 **百日高考项目文档**

——对对对队对对队

2019.5.11

目录

[百日高考项目文档 1](#_Toc8484283)

[一、游戏用例 3](#_Toc8484284)

[1. 游戏背景 3](#_Toc8484285)

[2. 游戏机制 3](#_Toc8484286)

[时间机制 3](#_Toc8484287)

[存档机制 3](#_Toc8484288)

[科目机制 3](#_Toc8484289)

[精力/压力机制 3](#_Toc8484290)

[刷题机制 3](#_Toc8484291)

[题目对抗机制（暂未实现） 4](#_Toc8484292)

[交友机制（暂未实现） 4](#_Toc8484293)

[状态机制 4](#_Toc8484294)

[统计机制 4](#_Toc8484295)

[3.用例描述 4](#_Toc8484296)

[1）进入游戏 4](#_Toc8484297)

[2）玩游戏 4](#_Toc8484298)

[3.）选择分科 4](#_Toc8484299)

[4） 摸底考 5](#_Toc8484300)

[5）刷题 5](#_Toc8484301)

[6） 高考 5](#_Toc8484302)

[7） 毕业 6](#_Toc8484303)

[8） 自动存档 6](#_Toc8484304)

[二、UML分析设计 7](#_Toc8484305)

[1. 用例图 7](#_Toc8484306)

[2. 概念类图 7](#_Toc8484307)

[3. 时序图 8](#_Toc8484308)

[4. 类图 10](#_Toc8484309)

一、游戏用例

## 1. 游戏背景

假设你是一名高三的学生，现在距离高考还有一百天，你为了考上一所好大学，又不得不面临升学的压力。

## 2. 游戏机制

### 时间机制

玩家从距离高考还有100天时开始游戏。

在特定条件下，玩家可以手动选择进入下一天。

### 存档机制

游戏一开始玩家可以选择读取旧存档或者新建存档

新建的存档需要玩家自己指定昵称，学校（也可以在系统提供的学校列表里面选择）

输入个人信息后需要选择文/理科，并进行摸底考试

### 科目机制

1. 玩家进入游戏时进行分科选择（文科/理科）随后进入摸底考试。

摸底考试总共六个科目共计十八道题，结果将用作为玩家生成题目等功能。

2. 根据玩家分科结果，分配到不同类型的科目：

文科玩家科目：语文、数学（文）、英语、政治、历史、地理；

理科玩家科目：语文、数学（理）、英语、物理、化学、生物。

### 精力/压力机制

1. 每日初始精力值为100，做任何事情都会减少精力值。精力可能因特殊事件发生加减（暂不考虑），精力小于10时玩家可以选择进入下一天。

2. 初始压力值为0，刷题会增加压力值，如果压力值到达一定量刷题功能会被限制，此时玩家需要通过玩游戏，交友等功能减少压力值。

### 刷题机制

1.题目分为6个等级，每次刷题会根据题目星级减少精力。

2. 题目类型为选择题，题目数据主要从各大学习平台中获取。

3. 刷题需要玩家指定科目,刷题模式，刷题量，系统将根据该科目的科目点数来决定抽取题目难度的比率

### 题目对抗机制（暂未实现）

玩家可以与其他玩家进行答题比拼

### 交友机制（暂未实现）

### 状态机制

显示玩家的当前状态和相关信息，比如：玩家名称，学校，精力值，压力值，上次考试成绩，距离考试时间，并以统计图的形式显示出玩家的成绩在不同科目上的分布情况。

### 统计机制

根据玩家平时的刷题/考试情况，后台自动进行汇总和整理（正确率，题目解锁数量，平均耗时），并以可视化的形式呈现。

## 3.用例描述

### 1）进入游戏

**执行者：**玩家

**目的：**通过新建游戏或者读取存档进入游戏

**事件流：**

1. 玩家启动游戏；
2. 玩家选择“新建游戏”或者“读取存档”（如果玩家有存档）；
3. 如果玩家选择“新建游戏”，将要求玩家选择一个档位进行新游戏，并执行以下4~6步骤；
4. 玩家选择分科；
5. 进行摸底考；
6. 开始玩游戏；
7. 如果玩家选择“读取存档”，玩家将从选择的存档读取游戏。

**前置条件：**一个想在游戏中学习的人

**后置条件：**成功进入游戏

### 2）玩游戏

**执行者：**玩家

**目的：**通过玩游戏进行学习

**事件流：**

1. 在游戏界面中，玩家可以选择刷题等活动；
2. 当精力值低于10时，玩家可以手动选择进入下一天；

**前置条件：**玩家进入游戏

**后置条件：**获得知识与快乐

### 3.）选择分科

**执行者：**玩家

**目的：**选择游戏人物的文理分科

**事件流：**

1. 进入新游戏，进入分科选择界面；
2. 玩家选择“文科”或“理科”；
3. 玩家确认选择；
4. 分科选择完毕，用例结束。

**前置条件：**玩家进入新游戏

**后置条件：**分科选择成功，游戏人物拥有相应科目的属性

### 4） 摸底考

**执行者：**玩家、系统

**目的：**进行游戏人物的摸底考，快速上手游戏

**事件流：**

1. 分科选择完毕后自动进入摸底考界面；
2. 系统根据给定的顺序展示每个科目的题目；
3. 玩家完成并提交当前科目，进入下一科目；
4. 玩家最后一个科目提交后，摸底考结束；
5. 系统根据做题情况展示摸底考分数；
6. 系统生成科目点数并记录到后台；
7. 玩家确认分数，用例结束。

**前置条件：**玩家完成分科选择

**后置条件：**玩家得知摸底考分数，系统生成游戏人物初始科目点数

### 5）刷题

**执行者：**玩家、系统

**目的：**做题目

**事件流：**

1. 玩家选择刷题，进入刷题界面；
2. 在刷题界面中，玩家选择科目；
3. 系统通过一定算法给出一道题目显示在刷题界面中；
4. 玩家做题并选择一个答案；
5. 答案正确，执行刷题奖励（如果有）；
6. 答案错误，执行刷题惩罚（如果有）；
7. 作答完毕，界面显示作答结果与题目解析，系统处理相关事件；
8. 在结果界面中，玩家可选择新一轮刷题；
9. 选择新一轮刷题，回到步骤2；
10. 玩家结束刷题，用例结束。

**前置条件：**玩家选择刷题

**后置条件：**玩家通过刷题有所收获，游戏人物属性改变

### 6）考试

**执行者**：玩家、系统

**目的：**通过分阶段的月考/期中考来督促玩家，提醒玩家学习，测试玩家进步情况

**事件流：**

1. 把一百天划分为三个阶段：100-70，69-40，39-10；

2. 每个阶段结束后系统会根据前一阶段的学习成果生成月考题目；

3. 按照语文，数学，理综/文综，外语的顺序生成题目，

玩家完成并提交当前科目，进入下一科目；

4. 玩家最后一个科目提交后，月考结束；

5. 系统根据做题情况生成月考分数；

6. 系统记录月考分数；

7. 玩家确认分数

**前置条件：**时间达到月考的时间

**后置条件：**游戏统计信息，生成考试分数

### 7） 高考

**执行者：**玩家、系统

**目的：**完成游戏的最终阶段

**事件流：**

1. 时间到达高考前第0天，系统自动进入高考界面；
2. 系统根据给定的顺序展示每个科目的题目；
3. 按照语文，数学，理综/文综，外语的顺序生成题目‘

玩家完成并提交当前科目，进入下一科目；

1. 玩家最后一个科目提交后，高考结束；
2. 系统根据做题情况与科目点数生成高考分数；
3. 系统记录高考分数；
4. 玩家确认分数，进入毕业界面，用例结束。

**前置条件：**时间到达高考前第0天

**后置条件：**游戏进入毕业界面，系统记录高考分数

### 8） 毕业

**执行者：**玩家、系统

**目的：**得知游戏最终结果

**事件流：**

1. 高考界面结束，进入毕业界面；
2. 系统根据高考分数随机分配录取院校；
3. 系统记录录取院校；
4. 播放游戏结束动画；
5. 游戏结束，返回标题菜单，用例结束。

**前置条件：**高考完毕

**后置条件：**游戏结束，系统记录录取院校

### 9） 自动存档

**执行者：**系统

**目的：**自动存档当前游戏进度

**事件流：**

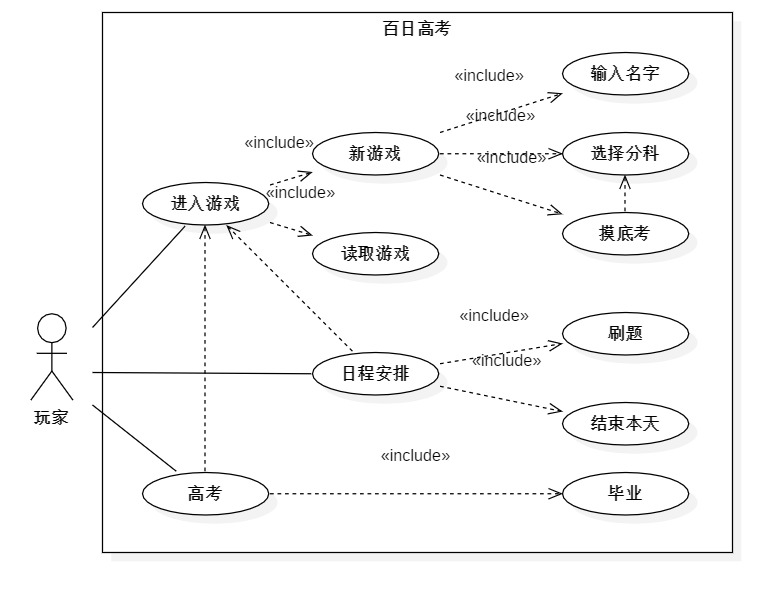
1. 在每一天的开始时刻，自动保存游戏进度到玩家选择的档位上；
2. 存档成功，用例结束；
3. 存档失败，告知玩家存档失败原因。

**前置条件：**玩家手动进入下一天

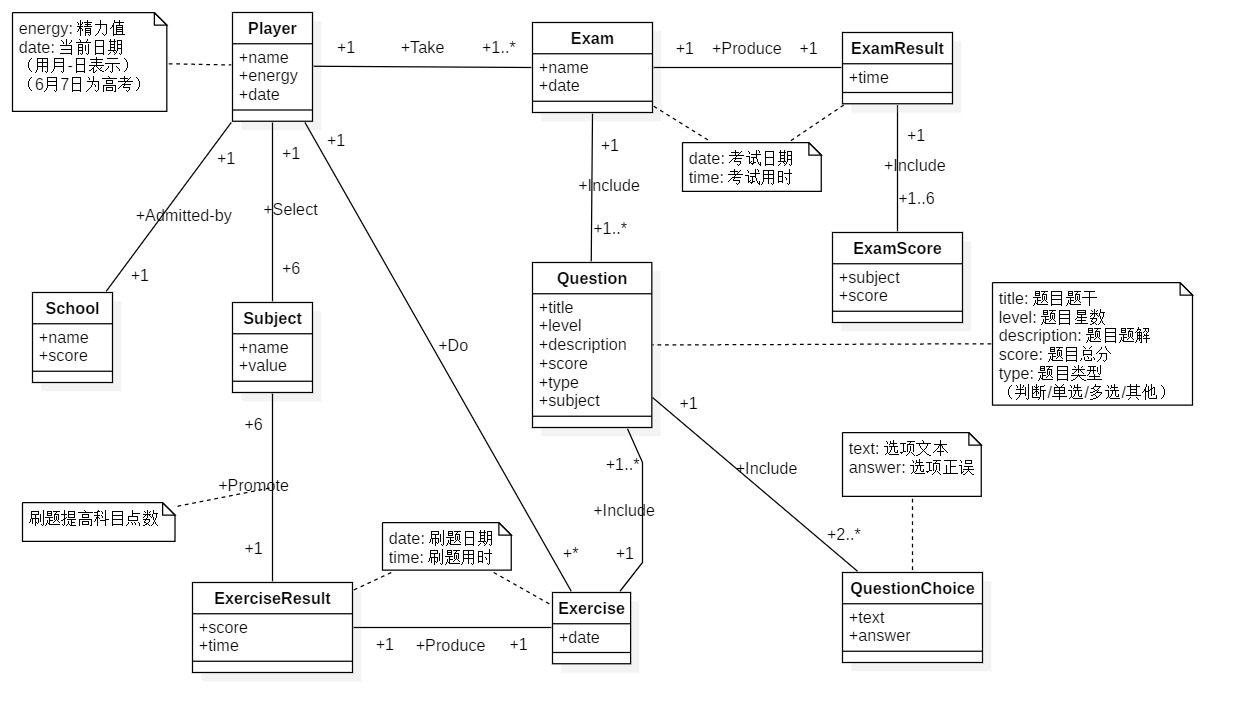
**后置条件：**存档成功或提示玩家存档失败原因

二、UML分析设计

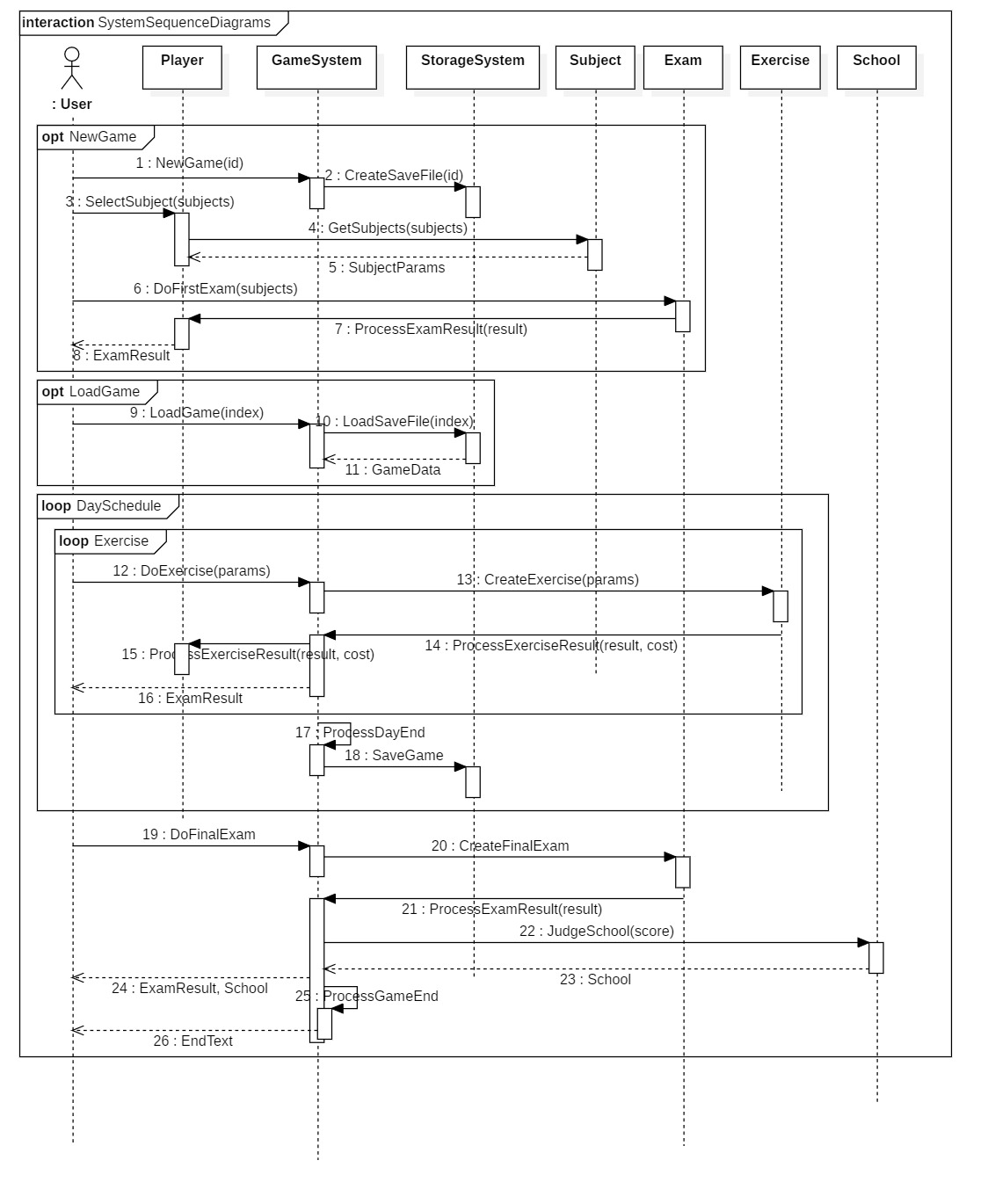
## 1. 用例图



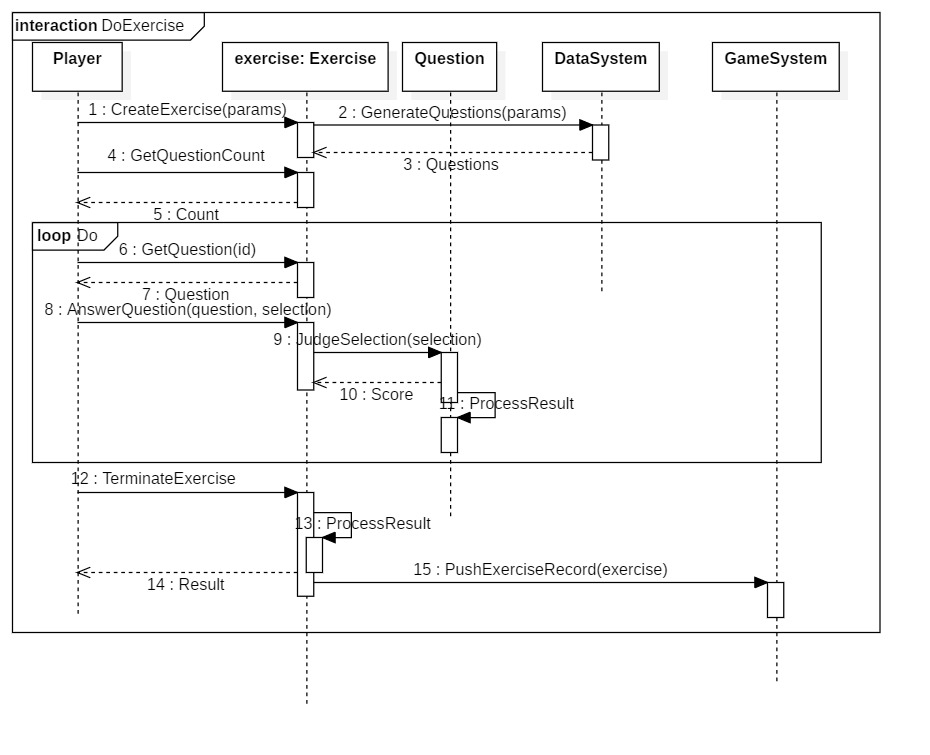
## 2. 概念类图



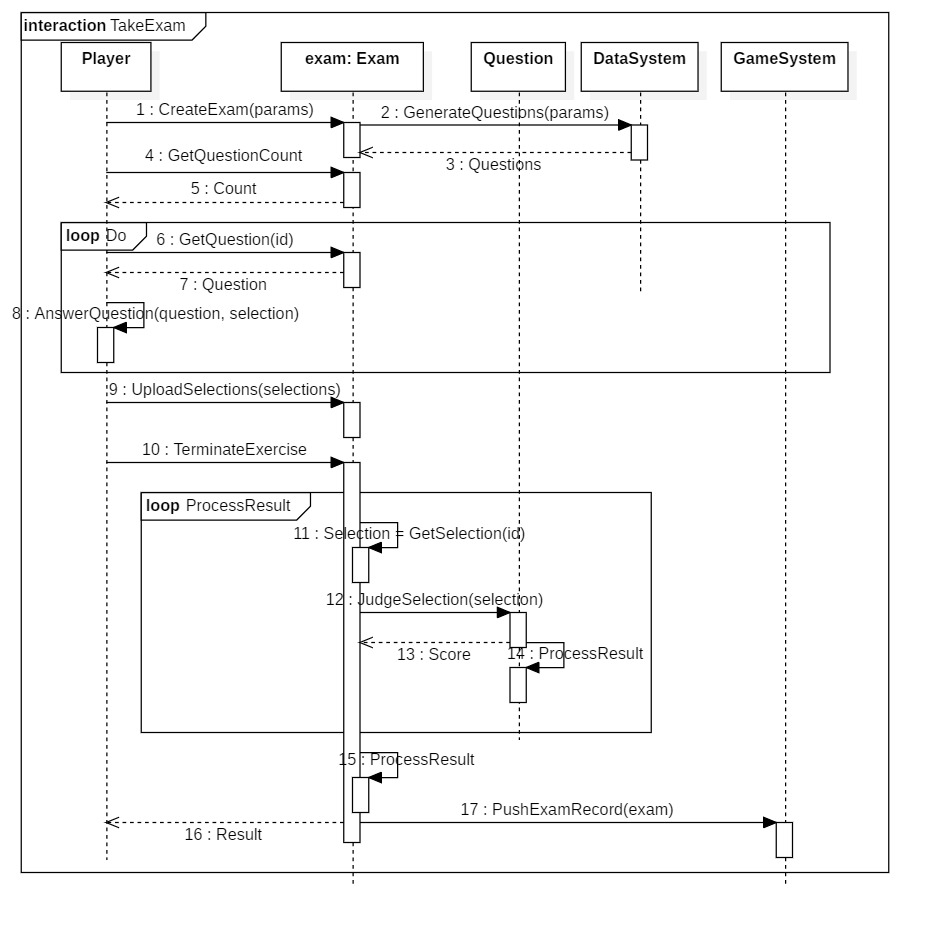
## 3. 时序图



刷题时序图：



考试时序图：



## 4. 类图

