PROJECT PLAN

AwaseConfigurations

AwaseConfigurations -project

AwaseConfigurations

Version	1.0	Final
Creator	Henri, Panu, Armen	10.10.2011
Approved	Tero Karvinen	5.10.2011

VERSION HISTORY

Version	Date	Creators	Comments
0.1	4.10.2011	Henri, Armen, Panu	First draft
1.0	10.10.2011	Henri, Armen, Panu	Final

Table of Contents

1 PROJECT DEFINITION

1.1 Project Background

Project is started as a part of Haaga-Helias network administration course and is done to provide the team members with required know-how of building a system for controlling small networks.
Project aims to simulate the functions and requirements of a small software development company ran by a single network administration after the initial setup is done. The greatest challenge to the project is to increase the efficiency of the administration by automation and simplified scripting.

1.2 Project Task

Project will provide it's members with a clearer picture of the network operations, configurations and centralized controls. Understanding of how to build a small company network and what is required to operate it.

1.3 Project Goals and Outcomes

Building a 30 computer network using linux systems with automated administration tools using Fabric, Cobbler and Config-Package-Dev (from DebAthena). These 30 computers will be split to servers and workstations.
Finding out whether or not chosen tools will be enough to run the network or do we need some more specific tools and if these are automated and efficient enough to provide for easy administration.
Installations are to be run with Cobbler, Configurations to be done with Config-Package-Dev and other administrative tasks through fabric.
Finally test and run all the use cases (Appendix 3.) and report the results and learnings to public blog on internet.

1.4 Scope

Administrative tasks mostly focus on issues over configuration files. Several user accounts will be created to the system but we will not focus very much on privileges. Backup systems and requirements will not be focused on. System will be designed exclusively to run linux based systems.

1.5 Team and risk analysis

SWOT:
Strengths of the team lie in already existing high level of computer knowledge and enthusiasm. All members of the team are motivated and interested over the subject and are able to provide high quality testing results from the project that will have actual real life relations.
Weaknesses of the team are the human factors as other commitments in life might interfere with the project work.
Opportunities for the project would be to finish ahead of time and succeed in bringing out even more indepth project with more results and online material.
Threats perceived are listed as Appendix 1.

1.6 Project organization

Project members and roles:
- Henri - Project team member + Project manager
- Armen - Project team member
- Panu - Project team member + Secretary
- Tero - Steering group

1.7 Environment

The interest groups will be our blog readers as all our work will be presented through the blog posts. Groups viewing our work will be network administrators and linux system administrators looking for information about various software and system solutions. Private users can also find our tutorials helpful in learning how to start using the particular system administration tools and also help deciding over program choices should they require such information for their own interests.

1.7.1 Interest groups of the results

- The readers of our blog
- System administrators
- Network administrators
- Web administrators
- Programmers
- Possible future employers/job interviewers

1.7.2 Implementation environment

The construction environment
- Haaga-helia UAS lab network (roughly 30 workstations)

- Ubuntu Linux
- One control machine
- One test machine

Implementation Software

- Fabric
- Cobbler
- Config-Package-Dev
- Reprepro repository (for configuration packages)
- Git using GitHub for version management

Environment software

- Servers: file-server, web-server (apache) [support for php?], ssh-server, database server (mySQL), irssi, dhcp, dns, (mail?)
- Workstations: openssh-server (for fabric), gnome, LibreOffice, firefox, thunderbird

1.8 Project Budget

Estimated requirements are roughly 33 920 € for the entire project. See Appendix 4 for details.

1.8.1 Work Expenses

Work hours for 3 members will amount to a total of 810 hours. With average estimated wage of 15 euros per hour we would require 12 150 € for completing this project. No weekend extras to be calculated due to normal employment policy expectations.

1.8.2 Purchases and extra expenditures

We will need 30 computers, 30 monitors, 30 keyboards and 30 mice. Some network cables, chairs, tables, electricity, internet connection, an office or at at least some kind of room. For a list of all purchases and costs see Appendix 4.

1.9 Project Timetable

Project will be completed between the dates 12.9.2011 and 16.12.2011.
Weekly timetable available as Appendix 3.

2 WORK PLAN

2.1 Phases

Initially the work evolves from project initialization tasks, looking at our resources and

requirements on what would make a good project and working up to choosing our platform and tools. Planning the entire system and pregeneration of the required scripts and packages while making sure the installation procedures will run smoothly.
After we have sound basis we move over to implementing the system and testing it's properties and functionalities. These results will also work as the close up for the project and will be documented online to the projects wordpress blog.

2.2 Tasks, workloads and deliverables

Tasks and their respective workloads are described in Appendix 2 - Phases / tasks, workloads, and timing.
Deliverables will be the reports published on the project's blog. A final report on the whole project, which will sum the findings, will be published separately upon the project's closure.

2.3 Timing

The project started on 12.9.2011 and will end 16.12.2011. We have reserved 810 hours of worktime for this project. There will be weekly steering meetings with our steering group.
- Detailed schedule is described in Appendix 2.

2.4 Working methods and standards

On some of the documents and reports we will be using HAAGA-HELIA reporting standards. The progress of our work will be published on our blog. Work will happen at school or at home depending on the individual time allocation from each members. No set times are arranged apart from the weekly meetings with the project councelor.

2.5 Project management

Documents and reports will have to be approved by Tero on our weekly steering group meetings.

3 QUALITY PLAN

3.1 Quality Goals

All the deliverables should meet appropriate standards, with readability and usability put in front. The results need to be reviewed and approved.

3.2 Quality Procedures and Responsibilities

Tests at various phases of the project (planning, implementation, testing) will be used as quality assurance methods, with especial attention paid to the testing phase. All the results need to be documented, errors have to be pointed to, suggestions for fixes should be proposed, approved, and possibly implemented.

3.3 Documentation- and Version Management

Documentation must meet adequate level of standards of good written communication. The project team will use docs.google.com for collaboration on various managerial/project-plan-related documents as well as for backing up the documentation. git/github will be used for version management. Final documentation will be published on our blog under GPL/LGPL licence.

Appendix 1. Risks definition and analysis

Risk	**Probability**	**Effect**	**Risk value** (prop effect)	**Reasons**	**Prevention**	**Preparation for risk consequences**
Project is not completed on time	Small 2	Average 3	6	Inability to keep up with the schedule	Proper project management and sticking to the project plan.	Project progression is internally monitored to be on schedule.
The network machines cannot be installed with Cobbler	Small 2	Big 4	8	DHCP/PXE server is interfering with the lab network	Limiting the use of Cobbler to specific MAC-addresses	The network machines will be installed from live-CDs
Failure to create configuration packages	Small 2	Average 3	6	Packages prove to be too difficult to create	Sufficient studying and testing of config package building.	Configuring will be done by copying proper config files to all of the machines.
Failure to accomplish all the goals	Small 2	Average 3	5	There were too many goals or they were too difficult to achieve.	Accurate definition of goals and tasks needed to achieve them at the planning phase.	Some of the planned goals will be ignored.
Documents are lost or destroyed	Very small 1	Big 4	4	Corrupted hard drives, wordpress and google disappear.	Backups and version management	Backup files will save the day.
Too much stress on team members	Small 2	Average 3	6	Difficult life situation or time issues.	Everyone participates with the resources they can handle according to their life situation.	Some of the planned goals or tasks are postponed where necessary.
Small team size	Small 2	Average 3	6	Some of the team members can't do the tasks they are given.	Early agreement about who's doing which tasks. Notify members early if you can't get the tasks done.	Some of the planned tasks will be ignored
Inability to work on project due to illness	Average 3	Average 3	9	A team member gets ill.	The ill one shall notify other members and they will do his tasks if it's possible.	The project will be delayed or some of the planned tasks will be ignored.
Not enough time to work on project	Small 2	Average 3	6	A team member can't fit the required project hours on his schedule.	A team member with timing issues will have to learn the art of multitasking and sleepworking	Some of the planned tasks will be ignored.

Appendix 2. Phases / tasks, workloads, and timing

Week /2011

	Phase/Task	**h**	**37**	**38**	**39**	**40**	**41**	**42**	**43**	**44**	**45**	**46**	**47**	**48**	**49**	**50**
		810	9	43	69	83	78	30	58	78	75	75	75	45	60	32
1	**Starting the project**	**25**														
1.1	Preparation for starting meeting	**5**	5													
1.2	Starting meeting	**3**	3													
1.3	Writing memo	**1**	1													
1.4	Starting the blog	**6**		6												
2	**Tools research and analysis**	**61**														
2.1	Tool research	**25**		15	10											
2.2	Tool analysis	**6**			6											
2.3	Testing the tools	**30**		20	10	10										
3	**Defining construction environment**	**66**														
3.1	Defining use cases	**40**			20	20										
3.2	Reporting and publishing	**20**			10	10										
3.3	Review	**6**			3	3										
4	**Planning**	**199**														
4.1	Planning the tests	**30**			5	5	10	10								
4.2	Planning the test environment	**20**				10	10									
4.3	Planning the test taks	**30**				10	20									
4.4	Configuration planning	**60**				10	20	10	20							
4.5	Usage planning	**50**					10	10	10	20						
4.6	Review	**9**					3		3	3						
5	**Implementation**	**260**														
51	Implementing the test environment	**50**							10	10	10	10	10			
5.2	Implementing use cases	**50**							10	10	10	10	10			
5.3	Configuring	**80**								20	20	20	20			
5.4	Testing	**80**								10	10	10	10	20	20	
6	**Testing the system**	**102**														

6.1	Testing	**50**									10	10	10	10	10	
6.2	Reporting and publishing	**30**									6	6	6	6	6	
6.3	Fixing	**22**									4	4	4	4	6	
7	**Project management**	**50**														
7.1	Work status reporting	**30**			3	3	3		3	3	3	3	3	3	3	
7.2	Steering group meetings	**10**		1	1	1	1		1	1	1	1	1	1		
7.3	Writing meeting memos	**10**		1	1	1	1		1	1	1	1	1	1		
8	**Closing the project**	**47**														
8.1	Creating the project closing report	**33**													15	18
8.2	Preparation of the closing meeting	**10**														10
8.3	Closing meeting	**3**														3
8.4	Writing a meeting memo	**1**														1

Appendix 3. Use Cases

CHARACTERS

ADMIN
name: Björn
expertise: Unix based systems (5 years), python programming (2 years),
interests: Linux

EMPLOYEE
name: Simo
expertise: Network programming (2 years), python programming (1 year).
interests: Beer

1 ADMIN USE CASES

1.1. Installing Gnome

Björn wants to be able to clean install gnome on all the workstation computers of the network as fast and efficiently as possible. This is required at initial setup and should be well documented in order to restart the entire system if necessary.

Possible solutions:
- Use fabric to run: sudo apt-get install gnome fabfile role: workstation
- Instead of running apt with fabric, send a script with fabric that will run the apt.

1.2. Editing source lists

Björn wants to modify repository source list of every workstation to ensure that all installations and updates will be possible without problems with the sources.

Possible solution:
Make a configuration package that modifies the source list.
Use fabric to send .deb configuration package to each machine.
Use fabric to install configuration packages: gdebi package.deb
fabfile role: workstation

1.3. Adding users

Björn wants to add user accounts to workstation computers in order to enable unique work environment for each employee.

1.4. SSH public key authentication

Björn wants to be able to automate his ssh logins in order to improve security, work speed and efficiency while he uses fabric.

1.5. Update interruption

Björn wants to be sure that the software and security updates are applied at earliest convenient times even if the computer was not accessible at the time of the original script run.

1.6 Enabling apache user dir

Björn wants to enable user directories on apache server so that employees can make their own web pages.This should be automated so that Björn only needs to run a simple fabric task:
fab -H webserver config(apache_userdir)

1.7 Enabling php for users

Björn wants to enable php for users so employees can use php on their web pages. This should be automated so that Björn only needs to run a simple fabric task:
fab -H webserver config(php_enable)

1.8 System status

Björn wants to check the system status of workstations and servers so that he can do configurations more efficiently. This should be automated so that Björn only needs to run a simple fabric task: fab -R workstations,servers status

1.9 File transfer to file-server

Björn wants to add files to the file-server so that employees can get them if needed. This should be automated so that Björn only needs to run a simple fabric task: fab -H fileserver putfile(file)

1.10 File transfer from servers

Björn wants to download a log file from each server in order to improve security. This should be automated so that Björn only needs to run a simple fabric task: fab -R servers getfile(log)

1.11 Shutting down workstations

Björn wants to shut down all of the workstations in order to decrease energy consumption. This should be automated so that Björn only needs to run a simple fabric task: fab -R workstations shut_down

1.12 Rebooting servers

Björn wants to reboot all of the servers in order to improve the server performance. This should be automated so that Björn only needs to run a simple fabric task: fab -R servers reboot

1.13 Upgrading workstations to a newer version of Ubuntu

A newer version of Ubuntu came out (not LTS). Björn wants to upgrade all workstations to the newer.

1.14 Upgrading servers and workstations to a newer version of Ubuntu

A newer Ubuntu LTS version has been released. Björn wants to upgrade all the servers and workstations

2 EMPLOYEE USE CASES

2.1. Updating system

As an employee I want to be sure to have my programs are up to date in order to maximize my productivity. This should be automated and not require me to do anything at any point.

2.2 File transfer to file-server

Simo wants to transfer his work files to the file-server in order to keep a backup copy of them. This should be automated so that Simo only needs to copy files from his work folder to a “Backup” folder on his desktop.

Appendix 4. Budget

Product	**Amount**	**Price**	**Total**
Computer	30	400,00 €	12 000,00 €
Monitor	30	90,00 €	2 700,00 €
Keyboard	30	15,00 €	450,00 €
Mice	30	15,00 €	450,00 €
Network cable	32	10,00 €	320,00 €
Chair	30	40,00 €	1 200,00 €
Table	15	70,00 €	1 050,00 €
Total			18 170,00 €
	Months	**Cost/month**	
Electricity	4	60,00 €	240,00 €
Internet	4	40,00 €	160,00 €
A room	4	800,00 €	3 200,00 €
Total			3 600,00 €
	Hours	**€/hour**	
Work expenses	810	15	12 150,00 €
Total			**33 920,00 €**

Appendix 5. Milestones

Project started - Week 37
Blog started - Week 38
Tools decided - Week 39
Git repo started - Week 40
Cobbler tested at lab and got working - Week 40
Project Plan finished - Week 41
All config packages created - Week 44
Complete fabfile(s) created - Week 45
Reprepro/github set up with the config packages - Week 45
The initial system tests done at the lab - Week - 47
All tests finished - Week 49
Closing report finished - Week 50
Project ended - Week 50