

Name: Jason Gunn

**Course: Computer Games
Production**

**Unit: Computer Games Production –
Group Project**

Unit Code: GMC2002

Assignment Number: 2

Unit Lecturer: Phill Richardson

Contents:

Page 1 – Cover Sheet

Page 2 – Contents

Page 3 - Introduction

Page 4 – Introduction Continued (Lifecycles)

Page 5 – Tool Support

Page 6 – Tool Support Continued (Disadvantages)

Page 7 – Conclusion And Bibliography

Page 8 - Bibliography

Group Project – Games Production.

Introduction:

Group Project Properties:

It is important for the group to work as a team. Learning each others strengths and weaknesses, this allows the group to support each other, and aids project completion.

The group will need to be able to work to deadlines, in order to ensure that projects are finished on time.

Members will also need to be able to produce individual work, which can be combined to produce the final project artefact.

Disadvantages Of Group Work:

There are disadvantages to working as part of a group:

Reliability is a large part of this. Members need to be punctual, and attend all meetings otherwise progress will be delayed.

The group also needs a certain amount of enthusiasm towards the project(s) they are undertaking, and have the ability to work together well as a team.

Advantages Of Group Work:

There are also advantages to group work:

A project will be divided between group members, meaning that individual work is potential reduced, this allows for projects to be completed faster, making deadlines easier to meet.

It also allows the group to undertake larger projects, that an individual would be unable to complete, meaning that better results can be achieved.

Tradeoffs And Compromises With Group Work

Because larger projects can be undertaken, the group can benefit as a whole, however this requires more effort from the group members to complete the project, they need to put in time, and commitment to ensure its success.

Problems will occur with group members, health issues, injuries and other personal problems may cause members to take time off, or leave the project completely. This means that the rest of the group may have to undertake a slightly larger work load, part way into the project.

Lifecycles:

Groups tend to go through several states during the course of their development, these include “Forming” “Storming” “Norming” and “Performing”. A group may not make it to the performing stage, as problems that occur before this can cause the group/project to fail. Groups also tend to go through these stages several times during their development, depending on events which occur, such as change of group members.

Forming:

This is the initial stage in which a group is forming. Group members are just starting to learn about each other, learning each others strengths, and weaknesses.

It is in this stage that the bond between group members begins to form, it is key that good leadership is provided, in order to guide the group in the correct direction.

Group members are just starting to learn about the project they are going to undertake, finding out what will be expected of them etc.

Groups tend to conform at this stage to ideas, and make effort to work with each other well, however lack the ability to work individually for the group.

Storming:

This is the second stage a group can go through, it is as the name suggests a time when problems are likely to occur within a group.

There will be disagreements between members, especially over the direction in which the project is heading. Most members will have different ideas, which will need to be discussed until a solution is found which satisfies the group as a whole.

Good leadership is needed in this stage also, to keep the group on the right track. This is when the group is most likely to fail, should problems not be resolved.

Norming:

During this stage the groups problems have mainly been resolved, the project can now be concentrated on. Leadership is still important, however the group members are now ready to take on more responsibilities individually.

The group therefore is beginning to work together as a team, members are learning each others pros, and cons. They are also more knowledgably about the project.

Group members tend to have a higher motivation during this stage, usually due to agreements between members on working standards, etc.

Performing:

At this stage the group is most able to complete the project, they know how the team performs, and are able to adapt the project around this. They also need the least guidance during this stage, and are able to complete work individually, for the group.

The group however needs to ensure that it does not become complacent during this stage, otherwise problems will occur, such as lack of motivation.

Tool Support:**Style Of Tool:**

The tool I have selected to talk about is SubVersion <http://subversion.tigris.org/>

This is a tool which allows easy management of files. These files can be group related documents, or project files such as graphics, code, etc. It allows members to easily backup files, by creating local copies on their computer. As well as having a current, and up to date version of the files online.

It also implements a version management system, which allows updated versions of files to be uploaded to the source, while older versions are kept, in case the group needs to revert to them.

The main program is hosted on a Linux server which supports Apache, and is an Open Source software, meaning that there are several distributions of it, all containing different functionality. It will also allow a group to modify the program to suit their needs, while keeping costs to a minimum.

Key Properties/Advantages:

- ❖ It is open source, no cost to use, and allows modification to suit projects.
- ❖ Allows incremental updates of files.
- ❖ Keeps old files in case the group needs to revert to them.
- ❖ Can be accessed by the whole group via the internet, and content can then be stored locally, until the next time it is updated.
- ❖ Runs on Linux servers using Apache.
- ❖ Can be applied to many different aspects of projects, documentation, final work etc.

The following quote is taken from a case study on the use of SubVersion, proving that it is indeed possible to recover an older version, if necessary:

(2007) **Escapegoat.org** *“That is, every time somebody changes the code, subversion makes a note of the change, and it’s always possible to roll the version backwards if the change turned out to be a big mistake.” Available from:* <http://escapegoat.org/2007/9/18/a-subversion-case-study> [Date Accessed: 1st November 2007]

Disadvantages:

SubVersion, is difficult to set up(install) which can lead to the inability to use it.

It also only supports Linux, so this may cause problems if the group does not have the correct hosting available to support it.

Being open source modifications to this tend to be complicated, and unsupported to some degree, meaning that modifications will need to be undertaken by a competent coder.

How To Use:

This program can be modified, to suit most projects.

Once the program is installed, then several aspects of the project can be managed at once, the group members then need to upload any necessary files to the main server.

The members can then download the latest versions to their local work stations, and modify them as required. Once modified they will then upload them to the server, and they will be saved as an incremented version.

Due to the file incrementing system, older files can easily be restored.

Example Of The Tool's Use In A Real Project:

SubVersion is used in many group projects, including Half Life 2 (Source Engine) modifications, for example, it is used in the “Lego” mod Block Party:

<http://blockpartymod.com/team.html>

Steve Kuhn (2005) **Block Party** “set up a Subversion source control server for the mod” Available from: <http://blockpartymod.com/team.html> [Date Accessed: 1st November 2007]

It is also used in the modification Insurgency: <http://www.insmod.net> which was worked on by a student at Lincoln University.

In the above cases it is used to manage planning, material, and project files:

- ❖ Assets – Sound, 3D Models, etc.
- ❖ Coding.
- ❖ Level design documentation
- ❖ Storyboarding
- ❖ Etc

Potential Future Developments:

Because this software is open source, many different iterations of it will appear in the future, providing different functions, which are suited to different projects.

It supports all file types, thus can be used specifically for any project, graphics design, recipe website etc.

Because of the functionality of this software, it is especially useful for group work, it allows all group members to store, access, and update files quickly and easily, meaning that the project can fluidly continue to be produced, even if all group members cannot be in the same location at once.

It allows the group to access important documents, such as meeting agendas, new ideas etc, so that they can be constantly up to date.

Conclusion:

Summary Of Group Working:

In group work, reliability is essential to the success of the project, members need to attend meetings, and complete work for set deadlines.

Working in a group allows for a larger project to be undertaken, and the ability to complete smaller tasks much more efficiently.

As a group works together they evolve, they understand how each other works, and how to use this to benefit the group.

Summary Of Chosen Tool:

SubVersion is a constantly evolving, open source program, it is free to use, and allows modification to suit different projects. It manages different types of files, and allows for different versions to be stored on a server in-case of problems, it will develop greatly in the future, offering more functionality to users, and will suit many different types of group projects.

Bibliography:

Quotes:

(2007) **Escapegoat.org** *“That is, every time somebody changes the code, subversion makes a note of the change, and it’s always possible to roll the version backwards if the change turned out to be a big mistake.” Available from:*

<http://escapegoat.org/2007/9/18/a-subversion-case-study> [Date Accessed: 1st November 2007]

Steve Kuhn (2005) **Block Party** *“set up a Subversion source control server for the mod” Available from: <http://blockpartymod.com/team.html> [Date Accessed: 1st November 2007]*

Websites Used:

<http://escapegoat.org/2007/9/18/a-subversion-case-study>

<http://blockpartymod.com/team.html>

<http://en.wikipedia.org/wiki/Forming-storming-norming-performing>

Other Sources Used:

Lincoln University Blackboard – Resources (Computer Games Production – Group Project – Lecture Notes – Phill Richardson)