**종합설계** Ⅱ

**교육용 클라우드 서비스**

**제안서**

|  |  |
| --- | --- |
| **프로젝트 명** | **오픈스택 기반 클라우드 플랫폼 구축** |
| **팀명** | **이러다 클라우드** |
| **문서 제목** | **제안서** |

|  |  |
| --- | --- |
|  | 박준상(팀장) |
| 심홍철 |
| 권성주 |
| 조정우 |

**CONFIDENTIALITY/SECURITY WARNING**

이 문서에 포함되어 있는 정보는 단국대학교 공과대학 소프트웨어학과 개설 교과목 종합설계Ⅱ 수강 학생 중 프로젝트 “오픈스택 기반 클라우드 플랫폼 구축”을수행하는 팀 “이러다 클라우드”의 팀원들의 자산 입니다. 단국대학교 소프트웨어학과 팀 “이러다 클라우드”의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다.

**문서 정보**

|  |  |
| --- | --- |
| **Filename** | 계획서-오픈스택 기반 클라우드 플랫폼 구축.docx |

**목차**

**1** **프로젝트 개요**................................................................................................................................................................................... 4

1.1 프로젝트 추진배경........................................................................................................................................................ 4

1.2 현황조사.......................................................................................................................................... 5

1.3 구현 목표.......................................................................................................................................... 6

**2** **추진 전략**............................................................................................................................................................ 7

2.1 선행과제............................................................................................................................................................................. 7

2.2.1 오픈스택 인프라 개발환경 구축................................................................................................... 7

2.2.2 PYTHON 프레임워크 분석............................................................................... 8

2.2 구현 내용.............................................................................................................................................................. 10

2.2.1 결과물 목록 및 상세 사양.............................................................................................................. 10

2.2.2 시스템 구성도......................................................................................................................... 11

**3** **기대효과 및 향후 과제**................................................................................................................................................................. 13

4 **프로젝트 비용**.................................................................................................................................................................. 14

5 **개발 일정**............................................................................................................................................. 15

**1 프로젝트 개요**

**1.1추진 배경**

오픈스택은 오픈소스 클라우드 컴퓨팅 소프트웨어로 2010년 나사와 랙스페이스 (Rackspace)의 공동프로젝트로 시작되었고, 현재 HP, AT&T, Intel을 비롯한 우리 나라 대기업들의 지원을 받고 있으며 전 세계에서 가장 큰 오픈 소스 커뮤니티를 보유하고 있는 클라우드 프로젝트입니다. 이러한 발전 가능성과 다양한 장점 때문에 오픈 스택을 기반으로 해당 프로젝트가 구축될 것 입니다.

* Open Source
	+ 라이선스 비용이 없어서 자유롭게 사용할 수 있습니다.
	+ 많은 개발자와 기업들이 프로젝트에 참여하고 있기 때문에 충분한 레퍼런스가 존재합니다.
	+ 접근이 용이하기 때문에 분석 및 개발이 비교적 쉽게 진행됩니다.
* Big Community
	+ 세계적으로 많은 관심을 받고 있는 프로젝트이기 때문에 지속적으로 업데이트가 이루어집니다.
	+ 버그 리포팅이 빠르고 해결책이 빠르게 업데이트 됩니다.
	+ 많은 지원과 관심 때문에 다른 클라우딩 소프트웨어와 비교해 실제 제품화에 유리하다..
* 다양한 하이퍼바이저 지원
	+ 오픈스택은 다양한 하이퍼바이저(ESXi, Hyper-V, KVM, Xen 등)들을 지원하기 때문에 편리합니다.
	+ 용도에 맞는 다양한 아키텍처를 구성할 수 있습니다.
	+ 다른 종류의 하이퍼바이저가 설치된 호스트와 연결하고 관리하는 것이 비교적 용이합니다.
* 8개의 독립적 컴포넌트로 구성
	+ 오픈스택은 8개의 독립적인 컴포넌트로 구성되어 있기 때문에 관리가 편리합니다.
	+ 실제 시스템을 운영하면서 안정성을 확보하기가 비교적 용이합니다.
	+ ****시스템 장애시에 원인을 발견하고 분석하기가 용이합니다.

**1.2현황 조사**

 현재 클라우드 컴퓨팅 소프트웨어를 기반으로 출시된 대표적인 서비스가 존재하고, 해당 프로젝트를 통해 구축될 클라우드 플랫폼 서비스와 국, 내외 대기업의 현황을 비교하여 목표를 설정하는 기준으로 설정합니다. 또한, 현재 향후 클라우드 시장의 트렌드와 전망을 분석하여 해당 프로젝트의 특장점을 세우고, 각각의 기업들이 직면하고 있는 문제점을 파악하여 사전에 대처 전략을 수립합니다. 현재 클라우드 시장에서 높은 점유율을 차지하고 있는 Amzon,KT,LG CNS를 표본으로 서비스명, 서비스 유형, 가격정책, 특징에 대한 항목들을 조사합니다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 회사명 | 서비스명 | 서비스 유형 | 가격정책 | 특징 |
| Amazon | AWS(Amazon Web Servies) | PaaS | 실제 사용률기반 | 서비스가 다양하고요금제가 굉장히 세분화 되어있음 |
| KT | G-Cloud | PaaS | 어떤 종류의Cloud 서버를선택하는 지에 따라 가격을 측정 | 공공기관에 최적화 되어있는B2B 전용 Cloud |
| LG CNS | LG CNS COUD | PaaS | 월 단위 결제 시스템 | OpenStack기반B2B, B2C클라우드 서비스 |

* AWS(Amazon) : 아마존 웹 서비스는 다른 웹 사이트나 클라이언트측 응용 프로그램에 대해 온라인 서비스를 제공하고 있습니다. 다른 개발자가 사용 가능한 기능을 제공하는 플랫폼을 제공하는 PaaS형태의 서비스 입니다. 현재 아마존이 제공하는 클라우드 제품과 서비스는 컴퓨트, 콘텐츠 딜리버리, 데이터베이스, 개발과 관리, 전자상거래, 메시징, 모니터링, 네트워킹, 결제와 과금, 스토리지, 지원, 웹트래픽, 워크포스 등이 있고, 각 분야에서 제공되는 기능들도 하루가 다르게 확장되고 있습니다.
* G-CLOUD(KT) : 자체적으로 개발한 CLOUD 서비스를 제공하고 있으며 공공기간을 대상으로 B2B 비즈니스를 중점적으로 진행합니다. 백업정책, 네트워크 보안 장비, 웹을 통한 UI등의 다양한 지원 정책을 운영하고 있고 강력한 보안을 제공함으로 타 업체와의 경쟁력을 확보하고 있습니다
* LG CNS CLOUD(LG CNS) : 국내에 얼마되지 않는 OpenStack 기반의 CLOUD 서비스를 제공하고 있으며 소스 분석을 통해 일부 기능들을 개발하여 추가적으로 제공하고 있습니다. 가장 최신 CLOUD 서비스이며, 유일하게 B2B와 B2C를 동시에 제공한다는 특징을 가지고 있습니다.

**1.3구현 목표**

현존하는 타 클라우드 서비스와 비교, 분석한 데이터를 바탕으로 프로젝트의 목표와 기대효과를 설정합니다. 다양한 상황에 대처할 수 있는 유동적인 클라우딩 환경을 구축함과 동시에 오픈스택의 혁신 기술을 적극적으로 이용하는 서비스를 구성합니다. 또한, 안정적인 운영을 위한 아키텍처 구성과 고객의 요구 적합한 환경으로 고객의 편의성과 신뢰성을 높이고, 최상의 서비스 만족감을 제공합니다. 향후 다양한 컨텐츠를 클라우드 플랫폼에 탑재하여 사용자의 실질적인 사용을 목표로 합니다.

* **구현 목표 : 교육용 클라우드 플랫폼 서비스 구축**
1. 리눅스에 최적화 된 환경을 설정합니다.
2. 커리큘럼에 맞는 학습 과제를 내장하고 있습니다.
3. 다양한 테스트 케이스를 통해 정합성 검증을 수행합니다.
4. 관리자가 과제를 수행하는 사용자를 손쉽게 점검할 수 있습니다.
5. KeyStone을 통해 보안성을 확보하며 사용자에게 Root권한을 허용합니다.
6. Dashboard를 통해 관리자가 손쉽게 시스템에 접근합니다.
7. Obejct Storage를 사용하여 빠르고 안정적인 서비스를 지원합니다.
8. 유동적으로 사용자에게 리소스를 할당합니다.
9. 리소스 가상화를 통해 효율적으로 관리합니다.
10. 오픈스택의 8개 컴포넌트가 유기적으로 동작합니다.

**2 추진 전략**

**2.1 선행 과제**

******2.1.1 오픈스택 개발 인프라 구축**

**Figure. Cloud 서비스 용도로 사용될 서버 사양**

오픈스택 기반의 클라우딩 플랫폼을 구축하기 위해서 크게 디바이스와 물리적 리소스 형태의 인프라가 필요합니다. 먼저 충분한 성능의 하드웨어를 확보하고 가상화 작업을 통해 리소스를 효율적으로 분산합니다. 따라서 사용자에게 원하는 작업을 원활히 수행할 수 있는 안정적인 VM 환경을 제공합니다.

**Figure . OpenStack을 위한 물리적 자원**

디바이스를 구동하는데 필수적인 전기, 네트워크, 스위치, 모니터링 수행 인력 등과 같은 물리적 자원과 더불어 디바이스를 안전하게 보관할 수 있는 장소를 확보합니다. 알맞은 자원들을 디바이스에 할당하고, 이를 통해 물리적인 자원으로 인해 발생하는 장애를 사전에 방지하고 안정적인 운영을 수행합니다.

**2.1.2 오픈스택 프레임워크 분석**

오픈스택은 Python으로 개발된 소프트웨어입니다. 따라서 오픈스택에서 제공하는 SDK또한 Python으로 제작되어 있고, 이를 이용해서 자원을 생성하고 관리할 수 있도록 구성되어 있습니다. 또한, 클라우드 내의 모든 API가 동작할 때 장고 프레임워크를 기반으로 구현된 Python프로그램이 이 수행되고, 내부적으로 구현되어있는 Python의 객체들을 호출하면서 클라우드 내에 필요한 작업들을 수행할 수 있게 합니다. 때문에 Pyhon을 사전에 학습하는 것은 오픈스택을 구현하는데 있어서 필수적인 요소입니다.

* **Python 구현 방법**
1. **컴포넌트 별 Python 구조 분석**
* 각 컴포넌트에서 Python 어떤 방식으로 동작하는지 조사하여 발표합니다.
1. **컴포넌트 별 API 조사**
* 각 컴포넌트에서 Python으로 구현된 API가 어떻게 동작하는지 코드를 통해 분석하여 발표합니다.
1. **프로젝트 서비스 구현**
* 프로젝트 아키텍처 구성에 필요한 추가적인 부분을 직접 구현합니다.
1. **통합 작업**
* 팀원 별로 구현된 기능들을 통합 및 테스트 하는 작업을 수행합니다.
* **Django-OpenStack**

****오픈 스택에서 대부분의 컴포넌트가 Python으로 쓰여진 장고 프레임워크로 이루어져 있고, 해당 프레임워크의 특징을 잘 나타내고 있습니다.

**Figure . 장고의 대표적인 특징 세가지**

**2.2 구현 내용**

**2.2.1 결과물 목록 및 상세 사양**

오픈스택은 컴퓨터,오브젝트 스토리지, 이미지, 인증서비스 등이 유기적으로 연결되어 하나의 커다란 클라우드 컴퓨팅 시스템을 구축합니다.

|  |  |  |  |
| --- | --- | --- | --- |
| **대분류** | **소분류** | **기능** | **내용**  |
| Basic service | Dashboard | Horizon | 오픈스택 시스템 관리자를 위한 Web UI 이며 관리자는 이 UI를 통해 오픈스택을 관리할 수 있다. |
| Compute | Nova | 오픈스택 시스템에 연결된 하이퍼바이저에 행하는 모든 행위를 시행/관리하는 역할을 하는 사실 상 오픈스택에서 가장 중요한 역할을 하는 소프트웨어 |
| Networking | Neutron | 네트워킹 서비스인 Neutron은 오픈스택 내의 IP address들과 네트워킹을 관리하는 기능을 제공하는 소프트웨어 |
| Storage Service | Object Storage | Swift | 오픈스택은 두 가지 타입의 스토리지 서비스를 제공하는 그 중에 하나인 Object Storage Swift 이다. 오브젝트 스토리지에 대해 간략하게 설명하면 데이터 스위프트를 오브젝트 단위로 저장하는 방식 |
| Block storage | Cinder | Swift와 함께 오픈스택이 제공하는 또 다른 스토리지 서비스는 Block storage –Cinder 입니다. 이 스토리지 서비스는 우리가 일반적으로 사용하는 컴퓨터의 하드디스크에서 사용되고 있는 방식으로 데이터를블록 단위로 저장하는 방식입니다.. |
| Shared Service | Identity Service | Keystone | 오픈스택 내의 증명과 허가 기능을 제공하는 소프트웨어 |
| Image Service | Glance | VM으로 생성될 이미지 파일들을 관리하는 기능을 제공하는 컴포넌트 입니다. 이미지 서비스를 이용하게 되면 자주 생성하는 VM같은 경ㅇ우에 매번 이미지 파일을 업로드하지 않고 업로드한 이미지 파일을 템플릿처럼 사용할 수 있습니다. |
| Telemetry | Ceilometer | 텔레메드라는 사용자의 사용량을 측정에서 알려주는 기능을 제공하는 컴포넌트로 서비스 프로바이더 회사들에게 유용한 기능을 제공하는 소프트웨어 |
| Support Service | Orchestration | Heat | 미리 작성된 스크립트와 준비된 템플릿을 이용하여 자동으로 개발 인프라를 구축할 수 있는 기능을 제공하는 소프트웨어 |
| Databse Service | Trove | 이름 그대로 오픈스택 시스템 상에서 데이터 베이스 서비스를 구축할 수 있게 도와주는 기능을 제공하는 소프트웨어 |

**2.2.2 시스템 구성도**

OpenStack은 총 9개의 서비스(Horizon, Nova, Glance, Swift, Ceilometer, Keystone, Cinder, Neutron, Heat)로 구성되어있습니다. 그리고 이러한 서비스들은 Neutron을 사용할 경우 3개의 node(Controller, Network, Compute)로 나눠집니다.

Compute Node에는 서비스를 관리하는 서비스들을 설치합니다. 데이터베이스는 MySQL이나 MariaDB를, 메시지 서비스는 RabbitMQ 또는 Qpid를 설치합니다. 기본적인 구성을 위한 서비스로는 인증을 담당하는 Keystone, 이미지 서비스인 Glance, Compute Node를 관리하는 Nova, Network 서비스를 관리하는 Neutron Server와 ML2 Plugin, Dashboard 서비스인 Horizon이 필요합니다. 추가적인 구성요소로는 필요에 따라서 Block Storage 서비스인 Cinder, Object Storage 서비스인 Swift, 데이터베이스 서비스인 Trove, Orchestration을 담당하는 Heat, 텔레미터 서비스인 Ceilometer를 설치할 수 있습니다.



Network Node에는 Neutron 서비스의 기본 플러그인 ML2을 설치하고, L2 에이전트인 OpenvSwitch, L3 에이전트, DHCP 에이전트를 설치합니다.

마지막으로 Compute Node에는 OpenStack의 기본 Hypervisor인 KVM이나 QEMU를 설치하며, Network 서비스로 ML2 플러그인, L2 에이전트인 OpenvSwitch를 설치합니다. 이때, Controller Node에 Ceilometer를 설치했다면, Compute Node에는 Ceilometer 에이전트를 설치해야 합니다.

Network Interface Card는 Controller Node일 경우 모니터링 및 관리용으로 한 개가 필요합니다. Network Node일 경우 Controller Node와 연결, Compute Node간의 연결, 외부 Network와 연결, 총 세 개가 필요합니다. Compute Node는 Controller Node와의 연결, Network Node와 Compute Node 간 연결, 총 2개가 필요합니다.

**3 기대효과 및 향후 과제**

* **기대효과**
	+ 맞춤형 클라우드 플랫폼

- 다양한 분야에서 그 분야에 최적화된 클라우드 플랫폼을 서비스합니다. 리눅스에 대한 최적의 클라우드 플랫폼을 기대할 수 있습니다.

* + 작업 효율성 증가

- 맞춤형 플랫폼을 통해 시간과 비용을 절감하여 효율성을 증가시킬 수 있습니다.

* + 교육의 질적 향상

- 효율성 증가로 인한 시간, 자원적 여유가 생기며 그 결과 교육의 질적 향상을 기대할 수 있습니다.

* + 양질의 개발자 양성

- 교육의 질적 향상으로 양질의 개발자 양성을 기대할 수 있습니다.

* **향후 과제**



* **다양한 분야에 대한 맞춤형 서비스**

- 증가하는 IOT분야에 맞춰 맞춤형 서비스 준비

- 다양한 분야에 대한 전문지식 및 전문가 확보

* **보안 강화 필요**

- 발전하는 해킹기술에 맞춰 보안강화

* **추가적인 컨텐츠 확보**

- 기존 분야에대한 추가적인 컨텐츠 확보

- 추가적인 분야에대한 컨텐츠 확보

* **오픈스택의 미사용기능 추가 연구개발**
* **다양한 분야에 대한 맞춤형 서비스**

- 증가하는 IOT분야에 맞춰 맞춤형 서비스 준비

**4 프로젝트 비용**

|  |  |
| --- | --- |
| **항목** | **예상치(만원)** |
| IDC-서버 HP DL360 10대 | 3500 |
| -서버 유지 비용-기타비용 별도 | 월100 |
| Device- Controller Node, Network Node |  |
| -Compute Node(Octa Core 이상 200만원) | 400 |
| -기타 광고 및 마케팅 비용 | 월200 |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| 합 | 4200 |

**5 개발 일정**

**12.14 – 12.15**

2016

Sep

Nov

Dec

**조 결성**

9.07

**1차 주제선정**

9.19

**설계 시작**

9.26

**Survey**

10.19

**제작 시작**

9.29

**제작완료**

11.23

**최종 발표**

12.14 – 12.15

**9.07 - 9.23**

18 days

**주제 선정**

**9.23 - 10.03**

**구상**

**9.26 – 11.02**

**설계**

**9.29 – 11.23**

55 days

**제작**

**11.23 – 12.07**

15 days

 **Test**

2 days

**발표 및 시연**

16 days

**최종 주제선정**

9.23

10 days

Oct

36 days