**Requirements List**

[1. Requirements Extracted from the Usage Scenarios 2](#_Toc279059900)

[2.1 Functional Requirements 2](#_Toc279059901)

[2.2 Non – Functional Requirements 3](#_Toc279059902)

[2. Requirements Collected from the State of the Art review 5](#_Toc279059903)

[2.1 Functional Requirements 5](#_Toc279059904)

[2.2 Non-Functional Requirements 7](#_Toc279059905)

[References 10](#_Toc279059906)

# Requirements Extracted from the Usage Scenarios

## Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **Actor** | **Requirement** | | **Scenario (s)** |
| Application Developer |  | Register in the Cloud4SOA platform | 2 |
|  | Edit/update profile | 2 |
|  | Create the application’ s semantic profile | 1, 2 |
|  | Search in the Cloud4SOA marketplace for the appropriate solution | 1, 2 |
|  | View the recommended PaaS offerings | 1, 2 |
|  | Establish agreements with Cloud4SOA enabled providers (when registered) | 2 |
|  | Deploy his application on a Cloud platform | 1, 2 |
|  | Deploy part of his application on another provider | 2 |
|  | Monitor, manage and configure his application’s lifecycle though a common interface provided by Cloud4SOA | 1, 2 |
|  | View details on the infrastructure and the services provided by each provider | 1, 2 |
|  | Tag / annotate the Cloud services and PaaS offerings | 2 |
|  | Rate PaaS providers (in terms of quality, reliability etc.) | 2 |
|  | View the user-generated comments and ratings for a specific PaaS provider | 2 |
|  | Withdraw his application(s) from the Cloud or from a specific provider | 2 |
|  | Migrate his applications to an interoperable Cloud platform | 2 |
|  | Re-negotiate the SLA with new requirements | 2 |
|  | Receive notification when an SLA violation is raised | 2 |
| Cloud Platform Provider |  | Join the Cloud4SOA marketplace | 1, 2 |
|  | Edit/update profile | 2 |
|  | Map his API with the interoperable Cloud4SOA API | 2 |
|  | Define their pricing model and the associated performance characteristics (semantic profile of his platform) and publish them in the Cloud4SOA service catalog | 1, 2 |
|  | Establish agreements/form alliances/collaborate with other providers in the Cloud4SOA marketplace | 1, 2 |
|  | Update the SLAs in the case of changes | 2 |
|  | Be “punished” by the Cloud4SOA platform when violating a negotiated agreement (SLA) | 2 |

## Non – Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement type** | **Requirement** | | **Scenario (s)** |
| Security |  | User level authentication | 2 |
|  | Data security/accuracy and Data privacy | 1, 2 |
|  | Network security | 1, 2 |
| Usability (user friendly front-end) |  | Consistency and standards | 1, 2 |
|  | Error prevention | 2 |
|  | Recognition rather than recall | 2 |
|  | Aesthetic and minimalist design of the user interface | 1, 2 |
|  | Help and documentation | 1, 2 |
|  | Visibility of system status | 2 |
| High availability / Reliability |  | Well handled downtime scenarios | 2 |
|  | Bug free software | 1, 2 |
|  | No / minimum data loss | 1, 2 |
|  | Provision of supplement storage at run-time in the case it is needed | 2 |
|  | Available technical support | 1, 2 |
| Performance |  | Minimum response time | 1, 2 |
|  | Database query optimization | 1, 2 |
|  | Avoid over engineering | 1, 2 |
|  | Frequent collection of resource information for the monitoring of their status and therefore the services performance | 2 |
|  | Prioritization of the resources provided amongst the providers of the Cloud by e.g. monitoring the traffic load | 2 |
| Configurability |  | The platform should be configurable by e.g. allowing users to change the ways they access a functionality | 1, 2 |
| Flexibility |  | The architecture should allow the removal/ addition of functionalities / technologies (version upgrades) | 2 |
| Interoperability |  | Interoperability between Cloud platform providers | 1, 2 |
|  | Common APIs | 1, 2 |
|  | Homogenous service descriptions (Provision of common terminology e.g. semantics, abstracting providers’ heterogeneity) | 1, 2 |

# Requirements Collected from the State of the Art review

## Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **Actor** | **Requirement** | | **Reference** |
| Application Developer |  | Transparently migrate data/ applications (portability) | [1] [2] [3] [4] |
|  | Manage resources across multiple Cloud providers | [5] [2] [4] [6] [7] [8] |
|  | Configure (user interface, workflow, business rules, customizable data model and metadata set) | [9] |
|  | Monitor execution performance in real-time | [5] [2] [4] [7] [6] [8] |
|  | Receive (SLS violation, failure) alerts | [10] |
|  | Manage the complete lifecycle of a service | [2] [11] |
|  | Orchestrate all the involved components | [12] |
|  | Reuse and share functions provided by existing application, enabling applications to work together (integration tools) | [2] [13, 14] [3] [9] |
|  | Support backup and restore | [10] [6] |
|  | Scheduling capabilities such as giving priority to precision or immediate responses. | [6] |
|  | Use testing tools | [2] |
|  | Use a common API (common set of tools) that supports provisioning/deployment/configuration and control across different Clouds | [7] [8] [2] [15] [9] [16] [11] |
|  | Use tools for defining and developing the service as well as to develop the images associated with the service | [11] |
|  | Search for resources including applications and middleware held by multiple Cloud systems. | [12] [6] |
|  | Select an appropriate Cloud provider by means of comparing the service quality requirement with the SLA of the Cloud systems provided by other providers. | [6] |
|  | Access and view the provided services listed in a service catalog | [11] [17] [18] |
|  | Enable user profiling, notification and subscription | [10] |
|  | Provide remotely activate resources (virtual machines, applications, and middleware). | [6] |
|  | Support developer community | [16] [9] |
|  | Support a marketplace (application selling business model, SLS adaptation and support, service billing policy) | [16] [9] |
| Cloud Platform Provider |  | Automatic update of a Cloud system’s SLA as a result of a change in the provider’s policy after SLA matching. | [6] |
|  | Monitor the status of the resources (dead/alive) and application components, services, and infrastructure to detect failures. | [10] [11] [6] |
|  | Monitor the performance collecting resource information for each service. | [6] |
|  | Establish federations/ collaborations of Clouds | [2] [14] |
|  | Manage the access to the resources/services | [2] |
|  | Publish service offerings in a service catalog (service characteristics, policies, application platform availability and performance) | [11] [4] [2] |
|  | Manage the service offering | [11] |
|  | Manage the contracts | [11] |
|  | Charge for provided services (billing functions, invoices, settlement, etc.) | [11] |
|  | Release resources | [6] |

## Non-Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **Requirement type** | **Requirement** | | **Reference** |
| Security |  | Data and applications security | [13] [3] [14] [7] [4] |
|  | Encryption | [10] |
|  | User authentication and single sign-on, authorization and role-based access control, security proof | [9] [8] [6] [10] |
|  | Tailored security policy | [9] |
|  | Regulatory security controls | [10] |
|  | Secure execution of monetary transactions | [14] |
|  | Licensing and security issues to span different Clouds | [11] |
| Scalability |  | Scalable infrastructure provisioning as needed | [14] [4, 14] [8] [7] [9] [2] [3] |
|  | Scalable deployment to multiple Clouds and between Clouds and on-premise systems | [13] [6] |
| Interoperability |  | Supporting commonly used standards, standard syntax, open APIs, widely available tools, technologies, methodologies, and best practices. | [1] [13] [9] [19] |
|  | Supporting abstraction (it hides many details of systems infrastructure and application infrastructure from developers and their applications) | [2] |
|  | Uniform service representation | [12] |
|  | Uniform service description (SLA offering), using standard formats | [12] [6] |
|  | SLAs with clear policies and guidelines for maintenance and version management of the platform and policies for version compatibility for APIs between the platform and the application. | [4] |
|  | Transfer monitoring, logging, auditing and control functions to the new provider by means of commonly defining formats | [19] [6] |
|  | Verify that the migrated services or applications are operating correctly in the new provider | [19] |
|  | Describe information in standard manner in order to be able to manage resources over multiple Cloud systems in an integrated manner. | [6] |
|  | A simplified interface/API with well-understood service offerings, pricing and contracts | [11] |
| Operational |  | Support self-services  Self-service provisioning, management and scaling | [15] [7] [11] |
|  | A self-service portal which exposes a well-defined set of services in a highly automated fashion | [11] |
|  | Automatic and seamless deployment | [1], [5], [15] |
|  | Intercept component-to-component communication. | [8] [12] |
|  | The development platform and the development tools are hosted in the Cloud and accessed through a browser | [5] |
|  | It provides a presentation interface (menu and navigator, user controls, display and rendering, reporting) | [10] [11] |
| Usability |  | Ease-of-Use | [3] |
| Availability |  | It should be accessible and available (at acceptable service levels). | [13] [3] |
| Reliability |  | Reliable | [13] [14] |
|  | It should provide metering, tracking and reporting items related to the service-level agreement, such as usage, availability, number of failures, and mean time to respond to and fix problems | [10] [11] |
|  | It can be accessed by multiple users at the same time (multi-tenant) | [11] |
|  | It should provide highly efficient service delivery | [11] |
| Other |  | Ensuring developer choice on languages, runtimes and tools | [1] [2] |

# References

[1] K. Mackie, "Microsoft lays out 'Open Cloud' vision " in [*http://gcn.com/articles/2010/08/10/ecg-microsoft-lays-out-its-open-cloud-vision.aspx*](http://gcn.com/articles/2010/08/10/ecg-microsoft-lays-out-its-open-cloud-vision.aspx), August 2010.

[2] R. Hat, "Red Hat PaaS: Bringing Open Choice & Application Portability to the Cloud," September 28, 2010.

[3] M. Subraya, "Graduating Cloud to the Enterprise: Platform-as-a-Service," in *PrudentCloud - Strategies for SaaS, Cloud Computing, Governance and Open Source*, 2010.

[4] Cisco, "Planning the Migration of Enterprise Applications to the Cloud," 2010.

[5] C. Keene, "What Is Platform as a Service (PaaS)?," *KeeneView Blog,* 2009.

[6] GICTF, "Use Cases and Functional Requirements for Inter-Cloud Computing," 2010.

[7] X. Chen, G. B. Wills, L. Gilbert, and D. Bacigalupo, "Using Cloud for research: A technical review," 2010.

[8] W. Vambenepe, "Desirable technical characteristics of PaaS," *William Vambenepe's blog - IT management in a changing IT world,* 2009.

[9] S. Kang, J. Myung, J. Yeon, S. Ha, T. Cho, J. Chung, and S. Lee, "A General Maturity Model and Reference Architecture for SaaS Service," in *Proceedings of 15th International Conference on Database Systems for Advanced Applications (DASFAA 2010)*, Tsukuba, Japan, 2010, pp. 337-346

[10] Intel, "Architecting Software as a Service for the Enterprise," 2009.

[11] M. Dodani, "The Practice of “Architecting” Cloud Solutions," *Journal of Object Technology,* vol. 9, 2010.

[12] "D10.1 Requirements Report, WP10 Requirements and Assessment Criteria NESSI Open Framework - Reference Architecture, Project URL <http://www.nexof-ra.eu.>"

[13] S. Technology, "Development in the Cloud: A Framework for PaaS and ISV Flexibility," July 2010.

[14] M. Dave, "Defining Platform-As-A-Service, or PaaS," *Bungee Connect Developer Network,* 2008.

[15] C. T. Lv, Q. Li, Z. Lei, J. J. Peng, and W. Zhang, " PaaS: A Revolution for Information Technology Platforms.," in *The 2010 International Conference on Educational and Network Technology (ICENT 2010)* Qinhuangdao, China, , 2010.

[16] V. Gonçalves, "Adding Value to the Network: Exploring the Software as a Service and Platform as a Service Models for Mobile Operators," in *MOBILWARE Workshops*. vol. 12, 2009, pp. 13-22.

[17] Cloud-Syandards.org, "Cloud Standards Overview," 2010.

[18] Cisco, "Cisco Cloud Computing - Data Center Strategy, Architecture, and Solutions Point of View White Paper for U.S. Public Sector

" 2009.

[19] C. S. Alliance, "Security Guidance for Critical Areas of Focus in Cloud Computing V2.1," April 2009.