

The web application in Assignment 2 is a world time zone converter, it based on my Assignment 1, with some minor modifications.

## **User guide**

The index page of the website is a login page, there are two users for testing, one is admin, the other is user, passwords are not needed.

The default regions to convert of each user are different. If you do not want to login as a user, you can visit the time zone page as a guest as well.

On the time zone page, if you have logged in, there is welcome information at the top.

There are two drop-down lists to select regions: the origin region and the region where you want to convert to. The real-time is set to the default time, it can be changed.

When two regions and time is set, click Enter, then the origin time zone and time, converted time zone and time are shown under the Enter button. You can submit more queries if you like.

For testing, a table contains all the regions and time zones are shown at the bottom of the page.

## **Cloud services are being used**

The application is running in the AWS and consists of three parts: the UI, which is web pages built on a web server, storage and a database. The original ideas to deploy the webserver on EC2 and stored files into S3 in web server on EC2 is covered in Lab 8 and 9, and I made some customizations for this assignment. For the details please refer to the files on GitHub. For the database, I built an instance on Amazon RDS and used the command line and MySQL to create tables and insert data.

To reach the web application is simple, just visit: `ec2-18-212-128-227.compute-1.amazonaws.com`. Then follow the user guide above.

The whole application is built on the cloud, the URL of the web page shows that it is based on EC2 and here is the snapshot of the instance running on EC2.

The screenshot displays the AWS Management Console interface for an EC2 instance. At the top, a navigation bar shows filters for Instance ID, Instance Type, Availability Zone, Instance State, Status Checks, Alarm Status, and Public DNS (IPv4). Below this, a table lists instance details: Instance ID (i-015d51496de9a0e85), Instance Type (t2.micro), Availability Zone (us-east-1b), Instance State (running), Status Checks (2/2 checks passed), Alarm Status (None), and Public DNS (IPv4) (ec2-18-212-128-227.compute-1.amazonaws.com).

The main section shows the 'Description' tab selected, displaying a detailed list of instance attributes:

Instance ID	i-015d51496de9a0e85	Public DNS (IPv4)	ec2-18-212-128-227.compute-1.amazonaws.com
Instance state	running	IPv4 Public IP	18.212.128.227
Instance type	t2.micro	IPv6 IPs	-
Elastic IPs		Private DNS	ip-172-31-44-207.ec2.internal
Availability zone	us-east-1b	Private IPs	172.31.44.207
Security groups	cosc349-web, cosc349-ssh. <a href="#">view inbound rules</a> . <a href="#">view outbound rules</a>	Secondary private IPs	
Scheduled events	No scheduled events	VPC ID	vpc-97e38ded
AMI ID	ubuntu/images/hvm-ssd/ubuntu-xenial-16.04-amd64-server-20190816 (ami-0378588b4ae11ec24)	Subnet ID	subnet-bc97c9e0

The URL of the image on the website shows the image is stored in S3. (<https://mahebei.s3.amazonaws.com/World Time Zones Map.png>) The screen-captures below show the database is run on RDS.

The screenshot shows the AWS RDS console for a database instance named 'cosc349'. The breadcrumb navigation indicates the path: RDS > ... > cosc349. Below the instance name, there are 'Modify' and 'Actions' buttons.

The 'Summary' section provides key details about the database instance:

DB identifier	cosc349	CPU	1.80%
Info	Available	Class	db.t2.micro
Role		Current activity	
Instance		Connections	0 Connections
Engine	MySQL Community	Region & AZ	us-east-1b

Connectivity & security		
Endpoint & port	Networking	Security
Endpoint cosc349.c9myo6s6lujg.us-east-1.rds.amazonaws.com	Availability zone us-east-1b	VPC security groups DB (sg-0ce486b3edf64e40f) ( active )
Port 3306	VPC vpc-97e38ded	Public accessibility Yes
	Subnet group default-vpc-97e38ded	Certificate authority rds-ca-2015
	Subnets subnet-c3d5c2cc subnet-d4eb8aea subnet-0ce2b06b subnet-0dc42940 subnet-bc97c9e0 subnet-e36931cd	Certificate authority date Mar 6th, 2020

The web page shows media resources stored in the S3 and makes queries to connect and search the database in RDS to convert time between time zones.

## Develop processes

The application is based on Assignment 1.

Vagrantfile is based on the materials in Lab 9 and I added web server and PHP installation and configuration to it.

Some minor modifications on the web page: connection to database changed to RDS, and an image stored in S3 is added.

## Interaction of VM and APIs

Since my web application for Assignment 1 is consists of one webserver and two database servers, so in this assignment, there is only one VM built on EC2 and one database on RDS. I also build an S3 bucket to store an image.

The web pages on the web server show the image stored in S3, and make searching queries in database on RDS.