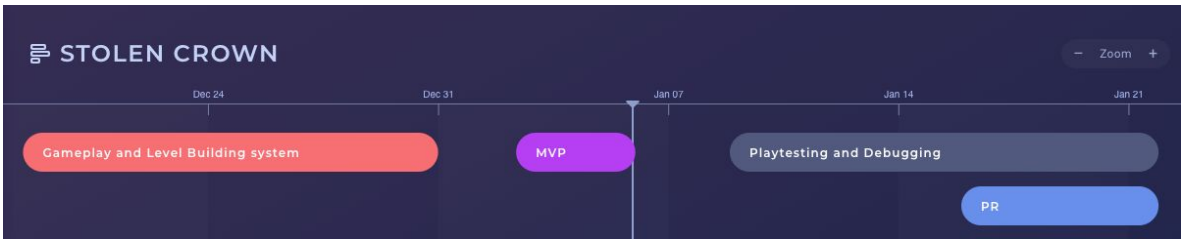
Stolen Crown

Game Design Document

by
Shahin Aliyev
Hasan Alakbar

TABLE OF CONTENTS

SCHEDULE & MILESTONES



MILESTONE 1 - Gameplay and Level Building system

- *Initial gameplay logic has been implemented and tested. Time revival functionality has been built.*
- *A level builder (see Supporting Materials) has been designed and coded, that allows a level designer to build levels without touching a line of code.*

MILESTONE 2 - Functional MVP on PC

- *Aforementioned resources have been used to build sample levels in the game. Sample game art have been introduced to make it all more interesting. Some playtesting has been done and found bugs have been resolved.*

MILESTONE 3 - Extensive Playtesting and Debugging

- *In order to properly identify and take care of bugs, extensive playtesting campaign will be arranged. The game will be tested on a range of devices to show all kinds of issues that might come up.*

MILESTONE 4 - Soft-launch

- *The game will be initially released in the limited area, which will allow to further reduce the friction when larger crowd gets access to the game, as currently we don't have resources to deal with large amount of support requests.*

GENERAL
OVERVIEW

THEME AND GENRE

- *"Stolen Crown" is a platformer, that will showcase both levelled and also infinite running capability.*
- *The game is themed around medieval Europe and Asia. The crown has been stolen from its rightful owner and tries to find its way home.*

GAMEPLAY MECHANICS (BRIEF)

- *User plays for the crown. In the future it is planned to allow selection of different crowns, each having a different model and animations.*
- *The game has simple mechanics that's mostly limited to **jumping up,***

***left and right**, to **gather collectibles** and avoid obstacles on the way to freedom.*

TARGETED PLATFORMS

- *Initial plan is to release the game on the **Steam network**. "Stolen Crown" has interesting level building system that can be used by players on **Steam Workshop** to generate interesting compositions. These can be further used by us to provide users on all platforms (most importantly mobile marketplaces) with user generated content.*
- ***Eventually**, game will be targeting **mobile platforms** (Android, Tizen, iOS) and most importantly iOS marketplace, mainly due to monetization strategy selected.*

MONETIZATION MODEL (BRIEF)

Game will make use of 2 monetization strategies:

- ***Paid game**: Game will be available at $0.99 price on **Steam**, **AppStore** and **PlayStore**. Games that have unique style and levelled structure have sufficiently high chance of getting featured on AppStore, which will play to the game's benefit.*
- ***In-game purchases**: Once purchased, all levels can be unlocked and played through. If a player wants to jump to a further level without first unlocking it, (s)he can pay $1.99 to unlock all levels in one Chapter.*
- *(Potentially) In case we go forward with releasing the game for Tizen, we are considering to have a **Free-to-play** mode, monetized only with ads. This is a low competition market, where users don't really have much capability to pay for what they use.*

CORE GAMEPLAY

CORE GAMEPLAY MECHANICS - 1

- ***Jumping up**: There are a number of obstacles in the game that requires a player to jump over/onto. As the height of the obstacles differ, player needs to get a good grasp of how much to jump to achieve required height, as mismanaging jump might be lethal.*

CORE GAMEPLAY MECHANICS - 2

- ***Jumping left and right**: All rooms have 4 lanes - 2 main ones and 2 short ones, where you need to get out fast, not to hit a column. Player needs to correctly time jumps as mistiming them, might end up in hitting an*

object mis-flight and lead to failure.

CORE GAMEPLAY MECHANICS - 3

- ***Collectibles and progression**: Levels include collectibles. Each level has a limited number of them and recorded collection count for each level is the maximum number a user has ever been able to gather. Collectibles limit progression in the game and in order to progress further, user needs to have a level of success in each level (gather 70% of available collectibles) or replay levels.*

CORE GAMEPLAY MECHANICS - 4

- ***Rewind time (planned)**: Some levels will include sands of time, which will help you rewind time to a previous state in case of failure, letting you replay from that moment on.*

STORY & GAMEPLAY

STORY - BRIEF

- *Everything starts in a land where everybody lives in prosperity. A thief from lands far-far away decides to steal some of the wealth, now that there is so much of it.*
- *While stealing valuables from the treasury, the burglar takes hold of the crown as well, which is can be sold for quite a price.*
- *When crown wakes up to find itself in unknown castle, it decides to try and escape from captivity and return to its rightful owner.*

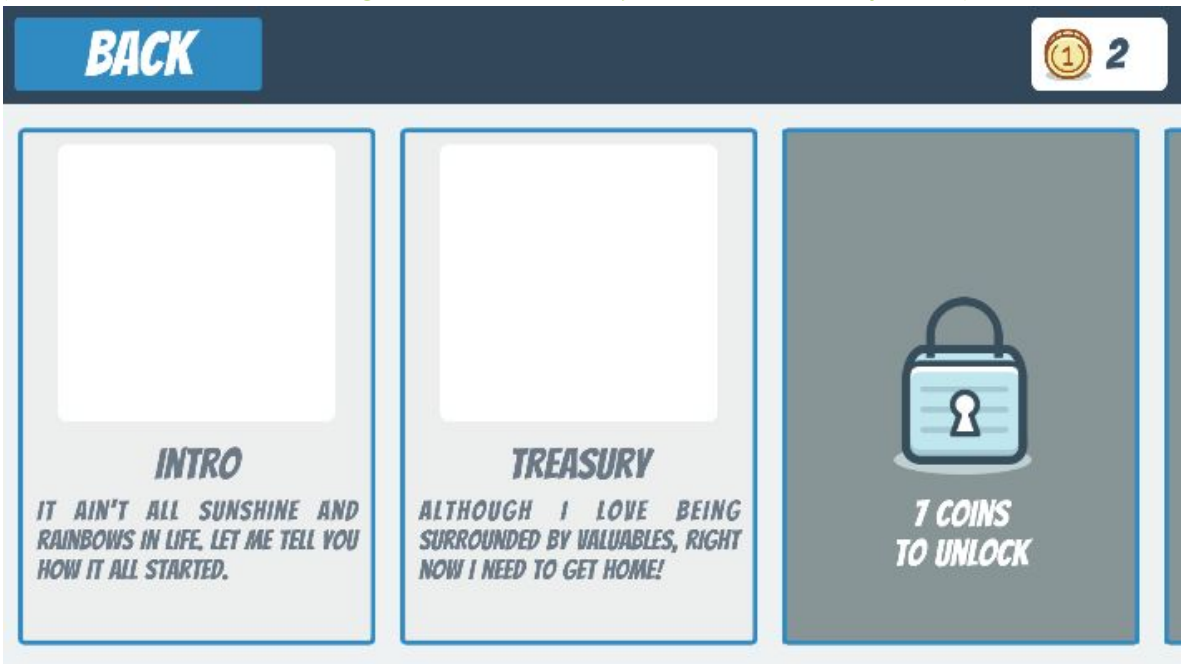
GAMEPLAY - BRIEF

- *Journey starts in the castle, where crown is kept captive. The crown*

needs to travel through a number of levels and chapters to get to its owner.

GAMEPLAY - DETAILED

- *The game is divided into chapters, which are further divided into levels.*
- *Each chapter takes place in a different setting (inside castle, outside castle, outskirts, city).*
- *Each chapter is divided into a number of levels that challenge the player for completion. By successfully gathering valuables on the way to finish, the player gets a chance to score sufficiently to advance to next level (which is locked according to the number of total valuables picked).*



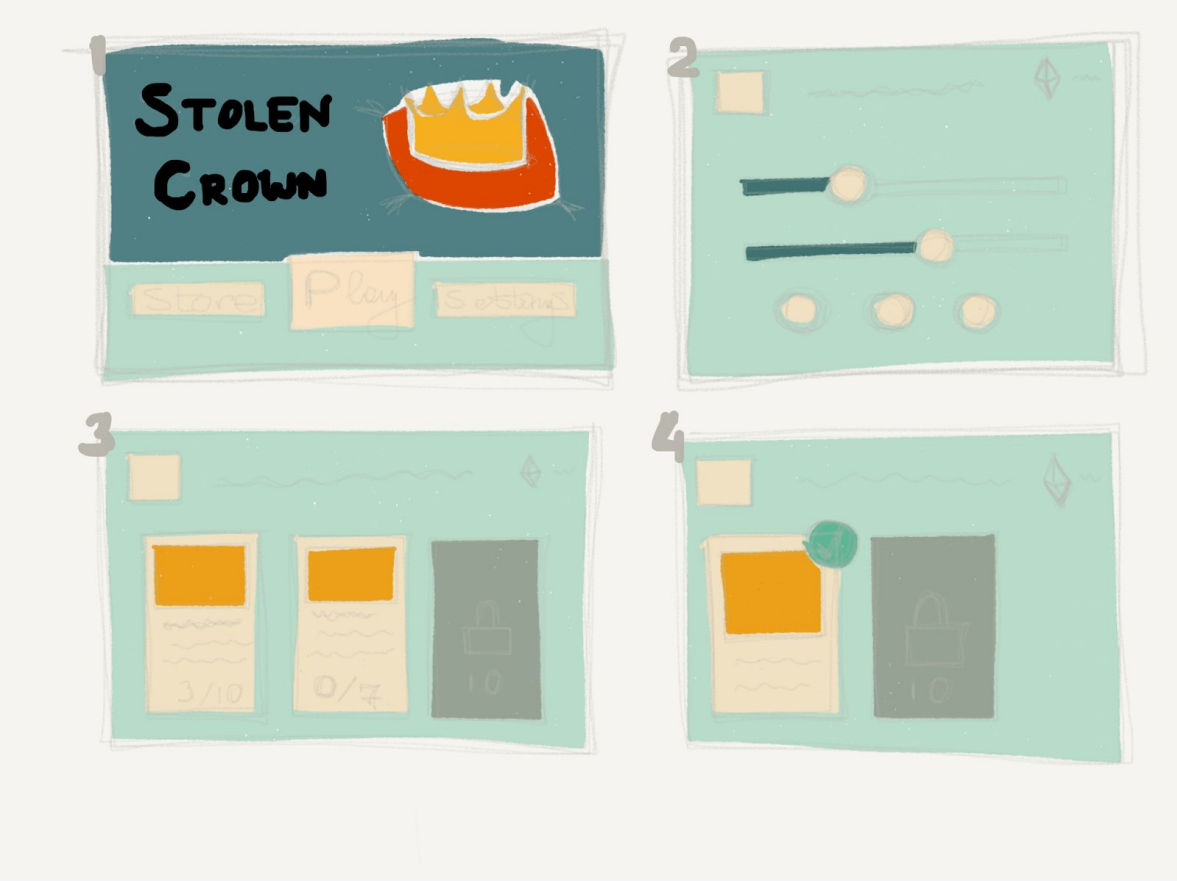
- *Game structure (chapters and levels) is as follows:*
  a. ***In the captivity (inside the castle)***
     MVP covers only this chapter
     i. *Intro*
     ii. *Treasury*
     iii. *Almost there*
     iv. *Prison break*
     v. *The big resque*

  b. ***Out for some fresh air (outside the castle)***
     To be further developed…

  c. ***No forest can stop me (in the forest)***
     To be further developed…

d. ***The big city, here I come! (city of the target castle we are going to)***
To be further developed...

ASSETS & EQUIPMENT



2D ASSETS

- *2 types of 2D assets are to be used in the game:*
    a. *UI elements:*
    We are using a couple of resources for this:
        i. ***Envato Elements*** *website (subscription) - for icons*
        ii. ***UI Builder Asset*** *we bought for general use, which allows*

to very fastly prototype different UI arrangements

b. *In game art:*
*Due to low budget and time restrictions, it was decided to go forward with hand-drawn art for intro cutscenes. These is to be made by us in **Photoshop**, using **Surface Pro**.*

3D ASSETS

- *3D models are to be prepared and adapted by the team, using **3ds Max**.*
- *These cover a range of different 3D models you would see in a castle. Walls, doors, windows, grounds, etc. are to be adapted to medieval style.*
- *Some models form the core of the game, so need to have much higher detail (Soldier, Curtain, etc).*

- *10 Skyboxes Pack : Day - Night is to be used for skybox creation.*

- *Testing has shown that a considerable amount of optimization needs to be done in order to make the game portable to mobile platforms (as current build with assets for a single theme is well beyond 100 MB range).*

SOUND ASSETS

Sound assets have been picked to be in alignment with the theme of the game (medieval castle setup):

- *Casual Game SFX (free) has been used for in-game sound effects.*
- *Dungeon Audio Kit has been purchased and 2 tracks from it have been used as gameplay and menu songs.*
- *Future plan is to work with a professional sound designer for final product. It will ensure much higher quality of the product.*

CODE

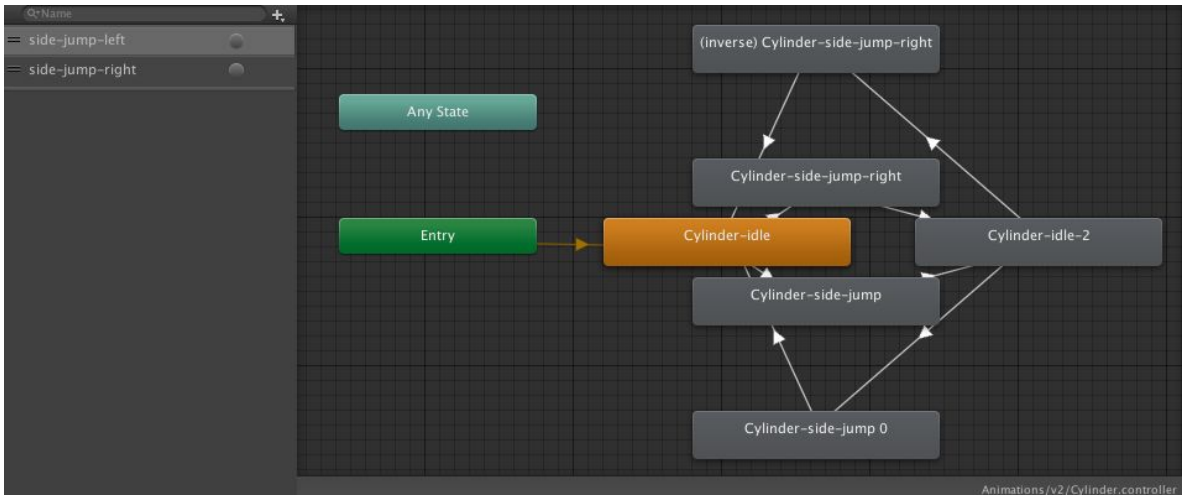
- ***Language**: All code is to be written with **C# using Unity libraries**.*
- ***IDE: JetBrains Rider** is to be the IDE of choice, as it aids production.*
- ***Commenting**: Commenting is to be avoided, as it creates additional load to be maintained. Instead of that, explanatory method and property names are to be used.*
- ***Style**: Code design needs to comply with "**All-In-One Code Framework Coding Standards by Dan Ruder, Jialiang Ge**".*

- *Cinemachine (free) is to be used for camera controls.*
- *Simple FX - Cartoon Particles (free) is to be used for some of the particle effects in the game.*

- *A special "level designer friendly" system (see Supporting Materials) has been built to aid production of the game. Each level consists of sections/rooms (see Supporting Materials), divided into subsections, all of which can be edited using the Inspector inside the Unity.*
- *Random level builder (for future infinite runner mode) has been implemented, which randomly fills rooms with geometry, decorations and obstacles.*
- *Time rewind functionality has been implemented, but needs to be ported to the final game. Check points need to be built into the level structure to mark exact points of return. Otherwise rewind might end up in a weird case (e.g. just about to land on obstacle).*

ANIMATION

- *All animations are to be done inside Unity, again by the core team. Animation panel is to be used for that purpose and an Animator is to be prepared to aid transitions between states.*
- *There aren't very many animations in the game. These are limited to jumps and rotation of the crown. Crown has 2 sides, so there will be 2 idle positions. Additionally, there will be 2 animations of jumping left and right, which can be easily reversed for return animations.*

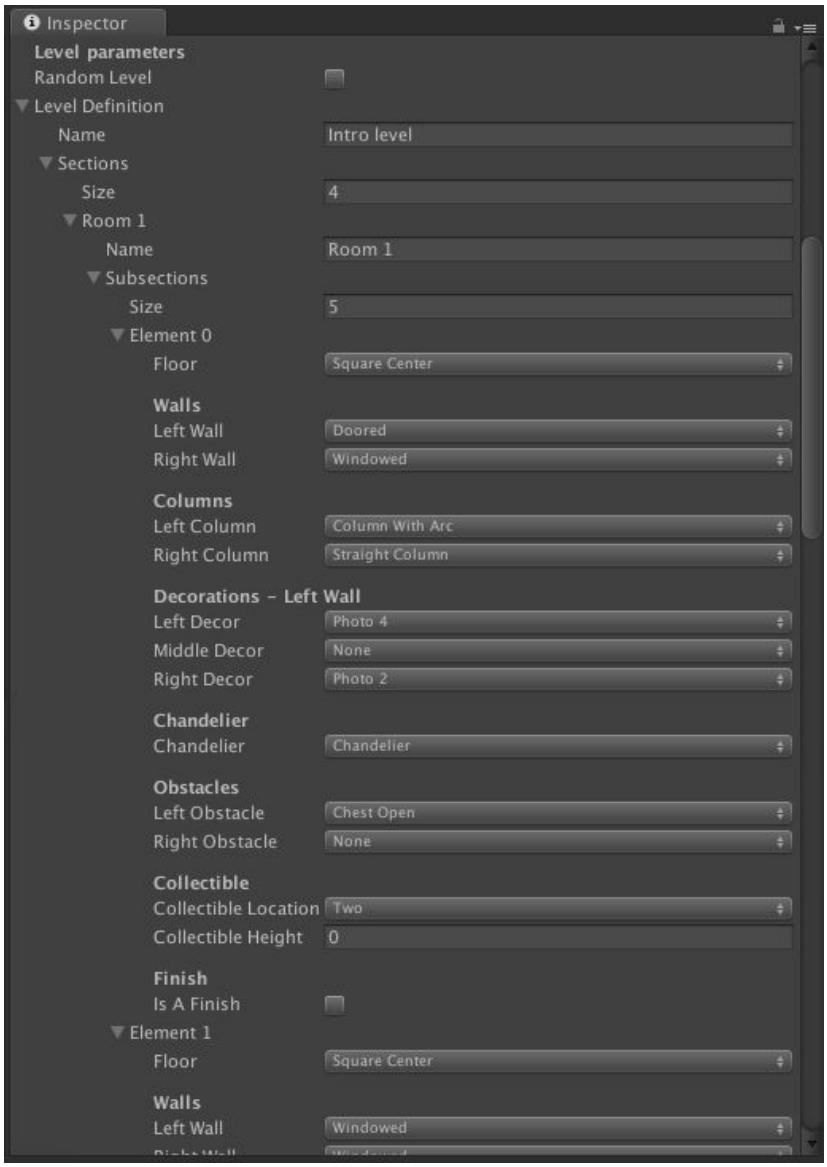


EQUIPMENTS

- ***iPad mini** and **Surface Pro** are being used for image production.*
- *A **Mac-mini** and **PC** is to be used to allow proper building and testing of the game.*
- ***Unity engine** is being used to accelerate production and portativeness.*
- ***Photoshop** is to be used for graphics manipulations.*
- ***3ds max** is to be used for 3D modelling.*

SUPPORTING MATERIALS

LEVEL BUILDER



LEVEL STRUCTURE