Coal Rush Design Document

Contents

Section 1 – Game Design

Section 2 – Technical Design

Section 3 – Production Plan

Section 1 – Game Design

1.1 General Game Overview

1.1.1 - **Intro**
You play as a Train conductor and are trying to beat your friend in a race while escaping from bandits. You must shuffle coal into a furnace to make your train go faster and avoid incoming obstacles by changing rails.

1.1.2 - **Story**
You play versus your worst enemy, and you must beat them by making it to town before them in a race with **(material)** while being chased by bandits. You control where the rails take you, but so does your enemy.

1.1.3 - **Players role and objectives**
The main goal of the game is to survive longer than your opponent in the race towards the town. You will encounter incoming **rocks, logs and animals** and to avoid them you might have to change rails. Both you and your opponent can change rails for the entire map to make you sure you go in the right direction.

1.2 General Gameplay

1.2.1 - **Gameplay**
The player will control the speed of their train by shuffling coal into a furnace. They will also use their shovel as a lever to change tracks on the map for both themselves and for the opponent to avoid incoming obstacles. If you fall behind the other player, you will lose 1 health point. Each player will start with **3 health points**, and if you lose them all the other player will win.
Animals on the rails can be scared away by using your train whistle, they will then move to adjacent rails.

1.2.2 - **Win condition**
You win the game if you manage to survive longer than your opponent.

1.2.3 - **Lose condition**
If you fall behind your opponent and get towards the bottom of the screen you will lose **1 health point**. If you lose all your lives, the other player will win the game. Obstacles will slow you down, so you will get closer to the bottom of the screen.

1.2.4 - **On-screen information**
- Incoming obstacle

- Health points (carts)

1.3 Play Mechanics

1.3.1 – **Player actions**
By shovelling coal pieces **(golf balls)** into the furnace you will make your train go faster and get higher up on the screen. You can also take your shovel and put it in a designated area and use it as a lever instead to change the direction of the rails. If you pull down a hanging handle you will activate the train whistle that makes a loud whistle sound to scare away animals on the rails.

The player shovels coal to increase speed, pulls the lever to change track direction, and pulls the whistle to move animals.

When playing, the player has to choose between shoveling coal and changing tracks, the two actions cannot be performed simultaneously due to the nature for the input.

The player will have to quickly identify when it's worth it to stop shoveling to go change the tracks, either to help themselves, or to hinder their opponent.

The player can change tracks onto the other players track, ramming and consequently slowing them down. When players ram each other, the player that is furthest ahead will lose less speed.

1.3.2 – **Controls**
Shovel – used to shovel go coal and as lever
Furnace – register how many coal pieces is put in
The main part of the physical input, where the player will shovel coal into. The player will use the shovel to pick up balls, and move them into the furnace. There the balls will roll into a funnel where each ball will pass a sensor that can read each passing ball as a separate input. The balls will be funneled back so that they can be picked up by the player continuously. See section 1.4.2 for the in-game effect.

Lever – changing the direction of the rails
A lever that can be pulled in two directions. When pulled to one side a button will be pressed. See section 1.4.2 for the in-game effect.
Whistle – scare away animals
A rope hanging from above that can be pulled down. The rope is attached to a spring that is holding down a button, when the rope is pulled the button is released. See section 1.4.2 for the in-game effect.

1.3.3 – **Game interface**

Screen shot

1.4 Player Characters

1.4.1 - **Character characteristic**

Trains

The player character is a train. Trains travel along tracks, moving upwards on the screen. The only direct control players have over their character is the forward speed. Trains can be indirectly controlled by changing the direction of the intersections of the rails they are travelling on. Trains have carts attached to them which serve as health points. Losing all carts will cause the player to lose.

1.4.2 How each character is controlled.

Shovel coal

Players will use our unique input to shovel coal which in the game translates to one button press per coal piece shoveled into the furnace, in this section I will be referring to “button pressing” when I talk about the input for shoveling coal.

The rate of pressing the button controls the speed of the train. Each press increases the current coal by one. The velocity of the train is then calculated based on that amount of coal. Speed gained from coal is affected by diminishing returns, so that players quickly accelerate from low speeds, but accelerate more slowly once their speed is high. Since the game is a race between two people, players will want to shovel coal as fast possible to increase their speed.

New: Each time the button is pressed, the train moves forward a small distance. Pressing the button

Trains lose speed over time, but only a very small amount. Trains mainly lose their speed by colliding with obstacles.

Change tracks

Players will use a lever to change the direction that intersections point. Intersection can be switched to either go left or right. All intersection are controlled simultaneously, which means they will all point in the same direction. Some intersections will instead of being able to go both left and right only be able to go in one direction but also go forward. For example, if an intersection can go both left and forward, and the intersection direction is currently set to forward, that intersection will instead go straight. Some intersections only have one direction, and are therefore not affected by changing the direction.

When a player presses a lever direction button, the direction of all intersections gets set to that direction. There is a small delay before the direction can be changed again after having been changed.

Whistle

Players will use the whistle to make animals, which are obstacles, move out of the way of their train. While the whistle is being pulled and held, animals will react to it.

1.4.3 A list and description of all the player character(s) characteristics, including abilities, behaviour, speed, idle animation's, upgradability (through power-ups/specific points in the game) and hit points.

1.4.4 How each player character interacts with non-player characters.

There are no NPC's (animals are considered obstacles).

1.4.5 How each player character is destroyed.

Trains are never destroyed, the player simply loses when their train has lost all of its carts.

1.4.6 Where player characters appear in the game.

Both trains start side by side at the start of each race.

1.4.7 A list of all animation's required to realise all the above-described player character(s) and their characteristics.

1.5 Non-player Characters (N.P.C's)

1.5.1 - **List of NPC's**

Animals (I think these should be considered obstacles).

1.5.2 A list and description of all the NPC characteristics, including abilities, behaviour, speed, intelligence, mobility and hit points.

1.5.3 How each NPC attacks the player.

1.5.4 How each NPC is destroyed.

1.5.5 Where NPC's appear in the game.

1.5.6 A list of all animation’s required to realise all the above-described NPC’s and their characteristics.

1.6 Player Resources

1.6.1 A list and description of all the player resources.

1.6.2 A list and description of all resource characteristics, abilities, lifespan and how they affect the player character(s), NCP’s, gameplay and game environment.

1.6.3 How the player character(s) gains control of the resources.

1.6.4 How each resource affects the game controls.

1.6.5 Where and how each resource appears n the game.

1.6.6 A list of all animation’s required to realise all the above-described player resources and their characteristics

1.7 Game Hazards

1.7.1 - **List of Game Hazards**

- Obstacles: Static obstacles (rocks, logs, etc), animals, canyons
- Bandits

1.7.2 A list and description of all hazard characteristics, abilities, lifespan and how they affect the player character(s), NCP’s, gameplay and game environment.

Obstacles will slow the player down when their train collides with them. Some obstacles will disappear after the player has collided with them, such as static obstacles and animals.

Bandits will detach carts when they catch up to them, which means the player loses one hit point. When a train loses a cart they will get a temporary speed boost to allow them to pull away from the bandits.

The position of the bandits is based on the position of the train furthest ahead. This means that when the two trains are a certain distance apart the one falling behind will lose a cart. The bandits will also slowly move upwards relative to the players, moving faster the more time that’s passed.

1.7.3 How the player character(s) avoids each hazard.

Most obstacles are avoided by changing tracks so that the obstacle isn’t in the path of the player. Obstacles are sometimes avoided by jumping over them, by traveling with enough speed over a jump. Certain obstacles like canyons can only be avoided by jumping over them.

Bandits are avoided by not falling too far behind the other player.

1.7.4 How each hazard affects the game controls.

They don’t

1.7.5 Where and how each hazard appears in the game.

Obstacles will appear on the tracks ahead of the player.

Bandits start outside and below the camera view. They will become visible when the players are a certain distance apart.

1.7.6 A list of all animation’s required to realise all above-described game hazards and their characteristics.

1.8 Game Environments

1.8.1 A comprehensive list and description of the game environment, broken down level by level.

1.8.2 A highly detailed description of the scenery and atmosphere of each level.

1.8.3 A detailed and complete list of all player character(s), NPC’s, hazards and resources that appear in each level, cross referenced to 1.4, 1.5, 1.6 and 1.7.

1.8.4 What objectives the player has to complete to finish each level.

1.9 Critical Points

1.9.1 A list and description the high point of the game, describing what the player has to do and what makes it a high point.

1.10 Level Flow

There is a fine art to level construction. Great games can often suffer with ill conceived and badly designed levels. Ways to design a bad game not only include making levels too hard or too easy, but also making winning vs, losing too arbitrary, or lacking in any form of learning curve. A good game works when the learning curve is just right. The curve needs to be smooth and continuous. E.g. you can see where you went wrong so you try again, this time you get a little further, and again see how you could have done better still, until you win.

1.10.1 A level by level description of how each level flows, including a breakdown of what the player actually has to do in each level, a list of when and where hazards, resources and NCP's appear, and when critical points happen.

1.11 Game Presentation

1.11.1 A full visual overview of the game presentation and its structure from start to finish using a flow-chart.

1.11.2 A full list of all presentation screens, and their function.

The camera is static following the players until one player is ahead by a certain amount. When this "soft" limit is reached, the camera zooms out giving the game more life and attention. Further along the "hard" limit might be reached, at which point the players can not move any further up the game screen.

1.12 Intermission Sequences

1.12.1 A description of all the intermission screens, the information or story they convey and when they happen.

1.12.2 A description of how the game transitions from interactive to passive at each intermission juncture.

1.12.3 Scripts for the above.

1.12.4 Ideas for voice talent/actors.

1.13 SFX and Music

1.13.1 A comprehensive and complete list of all SFX and music for the game, cross referenced with 1.4, 1.5, 1.6, 1.7, 1.8, 1.11, .1.12

1.14 Target Audience

1.14.1 An analysis of target audience and market.

1.14.2 As complete a list as possible of competitive software either released or pending.

1.14.3 Research into current buying trends.

1.15 Target Platforms

1.15.1 A list of platforms the game is being developed for.

1.15.2 Platform specific features where appropriate.

2.1 SKU Details <- (wont be needed, will delete later)

Detail the following information for each project's S.K.U.s;

2.1.1 Basic Minimum and Maximum Platform System Requirements

Including the Processor, memory, CD speed, OS required etc.

2.1.2 Special Requirements

Any other peripherals or memory requirements that are extra-ordinary for this SKU's normal setup.

2.1.3 Localised Versions

What languages are being supported by each SKU

2.1.4 Shipping Media

Amount and type of media that the product is to ship on.

2.2 Third Party Tools and Licences

Any third party tools or technology that are to be used by the product need to be listed here.

2.3 Production Pathways

This section deals with the specific details on how various assets are created and integrated into the final product. These Production Pathways are usually best represented in a diagrammatical form. (See the first section for an example.)

2.3.1 Art

For each of the sections below be sure to include pathways for both static and animated artwork. Try to include all details of necessary commercial and self-developed tools and their hardware requirements Also, if parts of the pathway are provided by a third party you should state it so here……

2.3.1.1 3D Art

Details of the creation of 3D models, animation’s textures etc. and how they are implemented into the game. Below is a rudimentary example…..

2.3.1.2 2D Art

All 2D filmed, rendered and hand art

2.3.1.3 Video

Live and modelled motion video creation..

2.3.2 Sound

2.3.2.1 Sound Effects

2.3.3.2 Music

FM, MIDI, Digital and CD

2.3.3 Localization

2.3.3.1 Text

2.3.3.2 Art

2.3.3.3 Sound and Music

2.4 Memory Management

For all different memory types and mass storage devices please detail a “memory usage” map.
I.e. DRAM, VRAM, CD 3D Card Textures etc.
N.B. For consoles please be sure to include VRAM for NTSC and PAL.

2.6 Equipment Requirements

Please list hardware and software requirements for equipment necessary for the project.

3.1 Introduction

The “Production Plan” section details how the product’s development is going to be managed and performed. It should contain most of the information pertaining to HOW a title is going to be made, with WHAT resources and WHEN constituent parts occur.

When completing this section try to be as concise and unambiguous as possible. For concepts or sections in the document that may be difficult to understand, diagrams or tables should be used for clarification. If technical terminology is used, please include a glossary of terms.

3.2 Software Development Process

This section is intended to give an account of the “Software Development Process” used by your company for games development. Do not deal title-specific issues but detail your general approach to games/software development.

Consider the following:

q Detail any Software Standards or similar your company uses.
q Do you follow any formal method for design, analysis and coding? If not, how is your software lifecycle structured?
q What methods are used to allocate, assign and track project taskes?
q Detail how problem reporting, bug tracking are handled.
q Detail how you test your products.
q Describe your system of date/code/document backup and archiving
q Detail what activities are external to the company. (e.g. filming, motion capture etc.)

3.3 Project Plan

This section should describe, in detail, how you plan to make this title in terms of scheduling resources and the respective costs incurred. It is highly probably that it will be necessary to divide your Project Plan into many “sub-plans” which focus upon specific project areas.

Present the Project Plan as a hierarchy and in some logical order. The top level “Overview” plan first, followed by the sub-plans that add increasingly more detail.

For each Project Plan or sub-plan there are certain sections that need to be completed. These sections are not mandatory and only need to be completed if and when relevant.

The Sections are :-

3.3.1 Section Title

The title of this part of the plan and a brief description of what it represents. (e.g. Level 1 game design.)

3.3.2 Schedule

This is a representation of the tasks or activities in this plan in terms of how much time has been allocated for them. This should be represented in diagrammatic form (A Gantt chart or similar) at an appropriate time resolution.

q For each activity in the plan its start/end dates and estimated man-weeks to complete should be visible. |f appropriate, a percentage complete should also be displayed in the chart.

q When activities are complex or are scheduled over long periods of time these should divide these into more manageable sub-activities.

q Any deliverables or milestones should be included.

q Include the “critical path” in the diagram.

q If activities are dependant on other activities this should be shown on the diagram.

3.3.3 Manning Profile

This section should depict how many staff are allocated to each task at any one time if not already done in 3.3.2

The following may help you in the production of this section;

q Represent the manning profile in a diagrammatic form for simplicity using the same time resolution as the schedule section. (A bar chart or histogram is preferred)

q Only represent the number of staff required, it is only necessary to identify individuals when they are critical to the task.

3.4 Risk Analysis and Contingencies

This section should identify the key risks and contingecies in this Project Plan. It is these areas that need to be focused upon to guarantee the project arrives as near to time and budget as possible.

The following is a list of potential risk areas that may help you complete this section;-

3.4.1 Scheduling

Time is certainly our most valuable resource. If when scheduling activities you feel that they are ambitious time-wise, you should these particular activities as risks. In this way you can see what areas in the project are going to need the most attention.

3.4.2 Manning

Did the manning profile identify any key areas of potential under/over manning? These will be the peaks and troughs in the manning profile. If no tasks can be re-scheduled to overcome these then they should be stated as a risk.

3.4.3 Technology

If the title is dependant upon the development of any new technology, or contains development areas you are not yet familiar with, then these need to be identified as risks. These items, if possible, should be scheduled as early into the project as possible and prototyping used to determine their feasibility.

3.4.4 Dependancies

Look for activities that have many dependencies. These are potential project risks and should be scheduled to be completed earlier whenever possible.

3.5 Organisation Chart

Provide a organisation chart of the people on the project with their title and responsibilities identified.

Appendix A – Game Design Terminology

Game Story

If a gameplay improvement dictates a storyline change, it should happen. The story should never take priority over the design.

Character

'Character' doesn't necessarily mean human. It's basically the object(s) that the player or computer controls during the game. It could be anything from a vehicle to a cartoon character.

Hazards

'Hazards' are non-character items that provide a threat to a player. They can be anything from spikes in the floor to gaps in the landscape. Anything that isn't a non-player character that can harm the player is a hazard.

Resources

'Resources' are supplementary items that provide help to a player. They can be anything from weapons to extra energy or even temporary vehicles. Anything that isn't a player character that can help the player is a resource.

High Points

The reason for the 'high points' is to allow the designer to see the overall follow of the game's high spots – the set pieces that people will remember the game for. What they are is fairly subjective, but bottom line, they should essentially be mini climaxes – mid bosses, critical puzzles, end bosses – that stand out from the bulk of the game. When all the critical points are listed, they can be tweaked to add pace and flow to the game. When all the critical points are listed, they can be tweaked to add pace and flow to the game so that there are 'quiet' patches of main game to contrast with the climatic 'high points'.

Level Flow

'Level Flow' is a fairly detailed description of how the level works and what the player has to do. This description should contain enough information to enable a designer to either construct a paper representation of the level, or design it straight on screen.