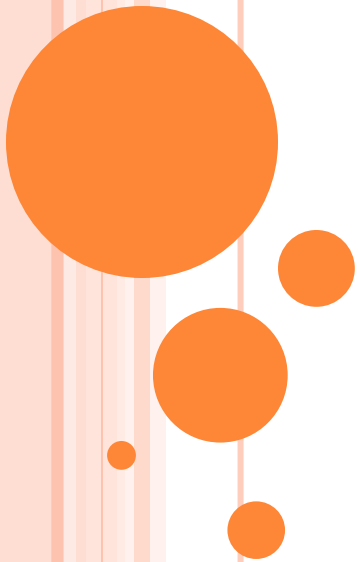
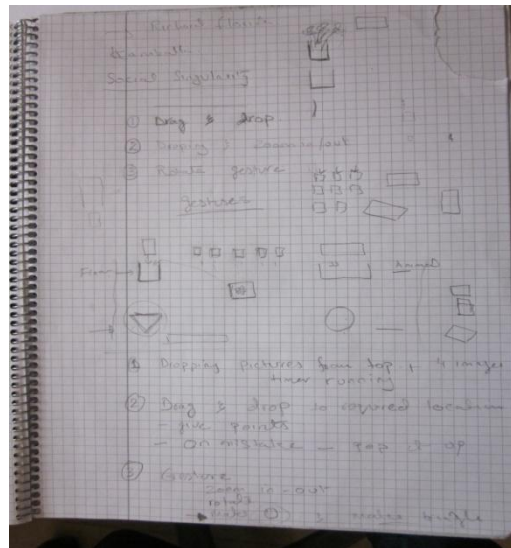
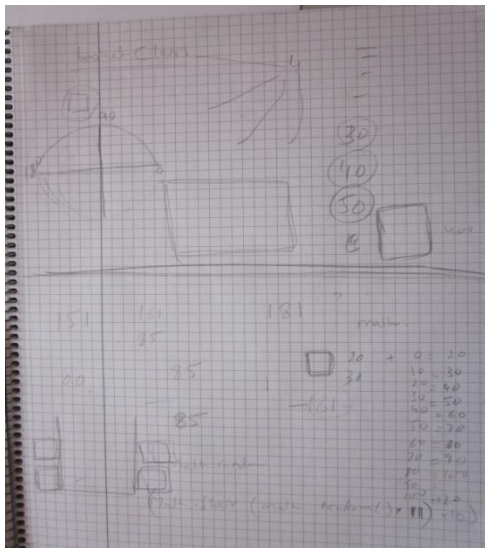


# **MULTI-TOUCH PROJECT DOCUMENTATION**

**Naushin Malik  
Lizzy (Xu Yan)  
Bilal**



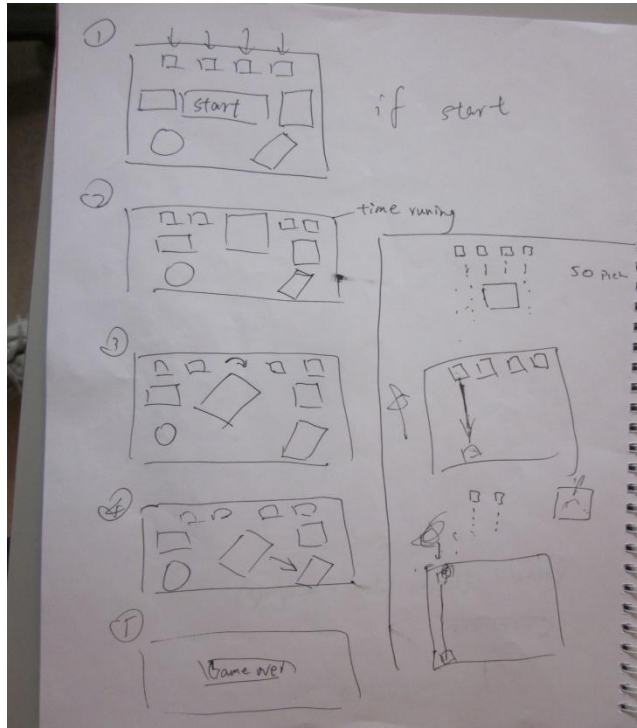
# SKETCHING (MAIN IDEA)



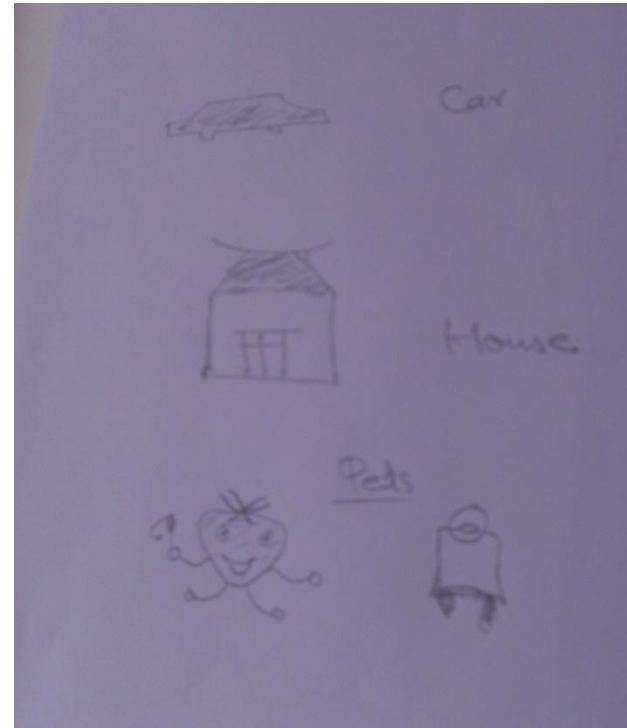
The initial rough sketches of the Idea we came up with.



# SKECTHING (CONTINUES)



Complete over flow of the game.

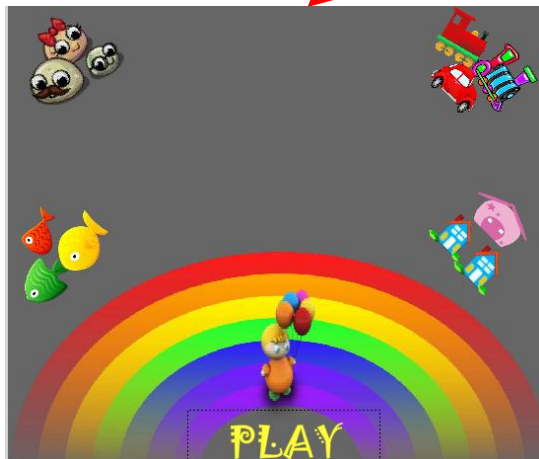
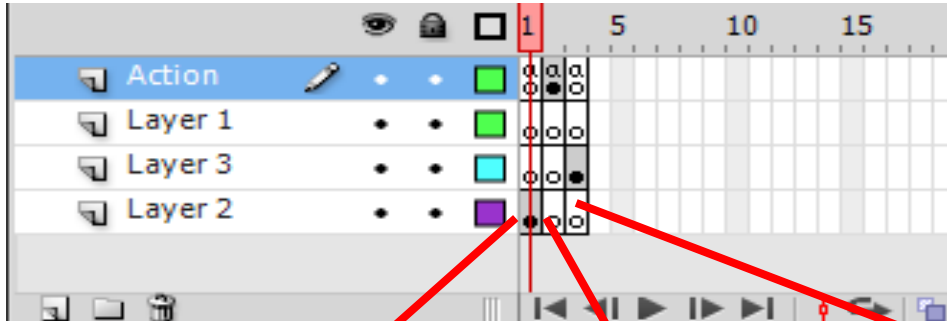


Sketching objects for the game.

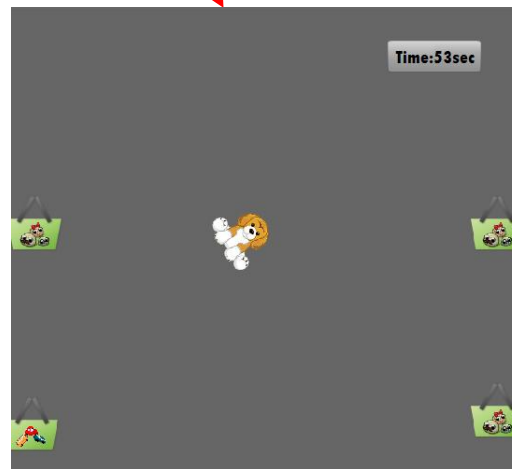


# NAVIGATION

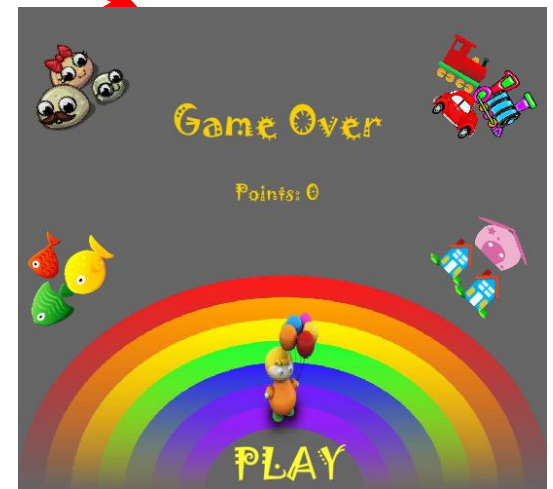
Use the timeline to create three sections and be able to navigate.



start



play

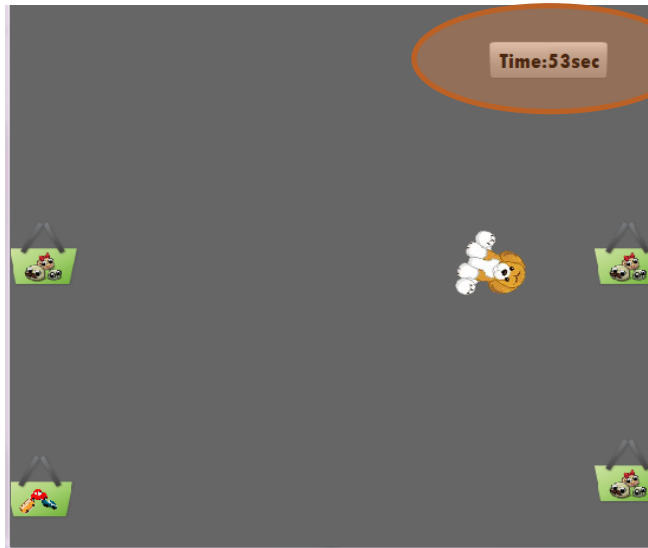


Game over



# COUNTDOWN

The countdown was used to control the time of playing.



Time:58sec

Time:1sec



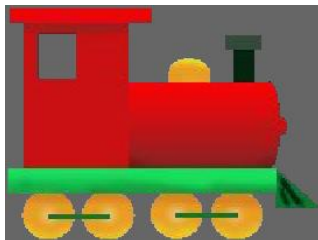
```
4 var i:int = 60;
5 time.label = "Time:" + i + "sec";
6 var count_game:Timer=new Timer(1000,i);
7 count_game.addEventListener(TimerEvent.TIMER, ontime);
8 function ontime(e:TimerEvent):void{
9     i--;
10    time.label = "Time:" + i + "sec";
11 }
12 count_game.start();
13
14 count_game.addEventListener(TimerEvent.TIMER_COMPLETE, countGameC);
15 function countGameC(event:TimerEvent){
16    trace("Game Over");
17    gotoAndStop("gameover");
18 }
19
```

# DIFFERENT CLASSES

Different classes



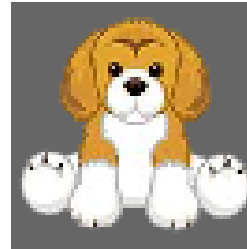
Cars



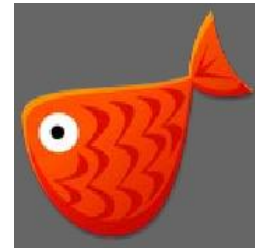
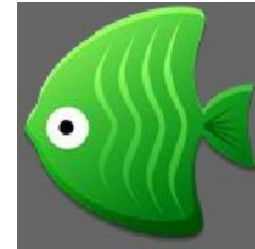
Houses



Pets



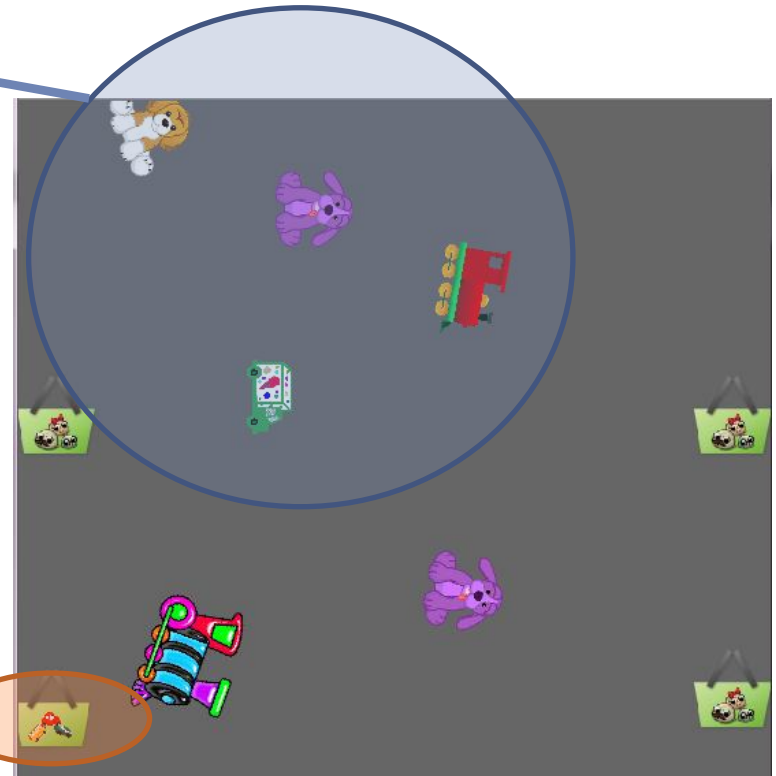
Fish



# OBJECTS DROP OFF RANDOMLY

```
80
81 function createobject(event:TimerEvent):void
82 {
83     rnd_number = Math.round(Math.random()* 9);
84
85     if ( rnd_number == 1)
86     {
87         var pet_mc:Pet = new Pet((100 + (Math.round(Math.random()* 30
88         pet_mc.setAngle(30+(Math.floor(Math.random()* 11)*10));
89         addChild(pet_mc);
90     }
91     else if ( rnd_number == 2)
92     {
93         var pet_mc2:Pet2 = new Pet2((140 + (Math.round(Math.random()*
94         pet_mc2.setAngle(30+(Math.floor(Math.random()* 11)*10));
95         addChild(pet_mc2);
96     }
97     else if ( rnd_number == 3)
98     {
99         var car_mc:Car = new Car((100 + (Math.round(Math.random()* 30
100         car_mc.setAngle(30+(Math.floor(Math.random()* 11)*10));
101         addChild(car_mc);
102     }
```

Objects have different angles.



```
32 //baskets
33
34 // pet basket
35 var petbasket:Basket = new Basket ();
36 addChild(petbasket);
37 ///car basket
38
39 var carbasket:Car_Basket = new Car_Basket ();
40 addChild(carbasket);
41
42 // house basket
43 var housebasket:Basket = new Basket ();
44 addChild(housebasket);
45
46 // toolbasket
47 var toolbasket:Basket = new Basket ();
48 addChild(toolbasket);
49
```

○ Add four baskets

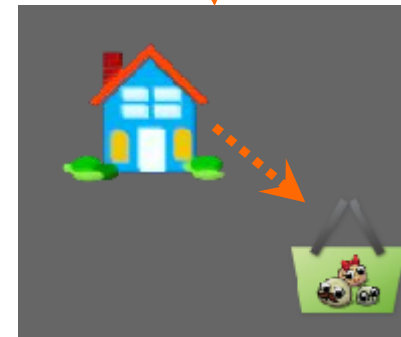
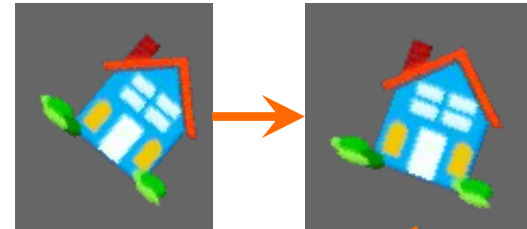


- TOUCH THE OBJECT AND MAKE IT STOP
- CORRECT THE ANGLE AND PUT IT IN CORRESPONDING BASKET

```
25
26     car.addEventListener(Event.ENTER_FRAME, fallingDownEnterFrame);
27     car.addEventListener(MouseEvent.CLICK, hitDetection);
28 }
29
30 public function getAngle():Number{
31     return rotate_angle; //return angle
32 }
33
34 //method for setting new rotation angle
35 public function setAngle(angle:Number):void{
36     rotate_angle = angle; //set new value to rotate_angle variable
37 }
38
39 public function fallingDownEnterFrame(event:Event):void
40 {
41     if(car != null)
42     {
43         car.y += speedY;
44         car.rotation = rotate_angle;
45     }
46     else
47     {
48         trace ("test");
49     }
50 }
51
52
53
54
```

•  
•  
•  
•  
•  
•

so much coding





# SCORE SYSTEM

```
142 function updateScore ():void
143 {
144     score += 100;
145     trace (score);
146     scoretimer.start();
147 }
148 }
149
150 function scorepopup(event:TimerEvent):void
151 {
152
153     addChild(score_object);
154 }
155 }
```

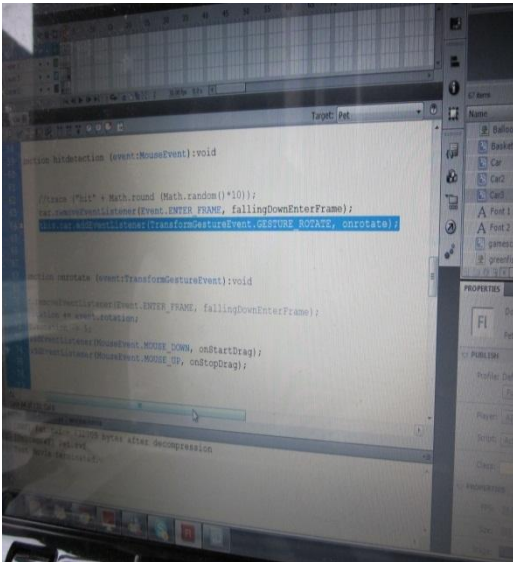


# PLAY AGAIN

```
1 stop ();
2
3 score_text.text = "Points: " + String(score)
4 start_mc.addEventListener(MouseEvent.CLICK, onreStart);
5 function onreStart(event:MouseEvent) {
6     gotoAndStop("Play");
7 }
8 }
```



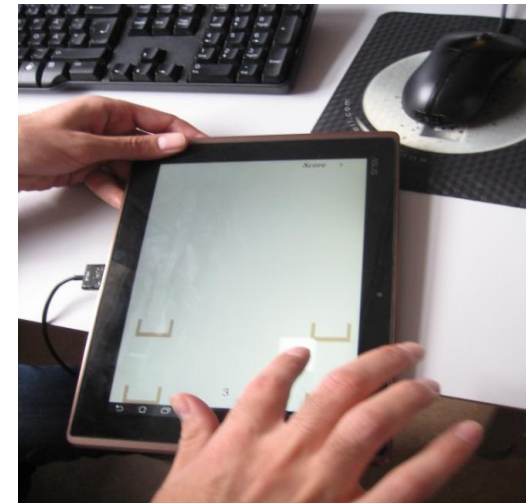
# CODING AND TESTING



Coding the program



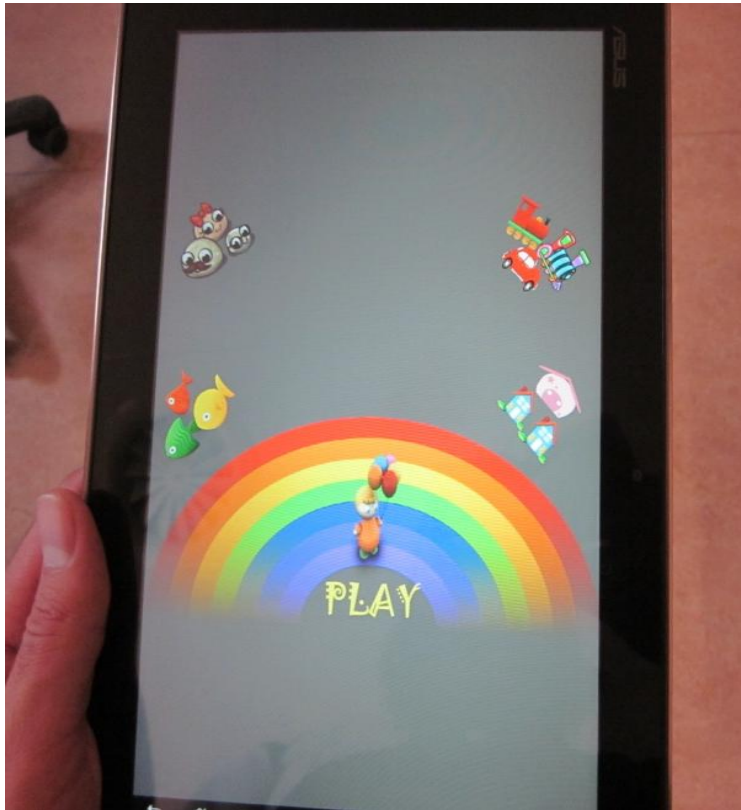
Discussing and Testing



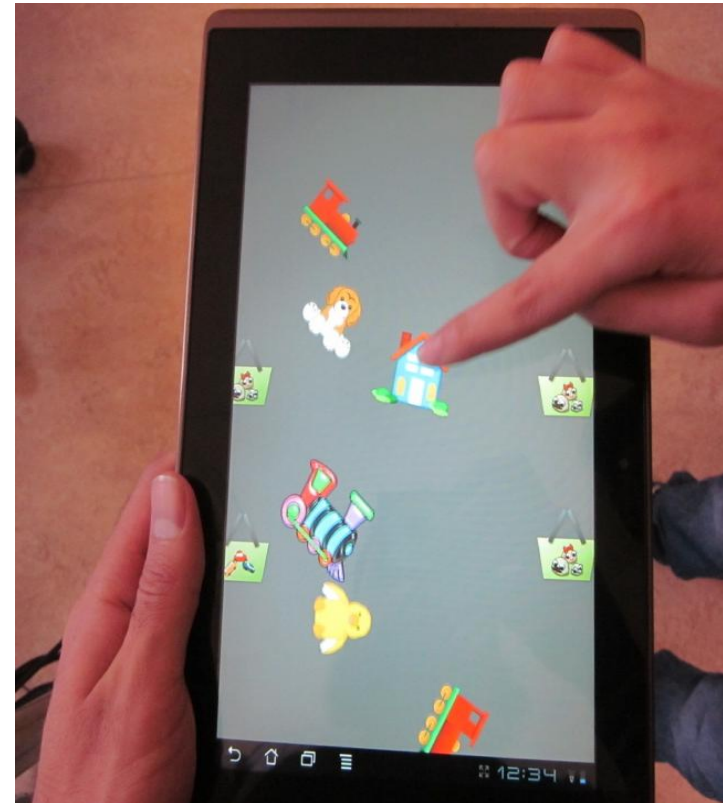
Initial Prototype Testing



# USER TESTING AND FINAL OUTPUT



Final Output

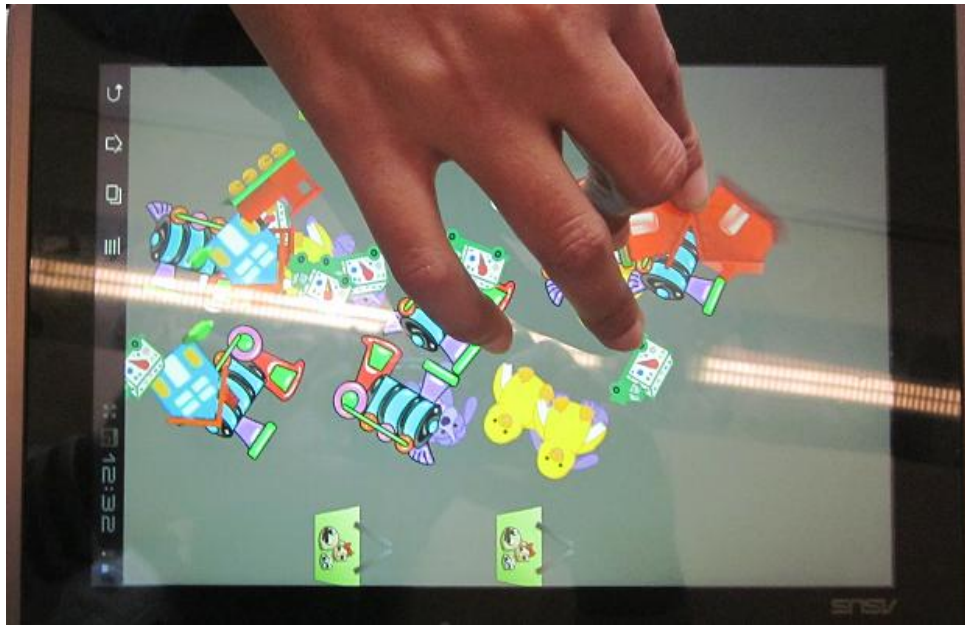


User testing



# PROBLEMS

- Rotations (Smooth)
- Add/ Remove Child from the stage after run.



# FUTURE

- We overcome to remove the child from the screen but I think we need to use a more dynamic approach .
- The Smooth rotation could be achieved through tween but due to time constraint , it was difficult to achieve before the deadline.
- Friction could also be the solution to smooth the rotation of the object.



*Thank You*

