Operational Concept Description (OCD)

Doodle Dash

Candice Academia

Johnny Chi

Nicholas Ruiz

Robert Steiminger

Stephen Sing

3/14/2017

Table of Contents

Operational Concept Description (OCD) i
Version History ii
Table of Contents iii
Table of Tables iv
Table of Figures v

1. Introduction 5

1.1 Purpose of the OCD 5

1.2 Status of the OCD 5

2. Shared Vision 6

2.1 Benefits Chain 7

2.2 System Capability Description 8

2.3 System Boundary and Environment 8

3. System Transformation 10

3.1 Information on Current System 10

3.2 System Objectives, Constraints and Priorities 10

3.3 Proposed New Operational Concept 12

3.4 Organizational and Operational Implications 16

Table of Tables

Table 1: The Program Model
6

Table 2: Capability Goals 9

Table 3: Relation to Current System
9

Table of Figures

Figure 1: Benefits Chain Diagram 4

Figure 2: Simple High level relation diagram of the game 10

Figure 3: Use case diagram of the game and menus.
10

Figure 4: Timeline of our project 11

41. Introduction

41.1 Purpose of the OCD

This document delivers the shared visions as well as goals of the stakeholders of the game Doodle Dash. The project owners are each member of the entire team, Dr. Moazeni as the success-critical stakeholder, volunteers as users.

41.2 Status of the OCD

The status of this OCD is currently at the Development Stage of the project. The scope of the game Doodle Dash currently focuses on the core features of the game and as time permits, could be extended to features listed on the backlog. Each team member have been designated as testers for each iteration of the build.

Currently, the game only has one level, and the menu is very basic. All of the menu functions are currently working. We have not yet integrated with the Google Play API. In terms of mechanics and backend of the game, they are mostly complete. The game is in a playable state with all parts of the game mechanic working as intended.

42. Shared Vision

Table 1: The Program Model

Assumption: All iterations of our build works.			
Stakeholders	**Initiatives**	**Value Propositions**	**Beneficiaries**
● Each team member: - Candice Academia - Johnny Chi - Nicholas Ruiz - Robert Steiminger - Stephen Sing	● Making sure to stay in schedule	● Offer a different and more exciting game for entertainment and de-stressing	● Users ● The team

42.1 Benefits Chain

- **Stakeholder(s):** Each team member, Users
- **Initiative:** Testing each iteration of the build, Evaluating what works and what does not work
- **Contribution:** Successful features added to the game after every iteration
- **Outcome:** Stable user interaction, playable game, improved experience playing the game
- **Assumption:** Every featured added to the game during every iteration works

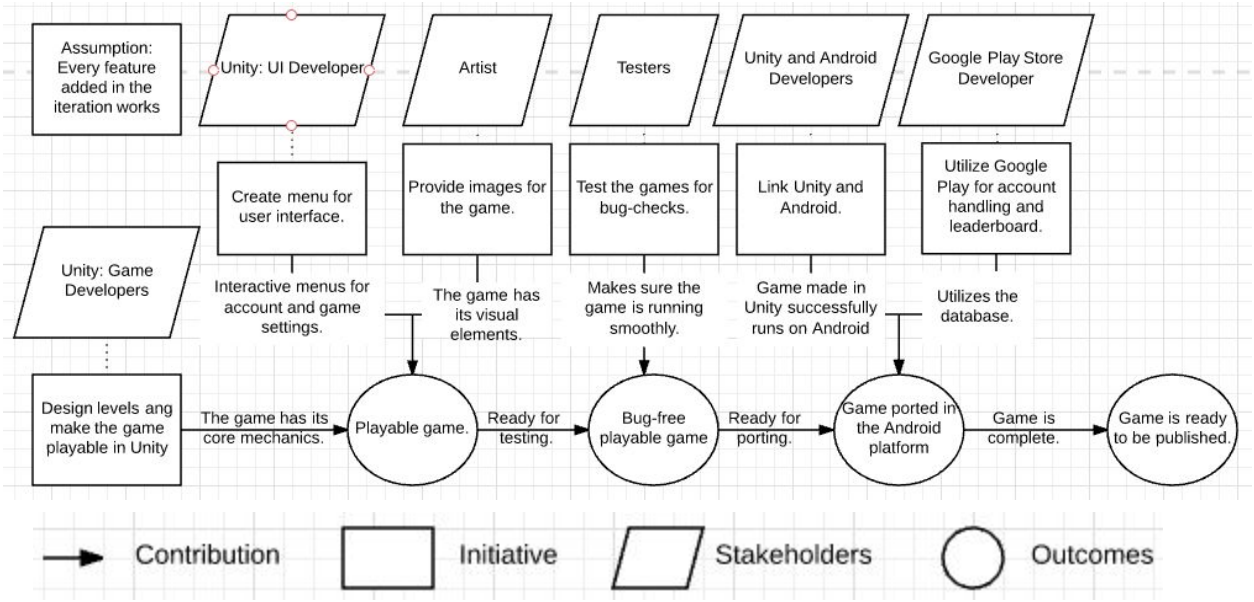

Figure 1: Benefits Chain Diagram

42.2 System Capability Description

This game is casual, easy and fun to play. The simplistic nature of the game makes it enjoyable as you complete the levels by jumping over obstacles while listening to . The closest competitor of the system would be Geometry Dash.

The main reason to use the game would be to have an alternative game to play. We offer a much more difficult experience when compared to Geometry Dash or even other rhythm based games on the market.

42.3 System Boundary and Environment

The team will need to be responsible for delivering and developing a few different systems for the game itself. The game and it's core mechanics have already been finished. We will still need to add in art assets as well as animations. On top of that, more levels will need to be added as well as the Google Play integration. A cleaner UI will be necessary as well as the customizable options. We're assigning different people to be in charge of each of the different aspects of it. In terms of need, having the customizable aspects of the game implemented will be a key component.

43. System Transformation

43.1 Information on Current System

3.1.1 Infrastructure

Our product will run on Android mobile platform but the game will be made in Unity. With scaling, it should work on any tablet or phone that is running on Android 6.0 or higher.

3.1.2 Artifacts

We are just making use of the different classes being provided to us in Unity. The key ones being the 2D Polygon collider. This makes handling everything related to collision much simpler. Another feature of unity that is seeing use is the camera and player controllers. These allow us to have the camera the way we want it, as well as have the controls and jump the way we want.

3.1.3 Current Business Workflow

In terms of workflow, since our schedules are all very different, we are all working remotely for the most part. All communication is handled through Slack with the GitHub plugin in order to give each member an idea of what has been pushed. We have separated out different branches for different components of the project to be worked on. This is so we can all work on different parts of the game without influencing the work of someone else. When different components are finished, we do a pull request in order to merge.

43.2 System Objectives, Constraints and Priorities

3.2.1 Capability Goals

The main capability goals that we have remaining are simply to get the game up and running as well as have the Google Play integration in. Creating a good UI/UX as well as nice animation and graphics will come after that. Here is just a brief overview of some of the capability goals we have right now:

Capability Goals	Priority Level
Have the game up and running!	High
UI/UX	med
Google Play Integration	High
Animation/Graphics	med

Table 2: Capability Goals

3.2.2 Organizational Goals

OG-1: Make sure the game is playable with the core mechanics and features.

OG-2: Make sure the game has Google Play Integration for leaderboards

OG-3: Implement a nice set of graphics and animations on top of the game

OG-4: Improve the UI/UX of the game

Constraints

CO-1: Android platform
CO-2: Zero Monetary Budget
CO-3: Unity

3.2.3 Relation to Current System

Table 3: Relation to Current System

Capabilities	Current System	New System
Roles and Responsibilities	Game - needs to be runnable and have basic gameplay and mechanics in.	Core game mechanics are in and are implemented in a good level design
User Interactions	The User will get to play the game	The user will get to play the game
Infrastructure	Unity	Unity, connected with Google Play inside Android
Stakeholder Essentials and Amenities	The essentials will be the game itself, it needs to be playable and usable. In this aspect	The new system will have Google Play integration which will include leaderboards, as well as fleshed out levels and animations
Future Capabilities	Google Play Functionality, connection to all android devices	More fleshed out levels and increased levels of customization

43.3 Proposed New Operational Concept

3.3.1 Element Relationship Diagram

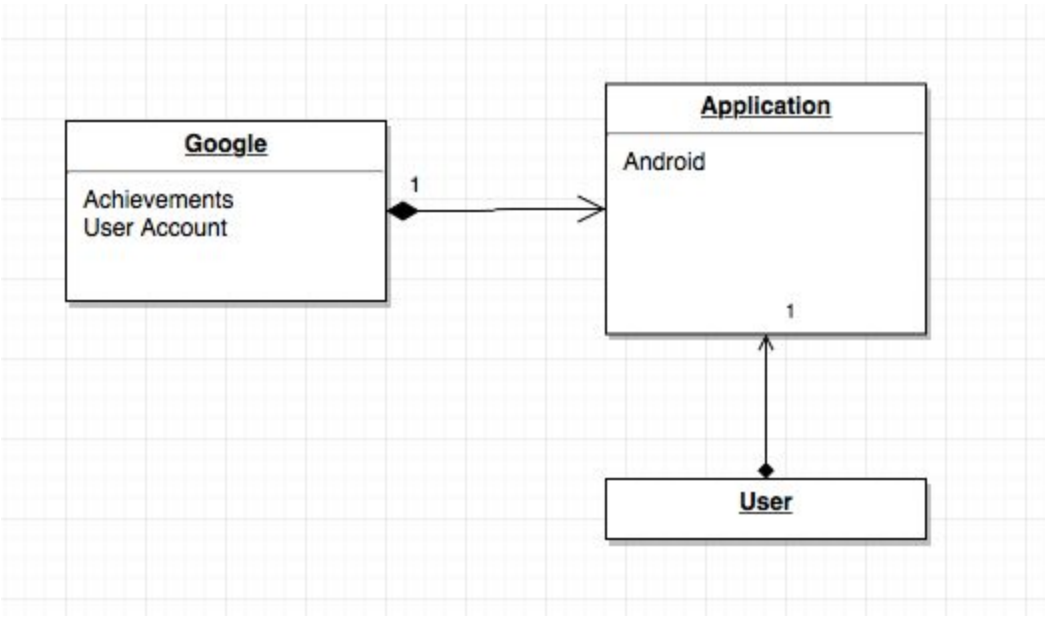

Figure 2: Simple High level relation diagram of the game

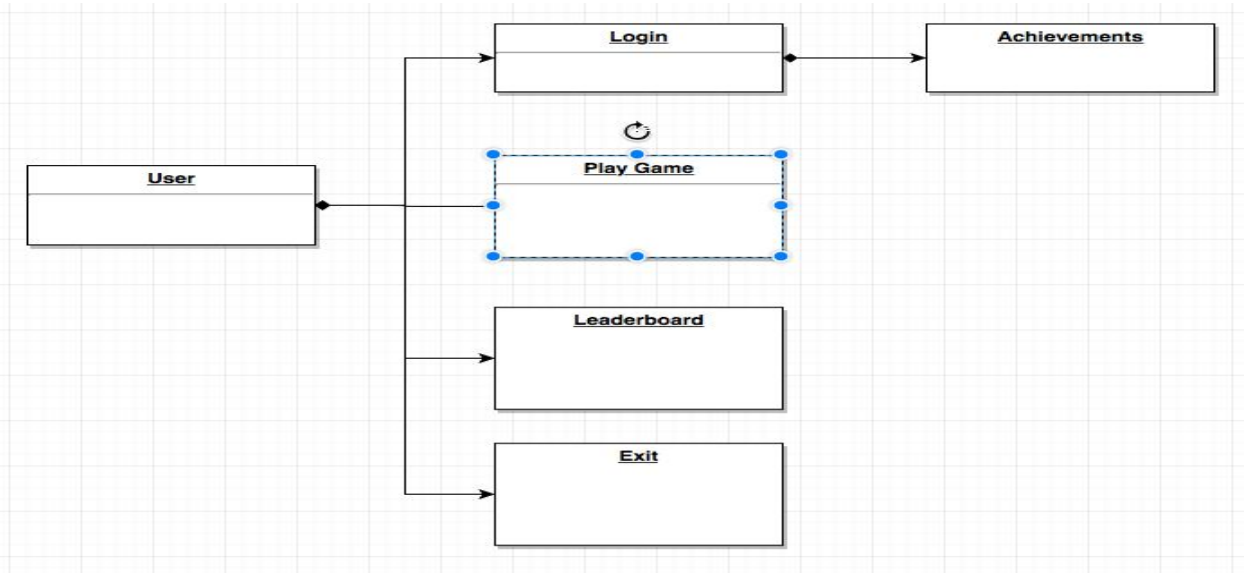

Figure 3: Use case diagram of the game and menus.

43.4 Organizational and Operational Implications

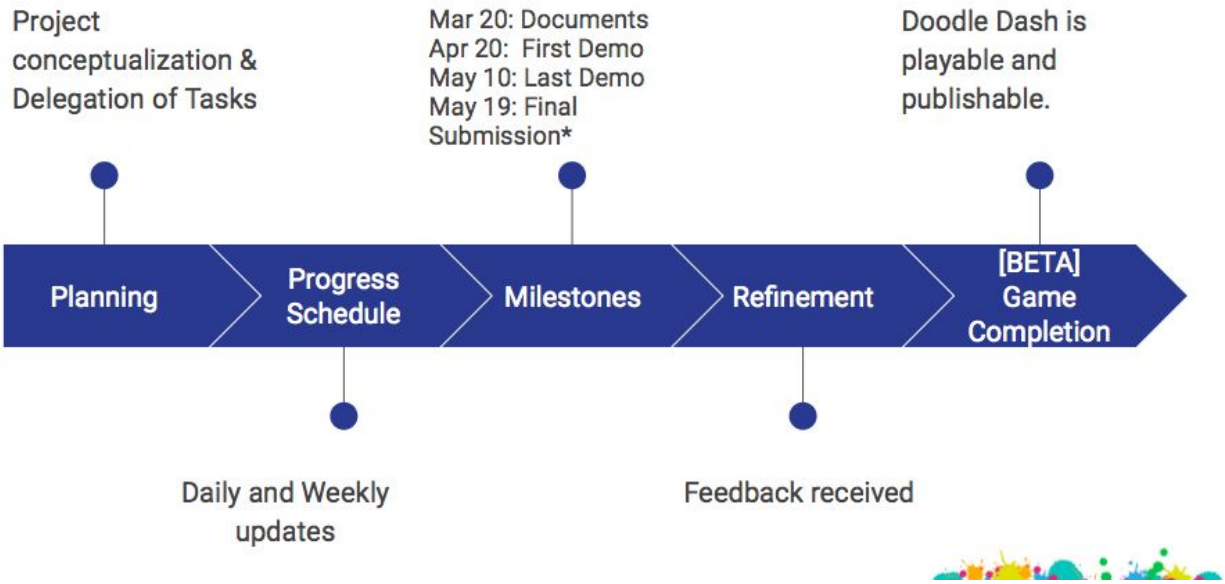

Figure 4: Timeline of our project

3.4.1 Organizational Transformations

There were several significant role changes to our organizational structure and roles based on the experience level of our group and team members. When working with Unity, many of the group lacked experienced and were retasked to focus on specific areas of the project such as level design and artwork. Although operational stakeholders were not affected, our team was delayed somewhat by the need to learn Unity. Another role change was the addition of the Google Play integration as only a single member had a unique certification which allowed us to implement the services into our game. This lead to an additional time-consuming delay as testing became a factor as it only worked on a single device.

3.4.2 Operational Transformations

Porting over the game from unity to android served as a significant challenge both in terms of testing if devices worked as well optimizing performance on different devices. A significant challenge was the fact that only one device was available for testing purposes which meant that testing was bottlenecked. Another issue was the fact that design on unity was set to fixed positions which meant that the overall layout was not as adaptable in all android formats.