Evolution游戏开发计划书

2025-6-15（更新）

V1.0.00000.1

1. 项目概述与目标设定

**项目周期：**

**游戏内容：**

**主题**：进化

**背景**：受限于当前技术水平，人类尚无法理解从碱基对层面编辑基因的基因 。然而，人类已经掌握了较为先进的基因编辑技术。为了获得更优质的基因，玩家可选择扮演多重角色（如科学家、群体领袖，战斗指挥官等），通过参与进化过程，结合自然选择与人工选择机制，以筛选出优秀的基因特征，创造更优秀的生命体。同时需要对抗巨型生态（地球与外星）神经网络（碳基、硅基即机械生命 量子？）（引入AI）（参考灵笼的玛娜生态），生存下去。

**目标平台**：PC/移动端

**目标用户群体**：学生（中、高教育）

**项目宗旨**：以游戏的方式传播知识，赚米哈哈哈

**游戏类型**：RPG，半开放世界，2D，回合制

**运行结构**：客户端+服务端，客户端开源，配官方客户端，服务端保密，向客户端发送特定格式的数据文件，渲染由客户端完成，服务端负责逻辑运算。

二、核心开发流程（分阶段说明）

阶段1：基础知识学习（5年，可选择边开发边学习）

全科医学导论课程将为我们提供一个全面的医学基础知识框架，涵盖从基础医学到临床医学的各个方面，使我们能够理解人体的复杂性和疾病的多样性。遗传学基础部分将深入探讨基因的结构、功能以及遗传规律，帮助我们掌握基因编辑技术的核心原理和应用。生物信息学课程将介绍如何利用计算机技术处理和分析生物大数据，使我们能够更好地理解生物系统的复杂性。

药学课程将使我们了解药物的发现、开发和应用过程，包括药物的作用机制、药理学以及药物设计的基本原则。内科学部分将重点讲解人体内部系统的功能和常见疾病的诊断与治疗，使我们能够更好地理解疾病的临床表现和治疗策略。生理学课程将深入探讨人体各系统的功能和调节机制，帮助我们理解人体在正常和病理状态下的生理变化。

生物化学课程将介绍生物大分子的结构、功能以及代谢途径，使我们能够理解生命活动的化学基础。细胞生物学部分将深入研究细胞的结构、功能以及细胞间的相互作用，帮助我们掌握细胞在健康和疾病状态下的变化。此外，人工智能与机器学习原理课程将为我们提供AI和机器学习的基本概念、算法和应用，使我们能够设计和开发智能化的游戏。

团队成员需要深入理解基因编辑技术，掌握其在医学和生物学中的应用，了解其潜在的伦理和法律问题。同时，进化论原理部分将使我们了解生物进化的机制和过程，帮助我们理解生物多样性和物种起源。AI算法的学习将使我们能够掌握机器学习和深度学习的基本原理，为后续的游戏设计与开发奠定坚实的理论基础。通过这些课程的学习，我们将具备跨学科的知识储备，为实现创新和高质量的游戏开发提供支持。

第二阶段：理念构思与模型制作，原型构建（持续六个月）

在这一阶段，研究团队将深入细化游戏世界观，精心策划角色、场景和交互方式。利用游戏引擎(暂定LibGDX，开源免费)，快速构建游戏原型，以测试核心玩法和用户体验的可行性。同时，开始规划游戏内部经济体系和社交功能，目的是确保游戏的长期吸引力和提升玩家的忠诚度。构建生物模型。

并进一步细化生物模型的行为模式与外观特征，使其更加栩栩如生，符合游戏世界观。同时，对游戏性能进行优化，确保在各种设备上都能流畅运行。此外，研究团队还将进行多轮内部测试，收集反馈，不断调整和完善游戏原型，为后续的开发阶段打下坚实的基础。在这一整个过程中，都将持续关注玩家需求与市场趋势，确保游戏最终能够赢得玩家的喜爱与市场的认可。

//后为ai生成，勿理

阶段3：技术突破与核心功能开发（1年）

聚焦于实现游戏中的关键技术挑战，如基因编辑模拟算法、AI生态神经网络的智能行为设计、以及2D回合制战斗系统的优化。与科研机构合作，引入最新的生物信息学研究成果，提升游戏的科学性与真实性。同时，加强游戏的稳定性与性能优化，确保流畅的游戏体验。

阶段4：内容扩展与测试调优（6个月）

根据玩家反馈与市场趋势，不断扩展游戏内容，包括新增剧情、角色、装备与关卡。进行多轮内部测试与外部封闭测试，收集数据，持续优化游戏平衡性、UI设计与用户体验。确保游戏在正式发布前达到最佳状态。

运营维护：

发布首日补丁（Day-1 Patch）。

制定内容更新计划（DLC/赛季活动）。

进行社区管理与客服响应。

进行数据分析：监控留存率、付费转化等指标，驱动版本迭代。

三、项目组织与资源规划

团队构成

核心角色：项目经理（Razier）、主程序、主美（由于随机性以及快速迭代，需要引入AI）、主策（Razier）、音效设计师（暂时不设战斗音效，）

程序：Razier

美术：

策划：

音效：

。

扩展支持：外包管理、本地化团队、QA外包。

资源需求

| 类别 | 具体内容 |
| --- | --- |
| 硬件 | 高性能工作站、测试设备（覆盖目标平台机型）、服务器集群 |
| 软件 | 开发工具（Visual Studio/Maya）、版本控制系统（Git）、项目管理工具（Jira） |
| 预算分配 | 研发（60%）、美术外包（20%）、市场推广（15%）、应急储备（5%） |

四、风险管理策略

技术风险：引擎兼容性问题→预留备用引擎方案，采用模块化代码设计。

市场风险：竞品抢先上线→持续竞品监控，储备差异化内容。

进度风险：关键路径延迟→设置里程碑缓冲时间（Buffer Time）+优先级重排。

合规风险：版号审核失败→提前研究政策，预留法律咨询资源。

五、进度控制与里程碑

采用关键里程碑驱动：

| 里程碑节点 | 时间点 | 交付成果 |
| --- | --- | --- |
| 概念验证完成 | M1 | 市场分析报告+核心玩法文档 |
| 可玩原型通过 | M4 | MVP版本（含基础战斗/移动） |
| Alpha版本 | M12 | 所有系统集成，内部测试启动 |
| Beta版本 | M18 | 功能完整，开启公测 |
| 金主版（Gold） | M24 | 最终修复完成，提交发布 |

迭代管理建议：每个里程碑后召开回顾会（Retrospective），调整后续计划。

总结

此框架强调动态控制与模块化设计：

灵活性：通过敏捷迭代适应需求变化（例如测试反馈调整关卡设计）。

质量保障：将测试嵌入全周期（单元测试→兼容性测试→UAT）。

成功关键：

确立清晰的需求冻结点（例如Beta后仅修BUG不加功能）。

建立跨部门协同机制（程序/美术/策划每日站会）。

实时风险仪表盘（跟踪进度偏差率/BUG解决率）。

实际应用中需根据项目规模进行裁剪（例如独立游戏可简化测试阶段），重点保持核心循环（设计-开发-验证）的高效运转。

1. 详细内容

游戏核心：

各种生物（化学仅限小分子层面）模型运算得出宏观数据，用于传统回合制游戏

游戏难度：

简单：给出相应的方法与解释，让用户选择

低：给出相应的方法

中等：给出相应操作+干扰项

困难：无提示，给出所有可操作选项

地狱（较难开发，涉及物理，重置原子模型，暂时只用牛顿力学，不考虑相对论，不知道算力够不够，所有化学、生物将发生改变，这或许已经超出这个项目的范围了，作为项目储备吧，或者引入不存在的元素构成新的分子，令其能够影响原来的生物模型）