CubeCraft: Team Report

Daniel Cullen, Sean Dillon, Vanya Eccles,
Samir Kumar and Lan Liu

May 8, 2017

1 ABSTRACT

Here we present CubeCraft, a serious game with the purpose of testing and exercising spatial cognition. The game was made in the span of eight weeks by a team of five students, using the Scrum methodology to manage the development procedure. This document details the motivations and development decisions during the production of CubeCraft, as well as the technical and architectural design and management methods employed.

2 INTRODUCTION

2.1 MOTIVATION

What is it about play that makes it so enjoyable? The question has spawned fields of study from disciplines as diverse as child psychology to military research, with the consensus that playfulness is strongly related to cognitive development [Whitebread *et al.*, 2014]. By virtue of simulating phenomena that exercises our mental faculties, we may enjoy play as a means to train our minds to cope with challenging circumstances [Spinka *et al.*, 2001]. With the emergence of video games, researchers have seized on this powerful new medium both to understand the nature of its popularity and how it could be tapped for addressing meaningful questions

about the brain. This has spawned interest in the application of 'serious games' designed for a purpose other than pure entertainment.

Using video games to understand, measure and improve cognition goes back to the 1980s, and they have increasingly been used by psychologists to aid understanding of skill acquisition, memory and attention amongst other areas. Games such as 'Space Fortress' have been specifically designed to assess cognition, whereas others have been assessed for their ability as a psychological stimulus, that is capable of eliciting a sensory or behavioural response. [Boot, 2015] Over 30 years of research into video games and cognition [Latham *et al.*, 2013] have studied the potential role that games play in improving functioning in hand-eye coordination, reaction time and visuo-spatial attention. Many scientists in the field have however been highly critical of the resultant 'brain training' industry, stating that current findings do not provide a sound basis to the industry's claims [Stanford Letter, 2014]

The development and appeal of three-dimensional scenes rendered in real time has brought much interest in how these games interact with our spatial cognition [Spence and Feng, 2010]. It is well understood that exposing animals to stimulating environments can improve the function of the hippocampus, a component of the brain that plays an important role in spatial memory and navigation, as well as improving performance on hippocampally-mediated memory tasks. Researchers at UC Irvine demonstrated that training naive video gamers in a rich 3D, but not 2D, video game led to significant improvement of hippocampus-associated cognition across several behavioural measures [Clemenson and Stark, 2015]. Similarly video games training may also lead to improvements in mental rotation, with games that exercise this in humans leading to increased performance in areas such as learning spatial anatomy [Lapaine *et al.*, 2014].

Our interest into the application of video games as tools for cognitive research and as tools for providing meaningful stimuli led us to the development of the game 'CubeCraft', built in the Unity3D game engine with the C# programming language. CubeCraft was inspired by work done with games in cognition research, and was built to provide a challenge in the areas of spatial cognition and mental rotation. It is however important to note as a disclaimer that CubeCraft was not directly built for specific research questions, and we certainly have no evidence that it is suitable as a therapeutic or training tool. We do however hope that it serves as a proof-of-concept for potential research tools, in addition to being a fun, intuitive and challenging game.

2.2 CubeCraft in Brief

2.2.1 The Game in a Nutshell

CubeCraft consists of the user being shown a partially completed cube, built from smaller unit cubes, that they can rotate and examine as seen below.

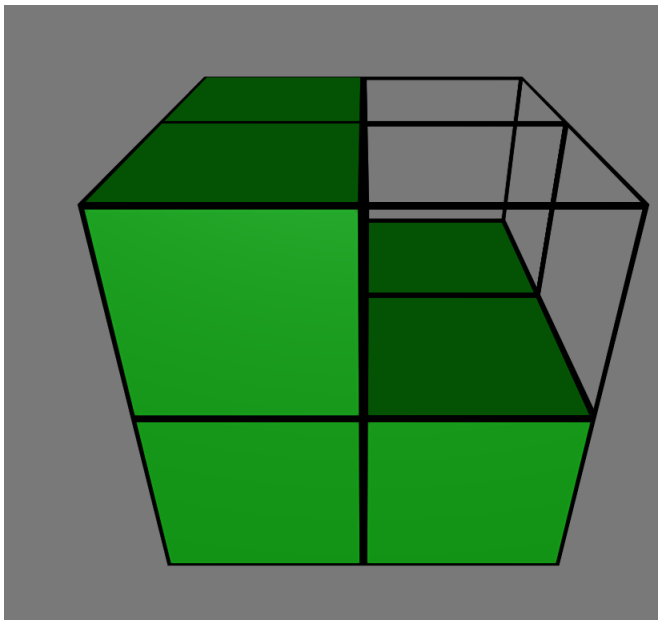


Figure 2.1: The Uncompleted 'Puzzle Shape'

The user has a defined amount of time to imagine the necessary configuration of cubes that would complete the puzzle shape, and memorise it. Then they are brought to another screen where they are given a blank grid and the ability to place new cubes. The user must then try to remember the shape of the imagined solution and build it. They are assessed on the correctness of their solution, in addition to the time taken to complete it

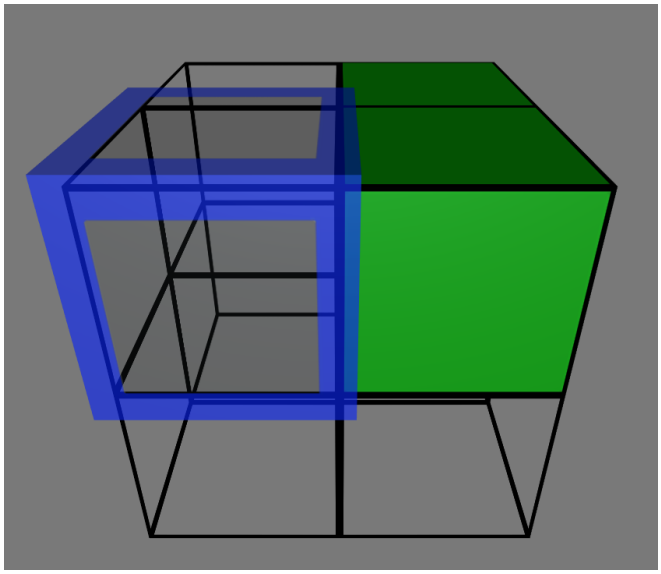


Figure 2.2: The Completed Solution to the earlier 'Puzzle Shape'

2.2.2 The Project in a Nutshell

We built the game to be accessible and engaging, and concentrated on features that would support this. To that end we concentrated on play testing and elicitation of user feedback through surveys. As the project was reasonably large in scope we put great effort into project management through implementing the Agile/Scrum framework, in addition to constructing and refactoring the codebase architecture so that our code was flexible and non-fragile. The rest of this report will provide a summary of the project, from the design of game mechanics through back end details and code architecture to project management.

A demonstrative video is available at the following YouTube link:

`https://www.youtube.com/watch?v=XXX1mEjbAYw&feature=youtu.be`

And the game can be played online via the following link (WebGL):

`http://162.243.28.43/CubeCraft.html`

3 Project Overview

3.1 Evolution of the Core Idea and User Feedback

At the beginning of the project, the aspect of the game that the team most concerned themselves with was the cube-building function of the game. In theory, the cube-building aspect of the game would be analogous with how someone would

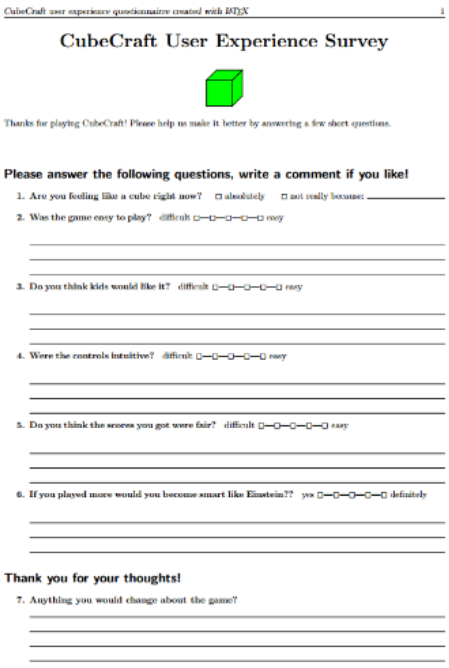
build a shape out of identical cubes in real life. To try capture how this is achieved, small cubes (pictured below) were purchased with which to practice building solutions or problems and also to have a physical representation when one of the team members was trying to describe a certain shape or object. Originally it was decided that having an 'Add' function and a 'Remove' function for placing and deleting cubes onto the grid would be sufficient, but on inspection of how people were building different shapes using the cubes in real life it was decided to add a 'Grab' function. This function could be used to pick up an already placed cube and move it to another location on the grid without deleting it. Having this functionality made the controls far more intuitive and thus easier to use.

Figure 3.1: Toy blocks used during project development

When the problems began to be constructed originally only problems of 2 by 2 by 2 scale were created. This was done to adhere to the team strategy of development which was to have a usable product early on and then build on it and make it better and more complex. Once any technical issues with this size problems were resolved and there was a working playable level, development of the larger scale problems began. Some difficulties appeared in the 3 by 3 by 3 problems which were not apparent in the smaller problems which needed addressing. The most pertinent of these issues was a visibility issue. Because of the increased number of objects in the scene, having a situation where the cube selected by the user would be impossible to see. Two solutions were decided to counteract this, firstly it was decided to include a rotation function whereby the user could rotate the grid around its central vertical axis. It was discussed whether to also include horizontal rotation but it was decided that having one way of rotating the grid was sufficient. Secondly, it

was decided to add a visibility layering system to the game. What this system did was if the user had a certain position selected, then any cubes already placed closer to the users viewpoint or above the selected position would become transparent. This allowed the user to see through cubes when operating on positions that were previously invisible to the user.

After the visibility issues were solved, the team began to play the game and brainstorm on what could make the game feel better or more intuitive. After one of these brainstorm sessions it was decided to add sounds and background music to the game. In particular, clicking and popping noises when cubes are added to or removed from the grid added another dimension to the game in terms of feel. At this stage the game was tested by users, consisting of other Masters students in TCD, in addition to friends and family. They would play through the six available levels and then fill out a questionnaire, this feedback would then be used to improve existing features and add in new ones requested by the users. After this first wave of questionnaires it was decided to improve the tutorial and the directions given to the user, also there was still reported problems with the visibility so to solve this issue the visibility layering system changed. The updated system has different levels of transparency according to how far back into the grid the selected position is, resulting in a much clearer grid. Once these changes were made a second round of market research began. This involved showing the improved game to the users who had filled out the questionnaire already, and collecting data from new users who had never seen the game before. (See questionnaires and data from feedback below)



CubeCraft user experience questionnaire created with LaTeX 1

CubeCraft User Experience Survey

Thanks for playing CubeCraft! Please help us make it better by answering a few short questions.

Please answer the following questions, write a comment if you like!

1. Are you feeling like a cube right now? ☐ absolutely ☐ not really because: _____
2. Was the game easy to play? difficult ☐—☐—☐—☐—☐ easy
3. Do you think kids would like it? difficult ☐—☐—☐—☐—☐ easy
4. Were the controls intuitive? difficult ☐—☐—☐—☐—☐ easy
5. Do you think the scores you got were fair? difficult ☐—☐—☐—☐—☐ easy
6. If you played more would you become smart like Einstein?? yes ☐—☐—☐—☐—☐ definitely

Thank you for your thoughts!

7. Anything you would change about the game?

Figure 3.2: First User Survey

CubeCraft user experience questionnaire created with LaTeX 1

CubeCraft 2nd User Experience Survey

Thanks for playing CubeCraft! Please help us make it better by answering a few short questions.

Thanks for playing CubeCraft again, we hope you like it even more now!

1. Did you notice any changes since you last played? ☐ sure did ☐ not really
2. Do you think the tutorial is helpful? very helpful ☐—☐—☐—☐—☐ not at all
3. Do you think the buttons had clear functions? clear ☐—☐—☐—☐—☐ vague
4. Were the controls easy to use this time? difficult ☐—☐—☐—☐—☐ easy
5. Now that you've played twice do you think this is your favourite game? yes ☐—☐ also yes

Thank you for your thoughts!

6. Anything you would change about the game?

Figure 3.3: Second User Survey

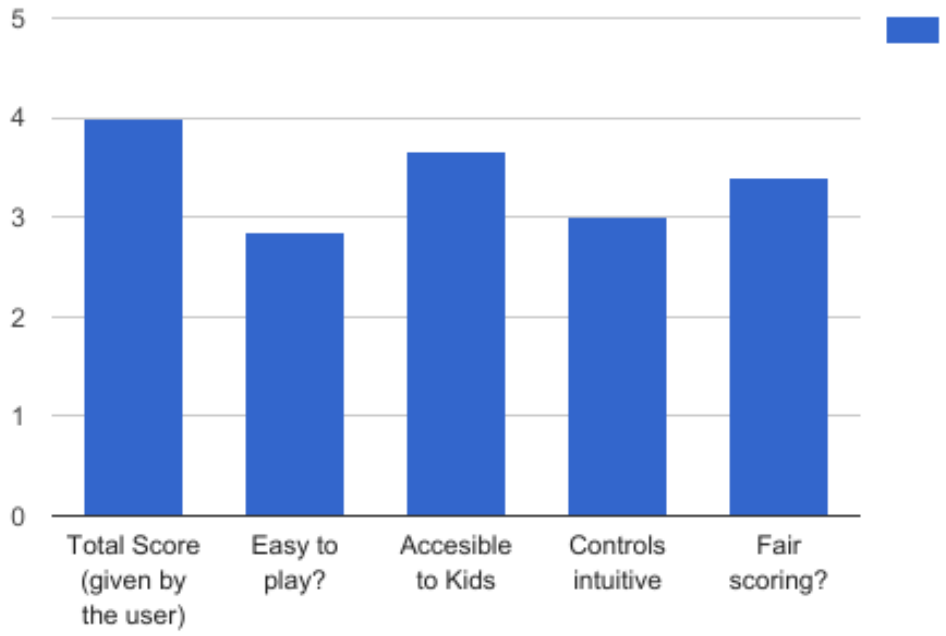


Figure 3.4: Second User Survey

After these two rounds of research and using the feedback provided the delivered product was found to be fun, easy to use and bug-free. The last step in the project was to create a build of the game for mobile, and a build of the game for use on a tablet device. Playing the game on a tablet device was very popular among users due to the intuitive controls when using a touch screen, and would be our choice for release versions of the game.

4 Game Mechanics

4.1 Navigation and Actions

The first challenge for the team was to come up with a decision about the navigation and the actions of the user inside the grid. After a lot of discussion and thought, the following rules were introduced:

- The location of the user is indicated by a special cube, dubbed the 'Ghost Cube' that looks different than the rest and does not block the visibility of the user.
- The possible actions of the user depend on the location and previous actions of the 'Ghost Cube':

  - If there is another cube at the same position, the user can 'Delete' the cube.
  - If there is another cube at the same position, the user can 'Grab' the cube. This means that wherever the 'Ghost Cube' moves, the grabbed cube will follow. If the 'Ghost Cube' has grabbed another cube, it cannot go to locations that have other cubes.
  - If there is no other cube at the same position and 'Ghost Cube' is not grabbing another cube, the user can 'Add' a new cube.
  - If there is no other cube at the same position and the 'Ghost Cube' is grabbing another cube, they can 'Release' the cube at that position.

- The freedom of movement is constrained only by the aforementioned rules.

This was the initial specification of the rules and the team decided to improve it according to the results after the completion of each of the sprint cycles. Thanks to the dynamic User Interface, the scheme felt easy to use and remained unchanged.

4.2 Rotation and Visibility System

For the improvement of visibility, a rotation functionality was added. The rotation is limited to the vertical axis and the user can observe their solution or the initial problem from 4 orthogonal angles. The reasoning behind this decision was that the navigation system as well as the visibility system assume an axis base and the movements can be only orthogonal to that base. By allowing a 360 rotational freedom, the direction of movement would be ambiguous and unnatural to the user. Finally, animation was added to show the transition between points of view. This is necessary due to the nature of the game; the user can be tricked about the angle of view since the grid may look very similar from orthogonal positions.

The visibility was further improved by the 'Visibility System'. The main idea is that the cubes between the point of view and the 'Ghost Cube' become transparent. The first approach was to shoot a ray between the camera and the 'Ghost Cube' and change the transparency of the hit cubes. The results showed that this was not a good solution because the user needed a clear view of the whole layer that the 'Ghost Cube' was located at. Based on this observation, the team developed a method that given the point of view and the location of the 'Ghost Cube', would change the transparency of the cubes appropriately. During the feedback phase, testers complained about the visibility. The team improved the system even further by applying multiple levels of transparency where the opacity level follows a sublinear curve to the distance. Due to the additive nature of transparency, it was very hard to find a good trade-off between the minimum opacity of the close cubes

and the visibility of the far cubes. Thanks to the fixed dimensions of the grid, these values were found, but the effectiveness of the system stops after a 3 by 3 grid.

Another observation by the team, during the improvement of the visibility system, was that the 'Ghost Cube' should be clearly visible, but the cube that it might have been standing on top, should also be clear. This was a contradicting request since by adding opacity to the 'Ghost Cube' would make it stand out more but hide the underlying contents and reducing the opacity to reveal them would make the 'Ghost Cube' less visible. The first idea that was applied was to curve holes on the side of the 'Ghost Cube'. In addition to this, different depth layers were used: the 'Ghost Cube' was rendered on top of everything else on the scene. Due to the transparency of 'Ghost Cube', the result looks natural.

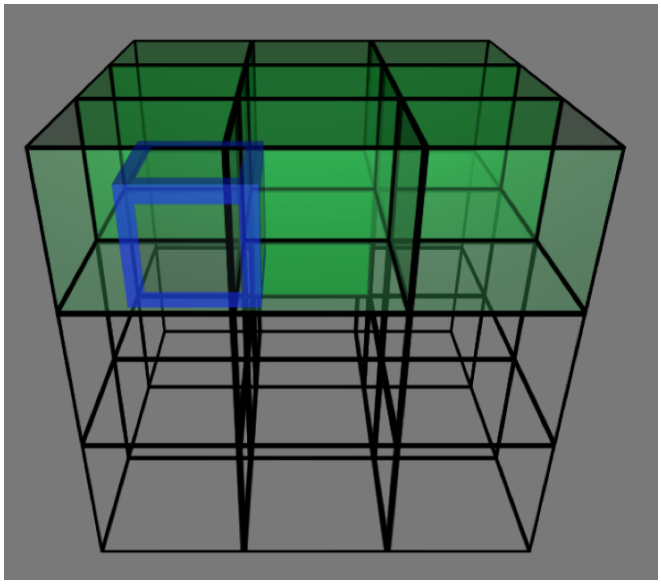


Figure 4.1: Visibility approach in CubeCraft

4.3 Solution Checking

As soon as the camera rotation was agreed as part of the game, the team realised that there might be a complication with regards to the correct solution. Since the cube has isotropic dimensionality, a correct solution can be formed in multiple ways. The solution checking system checks the solution against 8 symmetric configurations and for each of them, it provides the appropriate metrics for evaluating the solution. The configuration that is used is the one with the best results.

4.4 Levels and Player Evaluation

An important aspect of the application was the difficulty of each level and the adaption of the difficulty to the abilities of the player. The team agreed that implement-

ing a procedural system for level generation was not trivial since it would involve probabilistic theory and automatic refinement in a small discretized space. Instead, it was decided to create a set of difficulty levels manually and adjust them according to the user survey. For this purpose, an in-game tool was developed to save a constructed solution as a JSON file. This concept allowed the team to modify the JSON files that stored the levels, inside the game. The JSON format was ultimately used for loading anything that was required from the disk.

The user's performance had to be measured in detail so an appropriate feedback could be returned. The first measurement is the time it takes the player to submit his solution. The second is the number of false positives, false negatives, true positives and true negatives. With this input, the team designed and tested some formulas. The best formula turned out to be:

$$percent_{correct} = \frac{TP}{TP + FP + TN} \tag{4.1}$$

and the user is evaluated in a three stars system with the following rules (order dependent):

1. If the percentage is less than 40%, then the user fails with 0 stars.
2. If the percentage is less than 100%, the user wins with 1 star.
3. If the time submission time is less than a time threshold, the user wins with 3 stars.
4. If the time submission is over the time threshold, the user wins with 2 stars.

In addition to the stars, the team decided to provide some verbal feedback to the user so they would know if they were slow or they did a lot of mistakes. The above measurements allowed them to do so.

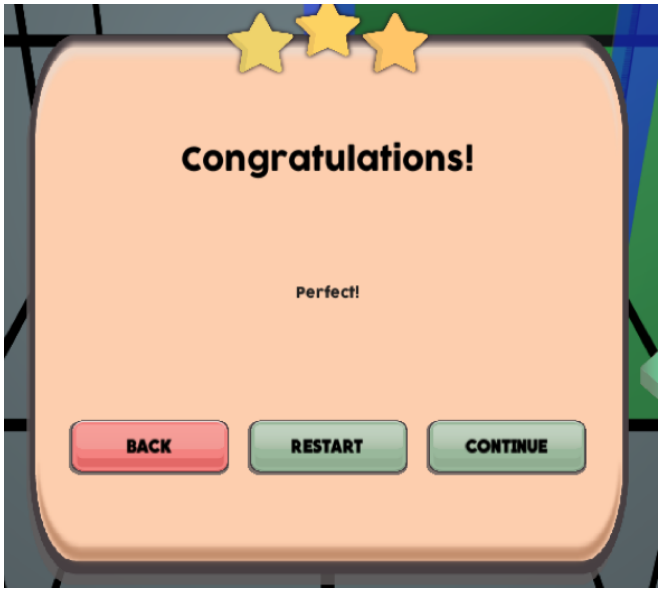


Figure 4.2: Player Feedback: 3 Stars

Figure 4.3: Player Feedback: 1 Star

5 Design

5.1 Overview of Architecture

The whole project can be concluded as layered architecture, which contains three parts: Presentation & Input, Business, Data.

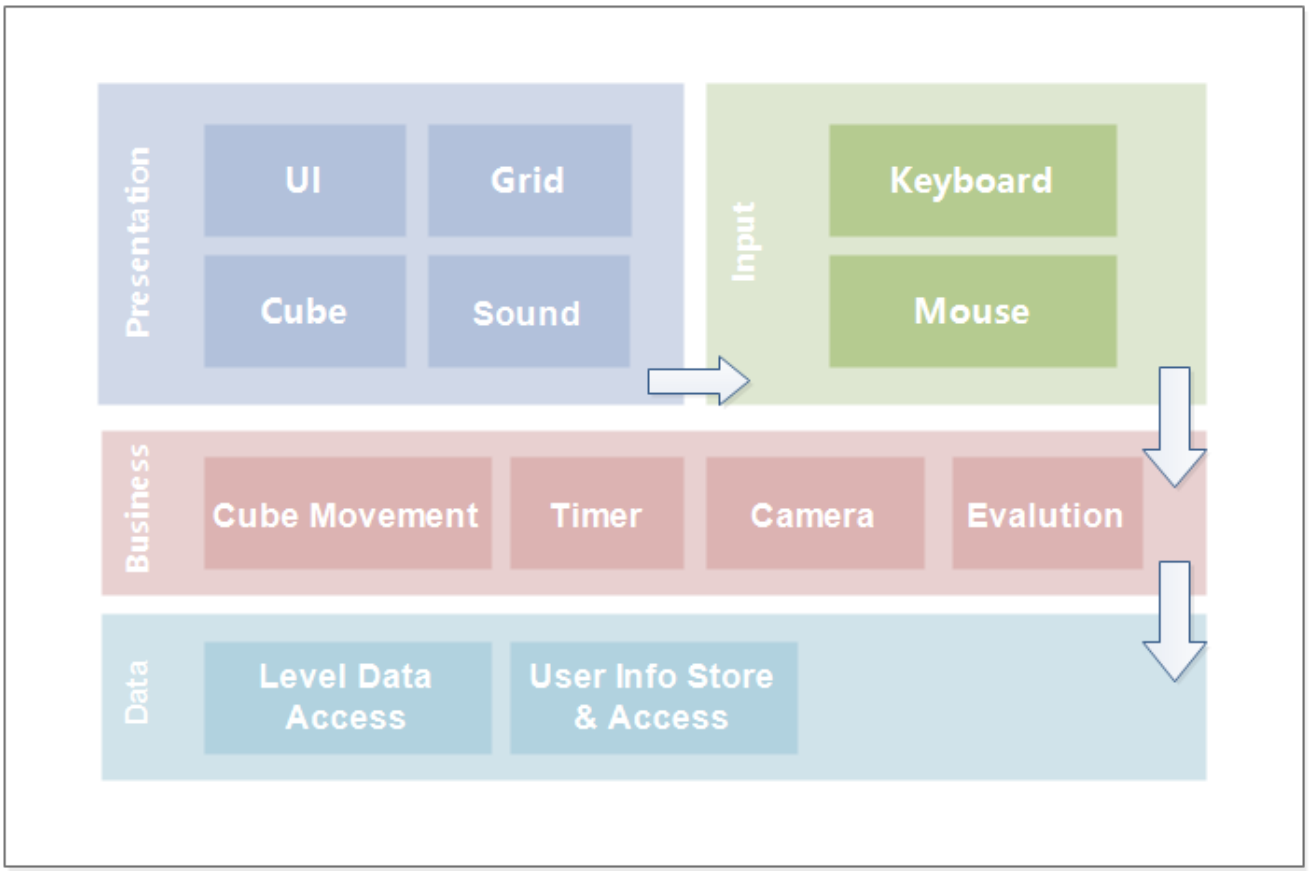


Figure 5.1: Layered Architecture

The presentation & Input layers provide all the audiovisual elements and user interfaces. UI can be divided in to three parts according to their usage; 'setting', 'in-game' and 'instruction'. Different types of manipulation are implemented in the Input part. For example, keyboard controls for PC and the swipe operation for tablet. Being user-friendly and adaptive to different screen size were taken into consideration in our design. With the development and refactor, encapsulation also improved adaptation to the Unity3d platform.

Business Layer is responsible to process the logic in game. Cube Movement defines the main operations that player can conduct on the cube, like Move, Add and Delete. Timer provides functions related to time, like the countdown in the game. Camera allows player to rotate the whole scene and based on the direction and position of ghost cube to render with different strategies. Evaluation will give player a grade in stars based on the time used and accuracy rate.

The Data Layer includes all functionality related to retrieving or storing data on the disk. Level Data Access saves and loads the required information to render a problem and check a provided solution against it. The User Info Store & Access acquires and updates the progress of the user in terms of what levels he has completed and what is the best score he ever achieved for each of them.

5.2 Implementation

5.2.1 Grid and Cube Rendering

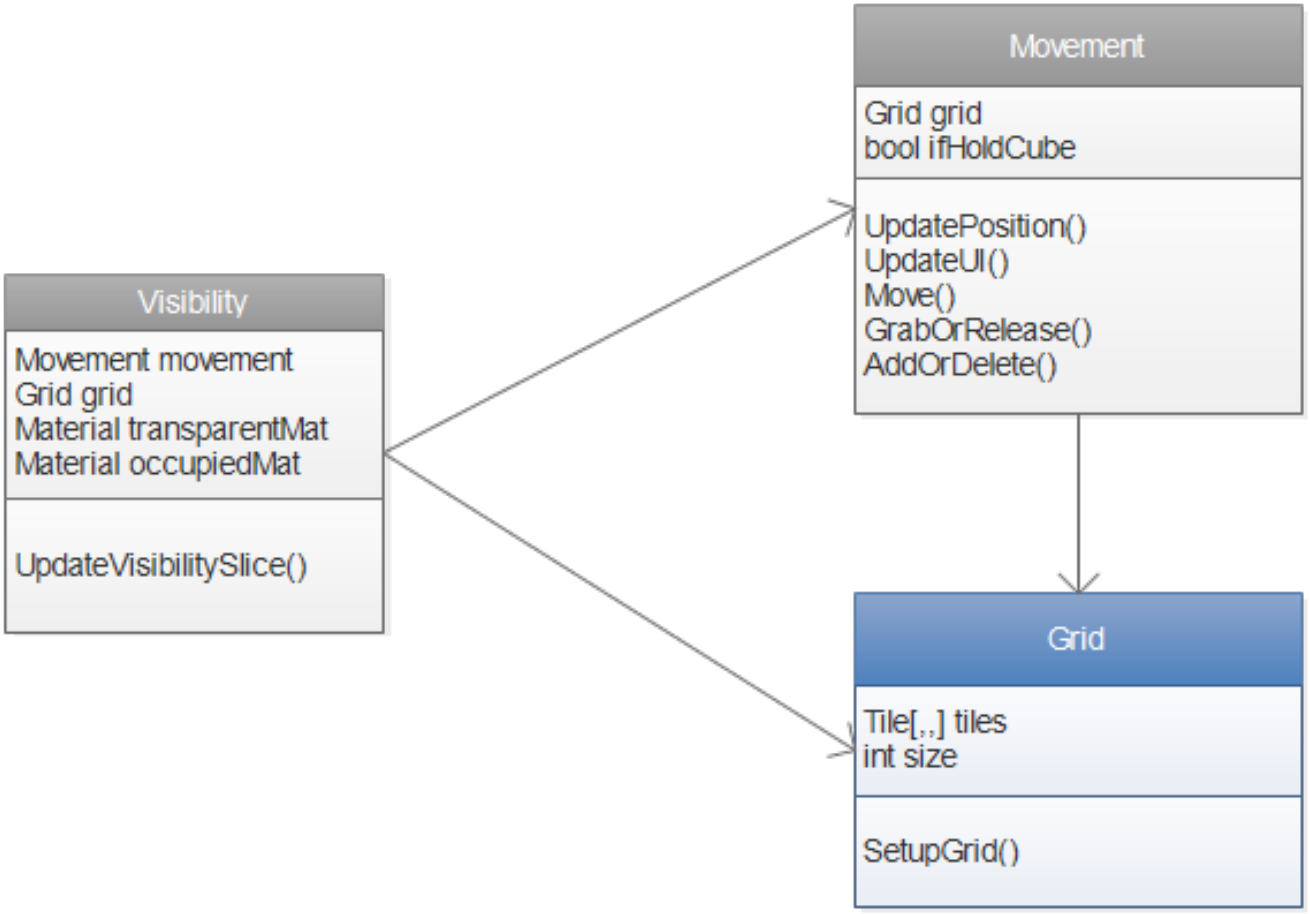


Figure 5.2: Class Diagram: Grid Cube and Rendering

There are three main scripts to process the rendering business of grid and cube. `Movement` class holds the reference of `Grid` class and `Visibility` class holds the reference of both `Grid` class and `Movement` class. `Camera` holds the Visibility script, which gets current position and view direction from `Movement` script and get grid size from `Grid` script to render the cube. To be more specifically, to make the cubes within the ghost cube and player transparent.

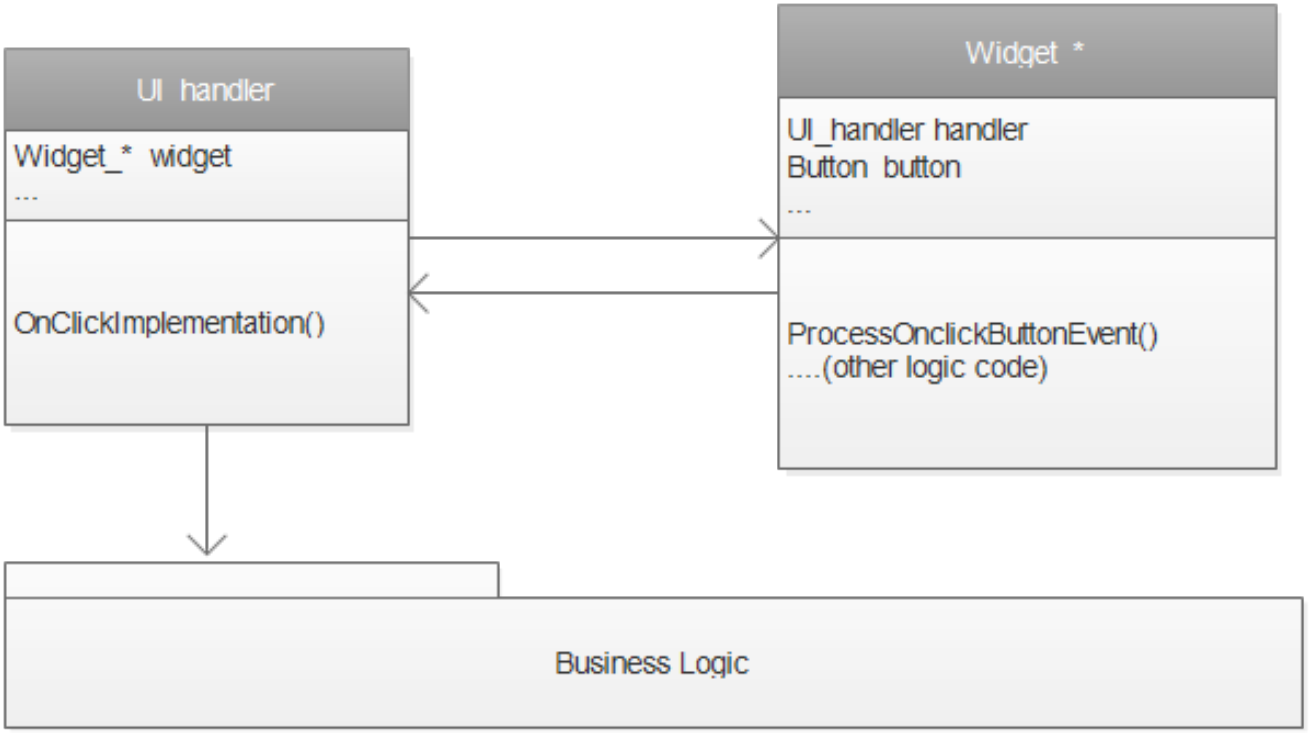


Figure 5.3: Class Diagram: UI handler and Widget System

Because UI relies on the Scene object in Unity3D to be visible and merging scenes is not well supported by git, splitting the logic code from UI part is beneficial for group developing. `UI handler` script is held by scene and work as facade to coordinate widgets and business logic part. For `Widget_*` script which is held by `UI widget` is only responsible to register and bind functions of `UI handler` to on-click events of the buttons. In this case, `Widget_*` only holds the call functions while `UI handler` holds the real implementations of on-click events.

6 PROJECT MANAGEMENT

6.1 SCRUM

The project management frame work used in the development of this project wass Scrum. Scrum is a subset of agile software development. This framework is used world-wide. Utilisation of this framework in the development of this project is discussed in the following section of this report.

6.2 WEEKLY SPRINT MEETING

The duration of our sprints were one week, however in certain circumstances (such as during reading week at Trinity) this was extended to two weeks. Weekly Sprint

meetings are one the key elements of this approach. These weekly sprint meeting are generally half an hour long depending on the status of the project development. They took place every Friday, when the team would get together and analyse the results from last sprint. Keeping it on the right track scrum master would place emphasis on ticking the tasks done from last off the list. Discussion of problems in the tasks from last sprint was done in this meeting and measures were worked out to resolve it. The team would collectively decide features to add in the game for next sprint. Backlog for the tasks for next sprint was noted down in this meeting. The sprint log document sheet was used to keep record of these important things under there topic names such as:

- Tasks Completed from last week.
- Task done this week.
- Task planned to do coming week.
- Task still in progress from past.
- Any urgent issues to resolve.

6.3 Back-log Sheet

The backlog sheet is the document which gets generated from the tasks decided for the sprint from the sprint meeting. These tasks were divided among all team members generally depending of their skill set and area of the project they been working before. These tasks were given the priority value per the urgency of the feature required by the game and how big the task was. These priorities were given 3 values points such that:

- Low - 2 Points
- Medium - 3 Points
- High - 5 Points

Below is the picture this backlog which make it easier to understand its columns.

Sprint 3	High	Complete	37	22	Yes
Visual 3d Grid	High	Complete	5	5	Yes
Ghost cube Movement In 3d	High	Complete	5	5	Yes
Add/Remove/Delete/Move Ghost Cube in 3d	High	Complete	5	5	Yes
Problem In 3d	Low	Complete	2	2	Yes
Solution Check in 3d	Low	Complete	2	2	Yes
Camera Rotation 4 views	Medium	Complete	3	3	Yes
Ghost Cube Movement according to Camera Prespc.	Medium	In Progress	3	0	Yes
Camera Rotation Control UI	Medium	In Progress	3	0	Yes
Visibility-Above Rows of the Ghost Cube transparent	Low	Not Started	2	0	No
Multiple Solution(Not concerned with Direction)	Low	Not Started	2	0	No
Refactoring Code Architecture	High	In Progress	5	0	Yes

Figure 6.1: Backlog Management Example

Next thing in this back log was the section which states that the task was complete or in progress or not even started yet. Next two columns were as afore mentioned for the value point. First value point columns were the points assigned per the priority and next was points given depending on the completion of that task. This backlog was used to keep a check on all the task been done or in progress.

6.4 Burndown Charts

Burndown charts are a usefully way to track the timing of completion of the tasks. It provides the insight of how the project is going regarding the time line. Two columns of value point assigned and given from the backlog sheet were used as data to generate this burndown charts. Burn down charts helped indicating the pace of tasks being done during the sprint. If task were completed before the time the line in the chart goes down from what it is meant to be. This step down indicates that more tasks should be added to the sprint. Vice Versa for the tasks taking longer than expected to complete.

6.5 Daily Meetings

Daily meetings of team member for 5 minutes were very useful. They helped all the team members to know the status of the sprint tasks and know what other team members been up to. One more advantage was that if any team member was having any issue he or she could discuss that and collectively we could find the solution for it. These daily meetings were kept short and to the point. It comprised of each scrum master asking each team member three basic and vital questions:

- What did you do yesterday?
- What will you do today?

- Any issues or problems faced?

With a consistent cycle of these elements, project was well managed leading to a successful result. These tools kept everybody in check with their progress. At the same time, they helped team members to keep a check on the project status. Below is the chart for all the sprints of project providing with insight of progress.

Figure 6.2: Burndown Chart Example

7 Conclusions

The key to a completed project was hard work channeled through an organized process. Our progress from the original idea to completed product owes a great debt to the common metaphor of the Agile framework; begin with a skateboard to build a car. This stepwise progress of tangible, playable iterations can be seen through our management documentation, available in the Appendix.

Rendering was implemented and tested from the player's perspective on all features, accompanied by well-integrated UI. UI was built to have self-explanatory navigation and functions. Code was cleanly refactored to make it well organised and reusable. The scoring system was thoroughly tested and improved with user feedback. Surveys were very crucial part of the project, and provided an insight of all the problems that a user can have while playing the game, as well as offering a glimpse of the response this game would have in the market. All the issues from the

user feedback were fixed with a reasonable compromise to get the best results for a general audience.

Through the collective efforts of each team member it was possible to excel in all the proposed aspects of the game. This would not have been possible without the use of a keenly followed project management process. Using the scrum framework was highly efficient and helped us navigate through many problems in development as they arose. It also allowed us to avoid many issues that can commonly occur in a collaborative project. It led to effective utilisation of all resources and skills of the team members.

Together we were able to produce CubeCraft, inspired by neuroscience research, tried and tested by a diverse sample of games players and completed to a level that is close to ready for shipping.

8 Acknowledgements

All of us on the CubeCraft team wish to thank all our user testers for their highly useful feedback, in addition we would like to thank Dr. Carol O'Sullivan for all her advice during development.

9 References

Boot, W. R. (Ed.). (2015). Video games as tools to achieve insight into cognitive processes. Frontiers Media SA.

Clemenson, G. D., & Stark, C. E. (2015). Virtual environmental enrichment through video games improves hippocampal-associated memory. Journal of Neuroscience, 35(49), 16116-16125.

Lapaine, P., Johnson, M., & Nguyen, N. (2014). Video game and mental rotation training improve spatial anatomy task scores (725.5). The FASEB Journal, 28(1 Supplement), 725-5.

Latham, A. J., Patston, L. L., & Tippett, L. J. (2013). The virtual brain: 30 years of video-game play and cognitive abilities. Frontiers in psychology, 4, 629.

Spence, I., & Feng, J. (2010). Video games and spatial cognition. Review of General Psychology, 14(2), 92.

Spinka, M., Newberry, R. C., & Bekoff, M. (2001). Mammalian play: training for the unexpected. The quarterly review of biology, 76(2), 141-168.

Stanford Letter: A Consensus on the Brain Training Industry from the Scientific Community, Max Planck Institute for Human Development and Stanford Center on Longevity, accessed 5/5/2017, http://longevity3.stanford.edu/blog/2014/10/15/the-consensus-on-the-brain-training-industry-from-the-scientific-community/

Whitebread, D., Basilio, M., Kuvalja, M., & Verma, M. (2012). The importance of play. Brussels: Toy Industries Europe.

10 Appendix

10.1 Sprint Burn Down Charts

			Sprint 1	Week 5	
Day	Dates	Days	Expected	Actual	Completed
Sunday	12/2/2017	6	36	36	
Monday	13/2/2017	5	28.8	31	5
Tuesday	14/2/2017	4	21.6	21	10
Wednesday	15/2/2017	3	14.4	14	7
Thursday	16/2/2017	2	7.2	2	12
Friday	17/2/2017	1	0	0	2

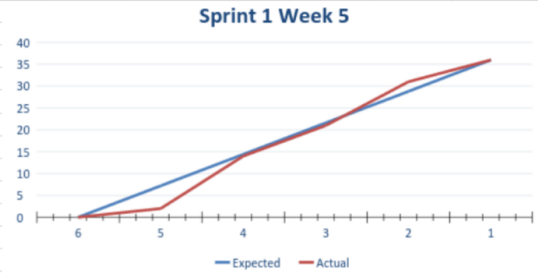


Figure 10.1: Sprint 1 Burndown

			Sprint 2	Week 6	
Day	Dates	Days	Expected	Actual	Completed
Sunday	19/2/2017	6	53	53	
Monday	20/2/2017	5	42.4	43	10
Tuesday	21/2/2017	4	31.8	28	15
Wednesday	22/2/2017	3	21.2	13	15
Thursday	23/2/2017	2	10.6	4	9
Friday	24/2/2017	1	0	0	4

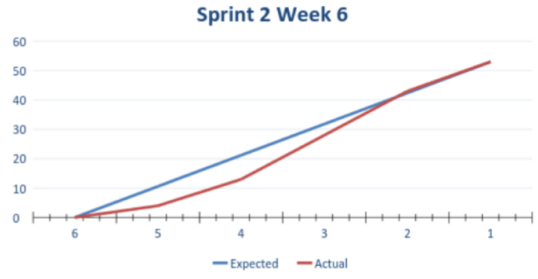


Figure 10.2: Sprint 2 Burndown

			Sprint 3	Week 8	
Day	Dates	Days	Expected	Actual	Completed
Sunday	5/3/2017	6	37	37	
Monday	6/3/2017	5	29.6	22	15
Tuesday	7/3/2017	4	22.2	15	7
Wednesday	8/3/2017	3	14.8	10	5
Thursday	9/3/2017	2	7.4	5	5
Friday	10/3/2017	1	0	2	3

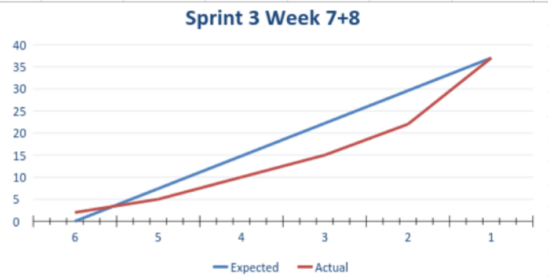


Figure 10.3: Sprint 3 Burndown

			Sprint 4	Week 9	
Day	Dates	Days	Expected	Actual	Completed
Sunday	12/3/2017	6	23	23	
Monday	13/3/2017	5	18.4	18	5
Tuesday	14/3/2017	4	13.8	13	5
Wednesday	15/3/2017	3	9.2	8	5
Thursday	16/3/2017	2	4.6	1	7
Friday	17/3/2017	1	0	0	1

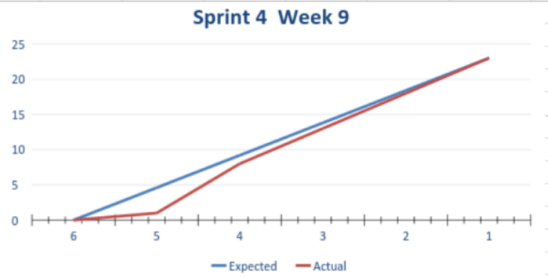


Figure 10.4: Sprint 4 Burndown

			Sprint 5	Week 10	
Day	Dates	Days	Expected	Actual	Completed
Sunday	19/3/2017	6	30	30	
Monday	20/3/2017	5	24	22	8
Tuesday	21/3/2017	4	18	17	5
Wednesday	22/3/2017	3	12	7	10
Thursday	23/3/2017	2	6	5	2
Friday	24/3/2017	1	0	0	5

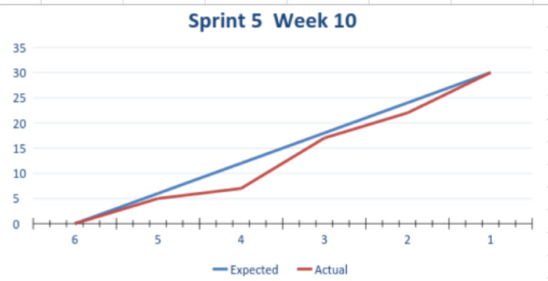


Figure 10.5: Sprint 5 Burndown

			Sprint 6	Week 11	
Day	Dates	Days	Expected	Actual	Completed
Sunday	26/3/2017	6	42	42	
Monday	27/3/2017	5	33.6	32	10
Tuesday	28/3/2017	4	25.2	22	10
Wednesday	29/3/2017	3	16.8	14	8
Thursday	30/3/2017	2	8.4	3	11
Friday	31/3/2017	1	0	0	3

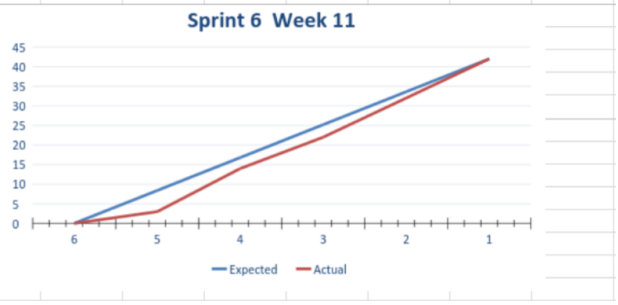


Figure 10.6: Sprint 6 Burndown

10.2 Sprint Daily Meetings

		Samir	Vanya	Dan	Lan	Sean
2/13/2017	1. What did you do yesterday?	Research	Research	Research	Research	Sprint Backlog
Monday	2. What will you do today?	Work On Line Rendrer	Started Line Rendrer	Setup Scene, Grid Class	Start UI, Materials	Find Formula For Grid
	3. Any Issues?					
2/14/2017	1. What did you do yesterday?	Finish Line Rendrer	Finish Line Rendrer	Created Scenes & classes	Buttons and Materials	Formula algo for Grid
Tuesday	2. What will you do today?	Project Management Docs	Grid	Cube Movement	Menu UI	Grid
	3. Any Issues?					
2/15/2017	1. What did you do yesterday?	Created Proj. Mang. Docs	Grid Finished	Cube Movm., User Input	Added Main Menu	Grid Finished
Wednesday	2. What will you do today?	BurnDown Chart	Solution Detector, Timer	Solution Detector, Timer	Continue On UI	Integration UI to Scene
	3. Any Issues?					*Data Structure Loading(P)*
2/16/2017	1. What did you do yesterday?	BurnDown Chart	Solution Detector, Timer	Solution Detector, Timer	Finished Main Menu	Data Structure Loading(S)
Thursday	2. What will you do today?	Testing	Timer Integ. Into UI	Timer Integ. Into UI	Win Message UI	Timer Integ. Into UI
	3. Any Issues?	*Pink UI Components(P)*				
2/17/2017	1. What did you do yesterday?	Fixed Pink Issue	Timer Integ. Into UI	Timer Integ. Into UI	Win Message UI	Timer Integ. Into UI
Friday	2. What will you do today?	Final Deployment	Final Deployment	Final Deployment	Final Deployment	Final Deployment
	3. Any Issues?					

Scrum Master =	Samir Kumar
	(P) = Problem
	(S) = Solved

Week 5

Figure 10.7: Sprint 1 Daily Meetings

			Samir	Vanya	Dan	Lan	Sean
Monday	2/20/2017	1. What did you do yesterday?	Refactoring Week1 Code	Refactoring Week1 Code	Refactoring Week1 Code	Refactoring Week1 Code	Refactoring Week1 Code
		2. What will you do today?	BrainStorm/Plan next sprint	BrainStorm/Plan next sprint	BrainStorm/Plan next sprint	BrainStorm/Plan next sprint	BrainStorm/Plan next sprint
		3. Any Issues?					
Tuesday	2/21/2017	1. What did you do yesterday?	Sprint assignment of tasks	Sprint assignment of tasks	Sprint assignment of tasks	Sprint assignment of tasks	Sprint assignment of tasks
		2. What will you do today?	AI problems/solution	Cube functions w/ Dan	Cube functions w/ Van	Working on ui scenes	AI problems/solution
		3. Any Issues?					
Wednesday	2/22/2017	1. What did you do yesterday?	on finding cell address	ghost cube w/ Dan	ghost cube moving/rendering	testing UI	helped Lan, storage for solutions
		2. What will you do today?	JSON	cube functions + grid	cube functionality	UI maintainance/test	cube funcitons w/ dan + van
		3. Any Issues?					
Thursday	2/23/2017	1. What did you do yesterday?	storage in JSON	w Dan+Sean on cube func	Van+ Sean on cube functions	created buttons/animations	cube funcitonality w dan + Van
		2. What will you do today?	storage of problem set	fixing issues, integrating UI	Resolving errors	implemented UI on screen	fixing errors, ui integration
		3. Any Issues?	Git	Git	Git	Git	Git
Friday	2/24/2017	1. What did you do yesterday?	started problem scene	integrated ghost cube w ui	fix UI	change/improve ui	fix ui integration/resolve git
		2. What will you do today?	finish problem scene	deploy build	deploy build	deploy build	finish problem scene
		3. Any Issues?					

Scrum Master =	Dan
	(P) = Problem
	(S) = Solved

Week 6

Figure 10.8: Sprint 2 Daily Meetings

			Samir	Vanya	Dan	Lan	Sean
Monday	3/6/2017	1. What did you do yesterday?	Refactor Daigram	Refactoring	Refactoring	Ui Research	Refactoring
		2. What will you do today?	Camera Rotaion Start	3D Grid	3D Grid	Camera Rot UI	3D Grid
		3. Any Issues?					
Tuesday	3/7/2017	1. What did you do yesterday?	Camera Rotaion Start	3D Grid	3D Grid	Camera Rot UI	3D Grid
		2. What will you do today?	Camera Rotaion Finish	3d Movement	3d Add Remove Delete	Refactoring	3d Visualisation
		3. Any Issues?					
Wednesday	3/8/2017	1. What did you do yesterday?	Camera Rotaion Finish	3d Movement	3d Add Remove Delete	Refactoring	3d Visualisation
		2. What will you do today?	3D Multiple Problems	Visibilty	Rotational Movement	Images for Buttons	Visibilty
		3. Any Issues?					
Thursday	3/9/2017	1. What did you do yesterday?	3D Problem	Visibilty,Refac(ui)	Rotational Movement	Images for Buttons	Visibilty
		2. What will you do today?	3D Prob/Solu.(rand)	Rot. With Visibility	Rot. With Visibility	Rotation Buttons	Rot. With Visibility
		3. Any Issues?					
Friday	3/10/2017	1. What did you do yesterday?	3D Prob/Solu.(rand)	Rot. With Visibility	Rot. With Visibility	Rotation Buttons	Rot. With Visibility
		2. What will you do today?		Presentation			
		3. Any Issues?					
				Week 8			

Figure 10.9: Sprint 3 Daily Meetings

			Samir	Vanya	Dan	Lan	Sean
Monday	3/13/2017	1. What did you do yesterday?	Animation research	Refactored Visibility	Audio research	UI movement buttons	Refactored Visibility
		2. What will you do today?	Camera Animation	JSON implementation	JSON imp.	UI movement buttons	JSON implementation
		3. Any Issues?					
Tuesday	3/14/2017	1. What did you do yesterday?	Camera Animation	JSON imp.	JSON imp.	UI movement buttons	JSON imp.
		2. What will you do today?	Camera Animation	Rotation ind.solution	Audio implementation	Level Scene UI	Rotation ind.solution
		3. Any Issues?					
Wednesday	3/15/2017	1. What did you do yesterday?	Camera Animation	Rotation ind. Solution	Audio implementation	Level Scene UI	Rotation ind. Solution
		2. What will you do today?	Documentation	Audio implementation	Fix Camera Issues	UI Levels imp.	Audio implementation
		3. Any Issues?					
Thursday	3/16/2017	1. What did you do yesterday?	Documentation	Audio implementation	Level rendering	UI Levels imp.	Audio implementation
		2. What will you do today?	Problem generation	Levels database	Levels database	UI Levels imp.	Levels database
		3. Any Issues?					
Friday	3/17/2017	1. What did you do yesterday?	Bank Holiday	Bank Holiday	Bank Holiday	Bank Holiday	Bank Holiday
		2. What will you do today?					
		3. Any Issues?					
				Week 9			

Figure 10.10: Sprint 4 Daily Meetings

			Samir	Vanya	Dan	Lan	Sean
Monday	3/20/2017	1. What did you do yesterday?					
		2. What will you do today?	volume control UI	add rotation function	visibility script	star on win panel	level scene
		3. Any Issues?					
Tuesday	3/21/2017	1. What did you do yesterday?	volume control UI	camera rotation,sound	Fix visibility problem	star on win panel	level scene, sound value
		2. What will you do today?	instruction help UI	count down timer	score calculation	count down time bar	check solution
		3. Any Issues?					
Wednesday	3/22/2017	1. What did you do yesterday?	initiated instruction	countdown bar; user survey	score function	made tip widget	star,score,deployment
		2. What will you do today?	improved instruction	solution checking	score function	UI scaling	connecting implementations
		3. Any Issues?					
Thursday	3/23/2017	1. What did you do yesterday?	finish instructions	do scoreing calculation	scoring calculation	position of instuction UI	navigation,symtax connection
		2. What will you do today?	new prototype	made the form,overal feedback	fix bugs	add tutorial panel	visibility problem
		3. Any Issues?					
Friday	3/24/2017	1. What did you do yesterday?	new prototype	made the form,overal feedback	fix bugs	add tutorial panel	visibility problem
		2. What will you do today?					
		3. Any Issues?					
				Week 10			

Figure 10.11: Sprint 5 Daily Meetings

			Samir	Vanya	Dan	Lan	Sean
Monday	3/27/2017	1. What did you do yesterday?					
		2. What will you do today?	Instruction buttons	Visibility + swipe	User survey	Improve UI	Visibility + swipe
		3. Any Issues?					
Tuesday	3/28/2017	1. What did you do yesterday?	Instruction buttons	Visibility in gameplay	Write User Survey	UI research	Visibility in gameplay
		2. What will you do today?	User survey	User survey	User survey	User survey	User survey
		3. Any Issues?					
Wednesday	3/29/2017	1. What did you do yesterday?	Instructions+Survey	Survey+Instructions	Survey+Instructions	UI improvements	swipe function + survey
		2. What will you do today?	Conduct survey	Complete build	Complete build	Complete build	Complete build
		3. Any Issues?					
Thursday	3/30/2017	1. What did you do yesterday?	Documentation	Complete build	Complete build	Complete build	Complete build
		2. What will you do today?	Conduct survey	Conduct survey	Conduct survey	Conduct survey	Conduct survey
		3. Any Issues?					
Friday	3/31/2017	1. What did you do yesterday?	Conducted Survey	Conducted Survey	Conducted Survey	Conducted Survey	Conducted Survey
		2. What will you do today?	Build and Demo	Build and Demo	Build and Demo	Build and Demo	Build and Demo
		3. Any Issues?					
				Week 11			

Figure 10.12: Sprint 6 Daily Meetings

10.3 Sprint Weekly Meetings

WEEKLY SPRINT STATUS REPORT 1

TEAM NAME CubeCraft

WEEK 5

MEETING DATE 17/2/2017

WEEK OF 13/2/2017 *through* 19/2/2017

NEXT MEETING 18/2/2017

TASKS COMPLETED FROM LAST WEEK
DECIDED GAME BY DISCCUSION AND VOTING FOR THE BEST IDEA (CUBE CRAFT)
1PAGE PROPOSAL AND PROPOSED UI FOR ALL GAME SCENES
RESEARCH INTO METHODOLOGIES
SELECTED DEVELOPMENT PLATFORM
DECIDED GAME ALGORITHM FOR PUZZELS, BACKLOG FOR FIRST SPRINT
FOUND GOOD UNITY TUTORIALS

TASKS BEING DONE IN THIS WEEK
LEARN UNITY RELAVANT TO THE PROJECT
SET UP UNITY AND GIT PROJECTS ON SYSTEMS OF ALL MEMBERS
STARTED OFF WITH PAIR PROGRAMMING TO GET UNDERTSANDING OF PROJECT STURCTURE
BASICS OF ALL GAME SCENES (MAIN MENU SCENE, GAMEPLAY SCENE, ETC)
NAVIGATION THOUGHOUT THE GAME USING UI(BUTTONS AND OTHER ACTIONS RELATING TO GAME PLAY)
BASIC GAME PLAY WITH TWO CUBES(PROBLEM AND SOLUTION)
GRIDS, CUBES, USER INPUT, TIMER, WIN WINDOW MESSEGE AND UI FOR BASIC GAME PLAY

MORE TASKS BEING DONE IN THIS WEEK
MANAGEMENT DOCUMENTS USING AGILE SCRUM(SPRINT LOGS, DAILY MEETING LOGS and BURN DOWN CHART)
SETUP OF PIN BOARD ON WEB INTERFACE TRELLO.
TESTING AND FIXING OF ISSUES(BUILD ISSUE, UI ISSUE AND INTEGRATION/DATA STRUCTURAL)
FINAL DEPLOYMENT FOR THE SKATEBOARD

TASKS TO BEGIN NEXT WEEK
COMPLEXITY OF GAMEPLAY
MULTIPLE CUBE PUZZELS IN 2D GRID
HIGHLIGHT GHOST CUBE AND ITS FUNCTIONS(GAMEPLAY)
UI(BUTTONS TO MAKE, REMOVE, GRAB AND RELEASE CUBE)
AI(CHECK THE SOLUTIONS FOR EACH PUZZELS)

TASK STILL IN PROGRESS
REFACTORING OF CODE
DEMONSTRATION OF SKATEBOARD

URGENT ISSUES/TEAM SELF ASSESSMENT
GIT SYNC

WEEKLY SPRINT STATUS REPORT 2

TEAM NAME CubeCraft

WEEK 6

MEETING DATE 24/2/2017

WEEK OF 20/2/2017 *through* 26/2/2017

NEXT MEETING 25/2/2017

TASKS COMPLETED FROM LAST WEEK
RESEARCH
BASIC GAME OPERATIONAL
NAVIGATIONAL UI
MENU BUTTONS AND MESSEGES
USER INPUT IN GAME PLAY
GRIDS, CUBES, USER INPUT, TIMER FOR BASIC GAME PLAY

TASKS BEING DONE IN THIS WEEK
HIGHLIGHT GHOST CUBE AND ITS FUNCTIONS(GAMEPLAY)
UI(BUTTONS TO MAKE, REMOVE, GRAB AND RELEASE CUBE)
STORAGE SETUP USING JSON FOR PROBLEMS AND SOLUTION
PLAYER ABLE TO ADD CUBE, REMOVE, MOVE CUBES THROUGH UI.
UI ENHANCEMENTS
INTUATIVE BUTTONS FOR SIMPLER CONTROLS

MORE TASKS BEING DONE IN THIS WEEK

TASKS TO BEGIN NEXT WEEK
EXTEND 2D MOVEMENT/GRID TO 3D
ADD USER CONTROLLED CAMERA
WORK ON SCENE VISIBILITY
REFACTORING THE ARCHITECTURE

TASK STILL IN PROGRESS
MULTILE PROBLEMS

URGENT ISSUES/TEAM SELF ASSESSMENT
GIT ISSUE WASTED TIME DURING THIS SPRINT

WEEKLY SPRINT STATUS REPORT 3

TEAM NAME CubeCraft

WEEK 7

MEETING DATE 10/3/2017

WEEK OF 6/3/2017 *through* 12/3/2017

NEXT MEETING 11/3/2017

TASKS COMPLETED FROM LAST WEEK

TASKS BEING DONE IN THIS WEEK
RENDER 3D GRID
GHOST CUBE MOVEMENTS IN 3D
ADD/REMOVE/DELETE/MOVE GHOST CUBE IN 3D
PROBLEMS IN 3D
SOLUTION CHECKING IN 3D
CAMERA ROTATION
GHOST CUBE MOVEMENTS IN 3D (CAMERA PERSP)

MORE TASKS BEING DONE IN THIS WEEK

TASKS TO BEGIN NEXT WEEK
JSON file format for problems
Rotation Independent Solutions
List of Levels
UI for selecting Levels
Sound FX and background Music
Camera Animation on Rotation

TASK STILL IN PROGRESS

URGENT ISSUES/TEAM SELF ASSESSMENT
GIT SYNC

WEEKLY SPRINT STATUS REPORT 4

TEAM NAME CubeCraft

WEEK 9

MEETING DATE 18/3/2017

WEEK OF 13/3/2017 *through*

NEXT MEETING 20/3/2017

TASKS COMPLETED FROM LAST WEEK
EXTEND 2D MOVEMENT/GRID TO 3D
ADD USER CONTROLLED CAMERA
WORK ON SCENE VISIBILITY
REFACTORING THE ARCHITECTURE

TASKS BEING DONE IN THIS WEEK
Debug/Refactor/Architecture
JSON file format for problems
Rotation Independent Solutions
List of Levels
UI for selecting Levels
Sound FX and background Music
Camera Animation on Rotation

MORE TASKS BEING DONE IN THIS WEEK

TASKS TO BEGIN NEXT WEEK
Player Scoring (i) Timer on problem and solution scene
Player Scoring (ii) False Positive/Negative solution score
Player Scoring (iii) Star system indicating completion
Tutorial Level
Settings/Help Buttons
User feeback elicitation

TASK STILL IN PROGRE

URGENT ISSUES/TEAM SELF ASSESSMENT

WEEKLY SPRINT STATUS REPORT 5

TEAM NAME CubeCraft

WEEK 10

MEETING DATE 24/3/2017

WEEK OF 20/3/2017 *through* 26/3/2017

NEXT MEETING 27/3/2017

TASKS COMPLETED FROM LAST WEEK
Debug/Refactor/Architecture
JSON file format for problems
Rotation Independent Solutions
List of Levels
UI for selecting Levels
Camera Animation on Rotation
Sound FX and background Music

TASKS BEING DONE IN THIS WEEK
Player Scoring (i) Timer on problem and solution scene
Player Scoring (ii) False Positive/Negative solution score
Player Scoring (iii) Star system indicating completion
Tutorial Level
Settings/Help Buttons
User feeback elicitation
Camera Animation on Rotation

MORE TASKS BEING DONE IN THIS WEEK

TASKS TO BEGIN NEXT WEEK
Feedback Response: Button repositioning
Feedback response: adjust visibility calculation
Feedback response (low priority): Swipe to rotate imp.
Conduct additional user survey on previous users
Conduct additonal survey on new users - + respond
Adjust help button instructions

TASK STILL IN PROGRESS

URGENT ISSUES/TEAM SELF ASSESSMENT

WEEKLY SPRINT STATUS REPORT 6

TEAM NAME CubeCraft

WEEK 11

MEETING DATE 31/3/2017

WEEK OF 27/3/2017 *through* 31/3/2017

NEXT MEETING 3/4/2017

TASKS COMPLETED FROM LAST WEEK
Player Scoring (i) Timer on problem and solution scene
Player Scoring (ii) False Positive/Negative solution score
Player Scoring (iii) Star system indicating completion
Tutorial Level
Settings/Help Buttons
User feeback elicitation
Camera Animation on Rotation

TASKS BEING DONE IN THIS WEEK		
Feedback Response: Button repositioning		
Feedback response: adjust visibility calculation		
Feedback response (low priority): Swipe to rotate imp.		
Conduct additional user survey on previous users		
Conduct additonal survey on new users - + respond		
Adjust help button instructions		

MORE TASKS BEING DONE IN THIS WEEK

TASKS TO BEGIN NEXT WEEK
Refactor project for release build
Prepare project summary presentation
Respond to user survey (though no major project changes)

TASK STILL IN PROGRESS

URGENT ISSUES/TEAM SELF ASSESSMENT