PROJECT SPECIFICATION

Hybrid Cloud SaaS: Three-Tier Web Application

Create Web App Blueprint Satisfying Business Requirements

CRITERIA	MEETS SPECIFICATIONS
Create a three-tier web application blueprint that satisfies the IaaS business requirements for self-service hybrid cloud deployment and security scenarios.	Submitted blueprint with: • Two SSH credentials • Four VM service tiers (two are web tiers on different infrastructure providers) • VM and tasks follow security standards specified • Two application profiles

Enhance the Blueprint with Hybrid Web Tier Scaling

CRITERIA	MEETS SPECIFICATIONS

Enhance the blueprint to provide PaaS by configuring the web application and hosting each service in the proper cloud as specified, configure hybrid web-tier scaling on public and on private clouds.	Submitted a blueprint with four service VM configuration of: • Load Balancer on private cloud • Web tier 1 on private cloud • Web tier 2 on public cloud • Database on public cloud Submitted blueprint has: • a web application, configured to read and write to the database • post-deployment operation: a web-tier that scales and orchestrate updates to the load balancer configuration • web tier scaling actions that updates the load balancer configuration across both web tiers • two profiles for deployment: small and medium, with specified VM resource and replicas configuration

Blueprint Enhancement: Database Backup and Restoral

CRITERIA	MEETS SPECIFICATIONS

Achieve a SaaS-like experience for developers with separate delegated, post-deployment operations to back up the database and restore the database.	Submitted a blueprint with: • Database backup custom action, adapted from the source code MySQL_backup_file.sh.txt to use the end-user, runtime specified password for the database root account • Database restore custom action, adapted from the source code MySQL_restore.sh.txt to use the end-user, runtime specified password for the database root account.

Suggestions to Make Your Project Stand Out!

- Make the web tier scale-in and scale-out actions driven by a variable/macro provided by the user instead of hard coding it to a static value 1.
- Make the database backup action and/or database restore orchestrate the stop and (re)start of web tier or HAProxy services before and after database operations for the proper order of operations and to prevent any database access from the web or load balancer services.
- Make cloud-init leverage macros/variables instead of hard coding account names and public keys.
- Make the database password a mandatory field, use regular expressions to validate 8 or more characters.