Notes: This document will need to be downloaded and saved to github regularly.

* How often?
* Who will be responsible?

# Introduction

As part of a government initiative to support growth of the game industry in [this region] , an award of local and national government funding has been granted as a startup fund for the development of [the subject game]. Prompted by the need to develop a quality product with limited funds and in view of the public interest; this study was commissioned to recommend the best software development approach. The report covers an explanation of several suitable approaches, their advantages, disadvantages, risks and benefits; and recommends a particular approach as the most suitable for this product.

The report as-delivered also includes a Practice Guide, as well as Tool Guides and a Learning Package. These additional artifacts are provided to enable the development team to put the recommended approach into practice.

## Agile vs Plan-driven Development

There are considered to be two main philosophies which drive software development - ‘traditional,’ plan-based development and agile development (Boehm & Turner, 2009). Both ideologies have their merits and drawbacks, described by Boehm and Turner as the difference between discipline and rigour of process and project adaptability respectively. Any software development project must assess the suitability for either of these main concepts and determine if one, the other or a mixture of both are required.

Plan-driven development evolved out of environments such as aerospace and commercial industries where a focus on documentation and processes and a change-averse environment are valued (Boehm & Turner, 2009). The advantage of this philosophy is an holistic plan for and view of a complete project and its expected outcomes - plan-driven development is predictable.

Comparatively, the agile philosophy eschews an upfront plan and makes smaller, feature-based, working modules of a product which is ultimately developed over several iterations (Meyer, 2014). Agile development environments are change-aware and are inherently more adaptable to evolving requirements and shifting goalposts.

In the case of [this study], an agile approach is favoured for the following reasons:

* Game development is not rigorously regulatedcitation needed
* The development environment is expected to be highly dynamic as it grows from four to (up to) fifty people,
* Product criticality is low
* No legacy code replacement/integration, and
* Product is not likely to require integration with external/custom systems

However, the projected size of the development team is not considered optimal for a fully agile approach. Boehm and Turner (2009) describe several scenarios where teams of 50 - 150 were successful, although not easily managed. It is for this reason the study does not completely rule out aspects of plan-driven development which may be beneficial in this case.

## Product Context

[Sez developing project/product context]

# Agile Development Approaches

This section outlines a collection of suitable development approaches, each drawing on the Agile philosophies of [insert philosophies with references].

## Crystal

[Sez researching Crystal]

“The Crystal Clear method covers smaller projects; Crystal Orange was the first to be developed and addresses larger projects.” (Meyer, 2014, p141)

Crystal Orange is for “up to 40 people, sitting in one building, working on a system that might cause loss of discretionary monies

**seven principles**

# Recommended Practice

According to Boehm and Turner (2009), “Analyzing the project risks is an effective way to help managers and practicioners determine the right weight of process.”

and

“Asking ‘*Is it riskier for me to apply (more of) this process component or to refrain from applying it?*’ and honestly evaluating the answer can lead to the definition of reasonable, practical and effective hybrids that balance discipline and agility.”

# Practice Guide

## Communication and Collaboration

## Requirements Engineering

## Planning and Tracking

## Change Management

## Quality Assurance

# Tool Guide

## Communication and Collaboration

## Requirements Engineering

## Planning and Tracking

## Change Management

## Quality Assurance

# Learning Package

## Goals

## Target Audience

## Plan

## Required Resources

## Activity

# Executive Summary

# Retrospectives

## Team Retrospective

## Individual Retrospective

### Gabby

### Sarah

### Wayne

# Bibliography

Boehm, B. & Turner, R. (2009). *Balancing agility and discipline.* Addison-Wesley.

Meyer, B. (2014). *Agile!*. Zurich, Switzerland: Springer.