信息科学技术学院实验报告

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **课程名称** | | **UML与可视化建模** | | | | **班级** | | **16软工4班** |
| **实验名称** | | **UML综合建模大作业** | | | | **教导教师** | | **曾少宁** |
| **姓名** | **占城岚** | | | **学号** | **1614080902411** | | **日期** | **2019.4.28** |
|  |  | |  | | |  | |  | |

**引言**

本次UML课程中，我所做项目，为自拟游戏“学院制裁”。这是一个RPG游戏。游戏战斗系统为PVE。开发这个游戏的目的就是为了吸引挑战技巧类人群来玩，从而获得成就感。本游戏主要新在PVE模式，在现有的游戏下，国内很少出现这种RPG类技能使用PVE游戏战斗系统，即使有，作为代表的也只是卡牌或者RougueLike。

a）该游戏中RPG主要分两大类角色。其一为学生角色，其二为zonbi类角色

b) 无论哪种角色都拥有两种技能。对于学生角色，技能其一为逃离，使用时间为1分钟，冷却时间为3分钟，使用时，加快玩家行动正常速度的3倍。技能二为变身，使用时间为1分钟，冷却时间为2分钟，使用时，玩家自行选择变为zonbi类角色的一类，最终变身。

c) 对于zonbi类角色，技能一为隐身，使用时间为1分半，冷却时间为2分钟，使用时，任何情况下，玩家隐身都不会被其他角色发现。技能二为融化，使用时间为2分钟，冷却时间为1分钟，使用时，玩家角色变形为一摊液体，能够进入任何缝隙中。

**一、需求模型**

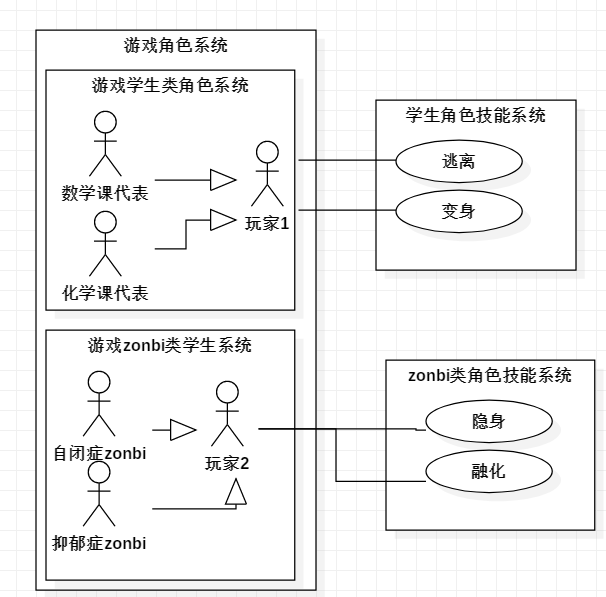


图1：学院制裁游戏技能系统用例图

|  |  |
| --- | --- |
| 用例编号： |  |
| 用例名称： | 逃离 |
| 用例描述： | 玩家可以点击该技能，加快行动速度 |
| 前置条件： | 玩家已进入战斗、且技能冷却时间为0s |
| 基本流程： | 1. 玩家点击‘逃离’技能按钮。 2. 系统检查该技能按钮点击，创建一条使用记录，修改玩家当前速度记录，修改技能冷却时间记录（从180s递减）。 3. 技能系统检查到原记录正常改变，返回3倍速度记录，返回系统确认信息。 4. 系统检查到该技能冷却时间记录为0s，显示玩家3倍速度，显示不可以使用该技能。 |
| 扩展流程： | 1、 系统检查到该技能点击， 创建一条BUG记录。  2、系统检查到该技能冷却时间记录为0s，玩家界面显示可以使用该技能，系统重新检查该技能按钮点击。 |
| 后置条件： |  |

|  |  |
| --- | --- |
| 用例编号： |  |
| 用例名称： | 变身 |
| 用例描述： | 玩家可以点击该技能，玩家选择变为zonbi类角色的一种 |
| 前置条件： | 玩家已进入战斗、且技能冷却时间为0s |
| 基本流程： | 1. 玩家点击‘变身’技能按钮。 2. 创建一条使用记录，玩家选择变身为zonbi类角色的一种，修改玩家当前角色模型记录。 3. 系统检查到原记录正常改变，返回系统确认信息。 4. 系统检查到该技能冷却时间记录为0s，显示玩家当前角色模型，显示不可以使用技能。 |
| 扩展流程： | 1、系统检查到原记录正常改变，创建一条BUG记录。  2、系统检查到该技能冷却时间记录为0s，玩家界面显示可以使用该技能，系统重新检查该技能按钮点击。 |
| 后置条件： |  |

|  |  |
| --- | --- |
| 用例编号： |  |
| 用例名称： | 隐身 |
| 用例描述： | 玩家可以点击该技能，玩家角色将不出现在其他玩家界面 |
| 前置条件： | 玩家已进入战斗、且技能冷却时间为0s |
| 基本流程： | 1. 玩家点击‘隐身’技能按钮。 2. 创建一条使用记录，修改玩家当前角色属性状态记录，系统检查该技能按钮点击，检查到原记录正常改变，返回当前角色属性状态修改的记录，返回确认信息。 3. 系统检查到该技能冷却时间记录为0s，显示对其他类玩家不可见状态，显示不可以使用该技能。 |
| 扩展流程： | 1、系统检查到原记录正常改变，创建一条BUG记录。  2、系统检查到该技能冷却时间记录为0s，玩家界面显示可以使用该技能，系统重新检查该技能按钮点击。 |
| 后置条件： |  |

|  |  |
| --- | --- |
| 用例编号： |  |
| 用例名称： | 融化 |
| 用例描述： | 玩家可以点击该技能，玩家角色将变为一摊液体，可以进入任何缝隙，技能使用结束则变回原先的角色模型。 |
| 前置条件： | 玩家已进入战斗、且技能冷却时间为0s |
| 基本流程： | 1. 玩家点击‘融化’技能按钮。 2. 创建一条使用记录，修改玩家当前角色属性状态。 3. 技能系统检查到原记录正常改变，返回修改玩家当前角色属性状态，返回系统确认信息。 4. 系统检查到该技能冷却时间记录为0s，显示玩家当前角色模型为一滩液体，显示不可以使用该技能。 |
| 扩展流程： | 1、系统检查到原记录正常改变，创建一条BUG记录。  2、系统检查到该技能冷却时间记录为0s，玩家界面显示可以使用该技能，系统重新检查该技能按钮点击。 |
| 后置条件： |  |

**二、过程模型**

****

图1：逃离技能活动图

该逃离技能玩家使用后能够获得维持60s的三倍速度。活动图表现的是该技能的实现机制。



图2：变身技能活动图

该变身技能玩家使用后能够获得维持60s的其他角色模型包括属性、方法以该模型的技能等等。活动图表现的是该技能的机制。

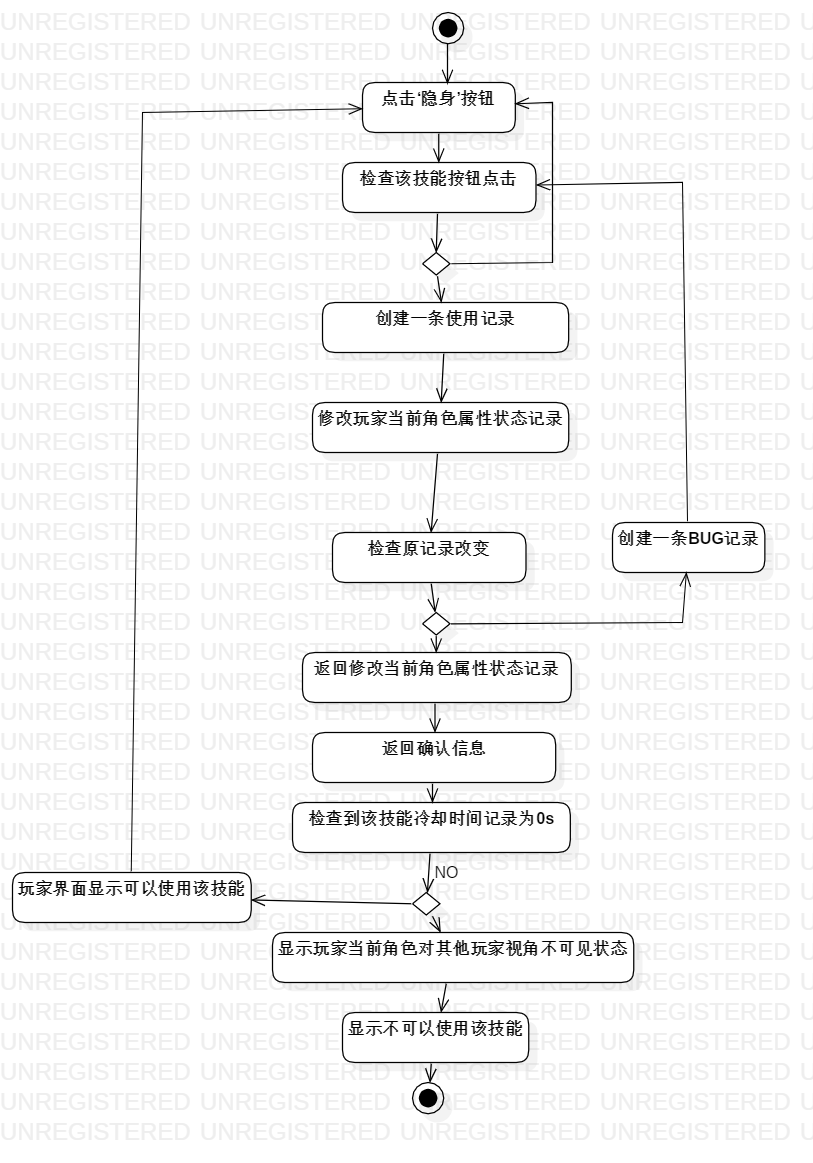


图3：隐身技能活动图

该隐身技能玩家使用后能够获得维持60s的隐身效果，即在其他玩家界面内60s不可见。活动图表现的是该技能的机制。

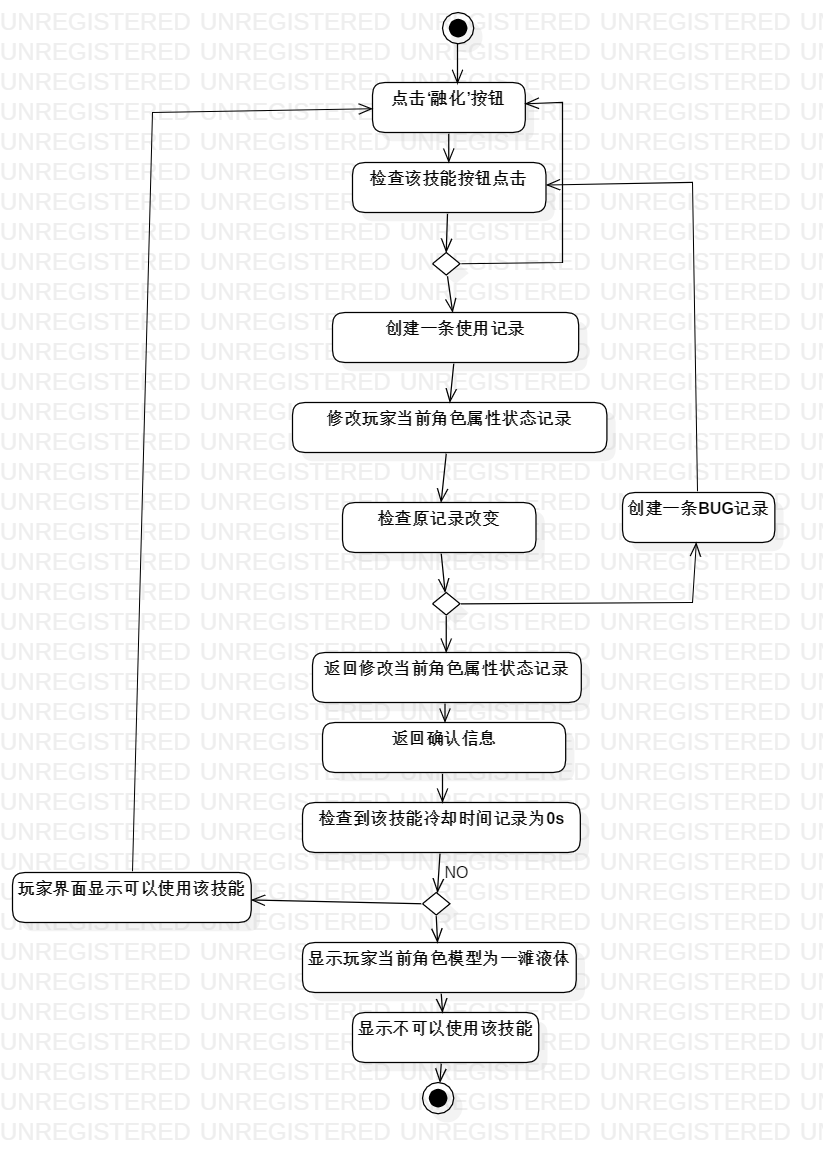


图4：融化技能活动图

该融化技能玩家使用后能够获得维持120s的角色形态表现为液体的角色状态模型。活动图表现的是该技能的机制。

**三、结构模型**

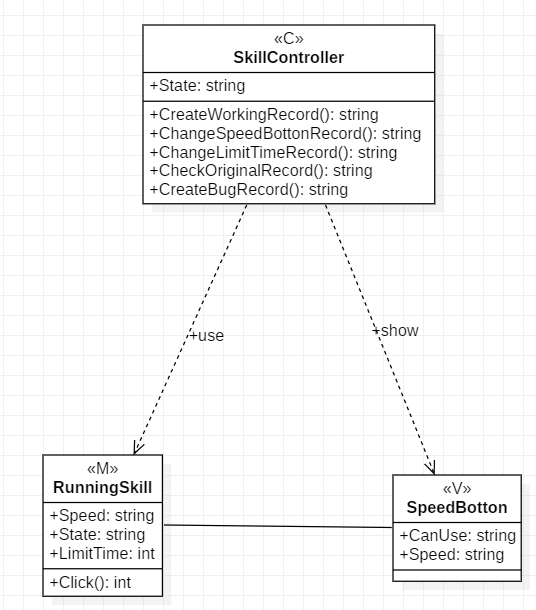


图1：逃离技能类图

该逃离技能的类图根据MVC建模设计出的技能系统该有的技能控制器类C：控制技能类/技能按钮类。可视化类V：速度按钮类。真正实现操作的模型类M:速度技能类。

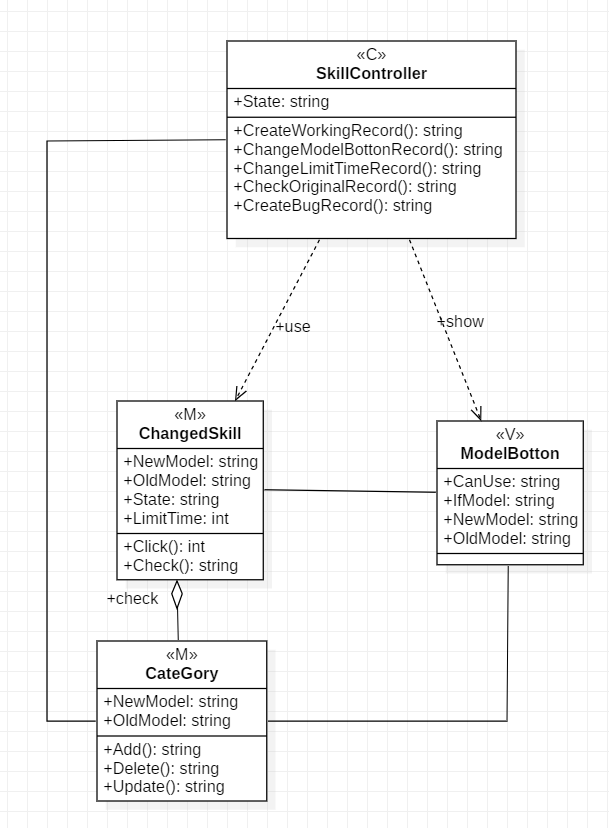


图2：变身技能类图

该变身技能的类图有技能控制器类C：控制技能类/技能按钮类。可视化类V：模型技能按钮类。真正实现操作的模型类M:修改技能类。子模型类M：游戏角色类。

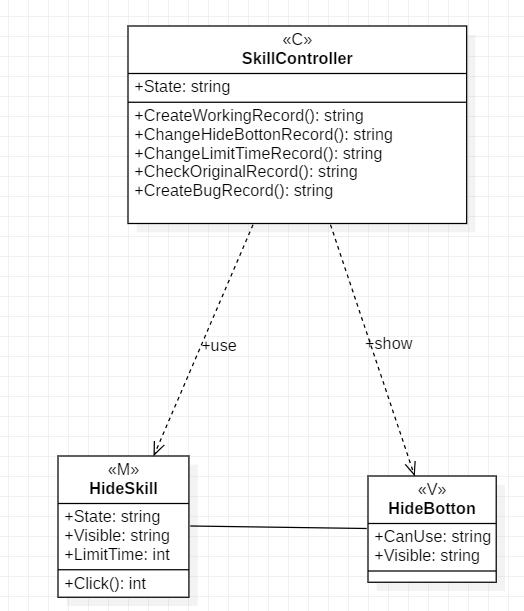


图3：隐身技能类图

该隐身技能的类图有技能控制器类C：控制技能类/技能按钮类。可视化类V：隐身技能按钮类。真正实现操作的模型类M:隐藏技能类。

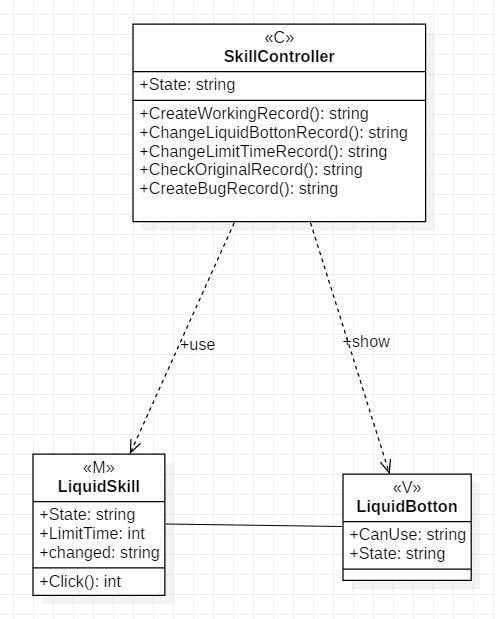


图4：融化技能类图

该融化技能的类图有技能控制器类C：控制技能类/技能按钮类。可视化类V：液体技能按钮类。真正实现操作的模型类M:液体技能类。

**四、交互模型**

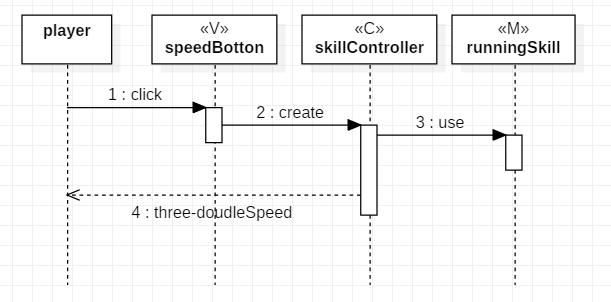
****

图1：逃离技能顺序图

该逃离技能顺序图表现了该技能使用后玩家角色获得3倍速度的从玩家到系统的整体时间流程

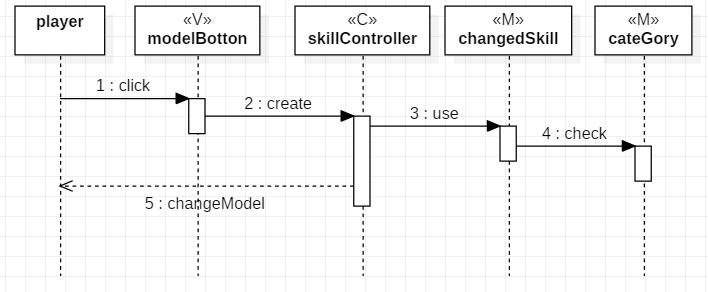


图2：变身技能顺序图

该变身技能顺序图表现了该技能使用后玩家角色变身从玩家到系统的整体时间流程

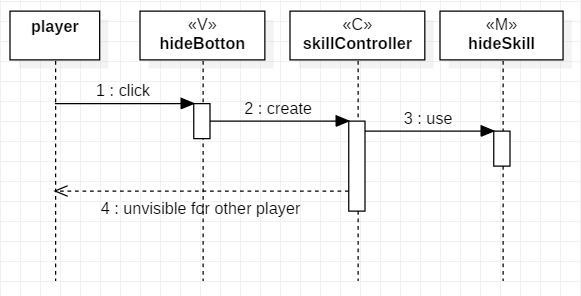


图3：隐身技能顺序图

该隐身技能顺序图表现了该技能使用后玩家角色隐身从玩家到系统的整体时间流程

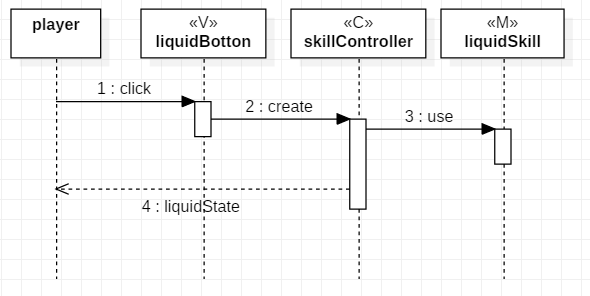


图4：融化技能顺序图

该融化技能顺序图表现了该使用后玩家角色融化从玩家到系统的整体时间流程

**五、状态模型**

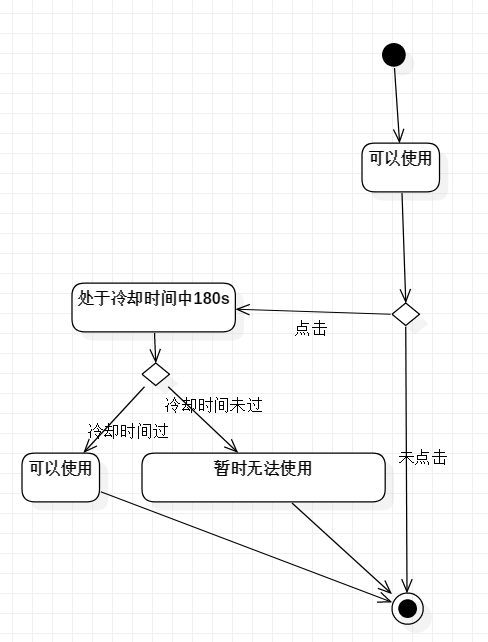
****

图1：逃离技能状态图

该逃离技能状态图展现技能在玩家使用到系统实现整个流程中，逃离技能会有哪些状态。

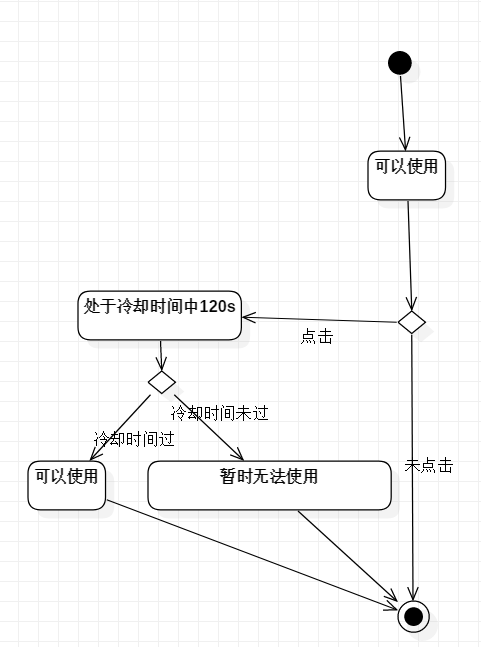


图2：变身技能状态图

该变身技能状态图展现技能在玩家使用到系统实现整个流程中，变身技能会有哪些状态。

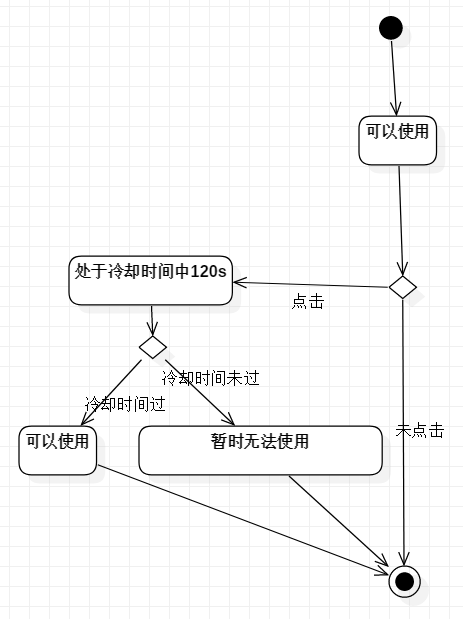


图3：隐身技能状态图

该隐身技能状态图展现技能在玩家使用到系统实现整个流程中，隐身技能会有哪些状态。

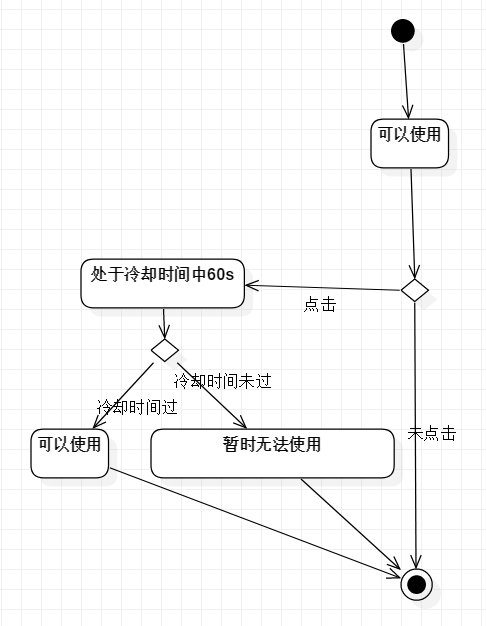


图4：融化技能状态图

该融化技能状态图展现技能在玩家使用到系统实现整个流程中，融化技能会有哪些状态。

**总结**

在UML实验课程中，我学会了StarUML软件的用法，而且在不断实验中，更加熟练。此外，还学会了最关键的UML建模方法，让我深刻的体验到了UML建模的重要性。

在整个实验过程中，我们首先接触了UML建模一些简单知识以及它的作用，是用来干嘛的，课后，出于自身对UML的兴趣，我去网上查阅了许多的知识，发觉自己对UML的了解还不够透彻，很多UML的知识都无法深刻的掌握。比如UML的顺序图，其实还有另外的画法，而且还有很多的类与类间的关系，在课堂上并没有做相应的拓展：如接口类的关系、聚合，泛化，组合等等。再深入的挖掘下去，就会涉及到真正的底层机制，例如：编程中如何实现类与类之间的这些关系，比如泛化就是继承！

其次在学习画类图这节课中，又接触到了新的知识MVC模型，关于这个模型我也搜索了很多的资料，不过对于只是了解UML而已，MVC模型我也没有做更多深层次的了解，只是大概了解了一下它所谓的三层结构。

另外在自己的职业规划中，我发现MVC模型和游戏设计中涉及到的MDA模型十分的相识，不是说内容，而是从三层结构上来讲。

最后，我认为对于建模和建模方法等知识，在学生阶段是十分有必要接触的，先不说掌握，因为建模方法在实战中，是需要经验的，而对于我们没有社会项目经验的大学生来讲，能够熟悉他们的基础知识和实用方法，就已经很不错了。作为后续的学习，我认为，我可以利用UML的建模，更加完善我的游戏设计。