Requirements and Analysis Document for Denty

1 Introduction

1.1 Purpose of application

This application is a recreational game in the fashion of a classic platform game with co-operation capabilities. Collaboration will be required to achieve success in the game.

1.2 General characteristics of application

Mainly this application works like a usual platform game (i.e. Super Mario), but with a twist. It's meant to be played by two players at a time; one controlling the main character (Denty, the building beaver) with the keyboard and one building the maps, as they are played, with the mouse.

1.3 Scope of application

The scope includes a standalone desktop application with local 2-player gameplay. The application does not include networking capabilities for gameplay. The game will not support a single player mode.

1.4 Objectives and success criteria of the project

- 3 playable levels: a tutorial level, a fast-paced level and a puzzle-level to demonstrate the different kinds of levels.
- Criteria of usability must be met in order to achieve success.

1.5 Definitions, acronyms and abbreviations

The two players are separated by their input methods, The "Keyboard player" controls the main character. The "Mouse player" controls a crosshair pointer that performs auxillary tasks.

"Block" - a solid part of the environment that invokes a collision with any moving object upon proximity

"Denty" - the main character controlled with the keyboard and represented as a sprite in the game view

"Environment" - The set of all objects that compose a playable level. Interaction with the environment means interaction with any object within the above mentioned set.

"Enemy" - a character residing in a level, with the aim to kill Denty. Enemies can be eliminated by Denty and by the mouse player throwing blocks at it.

"Level" - an area containing an environment that Denty is to navigate through. Has a goal in the end of the navigation, when Denty reaches this goal he clears the level and continues to the

next step in the game.

“Power-up” - A special block that Denty can take by touching it. Taking a power-up will either give Denty extra strength or provide the mouse player with more blocks.

2 Proposed application

2.1 Overview

Denty is a co-op platform game where one player controls the character with the keyboard. The other player places blocks on the level using the mouse. To beat a level the players must reach the end of the level by solving puzzles and helping each other.
In a typical level the viewport automatically scrolls forward. If the players cannot keep up the game is lost.

2.2 Functional requirements

- A menu screen with the possibility to start a game and set different options.
- Keyboard player can control the main character which is an immediate part of the environment.
- Mouse player can achieve a variety of tasks including: Placing block, moving block and throwing block.
- Three different levels to demonstrate the capabilities of the game. One tutorial level, one fast-paced level and one puzzle-level.
- Moving enemies who may harm Denty

2.3 Non-functional requirements

2.3.1 Usability

- Possible to reach the goal and advance to the next level
- Possible to start a game through the user interface.

2.3.2 Reliability

While the game should be reliable enough to play through the supplied levels the initial goal is to produce alpha-level stability.

2.3.3 Performance

Being a 2d game it shall run on entry-level hardware, capable of running Windows XP or later.

2.3.4 Portability

Denty will be developed in java, hence it should be runnable on all major desktop platforms. First release target will be a standalone application.

2.3.5 Implementation
All programming will be done in java, using Eclipse IDE. We will possibly Slick for OpenGL rendiring and different utilities and JBox2d for physics simulation.

2.3.6 Verification
We will develop automated tests, and of course perform user tests too, to see if it's playable.

2.3.7 Packaging and installation
We aim to at least provide a runnable jar-file, and if there's time, installers for Windows, Mac OS X and Debian based Linux distros.

2.3.8 Legal
We will probably use GNU GPL to license Denty, but we could be forced to use another license if third party libraries require it.

2.4 Application models

2.4.1 Scenarios

The use cases are relatively simple. No need for scenario

2.4.2 Use case model
See use case document.

Use cases priority
1. Moving
2. Jumping
3. Placing blocks
4. Completing a level
5. Throwing blocks
6. Taking damage
7. Defeating enemy
8. Defeating enemy by jumping
9. Starting game from menu
10. Adjusting controls

2.4.3 Static model

See appendix.

2.4.4 Dynamic model

See appendix.

2.4.5 User interface

The application will use a fixed GUI following standard conventions. It will look like a usual platform game, made for full screen playing. The game will be playable on virtually all resolutions, but it might be easier to play on higher resolutions because the GUI isn't scaled, hence more of the level will be seen at higher resolutions.

2.5 Possible future directions

The game is not meant for single play but it might become possible to adjust the controls to accomodate for that. However the pace of the game should make playing alone very hard.

Cutscenes between the levels to provide a story and context to the players. The story will be cheesy and full of early 90's era game references.

Scoring system and score-giving in-game items (coins or equal). Rewards based on reaching a score milestone such as extra lives or power-ups. Achievements would be another way of doing this.

Porting to a web-applet which will allow online distribution and gameplay.

Enemies appear as part of one of these categories: Enemies defeatable by Denty, or enemies not defeatable by Denty. Enemies of the second category can only be defeated by throwing blocks at them.

Bosslevels with more complicated enemies at the end.

Network vs. mode 2 vs. 2 where the keyboard players try to attack each other, and the mouse players try to throw blocks at the opposing Beavers.

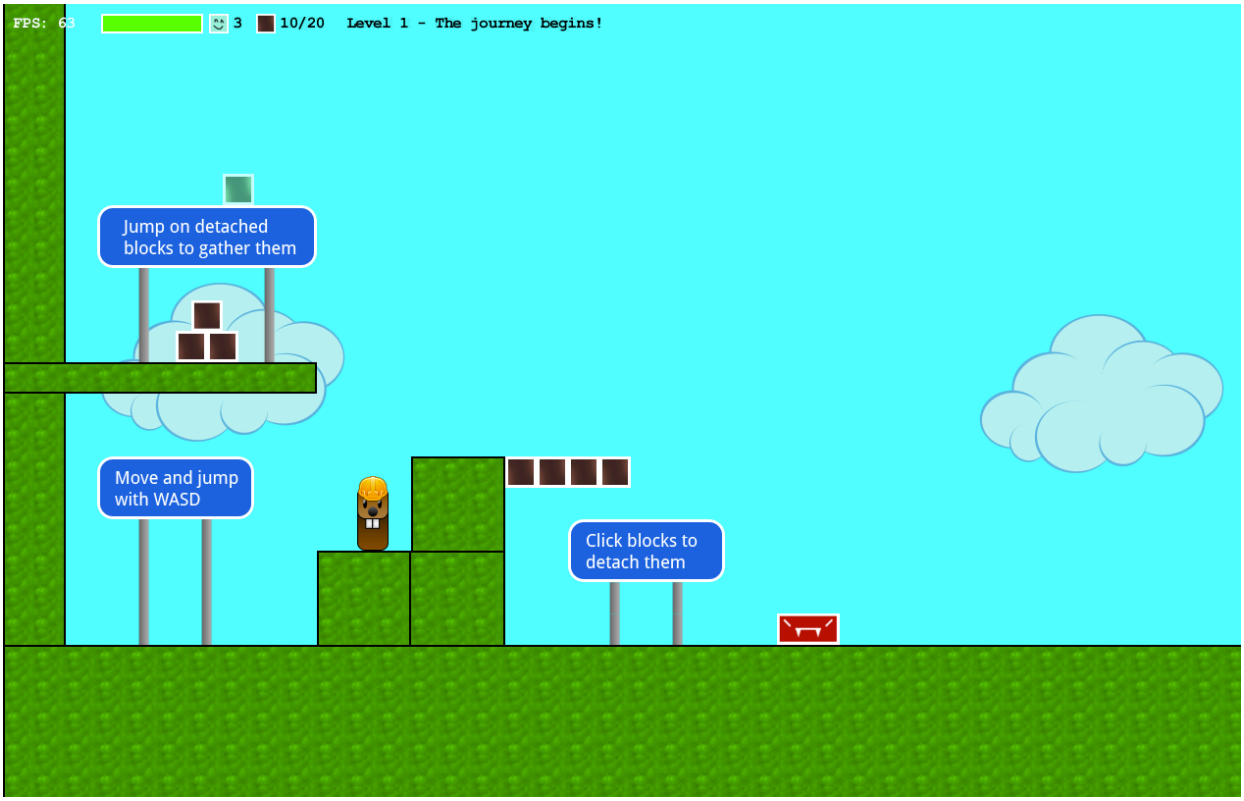
2.6 References

APPENDIX

Use cases

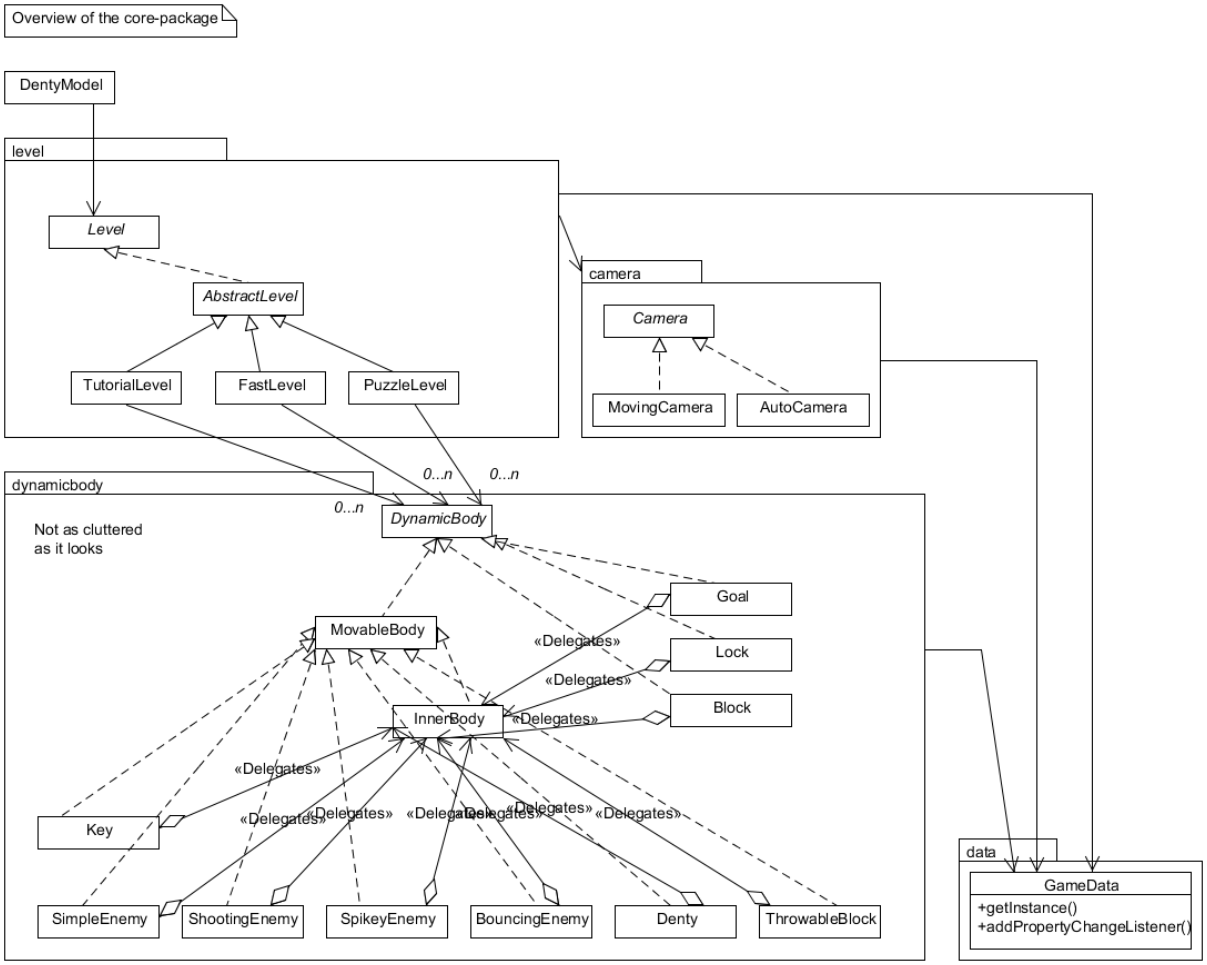
See Use case document

GUI

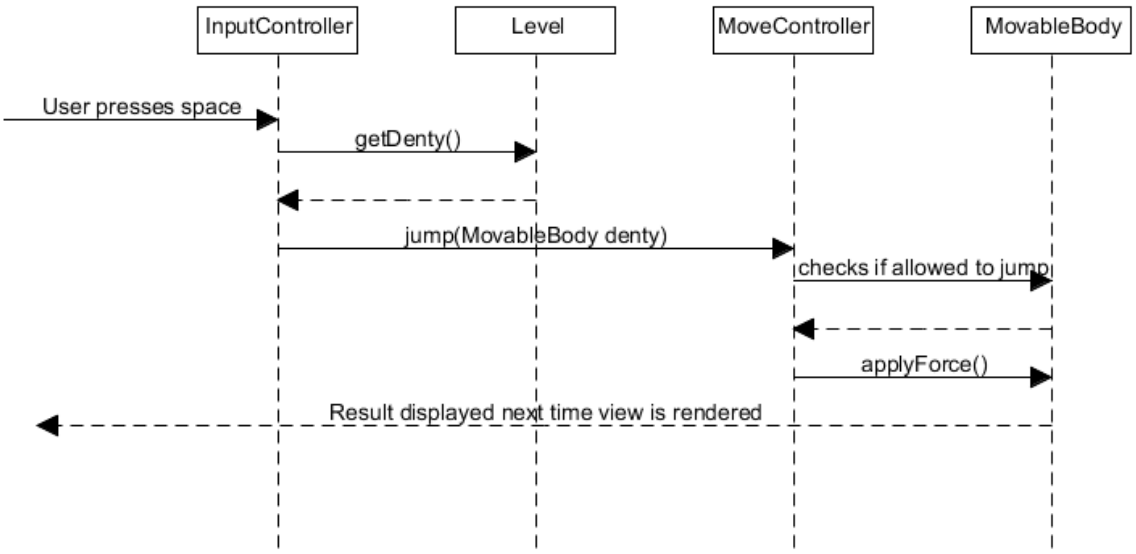


Object model

Static Model



Dynamic model



Version: 0.1.1

Date : 2011-05-26

Author

Sebastian Ljunggren
Nina Malm
Andreas Svanström
Max Witt