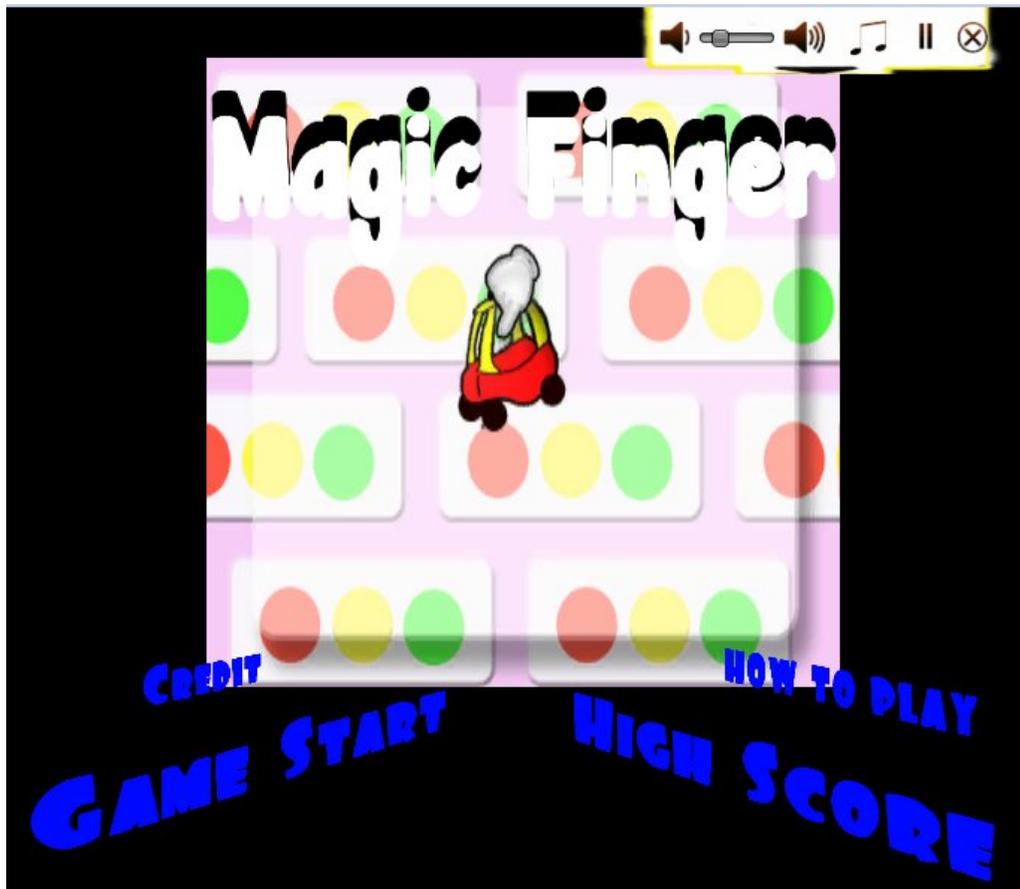


Yu Feng  
CGDD 4803  
Jeff Chastine  
04/30/2012

## Postmortem of Magic Finger

### Introduction:

I have finished most components of this game by the end of this semester. I have been through a lot of frustrations and desperations during the process of planning this game to the closure. There was once I even thought about giving up the whole idea of this game. Whereas, I insisted and made the most use of what I have learned to complete "Magic Finger". I am so delight because all I have learned is shown in this project. All my hard work has been paid off. For instance, the engine I used for this game is Unity 3D. I learned this engine when I took scripting course. So when I started this game, I found actually I knew how to do most of the works and the game went very well for the first a few weeks. The game idea could be founded in CGDD2002 and CDGG4003 classes. For instance, I got the idea and realize the importance of inserting video tutorial, game story and music from that two classes.



**Mian Menu Scene**

I would like to show my gratitude to all of the friends who have supported and helped me in many different ways. I want to give my thanks especially to Yi Xing, my artist; Zolatan, a good friend who revises my post mortem; Yuan Yuan and Pengyun Shen who run the test of this game. I could not finish this game without their help. Through this

experience, I feel the significance of team work. To some extent, it is the key to the success of a game. An outstanding team could not only help me save time, but also provide constant advice to make the game much better. I will illustrate how I design this game in details in the following paragraphs.

## **Game Idea:**

The idea of "Magic Finger" came up even before I took this studio class. Therefore it is somehow different.

Initially I decided to design a game about saving people's lives in the context of air crashes . There are several reasons why I intend to design a game like that. First of all, I read news everyday online. I find that the figure of injuries and deaths caused by airplane keeps increasing. Whenever I hear about tragedies like that, no matter it is an air crash or a car accident, I feel so sad. Sometimes I even have the sense of guilty. I am wondering if there is something I could do to help? Then I think about creating a game based on the idea of air crashes. On the one hand, I want to mourn the people who died in the airplane accidents. On the other hand, I hope this game is able to arouse people's attention about the safety of taking an airplane.

A brief introduction of this game would be:

When an airplane crashes or explodes, passengers will fall down from the plane. The job for the player is to click each passenger to open the parachute. And the passenger's life is saved in this way.

However, I met a lot of questions when I actually began designing this game. I kept asking myself questions such as "how to increase the coefficient of difficulty," "how am I going to make this game interesting and fun". But the most important thing is that I find the game is unrealistic and unpractical. In reality, is it possible for each passenger to carry a parachute with him/her? What is chance for survival if an airplane explodes or crashes? All of sudden, I realize this idea is meaningless.

Struggling for a few days, I decide to change this original idea. I start thinking about train accident, ship wreck instead. Finally, car accident came to my mind. After all, car accident is much closer to people's daily life. I feel maybe people would pay more attention to this theme.

I plan to create a game based on the idea of car accident. The core of this game is to avoid collisions among cars, also between cars and pedestrians. For instance when two cars are in the danger of hitting each other, player need click either one of the two cars to stop them getting any closer. The rule is applied to stop the hit between a car and a pedestrian as well.

It took me a few days to think about the name for this game. In the end, I tried the name" Accidental Safer" and discussed it with my professor. Obviously "Accidental Safer" show a grammar mistake. Without any hesitation, my professor asked me to change another appropriate name. That where "Magic Figure" come from. The idea and name of my game finally settled down.

## **Team Choosing:**

At the very beginning, I decided to do the whole work all by myself. I trusted myself so much that I didn't

want to ask any help from anyone else. In my point of view, it is nothing but a simple casual game, which means there aren't many professional techniques needed. What is more, since I had started drawing and sketching when I was a kid, I am pretty good at it. Despite I must transfer a paper drawing to an electronic one, I am not afraid at all. I have been using tablet for a year, I never thought it would cost me a lot of time to finish several easy pictures. Plus I only need 10 images for this game. Based on my drawing experience, I estimate that three to four days would be the maximum. Moreover, I plan to use Unity 3D to build this game. I have used this engine for more than one year and had developed two projects. I feel I have solved a lot of Unity techniques. "It should be easy", I told myself.

With such a confidence, I even refused the help from my professor. He had found professionals to assist us for the music thing. Being an international student, communication is a challenge for me. It would probably take me some time to try to communicate and to exchange ideas with them. I didn't have so much time to waste. That is why I turned down my professor's help.

These are some objective reason I believe. In fact, deep in my heart, I have always been expecting to develop a game which is totally designed all by myself. I regard this studio work as a huge opportunity for me. I hope I could make an enormous breakthrough this time.

I was stuck at the very beginning. I consider myself as a Unity professional. As a result, I overestimated my skills. I spent almost one day trying to create an animated menu. That was the time when I realized that I could not finish this game completely by myself. I have technique problems to solve, images to draw, and coding to write. The pressure is on. I need someone, at least an artist to assist me.

I am on the way to looking for help. The first thing came to my mind was to begin contacting my old friends and schoolmates to see if someone was willing to do the artist work. I have three close friends who are professional in cartoon design. So I decided to contact them first. I emailed them about my idea and plan of this game as well as the detail information about what kind of images I want, how many images I need. I waited day after day, but nobody replied me until a couple days later. I could understand that they were all quite busy because they were applying to graduate schools or looking for jobs. One of my female college friends said she may give me a little help. Therefore, just two days later she emailed me back and said she was too busy to help. Another college friend showed interest in my project. He did draw some images for me. However, due to time conflict (For instance, when it is daylight in Marietta, it is the opposite in Beijing.) Communication has really become a tough problem between him and me. What I can do is to leave a message or send him an email. He never responded me in time.

I become so depressed and desperate. I need to find a reliable friend who is here in Marietta to assist me. By chance, I met someone whose name is YiXing. She is my schoolmate back to we were in North China University of Technology, China. She is also a two-plus-two program student currently studying in Southern Polytechnic State University. Her major is something related to dress and clothes design. I was thinking that she would have a good drawing experience. I called her and scheduled an interview with her. She was quite interested in what I was doing and agreed to help me instantly. In order to reduce the amount work of mine, she even bought a tablet.

Her serious working attitude impressed me.

Under the help of Yi, I could do the work much faster. It was a nice cooperation with her. She is such a responsible and inspiring partner. She always gave me what I need in time. When I was not so satisfied with her work, she would be very patient to listen to my opinions and then revise it according to my requirements.

Nothing is going to be perfect forever. The problems come out. First of all, she is not a professional drawer, even though she works very hard to try her best to reach my standards, she failed. In some cases, I had no choice but to totally revise what she had already drawn. In addition, since she is graduating from SPSU this semester, she had her portfolio to prepare and a bunch of homework to work on. Time for both of us is so important. Besides, she was upset after several time of being told to redo the work. She kept saying sorry to me.

I could not afford to lose her help as the deadline is nearer and nearer. I have to compromise. I encouraged her to build her confidence. And we reached a decision that she drew the first draft and I would make changes based on her work. I am busy, at the same time, I am still grateful. She does me a great favor when I am desperately need help.



### **Platform Choosing:**

Whenever I have to decide which game platform to select, I always struggle. On one hand, I am keen on trying different platforms especially ones I never used before. On the other hand, due to a limited timeline, I have to compromise and select the game platform that I am comfortable with.

There was no exception for me this time either. I chose the Unity3D because I had some experience on this game engine. I took CGDD3103 before and in that class we created an FPS game using Unity3D. Only when using a game engine I am experienced with, I have the opportunity to finish building the game on time and demo to everyone. This is especially pertinent to have it to show to a company in my portfolio. In my mind, showing your game to someone is not just showing how strong your programming skills are but also showing your ability in game design.

In case anyone doubts that I am choosing Unity3D engine just due to my previous experience with it, rather than the engine itself, I would like to show how strong the Unity3D engine is.

Unity3D, as the name suggests, is a 3D game engine used for creating 3D video games or other interactive contents such as architectural visualizations or real-time 3D animations. Unity3D's development environment runs on Windows and Mac OS X, and the game can run on Windows, Macs, Xbox, PS3, Wii, Web browser, Iphone/Ipad, as well as the Android platform. Therefore, we can easily image that this one game we created can run on any of those platforms without any trouble. It truly saves designers and programmers lots of time.

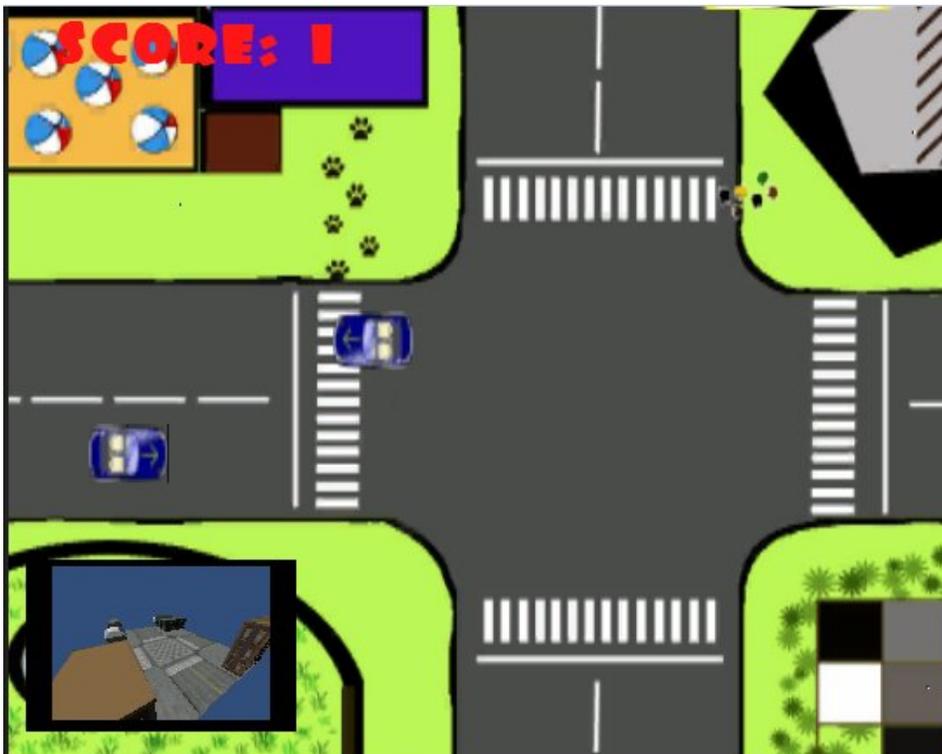
What's more, unity3D uses script based programming so programmers do not need to write plenty of code

to implement a small function like in C++. Ease of use is another big advantage in designers or programmers using unity3D.

In addition, unity3D supports plenty of music, picture, and 3d mode formats. This helps if programmers are confused about how to convert .mp3, .ma, .png to fit unity3D engine. Only thing they need to do is drag them into unity3D, and it will take care of converting stuff automatically.

## Game Design:

As soon as I finished the game idea, I moved to working on the game design document. There I needed to write down more specific requirements such as game title, background story, input and so on. Since, I did not start all of these requirements right away, I decided to think MDA framework first which are mechanic, dynamic, and aesthetics because "MDA is a formal approach to understanding games -- one which attempts to bridge the gap between game and design and development. game criticism and technical game research." In my project, I tried to create a casual game which would be suitable for any age of player. The reason why I chose the casual game format rather than educational, RPG or FPS is because I know casual games have very simple rules and lack of commitment required in contrast to more complex games such as FPS or RPG. For players that means



**Level One - Day Light**

that they are easy to play. For me, as a designer and a programmer, it does not need to cost lots of time to build. Considering my team members and my time schedule, that was the best selection.

After I decided on the game style, I started to think about the game levels. Initially, I determined to create two big levels which meant two different maps. However, because of the artist who did not have enough time to help me

achieve this goal, I had to give up at last and try to design the two small levels instead. Besides the big levels, I tried to involve some small levels in each of big level. For example, in my first map, players can control only two types of cars, but after they gain enough points, there more cars show up in the map. A total of up to four



### Level Two - Night

to a different level which means a different map to enjoy different situations.(but as I said, we still have not built more than one map.)

The idea of setting input as a single mouse click came out of an Iphone game which I played before. I think sticking to simple inputs will help players be able to concentrate more on the game play itself at the beginning rather than having to remember a whole bunch of inputs at the start. If they are core players, complex

input may increase their interest in this game, but for a player who just wants to enjoy a relaxed game play, this might not be a smart choice. Considering the game is for most people, I chose the simple mouse click as my primary game input. What's more, this game I created is more for people to spend some time relaxing their mind rather than spending much time to learn how to play.



### Storyboard

Finishing talking about the level design and input., let's talk about how the game is implemented. First, when the game starts and after the play goes into the main menu, there are four different selections which are game play, high score, tutorial and credit. Players will see that the menu bars are a little tilted. The reason why I did that was because I wanted to show the menu bars more like 3D rather than just placing them into a plane. After player hovers the mouse on one of them, it will move out a little bit and, at the same time, a sound comes out to emphasize they selected that menu bar. When the player clicks one of them, the camera will rotate to that layer. The rotation effect I created was also to show the player that this is not a simple 2D game. It included many 3D elements. Interesting enough, I put the video tutorial in my main menu scene because I think some players do not like to play tutorials in the game especially a casual game. Therefore, I think the video could help them to solve this problem by just having them take a look at the video and then move into the game directly.

After the player clicks the game play button, the camera will move to the game story which is a very important part of the whole game. I realized how important the game story was when I was taking CGDD 4003 and CGDD 4303. I knew the game story goes through the whole game like a chain, so a good story will make the game more fun. Therefore, I created a story layer. Each of the story boards will move towards the player after a while and disappear after they penetrate the camera. That's also another place where I show the player it is not a only 2D game.



**Tutorial One**

While they finish reading storyboards, the camera will navigate players to the game scene. At first they will play two tutorials. I learned the importance of tutorials and how to design a good tutorial from a game research named "The secret to a strong tutorial: Make it fun!" In that article, the writer tells us that the tutorial is not only to teach new players the rules of the game, but also to show the player how fun the game is.

He also said "...the whole reason people play games is to enjoy them and their first impression will go a long way in shaping their future experience." Therefore, if the tutorials were badly designed, no one would like to play the real game. Although in my game, I just use the tutorial to show the rules of my game and it seems not

very fun, but my tutorials are short, so before the players feel bored, the game will already begin.

In the real game, the player needs to avoid the car colliding with other cars or people. When the car moves out of the edge, the player will get one point. The players should try their best to get as many points as possible because, after they finished the game, their score will appear in the "high score" board which will only show the five highest scores.

In order to make the game more fun, I tried to create a 3D version of the magic finger after Dr. Chastine gave me some good advice. In the 3D version, the functions are exactly the same as in the 2D except the buildings and cars are created using 3D models. These will give players some version impacts and attract them to keep playing my game.

Besides using the 3D version to attract players, I also created beautiful music to keep my players engaged in my game. Although the music right now is just picked up randomly, I know the music also occupies a great deal in the game, so in the future, I will definitely update to more attractive music.

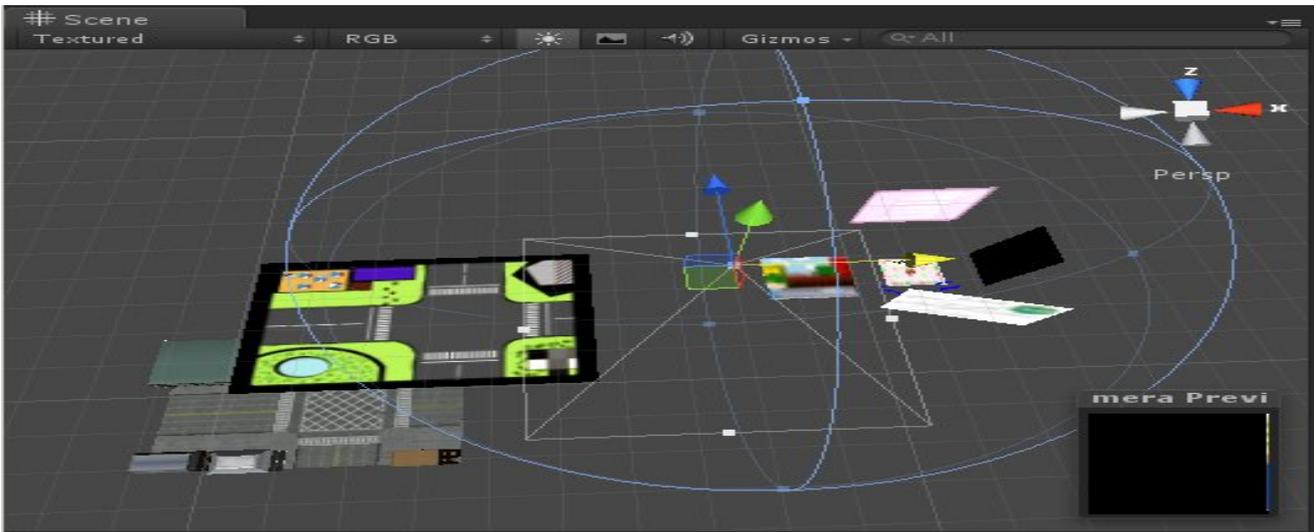


**3D Version**

This is all the details of MDA. After I finished the game documents. It was time to move to designing a timeline at the beginning when I finished my initial timeline. I felt confident. I thought it would be no trouble for me to follow this timeline. However, after I finished this semester, I found I changed many details in my timeline. For example, the artist promised me to continue giving me textures but she ended up giving me textures at the

beginning of March only. That means I could not improve more than one map at the end of the semester.

Therefore, I changed the timeline to finish two small levels which were the day light and night rather than two large maps. I changed my timeline in a total of more than five different places at the end.



These are all my game layers

### Technical Problems:

Before I started coding my game, I listed all of the challenging parts. In this game, my list of challenging parts included a moveable menu bar, menu bar click, moving and rotating camera, car collision, event triggering, moving people animation, video/music importing, and attaching textures to 3d models. After listing all of the challenges I needed to face, I calculated the approximate time for finishing all challenges so that I could further figure out the whole cost of my project. What's more, I knew more specifics about what I was supposed to do; in other words, I have broken down the conundrum of my main goal of creating "magic finger" into small, more manageable goals. I always think this is a very good way to start a big problem. At least this way helped me feel less afraid of this project.

When I finished listing my challenges, the rest of the things seemed easier. All I needed to do was find answers. My best place to find such answers was YouTube. As a foreigner, reading some materials or books would cost me much time. What's worse, misunderstanding the contents of materials would sacrifice more time. However, video tutorials helped me solve all of these problems. Therefore, YouTube was my first priority in seeking tutorials. To some of my challenging issues, like menu bar moving, menu bar clicking, importing video/music and attaching textures to 3d models solutions were all found on YouTube. Seeing how to solve these problems in videos and then imitating them was easy to do and easy to memorize. After watching the videos, I looked at the comments. I found that sometimes I can find more valuable resources from these comments. Second way for me to find answers is using Unity answer blog. This is a great place to find all kinds of problems. In there, different people help me find answers after I post my problems. With others' help, I finally figured out the rest of my challenges.

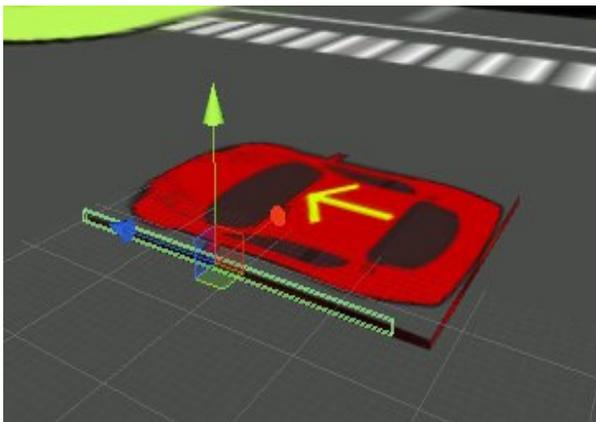
After finding all of the solutions, I started to create a test version of studio game in Unity3D. In this version, I tested all of the solutions to make sure that they worked fine, and saved this test version so that it could help me

in the future should I meet the same problems. Not only did I save the test version of my project, I also wrote down some related codes in my notes. I thought keeping these notes and projects will definitely help me save lots of time finding solutions in the future.

When I finished finding solutions for and implementing all of these challenges, I began to organize my code structures. In the past, I really did not care about organizing structures, but this bad habit had increased difficulty of coding. At the beginning, I wrote different functions into each object and then I created some boolean variables in the main class to manipulate all the other functions. At first, this way went smooth because I did not have so many functions so it was easy to control. However, after the codes increasing in complexity, I was confused on some of the boolean variables. I started to forget which one controlled which function. Luckily, I did not insist on my continuing writing in this style. I switched to state machines which including reset, tutorial one, tutorial two, play game, replay, and story moving. These six states replaced many boolean variables and made the codes became easy to read. After I switched to the state machine method, I realized the importance of organizing code. A well organized code helps programmers not only be able to search fast but also to read easily.

After I finished reorganizing, the next problem came out which was moving the camera to different positions. I could not imagine that this simple script would block me for a long time because it was worked successfully in my previous project when I was taking CGDD 3103. At first, I used the same function as my previous project which was a Translate function, but the camera could not move after the storyboard layer moved away. Then I switched to the Lerp function, but it still could not move smoothly. It seemed like something blocked the camera's moving path, but these functions worked perfectly in my test version. At last, I changed to camera position.y in my update function and it worked. The reason why the previous three functions were not working was still not found.

The problems came out one after another. When I figured out camera moving, another issue came out. I even doubted whether I really needed a test version because many functions worked fine in the test version but they worked bad after I combined them with others. Here is another example, the car collision worked fine in my testing version, but in my real version, the collision function died. The reason was terrain. After the car appeared on the terrain, it would fall on the terrain because of gravity. Therefore, they could not avoid colliding. In order

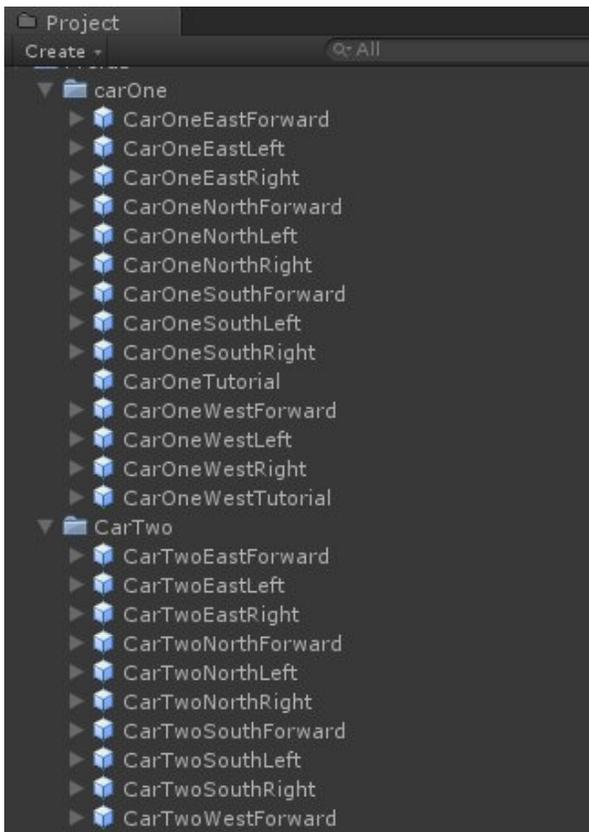


to solve this problem, I created several cubes around the car and left some space between the terrain and cubes and then I put OnCollision event into those four cubes rather than in the car itself. This was the only way the collision could be detected.

These problems were easy to fix. It only cost me a couple of hours to fix each of them. Nevertheless, a huge problem was waiting for me. The problem was how to make the same car appear prefab but moving in a different

direction. Initially, I tried to control the car appearing function in my main class and it worked well. The car

could populate in four different directions. However, when the car moved into the trigger area, in the center of the map, that was used to tell the car to either turn left, right or go straight, the rest of the cars that were not in the trigger zone would move in exactly the same direction as the car in the trigger zone. That really confused me a lot. I kept asking myself why this would happen. Obviously the cars were different. Since then, I tried different methods to fix this problem but had no effect. At last, I used a stupid way to make that happen which was creating different prefabs which only held one direction (like the picture shows below). I knew they really cost a lot of memory, but without using this way, I could not figure out a better way to overcome this problem at that time. After struggling for three days seeking a better idea, I had to give up and use this idea because I was already late for my timeline. I knew that if I did not move forward, I may not finish this game in the end.



After I decided to go this way, all I needed to do was generate different prefabs to hold each of the cars. This way worked perfectly, but cost a lot. When I moved into the 3d version of the game, I could not create the prefab like in the 2d. Otherwise the game would become slow and super large because of my many models. I am still aware this is a terrible way to make different cars appear in this game. Hopefully, I can improve this function to another effective way in the near future.

Although I had to face many challenges while I was coding, I figured out most of them and finally came to the end.

All in all, during the coding, although I met a lot of problems that I tried to fix, I still strive to overcome them. What follows is my summary of points that still need improving in the future.

First, how to organize the code is very important and worth deep study. I think I should accumulate different data structures and algorithms in my daily life so that one day they can help me build a highly efficient game.

Second, I need to keep searching some good tutorials and watch them to improve my Unity3D skills. I think I should at least be proficient in one game engine in order to give me more opportunities to find a job in the game industry.

Third, I need to keep improving my problem solving abilities. In the future, I hope to find more efficient, faster, and more useful solutions.

## Testing:

### From my tester (Yuan Yuan):

"Magic Finger" is an amazing game. I (Yuan Yuan) played this game right after Yu Feng Finished it. It was rough back then, therefore, it was brilliant in my opinion. There are several reasons why I like this game. First of all, the game idea is pretty awesome. Based on what Yu Feng has told me "I create this game because I see news about car accidents, I feel I should do something". This is an simple casual game, but the deeper meaning hiding behind the story is to arouse people's attention about driving safety. Secondly, the game's coefficient of difficulty is increasing step by step. Initially, the game is played in the daylight and later in the evening. The difficult increases when it changes from daylight to evening, which indicates that driving in the evening is much harder than driving in the dark. This change makes people be aware of that in the evening, they should be more careful.

Thirdly, the instruction of this game is clear. In the main content, there are "credit", "game start", "high score", and "how to play". When you click, for instance "game star", there will be a background story showing to the player. The part I like the most is the tutorial instruction in the "how to play" section. Traditionally, when designers design a game, they tend to use text to teach people how to play. However, the fact is people are not inclined to read the long, boring instruction. Considering about this situation, Yu make a video tutorial to providing the information to its players.

Fourthly, I would like to talk something about the graphic used in this game. I like the design, the colors in the images are harmonious and contrast, such as the combination of red and green. The scene is realistic, which gives players a feeling that they are actually driving on the street.

Everything has two sides. The game also has some drawbacks. First thing comes first, the background music and the whole game doesn't match so well. The music is so beautiful, peaceful and elegant. Think about how a casual car accident game has any connections with such a beautiful song?

What is more, the car's speed is too fast. It is so fast that players have some difficulties to play the game, let alone get to the next level. When I played this game at the first time, I found myself being so busy. I could make the game last for more than five seconds. I highly suggest Yu to decrease the speed to make the game more playable.

Overall, it is still a good game.

### From my tester (Pengyun Shen)

I(Pengyun Shen) was very glad that Yu asked me to test his game. I am a super game fan. In my point of view, this game is good, especially the game idea. Some of the games that I have been playing with, such as Plants VS Zombies, is full of fun. Therefore, what is the point of this game if you try to think further? Yu's game is aimed to arouse a fact to the public that nowadays car accidents have become more and more serious. It's high time we did something to make people pay more attention to this problem

The excellent idea is the most impressive components of this game. Therefore, there are something negative

that I would like to point out. First and foremost, the game is not so fun. Even though the idea of this time is to educate people, it should be fun. Nobody wants to spend time on a boring game. What is more, think about the potential players of this game. In my opinion, most of them would be kids and youth. It doesn't have a large market share. Taking Plants VS Zombies as an example. This game could be played by youth, adults, middle-age and even senior. The reason is simple: It is easy to learn and fun to play. Comparing with Plants VS Zombies, Yu's Magic Finger fails to be competitive in the game market. In addition, I have the same thought as another tester, the speed of cars and pedestrians should be less faster. When I was testing the game, I was unable to make it to the second level. I don't like the background music either. It doesn't have any connections with this game. Whereas I still feel that this game is a success. This is the first time that Yu completes a game on his own. The design part, including the choose of colors, images, and layouts are very cohesive and consistent.

### **From myself:**

Problems are involved in this game. When you start the game, you can choose different options in the menu section. The camera is supposed to rotate until it locks the target layer. However, the position is really hard to locate. The function button doesn't do its job the first time I ran a test. When I finally learned how to make the camera locate at the right layer, I found I could not go back to the main menu from the current layer. None of these problems is part of my plan.

Another problem is about the smoothness of the transition from the storyboard to the game layer. To make the background stories move from one to the other, I used Vector3.lerp functions. Originally, right after the end of the background stories, the camera is supposed to be moving to the game scene. It never moves the way I wanted.

There are other technique problems as well:

- (a)When I click a moving car, the car is still moving instead of stopping.
- (b)"Replay Layer" doesn't show up after the collisions between pedestrians and cars, or cars and cars.
- (c)Cars and pedestrians are moving at the same speed, which means when a car comes, so does the pedestrian. This makes the game much easier.
- (d)The speed I set for the car is one hundred and twenty. I didn't realize the speed was so fast until I tried once. It was so fast that I could hardly make it to the second level of this game.
- (e)View conversion from 2D to 3D is troublesome. On the one hand, the size and direction of camera is hard to adjust. For instance, the camera is supposed to be decreasing when it converts from 2D to 3D. On the other hand, even if the camera is able to enlarge or lessen, the letters could not do the same thing. Consequences are that, the objects are covered by the letters.

These problems I have met make me realize that I am far beyond completing this game. If the game cannot function smoothly and successful, the game fails. That is also when I realize game testing is so important. Game testing could help game designers learn problems and game. After several times of testing, my game becomes much better and fun.

What is more, I feel there are some problems which could be avoided. Whereas I didn't think through all the problems because of the limited time. This experience teaches me the importance of details. In my case, if I could have used more efficient well organized codes, I don't need to make so many changes later.

Still, I learn more from this experience. I am delight that I could design a complete game in such a short period all by myself. Beside, I am learning how to do a team work; the ability of time management, and game design.

## **Conclusion:**

This is my experience. Even though time is limited, we are still able to achieve the core content of this game. In the middle of the game, there are so many problems popping out, such as the game schedule is not comprehensive. What is more, I also find design problems which make me unable to finish the work in time. For instance, I was thinking that I could use only one week to overcome all the difficulties and challenges. However, it took me nearly one month.

The schedule is not considered comprehensive because I do not have a deep understanding of the game engine. So I plan to learn more about Unity 3D and master more ways to solve different problems. At the same time, I need to take in more knowledge about how to design a game. By achieving this goal, I am intended to read more game design magazines. Another huge problem is coding. I still need to improve my coding ability. Due to the unclear code structures, I spent a lot of time to revise them. As for the design part, the maps is so small that players have no time to respond what is happening. The game is over very soon. The speed set for cars and pedestrians is too fast which increases the game's difficulty in a indirect way. From this game, I tell myself that I need to think about more in terms of designing.

The bright side is that this game is a combination of 2D and 3D. Players would enjoy both 2D and 3D in just one game. Another success is the "level design". The coefficient of difficulty is increasing step by step. All in all, I think this game is successful.

Again, I would like to thank my friends for helping me all the time. They are Yi Xing, Zoltan, Yuan Yuan and Pengyun Shen. I will work on and make this game become perfect in the future, and hopefully one day I can make this game not only playable in web version but also playable in multiple platforms.