Project PI : Groupe 3
Class : 4 ARCTIC 8

Functional needs and technical constraints extraction

Team Membres : Mohamed amin Souid - Malek Zegheouf - Hichem Mejri - Aziz Omri Chaima Boughanmi - Houssem Benelouefi - Mahmoud Meziou

Objectifs	Functional needs	Technical constraint	Products	Choice
virtualized computing resources on demand.	**Infrastructure Cloud**	●Be an open source solution. ● Have a modular/ Multi-tenant architecture. ● Provide orchestration capabilities. ● Support a large choice of hypervisors. ● Support private and hybrid deployment. ● Leverages commodity hardware. ● Deliver simple management of cloud resources through a centralized dashboard. ●Provide standards for building open, massively scalable clouds.	●OpenStack	
		●Provide self-healing, reliable, data redundancy protection from failures. ● Scale vertically and horizontally-distributed storage. ● Backup and archive large amounts of data with linear performance. ● Easily add capacity.	●KVM ●Hyper-v ●VmWare	

Project PI : Groupe 3
Class : 4 ARCTIC 8

virtualized resources	**Virtualization**	● Fully supported and compatible with the infrastructure management solution ● Can use a wide variety of certified Linux-supported hardware platforms. ● Supports live migration ● Scales to match demand load if the number of guest machines and requests increases ● Allows the most demanding application workloads to be virtualized		
Minimize infrastructure resources provisioning time, eliminate repetitive tasks and reduce mistakes & errors risk.	**Automation**	● An open-source software. ● Facilitates a smooth connection with the foundational infrastructure. ● Uses templates that describe the infrastructure for a cloud application in a text file that is readable and writable by humans, and can be checked into version control. ● Can be used to specify the relationships between resources (e.g. this volume is connected to this server) ● Should allow integration with other components of the cloud infrastructure. It	●Ansible ●Terraform	

		will allow creation of most resource types as well as some more advanced functionality such as instance high availability, instance auto scaling, and nested stacks.		
Provide the deployment environment required to run the web application.	**Deployment**	● Hardware and Software Compatibility ● Scalability ● Containerization and Orchestration ● Data Migration	●Ansible	
Centralized management and orchestration for all the deployment environments	**Orchestration**	● Installable almost on any Linux distribution such as Debian or Ubuntu. ● Installable automatically using configuration management tools. ● Be a standard and be available on a large number of platforms. ● Facilitate both declarative configuration and automation. ● Full integration with the cloud infrastructure services, specifically in terms of identity management, networking and block storage. ● Able to deliver deployment environments as a service. ● Based on standard solutions and available on a large number of platforms. ● Facilitate both declarative configuration and automation.	●Kubernetes ●OpenShift ●Docker Swarm	

		● Provide deployment patterns and templates. ● Provide service discovery and load balancing ● Ensure self-healing.		
Effectively observing and visualizing the system's performance and health.	**Surveillance**	● Real-Time Data Collection: The configuration management tool should enable the configuration and deployment of components capable of collecting real-time data from the entire infrastructure, including servers, storage, and containers. ● Dashboard Generation: It must be capable of creating and maintaining configurations of customized dashboards to visualize essential metrics and statistics for infrastructure management. ● Alerts and Notifications: The tool should provide capabilities for configuring custom alerts and real-time notifications in the event of threshold breaches or critical issues. ● Scalability: It should allow for the automatic addition or removal of monitoring resources to adapt to variations in workload. ● Security: The tool must ensure an appropriate level of security by maintaining data confidentiality and providing robust authentication and	●Terraform ●Prometheus	

		authorization mechanisms.		