc3QxQGV4YW1wbGUuY29tliwiaWQiOEsImIhdCl6MTYzMjC2MTQ4OS..."}, "variables": {}, "extensions": {}, "operationName": "onMessageAdded", "query": "subscription onMessageAdded { messageAdded { id text } }")	228	20:51:41.823		
↑ ("id": "1", "type": "start", "payload": {})	297	20:51:41.823		
↓ ("type": "connection_ack")	25	20:51:41.847		
↓ ("type": "data", "id": "1", "payload": {"data": {"messageAdded": {"id": 25, "text": "asd", "chat": {"id": 1, "typename": "Chat"}, "user": {"id": 2, "typename": "User"}, "typename": "Message"}}})	178	20:52:01.828		

New message

Chapter 11: Writing Tests for React and Node.js

```
Graphbook application test
  1) renders and serves the index page

0 passing (27ms)
1 failing

1) Graphbook application test
   renders and serves the index page:
     Uncaught AssertionError: expected [Error: connect ECONNREFUSED 127.0.0.1:8000] to not exist
      at Object.exist (node_modules/chai/lib/chai/interface/should.js:208:38)
      at Request._callback (test/app.test.js:43:24)
      at self.callback (node_modules/request/request.js:185:22)
      at Request.onRequestError (node_modules/request/request.js:877:8)
      at Socket.socketErrorListener (_http_client.js:406:9)
      at emitErrorNT (internal/streams/destroy.js:92:8)
      at emitErrorAndCloseNT (internal/streams/destroy.js:60:3)
      at processTicksAndRejections (internal/process/task_queues.js:80:21)
```

Graphbook application test

✓ renders and serves the index page (105ms)

1 passing (110ms)

Graphbook application test

✓ renders and serves the index page (71ms)

404

✓ redirects the user when not matching path is found

authentication

✓ redirects the user when not logged in

✓ allows the user to sign up (118ms)

✓ allows the user to query all chats

Chapter 12: Continuous Deployment with CircleCI and AWS

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
sgrebe/graphbook	latest	fe30bceb0268	27 minutes ago	1.22GB
node	10	75a3a4428e1d	3 days ago	894MB

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
08499322a998	sgrebe/graphbook	"npm run server"	4 seconds ago	Up 3 seconds	0.0.0.0:8000->8000/tcp	dreamy_knuth

Amazon RDS

[Dashboard](#)

Instances

Clusters

Performance Insights

Snapshots

Automated backups

Reserved instances

Subnet groups

Parameter groups

Option groups

Events

Event subscriptions

Recommendations 1

Resources

You are using the following Amazon RDS resources in the EU (Frankfurt) region (used/quota)

DB Instances (1/40)	Parameter groups (1)
Allocated storage (20.00 GB/100.00 TB)	Default (1)
Click here to increase DB instances limit	Custom (0/100)
Reserved instances (0/40)	Option groups (1)
Snapshots (30)	Default (1)
Manual (0/100)	Custom (0/20)
Automated (6)	Subnet groups (1/50)
Recent events (4)	Supported platforms VPC
Event subscriptions (0/20)	Default network vpc-58231f33

Create database

Amazon Relational Database Service (RDS) makes it easy to set up, operate, and scale a relational database in the cloud.

[Restore from S3](#) [Create database](#)

Note: your DB instances will launch in the EU (Frankfurt) region

Service health

View service health dashboard

Current status Details

Amazon Relational Database Service (Frankfurt) Service is operating normally

Additional information

Getting started with RDS

Overview and features

Documentation

Articles and tutorials

Data import guide for MySQL

Data import guide for Oracle

Data import guide for SQL Server

New RDS feature announcements

Pricing

Forums

Database Preview Environment

Get early access to new DB engine versions, before they're generally available. The RDS database preview environment lets you work with upcoming beta, release candidate, and early production versions of PostgreSQL engines. Preview environment instances are fully functional, so you can easily test new features and functionality with your applications. [Info](#)

Preview PostgreSQL in US EAST (Ohio)

RDS > Create database

Create database

Choose a database creation method [Info](#)

Standard create You set all of the configuration options, including ones for availability, security, backups, and maintenance.

Easy create Use recommended best-practice configurations. Some configuration options can be changed after the database is created.

Engine options

Engine type [Info](#)

<input type="radio"/> Amazon Aurora 	<input checked="" type="radio"/> MySQL 	<input type="radio"/> MariaDB 
<input type="radio"/> PostgreSQL 	<input type="radio"/> Oracle 	<input type="radio"/> Microsoft SQL Server 

Templates

Choose a sample template to meet your use case.

Production

Use defaults for high availability and fast, consistent performance.

Dev/Test

This instance is intended for development use outside of a production environment.

Free tier

Use RDS Free Tier to develop new applications, test existing applications, or gain hands-on experience with Amazon RDS.

[Info](#)

Settings

DB instance identifier [Info](#)

Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

graphbook

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

▼ Credentials Settings

Master username [Info](#)

Type a login ID for the master user of your DB instance.

admin

1 to 16 alphanumeric characters. First character must be a letter

Auto generate a password

Amazon RDS can generate a password for you, or you can specify your own password

Master password [Info](#)

.....

Constraints: At least 8 printable ASCII characters. Can't contain any of the following: / (slash), '(single quote), "(double quote) and @ (at sign).

Confirm password [Info](#)

.....

DB instance class

DB instance class [Info](#)

Standard classes (includes m classes)

Memory optimized classes (includes r and x classes)

Burstable classes (includes t classes)

db.t2.micro

1 vCPUs 1 GiB RAM Not EBS Optimized



Include previous generation classes

Connectivity



Virtual private cloud (VPC) [Info](#)

VPC that defines the virtual networking environment for this DB instance.

[Create new VPC](#)



Only VPCs with a corresponding DB subnet group are listed.

After a database is created, you can't change its VPC.

Subnet group [Info](#)

DB subnet group that defines which subnets and IP ranges the DB instance can use in the VPC you selected.

[Create new DB Subnet Group](#)



Public access [Info](#)

Yes

Amazon EC2 instances and devices outside the VPC can connect to your database. Choose one or more VPC security groups that specify which EC2 instances and devices inside the VPC can connect to the database.

No

RDS will not assign a public IP address to the database. Only Amazon EC2 instances and devices inside the VPC can connect to your database.

VPC security group

Choose a VPC security group to allow access to your database. Ensure that the security group rules allow the appropriate incoming traffic.

[Choose existing](#)

Choose existing VPC security groups

[Create new](#)

Create new VPC security group

New VPC security group name

graphbook-src-grp

Availability Zone [Info](#)

eu-central-1a



▼ Additional configuration

Database options, encryption enabled, backup enabled, backtrack disabled, Performance Insights enabled, Enhanced Monitoring enabled, maintenance, CloudWatch Logs, delete protection disabled

Database options

Initial database name [Info](#)

graphbook

If you do not specify a database name, Amazon RDS does not create a database.

Security group rules (2)



Filter security group rules

< 1 >

Security group	Type	Rule
graphbook-src-grp (sg-0a08126acbd582acb)	CIDR/IP - Inbound	92.97.27.239/32
graphbook-src-grp (sg-0a08126acbd582acb)	CIDR/IP - Outbound	0.0.0.0/0

The screenshot shows the CircleCI web interface. On the left is a sidebar with a user profile (Seb155, Sebastian Grebe), navigation links (Dashboard, Projects, Insights, Organization Settings, Plan), and status indicators (Status: OPERATIONAL, Docs, Orbs, Help). The main area has a header "Projects" with a help card about following projects. It includes a search bar for "Repo name", a "Follow All" button, and a "Learn" sidebar with "Projects on CircleCI" and "Popular docs". Below is a list of projects: Fullstack-React-and-GraphQL, Hands-on-Full-Stack-Web-Development-with-GraphQL-and-React-2nd-Edition, nmgn, hcaptcha, plantguru-dashboard, balenax, whatkindofdeveloperareyou, and hcaptcha-integrations-list. Each project entry has a "Set Up Project" button and a three-dot menu. At the bottom, a pipeline for "Fullstack-React-and-GraphQL" is shown with a "build (81)" step that succeeded 6s ago. The pipeline details include duration, queue time, executor, branch, commit, and author. The "STEPS" tab is active, showing parallel runs for the build step. The steps listed are: Spin up environment (5s), Preparing environment variables (0s), Checkout code (0s), and echo "This is working" (0s).

▼ Load Balancing

Load Balancers

Target Groups New

Basic configuration

Load balancer name

Name must be unique within your AWS account and cannot be changed after the load balancer is created.

graphbook-alb

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Scheme [Info](#)

Scheme cannot be changed after the load balancer is created.

Internet-facing

An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. [Learn more](#)

Internal

An internal load balancer routes requests from clients to targets using private IP addresses.

IP address type [Info](#)

Select the type of IP addresses that your subnets use.

IPv4

Recommended for internal load balancers.

Dualstack

Includes IPv4 and IPv6 addresses.

Network mapping [Info](#)

The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

VPC [Info](#)

Select the virtual private cloud (VPC) for your targets. Only VPCs with an internet gateway are enabled for selection. The selected VPC cannot be changed after the load balancer is created. To confirm the VPC for your targets, view your [target groups](#).

-
vpc-03d1f06640bb71e59
IPv4: 172.30.0.0/16



Mappings [Info](#)

Select at least two Availability Zones and one subnet per zone. The load balancer routes traffic to targets in these Availability Zones only. Availability Zones that are not supported by the load balancer or the VPC are not available for selection. Subnets cannot be removed after the load balancer is created, but additional subnets can be added. Availability Zones that are not supported by the load balancer or the VPC are disabled. At least two subnets must be specified.

eu-central-1a

Subnet

subnet-08449d70f77ff7d55



IPv4 settings

Assigned by AWS

Security groups [Info](#)

A security group is a set of firewall rules that control the traffic to your load balancer.

Security groups

Select security groups



Create new security group

graphbook-src-grp sg-0a08126acbd582acb X
VPC: vpc-03d1f06640bb71e59

Listeners and routing [Info](#)

A listener is a process that checks for connection requests, using the protocol and port you configure. Traffic received by the listener is then routed per your specification. You can specify multiple rules and multiple certificates per listener after the load balancer is created.

▼ Listener HTTP:80		Remove
Protocol	Port	Default action Info
HTTP	: 80 1-65535	Forward to graphbook-target-group Target type: Instance
HTTP G Create target group		
Add listener		

Basic configuration

Settings in this section cannot be changed after the target group is created.

Choose a target type

- Instances**
 - Supports load balancing to instances within a specific VPC.
- IP addresses**
 - Supports load balancing to VPC and on-premises resources.
 - Facilitates routing to multiple IP addresses and network interfaces on the same instance.
 - Offers flexibility with microservice based architectures, simplifying inter-application communication.
- Lambda function**
 - Facilitates routing to a single Lambda function.
 - Accessible to Application Load Balancers only.

Target group name

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

Protocol **Port**

HTTP	: 80
------	------

VPC
Select the VPC with the instances that you want to include in the target group.

-
vpc-03d1f06640bb71e59
IPv4: 172.30.0.0/16

Protocol version

- HTTP1**
Send requests to targets using HTTP/1.1. Supported when the request protocol is HTTP/1.1 or HTTP/2.

Register targets

Available instances (0)

Instance ID	Name	State	Security groups	Zone	Subnet ID
No Available instances					
0 selected					
Ports for the selected instances					
Ports for routing traffic to the selected instances (separate multiple ports with commas):					
80					
Include as pending below					

Targets (0)

All	Health status	Instance ID	Name	Port	State	Security groups	Zone	Subnet ID
No instances added yet								
Specify instances above, or leave the group empty if you prefer to add targets later.								

0 pending [Cancel](#) [Previous](#) [Create target group](#)

Summary

Review and confirm your configurations. [Estimate cost](#)

Basic configuration Edit	Security groups Edit	Network mapping Edit	Listeners and routing Edit
graphbook-alb	• graphbook-src-grp sg-0a08126acbd582acb	VPC vpc-03d1f06640bb71e59 • eu-central-1a subnet-08449d70f77ff7d55 • eu-central-1b subnet-0a987225dd388cb36 • eu-central-1c subnet-0d7bfa818f9ab290e	• HTTP:80 defaults to graphbook-target-group
Tags Edit	<i>None</i>		
Attributes			
<p><i>ⓘ Certain default attributes will be applied to your load balancer. You can view and edit them after creating the load balancer.</i></p>			

Amazon Container Services

ECR > Repositories

[Private](#) [Public](#)

Private repositories [Create repository](#)

Repository name	URI	Created at	Tag immutability	Scan on push	Encryption type
No repositories					
No repositories were found					

Create repository

General settings

Visibility settings [Info](#)

Visibility settings

- Private**
Access is managed by IAM and repository policy permissions.
 - Public**
Publicly visible and accessible for image pulls.

Repository name

Provide a concise name. A developer should be able to identify the repository contents by the name.

532375010624.dkr.ecr.eu-central-1.amazonaws.com/ graphbook

9 out of 256 characters maximum (2 minimum). The name must start with a letter and can only contain lowercase letters, numbers, hyphens, underscores, and forward slashes.

Tag immutability [Info](#)

Tag immutability Enable tag immutability to prevent image tags from being overwritten by subsequent image pushes using the same tag. Disable tag immutability to allow image tags to be overwritten.

- Disabled

 Once a repository is created, the visibility setting of the repository can't be changed.

Successfully created repository graphbook		View push commands			
ECR > Repositories					
Private Public					
Private repositories (1)					
<input type="text"/> Find repositories	 View push commands Delete Edit Create repository	 1 			
Repository name	URI	Created at	Tag immutability	Scan on push	Encryption type
graphbook	552375010624.dkr.ecr.eu-central-1.amazonaws.com/graphbook	24. September 2021, 15:50:02 (UTC+04)	Disabled	Disabled	AES-256

The screenshot shows the AWS ECS Clusters page. The left sidebar lists navigation options: Amazon ECS, Clusters (which is selected and highlighted in orange), Task Definitions, Account Settings, Amazon EKS, Clusters, Amazon ECR, Repositories, AWS Marketplace, Discover software, and Subscriptions. The main content area has a heading 'Clusters' and a description: 'An Amazon ECS cluster is a regional grouping of one or more container instances on which you can run task requests. Each account receives a default cluster the first time you use the Amazon ECS service. Clusters may contain more than one Amazon EC2 instance type.' Below this is a link to 'ECS documentation'. Two buttons are present: 'Create Cluster' (blue) and 'Get Started' (grey). The 'View' dropdown menu is set to 'list', with other options 'card' and 'grid'. A message 'No clusters found' is displayed. At the bottom right, there are pagination controls showing '0 - 0 of 0'.

Select cluster template

The following cluster templates are available to simplify cluster creation. Additional configuration and integrations can be added later.

Networking only ⓘ

Resources to be created:

- Cluster
- VPC (optional)
- Subnets (optional)

ⓘ For use with either AWS Fargate or External instance capacity.

EC2 Linux + Networking

Resources to be created:

- Cluster
- VPC
- Subnets

Auto Scaling group with Linux AMI

EC2 Windows + Networking

Resources to be created:

- Cluster
- VPC
- Subnets

Auto Scaling group with Windows AMI

Configure cluster

Cluster name* 

Create an empty cluster

Instance configuration

Provisioning Model On-Demand Instance

With On-Demand Instances, you pay for compute capacity by the hour, with no long-term commitments or upfront payments.

Spot

Amazon EC2 Spot Instances let you take advantage of unused EC2 capacity in the AWS cloud. Spot Instances are available at up to a 90% discount compared to On-Demand prices.

[Learn more](#)

EC2 instance type*  

Manually enter desired instance type

Number of instances* 

EC2 AMI ID*  

Root EBS Volume Size (GiB) 

Key pair  

Networking

Configure the VPC for your container instances to use. A VPC is an isolated portion of the AWS cloud populated by AWS objects, such as Amazon EC2 instances. You can choose an existing VPC, or create a new one with this wizard.

VPC vpc-03d1f06640bb71e5...  

Check the structure for [vpc-03d1f06640bb71e59](#) in the Amazon EC2 console.

Subnets

- subnet-08449d70f77ff7d5 5 (172.30.0.0/24) - eu-central-1-a assign ipv6 on creation: Disabled
- subnet-0a987225dd388cb 36 (172.30.1.0/24) - eu-central-1-b assign ipv6 on creation: Disabled
- subnet-0d7bfa818f9ab290 e (172.30.2.0/24) - eu-central-1-c assign ipv6 on creation: Disabled

Select a subnet... 

Auto assign public IP

Disabled  

Security group

sg-0a08126acbd582ac...   

Rules for [sg-0a08126acbd582ac](#) in the EC2 Console.

Configure task and container definitions

A task definition specifies which containers are included in your task and how they interact with each other. You can also specify data volumes for your containers to use. [Learn more](#)

Task definition name* ?

Requires compatibilities* EC2

Task role ? refresh

Optional IAM role that tasks can use to make API requests to authorized AWS services. Create an Amazon Elastic Container Service Task Role in the [IAM Console](#)

Network mode ?

If you choose <default>, ECS will start your container using Docker's default networking mode, which is Bridge on Linux and NAT on Windows. Windows tasks support the <default> and awsvpc network modes.

Task execution IAM role

This role is required by tasks to pull container images and publish container logs to Amazon CloudWatch on your behalf. If you do not have the ecsTaskExecutionRole already, we can create one for you.

Task execution role ?

Task size



The task size allows you to specify a fixed size for your task. Task size is required for tasks using the Fargate launch type and is optional for the EC2 or External launch type. Container level memory settings are optional when task size is set. Task size is not supported for Windows containers.

Task memory (MiB)

The amount of memory (in MiB) used by the task. It can be expressed as an integer using MiB, for example 1024, or as a string using GB, for example '1GB' or '1 gb'.

Task CPU (unit)

The number of CPU units used by the task. It can be expressed as an integer using CPU units, for example 1024, or as a string using vCPUs, for example '1 vCPU' or '1 vcpu'.

Task memory maximum allocation for container memory reservation



Task CPU maximum allocation for containers



▼ Standard

Container name* graphbook-container i

Image* 532375010624.dkr.ecr.eu-central-1.amazonaws.com/graphbook:latest i

Private repository authentication* i

Memory Limits (MiB)* Hard limit i

+ Add Soft limit

Define hard and/or soft memory limits in MiB for your container. Hard and soft limits correspond to the 'memory' and 'memoryReservation' parameters, respectively, in task definitions.

ECS recommends 300-500 MiB as a starting point for web applications.

Port mappings Host port Container port Protocol i

0 8000 tcp x

+ Add port mapping

Environment variables

You may also designate AWS Systems Manager Parameter Store keys or ARNs using the 'valueFrom' field. ECS will inject the value into containers at run-time.

Key

database	Value	graphbook	x
host	Value	graphbook.cdua3y1nolio.eu-central-1.rds.amazonaws.	x
password	Value	admin	x
username	Value	admin	x
JWT_SECRET	Value	yCFwkSgD4qtsTkw6qJ5oxYKvaTGRLHdf3KiCzbfmkF	x
NODE_ENV	Value	production	x
AWS_ACCESS_KEY_ID	Value	AKIAXX5AF7VAGHPPXWNM	x
AWS_SECRET_ACCESS_KEY	Value	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	x
Add key	Add value		

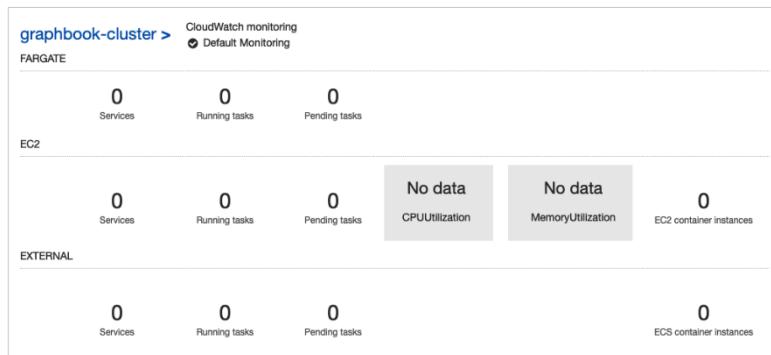
Log configuration Auto-configure CloudWatch Logs

<i>Log driver</i>	awslogs	<i>i</i>	
<i>Log options</i>	<i>Key</i>		
	awslogs-group	Value	/ecs/graphbook-task-de
	awslogs-region	Value	eu-central-1
	awslogs-stream-prefix	Value	ecs
	Add key	Value	Add value

Container definitions

Add container

Container Name	Image	Hard/Soft memory limit	CPU Units	GPU	Essential	Actions
graphbook-...	532375010624.d...	--/--			true	



Configure service

A service lets you specify how many copies of your task definition to run and maintain in a cluster. You can optionally use an Elastic Load Balancing load balancer to distribute incoming traffic to containers in your service. Amazon ECS maintains that number of tasks and coordinates task scheduling with the load balancer. You can also optionally use Service Auto Scaling to adjust the number of tasks in your service.

Launch type FARGATE EC2 EXTERNAL

[Switch to capacity provider strategy](#) i

Task Definition Family Enter a value i

Revision i

Cluster i

Service name i

Service type* REPLICA DAEMON i

Number of tasks i

Minimum healthy percent i

Maximum percent i

Load balancing

An Elastic Load Balancing load balancer distributes incoming traffic across the tasks running in your service. Choose an existing load balancer, or create a new one in the [Amazon EC2 console](#).

- Load balancer type***
- None**
Your service will not use a load balancer.
 - Application Load Balancer**
Allows containers to use dynamic host port mapping (multiple tasks allowed per container instance). Multiple services can use the same listener port on a single load balancer with rule-based routing and paths.
 - Network Load Balancer**
A Network Load Balancer functions at the fourth layer of the Open Systems Interconnection (OSI) model. After the load balancer receives a request, it selects a target from the target group for the default rule using a flow hash routing algorithm.
 - Classic Load Balancer**
Requires static host port mappings (only one task allowed per container instance); rule-based routing and paths are not supported.

Service IAM role ⓘ

Load balancer name ⏪ ⏴

Container to load balance

Container name : port ⏪ **Add to load balancer**

Container to load balance

graphbook-container : 8000 Remove ✕

Production listener port* ⓘ

Production listener protocol* HTTP

Target group name ⓘ

Target group protocol HTTP ⓘ

Target type instance ⓘ

Path pattern Evaluation order default

Health check path ⓘ

Additional health check options can be configured in the ELB console after you create your service.

Project Settings

Fullstack-React-and-GraphQL

Overview

Advanced

Environment Variables

SSH Keys

API Permissions

Jira Integration

Slack Integration

Status Badges

Webhooks

Environment Variables

Environment variables let you add sensitive data (e.g. API keys) to your jobs rather than placing them in the repository. The value of the variables cannot be read or edited in the app once they are set.

If you're looking to share environment variables across projects, try [Contexts](#).

Name	Value	Add Environment Variable	Import Variables
AWS_ACCESS_KEY_ID	xxxxXWNM		X
AWS_DEFAULT_REGION	xxxxal-1		X
AWS_ECR_ACCOUNT_URL	xxxx.com		X
AWS_REGION	xxxxal-1		X
AWS_SECRET_ACCESS_KEY	xxxx+7YJ		X
DOCKERHUB_PASSWORD	xxxx7Aa_		X
DOCKERHUB_USERNAME	xxxxi55		X

Parallel runs

1 / 1 parallel run

Parallel runs	
	0 02:36
	Use parallelism to run faster tests
	Parallelism speeds up tests by splitting them across multiple executors.
	Spin up environment 9s
	Container circleci/mysql:8.0.4 2m 27s
	Preparing environment variables 0s
	Checkout code 0s
	npm install 1m 12s
	Install Sequelize 11s
	Waiting for MySQL to be ready 0s
	Run migrations for test DB 1s
	Run tests 1m 0s

Pipeline	Status	Workflow	Branch / Commit	Start	Duration	Actions
Fullstack-React-and-GraphQL 58	Success	build-and-deploy	master 5f430cf AWS ECS	6m ago	6m 27s	
		Jobs				...
			test 108		1m 54s	
			aws-ecr/build-and-push-image 109		3m 50s	
			aws-ecs/deploy-service-update 110		27s	

Load balancer: graphbook-alb-ecs

Description Listeners Monitoring Integrated services Tags

Basic Configuration

Name	graphbook-alb-ecs
ARN	arn:aws:elasticloadbalancing:eu-central-1:532375010624:loadbalancer/app/graphbook-alb-ecs/62cf23236f3610ae
DNS name	graphbook-alb-ecs-1975100603.eu-central-1.elb.amazonaws.com (A Record)
State	Active
Type	application
Scheme	internet-facing
IP address type	ipv4
	Edit IP address type
VPC	vpc-002a66fa64ec59e3a
Availability Zones	subnet-00579126b1cab1578 - eu-central-1a
	IPv4 address: Assigned by AWS