

2023春季学期北京大学《程序设计实习》

QT大作业结项报告

25号 3W项目组

汪之立 吴乐鸿 吴亮宇



1 程序功能介绍

《Escape from Deadline》是我们设计的一款角色扮演类休闲游戏。玩家将在游戏中扮演大一的攻城狮，展开一场与 P 大中最为凶猛可怖的怪兽——DDL 的殊死斗争。

游戏总体由分段剧情和三个关卡的休闲小游戏构成。关卡的总体目标都是抢在 DDL 之前完成，也即吃掉图标，但是各关的玩法不同。第一关是在迷宫地图中通过键盘控制角色移动，赶在 DDL 线扫过前完成任务。第二关是鼠标拖拽角色完成任务同时防止被 DDL 追上。第三关是鼠标引导角色躲避 DDL 发出的各种子弹，完成夹杂其中的任务击败 DDL。地图上也会刷新各类 buff 或 debuff，进一步提高趣味性。游戏失败也不要灰心，会解锁有趣的支线结局。集全所有结局可以开启番外篇剧情。

音乐音效上我们精心挑选了电子风的背景音和多样的音效，丰富玩家的游玩体验。除了关卡外，游戏设计了可以快速跳转的菜单页面和相应的存档，方便玩家多次游玩。设置页面集成了音乐音效的音量调节。此外，其中的重置进度的按钮玩家一定要慎重点击哦！



图一：游戏菜单界面

2 项目各模块和类设计细节

我们的项目主要由关卡地图和游戏窗口两部分组成。关卡地图部分负责三个关卡所有玩法的底层逻辑和顶层显示的实现，而游戏窗口部分就负责与玩家交互部分和额外功能的实现。两个部分之间的交互有继承关系的由相应函数接口实现，其余使用信号进行交互。

2.1 关卡地图的实现

2.1.1 GameMap 类（关卡地图基类）

基于三关共同的元素，我们以 GameMap 类作为关卡地图的基类，以 Figure 类作为攻城狮和敌人（Deadline）的基类，以 TaskBuff 类作为任务和加成的基类。代码实现中最关键的函数是 time_event 函数，通过与一个 QTimer 计时器连接，可以实现每隔很短的时间间隔调用一次该函数。time_event 函数包括：改变所有物体速度，判断物体之间的相遇（如玩家吃到 task，子弹击中玩家等），执行触发的效果（如玩家扣血），向上层窗口更新信息（如显示被吃掉的

task 名称), 判断游戏是否结束等。我们将 time_event 的触发间隔设定为 120ms, 但为了游戏的流畅性, 将移动所有物体的时间间隔缩小至 30ms。

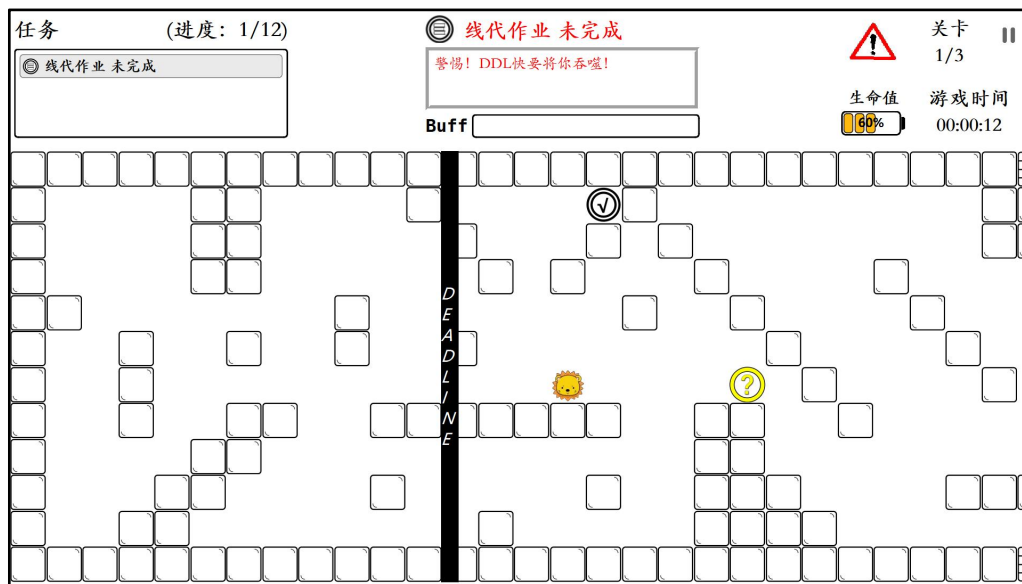
2.1.2 TaskBuffProcessor 类 (事件处理器类)

代码实现的第二个关键是实现各种效果的定时和延时触发, 如: 玩家吃到“原地停留 4 秒”的 debuff 时, 需延时 4 秒之后接触限制; 玩家受伤后图标会闪烁, 需 1 秒后停止闪烁; 部分 task 需要定时出现和消失等。为此, 我们为每个 taskbuff 写了两个成员函数: effect 和 end_effect, 分别对应立刻触发的效果和延时触发的效果。例如, 对于“原地停留 4 秒”的 debuff, 其 effect 函数会将玩家设置为不可移动, 而其 end_effect 函数则会将玩家重新设置为可移动。为了实现定时触发 end_effect, 我们写了 TaskBuffProcessor 类 (简称 Processor) 用于处理所有上述效果。每个 Processor 包含两个成员变量: 处理时间和指向 taskbuff 的指针。在 GameMap 中我们设置了一个 multiset 存放所有 Processor, 使它们按处理时间排序。每次调用 time_event 函数时, 我们会将当前时间与 multiset 中第一个 Processor 的处理时间比较, 如果当前时间小, 说明没有效果需要处理; 否则, 这个 Processor 对应的 taskbuff 的 end_effect 就会被触发。对于其他诸如闪烁、定时出现等效果, 我们会通过特殊的 Processor 用类似思路实现。

2.2.3 三个关卡实现

除了上述核心代码, 三关具体实现上各有不同。

第一关为迷宫式设计, 玩家通过键盘上下左右移动, 并需躲避 Deadline 的追赶。Deadline 会以一条竖线的形式从左到右移动, 表示时间的推移。第一关独有的技术要点在于, 由于地图宽度远大于屏幕宽度, 需要使用 sliding window 的方式让地图随角色左右移动。



图二：第一关示意

第二关玩家通过鼠标“点-拉”的方式给予玩家特定方向的速度, 而 Deadline 不再是一条竖线, 而是一个会追赶玩家的怪兽。第二关独有的技术要点包括玩家的速度衰减及 Deadline 追赶玩家的速度与方向的实时调整。



图三：第二关示意

第三关 Deadline 不直接追赶玩家，而是发射子弹攻击玩家，其中子弹的模式包括：1) 水平发射 2) 随机弹射 3) 瞄准玩家加速。玩家的移动方式变为用鼠标控制加速度，从玩家指向鼠标的向量决定了玩家加速度的大小和方向，因此表现出玩家具有“动量”，为游戏增添难度和趣味。第三关独有的技术要点包括：1) 子弹的发射、随机初始参数设定和移动 2) 加速度的实现 3) taskbuff 的移动。



图四：第三关示意

2.2 游戏存档的实现

在程序运行中，游戏数据由游戏数据类 Game 操控，依照玩家进度更改数据。Game 中包含三个 bool 型数组存储游戏进度和两个 double 变量保存音量信息。程序关闭时游戏数据存储在 statistics.json 文件中。游戏的读档发生在主窗口构造时，其余子窗口只有这一变量的引用作为成员变量，存档则是放在重载后的 closeEvent 函数中，基本杜绝了数据丢失或错误的可能。

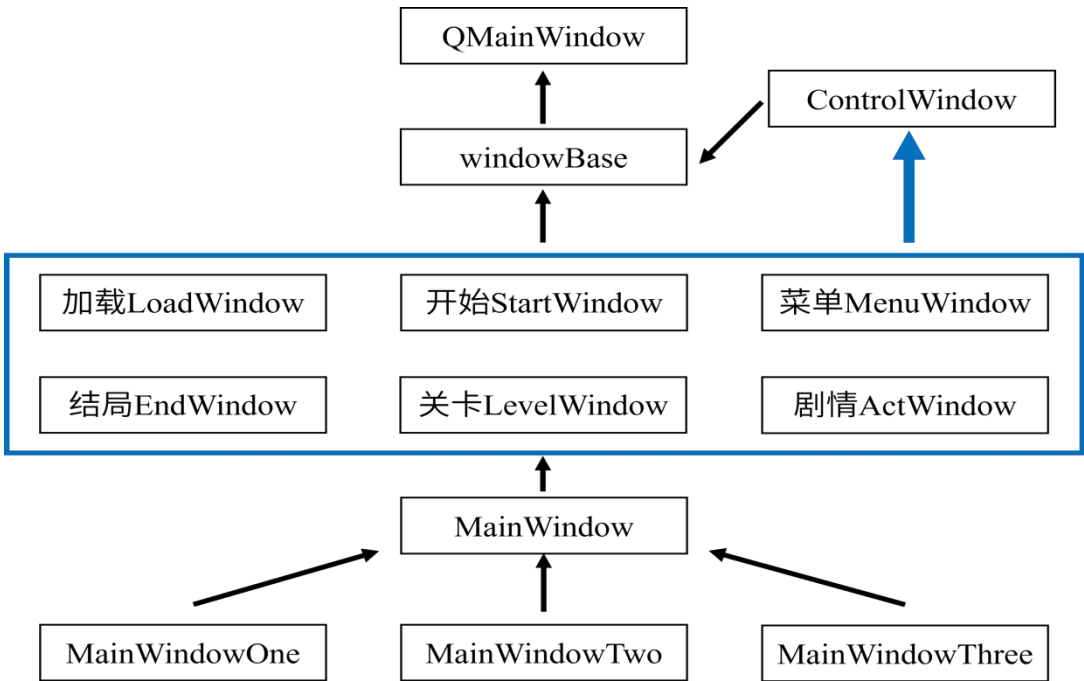
此外，我们规定了事件，也即地图刷新的任务或者 Buff\Debuff 的信息类 task_info。其记录了图标、名称、描述、突出显示四种属性，方便后续调用。

2.3 游戏窗口的实现

游戏窗口主要分为三种部件：窗口页面、对话框以及 CoverMask。

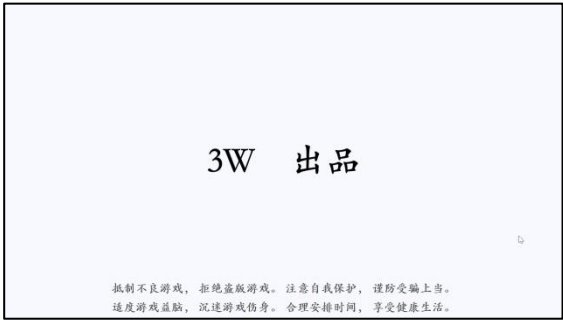
2.3.1 窗口页面

窗口页面是游戏各个主要界面的实现。由于我们的项目涉及较多不同的窗口页面，且 Qt 原本的 QMainWindow 不能满足频繁流畅的窗口切换需求，我们由 QMainWindow 派生出自己的游戏界面基类 windowBase，规定了窗口样式、部分窗口属性以及切换信号。具体的窗口页面都是从此派生而来，大多使用 ui 进行布局，QSS 样式表进行标准部件美化。图二展示了所有 7 类窗口页面的具体继承关系。接下来我们逐一进行介绍。



图五：页面继承关系图（细箭头代表继承关系，粗箭头代表包含对应的指针）

● LoadWindow（加载界面）&StartWindow（开始界面）



图六：LoadWindow 截图



图七：StartWindow 截图

LoadWindow 和 StartWindow 仅仅是模仿正式游戏设计的动画效果，实际的加载并不在此处进行。设计上主要通过更换背景图片完成动画效果。

● MenuWindow（菜单界面）



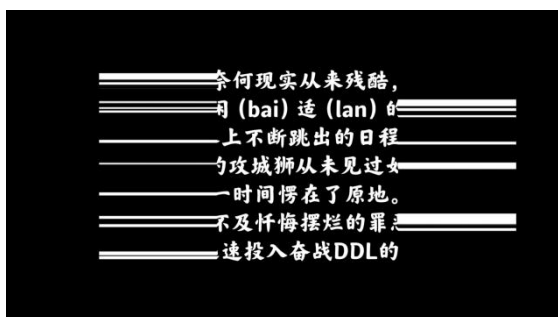
图八：初始 MenuWindow



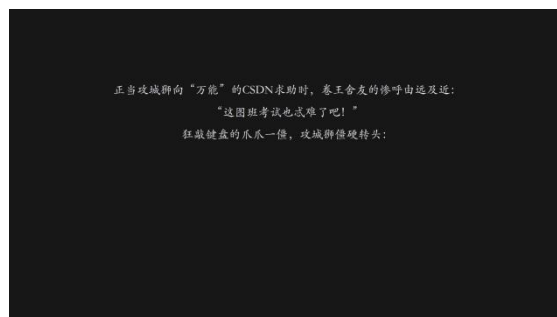
图九：最终 MenuWindow

MenuWindow 是显示游戏进度、快速跳转游戏和设置、帮助等辅助功能的入口所在的窗口页面。左上角是相框风格的游戏图标，右上角是辅助功能的入口图标，光标移至图标处会加深突出显示。下方依照游戏数据以结构词+概要的形式显示游戏进度。结构词是自定义的 QPushButton，用于跳转对应游戏进度；概要是用自定义的 QLabel 显示，用于提示相应内容和吸引玩家兴趣。

● ActWindow（剧情界面）&EndWindow（结局界面）



图十：ActWindow 截图



图十一：EndWindow 截图

这两个界面都是推进剧情的界面，不同点在于 ActWindow 涉及较长的主线剧情且附带一些配音，是通过播放预先制作的视频实现的；而 EndWindow 对应短小的支线结局，是通过读取 json 文件中的文本逐条显示的方式实现的。玩家通过鼠标点击翻页与界面交互，内部是由 QStackedWidget 切换显示页面实现翻页。

● LevelWindow（关卡界面基类）

LevelWindow 主要是规定关卡显示状态的基类窗口页面。主要按比例分为下方空白的游戏区和上方的信息栏。信息栏部分分为四个板块，即 DDL_List（事件记录与显示栏）、Warning_icon（警告图标）、live_bar 或者 DoubleLive（生命条）和 pause_block（辅助信息显示和暂停按钮）。由于自定义控件较多，本窗口采用纯代码实现。

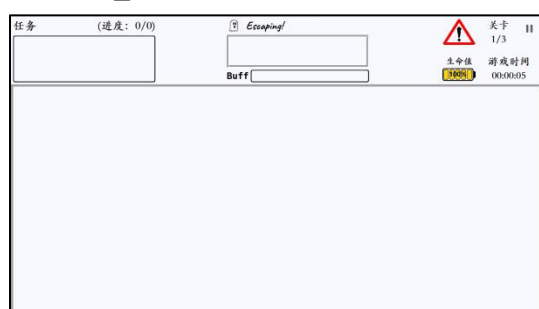
DDL_List 包括任务栏、Buff 栏和信息栏三部分。前两个是两个方向的 QListWidget。每当

发生事件时，依照信息构造 `QListWidgetItem` 并置于对应 `QListWidget` 的首位，同时依照事件编号将信息加入游戏数据类 `Game` 相应 `QHash<int, QListWidgetItem*>` 以及 `QHash<QListWidgetItem*, task_info>` 变量中存储。信息栏由 3 个 `QLabel` 组成，分别显示图标、名称和描述。每当一个新的 `QListWidgetItem` 被创建或者玩家手动点击栏目中的 `QListWidgetItem` 时，信息栏就会显示对应事件的信息。第三关中任务栏就隐去了。

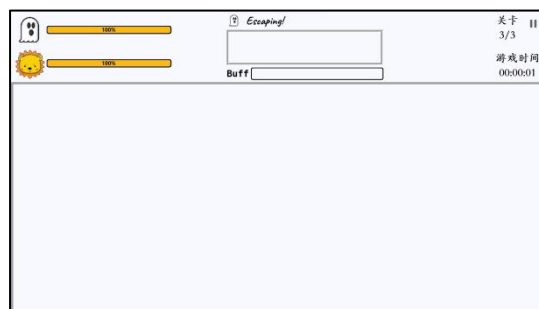
`Warning_icon` 是一个重载了 `paintEvent` 的 `QWidget`。其包含三种显示模式：不显示、随给定变量线性变化大小、以最大大小抖动，对应了不同危急情况下的反应。

`live_bar` 是前两关电池型生命值的显示，包括一个自定义的 `MyLabel`，`QProgressBar` 以及小段绘图。通过重载 `resizeEvent` 实现了字号按外框高度变化和保证血量格数恒定。随着血量的减少，血条的颜色也会发生变化。`DoubleLive` 是第三关的连续血量条。除了上述特点外，这次血量条减少时有着一定的动画效果。

`pause_block` 包含了显示辅助信息的 `MyLabel` 和暂停按钮。



图十二：第一、二关 LevelWindow



图十三：第三关 LevelWindow

● ControlWindow（控制主窗口）

以上的窗口都是显示在前端的窗口页面，而后端的 `ControlWindow` 是控制切换的窗口。所有上述窗口页面都是存储 `ControlWindow` 的主要部件 `QStackedWidget` 中。当需要切换窗口时，顶层窗口会发出 `changeWindow(index)` 的信号，被 `ControlWindow` 的 `toWindow` 函数接收后转到对应页面。关注到内存和时延的问题，我们将占用内存少的 `LoadWindow`、`StartWindow` 和 `MenuWindow` 直接存放在栈中，其余窗口都是采用显示才创建、关闭即删除的策略。同时为了保证游戏的流畅性，删除操作是通过 `SingleShot` 推迟进行的。

游戏数据和存读档操作的函数同样是在这个窗口中存储和进行的。

2. 3. 2 对话框

项目中所有辅助功能和单独页面都是通过模态显示的对话框实现的。对话框基类 `MyDialog` 规定了窗体各个属性、依照父窗体重设位置和关闭信号。由此派生出五个对话框类。

● StartDialog（初始对话框）

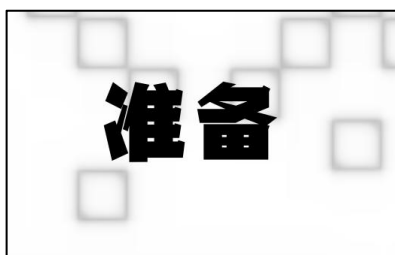
用于屏幕中央的单行文字动画显示。

● PauseDialog（暂停对话框）

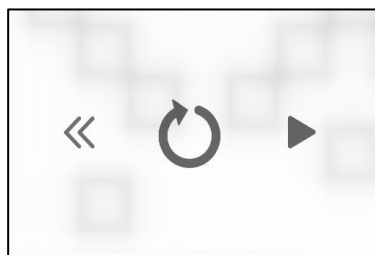
游戏时暂停界面。三个按钮从左到右分别实现了返回菜单、重新开始和继续游戏的功能。

● SetDialog（设置对话框）

设置音量和重置进度界面。设置滑条使用 QSlider 实现，可以通过键盘或鼠标调整。



图十四：StartDialog



图十五：PauseDialog



图十六：SetDialog

- HelpDialog(帮助对话框)

总体游戏介绍和帮助文档，使用 QScrollArea 实现滚动显示。

- AccDialog(成就对话框)

由场景图片和描述组成。一定时间后刷新成就框，点击返回菜单。

- TipDialog(提示对话框)

每关开始提示玩法和要求。使用 QPropertyAnimation 实现压缩进入和淡出。



图十七：HelpDialog



图十八：AccDialog



图十九：TipDialog

2.3.3 CoverMask

CoverMask 继承自 QWidget，本质上就是覆盖页面的一个全黑图层，用于实现界面的遮挡以及页面淡入淡出的效果。由于 QStackedWidget 切换页面时可能将下层页面暴露导致画面闪现，就在前一个页面的 hideEvent 中升起 CoverMask，然后定时关闭来遮挡缺陷。此外，播放背景音乐存在时延，同样使用这样的办法实现音画同步。淡入淡出是通过 QPropertyAnimation 更改窗体透明度完成的。

3 小组成员分工情况

汪之立（组长）：剧情文案编写、各游戏界面和辅助功能实现、界面串联

吴乐鸿（组员）：事件文案和图标设计、三个关卡具体实现和丰富美化

吴亮宇（组员）：部分界面设计、部分素材搜集、剧情实现、帮助文案编写

4 项目反思与总结

4.1 项目特色与亮点

我们在构思这个项目的时候重点就关注了原创性和趣味性两个关键词。原创要求我们无论是剧情还是玩法上要有所创新，不能单纯照搬照抄；趣味要求剧情设计合理、关卡难度适当，达到吸引玩家的目的。同时，我们也秉持“给休闲以生活”的宗旨，希望能通过游戏带给玩家生活上的启迪和激励。

关卡设计上的亮点主要在于后两关的人物的移动方式上。第二关角色采用了点拉拖拽给予速度的方式，移动过程中还会不断减速；怪兽的速度和角色的距离成负相关，进一步提高了游戏难度。这样的设计参考了学习动力如果不加以刺激就会不断衰减的现实。第三关角色采用了向光标所指位置加速而非直接运动的方式，同时设计了三种子弹、任务限时等内容。这样既增加了角色移动的难度，又迫使玩家必须移动，提高了关卡的可玩性。

剧情上的亮点在于缝合了生活性和趣味性。总体以大一信科的生活为蓝本，同时融合了各类梗和奇幻诡异的元素，力图达到源于生活、超越现实、合于趣味的效果。主线剧情通过三个关卡的推进，展现大一的攻城狮从逃避 DDL 到直面 DDL 从思想到行为的转变，侧面反映当代大学生的精神风貌。主线剧情之外，各关失败也会解锁支线结局，进一步丰富游戏主题。

4.2 项目改进之处

在最终完成项目时，我们对于最初设计稿的非必要部分进行了部分删减以适应工期，整个项目还有不少值得改进的部分。现罗列如下：

- 1 剧情部分和玩家的交互性有待增强，配上简单的矢量动画
- 2 关卡中间的过渡剧情和剧情配音
- 3 DDL 和攻城狮形象的动画效果
- 4 多存档和关卡断点续玩
- 5 菜单页面更多的动画效果，进一步美化界面

此外，在音乐音效、画面设计等方面的原创性等方面也有待加强。

4.3 工作总结

总体项目推进的过程还算顺利，基本在预计工期时间内完成了大部分的功能。作为新人合作组成的项目组，我们在这次项目完成过程中收获了不少的经验和方法。

项目按时完成的重要原因在于常驻的计划性和灵活性。我们在立项时就已经草拟了总体项目计划并划分了阶段。每周我们都固定时间开例会以检查上周进度和制定下周具体任务，保证项目不断推进。而在组员完成出现困难时，项目组也通过及时的沟通给予帮助或者调整计划灵活应对。

但是由于经验不足，项目具体完成时还是遇到了不少的阻力。首先，虽然每周都有任务，但是由于拖延症和规矩过于宽松等一系列原因，时常出现上周任务不得不延期，导致最后熬夜赶工和任务分配并不平衡的问题。其次，由于没有系统学习 Qt 完整的功能，在控件应用和美化过程中常常需要查询大量资料，代码实现时又因为功能添加等问题常常改弦易辙，耗费了大量的时间精力。最后，设计接口时由于考虑不周、注释不明确等原因，引起了不少的误解和无法推进的困境。

通过这次项目，我们巩固了面向对象的编程的思想，接触了 GUI 编程的常用语言 Qt 基本内容，熟悉了常用的工具 Github、Markdown 语言等等。此外，这次项目锻炼了我们团队协作能力和沟通能力，为我们下一步的学习生活夯实了基础。

最后，非常感谢北京大学程序设计实习课程的老师 and 助教提供一次这么有趣和宝贵的机会，也感谢老师和助教一学期的辛苦付出！