Integrated Project 3

Project Plan

Shoogle

Contents

[Project Background 3](#__RefHeading__1032_201142130)

[Project Justification 3](#__RefHeading__1034_201142130)

[Project Scope 3](#__RefHeading__1036_201142130)

[Target Audience 3](#__RefHeading__1038_201142130)

[Competitor Analysis 4](#__RefHeading__1040_201142130)

[Envisaged Functionality 4](#__RefHeading__1042_201142130)

[Expected Implementation Environment 5](#__RefHeading__1044_201142130)

[Project Non-Functional Requirements 5](#__RefHeading__1046_201142130)

[Proposed Lifecycle for the Project 6](#__RefHeading__1050_201142130)

[Justification of Chosen Lifecycle 6](#__RefHeading__1052_201142130)

[Organisation of Group 7](#__RefHeading__1054_201142130)

[Allocation of Roles 7](#__RefHeading__1056_201142130)

[Risk Analysis 8](#__RefHeading__1060_201142130)

[Risk Management Strategies 8](#__RefHeading__1062_201142130)

[Platform Analysis 10](#__RefHeading__1066_201142130)

[Software 10](#__RefHeading__1068_201142130)

[Hardware 10](#__RefHeading__1070_201142130)

[Required Skills 10](#__RefHeading__1072_201142130)

[Staff Development 11](#__RefHeading__1074_201142130)

Project Introduction

# Project Background

This project focuses on a game where the player is put in the role of a university student, who must overcome challenges typical of daily life in a university by balancing their time, work, money and stress levels.

The project consists of eleven members, split into four sub-groups:

| Design | Art | Audio | Programming |
| --- | --- | --- | --- |
| Suneil Rakhra | Stanislav Valov | Michael Rennie | Alisdair Muircroft |
| Mitch Alexander | Katarzyna Gostanska | Ross Dalziel | Andrew Linsday |
|  | Vladislav Prokudins |  | Sean Cameron |
|  | Tom Watson |  |  |

# Project Justification

The group felt that undertaking this project would allow each and every member to contribute to something bigger than themselves, and in so doing would provide an opportunity to develop everyone's skills, from hard skills such as learning specific software (for which see below), and soft skill including team-building, project management, working to tight deadlines, negotiation and compromise.

The group chose this type of project to work on as it was felt that the theme of university stresses could be mapped on to in-game challenges, and that learning, adapting and developing through university matched common game mechanisms such as “experience points”, skill training and steadily-increasing difficulty curves. Every member of the group has had experience of coping with university life, and as such, everyone's personal lived experience can be used to draw ideas from and get inspiration from, which could make it easier for each group member to invest in the project as a whole, as well as to output ideas that would resonate with players who were themselves trying to cope with university life.

# Project Scope

## Target Audience

Our game's target audience is **university students**, **aged between 17-26**, who have a **basic literacy in gaming**. The content of the game – in particular university life – would be best suited to this demographic, as it reflects topics and issues that are encountered by the demographic on a daily basis.

After creating and distributing an online poll aimed at current and former university students, it was found that 78% of respondents used videogames to relieve stress. The types of games they played were varied but balanced, with 7 preferring games that were relaxing, 13 preferring games that took their mind off of their problems, 10 enjoying games that energised them to tackle their problems, and 8 preferring games involving other characters dealing with their own problems. With this in mind, the group moved towards designing a game that focused on exhilaration and excitement, without necessarily providing a calming or soothing effect.

The survey also showed that a majority (52%) of people who responded did not play games more when they were stressed, or less when they were stressed. As a result, the group decided that a game only aimed at calming the player would have little life outside of times when students felt stressed, i.e., before coursework deadlines and before exams.

Lastly, the survey showed that many respondents felt they and other students were ill-prepared for the stresses of university life. To that end, the group decided to gear the game towards raising awareness of how hectic and stressful university life can be, as opposed to dealing with it after the fact.

We have deliberately chosen not to aim the game towards specific sexes/genders, ethnic groups or other demographics, but we accept that due to the game's theme (university life), its constituent members (visibly able, predominantly white, living in the West), and the scope of the project itself (a 12-week intensive university project), the game will be necessarily biased towards white able-bodied individuals with some financial stability.

## Competitor Analysis

When creating a new game, especially a new intellectual property, market research is key in the initial design stage of development. This ensures that companies do not saturate the market with games that are identical or games that gamers do not want to play. Most games that relate to stress are about relieving stress and are usually small internet games. However there are professional developed games that are specifically developed to combat stress. These companies are not always focussed on stress but create content for general well-being.

‘Brain Resource’ is a company that specialises in mental health and brain training. This company helps to promote positive and efficient work environments by using their various products. These products vary to test and improve upon a person mental health and the brains performance.

‘SuperBetter’ is another company that has been set up to specifically deal with stress and self-improvement. They have created a website this certain activities for the player to accomplish in the real world to help improve upon their life. These activities help promote a healthier mental and physical lifestyle.

The internet is also has a large amount of stress relieving games. However, these games tend to be made by independent games developers or individual people. There are also websites that are filled with mini-games very simple and have sections with games that were created to relieve stress. StressReliefPig is a website that has been set up to help people cope with stress. The website contains tips games to relief stress and sells toys such as stress balls.

When looking through the large amount of games that are to relieve stress, there are two distinct groups of games: Games that involve destruction and games that calming and usually invoke a large amount colour and music.

Destructive games are games that allow the player to destroy a certain object, such as a computer or a mobile phone. One of the most popular games of these games is called “Desktop Destroyer”. In this game the player can use various tools, such as hammers, chainsaws and flamethrowers, to “destroy” their desktop. Destructive games also usually have elements of violence within them. The violence is quite often directed at a Boss or Manager in a generic workplace. The artistic style is usually in a cartoon fashion and the level of violence can be anywhere between slapstick to killing the character.

Calming games tend to have more variety to them but there are always elements that are generally the same. The engage our auditory and visual senses by using vivid colours or distinct art style and soothing music.

While these two types of games are very different in their approach to reduce the players stress level there both have something in common, submission. When people play these games they tend to want to switch off and not think about whatever is stressing them out. They are playing the game for the sake of playing the game there is very little achievement or difficulty involved in these games. Another thing that the games have in common is that they tend to be very short and last usually between thirty seconds and a couple of minutes.

However the product that Shoogle aims to create will highlight to students the stresses they may face at university and educate them about how prioritise and manage their work load and social life. When put in to this context there are very few serious games within the field to compete with. The biggest competition may come from other groups assigned to this task, as they will also be trying to create a similar product.

It must also be noted that some people play video games to relieve stress but the video games they play may not be marketed as stress relieving games. Through the team survey, over 78% of the participants said that they played video games, along with other activities, to reduce stress.

## Envisaged Functionality

In the game, the player character is continually running forward on a path. The path is segmented into squares, and each square has a number on it that counts down until “exam time”. While on each square, a number of Activity Icons randomly appear on the screen in various places – one for Study, one for Work, and one for Party. The player can click as many icons as they like per square in order to perform those actions, which will change the level of each of the meters they have.

The player has three meters – work, money, and social – which fill up or decrease whenever the player clicks one of the floating Activity Icons – Study, Work or Party.

* Knowledge Meter increases whenever the player hits a STUDY icon, and decreases when they hit a WORK icon.
* Money Meter increases when the player hits a WORK icon, but decreases when they hit a PARTY icon.
* Social meter increases when the player hits a PARTY icon, but decreases when they hit a STUDY icon.

Certain squares on the path may have bonuses or penalties, which are indicated when the player steps on them. For example, there may be a square that doubles the Knowledge points added when the player hits a Study Icon, or a square that cancels any Social points when hitting a Party Icon; because the game is fast-paced, players can make mistakes and accidentally pick a Party Icon when it may not give them points. These interactions allow for the meters to change and vary over time.

There is also a Stress Meter, which acts as the player character's overall “health” - if it reaches maximum, the player loses the game. The stress meter slowly increases over time, but can be decreased by deliberately not choosing any Activity Icons – this causes the Knowledge, Money and Social meters to all decrease, but prevents the player losing the game.

If any of the meters reaches 10% capacity, the amount in the stress meter is increased slightly.

## Expected Implementation Environment

The main development software the group will use was specified in the project tender, which is **Unity 3D**. Almost all of the team-members have experience with using Unity 3D, and as such, the time required for learning a new software environment is drastically reduced, allowing for more time to create assets and other content. The group also decided that Java/C++ would be used for most of the implementation, as the programming team felt it is the programming language they are strongest with, but that it also allows them to research new ways in which to use it for game development.

## Project Non-Functional Requirements

A number of non-functional requirements were identified that would also influence the game’s development, including:

* The game must be developed using Unity 3D
* The game must be developed using Java
* A working prototype of the game must be finished by week 12 of the trimester.

Project Organisation

In this section, Shoogle's organisational structure will be outlined.

# Proposed Lifecycle for the Project

The lifecycle proposed for this project is SCRUM. This lifecycle has been analysed to determine the requirements of the methodology. It has also been compared to other lifecycle methodologies and agile methods to determine what is the most appropriate for this project.

The SCRUM methodology is a high level approach which focuses on the management of the product production. The SCRUM methodology can also use practices from other methodologies such as extreme programming and the rational unified process. SCRUM also has defined roles for persons involved in the project these are the “Scrum Master” who fills a role similar to a project manager, the “Product holder” who represents the stakeholders in the project and their interests and the “Team” which contains all the developers of the project.

SCRUM consists of four main sections:

1. The first section is the **Product Backlog**; this high level document contains descriptions of all features and items to be included in the product. These items are sorted by what their priority is in the finished product.
2. The second part of SCRUM is the **Sprint Backlog**, which contains the items to be worked on in the sprint. The items to work on are decided on at the Sprint Planning Meeting. This is a meeting with all members involved in the project before a sprint. The Product holder presents what features they want completed in this sprint the team then discuss these feature and decide what they will be able to produce in the time allotted. The items that are agreed upon to be worked on during the sprint are put into the sprint backlog. The sprint backlog cannot be changed until the end of the sprint when the meeting is over, thus the requirements for the sprint are “set in stone”.
3. The third section of SCRUM is the **Sprint** itself. The sprint is the main iterative process of the methodology. A sprint consists of one thirty day sprint with continuous one day sprints. The one day sprints consist of a Daily Review Meeting were all members of the team spend fifteen to twenty minutes discussing what work is to be done on that day. These meetings have certain rules that must be followed these are the time limit must be constrained to, all attendees must stand, it must be at the same location and time every day and each member will answer three questions. These questions are what you did yesterday, what are you planning to do today and do you have any problems that will stop you completing the task set to you. This meeting allows the Scrum Master to keep track of progress throughout the project. At the end of the sprint there is a Sprint Review Meeting which reflects on the progress made throughout the sprint. Any items not complete are put back into the sprint backlog to be review in the next sprint planning meeting.
4. The fourth section of SCRUM is the **Working Increment** of the software. This is the current version of the software which is showed to customers at regular intervals to get feedback and allow for continuous testing.

# Justification of Chosen Lifecycle

There were a number of reasons the SCRUM lifecycle was chosen to be used in this project. As SCRUM is an iterative process problems found in modules or the original plan can be modified in the next iteration, this is superior to methodologies such as the waterfall life cycle which does not allow this iterative process thus the problems stay.

Due to the fact that SCRUM creates its features in a modular fashion, features can be show to the stakeholders earlier to give a better idea of what they are looking for in the final product, it also allows faster testing to the customers which will give constant feedback on the current version of the product. This is possible as the stakeholders and customers are treated as members of the team and are involved in the development of the product which might not be done in other methodologies. The daily meetings allow constant updates on the progress of the project which highlights the amount of work done by each individual and promotes a collaborative effort as members of the team might not want to be seen as not making a contribution on the project.

Another reason SCRUM was chosen was that the fact that the features are decided at the start and not changed much throughout the project just the priority of them means the project is easier to manage and has clear goals that the team has to work towards in each sprint and will give easier risk management as it will allow the risk of each requirement to be calculated separately this giving greater clarity on the risk of the project as a whole. The Sprint allows the use of other practices such as XP which will allow a more efficient workforce, thus the product will be completed quicker and to a higher standard.

Due to these reasons the SCRUM methodology was chosen to be used in this project.

# Organisation of Group

The group is split into four smaller teams – design, programming, audio and art – and each of these teams except audio has a team Lead, who co-ordinates and oversees the work performed by that team, each of whom were elected during the group's first meeting; because the audio team are not involved in the group until week five or six. The group also has a project co-ordinator, who maintains and manages the group as a whole.

# Allocation of Roles

|  |  |
| --- | --- |
| **Design**  Suneil Rakhra (*Lead Designer*)  Mitch Alexander (*Project Co-ordinator*) | **Audio**  Michael Rennie  Ross Dalziel |
| **Art**  Stanislav Valov (*Lead Artist*)  Katarzyna Gostanska  Tom Watson  Vladislavs Prokudins | **Programming**  Alisdair Muircroft (*Lead Programmer*)  Sean Cameron  Andrew Lindsay |

The Lead Designer, Programmer and Artist will be responsible for ensuring that every member of their sub-group fulfils their duties and reports any problems with the project to the Project Co-ordinator.

The Project Co-ordinator is responsible for organising the project as a whole, including taking minutes at meetings, making other team-members aware of approaching deadlines, deal with grievances between team members, scheduling meetings, and overseeing the project pipeline so that no team members are left without work due to a backlog of work in another sub-group.

Risk Management

During development, there is always a possibility that the team will encounter difficult circumstances that can negatively impact the progress of the project. To safeguard against potential weaknesses and threats, a number of possible risks have been identified and ranked by the probability of their occurring, and the impact that they would have on the project as a whole.

Once probable risks were identified, a number of ways to counteract risks were identified. These include **avoidance measures**, which are intended to lessen the probability of the risk occurring, and **contingency plans**, which are intended to lessen the impact of the event if it occurs.

## Risk Analysis

|  |  |  |
| --- | --- | --- |
| **Risk Event** | **Probability** | **Impact** |
| 1. A team member leaves the project. | Low | Medium |
| 2. The project's progress is behind schedule during an important deadline. | Low | Medium |
| 3. The client's requirements are not met by the project. | Low | High |
| 4. The user's requirements are not met by the project. | Low | High |
| 5. There is a grievance between two or more team members. | Low | Medium |
| 6. A team member does not have the skills to complete a given task. | Medium | High |
| 7. Assets needed for the project are unavailable. | Medium | Medium |
| 8. A team member repeatedly performs poorly. | Medium | Medium |
| 9. A given asset or element of the project cannot be made due to time or skill restraints. | High | Medium |

## Risk Management Strategies

|  |  |  |
| --- | --- | --- |
| **Risk Event** | **Avoidance Strategies** | **Contingency Strategies** |
| 1. A team member leaves the project. | * Facilitate communication between team members to encourage team members to discuss problems early and openly | * Alert supervisor (Brian McDonald) and module leader (Gianna Cassidy). * Reassign team member's duties to other team members from the same department. |
| 2. The project's progress is behind schedule during an important deadline. | * Provide consistent feedback to other team members about the current progress towards deadline * Ensure all team members are aware of their responsibilities and deadlines. | * Create emergency meeting to assess where time-consuming problems occurred and why. * Redistribute tasks if necessary and work overtime until deadline is caught up with, and significant progress is made towards next deadline. |
| 3. The client's requirements are not met by the project. | * Meet with client and supervisor regularly * Ask client and supervisor questions regarding their requirements to ensure a full needs elicitation has been done * Assess how decisions made in the project fulfil client's requirements in project documentation (available on GCU Learn) | * Assess what project assets do not fulfil the requirements and remove them. * Arrange a meeting with client and supervisor to perform another needs elicitation. |
| 4. The user's requirements are not met by the project. | * Perform a survey to gauge general requirements. * Use social media to elicit feedback from fans, testers, target audience and players. | * Arrange a meeting with client and supervisor to perform another needs elicitation. |
| 5. There is a grievance between two or more team members. | * Facilitate communication between team members to encourage team members to discuss problems early and openly. * Encourage a positive, collaborative atmosphere and do not tolerate bullying or negative behaviours. | * Arrange separate meetings with team members with grievances to discuss problems * Arrange meeting with team members to discuss grievance and negotiate a compromise * If grievance continues, inform supervisor and/or module leader. |
| 6. A team member does not have the skills to complete a given task. | * Encourage team members to consider their own skills before embarking on given tasks and to report if they feel uncomfortable or unable to complete tasks. * Distribute various simple and complex tasks over time to team members to allow them to show their abilities. | * Discuss with another team member from the same department if they can switch tasks, or take on more tasks. |
| 7. Assets needed for the project are unavailable. | * Utilise systems that are “always on”, e.g., Facebook, Dropbox, and encourage their use | * Discuss with another team member from the same department if they can switch tasks, or take on more tasks. |
| 8. A team member repeatedly performs poorly. | * Facilitate communication between team members to encourage team members to discuss problems early and openly. | * Arrange meeting with team member to discuss their performance, and if it continues, alert supervisor and module leader. |
| 9. A given asset or element of the project cannot be made due to time or skill restraints. | * Encourage team members to consider their own skills before embarking on given tasks and to report if they feel uncomfortable or unable to complete tasks. * Compile workflow breakdown charts to determine reasonable timescales for assets to be created | * Negotiate with another team member from the same department to see if they can take on more tasks * Assess the priority of the asset/element with relevant department, and if it can be replaced or ommitted. |

Hardware & Software Requirements

## Platform Analysis

The game currently being developed by us is a game in which a player moves along a path as objects appear on the screen for a set time and the player must click on the objects to collect them and keep certain meters from getting to the top and other to get higher before they reach the end of the level. These objects will be randomly spawned on the screen and will also be destroyed after a length of time is destroyed. All of these elements will in the end determine the platform that the game is made for, this and the target market. The platforms considered include PC, Console, Tablet, Web and Mobile devices.

As part of the project specification, the group was instructed to use Unity 3D development environment, which supports a number of programming languages and allows us to create software for a wide range of platforms including Windows, iOS, Android, Consoles and Web. Ultimately the programming team decided that Java was the most suitable for our game as it allows for great flexibility around many platforms as most hardware supports Java. This also allows for potential porting between platforms should the opportunity arise.

## Software

The platforms each bring their own advantages and disadvantages. Windows is the most widely used front where almost everyone has a Windows PC in their home. This would allow for easy access to allow other people to test the game in the comfort of their own home and would allow us to utilize the full capabilities of a desktop computer. RAM limitations are unlikely an issue as the game itself is likely to demand little CPU power and is not particularly RAM intensive; an up-to-date desktop computer should have no issues running the game. The issues with running on Windows and indeed all desktops is that they are not mobile, and this will restrict when and where the game can be played; as this game can easily be played in short spurts rather than in hour-long sessions, this may not be ideal.

The iOS platform has many of the same advantages as Windows, however iOS is not specific to desktop computers and would allow for multi-platform use. iOS spans across phones, tablets and even mp3 players; this means that the market for the game would be expanded further than simply focusing on players using a Windows desktop. It also adds in the element of mobility as the iOS phones would allow for the game to be played on the move. It would also make testing easier as Mac computers are provided in the university lab rooms. However, when working with iOS it is very difficult to get the hardware and software specifications correct for all the possible devices that the game may be played on; for example, the RAM limitations for an iPod would be different for those of a Mac. This could cause issues if the games hardware specifications become very specific. Also, iOS devices tend to be aimed at a market with a moderate amount of disposable income; although many students nowadays tend to buy iOS devices, they are by no means ubiquitous for all students.

Android is similar in terms with iOS devices in that it is multi-platform and includes tablets and mobile devices. Android is more applicable to students as it tends to be a cheaper alternate to iOS devices with many of the same applications, however the hardware tends to be less powerful so things would have to be simplified, or at least not as CPU intensive as the desktop computer counterpart; iOS and Android would not be able to cope with the stresses of a large game compared to a desktop computer.

## Hardware

Desktop computers tend to be more powerful in almost all aspects of hardware and software compared to all other platforms, they allow for many different set ups and tend to be very user specific. The CPU and RAM wouldn’t really be an issue for this type of game and the game play wouldn’t be bothered by much. The major flaw with desktop gaming is the lack of mobility in the platform itself and the drop in/drop out nature of the game would not lend very well to the idea of being able to do this as you could only play it under very controlled circumstances.

Tablets are very mobile and tend to have less system specs in terms of the GPU and memory available to it. It also has constraints of battery life and will simply turn off if battery power runs out. All these disadvantages come with a major advantage in the fact that the tablet is highly mobile. Usually coming with a touch screen to allow the game to take some interesting changes. The idea of a touch screen would implement into our game very well with the idea of touching things as you run. Tablets can be played anywhere, but the games that are typically played on these aren’t complex or graphically taxing as the CPU is running many tasks and doesn’t have a lot to spare.

Mobile devices are very similar to tablet devices were mobility is the primary object is to be as mobile as possible, these devices are also mobile phones and tend to be a lot lower spec compared to tablets, small in size and tend to have more buttons and less on the touch screen, even though nowadays the phones tend to be going more mobile. Most of the applications on the mobile phones tend to be run on the CPU rather than being allocated some work to an independent graphics card. These devices tend not to have much in terms of RAM or CPU power but they can be played anywhere as long as there is battery within the device.

Web is an interesting platform to work with as everyone has access to the web in some form or another. This allows the game not only to be played on almost any system, but the game becomes global very quickly. The major disadvantage with developing a web game is the idea of streaming, the game must be fairly low quality and must be able to run on systems via streaming all the data from the game, using the internet, which means most of the game must be preloaded and not most of the work must be done on the computer streamed to rather than through the code.

In the end we decided to develop the game for a Windows Desktop Computer, as it allows for porting from a computer to mobile devices, and it is far easier to take things away from a game to make it playable on a mobile or tablet device.

## Required Skills

The skills required for this project include:

* good working knowledge of Java & Java APIs
* good working knowledge of Unity environment
* understanding of the specifications of the Windows platform

As a whole, the programming team is well-versed in all of these areas, although time may be required to find, understand and integrate new Java techniques and code libraries that are unfamiliar to the software engineers. All members of the programming team have experience with Java, Unity 3D and Windows, and they are accessible enough that time can be taken to experiment with new approaches and assets.

## Staff Development

It is important for members of the team to not only utilise their current working knowledge of the hardware and software required for the project, but to build upon this knowledge and develop their skills further.

Work Breakdown Schedule

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Week** | **Design** | **Art** | **Programming** | **Audio** |
| **1 (14/01)** |  |  |  |  |
| **2 (21/01)** | Submit **Project Plan** |  |  |  |
| **3 (28/01)** | Submit **Design Document Plan**  Work on **Design Document**  Work on **Idea Pitch**  Create **Design Videoblog** | Create **Game Concept Art** | Create **Basic Game Engine** (running, clickable Actions) |  |
| **4 (04/02)** | Submit **Design Document**  Present **Idea Pitch** | Present **Game Concept Art** |  |  |
| **5 (11/02)** | Create **Art Video Blog** | Create **Game Level Art**  Create **Art Video Blog** | Create **game balancing mechanisms (meters, scores)** |  |
| **6 (18/02)** |  |  | Implement **Game Level Art** |  |
| **7 (25/02)** | Present **Tech Demo**  Create **Tech Video Blog** |  | Present **Tech Demo**  Create **Tech Video Blog** |  |
| **8 (04/03)** | **Testing & Balancing** | Create **GUI art** | Perform Alpha Test | Get **Audio Requirements** |
| **9 (11/03)** | **Testing & Balancing** |  | Implement **GUI Art** | Create **Audio Assets** |
| **10 (18/03)** | **Testing & Balancing** |  | Implement **Audio**  Perform **Beta Test** |  |
| **11 (25/03)** |  |  |  |  |
| **12 (1/04)** |  |  |  |  |
| **13 (8/04)** | Present **Final Pitch** | Present **Final Pitch** | Present **Final Pitch** | Present **Final Pitch** |

