

# Step 1 - How to deploy Frontends to AWS



New things we will learn include

1. Object stores (S3)
2. CDNs (Cloudfront)

Step 1 - Signup and get an AWS account.

Step 2 - Make sure you can access S3 and cloudfront (this will automatically happen if you are the root user of that account)

The screenshot shows the AWS Console Home page. On the left, under 'Recently visited' services, 'S3' and 'CloudFront' are highlighted with red arrows pointing to them. The main area displays the 'Applications' page, which is currently empty. It includes a search bar, a table header with columns for 'Name', 'Description', 'Region', and 'Originating account', and a button to 'Create application'.

# Step 2 - Build your React frontend



This approach will not work for frameworks that use Server side rendering (like Next.js)

This will work for basic React apps, HTML/CSS/JS apps

## Go to your react project

```
cd /link/to/your/react/project
```

Copy

## Build your project

```
npm run build
```

Copy

## Try serving the HTML/CSS/JS locally

```
npm i -g serve  
serve
```

Copy

At this point you have basic HTML/CSS/JS code that you can deploy on the internet.

You might be tempted to host this on an EC2 instance, but that is not the right approach

The next slide explains why

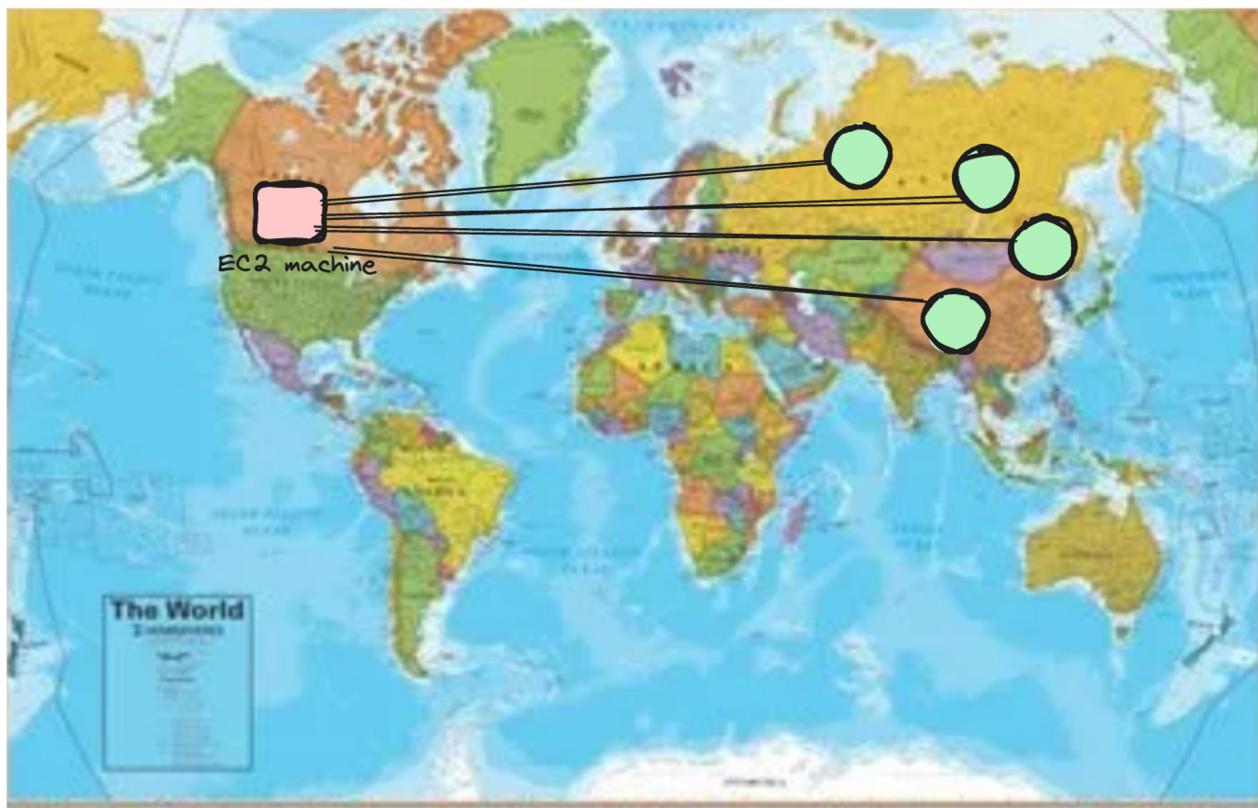
# Step 3 - What are CDNs?

A CDN stands for **Content Delivery Network**.

As the name suggests, it's an optimal way for you to deliver content (mp4 files, jpgs and even HTML/CSS/JS files) to your users.

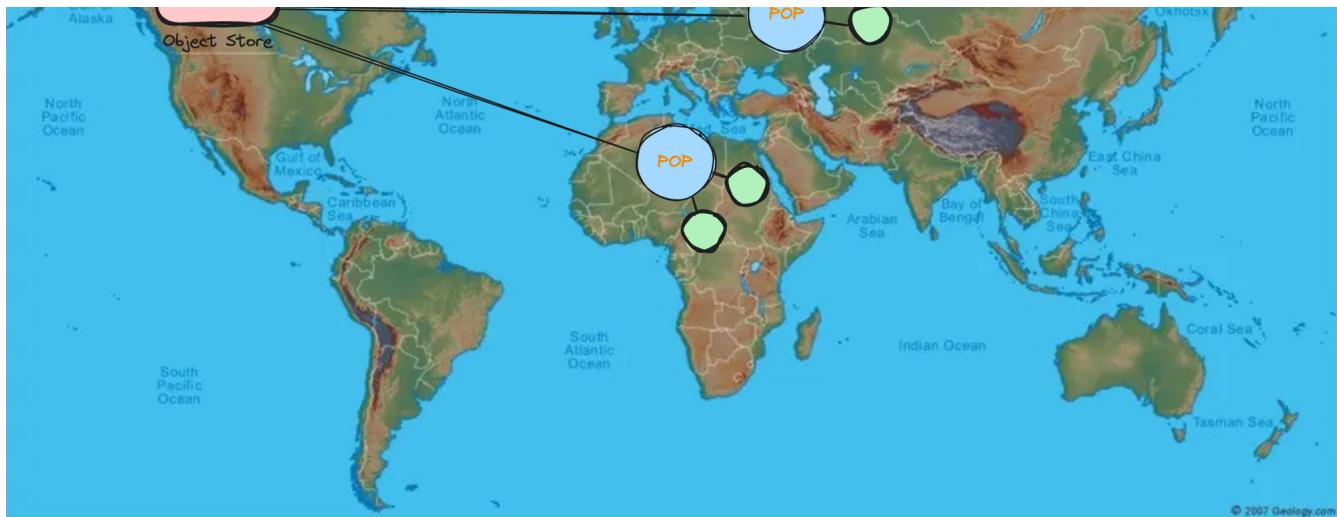
It is better than serving it from a VM/EC2 instances because of a few reasons -

## 1. EC2 machine approach



## 2. CDN approach





1. For frontends, mp4 files, images, `Object stores + CDNs` are a better approach.
2. You can't use the same for backends, since every request returns a different response.  
Caching doesn't make any sense there.



You can use edge networks for backends (deploy your backend on various servers on the internet) but data can't be cached in there.

Great video on how Hotstar scales their infrastructure during cricket matches (they use CDNs heavily)

**How Hotstar Scaled 25 Million Users**

Concurrency Pattern!

Day One: 13.9M  
NZ Betting

Day Two: 16.2M  
NZ Betting

Day Three: 25.3M

Gaurav Kamboj  
Cloud Architect at Hotstar  
@oyehooye

**20 K Views**

**hotstar tech\_**

A video thumbnail for a presentation titled "How Hotstar Scaled 25 Million Users". The thumbnail features a speaker on the left and a graph on the right. The graph shows concurrency patterns over three days: Day One (13.9M), NZ Betting; Day Two (16.2M), NZ Betting; and Day Three (25.3M). The graph has a white box highlighting the peak on Day Three. The text "Concurrency Pattern!" is displayed above the graph. The speaker's name, Gaurav Kamboj, is listed as a Cloud Architect at Hotstar with his Twitter handle @oyehooye. The video has received 20K views. The logo for "hotstar tech\_" is at the bottom.

# Step 4 - Creating an object store in AWS

In AWS, S3 is their object store offering.

You can create a **bucket** in there. A **bucket** represents a logical place where you store all the files of a certain project.

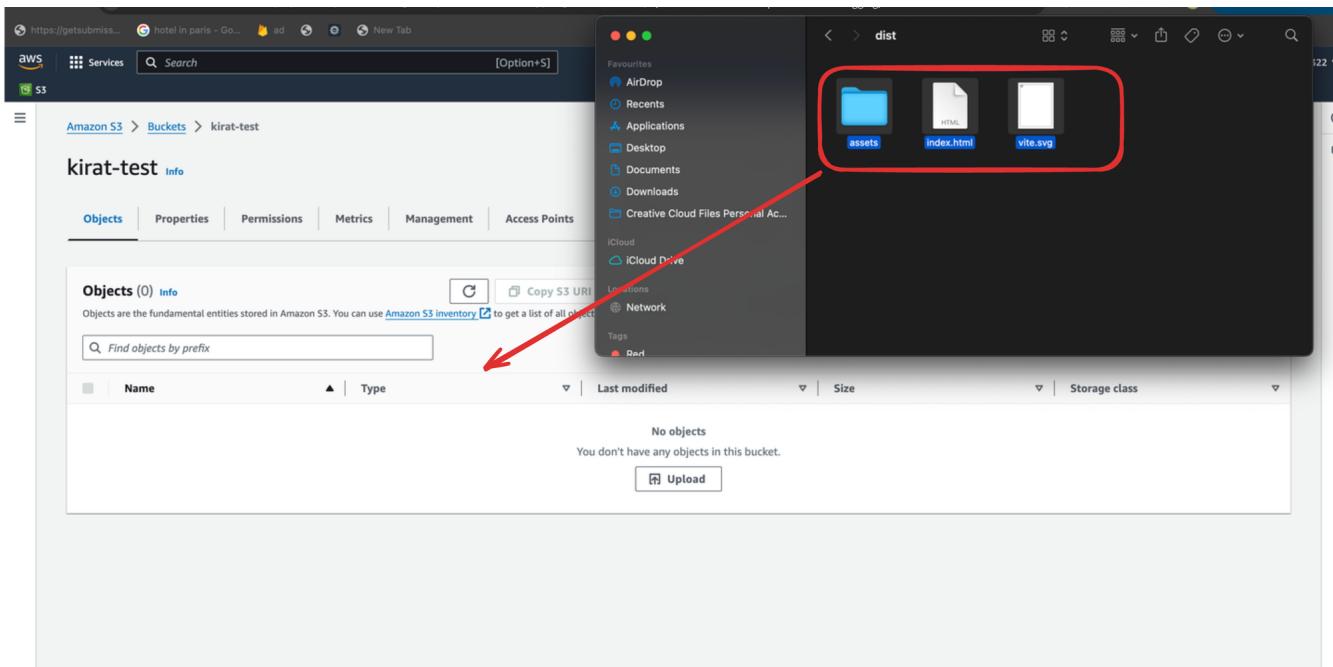
The screenshot shows the AWS S3 console with the 'General purpose buckets' tab selected. There are 12 buckets listed. The columns are Name, AWS Region, Access, and Creation date. One bucket named 'test11123123' is shown in detail: it's in the Asia Pacific (Mumbai) region, has 'Bucket and objects not public' access, and was created on June 10, 2023, at 21:51:06 (UTC+02:00). At the top right of the list, there are buttons for Copy ARN, Empty, Delete, and Create bucket. A red arrow points from the text above to the 'Create bucket' button.

The screenshot shows the 'Create bucket' wizard. The first step, 'General configuration', is displayed. It includes fields for 'AWS Region' (set to 'Asia Pacific (Mumbai) ap-south-1'), 'Bucket name' (set to 'kirat-test'), and a note about unique bucket names. Below these are sections for 'Copy settings from existing bucket - optional' and 'Object Ownership'. A red box highlights the 'Bucket name' field.

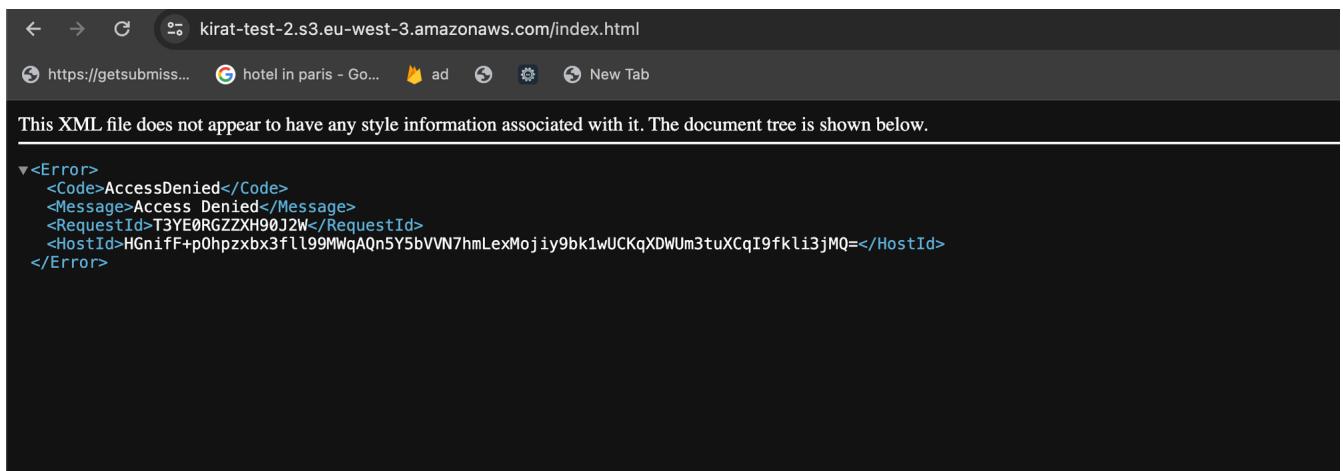
The screenshot shows the Amazon S3 console interface. At the top, there is a green banner with the message "Successfully created bucket 'kirat-test'". Below the banner, there is a link to "View details" and a close button (X). On the left, there is a navigation bar with "Amazon S3" and "Buckets". In the center, there is a section titled "Account snapshot" with a sub-section "Last updated: Feb 17, 2024 by Storage Lens. Metrics are generated every 24 hours. Metrics don't include directory buckets." A "Learn more" link is also present. On the right, there is a "View Storage Lens dashboard" button and a refresh icon.

# Step 5 - Upload the file bundle to S3

Upload all the files in the `dist` folder of your react project to S3



# Step 6 - Try accessing the website



You might be tempted to open your S3 bucket at this point, but don't

Your S3 bucket should be blocked by default, and you should allow CloudFront (CDN) to access it.

# Step 7 - Connecting Cloudfront

## Step 1 - Create cloudfont distribution

Go to cloudfront and create a new distribution. A `distribution` here means you're creating a place from where `content` can be distributed.

The screenshot shows the AWS CloudFront 'Distributions' list. There are two existing distributions listed:

ID	Description	Type	Domain name	Alternate do...	Origins	Status	Last modified
E3JCIJ9O0E2RYZ	-	Production	dibs5cabw92oe...	fe.100xdevs.com	kirat-test-2.s3.eu-wes	Enabled	February 19, 20...
E2B3PG65NKSQMO	-	Production	d2bq1lfgmpm8...	-	kirat-test.s3-website.:	Enabled	February 19, 20...

A red arrow points to the 'Create distribution' button at the top right of the table.

## Step 2 - Select your S3 bucket as the source

The screenshot shows the 'Origin' configuration page. The 'Origin domain' field is set to 'kirat-test-2.s3.eu-west-3.amazonaws.com'. The 'Name' field is set to 'kirat-test-2.s3.eu-west-3.amazonaws.com'. Under 'Origin access', the 'Origin access control settings (recommended)' option is selected, and a note states 'Bucket can restrict access to only CloudFront.' The 'Origin access control' dropdown is empty, and a warning message says 'This field cannot be empty'. A yellow warning box at the bottom states 'You must update the S3 bucket policy' and 'CloudFront will provide you with the policy statement after creating the distribution.'



Origin Access Control (OAC) is a feature in Cloudfront, which allows you to restrict direct access to the content stored in your origin, such as an Amazon S3 bucket or a web server, ensuring that users can only access the content through the CDN distribution and not by directly accessing the origin URL.

By the end of this, you should have a working cloudfront URL.

The screenshot shows the AWS CloudFront 'Distributions' page. The distribution ID is E3JCIJ900E2RYZ. The 'General' tab is selected. In the 'Details' section, the distribution domain name is listed as 'dibs5cabw92oe.cloudfront.net'. Below it, under 'Settings', the 'Alternate domain names' section contains 'fe.100xdevs.com'. A red arrow points from the distribution domain name field to the alternate domain names section.

## Step 8 - Connect your own domain to it

Websites aren't fun if you have to go to a URL that looks like this - <https://dibs5cabw92oe.cloudfront.net>

Connect your own custom domain by following the given steps -

### 1. Select edit on the root page

The screenshot shows the AWS CloudFront 'Distributions' page. The distribution ID is E2B3PG65NKSQMO. The 'General' tab is selected. The distribution domain name is listed as 'dibs5cabw92oe.cloudfront.net'.

The screenshot shows the AWS CloudFront distribution settings page. At the top, there's a 'Details' section with fields for 'Distribution domain name' (d2bq1lmpgm8dp.cloudfront.net), 'ARN' (arn:aws:cloudfront::163679972322:distribution/E2B3PG65NKSQMO), and 'Last modified' (February 19, 2024 at 5:01:19 AM UTC). Below that is a 'Settings' section. On the right side of the 'Settings' section, there's an 'Edit' button, which is highlighted with a red arrow. The 'Settings' section includes fields for 'Description', 'Price class' (set to 'Use all edge locations (best performance)'), 'Alternate domain names' (empty), 'Supported HTTP versions' (HTTP/2, HTTP/1.1, HTTP/1.0), 'Standard logging' (Off), 'Cookie logging' (Off), and 'Default root object' (empty).

## 2. Attach a domain name to the distribution

The screenshot shows the 'Edit settings' page for a CloudFront distribution. The 'Settings' section is open. Under 'Price class', 'Use all edge locations (best performance)' is selected. In the 'Alternate domain name (CNAME) - optional' section, the input field contains 'fe.100xdevs.com' and has a 'Remove' button next to it. There is also an 'Add item' button. A red box highlights this entire section. Below it, a note says '(i) To add a list of alternative domain names, use the bulk editor.' Under 'Custom SSL certificate - optional', there's a dropdown menu set to 'Choose certificate' and a 'Request certificate' button. In the 'Supported HTTP versions' section, 'HTTP/2' is checked and 'HTTP/3' is unchecked. The 'Default root object - optional' section is shown with an empty input field.

## 3. Create a certificate

Since we want our website to be hosted on HTTPS, we should request a certificate for our domain

[Edit settings](#)

## EDIT SETTINGS

### Settings

#### Price class | [Info](#)

Choose the price class associated with the maximum price that you want to pay.

- Use all edge locations (best performance)
- Use only North America and Europe
- Use North America, Europe, Asia, Middle East, and Africa

#### Alternate domain name (CNAME) - *optional*

Add the custom domain names that you use in URLs for the files served by this distribution.

[Remove](#)
[Add item](#)

To add a list of alternative domain names, use the [bulk editor](#).

#### Custom SSL certificate - *optional*

Associate a certificate from AWS Certificate Manager. The certificate must be in the US East (N. Virginia) Region (us-east-1).


[Request certificate](#) 

#### Supported HTTP versions

## Step 4 - Follow steps to create the certificate in the certificate manager

Certificate status																	
Identifier fb2088a7-c75f-4a42-91af-2a7fb4600a0e	Status Issued	ARN <a href="#">arn:aws:acm:us-east-1:163679972322:certificate/fb2088a7-c75f-4a42-91af-2a7fb4600a0e</a>	Type Amazon Issued														
Domains (1)																	
<table border="1"> <thead> <tr> <th>Domain</th> <th>Status</th> <th>Renewal status</th> <th>Type</th> <th>CNAME name</th> <th>CNAME value</th> </tr> </thead> <tbody> <tr> <td>fe.100xdevs.com</td> <td> Success</td> <td>-</td> <td>CNAME</td> <td><a href="#">_a96f3ef5c0e0ef282152985dfb428092.fe.100xdevs.com.</a></td> <td><a href="#">_21845a5bfaa0b0cbbb6b8a55b28c5501.mhbtspdnt.acm-validations.aws.</a></td> </tr> </tbody> </table>					Domain	Status	Renewal status	Type	CNAME name	CNAME value	fe.100xdevs.com	Success	-	CNAME	<a href="#">_a96f3ef5c0e0ef282152985dfb428092.fe.100xdevs.com.</a>	<a href="#">_21845a5bfaa0b0cbbb6b8a55b28c5501.mhbtspdnt.acm-validations.aws.</a>	<a href="#">Create records in Route 53</a> <a href="#">Export to CSV </a>
Domain	Status	Renewal status	Type	CNAME name	CNAME value												
fe.100xdevs.com	Success	-	CNAME	<a href="#">_a96f3ef5c0e0ef282152985dfb428092.fe.100xdevs.com.</a>	<a href="#">_21845a5bfaa0b0cbbb6b8a55b28c5501.mhbtspdnt.acm-validations.aws.</a>												
Details																	
In use	Serial number	Requested at	Renewal eligibility														

These DNS settings are active. Changes are published immediately, but may take time to propagate

Resource records

Resource records point to the services that your domain uses, including web and email services. [Learn more about resource records](#)

**Custom records**  
100xdevs.com/A, \_a96f3ef5c0e0ef282152985dfb428092.fe.100xdevs.com/CNAME and 27 more

[Manage custom records](#)

Host name	Type	TTL	Data
100xdevs.com	A	1 hour	76.76.21.21
_a96f3ef5c0e0ef282152985dfb428092.f e.100xdevs.com	CNAME	1 hour	_21845a5bfaa0b0cbbb6b8a55b28c5501.mhbtsbpndt.acm- validations.aws.

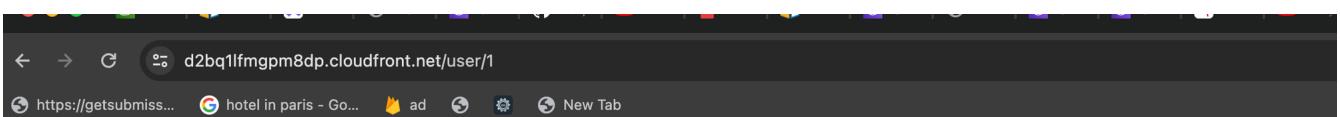
## Step 5 - Add a CNAME record for the website to point to your cloudfront URL

fe.100xdevs.com	CNAME	1 hour	dibs5cabw92oe.cloudfront.net.
-----------------	-------	--------	-------------------------------

That's it, you have a fully running react project hosted on HTTPS on a custom domain

## Step 9 - Error pages

You will notice a problem, whenever you try to access a route on your page that isn't the index route (/user/1) , you reach an error page



- Code: AccessDenied
- Message: Access Denied
- RequestId: V17AB7NEC9FRRDWX
- HostId: OJzE4K3MrighV9+NivXXYCb1ueDb26IEZ6MEVL99vUfhQZkiYXW9K1IUjtAvpFMRyx/IoMVnqaw=

This is because CloudFront is looking for a file `/user/1` in your S3, which doesn't exist.

To make sure that all requests reach `index.html`, add an `error page` that points to `index.html`.

The screenshot shows the AWS CloudFront console. In the top navigation bar, 'CloudFront' is selected under 'Distributions'. Below it, a distribution named 'E3JCIJ9O0E2RYZ' is chosen. On the left, a sidebar lists 'General', 'Security', 'Origins', 'Behaviors', and 'Error pages', with 'Error pages' currently active. At the top right, there's a 'View metrics' button. The main content area is titled 'Error pages' and contains a table with columns: 'HTTP error code', 'Minimum TTL (seconds)', 'Response page path', and 'HTTP response code'. A 'Create custom error response' button is located at the bottom right of the table.

[CloudFront](#) > [Distributions](#) > [E3JCIJ9O0E2RYZ](#) > [Edit error page response](#)

## Edit custom error response

The screenshot shows the 'Edit error page response' dialog. At the top, it says 'Error response [Info](#)'. The first section is 'HTTP error code', which contains a dropdown menu showing '404: Not Found'. The next section is 'Error caching minimum TTL', with a text input field containing '0'. Under 'Customize error response', there are two radio buttons: 'No' (unchecked) and 'Yes' (checked). The 'Yes' option leads to a 'Response page path' input field containing '/index.html'. The final section is 'HTTP Response code', with a dropdown menu showing '200: OK'. At the bottom right, there are 'Cancel' and 'Save changes' buttons.



You might have to invalidate cache to see this in action.

