Dinner Party

- Technical Risk Assessment -

Andrew Rimpici

- EGD-220-04 -

TABLE OF CONTENTS

DIFFICULTY SCALING SYSTEM

Document Scaling System

Scaling Overview

- This document uses a 1 through 5 difficulty scaling system where 1 represents very easy, and 5 represents extremely difficult.

Difficulty Representation

- The difficulty will be represented using puzzle pieces shown below:

	**Represents 1 out of 5** Very Easy - No trouble to implement
	**Represents 2 out of 5** Easy - No trouble to implement but may take more time than usual
	**Represents 3 out of 5** Medium - More thought than usual went into planning and implementing
	**Represents 4 out of 5** Moderately Difficult - A lot of planning and time went into implementing
	**Represents 5 out of 5** Very Hard - Very difficult to plan and to implement. Most likely would take too long to implement before a deadline

THE DELIVERY PLATFORM

Main Delivery Platforms

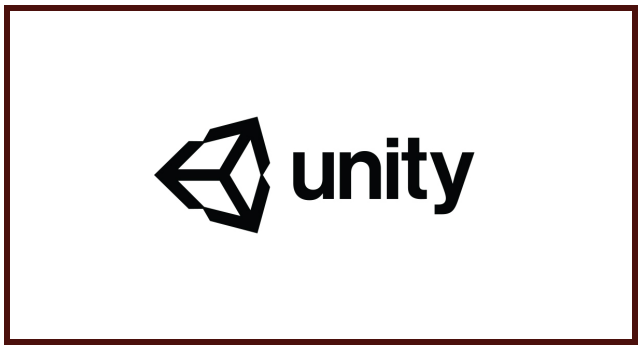
iPad/Kindle Fire/Tablet Market

- *Dinner Party* is designed with iPad heavily in mind.
- The Kindle Fire is also a very attractive option to deliver *Dinner Party* on as it is a popular tablet device and would not take a lot of time to implement support for it.
    - The only downside to supporting the Kindle Fire is maintaining support for the multiple screen sizes offered; but we would have to do this for Android tablets anyway, so this should not be of huge concern.
- The game utilizes and was intended for tablets as these devices can be easily passed around.
- Tablets are also a good option as they provide touch input which is easy and intuitive for most users.
- *Dinner Party* is heavily focused on passing a device around a group of people, making tablets a favorable delivery platform.

THE DEVELOPMENT ENVIRONMENT

Programming Environment with Unity

Using The Unity Engine



- The game will be developed in the Unity Game Engine. Unity comes with a lot of built-in features that are right at the hands of the developer's. Having access to these features eliminates the need to construct a game engine tailored to the game from the ground up, thus *saving a great deal of time*.
- A majority of the team is comfortable using the Unity Engine which translates to a more efficient workflow.
- Unity is also extremely useful because it allows for cross-platform development.
    - This comes in handy as the developers can simply export the game to a mobile or tablet version of the game with little overhead—*saving time and money*.

THE DEVELOPMENT ENVIRONMENT
(Continued)

Resource Environment

Adobe Photoshop

- All of the art assets used for the game will be created in the Adobe Photoshop art program. Photoshop is a very widely known and well supported graphics tool that allows for users to create things of all shapes and sizes. It can be used to draw or edit photos. The program mainly works in a raster format and not vector format which works perfectly for *Dinner Party* as the assets will be pixel based and not vector based.

THE DEVELOPMENT ENVIRONMENT
(Continued)

Version Control

Subversion

- The choice of version control for this project is Subversion (SVN). In Pineapple, there is a subversion repository linked where all of the team members can monitor and update the game files. The use of Subversion will help the team make sure all of the game files are up-to-date and where they should be in regard to meeting deadlines.

- In the repository on pineapple there is a root folder called “Unity Tree” that holds all of the different branches and builds for the game. This makes it more convenient to pinpoint certain builds that might have bugs in them in addition to keeping backups and snapshots of different build milestones.

GAME MECHANICS AND SYSTEMS

Game Systems

User Setup

- At the beginning of each game, the users need to pick how many players are in the game, as well as how many different types of roles they would like in the game. Once the users input this information into the game, the game randomly assigns each user a role.

Pass and Play

- After all users are assigned a role, the game allows for each player to see their given role. The users pass the tablet to the current user and then that user can click a button to read what role the game gives them.

Three Course Meal

- The game goes on for three rounds before the game ends and determines the winners. Each round the players are able to swap, rotate, or look at meals on the table to try and figure out which meal is poisoned and who is poisoning them. After each round, there is a voting phase.

Meal Interactions

- Each round, all of the living players are able to take an action. The actions a player can choose from is:
    - Rotating the Lazy Susan - Rotating the Lazy Susan causes all of the meals on the table to rotate in the direction the player desires. The player can control the direction with their finger by dragging the touch screen.
    - Swapping two meals - A player can swap two meals by tapping on a meal and then tapping on a second meal. Once the second meal is tapped, the two meals switch.
    - Looking at a meal - A player can look at a meal by long pressing a meal. After a certain amount of time, the player can lift up their finger and a popup window will show whether the meal has special abilities or not.
    - Giving your action to another player - A player has the option to give up their turn, allowing another player to use it, by tapping the other players name tag instead of a meal on the table. When a player taps another players name tag, their turn will be passed along to the tapped player. The chosen player must then take their turn right away. *Giving up a turn may be useful if there is a chemist player that can check if meals are poisonous.

GAME MECHANICS AND SYSTEMS
(Continued)

Voting Phase

- After each round, all remaining players can choose to vote out another player if they deem them to be suspicious. The voting phase only occurs if a player died in the previous round, or if there are more than four players to vote someone out, as the cops need ample amounts of witnesses.

Game Mechanics

Touch Input

- Because *Dinner Party* is designed for tablets, the game uses a touch input system. Touch is fairly easy to implement using Unity which saves on time which can be used in other areas of development for the game. In addition to being nice to develop, touch is intuitive for users that use tablets.

THE ART PIPELINE

Art Dimensions and Scaling

Ratio and Screen Size

- Since *Dinner Party* is primarily a game for iPads and Tablets, background art assets are drawn with a 4:3 aspect ratio in mind as this is the ratio that the iPad uses. *Dinner Party* uses dark, muted colors, interspersed with bright whites. The main colors outside of the monochrome are red and purple.
- The largest iPad resolution is 2734x2048 pixels and we plan to tailor the art to this screen size. Info/Menu/Next/Back buttons are all planned to be created with 593x224 pixel dimensions to fit nicely on the iPad screen size. The rotating “Lazy Susan” in the center of the table in *Dinner Party* is 1367 x 1024 pixels. Scaling to the largest iPad model will ensure that the art assets will only downscale when played on other models. We chose to do this because downscaling is always better than upscaling art as upscaling leads to blurriness and/or pixelation.

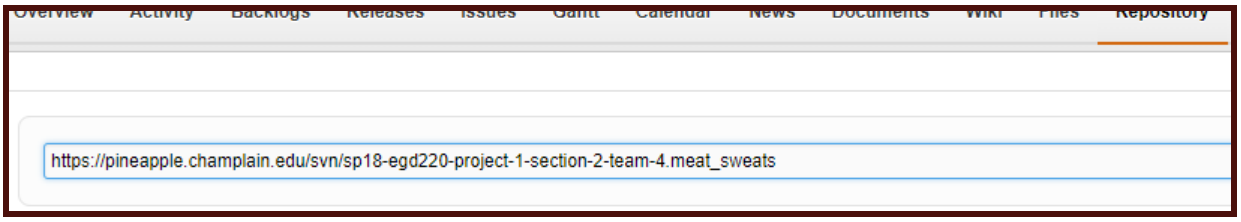
Using the Repository

Step 1 - Download TortoiseSVN

- The first step is to download TortoiseSVN. TortoiseSVN is a client software that allows the user to upload documents and files to a server for other members with access to view and manipulate. TortoiseSVN is free, easy to learn, and easy to use.

Step 2 - Navigate to Pineapple

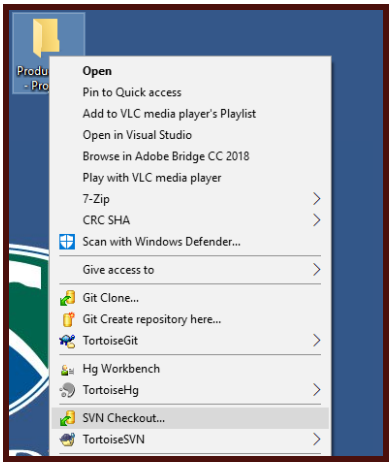
- Once downloaded, the user should log into Pineapple and under the team's "Repository" tab, there will be a URL.



THE ART PIPELINE
(Continued)

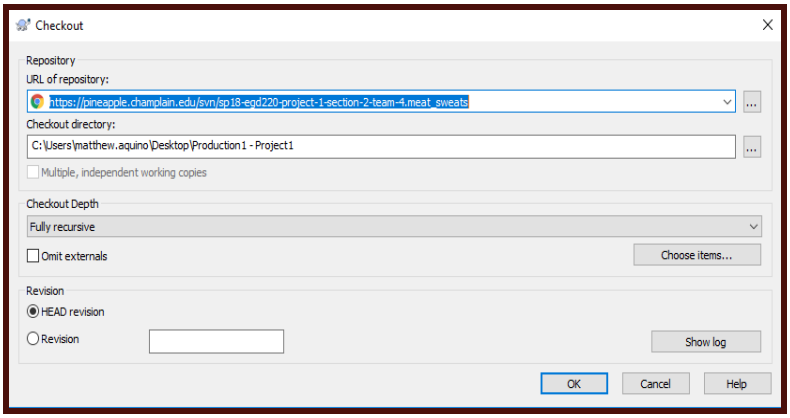
Step 3 - Create TortoiseSVN workspace folder

- The user should copy the URL to the computer's clipboard by using the hotkey CTRL + C. Once that is done, the user should create a folder on their desktop or other location of choice. Right-click on the folder and choose the option called "SVN Checkout."



Step 4 - Paste Pineapple URL in TortoiseSVN

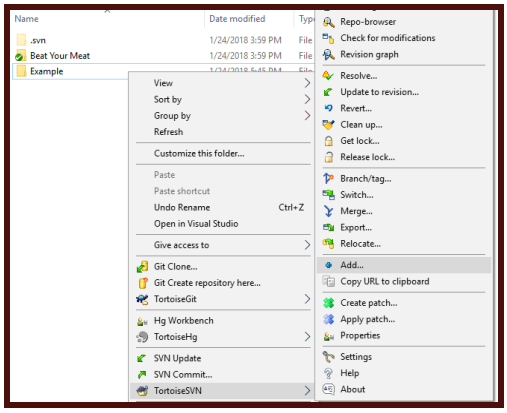
- The URL from the previous step should automatically be placed in the window that pops up. The user should click "OK" and log in with



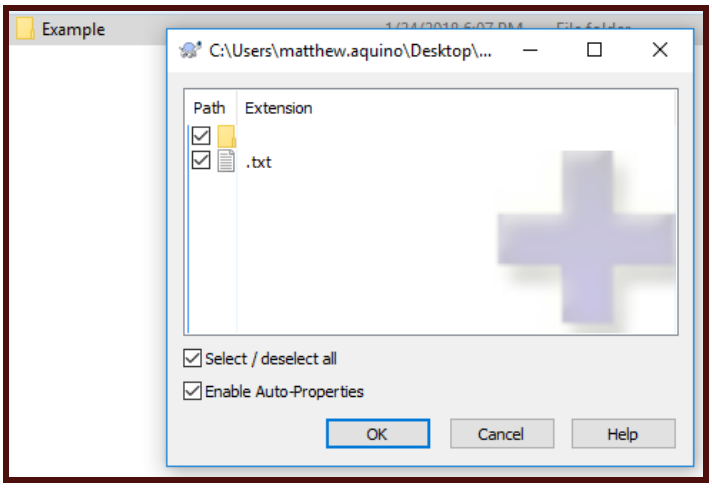
their Pineapple credentials when prompted. This will load all folders in the repository into the folder they've created.

Step 5 - Add Files/Folders to the Repository

- If the user wishes to add a new folder to the repository, all they simply have to do is copy the new folder into the repository folder, right-click on it, highlight the "TortoiseSVN" option, and select "Add."



- Select the folders you wish to add, or choose specific files to add, and click "Ok."



THE ART PIPELINE
(Continued)

Step 6 - Commit New Files to the Repository

- Finally, simply commit the folder by right-clicking on it, and selecting “SVN Commit...”. Click “Ok” and the folders should be uploaded to Pineapple. It is very important to always update and commit any new versions of work so that the team is always up-to-date.

Loading the Art in Unity

- The art will be loaded directly in by the designers and / or programmers once the art has been uploaded into the repository.

THE DESIGN PIPELINE

Unity Engine

GUI Interface

- *Dinner Party* focuses heavily on UI elements to make the game understandable for the players involved. Designers as well as Artists are able to go into Unity and add in the desired UI elements in certain Scenes which will later be wired into the game by the Programmer.
- The Unity build is set up to utilize as many different Scenes as possible. This allows Designers and Artists to easily go into the build and set up the UI however they want on any of the available Scenes. Once the Designers and Artists are finished adding the UI elements to each Scene, the programmer can go in and wire up the UI to work with the internals of the game.
- There is currently a Designer on the team that is familiar with programming which allows for more customizable options in code for him to tweak and add to the build.
- UI is very important for *Dinner Party* to work as effectively as possible, which is why it is encouraged for the Designers and Artists to plan and implement intuitive and helpful UI placement.

Milestone Updates

Milestone #1

Deliverables:
Artist Concepts ● Three styles for possible art direction were created. ***Visual Design Document (VDD)*** ● Guide for core mechanics. ***Game Rules Document*** ● Rules pertaining to the game were outlined. ***Physical prototype*** ● Board was created and tested.
Goals for Next Milestone:
● Create a functioning digital prototype that effectively conveys the game's core mechanics

Milestone Updates
(Continued)

Milestone #2

Deliverables:
Digital prototype ● Start to an implementation of the core mechanics. ***Art Style*** ● Solidified the art style for the game ***Technical Plan*** ● The technical plan has been started and is currently being iterated upon.
Goals for Next Milestone:
● Deeper implementation of Core Mechanics

Milestone Updates
(Continued)

Milestone #3

Deliverables:
Digital prototype ● Continue work on the implementation of the core mechanics and add in more functionality to the game rounds and player interactions. ***Technical Plan*** ● The technical plan is currently being iterated upon.
Goals for Next Milestone:
● Have a working build that can potentially be QA tested.

Milestone Updates
(Continued)

Milestone #4

Deliverables:
Digital prototype ● Have a build that can be used for QA **almost** on its own. ***Technical Plan*** ● The technical plan is currently being iterated upon.
Goals for Next Milestone:
● Have a completely working build for QA. ● Implement some art assets into the game.

Milestone Updates
(Continued)

Milestone #5

Deliverables:
Digital prototype ● Have a build that can be used for QA on its own. ○ Still fairly buggy ○ More art is in the game to give it a better atmosphere. ***Technical Plan*** ● The technical plan is currently being iterated upon.
Goals for Next Milestone:
● Have a working game to play for next milestone. ○ Reduce bugs in the build to allow for better playability. ● Polish the build and add in remaining art assets.

Milestone Updates
(Continued)

Milestone #6

Deliverables:
Digital prototype ● A fully functioning prototype that is working and playable. ○ All art is now in the game. ○ Music is in the game to immerse the players. ***Technical Plan*** ● The technical plan is completed for the final milestone.
Goals for Next Milestone:
● NA