Product planning

TI2805 Context Project, Computer Games, Group 1 (Building Blocks)

Gert Spek
gspek
4216806

Niels Bakker
ncbakker
4161394

Jehan da Camara
jdacamara
4207858

Maarten van Elsas
mvanelsas
4176898

Leon Helsloot
lhelsloot
4235991

May 23, 2014

Contents

1 Introduction

This document describes the planning for the *Building Blocks* game. Included in the planning are a roadmap, describing the schedule for major releases, and a product backlog containing user stories.

The project contains a total of eight sprints, with each sprint starting on Friday and ending on Thursday.

2 Product

2.1 High-level product backlog

The product backlog contains the following main components:

- have a client that can show the structure;
- have a server that can control/check the structure;
- have the server and client be able to communicate;
- have the half block view/combining available and communicate it with the server;
- present everything in a visually attractive way;
- simplify all components so that the controls can be ”figured out” simply by playing.

2.2 Roadmap

Our roadmap will have these releases in the mentioned weeks, with the mentioned goals.

- Alpha, week 3. This should be a first playable version, it should have the basic game in place (although all elements) furthermore it should allow multiple players to play in the same map.
- Beta, week 6. In this version most elements of the gameplay should be implemented, and focus should be shifted to issue-resolving/polishing.
- Final, week 8. This is the fully playable game, it should include all game elements and be fully playable. Furthermore it should be tested and functioning.

3 Product backlog

3.1 User stories of features

- As a player I want to be able to remove a block so I have more control in the construction.

  Scenario 1 : A block placed is placed at a wrong location. This would lead to that the structure cannot be completed. So the option to remove a block would be ideal to continue with the task at hand.

  Scenario 2 : Someone else placed a wrong block at a wrong location and I notice this. So would be constructive if I can remove this block to bring the team closer to their goal of bulding the goal structure.

- As a player I want to know when a game has ended so I can be aware of the game state.

  Scenario 1 : The time ran out given that the time limit has been reached. The game should inform the user that the game has ended.

  Scenario 2 : My team placed the last block to finish the goal structure and win the game. The game then should inform all the teams that one team has finished building the goal structure.

  Scenario 3 : If one the team has placed the last block of their structure, the other team should be informed and the game will be terminated.

- As a player I want to be able to see the global structure so I can visualize the goal structure

  Scenario 1 : When we are building a structure we should be able to view the goal structure to remember what we are building.

3.2 User stories of technical improvements

- As a player, my view of the structure should be stable, or in other words the amount of jitter should be minimized.

3.3 User stories of know-how aquisition

3.3.1 Joining a game

- As a spectator, I want to know that I can join a game.
- As a spectator, I want to know how to join a game.

3.3.2 Viewing game progress

- As a player, I want to know that I can see a structure when aiming my camera at a marker.
- As a player, I want to be able to see the goal structure, so that I know what to build.
- As a player, I want to be able to see the current team structure, so that I know where to place a block.
- As a player, I want to be able to see the current structure of the opposing team.
- As a spectator, I want to be able to see the goal structure.
- As a spectator, I want to be able to see the structures of all teams, so that I can enjoy watching the game.

3.3.3 Viewing own game state

- As a player, I want to know whether I have a full block or a half block.
- As a player, I want to know whether I am building or deleting.
- As a player, I want to be notified when joining two half blocks succeeds.
- As a player, I want to know which team I am in.

3.3.4 Playing the game

- As a player, I want to know what the goal of the game is.
- As a player, I want to know what to do with a half block.
- As a player, I want to know what to do with a full block.

- As a player, I want to know how to join two half blocks.
- As a player, I want to know how to place a block.
- As a player, I want to know that destroying a block may lead to a penalty.
- As a player, I want to know at which marker to build the game.

3.4 Initial release plan

The alpha release (week 3) has the following as the Minimum Releasable Features (MRFs):

- Players can place complete blocks in the structure.
- Players can remove misplaced blocks from the server.
- Each player interacts with the same structure which is managed by the server.

The beta release (week 6) has the following MRFs:

- Players can join the game.
- Players get (partial) blocks from the server.
- Players can combine blocks creating larger blocks in a mix of the colours.
- Server keeps track of block placement and colour and recognizes when goal structure is finished.
- Server checks player moves for validity.

The final release (week 8) has the following MRFs:

- Players can join a team
- Once the goal structure is finished, the server points out the winning team.

4 Definition of done

In order to ensure there are no different definitions of done, we will specify the one we will use in this section. It contains our definition of done for a feature, a sprint and a release.

Every new feature is developed in a separate branch. It's done once it's functional, properly formatted and well tested. Proper testing consists of automated tests where possible. If these are impossible or infeasible then manual tests will have to be performed instead. The feature can only be considered functional if all tests are passed. Once the feature branch is considered done by the developer, a pull request will be submitted. The lead programmer will then evaluate whether the feature is done by the before-mentioned criteria. If so, the branch will be merged into the master.

Our sprints have a set time of a week. This means the sprint is finished at that time. Any unfinished features will be returned to the product backlog and most likely added to the sprint backlog of the next sprint.

A release is done once all features in the release are done and user tests have been passed. User tests are led and evaluated by the QA tester. If the user test is passed, the release is done.

5 Glossary

Client The *Building Blocks* smartphone app to be used by players of the game.

Issue-resolvement The act of removing unwanted, but not breaking elements from the game (for example, an element not disappearing when it should.)

Polishing The act of make the game more visually attractive/easier to use.

Server A computer to which every client connects, responsible for regulating the game.

Spectator A person not currently playing the game, but watching the game being played.

Jitter Quick jumpy movements of the structure.