

## AWS Academy – Final Project

**Activity**

This activity provides students with an opportunity to architect a solution to move an existing application to AWS.

**Pre-requisites**

This lab requires:

- Access to a notebook computer with Wi-Fi running Microsoft Windows, Mac OS X, or Linux (Ubuntu, SuSE, or Red Hat).
  - **Note** The qwikLABS lab environment is not accessible using an iPad or tablet device, but you can use these devices to access the student guide.
- For Microsoft Windows users: Administrator access to the computer.
- An Internet browser such as Chrome, Firefox, or Internet Explorer 9 (previous versions of Internet Explorer are not supported).
- An SSH client, such as PuTTY.

**Duration**

8–10 hours

**GoGreen  
Overview**

GoGreen is a regional insurance company based in San Diego, California. GoGreen has a three-tier CRM web application, supporting sales users at headquarters, remote users, and mobile users.

The application supports sales personnel to input new customers' data and edit current customers' information. GoGreen's goal is to reduce the use of paper, so whenever possible, users can also upload documents such as signed contracts and pictures of assets.

The application is extremely important to the business. In addition to storing customers' data and documents, it also converts these documents into multiple formats, such as images for web and mobile use.

Most of the users are located on the west coast of the United States, but because some of GoGreen's customers have presence in Europe and South America, sales representatives travel to these locations and need to access the application with acceptable performance.

Currently the application runs on premises at GoGreen's data center and consistently suffers performance and reliability issues.

A year ago the CIO asked the IT team to ensure that the application handled increased traffic and users.

As a result, the IT team overprovisioned resources, which helped alleviate the problem for the first three months. This process had to be repeated a few times.

GoGreen's CIO, in conjunction with the CFO, concluded that purchasing new equipment every 2–4 months is not a viable solution for this environment.

The procurement process, which is always expedited, takes about 20 days, deploying the added capacity takes 1 week, and it costs around \$100,000.

As problems escalate, GoGreen hired you as Cloud Architect to help them move this application to AWS.

**Current  
Environ-  
ment**

- Web tier:
  - 6 x Virtual Machines (2 vCPUs / 4GB memory)
  - SUSE Linux Enterprise Server 12
  - Apache web server
  - PHP server with SQL connectors
  - PHP files
- Application tier:
  - 5 x virtual machines (4 vCPUs/32-GB memory)
  - SUSE Linux Enterprise Server 12
  - Java SRE 7
  - Java application files
- Database tier:
  - 2 x virtual machines ( 8 vCPUs/48-GB memory/5.5-TB storage)
  - SUSE Linux Enterprise Server 12
  - MySQL 5.6.22 database cluster

**Requirements**

- Environment:
  - Can only be managed by users located at GoGreen's data center.
  - Headquarters and remote users require encrypted access to the application .
  - Mobile users establish a connection with the application and are required to keep the same session.
  - Establish a baseline for the number and type of instances needed.
  - Developers require a central repository to push the production version of the code for the Web and Application tier servers.
  - Recovery Point Objective for the application is four hours.
  - Recovery Time Objective for the application is 30 minutes.
  - Must support a user base that is expected to grow 90% in the next three years
  - Documents and pictures must be kept for five years. However, these files are rarely requested after three months
- Web tier:
  - Architecture must be flexible and must handle any peak in traffic or performance.
  - Servers are currently at 75% of memory capacity all the time. This number has to decrease to between 50% and 60% when initially moved to AWS.
  - The overall acceptable incoming network bandwidth is between 300 Mbps and 750 Mbps.
  - Application admins want to be notified by email if there are more than 100 "400 HTTP errors" per minute in the application.
- Application tier
  - Architecture must be flexible and must handle any peak in traffic or performance
  - Servers are currently at 90% of memory and CPU capacity all the time. This number has to decrease to between 50%

	<p>and 60% when initially moved to AWS.</p> <ul style="list-style-type: none"> <li>○ Need eventual Internet access for patching and updates.</li> <li>○ Overall memory and CPU utilization should not go above 80% and 75% respectively or below 30% for both.</li> <li>○ As servers are added/removed, the application needs a centralized location to store logs.</li> </ul> <ul style="list-style-type: none"> <li>• Database tier <ul style="list-style-type: none"> <li>○ Needs to be moved with as few changes to the data structure as possible.</li> <li>○ Maximum downtime during migration is two hours.</li> <li>○ Database needs consistent storage performance at 21000 IOPS.</li> <li>○ Need eventual Internet access for patching and updates.</li> </ul> </li> </ul>
--	--

---

<b>Cost Considerations</b>	<p>The proposed solution must take into consideration all the technical requirements and the most conscious financial options.</p> <ul style="list-style-type: none"> <li>• Type of instances and payment models</li> <li>• Number of instances</li> <li>• Estimated monthly cost for the solution</li> </ul>
----------------------------	---

---

---

**Project  
Plan**

- Provide a diagram of your proposed solution with as many details as possible.
  - Provide full documentation about all aspects of configurations in the environment.
    - Consider it as a guide for someone to deploy the application on AWS.
    - Details are very important.
  - A CloudFormation template of the environment in addition to the documentation will be accepted.
-