**3D**仿真钓鱼游戏

总

体

设

计

报

告

****

课 程： 软件工程导论

题 目： 3**D仿真钓鱼**游戏**总体设计**

专 业： 软件工程

班 级： **软工1701及1702**

小组成员：**郑鸿棣31701298(组长)**

**徐余浩 31701297**

**陈瑜安31701279**

目录

一．引言 3

1.1编写目的 3

1.2项目背景 3

1.3参考资料 3

二．程序系统结构 3

2.1 数据流图 3

2.2E-R图 4

2.3状态转换图 5

2.4 程序方框图 6

三．钓鱼游戏设计说明 7

3.1程序描述 7

3.2功能 7

3.3性能 7

3.4 输入项 7

3.5 输出项 8

3.6 算法 8

3.7 接口 10

3.8储存分配 10

3.9注释设计 10

3.10限制问题 11

3.11 尚未解决的问题 11

四．项目计划展示 11

预计3个月完成该项目 11

五．界面原型展示 12

六．软件测试 14

七．软件维护 14

# 一．引言

## 1.1编写目的

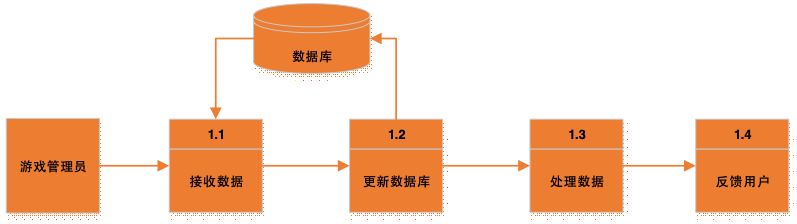
为那些想体验一下钓鱼或者偶尔钓一下鱼的人，提供一款游戏来缓解念想。

## 1.2项目背景

钓鱼，作为一种悠闲自得的运动项目，在现在快节奏的生活下，便变成了难以实现的奢望。因此，我们希望为那些想体验一下钓鱼或者偶尔钓一下鱼的人，提供一款3D仿真钓鱼游戏，让他们在移动设备上体验钓鱼的乐趣。

## 1.3参考资料

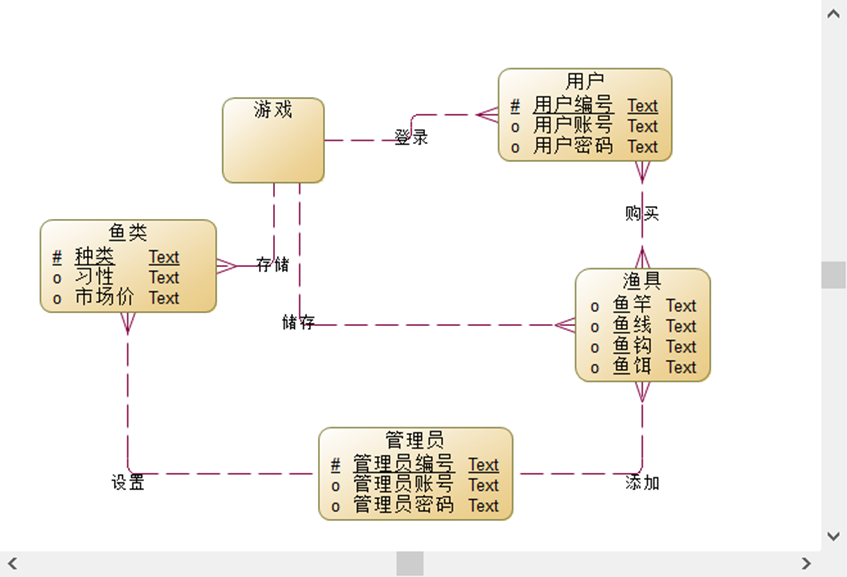
1 张海藩，牟永敏. 软件工程导论. 北京：清华大学出版社，2013

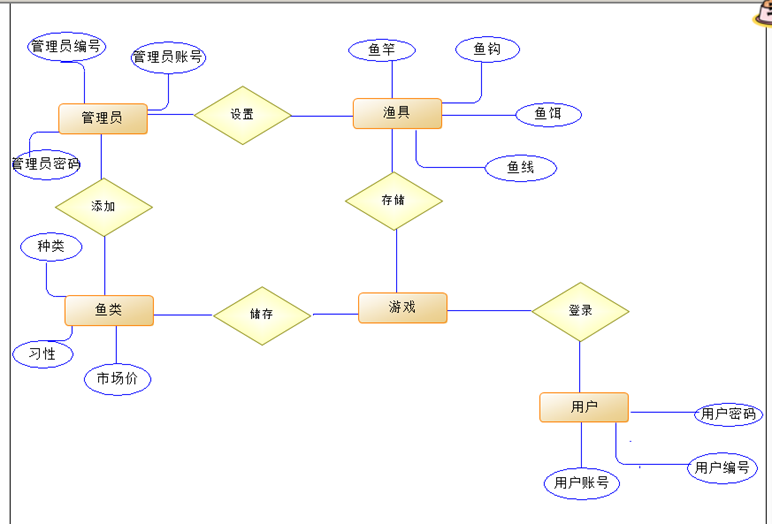
2 刘国柱. Unity3D/2D 游戏开发从0到1. 北京：电子工艺出版社，2018

# 二．程序系统结构

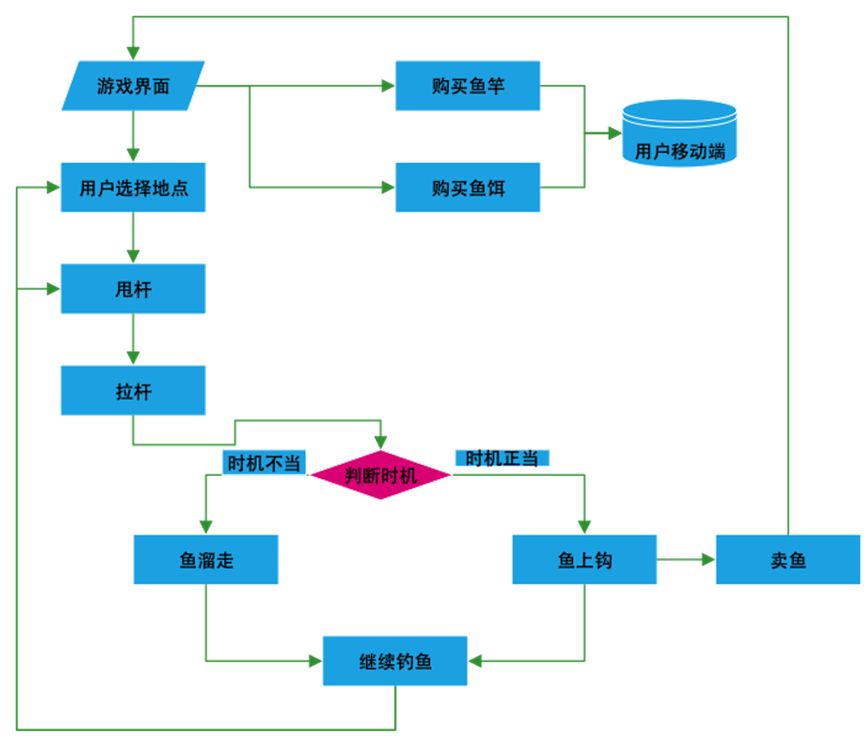
## 2.1 数据流图

## 2.2E-R图

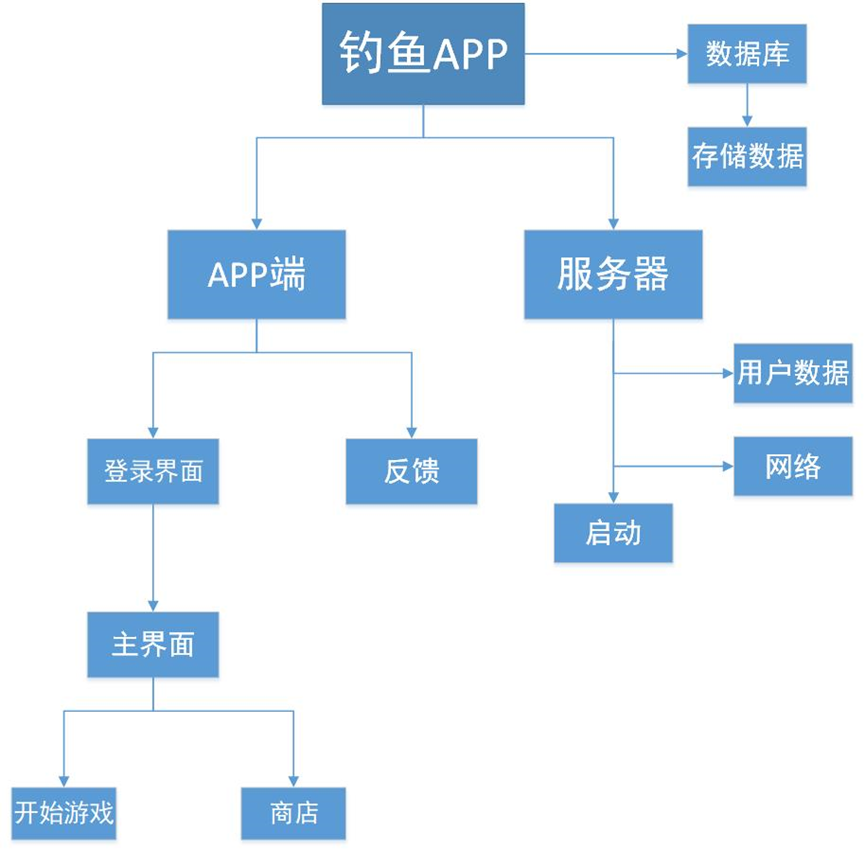
****

****

## 2.3状态转换图

****

## 2.4 程序方框图

****

# 三．钓鱼游戏设计说明

## 3.1程序描述

我们采用建立一个大的数据库的方式，将这些需要的数据都储存进去。游戏中的大部分功能则用UNITY3D引擎实现，并利用这个引擎进行大量建模。如果unity不支持一些建模，则用photoshop、flash等来实现。

## 3.2功能

以第一人称视角呈现，并区分钓鱼方式，分为传统钓、悬坠钓、竞技钓、台钓等。同时本软件能实现包括钓鱼地点（池塘、水库、海洋等），钓鱼用品（鱼钩、鱼线、鱼竿等），鱼饵（蚯蚓、苍蝇、蛆等）的选择。如果允许的话我们还会加入充值系统，排行榜系统。

## 3.3性能

对程序的性能要求，能够在装有IOS系统或者安卓系统的基本上所有的手机使用。

## 3.4 输入项

系统名：”用户模块”

由下列模块调用：”主界面模块”

输入：用户名、密码、密码确认

输出：注册结果、更新用户信息主文件

处理内容：如果用户名已存在 然后提示：该用户名已被注册

如果用户名小于4字符或者大于 16字符 then 提示：非法用户名

如果用户名大于4字符以及小于16字符 then 提示：该用户名可用

如果密码小于6字符或者大于20字符 then提示：该密码不符合要求

如果密码确认与原密码不匹配 然后提示：两次输入密码不一致

内部数据：用户信息主文件

备注：无

制作人：郑鸿棣

制作时间：2019.04.05

## 3.5 输出项

TBD

## 3.6 算法

using UnityEngine;  
using System.Collections;  
using UnityEngine.UI;  
  
public class 按钮 : MonoBehaviour {  
 *//public GameObject Gmenue;*  
 public GameObject btnObj ;  
 public GameObject caidan;  
 public Sprite expan;  
 public Sprite back;  
 Button btn;  
 bool isshow=false ;  
 *// Use this for initialization*  
 void Start () {  
 caidan.SetActive (isshow);  
 btn = btnObj.GetComponent<Button>();  
 btn.onClick.AddListener(delegate ()  
 {  
 isshow=!isshow;  
 caidan.SetActive (isshow);  
 if (isshow)  
 {  
 btn.GetComponent<Image>().sprite=expan ;  
 }  
 else {  
 btn.GetComponent<Image>().sprite=back;  
 }  
 } );  
 }  
  
 *// Update is called once per frame*  
 void Update () {  
  
 }  
}

using System.Collections;  
using System.Collections.Generic;  
using UnityEngine;  
using UnityEngine.UI;  
  
public class 转跳界面 : MonoBehaviour {  
  
 public Button mButton;  
  
 *// Use this for initialization*  
 void Start () {  
 *//Gets ButtonMount*  
 Button btnMount = mButton.GetComponent<Button>();  
 *//add a listener to ButtonMount, executing TaskOnClick() when click ButtonMount*  
 btnMount.onClick.AddListener(TaskOnClick);  
 }  
  
 void TaskOnClick()  
 {  
 *//Loading Scene1*  
 UnityEngine.SceneManagement.SceneManager.LoadScene(x);  
 }  
  
}

using System.Collections;  
using System.Collections.Generic;  
using UnityEngine;  
  
public class 转跳弹窗 : MonoBehaviour {  
  
 *// Use this for initialization*  
 void Start () {  
   
 }  
   
 *// Update is called once per frame*  
 void Update () {  
   
 }  
 public void Click\_test(){  
 UnityEditor.EditorUtility.DisplayDialog("提示", "该内容尚未开发，尽情期待", "确认");  
 }  
}

## 3.7 接口

mysql

## 3.8储存分配

对于钓到的鱼的质量以及使用的渔具和鱼饵储存在用户的手机上，若有排行榜功能，我们则会建立一个服务器和数据库，使得数据储存在上面。

## 3.9注释设计

地图：用unity3d的terrain组件，点击paint texture按钮，给“白模”的地形系统绘制特定的地形贴图纹理，再通过抬高地形，往里面注入水体，来达到制作成河流、海水、湖泊等地形。最后用trees按钮，添加一些树木以及花草在上面。用碰撞体来创建一个“空气墙”，不让用户走出我们设定的范围。

人物：导入characeters角色包，选择第一人称，把主摄像机当作角色的眼睛。

鱼竿、鱼饵、鱼类：在网上搜索各自的包，将他们导入，再自己建模来补充一些

钓鱼功能：用手机的平衡模拟器模拟甩杆，用点击来控制力度使其在最佳范围内。

UI界面：创建UGUI界面。再用canvas创建画布，用eventsystem（事件管理）、panel（整体移动）、text（文字）、image（图像）、button（按钮，使其动起来）控件完善

## 3.10限制问题

因手机差异，可能会导致一些功能出现问题

## 3.11 尚未解决的问题

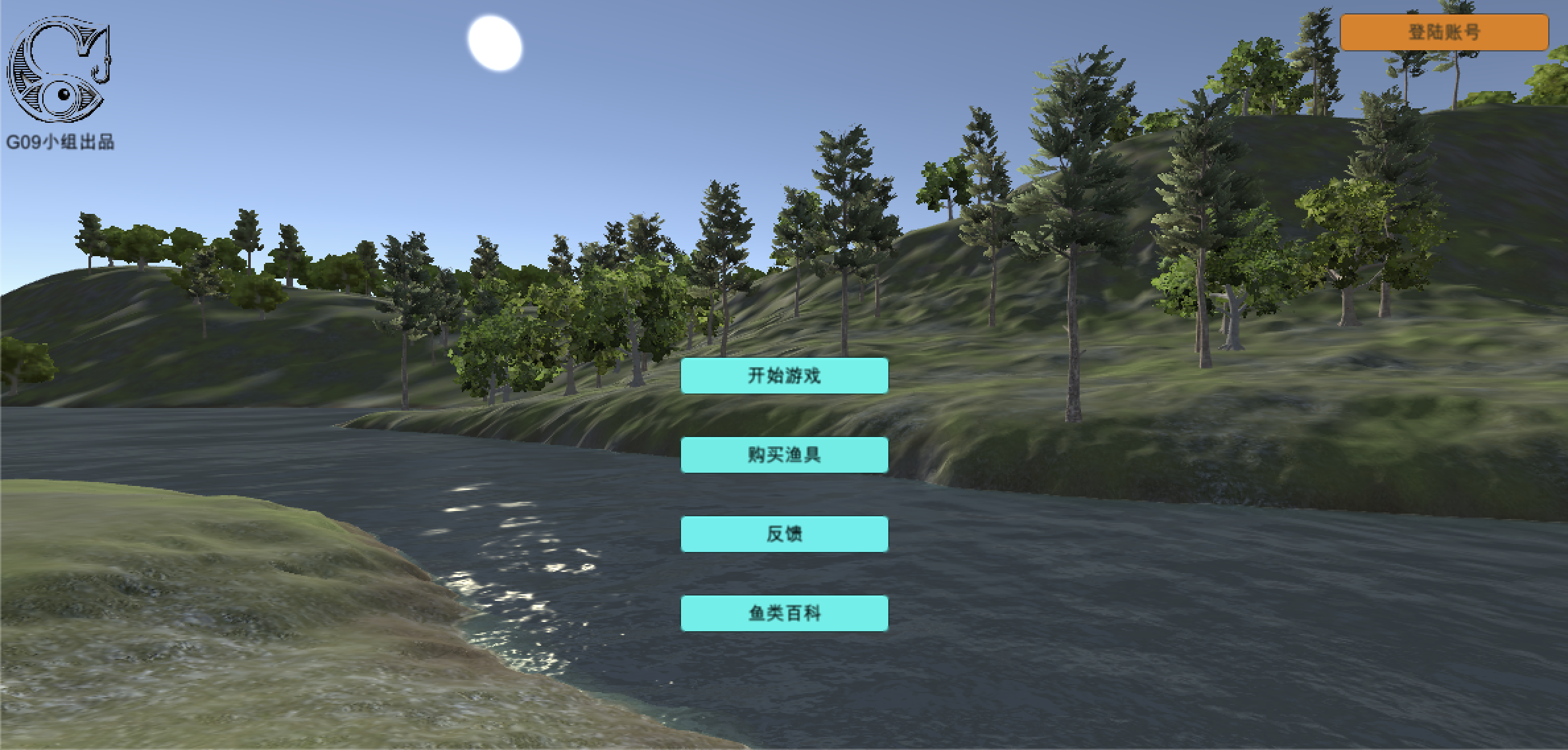
还未能完全实现钓鱼功能

建立的模型还不够

# 四．项目计划展示

# 预计3个月完成该项目

# 五．界面原型展示

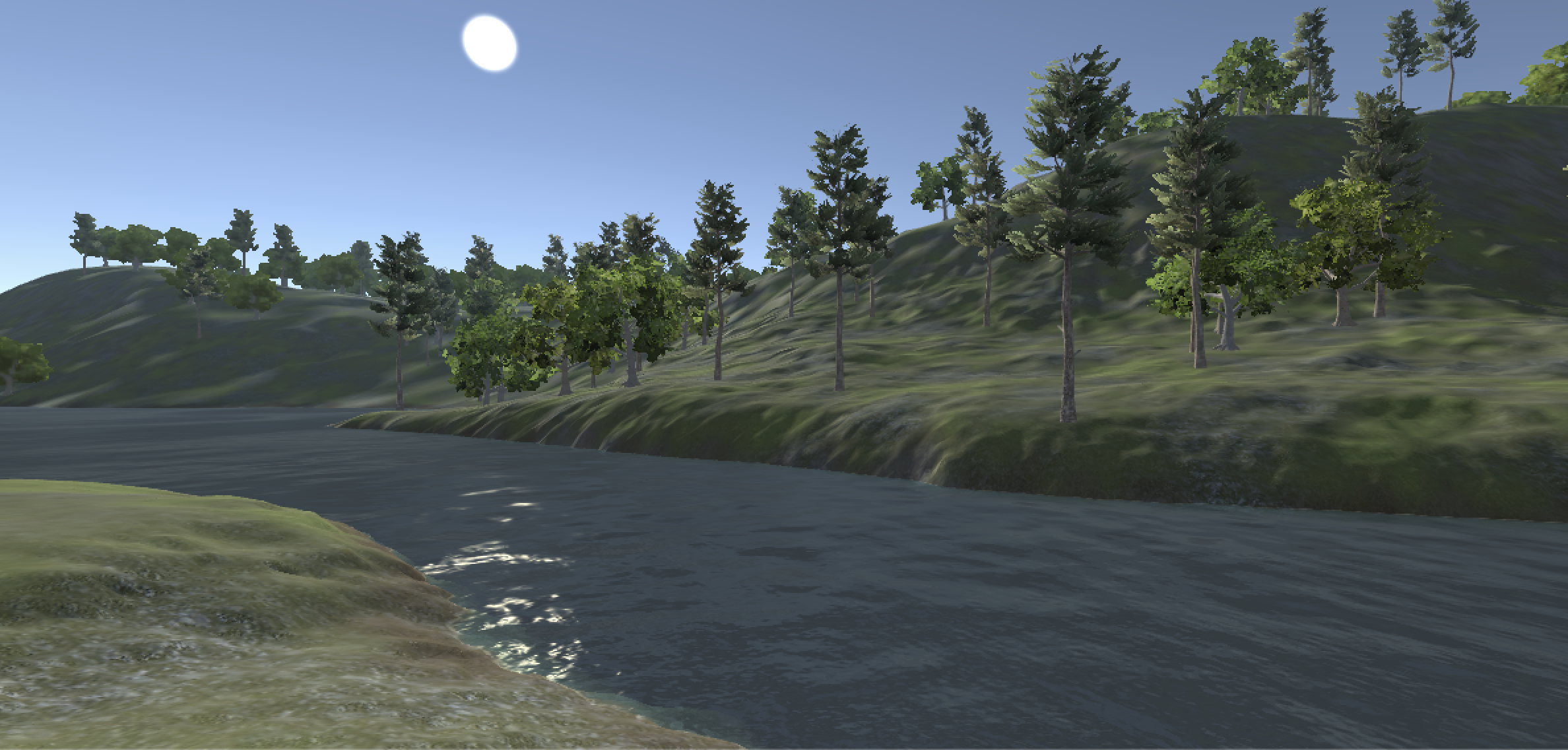
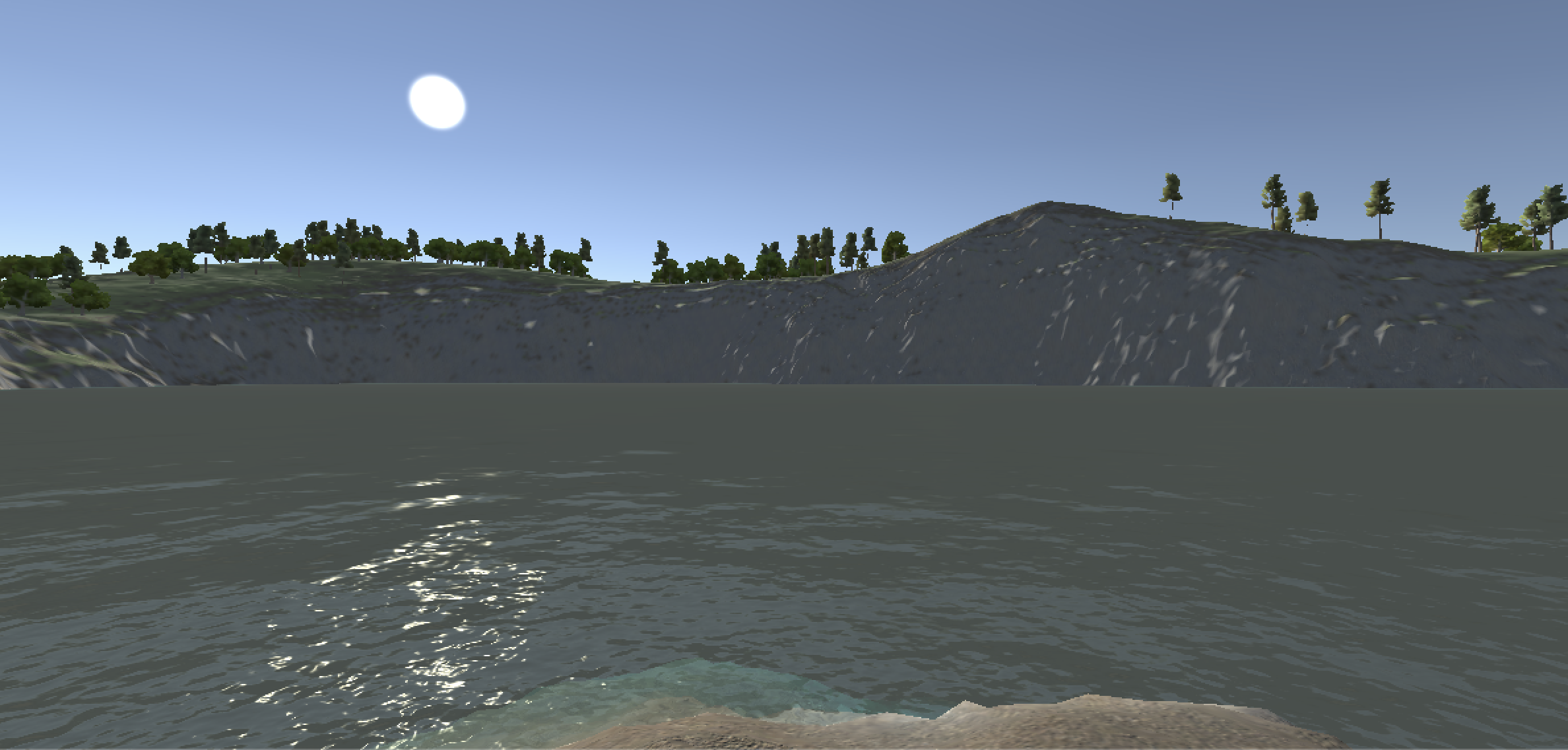
初始界面

购买渔具

反馈

鱼类百科

TBD

钓鱼场所（初始/进阶）

# 六．软件测试

大致测试流程：在完成这个软件之后，自己先进行跳水，之后再寻找多个不同用户以及多个不同的手机来使用这个软件，让他们尽量走遍地图，并尝试在不同的位置钓鱼。测试这个软件同时能够游玩的人数极限，以及一些未发现的问题，还有我们设计的钓鱼功能是否合理，操作起来是否变扭。

根据杨老师的说法，我们将会由并不是做某一块功能的组员来测试这一块功能，以此更好地模拟用户测试。测试的范围：1.建立的模型是否会出现穿模等问题 2.用户能否甩杆到他们想要的位置 3.钓上的鱼几率是否过低或者过高 4.钓鱼的操作是否过于繁琐

# 七．软件维护

建立一个维护组织，也就是将不同的部分交给分工在这个部分的人责任。随后确定报告和评价的过程，为每个维护要求规定一个标准化的事件序列，并建立一个适用于维护活动的记录保管过程，记录下具体维护了什么，最后评价这个维护活动。