

# AWS Management Console

QA will use the AWS Management Console much the same way it uses Jenkins to deploy updates. But we can also monitor our test instances and troubleshoot problems.

## Deploying a code branch to a QA instance

1. In the JIRA card you're testing get the branch name.
2. Log in to Github and navigate to the repo.
3. In the Branches dropdown, select the branch you're testing.
4. Click the link to the latest commit.



chasdl Merge remote-tracking branch 'origin/7.5-dev' into SUG-1106 Latest commit 94e2ba3 6 days ago

5. Copy the commit ID from the page or the end of the URL.

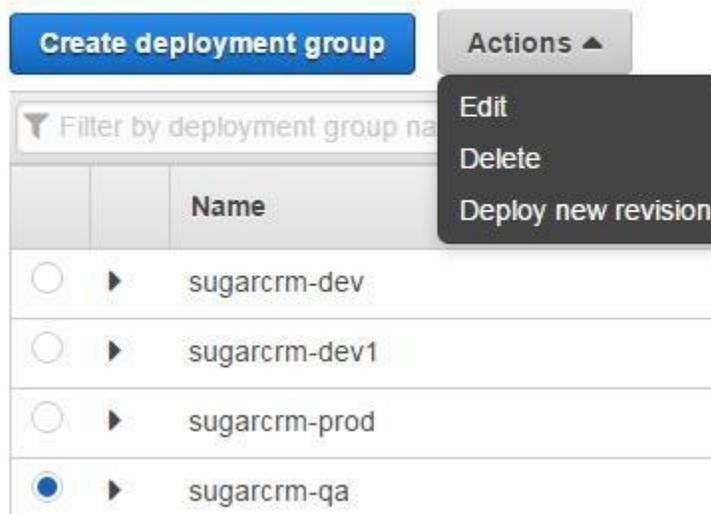


2 parents 947033e + 62c5392 commit 94e2ba3572bec9251a25536a9bfe018fe5e29d12

6. Log in to the [AWS Management Console](#) using your Learning House Active Directory credentials.
7. Navigate to the Services page and click CodeDeploy.
8. Click the link to the application you're testing.
9. Under Deployment Groups select the instance you want to deploy to and select Deploy New Revision in the Actions dropdown.

### Deployment groups

View, edit, and redeploy revisions to your deployment groups



Create deployment group Actions ▲

Filter by deployment group name

	Name
<input type="radio"/>	sugarcrm-dev
<input type="radio"/>	sugarcrm-dev1
<input type="radio"/>	sugarcrm-prod
<input checked="" type="radio"/>	sugarcrm-qa

10. On the Create New Deployment page, select the application and the instance you're deploying to.
11. Under Revision Type, select My application is stored in GitHub.
12. In the repository name box, type *thelearninghouse/[repository name]*; for example: *thelearninghouse/sugarcrm*.
13. In the Commit ID box, paste the ID you copied from GitHub.
14. In the DeploymentConfig dropdown, select *CodeDeployDefault.OneAtATime*.
15. Click Deploy Now.

16. The progress spinner displays as the request is processed.