

|  |  |  |
| --- | --- | --- |
| Date: 25/03/2015 | Version: 3.0 | Supervisor: Anne Philpott |
| Clients: |  |  |



**Team details**

**Current Team Members**

Charlotte Paterson      *Software Development & Computer Science*

pdy2956@aut.ac.nz

+64 21 154 7735

Gaoxin Huang           *Software Development*

gfx0590@aut.ac.nz

+64 21 0231 7756

Yue Li                       *Software Development*

pbq6275@aut.ac.nz

+64 21 0817 9058

**New Team Members**

Siatua Uili                                   *Software Development*

wbv5134@aut.ac.nz

+64 21 156 9048

William Stokes   *Software Development*

zhf3041@aut.ac.nz

+64 22 391 9890

**Supervisor**

Anne Philpott              *Senior Lecturer*

anne.philpott@aut.ac.nz                                             School of Computing and Mathematical Sciences

(09) 921-9999 ext 5216

**Clients**

Helen Varley Jamieson      *Artist and Original UpStage Project Coordinator/Founder*

Germany/Europe

helen@creative\_catalyst.com

Vicki Smith                      *Artist, Educator and Original UpStage Founder*

Wellington

vicki@UpStage.org.nz

**Table of Contents**

Contents

[**Team details** 1](#_Toc414986336)

[**Current Team Members** 1](#_Toc414986337)

[**New Team Members** 2](#_Toc414986338)

[**Supervisor** 2](#_Toc414986339)

[**Clients** 2](#_Toc414986340)

[**Table of Contents** 2](#_Toc414986341)

[**Terms of reference** 4](#_Toc414986342)

[**Rationale** 4](#_Toc414986343)

[**Scope and objectives** 4](#_Toc414986344)

[**Project Approach** 5](#_Toc414986345)

[**SCRUM** 5](#_Toc414986346)

[**Pair Programing** 5](#_Toc414986347)

[**Design Science** 6](#_Toc414986348)

[**Project Plan** 6](#_Toc414986349)

[**Communication Plan** 7](#_Toc414986350)

[**Skills and knowledge involved** 7](#_Toc414986351)

[**Personal Skills** 7](#_Toc414986352)

[ **Self-Management** 7](#_Toc414986353)

[ **Time Management** 8](#_Toc414986354)

[ **Problem Solving** 8](#_Toc414986355)

[ **Communication** 8](#_Toc414986356)

[ **Teamwork** 8](#_Toc414986357)

[ **Attitude** 8](#_Toc414986358)

[**Professional Skills** 8](#_Toc414986359)

[ **Ethics** 9](#_Toc414986360)

[ **Project Management** 9](#_Toc414986361)

[ **Project/Progress Tracking** 9](#_Toc414986362)

[ **Risk Management** 9](#_Toc414986363)

[**Technical Skills** 9](#_Toc414986364)

[**Estimation of all costs incurred** 10](#_Toc414986365)

[**Appendix** 10](#_Toc414986366)

[**A: UpStage Functionality as in semester 2 2014 proposal** 10](#_Toc414986367)

[**B: Graphic of planning for semester 1 by Yurek Li** 13](#_Toc414986368)

[**B: Graphic of planning for semester 1 & 2 by William Stokes** 15](#_Toc414986369)



**Terms of reference**

Upstage is an open source multi media, interactive cyber performance system. The variety of features allow for upstage to be used to put on a show or performance of some kind. it could easily function as a multipurpose tool for entertainment, education, communication and possibly many other things. It is intended to be easy to access.

Upstage features a stage, a chat box and a palette of tools for interacting with the stage. Avatars, props and backgrounds can be put onto the stage, a draw function can be used to draw on the stage in multiple colours, different sized brushes and different levels of transparency

The developing team at AUT has been through many iterations over the last ten years working for clients Helen V. Jamieson and Vicki Smith.

**Rationale**

UpStage exists in its third version which has many features but at the same time many issues that need to be resolved, this piece of software has suffered from a number of factors. The code is very hard to read and comprehend, with poor variables and class naming, for instance one class is called ‘things’, and it is lacking in any form of coding standards now.

Our clients wants to have upstage continue with more of a focus on mobile platforms, but for this to happen there will have to be a large number of changes to the existing code, for instance it is very difficult to get flash to work on a mobile device and flash is a component of UpStage.

UpStage already has a quite well established user base, it would be remiss to let the current UpStage system go unmaintained and lose many of these users. So we intend to perform regular maintenance on the existing Upstage product until it can be replaced.

Perhaps the most important thing about the UpStage project is that it has been an AUT R&D project Since 2006 and it has been in the works since 2002, AUT has committed its time and resources to this project already and stopping now when UpStage needs to be brought up to date would reflect poorly on the university.

**Scope and objectives**

The **long term** goal of the UpStage project is to produce a new version with all the same functionality and perhaps some added features using more up to date technologies to achieve this functionality. The product will need to be made in a way that reduces coupling and allows for future modifications to be easily implemented.

Before this can be achieved, we need to identify technologies that can replace the current technologies used in UpStage. The current main technologies include flash, a piece of text-to-voice software and a video streaming technology.

However these goals will be meaningless if the current user base of upstage were to leave. Upstage in its current state is facing many technical issues that are making it hard for users to enjoy the product to its full potential, fixing these problems and maintaining the current version of Upstage are necessary to prevent user migration.

**Project Approach**

Since UpStage is an ongoing project, the senior members of the team have researched and decided on some methodologies and practices to use already. Aspects of the following approaches, SCRUM and Design Science, are already in use by the project team with the latter being the approach we will be using more this semester.

We will also be deriving or taking lessons or ideas from some practices such as pair-programing and scrum.

**SCRUM**

Scrum is an agile framework for completing projects, and is stated by many to offer great benefits.(James, M. (n.d.)) We will not be using scrum but we will try to take some lessons from it to use in our project. The ideas of having brief daily meetings, the daily standup, is something we would like to take away from and have some form of short team meeting every other day.

**Pair Programing**

The core concept of pair programing offers more benefit to us right now than the actual practice itself, this is not to say that the actual practice doesn’t offer us any benefits but rather that we will benefit from working in pairs. In pair programing all code to be sent into production is created by two people working together at a single computer. Pair programming increases software quality without impacting time to deliver. It is counter intuitive, but 2 people working at a single computer will add as much functionality as two working separately except that it will be much higher in quality. With increased quality comes big savings later in the project.(Wells, D(1997, 1999))

We can make it easier on the new student’s in Upstage, the two presenting this proposal, by working in tandem with the more experienced team members. this will also offer new input from the new students that the more experienced students may not have thought of.

**Design Science**

Design Science takes the traditional scientific method and applies it to researching for the development of a product.

UpStage’s main problem is its dependency on other dying technologies to perform certain functionalities.

Design Science is initiated when a practical problem exists. This approach will use artifacts (ie methods, models and frameworks) to find a new generic solution for the given issue and this has to be applicable to many similar situations. It will allow us to hypothesise, test and record results then discard or improve on the applied hypothesis. The design science approach will involve the use of proof of concepts to apply concepts, create a testing prototype and record results of the testing prototype (Bider, I., Johannesson, P., Perjons, E., Johansson, L., (2012)). This is the basic idea of how we will apply design science, we want to create proofs of concept for methods of replacing the current technologies with new technologies to find what UpStage is able to adopt to solve the given problem.

**Project Plan**

We currently have a plan for the first semester drawn up by the more experienced group members. This plan runs on two key needs, the first is the need for a new clean and tidy version of UpStage that can replace the current version. The means by which to do this is currently being researched. The second is the need to maintain the current UpStage until the new UpStage is complete so that it can be used by its existing user base. The idea is that two groups will exist at any one time, each will attend to one of the two main needs with team members changing to working in the other group on a regular basis.

As for the second semester the first order of business will be introducing our new members to the UpStage project by hopefully giving them a chance to use the product first hand as early as possible. We want to start off quickly so that the new members can create their proposal and so that we can complete our mid semester review quickly.

After this initial phase we should hopefully be ready to begin the development of a new version of UpStage, dependent on the research conducted in semester 1.  This would involve reassessing the requirements of UpStage, planning the new UpStage and building it, as a whole or iteratively.

While this is happening there is still the maintenance of the current upstage to be attended to. This will be best conducted in a similar way to how it was in the first semester of 2015.

**Communication Plan**

Communication in our group, with the clients and with our supervisor is a complicated matter due to a number of factors. These range from one of our clients living in another country, germany, to our supervisor having a very busy schedule.

We will use different forms of communication to for purposes. E-mail will be the primary method of communicating with our clients and our supervisor outside of meetings with them. We can have in person meetings with our supervisor but this would be difficult to do with our clients due to their location, one living in germany and the other living in “unknown location” We can meet our clients on upstage itself and this is how we have meet them so far.

Our group members have more options for communication available to us, seeing as we are all AUT students we can easily meet up on campus when we have spare time, two weekly meeting times will be set up as well as a set time to work on the project. We also have a group facebook page for voicing ideas, updating others on when non-scheduled meetings are planned, what we have done and other exchanges of information. Then there is the GIThub repository where we can store documents and the public comit of Upstage. E-mail is one of the other ways we have to communicate. Our least desired form of communication is via cellphone, we intend this to be more of an urgent way of contacting group members.

**Skills and knowledge involved**

**Personal Skills**

* **Self-Management**

It is important for team members to be able to able to perform tasks on their own, plan out what they need to do without relying on other members of the group to tell them what to do. If meetings are required to make every decision then the project will simply grind to a halt with people waiting for the next meeting to start on anything.

Also as group members have other commitments it is important that we take care of these commitments as well so that they do not interfere with the project.

there is one last aspect of self-management that our group members will need to perform to the best of their abilities. We need to have enough self control to know when to stop, this is meant both in terms of when to change focus and to not overwork ourselves.

* **Time Management**

In conjunction to Self-Management we also need to manage our time well, this means using the times we have available to meet up properly, ensuring that time for working on this project is set aside and that it is used for working on this project. dedicating time for the different parts of our lives so that we are able to do what we need to do for the project without causing stress over time and deadlines.

* **Problem Solving**

Problem solving is software development in a nutshell or at least what it is supposed to be, as we are all software developers on this team we have been working on solving problems for some time now, and we have gained skills in team collaborations and coming up with ways to improve our team and any issues that may arise.

We may seek guidance from our senior advisors and supervisor when necessary.

* **Communication**

Team members are required to communicate amongst each other as well as with clients and the supervisor so all parties can work well together and maintain a shared understanding of the project and the progress of the team as well as discuss issues or plans for UpStage.

* **Teamwork**

Everyone is expected to make an effort to contribute time and effort into helping the team complete their goals in UpStage. Although each individual have other obligations outside of UpStage, it is still required that members perform their given tasks for this project and help each other out when struggling to produce what is required of them.

* **Attitude**

There are certain expectations of how our group members will behave. We expect all team members to take a sensible and respectful approach to working on upstage, working with the other members of the team, our supervisor, our clients and any other parties we interact with.

**Professional Skills**

* **Ethics**

Ethics is a complicated subject, none the less we should approach all of our software development projects in an ethical manner by which we mean that we shall do nothing to purposely harm anyone or anything in our attempt to complete our project. We will make an effort to cite all works and not practice plagiarism.

* **Project Management**

The group will need to have a firm hand on the project at all times to make sure that it is progressing according to plan.

* **Project/Progress Tracking**

All time and work dedicated to the project must be recorded. This can be achieved in many ways, including work logs which some of our team members are doing, meeting minutes which are recorded and various other methods.

* **Risk Management**

Risk management is easy to get wrong, it is easy to forget that one thing that will inevitably go wrong but by accounting for as many possible risks as we can think of we can come up with ways to prevent them or if they do happen react appropriately. We will create a detailed risk management plan that we will consult plan when it is needed.

**Technical Skills**

These are the technical skills that have been defined in the past as necessary for working with Upstage.

* Python 2
* HTML and XHTML
* JavaScript
* ActionScript 2
* CSS
* Linux/Debian OS
* Terminal and Shell
* GitHub (for code)
* Subversion (for documentation)

Now however since we are investigating new ways to implement the features of upstage to prevent it from becoming redundant we will need to learn other technical skills. As we are still researching these alternatives we do not know all the technical skills we will need.

**Estimation of all costs incurred**

Upstage is an open source project, this means that the only resource that is not per say free is the time of the team and any hardware for the project. The project has been ongoing for quite some time, this means that most if not all of the required hardware is already in place.

In terms of the teams time we intend to dedicate at least 8 hours a week to the upstage project.

**Appendix**

**Auckland University of Technology**

**Bachelor of Computer & Information Sciences**

**Research & Development Project**

**Disclaimer:**

**Clients should note the general basis upon which the Auckland University of Technology undertakes its student projects on behalf of external sponsors:**

*While all due care and diligence will be expected to be taken by the students, (acting in software development, research or other IT professional capacities), and the Auckland University of Technology, and student efforts will be supervised by experienced AUT lecturers, it must be recognised that these projects are undertaken in the course of student instruction. There is therefore no guarantee that students will succeed in their efforts.*

*This inherently means that the client assumes a degree of risk.  This is part of an arrangement, which is intended to be of mutual benefit.  On completion of the project it is hoped that the client will receive a professionally documented and soundly constructed working software application, some part thereof, or other appropriate set of IT artefacts, while the students are exposed to live external environments and problems, in a realistic project and customer context.*

*In consequence of the above, the students, acting in their assigned professional capacities and the Auckland University of Technology, disclaim responsibility and offer no warranty in respect of the “technology solution” or services delivered, (e.g. a “software application” and its associated documentation),both in relation to their use and results from their use.*

**A: UpStage Functionality as in semester 2 2014 proposal**

**Audience**

●      Can access a link to find user’s local time in the homepage

●      Can enter a stage

●      Can send text messages in the stage

●      Can register to be a Player

●      Sign in with the registered account

**Player**

●      Has the same functionalities as audience

●      Can log out

●      Will have tool bar in the stage if the player is assigned to this stage

●      Can put avatars in the stage

●      Can move the avatars in the stage

●      Can stop the avatars in the stage

●      Can select a prop at the bottom right side of the window

●      Can select a background at the bottom left side of the window

●      Can clear the avatars in the stage

●      Can play audio in the stage

●      Can turn volume up/down for audio in the stage

●      Can stop a single audio

●      Can stop all audio

●      Can rename their current avatar

●      Can draw line with chosen color in the stage

●      Can select the type of line to draw

●      Can change the thickness of line to draw

●      Can clear the line

**Creator**

●      Can log out

●      Create a new Stage

○      Enter the stage’s full name and short name as a url

●      Edit a stage

○      Modifying the stage’s full name and short name for url

○      Modify the splash message

○      Can choose if the debug messages are printed on the backdrop

○      Can change the props and backdrops’ color

○      Can change the chat window’s color

●      Upload media so that it can be used in stage

○      Choose the type of media for uploading: avatar, prop, backdrop, audio or video-avatar

○      Modify media’s name and tags

○      For avatar, a voice and be chosen

○      The chosen voice can be tested

○      The number of frames can be selected

○      A local file can be chosen to attach

○      Choose the stages to assign the media to.

○      For prop, the number of frames can be selected

○      Choose the stages to assign the media to

○      For backdrop, the number of frames can be selected

○      Choose the stages to assign the media to

○      For audio, choose the type of this audio: sound effect or music

○      A local file can be chosen to attach

○      Choose the stages to assign the media to

○      For video-avatar, choose from existing streams in the /media/video directory

○      Choose the stages to assign the media to

●      View an uploaded media

○      Display specific media by choosing filters

○      Add more filter to display an eligible media

○      Can remove the filter

○      Can reset the searching filter and result

○      Search by media’s tags

○      Edit an uploaded media by modify name, voice and the stages to assign

○      Can display the detail of the media

○      Can save changes

○      Can delete the media

○      Delete the media even if in use

●      Edit players:

○      Can modify their password

○      Can modify their email

○      Can modify their user type

●      Enter a stage

○      Can send text message in the stage

○      Be assigned to a stage :

○      The message sent can be read out as a chosen voice

○      Has the tool bar(same toolbar as player’s) in the stage if the creator is assigned to this stage

●      Can create a new player account

○      Can enter the username, password, confirm password and email address

○      Select the type of the new player account: player, maker, unlimited maker, admin or creator

○      Able to save

●      Can edit existing player details

○      Can display the detail of all existing player accounts

○      Can select one of them to change email, password

○      Update the changes

○      Delete the player’s account

●      A link to edit page mode

○      Can edit the homepage

○      Can edit the workshop

○      Can edit the player page

○      Can edit the stages page

○      Can edit the sign up page

○      Submit the changes

○      Reset the page to be default

**Maker**

●      Has the same functionality as creator except they can’t create new players or edit.

●      Has an upload limit for file sizes when uploading files (10MB)

**Unlimited Maker**

●      Has the same functionality as maker

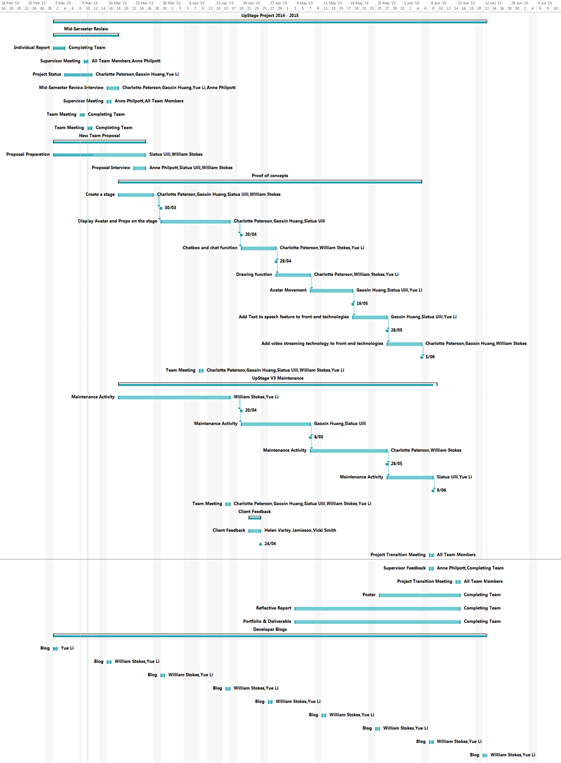
●      Has no limit for file size when uploading media

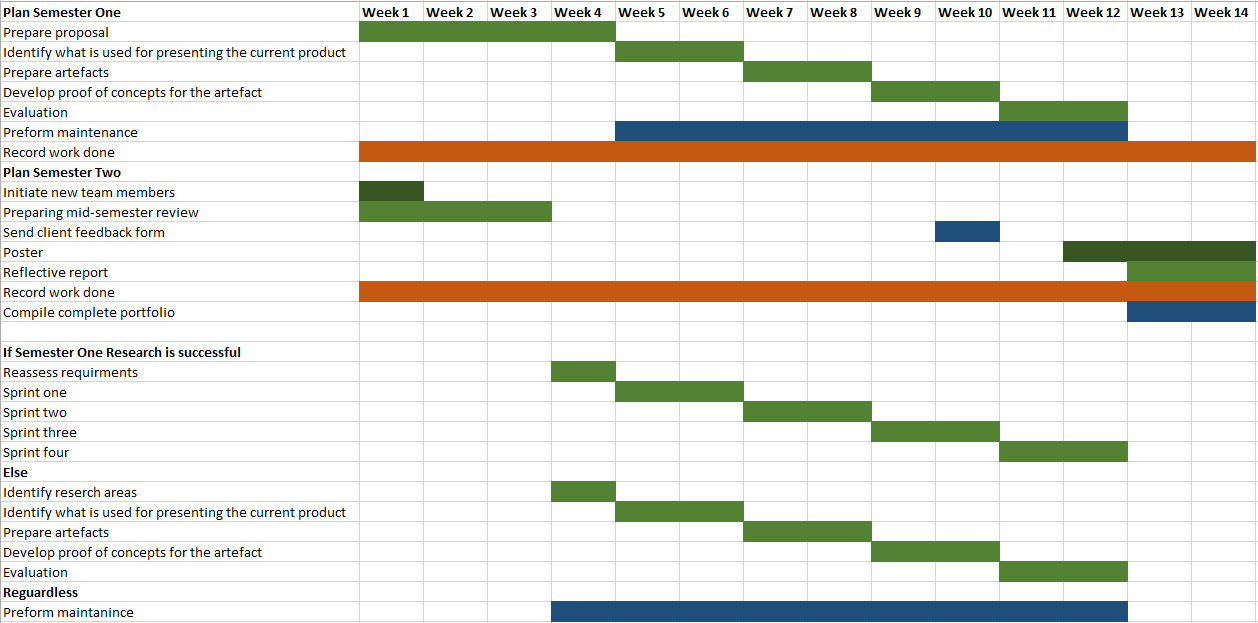
**Admin**

●      Has the same functionality as creator but can’t edit or create creators.

●      Has an upload limit for file sizes when uploading files.

**B: Graphic of planning for semester 1 by Yurek Li**



**B: Graphic of planning for semester 1 & 2 by William Stokes**

**References**

James, M.(n.d.)*The Scrum Reference Card* Retrieved from <http://scrumreferencecard.com/scrum-reference-card/>

Wells, D.(1997, 1999) Pair Programing Retrieved from

<http://www.extremeprogramming.org/rules/pair.html>

Bider, I., Johannesson, P., Perjons, E., Johansson, L., (2012)*Design Science in Action: Developing a Framework for Introducing IT Systems into Operational Practice*