ABSTRACT

*Setup Server is Command Line Interface control panel for Linux based operating system hosted on virtual private server. Setup Server is expedient way of CLI for creating, controlling and managing HTML, PHP web site hosted on Linux/Ubuntu based virtual private server. By Setup Server, web server-APACHE, database server-MYSQL, web based database control portal-PHPMYADMIN and PHP can be managed and controlled on virtual private server. Setup Server enhances efficiency, performance, capability of web site management.*

ACKNOWLEDGEMET

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely fortunate to have got this all along the completion of our project work. Whatever we have done is only due to such guidance and assistance and we would not forget to thank them.

We owe our profound gratitude to our project guide **Prof. Rupal Parmar**, who took keen interest on our project work and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system.

We heartily thank our, **Mr. M.B. Chaudhary**, Head of Department, Information Technology, for his guidance and suggestions during this project work.

We are thankful to and fortunate enough to get constant encouragement, support and guidance from all Teaching staffs of Department of Information Technology which helped us in successfully completing our project work. Also, we would like to extend our sincere regards to all the non-teaching staff of department of Information Technology for their timely support.

Team Linvecklare,

Vikas Kapadiya

Harshil Jethava

Sanjay Rathod

TABLE OF CONTENTS

[ABSTRACT i](#_Toc448144098)

[ACKNOWLEDGEMET ii](#_Toc448144099)

[1 INTRODUCTION v](#_Toc448144100)

[1.1 PROBLEM STATEMENT v](#_Toc448144101)

[1.2 OBJECTIVES vi](#_Toc448144102)

[LIST OF TABLES vii](#_Toc448144103)

[LIST OF FIGURES viii](#_Toc448144104)

[LIST OF ABBREVIATIONS AND NOMENCLATURE ix](#_Toc448144105)

[2 Introduction 1](#_Toc448144106)

[2.1 Problem Statement 1](#_Toc448144107)

[2.2 Purpose 1](#_Toc448144108)

[2.3 System Overview 1](#_Toc448144109)

[2.4 Objectives 1](#_Toc448144110)

[2.5 Goals and Vision 2](#_Toc448144111)

[3 Background 3](#_Toc448144112)

[3.1 Linux Operating System 3](#_Toc448144113)

[3.2 Components of Linux System 4](#_Toc448144114)

[3.2.1 Linux Operating System has primarily three components 4](#_Toc448144115)

[3.2.2 Features of Linux 4](#_Toc448144116)

[3.3 Ubuntu 4](#_Toc448144117)

[3.4 Python 6](#_Toc448144118)

[3.4.1 Features of Python 7](#_Toc448144119)

[3.5 Cement Python 7](#_Toc448144120)

[4 Requirements Specification 7](#_Toc448144121)

[4.1 User Requirement 7](#_Toc448144122)

[4.2 Software Dependencies 7](#_Toc448144123)

[4.3 Performance Requirements 8](#_Toc448144124)

[4.4 Hardware Requirements 8](#_Toc448144125)

[4.5 Operating System Requirements 8](#_Toc448144126)

[4.6 Port Requirement 8](#_Toc448144127)

[4.7 Constrain & Assumption 8](#_Toc448144128)

[5 Design 9](#_Toc448144129)

[5.1 Activity Diagram 9](#_Toc448144130)

[5.2 Sequence diagram 20](#_Toc448144131)

[5.3 Use case diagram 24](#_Toc448144132)

[5.4 Data Dictionary diagram 26](#_Toc448144133)

[6 Installation and Instruction 27](#_Toc448144134)

[7 Future Work 28](#_Toc448144135)

[8 Summary 29](#_Toc448144136)

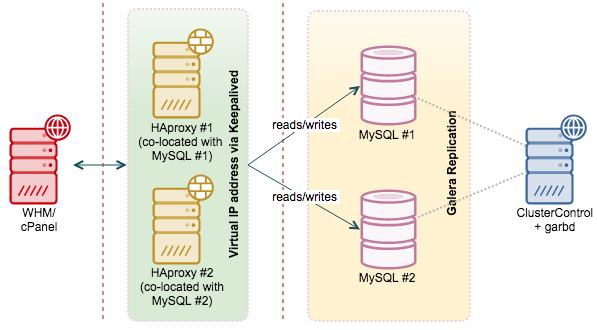
[9 References 30](#_Toc448144137)

# INTRODUCTION

## Current approach

Currently for managing PHP and HTML web site on virtual private server, cPanel is used. cPanel provides wide range of functionality for web site management.

To use cPanel, it must be installed on virtual private server on which web site is going to host. cPanel follow license policy for installation. Below figure demonstrate cPanel work methodology.



cPanel has to install on virtual private sever. Based on user can manage server through web based graphic interface.

## 

## The Problem with current approach

Thus cPanel is widely used by many web developers and IT professionals for that web hosting and web site management, it contains serval disadvantages and loopholes. Below list describe disadvantage and loopholes:

1. cPanel is meant for websites that are small to medium. It is consuming quite a lot of server’s resources as it requires at least 256MB of RAM, and this amount probably won’t even run a website getting 400 hits a day without running out of memory or running into swap space on the server. Most servers will have much more RAM than this, but a base install should not require more than 50 to 60MB of ram.
2. cPanel is software which contains vulnerabilities just like any other software. cPanel updates run nightly, however what happens if someone discovers a security risk and decides to attack server before the updates are pushed to server covering the vulnerability?
3. cPanel costs money to license. Yearly subscription of cPanel is approximately $200, which is huge for small and medium size website development firm or any individual developer.
4. Available cPanel not provide customization so you cannot install single service module or any single web service in your VPS. For some basic service or web app you have to install complete cPanel in your VPS.
5. cPanel affects servers` speed, efficiency and throughput based on web site and database size.
6. Available control panel like cPanel provide too many features which are not going to use by developer in most scenario.
7. More than 60 million websites are powered by WordPress though most of control panels are not optimize for WordPress.
8. Most control panel have already bloated software which require lots of resources, not use in general purpose.
9. Nowadays control panels are too complex for non-geek so maintain and installation of own web server is too tedious for them.

## Aim of the project

This project is first attempt to implement simple and efficient CLI of web server for user. Its aims are:

* To provide one-line solution for web server through CLI.
* Reduce effort of user /administrator for maintaining, governing web server.
* Enhance productivity of web application by providing simple and effective application based solution.
* Provide capability to entry level audience to interact with server for his/her web site.

## Scope of project

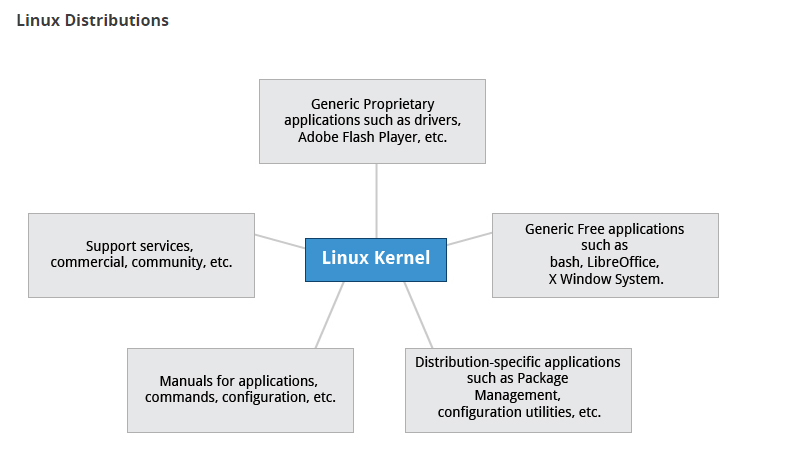
## Assumptions and constraints

# BACKGROUND

## Linux Operating System

Linux is, in simplest terms, an operating system. It is the software on a computer that enables applications and the computer operator to access the devices on the computer to perform desired functions. The operating system (OS) relays instructions from an application to, for instance, the computer's processor. The processor performs the instructed task, then sends the results back to the application via the operating system.

Explained in these terms, Linux is very similar to other operating systems, such as Windows and OS X.



But something sets Linux apart from these operating systems. The Linux operating system represented a $25 billion ecosystem in 2008. Since its inception in 1991, Linux has grown to become a force in computing, powering everything from the New York Stock Exchange to mobile phones to supercomputers to consumer devices.

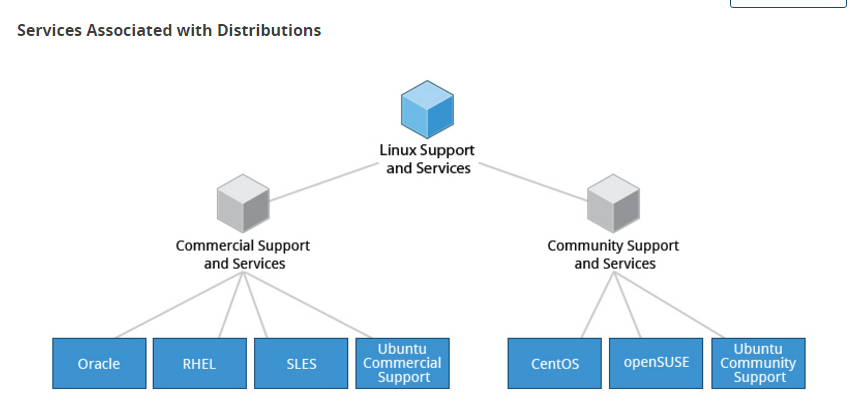
As an open operating system, Linux is developed collaboratively, meaning no one company is solely responsible for its development or ongoing support. Companies participating in the Linux economy share research and development costs with their partners and competitors. This spreading of development burden amongst individuals and companies has resulted in a large and efficient ecosystem and unheralded software innovation.

Over 1,000 developers, from at least 100 different companies, contribute to every kernel release. In the past two years alone, over 3,200 developers from 200 companies have contributed to the kernel--which is just one small piece of a Linux distribution.

This article will explore the various components of the Linux operating system, how they are created and work together, the communities of Linux, and Linux's incredible impact on the IT ecosystem.

### Flavor of Linux OS

Linux has a number of different versions to suit nearly any type of user. From new users to hard-core users, you’ll find a “flavor” of Linux to match your needs. These versions are called distributions (or, in the short form, “distros.”) Nearly every distribution of Linux can be downloaded for free, burned onto disk (or USB thumb drive), and installed (on as many machines as you like).

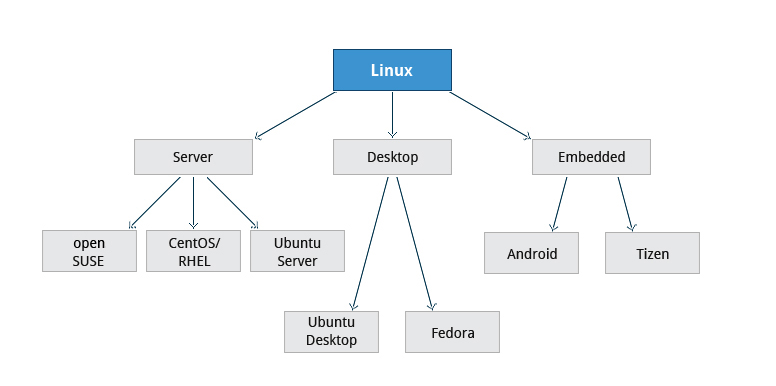


The most popular Linux distributions are:

* [Ubuntu Linux](http://www.ubuntu.com/)
* [Linux Mint](http://www.linuxmint.com/)
* [Arch Linux](https://www.archlinux.org/)
* [Deepin](http://www.linuxdeepin.com/index.en.html)
* [Fedora](http://fedoraproject.org/)
* [Debian](https://www.debian.org/)
* [openSUSE](http://www.opensuse.org/en/).

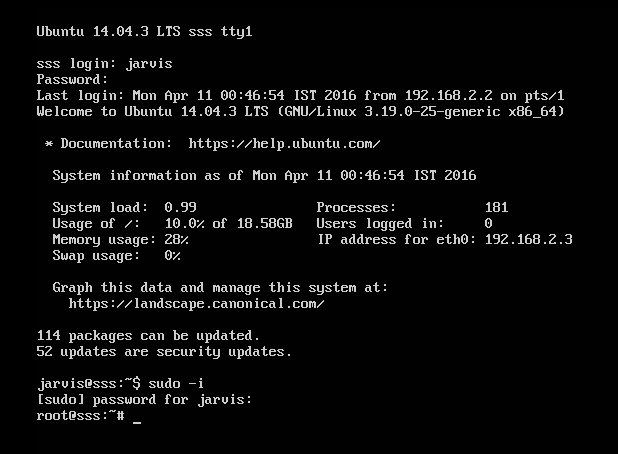
### Linux server OS

Some of the above server distributions are free (such as Ubuntu Server and CentOS) and some have an associated price (such as Red Hat Enterprise Linux and SUSE Enterprise Linux). Those with an associated price also include support.



### Ubuntu server OS

Ubuntu Server 14.04 LTS, which goes live today, is the third Ubuntu LTS release designed for cloud. After more than four years, we finally see the cloud coming of age, with OpenStack at its heart. Ubuntu has for a long time been the platform of choice for running enterprise workloads such as web infrastructure. Today, Ubuntu OpenStack sits at the heart of cloud infrastructure at some of the world’s largest and most innovative companies. In fact, we are so confident of the maturity of OpenStack, that we’ve decided to support the Icehouse release for five years, just like the Ubuntu Server release it comes with. So, 14.04 is effectively both Ubuntu LTS and OpenStack LTS together.



* Supported for five years by Canonical
* Certification as a guest on AWS, Microsoft Azure, Joyent, IBM and HP Cloud
* Updates to Tomcat (v7), Postgresql (v9.3), Docker v(0.9), Puppet (v3.0), Qemu (v2.0), Libvirt (v1.2), LXC (v1.0) and MySQL (v5.5)
* The first production release of Open vSwitch 2.0 with full kernel integration
* Runs on x86, x86-64, ARM v7, ARM64 and Power

### Debain server OS

An operating system is the set of basic programs and utilities that make your computer run. At the core of an operating system is the kernel. The kernel is the most fundamental program on the computer and does all the basic housekeeping and lets you start other programs.

Debian systems currently use the Linux kernel or the FreeBSD kernel. Linux is a piece of software started by Linus Torvalds and supported by thousands of programmers worldwide. FreeBSD is an operating system including a kernel and other software.

However, work is in progress to provide Debian for other kernels, primarily for the Hurd. The Hurd is a collection of servers that run on top of a microkernel (such as Mach) to implement different features. The Hurd is free software produced by the GNU project.

A large part of the basic tools that fill out the operating system come from the GNU project; hence the names: GNU/Linux, GNU/kFreeBSD, and GNU/Hurd. These tools are also free.

Of course, the thing that people want is application software: programs to help them get what they want to do done, from editing documents to running a business to playing games to writing more software. Debian comes with over 43000 packages (precompiled software that is bundled up in a nice format for easy installation on your machine), a package manager (APT), and other utilities that make it possible to manage thousands of packages on thousands of computers as easily as installing a single application. All of it free.

It's a bit like a tower. At the base is the kernel. On top of that are all the basic tools. Next is all the software that you run on the computer. At the top of the tower is Debian — carefully organizing and fitting everything so it all works together.

## Python

Python is a widely used general-purpose, high-level programming language. Its design philosophy emphasizes code readability, and its syntax allows programmers to express concepts in fewer lines of code than would be possible in languages such as C++ or Java. The language provides constructs intended to enable clear programs on both a small and large scale.

Python supports multiple programming paradigms, including object-oriented, imperative and functional programming or procedural styles. It features a dynamic type system and automatic memory management and has a large and comprehensive standard library.

Python interpreters are available for installation on many operating systems, allowing Python code execution on a wide variety of systems. Using third-party tools, such as Py2exe or Pyinstaller. Python code can be packaged into stand-alone executable programs for some of the most popular operating systems, allowing the distribution of Python-based software for use on those environments without requiring the installation of a Python interpreter.

### Python cement

Cement is an advanced CLI Application Framework for Python. Its goal is to introduce a standard, and feature-full platform for both simple and complex command line applications as well as support rapid development needs without sacrificing quality. Cement is flexible, and its use cases span from the simplicity of a micro-framework to the complexity of a mega-framework. Core features include (but are not limited to):

* Pieces of the framework are customizable via handlers/interfaces
* Extension handler interface to easily extend framework functionality
* Config handler supports parsing multiple config files into one config
* Argument handler parses command line arguments and merges with config
* Log handler supports console and file logging
* Plugin handler provides an interface to easily extend your application
* Hook support adds a bit of magic to apps and also ties into framework
* Handler system connects implementation classes with Interfaces
* Output handler interface renders return dictionaries to console
* Cache handler interface adds caching support for improved performance
* Controller handler supports sub-commands, and nested controllers
* Zero external dependencies of the core library
* Tested on Python 2.6, 2.7, 3.2, 3.3, and 3.

## Apache web server

The Apache HTTP Server, colloquially called Apache. is the world's most used web server software. Originally based on the NCSA HTTPd server, development of Apache began in early 1995 after work on the NCSA code stalled. Apache played a key role in the initial growth of the World Wide Web,[4] quickly overtaking NCSA HTTPd as the dominant HTTP server, and has remained most popular since April 1996. In 2009, it became the first web server software to serve more than 100 million websites.[5]

Apache is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation. Most commonly used on a Unix-like system (usually Linux), the software is available for a wide variety of operating systems besides Unix, including eComStation, Microsoft Windows, NetWare, OpenVMS, OS/2, and TPF. Released under the Apache License, Apache is free and open-source software.

## PHP

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. Originally created by Rasmus Lerdorf in 1994,[3] the PHP reference implementation is now produced by The PHP Group.[4] PHP originally stood for Personal Home Page,[3] but it now stands for the recursive backronym PHP: Hypertext Preprocessor.

* Scalar type hints
* Return type declarations
* Anonymous classes
* The Closure::call() method
* Generator delegation
* Generator return expressions
* The null coalesce operator
* The space ship operator
* Throwables
* Level support for the dirname() function
* The Integer division function
* Uniform variable syntax

## MariaDB

MariaDB is one of the most popular database servers in the world. It’s made by the original developers of MySQL and guaranteed to stay open source. Notable users include Wikipedia, Facebook and Google.

MariaDB turns data into structured information in a wide array of applications, ranging from banking to websites. It is an enhanced, drop-in replacement for MySQL. MariaDB is used because it is fast, scalable and robust, with a rich ecosystem of storage engines, plugins and many other tools make it very versatile for a wide variety of use cases.

MariaDB is developed as open source software and as a relational database it provides an SQL interface for accessing data. The latest versions of MariaDB also include GIS and JSON features.

## Phpmyadmin

phpMyAdmin is a free software tool written in [PHP](http://php.net/), intended to handle the administration of [MySQL](http://mysql.com/) over the Web. phpMyAdmin supports a wide range of operations on MySQL and MariaDB. Frequently used operations (managing databases, tables, columns, relations, indexes, users, permissions, etc) can be performed via the user interface, while you still have the ability to directly execute any SQL statement.

* Intuitive web interface
* Support for most MySQL features:
* browse and drop databases, tables, views, fields and indexes
* create, copy, drop, rename and alter databases, tables, fields and indexes
* maintenance server, databases and tables, with proposals on server configuration
* execute, edit and bookmark any SQL-statement, even batch-queries
* manage MySQL user accounts and privileges
* manage stored procedures and triggers
* Import data from CSV and SQL
* Export data to various formats: CSV, SQL, XML, PDF, ISO/IEC 26300 - OpenDocument Text and Spreadsheet, Word, LATEX and others
* Administering multiple servers
* Creating graphics of your database layout in various formats
* Creating complex queries using Query-by-example (QBE)
* Searching globally in a database or a subset of it

## Web hosting

A web hosting service is a type of Internet hosting service that allows individuals and organizations to make their website accessible via the World Wide Web. Web hosts are companies that provide space on a server owned or leased for use by clients, as well as providing Internet connectivity, typically in a data center. Web hosts can also provide data center space and connectivity to the Internet for other servers located in their data center, called colocation, also known as Housing in Latin America or France.

Hosting options available are:

* Free Hosting
* Shared Hosting
* Dedicated Hosting
* Collocated Hosting

### Free Hosting

Free web hosting can be a good choice when you just want to build a non-critical website for fun. Very often in a free hosting environment, connection speed is slow, website can be down frequently, and advertising banners is automatically added to your website. Some companies require you to purchase your domain name to receive free hosting services from them, while others offer you a free subdomain under them, such as [yourname.webhost.com]. Be careful as you will not be able to transfer these free subdomains. 

### Shared Hosting

In a shared hosting environment, your and other website owners shared one server. This includes sharing the physical server and the software applications within the server. Shared hosting services are affordable because the cost to operate the server is shared between you and these other owners. There are, however, a number of down sides, such as being slower. 

### Dedicated Hosting

In a dedicated hosting environment, you have the entire web server to yourself. This allows for faster performance, as you have all the server’s resources entirely, without sharing with other website owners. However, this also means that you will be responsible for the cost of server operation entirely. This is a good choice for websites that requires a lot of system resources, or need a higher level of security. 

### Collocated Hosting

In this type of hosting, you will purchase your own server and have it housed at a web host’s facilities. You will be responsible for the server itself. An advantage of this type of hosting service is you have full control of the web server. You can install any scripts or applications you need.

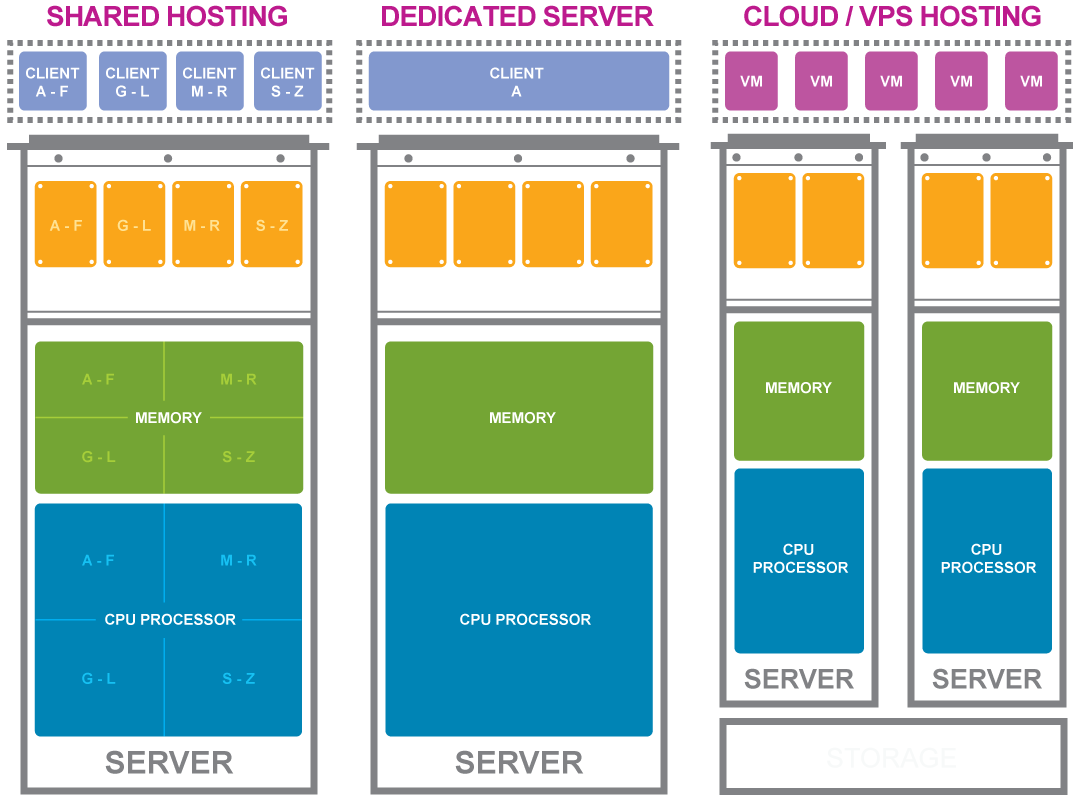
## Virtual Private Server

A virtual private server (VPS) is a virtual machine sold as a service by an Internet hosting service.

A VPS runs its own copy of an operating system, and customers have superuser-level access to that operating system instance, so they can install almost any software that runs on that OS. For many purposes they are functionally equivalent to a dedicated physical server, and being software-defined, are able to be much more easily created and configured. They are priced much lower than an equivalent physical server, but as they share the underlying physical hardware with other VPSs, performance may be lower, and may depend on the workload of other instances on the same hardware node.

### Comparison of different VPS

Now that you’ve got a basic understanding of how dedicated servers and virtual private servers work, let’s dig into the differences between these hosting options.



#### **Price**

The most noticeable difference between a dedicated server and VPS is the price. For example, even the most expensive Linux-based VPS may cost less per month than the least expensive dedicated Linux server — even with identical configurations. Which brings to mind the second difference between the two.

Configuration

You must consider configuration when deciding between dedicated server and VPS options. Hosting providers generally provide the capability for a lot more horsepower in a dedicated server than with a virtual server. Take GoDaddy for example. At the time of this writing, its largest VPS can have 8 GB of virtual memory available, while its largest dedicated server can have 32 GB of real memory.

If your business requires memory-intensive server applications — such as applications that convert graphics or videos in real time, run Big Data-style analytics, or handle many encrypted sessions with end users — you’re going to want the extra memory. That’s where making the wrong hosting choice could have an impact.

Network bandwidth and physical storage

Likewise, you should also think through network bandwidth and physical storage differences. A VPS shares the server’s storage space and network bandwidth with all of the other virtual servers on that piece of hardware; dedicated servers don’t. In fact, they have complete ownership of their bandwidth.

Then there’s performance. Both VPS and dedicated servers offer high performance, but dedicated may be more predictable. Reason being, all of the processing power, memory, storage, and bandwidth are working on your tasks. On a VPS, you will generally have all the performance you need, but the amount of available resources depends on the resource utilization of all the other virtual machines on that physical server.

Flexibility

What you gain in bandwidth and performance on a dedicated server, you may trade off when it comes to flexibility. That’s where VPS rules. A big benefit to a VPS is the ease with which you can make changes to its physical configuration. Why? It’s virtual. Need more storage, bandwidth, or memory? Just contact your hosting provider, and chances are, a service rep can assign the change in short order.

By contrast, making such changes to a dedicated server (a genuine hardware device) requires an actual human being to assist, or may require a migration to a different server. And that is more difficult, and therefore, may be more expensive.

Managed hosting services

Company capabilities also matter — particularly when it comes to a potential scenario of migrating from one type of hosting to another. Most small- and mid-sized businesses benefit greatly from managed services, where the hosting provider takes care of many server management tasks on your behalf. Conversely, if you decide to manage your servers yourself, it’s on you to migrate your applications and data, test the new setup thoroughly, and manage all aspects of the experience.

### Advantages of VPS over web hosting

Over the past several years, virtual private servers, or VPS hosting has become extremely popular. As these virtualization technologies increase in performance and functionality, this is only expected to continue. There are a number of reasons why an organization would choose to host their website or application on a VPS. From a lower carbon footprint to having the ability to scale quickly and painlessly, it makes sense for many organizations. Here are some reasons why VPS hosting will only continue to get more popular:

More stability and reliability for basic web hosting needs

Shared web hosting will soon be a thing of the past. With many hosting companies over selling their servers and piling on as many as thousands of customers on to the same web server, the reliability of their services will soon diminish.

When hosting on a shared server, your website’s uptime and performance can be impacted by other websites on the same server. What this means is that if your web server happens to also host a 12 year old wanna-be programmer that happens to crash the server, your website will suffer from this as well. It’s important to ask yourself if you are willing to take those kinds of risks with your websites, especially if it is used primarily for business.

More control compared to shared hosting

Another benefit of hosting on a VPS package is that you get complete root access to your environment. This way, if you need a custom software package installed, you can do so without having to wait for your hosting provider to support it. Shared web servers are typically optimized for security and performance as best as possible, and this means that there are many popular software packages that are not support due to their security limitations. Having your own virtual environment allows you to bypass all of those issues.

environmentally friendly, and more efficient use of resources

Green hosting and the use of eco-friendly technologies has been gaining a lot of popularity over the past few years. It is important to do your part to make sure your carbon footprint is as little as possible. VPS Hosting can help you achieve this. With dedicated server hosting you are taking all the resources of a server – which means you are the only person benefiting from that server’s power consumption. However, with a virtual private server, a large dedicated server is sliced or divided into many different virtual environments. This way, many more people shared the resources of that physical server.

Easy scalability when needed

Some websites are established and do not expect much variance in the amount of traffic they received. For them, this factor might not be that important. However, for someone who is starting a new website with hopes of growing it into something much larger, being able to scale your hosting resources without any downtime or technical issues is very important.

When you host with a VPS hosting account, your environment is hosted within what is called a container. This container is allocated a certain amount of resources depending on the package you purchased. The great thing about how these containers work is that they can easily and quickly be allocated more or less resources as you need them. For example, if you need to upgrade your ram quickly when you are expecting a high surge of visitors, you can simply add more ram to your container with the click of a button. If you were using dedicated hosting, someone would have to physically install the new ram into your server – which would result in downtime and lost traffic.

Cost effective solutions for small websites

VPS hosting solutions are much less expensive now than they were only a few years ago. With advances in virtualization technologies, the prices are only expecting to decrease. Because of this, VPS hosting is now an option for all size websites – even if you are just starting a brand new site. You can get a small private hosting environment for as little as $10 per month – this is just as cheap as many shared hosting accounts but without all the risk and performance issues associated with them.

## cPanel

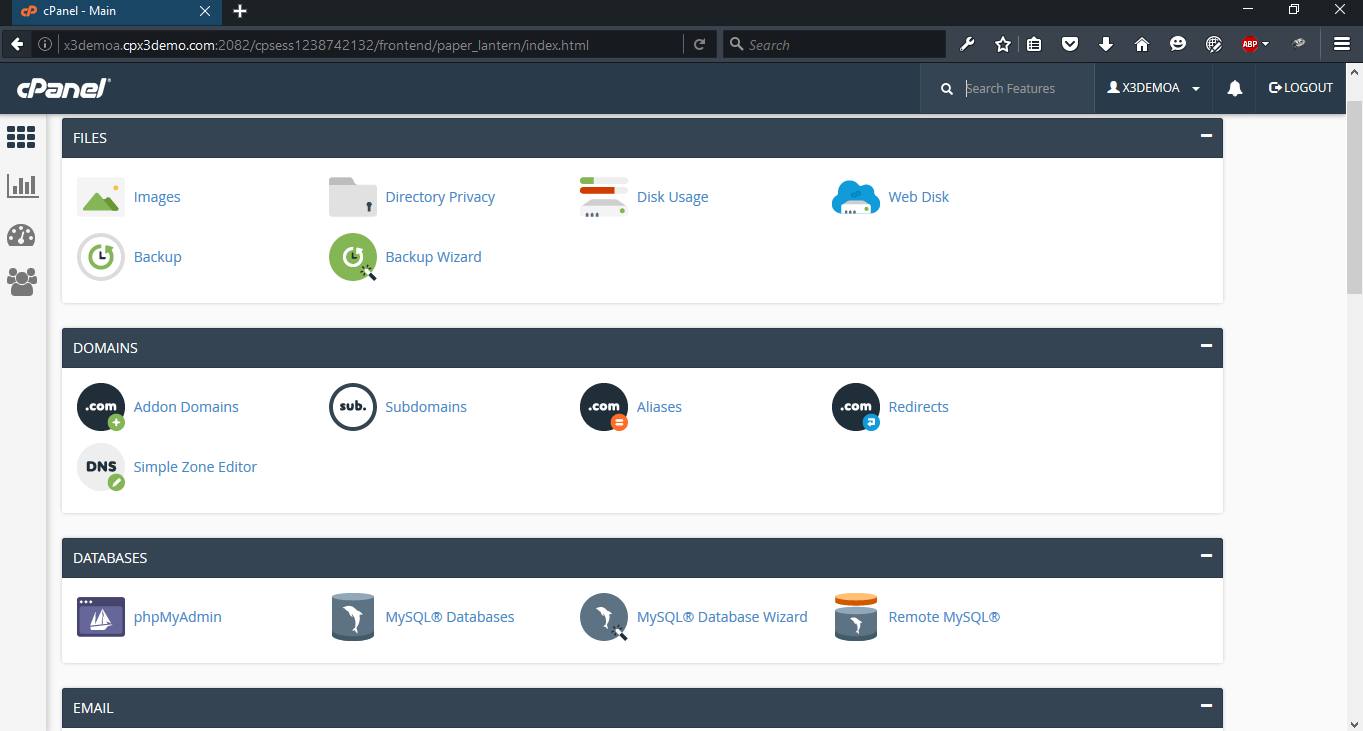
cPanel is a Linux-based web hosting control panel that provides a graphical interface and automation tools designed to simplify the process of hosting a web site. cPanel utilizes a 3 tier structure that provides capabilities for administrators, resellers, and end-user website owners to control the various aspects of website and server administration through a standard web browser.

In addition to the GUI, cPanel also has command line and API-based access that allows third party software vendors, web hosting organizations, and developers to automate standard system administration processes. [2]

cPanel is designed to function either as a dedicated server or virtual private server. The latest cPanel version supports installation on CentOS, Red Hat Enterprise Linux (RHEL), and Cloud Linux OS. [3] cPanel 11.30 is the last major version to support FreeBSD. [4][5]

Application-based support includes Apache, PHP, MySQL, PostgreSQL, Perl, and BIND (DNS). Email based support includes POP3, IMAP, and SMTP services. cPanel is accessed via https on port 2083.

Once installed, cPanel cannot be easily removed. cPanel's FAQ states that the best way to uninstall cPanel is by reformatting the server. [6] However, uninstall guides are available online for expert server administrators who do not wish to reformat their server. Similarly, it should only be installed on a freshly installed operating system with minimal prior configuration.



### **Features** of cPanel

The quantity of tools offered by cPanel web hosting represents a near unlimited number of possibilities in the way a user can manage their website or server.

The top features alone allow each user to fully manage their online presence professionally and with minimal training involved.

Security

Securing a website and server can be a hassle for even the most seasoned professional. cPanel helps its users by providing the tools needed to help reduce the possibilities of attacks all the while giving the website owner secure access to their server.

File Management

With multiple domains, subdomains, email accounts, CMS platforms, and users who access the server via FTP; A website owner can quickly become overwhelmed with trying to manage the server. cPanel makes this easy with their File Management tools

Domain Management

Customers running a server will for the most part need to create sub-domains, add domains to their hosting account, or park domains. cPanel makes these features available to our customers

Apps Galore

Take advantage of third-party software for blogs, bulletin boards, guest books, e-commerce, and more to build a robust, dynamic site.

Databases Management

Store large amounts of data and limit access using MySQL and PostgreSQL databases.

## 

# Specification & Design

## User Requirement

There is only one user who interact with application, administrator of web application. The users interact with this application through a terminal on server. This terminal presents a CLI for a user to interact with server for executing command.

## Software Dependencies

Server Setup application has terminal provided by default from Linux for user interaction. There are used very modern frameworks for developing its backend.

There are many third party python libraries and many other plugins used in this application for performing various tasks, the list is as follows:

* Python installed
* Cement Python
* Mustache
* Nose Testing
* SQLite
* Config Parse

## Performance Requirements

Since the application use based server client architecture, and use the large source information from distant server, it fetches data from internet. Internet bandwidth is major performance parameter. As System uses web server of user as source of massive data, multiple request and response is needed to handle in quick time for instant result back to user.

## Hardware Requirements

To use Setupserver it’s required a PC/Laptop with good internet connection.

Following are requirements from server side

* 2GB RAM (4GB recommended)
* 50GB free space in HDD

## Operating System Requirements

Server setup requires following operating system

* Ubuntu 12.04/14.04
* Debain 6/7

## Port Requirement

Following table contain required port for sending and receiving request for server.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Port** | **Inbound** | **Outbound** |
| SSH | 22 | YES | YES |
| HTTP | 80 | YES | YES |
| HTTPS/SSL | 443 | YES | YES |
| SS Admin | 2222 | YES | NO |
| GPG Key Server | 11371 | NO | YES |

## Activity Diagram

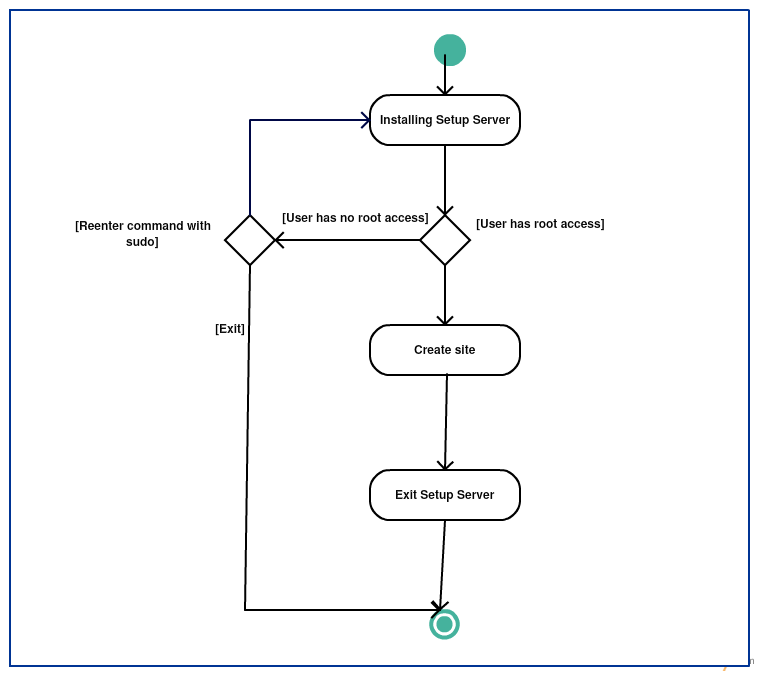
****

Figure 4 Activity diagram for Installation

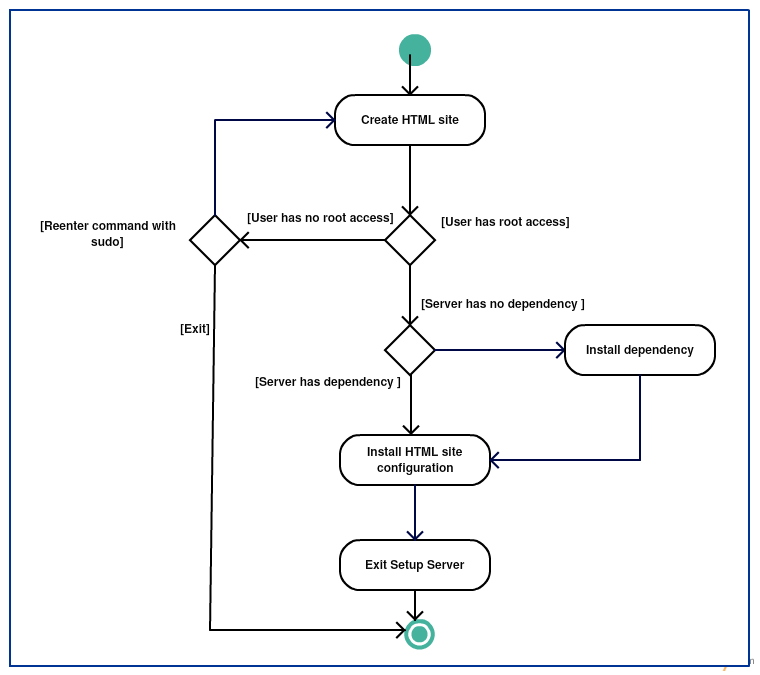


Figure 5 Activity diagram for creating site

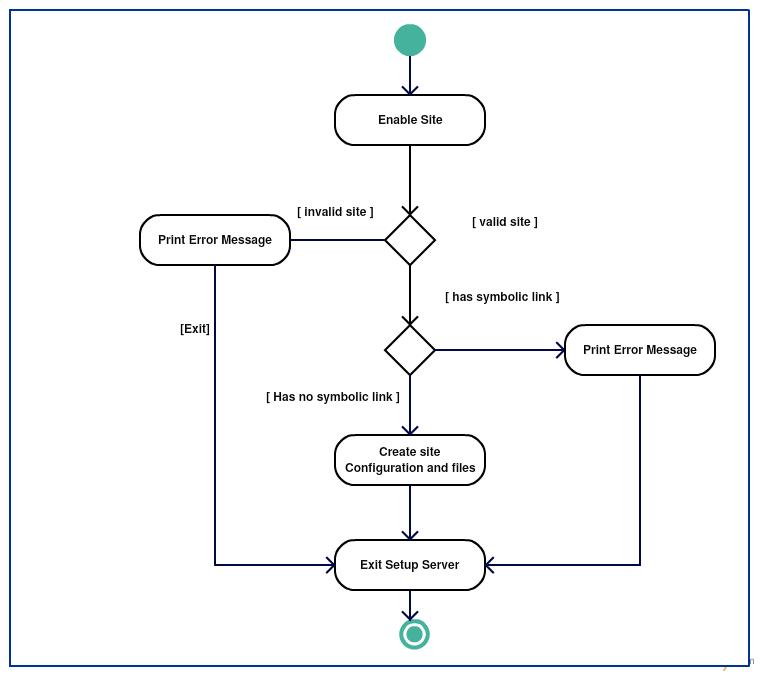


Figure 6 Activity diagram for enabling site

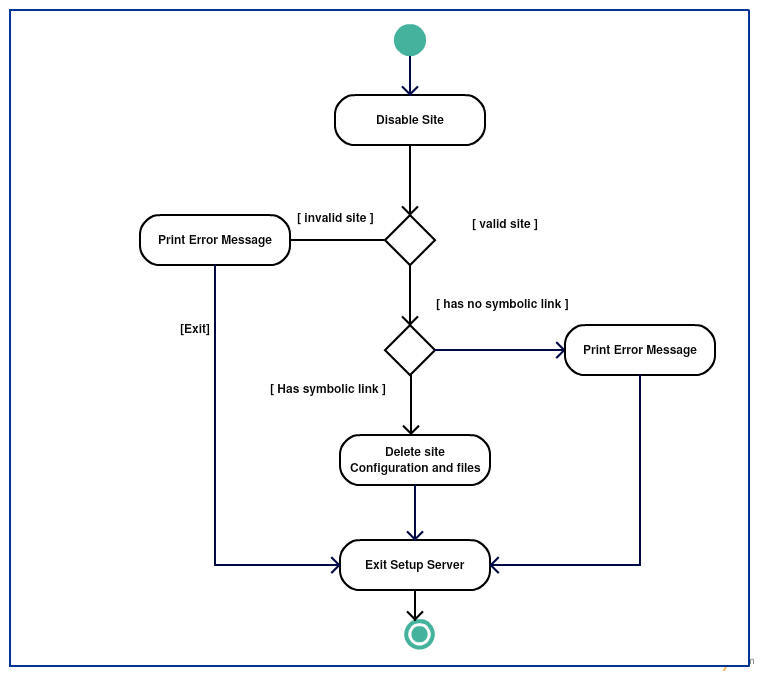
****

Figure 7 Activity diagram for disable site

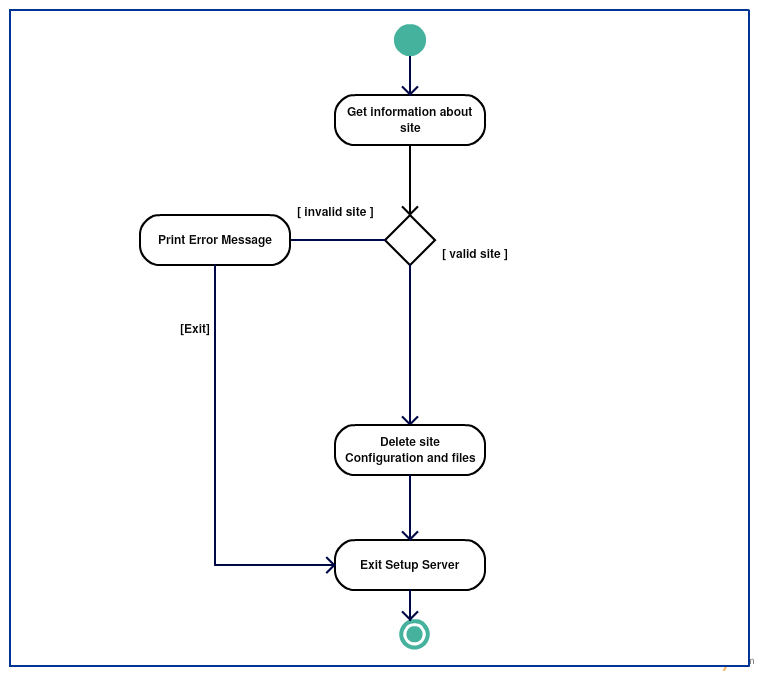
****

Figure 8 Activity diagram for getting information of site

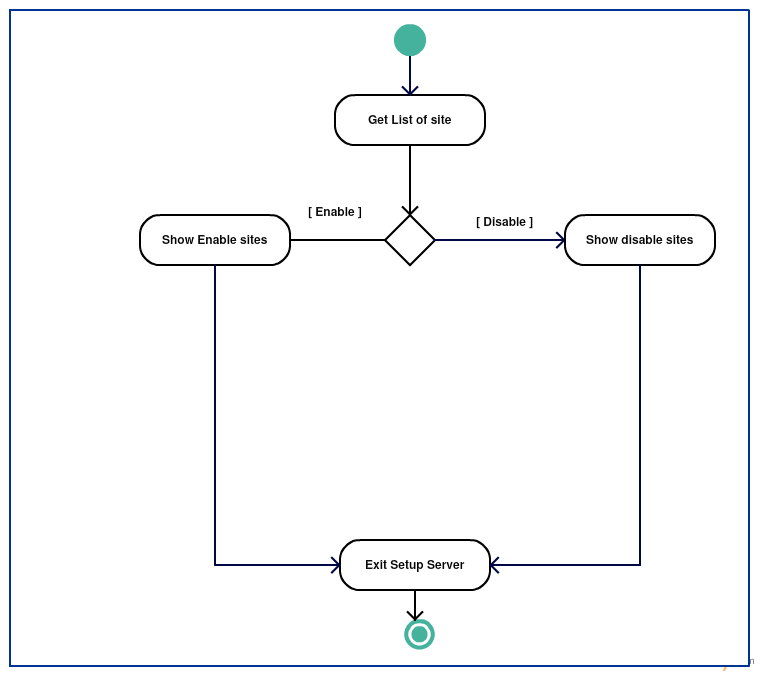
****

Figure 9 Activity diagram for getting list of sites

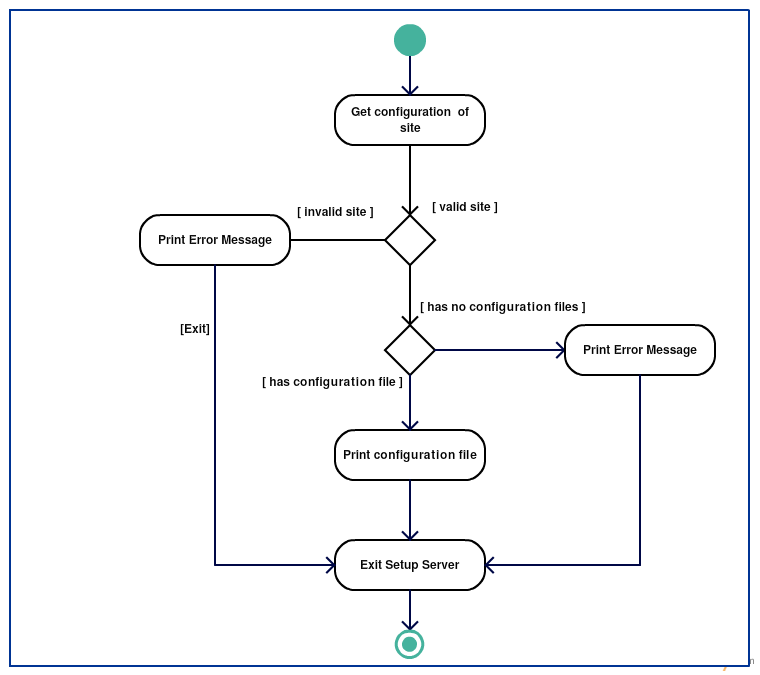
****

Figure 10 Activity diagram for showing configuration of site

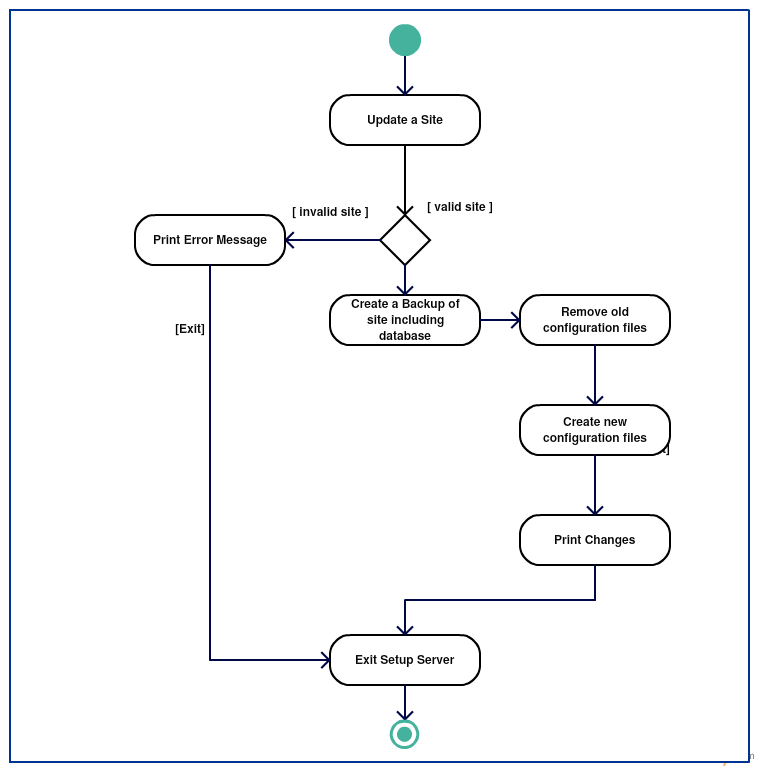
****

Figure 11 Activity diagram for updating site

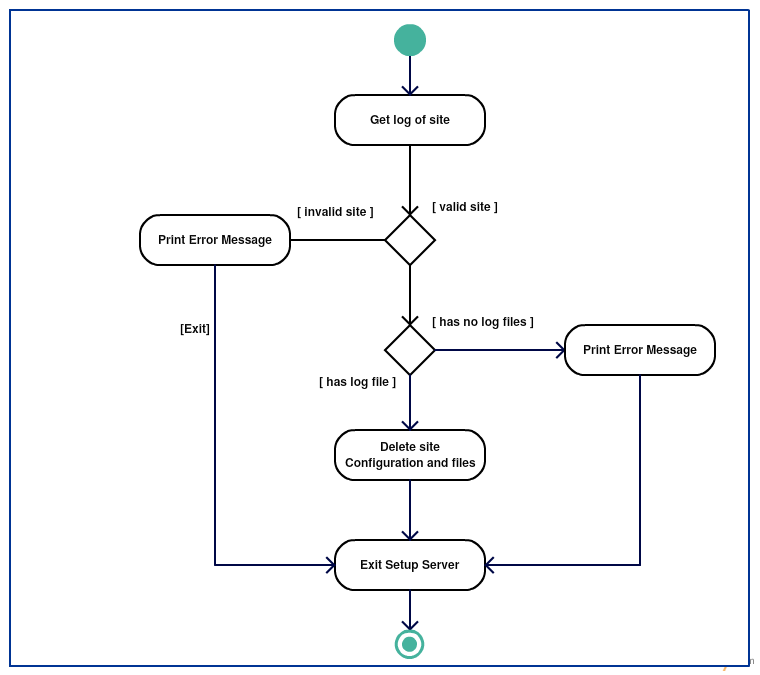
****

Figure 12 Activity diagram for showing log of site

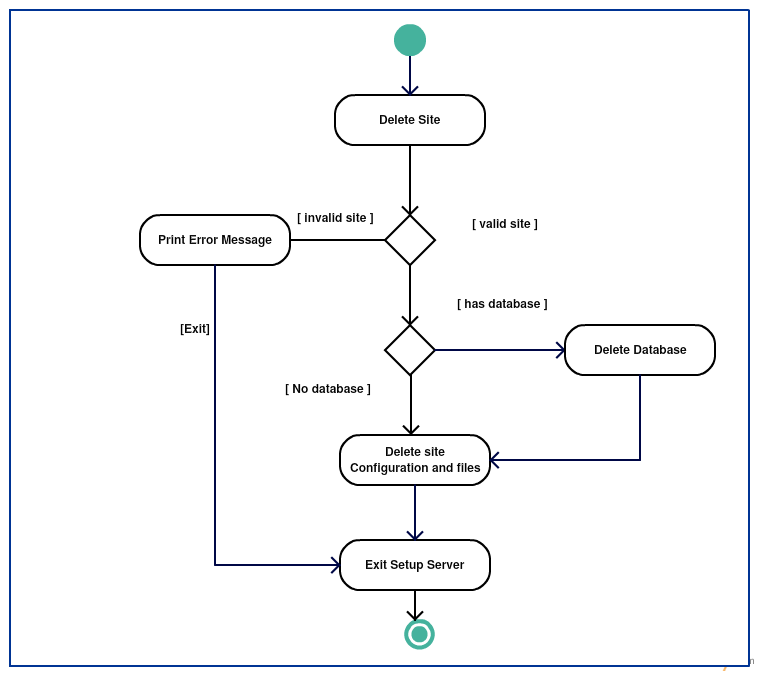
****

Figure 13 Activity diagram for deleting site

## Sequence diagram

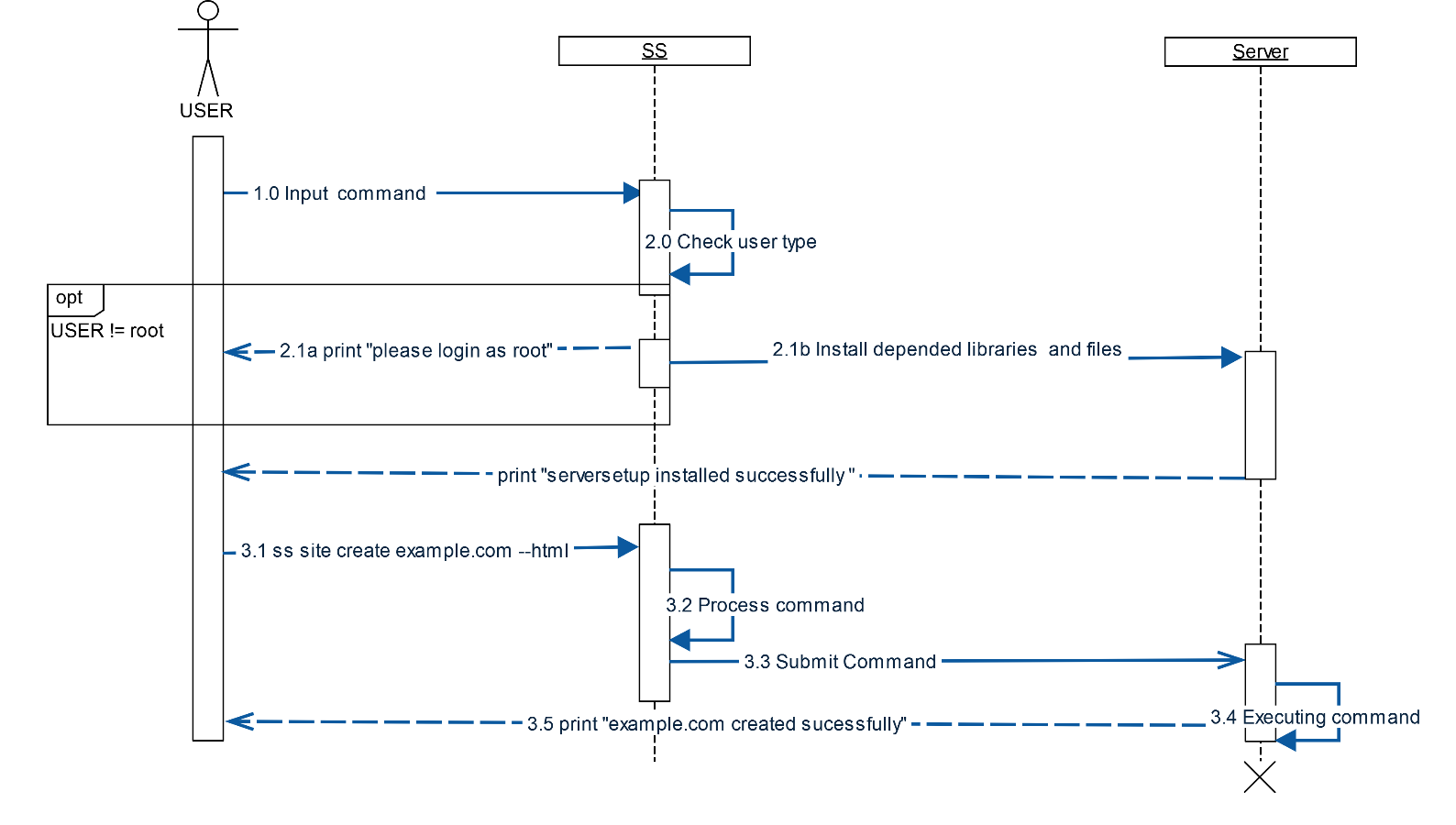
****

Figure 14 sequence diagram for installation

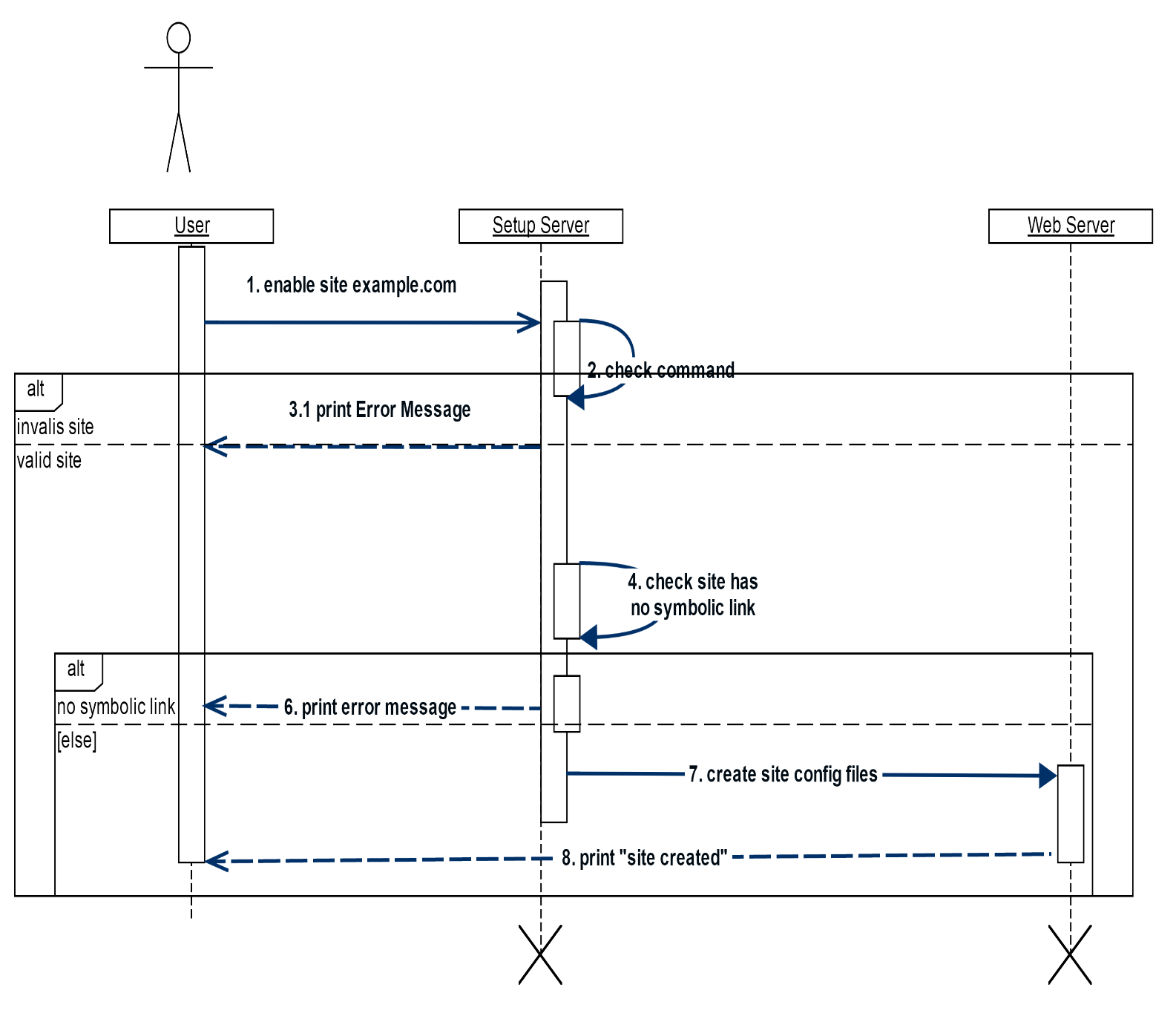


Figure 15 sequence diagram for enabling site

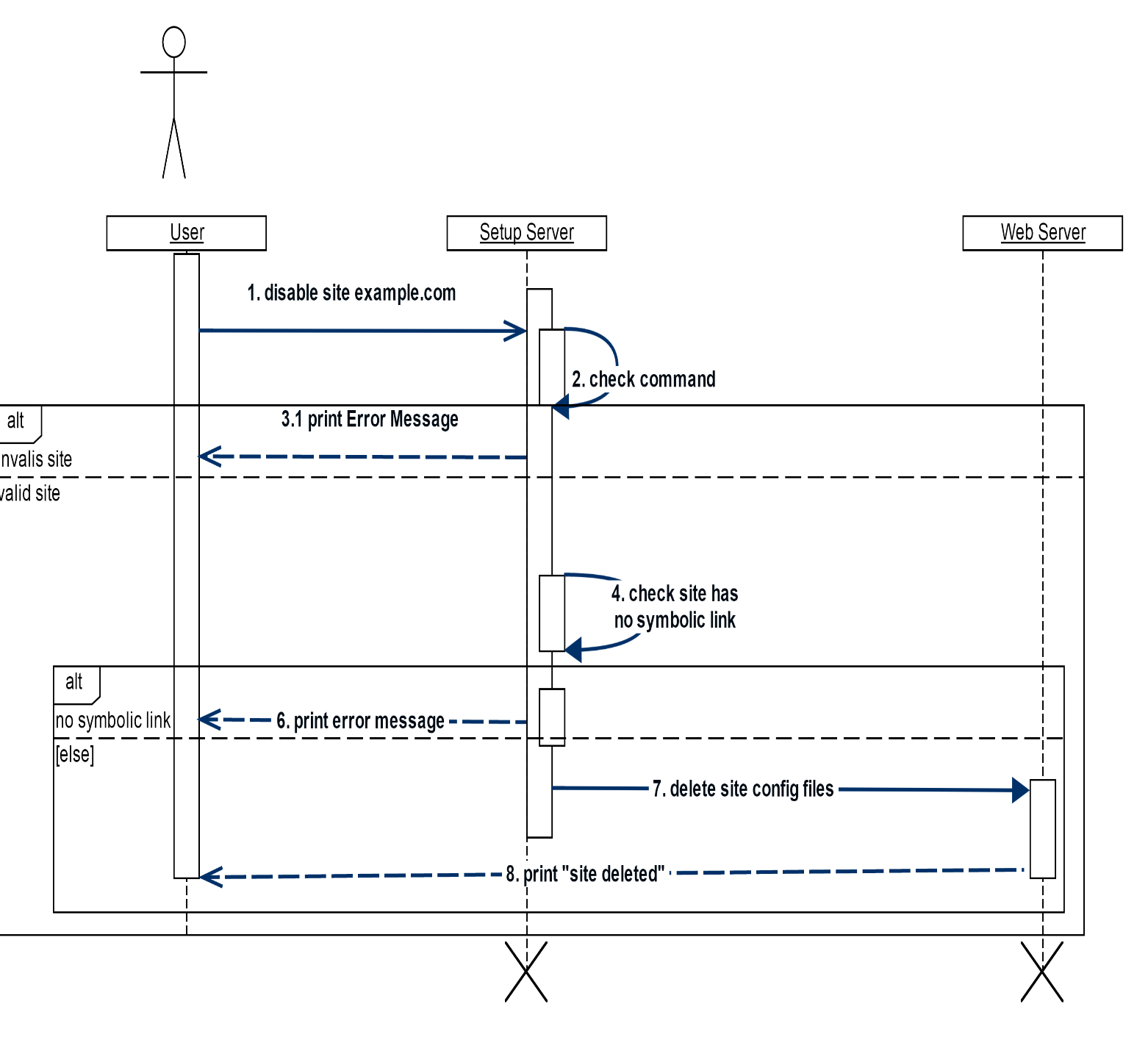
****

Figure 16 sequence diagram for disabling site

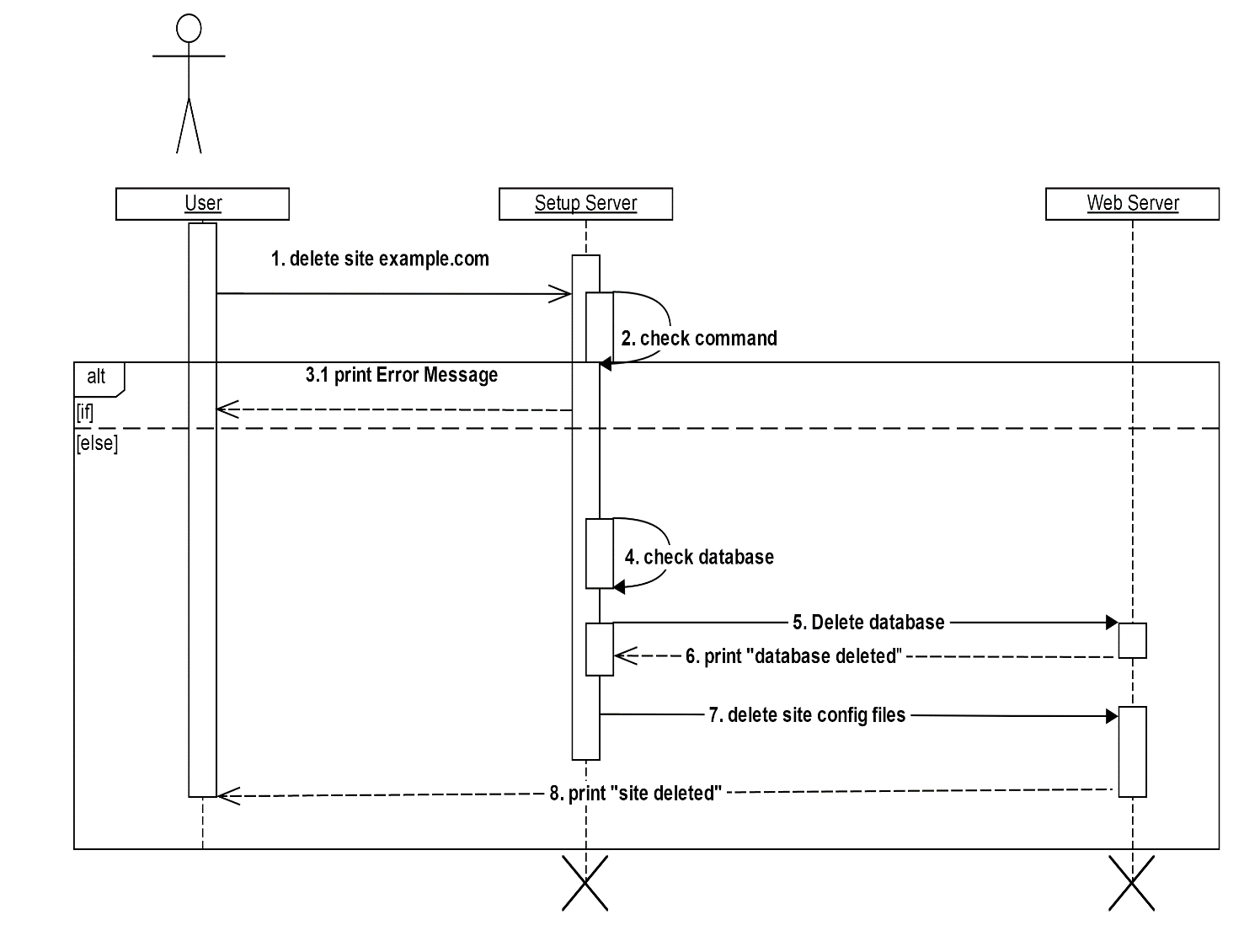
****

Figure 17 sequence diagram for deleting site

## Use case diagram

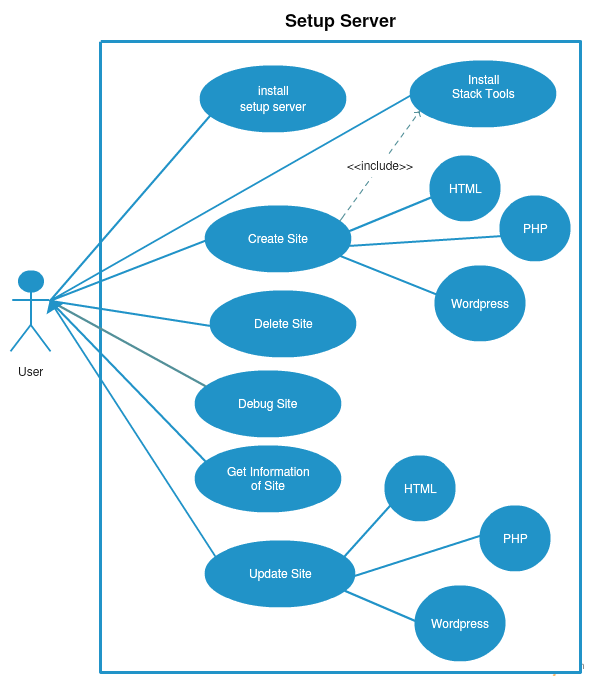
****

Figure 18 use case diagram for installation

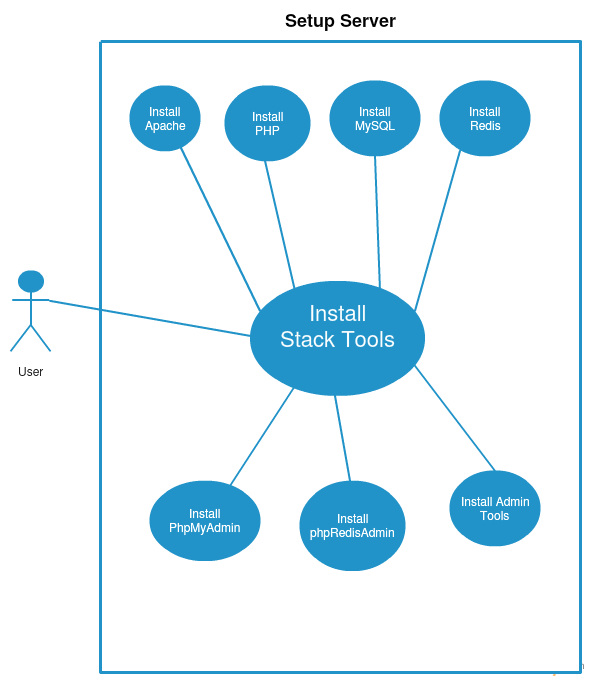
****

Figure 19 use case diagram for installing stack tools

## Data Dictionary diagram

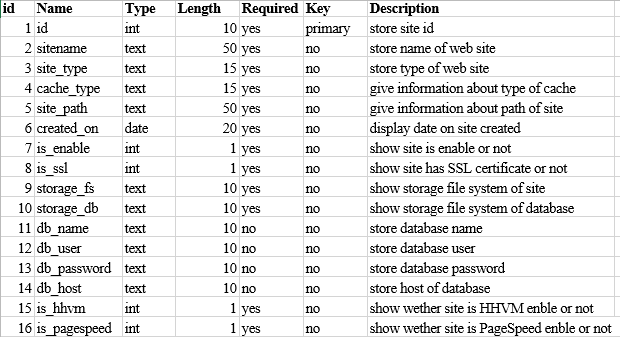


Figure 20 data dictionary diagram of used variable in server setup

# Implementation

# Result and Evaluation

# Future Scope

# Conclusions

# Reflection

# References

# Introduction

This section gives a scope description and overview of everything included in this Project Report. Also, the purpose for this document is described and system overview along with goal and vision are listed.

## Problem Statement

Following are some current problems at server administration level for web application like PHP, WORDPRESS, and HTML.

## Purpose

The purpose of this document is to give a detailed description of Project. It will illustrate the purpose and complete declaration for the development of system. It will also explain application constraints, interface and interactions with Linux server. This document is primarily intended to anyone who wants to get an overview of how Setup Server works, its outcomes and possible usage in future.

## System Overview

Server Setup takes input from user in command form and by processing that command generate appropriate output for web application to maintain server utility

## Objectives

## Goals and Vision

This application aims for providing CLI utility to user for maintain, govern and manage web server in easy manner.

# Background

This chapter introduces the reader to a description of the Linux server in section 2.1, Section 2.2 give general concept of python programming language, that is was used by this project. Then it informs the reader Cement -framework of python programing language for develop CLI application. Finally, when all the related concepts have been explained, in 2.4 explain similar CLI application for brief idea of project.

## Linux Operating System

Linux is, in simplest terms, an operating system. It is the software on a computer that enables applications and the computer operator to access the devices on the computer to perform desired functions. The operating system (OS) relays instructions from an application to, for instance, the computer's processor. The processor performs the instructed task, then sends the results back to the application via the operating system.

As an open operating system, Linux is developed collaboratively, meaning no one company is solely responsible for its development or ongoing support. Companies participating in the Linux economy share research and development costs with their partners and competitors. This spreading of development burden amongst individuals and companies has resulted in a large and efficient ecosystem and unheralded software innovation.

Over 1,000 developers, from at least 100 different companies, contribute to every kernel release. In the past two years alone, over 3,200 developers from 200 companies have contributed to the kernel--which is just one small piece of a Linux distribution.

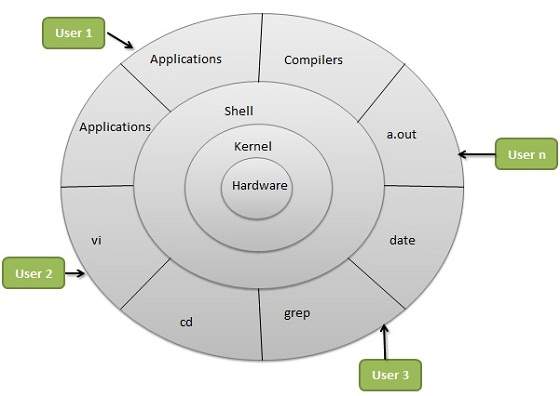
****

Figure 1Linux Architecture

## Components of Linux System

### Linux Operating System has primarily three components

* **Kernel** - Kernel is the core part of Linux. It is responsible for all major activities of this operating system. It is consists of various modules and it interacts directly with the underlying hardware. Kernel provides the required abstraction to hide low level hardware details to system or application programs.
* **System Library** - System libraries are special functions or programs using which application programs or system utilities accesses Kernel's features. These libraries implements most of the functionalities of the operating system and do not requires kernel module's code access rights.
* **System Utility** - System Utility programs are responsible to do specialized, individual level tasks.

### Features of Linux

Following are some of the important features of Linux Operating System.

* **Portable** - Portability means software’s can works on different types of hardware’s in same way. Linux kernel and application programs supports their installation on any kind of hardware platform.
* **Open Source** - Linux source code is freely available and it is community based development project. Multiple team’s works in collaboration to enhance the capability of Linux operating system and it is continuously evolving.
* **Multi-User** - Linux is a multiuser system means multiple users can access system resources like memory/ ram/ application programs at same time.
* **Multiprogramming** - Linux is a multiprogramming system means multiple applications can run at same time.
* **Hierarchical File System** - Linux provides a standard file structure in which system files/ user files are arranged.
* **Shell** - Linux provides a special interpreter program which can be used to execute commands of the operating system. It can be used to do various types of operations, call application programs etc.
* **Security** - Linux provides user security using authentication features like password protection/ controlled access to specific files/ encryption of data.

## Ubuntu

**Ubuntu** is a Debian-based Linux operating system and distribution, with Unity as its default desktop environment for personal computers including smartphones in later versions. Ubuntu also runs network servers. It is based on free software and named after the Southern African philosophy of Ubuntu which often is translated as "humanity towards others" or "the belief in a universal bond of sharing that connects all humanity".

Development of Ubuntu is led by UK-based Canonical Ltd. a company. Canonical generates revenue through the sale of technical support and other services related to Ubuntu.

Ubuntu has a server edition that uses the same APT repositories as the Ubuntu Desktop Edition. The differences between them are the absence of an X Window environment in a default installation of the server edition (although one can easily be installed including Unity, GNOME, KDE or XFCE) and the installation process. The server edition uses a screen mode character-based interface for the installation, instead of a graphical installation process. Ubuntu 10.04 Server Edition can also run on VMware ESX Server, Oracle's Virtual Box and VM, Citrix Systems XenServer hypervisors, Microsoft Hyper-V, QEMU, Kernel-based Virtual Machine, or any other IBM PC compatible emulator or virtualizer.

Ubuntu 10.04 turns on AppArmor (security module for the Linux kernel) by default on key software packages, and the firewall is extended to common services used by the operating system. Ubuntu 12.04 LTS Server Edition supports three major architectures: IA-32, x86-64 and ARM. Minimum RAM memory requirements are 128 MB. Ubuntu 14.04 LTS Server Edition includes MySQL 5.5, OpenJDK 7, Samba 4.1, PHP 5.5, Python 2.7.



Figure 2 Ubuntu 15.04 (Vivid Vervet) screenshot

****

Figure 3 Ubuntu 12.04 Server

## Python

CPython, the reference implementation of Python, is free and open-source software and has a community-based development model, as do nearly all of its alternative implementations. CPython is managed by the non-profit Python Software Foundation.

### Features of Python

* **Simple**
  + Python is a simple and minimalistic language. Reading a good Python program feels almost like reading English, although very strict English! This pseudo-code nature of Python is one of its greatest strengths. It allows you to concentrate on the solution to the problem rather than the language itself.
* **Easy to Learn**
  + As you will see, Python is extremely easy to get started with. Python has an extraordinarily simple syntax, as already mentioned.
* **Free and Open Source**
  + Python is an example of a FLOSS. In simple terms, you can freely distribute copies of this software, read its source code, make changes to it, use pieces of it in new free programs, and that you know you can do these things. FLOSS is based on the concept of a community which shares knowledge. This is one of the reasons why Python is so good - it has been created and is constantly improved by a community who just want to see a better Python.
* **High-level Language**
  + When you write programs in Python, you never need to bother about the low-level details such as managing the memory used by your program, etc.
* **Portable**
  + Due to its open-source nature, Python has been ported (i.e. changed to make it work on) to many platforms. All your Python programs can work on any of these platforms without requiring any changes at all if you are careful enough to avoid any system-dependent features.
  + You can use Python on Linux, Windows, FreeBSD, Macintosh, Solaris, OS/2, Amiga, AROS, AS/400, BeOS, OS/390, z/OS, Palm OS, QNX, VMS, Psion, Acorn RISC OS, VxWorks, PlayStation, Sharp Zaurus, Windows CE and even PocketPC
* **Interpreted**
  + This requires a bit of explanation.
  + A program written in a compiled language like C or C++ is converted from the source language i.e. C or C++ into a language that is spoken by your computer (binary code i.e. 0s and 1s) using a compiler with various flags and options. When you run the program, the linker/loader software copies the program from hard disk to memory and starts running it.
  + Python, on the other hand, does not need compilation to binary. You just run the program directly from the source code. Internally, Python converts the source code into an intermediate form called byte codes and then translates this into the native language of your computer and then runs it. All this, actually, makes using Python much easier since you don't have to worry about compiling the program, making sure that the proper libraries are linked and loaded, etc, etc. This also makes your Python programs much more portable, since you can just copy your Python program onto another computer and it just works!
* **Object Oriented**
  + Python supports procedure-oriented programming as well as object-oriented programming. In procedure-oriented languages, the program is built around procedures or functions which are nothing but reusable pieces of programs. In object-oriented languages, the program is built around objects which combine data and functionality. Python has a very powerful but simplistic way of doing OOP, especially when compared to big languages like C++ or Java.
* **Extensible**
  + If you need a critical piece of code to run very fast or want to have some piece of algorithm not to be open, you can code that part of your program in C or C++ and then use them from your Python program.
* **Embeddable**
  + You can embed Python within your C/C++ programs to give 'scripting' capabilities for your program's users.
* **Extensive Libraries**
  + The Python Standard Library is huge indeed. It can help you do various things involving regular expressions, documentation generation, unit testing, threading, databases, web browsers, CGI, ftp, email, XML, XML-RPC, HTML, WAV files, cryptography, GUI (graphical user interfaces), Tk, and other system-dependent stuff. Remember, all this is always available wherever Python is installed. This is called the 'Batteries Included' philosophy of Python.

## Cement Python

# Requirements Specification

Table 1 port requirement

## Constrain & Assumption

Server Setup return output basis on user input User have to input meaning command for getting appropriate output. It also assumes that the input is in proper standardized command and not some random characters. Any incorrect command in input which is not found in a standard command may result in error.

For now, Server Setup only support Ubuntu 12.04/14.04 and Debain base distribution. In future support version will provide to other distribution

# Design

# Installation and Instruction

Installing server setup on user machine require good internet connection. After installing server setup first user can run server setup on server machine with limit features.

User must have root access for executing server setup command.

Installation step for server setup

1. Grab a copy of installation shell file

*wget -qO ss bit.do/ssinstall*

1. Run ss file with root access.

*sudo bash ss*

This command will execute shell file name with root access .ss shell file will install every decencies and required CLI tools in server.

1. To create with mysql database run following command.

*ss site create example.com –mysql*

1. To create with wordpress run following command.

*ss site create example.com –wp*

# Future Work

Server setup will contain following feature in future version.

* **Easy debugging:**  Setup Server will have separate function or tools for debugging server related error.
* **Automated Monitoring with Email Alerts:** Setup server will provide email alerts for Server Error, Components status, backup alerts and failure report.
* **Mail server:** Server Setup will provide inbuilt SMTP/POP3/ IMAP mail server.
* **Shared Hosting Support:** Setup Server will support shared hosting environment.
* **Easy migration:** Setup Server will provide tools for migrating website from other control panel.
* **Easy Cloning:** Setup Server will provide support for easy cloning in other server.
* **Offsite /Peer Backups:** Setup Server will provide tools with GUI for taking offline backup.

# Summary

* Setup Server Application is easy to understand for first time users by going through the documentation.
  + User can install web component using single command.
  + User can take backup of whole site using single command.
  + User can change cache type of site using single command.
  + User can enable or disable HHVM or Page Speed for particular site using single command.

# References

1. Ubuntu Server Guide

https://help.ubuntu.com/lts/serverguide/

1. Python Documentation

https://www.python.org/doc/

1. Python Documentation for version 3.x

https://docs.python.org/3/

1. Cement Framework Documentation

http://builtoncement.com/2.6/

1. Apache Installation Guide

https://httpd.apache.org/docs/2.2/install.html

1. PHP Installation Guide

http://php.net/manual/en/install.php

1. MariaDB MySQL Installation Guide

https://mariadb.com/kb/en/mariadb/getting-installing-and-upgrading-mariadb/

1. Redis Installalation Guide

http://redis.io/documentation

1. HHVM Installation Guide

http://docs.hhvm.com/manual/en/install-intro.install.php

1. PageSpeed Installation Guide

https://developers.google.com/speed/pagespeed/module/download

1. Nose Testing Documentation

https://nose.readthedocs.org/en/latest/

1. Mustache Documentation

https://mustache.github.io/mustache.1.html