

## 澳大利亚3-12年级学生学习与作业平台开发方案

本方案旨在设计并实施一个面向澳大利亚3-12年级学生的在线学习与家庭作业平台，参考了IXL和百词斩等知名在线学习平台的特点。平台将支持学生日常背单词与完成作业，并提供自动批改、错题本、奖励机制等功能，以激励学生高效学习。以下将从功能模块、用户角色、内容展示、文件上传、后台架构、自动批改、App解锁、UI/UX设计、技术栈和开发计划等方面进行详细说明。

### 功能模块列表与说明

平台拟包含以下主要功能模块：

- **背单词模块：**提供英语词汇学习功能。每个单词配有图片、发音和情景例句，帮助学生通过多感官刺激来记忆单词<sup>1</sup>。学习方式参考百词斩的“图背单词”方法，以图片加例句的选择题形式呈现单词释义<sup>2</sup>。系统支持单词发音和例句朗读，提供从词根、例句、释义等多层次的提示，帮助学生理解记忆<sup>3</sup>。此外，结合“记忆曲线”算法实现智能复习，即根据遗忘规律定期将学过的单词加入复习列表，提高记忆效率<sup>1</sup>。
- **作业布置模块：**供教师或家长发布家庭作业或练习题。支持录入各科作业题目（可包括文字说明、公式、图片等），并指定年级和科目。发布的作业可设置截止日期、分值等信息。学生登录后可查看并完成相应作业。作业形式包括选择题、填空题、简答题等；对于选择题和填空题，系统可自动判分，简答题则支持教师/家长在线批阅。每份作业完成后，学生可即时看到客观题得分和答案解析，获得及时反馈，以便了解错题原因<sup>4</sup>。教师/家长也可查看学生的提交状态和成绩报告。
- **自动批改与评分模块：**系统对客观题进行自动判分，对主观题提供辅助批改工具。对于数学、物理、化学等涉及公式和符号的题目，平台支持标准答案的多种等价形式匹配。例如数学题支持判断学生答案的数学等价性（如分数 $\frac{1}{2}$ 与小数 $0.5$ 应判定为相等），利用数学表达式解析引擎将学生答案规范化后与标准答案比对。此外，平台采用LaTeX/KaTeX技术渲染题目和答案中的公式，保证在网页上正确、美观地显示分数、根号、化学方程式等符号<sup>5</sup><sup>6</sup>。高级功能上，可参考Gradescope等平台的做法，引入AI识别手写公式的能力，以自动识别批改学生书写的数学符号表达<sup>7</sup>。自动评分模块在批改完后将成绩记录入学生档案，并将错题标记供错题本模块使用。
- **错题本模块：**系统自动归集每个学生在练习和作业中做错题目，形成个人错题本<sup>4</sup>。错题本中记录题目、学生的错误答案、正确答案以及解析。学生可以随时查看自己的错题及解析，了解错误原因<sup>4</sup>。同时提供“错题重练”功能，允许学生针对错题进行强化练习<sup>8</sup>。系统会随机抽取错题让学生重复作答，当某题连续两次作答正确后，即认为已掌握，系统标记该题已“消灭”，后续练习中不再出现<sup>8</sup>。若未连续答对两次，则该错题会在本次练习中反复出现，直到达标为止。这一机制帮助学生有针对性地克服弱项，避免一错再错<sup>8</sup>。错题本还统计每个学生的错题数量、错题率及“错题消灭率”等数据，学生可以直观了解自己的薄弱环节；管理员/家长亦可在后台查看学生错题练习情况，及时掌握学习动态<sup>9</sup>。
- **CSV内容导入模块：**为了便于批量导入学习资源，平台支持通过CSV文件批量上传单词列表、题库等内容。管理员、教师或家长可以下载系统提供的CSV模板，将自有内容填入后上传至平台。比如，单词表CSV模板包含字段：英文单词、释义、例句、图片URL/文件名等，上传后系统自动生成相应的单词卡片；作业题目CSV可包含：题干文本、选项（对于选择题）、正确答案、难度、科目等字段。CSV导入机制有助于内容的快速扩充，方便将现有资料移植到平台上使用<sup>10</sup>。上传时系统会校验CSV格式与内容完整性，解析后将数据存入数据库。如有图片等文件，需要提前上传至指定媒体库并在CSV中引用其

路径，或者通过压缩包一起上传。通过CSV批量导入，**大幅降低**了内容管理的工作量，让教师和家长更轻松地分享学习资源。

- **每日学习成果奖励模块**：平台实现“每日学习成果解锁iPad指定App”的特色功能。具体规则为：如果学生当天在数学练习中得分  $\geq 90\%$ ，则可解锁娱乐类App（如YouTube或游戏）15分钟；若数学练习满分100%，则解锁30分钟。同理，英语练习达到上述条件也可获得相应时长奖励。平台会根据学生当天的练习成绩自动计算可解锁时长，并生成一个“解锁令牌”。通过家长端，家长可以查看孩子当日学习达标情况，并触发解锁流程。技术实现上，可考虑与家长控制应用或设备管理工具集成。例如，借鉴ScreenTreat等第三方App的理念，将孩子的屏幕时间与任务完成情况相关联：**孩子完成指定学习任务后即可获得额外屏幕时间奖励**<sup>11</sup>。我们的平台可以通过与此类应用的API对接，自动将奖励时长授权到孩子的设备上，从而**解锁指定App一段时间**<sup>11</sup>。在未引入第三方整合的情况下，也可以采用家长确认的方式：当孩子达到奖励条件时，系统通知家长，由家长在iPad上手动放行对应App并启动计时。无论实现细节如何，该模块的核心目的是将学习成果直接转化为孩子喜爱的娱乐时间，从而形成正向反馈，激励学生努力完成每日学习目标。
- **用户管理与权限模块**：支持多角色的用户体系（学生、家长、教师、管理员）。该模块提供用户注册、登录、权限分配，以及用户信息管理等功能。管理员可以创建和管理学生/家长账号，设置家长与学生的关联关系（如Grace和Dan分别与Michael或August绑定为家长子女关系），以及管理教师账号。每个用户登录后将根据其角色看到不同的界面和功能（例如学生看到练习和作业，家长看到孩子的成绩与报告等）。Dan用户在本方案中既是家长又是管理员，系统需支持一个账户具备多重角色权限。在安全方面，用户密码需加密存储，敏感信息要保护，确保仅授权用户才能访问相应数据。
- **数据分析与报告模块（附加）**：为了帮助家长 and 教师了解学生学习情况，平台应提供基础的数据分析功能。包括学生的练习时长、正确率、成绩趋势、错题知识点统计等报告<sup>9</sup>。家长可以查看自己孩子的每日/每周学习报告，教师/管理员可以查看全班或全体学生的统计数据，从中发现共性问题。报告应支持导出或邮件发送，例如每日自动给家长发送孩子的进度报告<sup>12</sup>。此模块为非必需的增强功能，可在后期优化阶段加入。

以上模块共同构成完整的平台功能集。其中，背单词、作业布置、自动批改、错题本、奖励机制是核心模块，将优先实现；用户管理和内容导入提供支撑保障；数据分析则为锦上添花的扩展功能。

## 用户角色与权限说明

平台涉及四类用户角色：**学生、家长、教师、管理员**（本方案示例中教师角色未单独列出，可与家长权限相似或由管理员兼任）。

- **学生（如Michael、August）**：学生是平台的主要使用者。学生账号可浏览并学习分配给自己的内容，包括背单词练习和老师/家长布置的作业。学生可以查看自己的学习任务列表、完成练习提交答案，并获得即时反馈与成绩。学生只能查看与自己相关的内容和数据，无权修改题目内容、查看他人数据等。学生账号下会关联其错题本和学习记录数据，供其个人查看复习。**权限总结**：完成练习/作业、查看自己的成绩和错题、获得奖励时间通知。
- **家长（如Grace、Dan）**：家长账号主要用于监督和辅助孩子学习。每个家长账户可绑定一个或多个学生账户，方便查看孩子的学习进度和成绩。家长可发布或上传作业给自己孩子练习（相当于在家庭环境中充当出题者的角色），也可以查看孩子提交的作业答案和系统自动评分。家长能浏览孩子的错题本，了解哪些知识点是薄弱项<sup>9</sup>。此外，家长有权限管理“学习成果解锁App”功能：查看孩子是否达成当天目标，并执行解锁操作（或授权系统自动解锁）。**权限总结**：绑定查看孩子的数据、布置作业给孩子、审核/查看作业结果、启动奖励解锁。

- **教师（未来扩展）**：如果引入教师角色，则教师可管理其班级学生的数据。教师能批量上传作业、查看班级整体情况、给多名学生布置任务、以及查看班级错题统计等。教师权限类似家长但作用域更大（对应多个学生），可以看作是一类特殊的家长账号。由于本方案用户示例中未包含专门教师账户，实现上可由管理员承担教师职能或在后期增加教师模块。
- **管理员（Dan，同时也是家长）**：管理员拥有最高权限，负责平台的运维管理工作。管理员可以进行用户管理（创建/删除账户、分配角色、重置密码等）、内容管理（通过CSV批量导入单词和题库、审核教师/家长上传的题目内容）、系统配置（如设置奖励分数阈值90%、满分100%的参数，配置可解锁的App列表），以及查看全站的数据分析报告<sup>9</sup>。管理员还能管理平台后台的数据库和文件存储，处理敏感操作。Dan作为示例管理员，同时也是家长，意味着Dan登录后可以访问管理员后台管理界面；如果以家长身份使用，则可切换到家长视角查看自己孩子August的学习情况。这要求系统支持根据用户权限动态调整界面与功能选项。

**权限控制**方面，系统将根据用户角色在前后端都实行访问限制。例如前端菜单和路由会区分学生与家长的可见项；后端API在处理请求时会验证用户身份和权限，防止未授权访问他人数据或管理员操作。重要管理操作需要管理员账号且可能要求再次确认或更高级别认证（如二次登录验证）以确保安全。总体来说，权限设计遵循最小权限原则，各角色只能使用其教育场景所需的功能，确保数据隐私与系统安全。

## 内容展示逻辑（尤其科学学科公式）

平台需适配各学科内容的展示需求，特别是数学、物理、化学等科学学科经常涉及公式、图表、特殊符号，其呈现需准确清晰：

- **数学公式与符号**：采用LaTeX或类似的标记语言来存储和渲染数学表达式。在题目编辑时，教师/家长可以使用LaTeX语法编写公式（或通过公式输入工具插入）。前端使用MathJax或KaTeX库将LaTeX渲染为浏览器可显示的格式<sup>5</sup>。例如，分数、平方根、积分符号等都会以专业排版呈现，和课本印刷效果一致<sup>5</sup>。这样无论是在题干、选项还是答案解析中，数学内容都清晰易读，避免因浏览器或字体问题导致的公式错误显示。Gradescope等在线批改平台也采用了KaTeX技术来确保公式呈现的正确性<sup>6</sup>。对于简单的算式，系统也支持使用Unicode字符（如 $\frac{1}{2}$ ,  $\sqrt{\quad}$ ）直接呈现，但推荐使用统一的LaTeX方式，确保复杂公式（矩阵、分段函数等）也能正确显示。
- **物理内容**：物理题目可能涉及公式、单位符号、向量符号等。处理方式与数学类似，使用LaTeX渲染。同时，物理题目有时需要插图或电路图，平台支持在题目中插入图片或SVG图形来辅助说明。如果需要交互式展示（如演示模拟实验），可在后期考虑嵌入HTML5动画或引入第三方交互工具。在文本描述上，注意正确显示上标下标（如加速度单位  $\text{m/s}^2$  应正确显示平方），这些均可通过LaTeX的`^`和`_`语法实现<sup>13</sup>。另外，对于物理中的特殊符号（ $^\circ$ 表示度， $\Omega$ 表示欧姆等），平台会提供符号插入选项或允许直接复制粘贴符号，确保题目录入者方便输入，学生端准确呈现。
- **化学内容**：化学学科常出现化学式和化学方程式，例如 $\text{H}_2\text{O}$ 、 $\text{Na}^+$ 、结构式等。平台同样采用LaTeX或mhchem插件来渲染化学式<sup>14</sup>。mhchem是专门用于在LaTeX中书写化学方程的宏包，Gradescope等平台的方案表明可通过KaTeX集成mhchem实现化学式的正确显示<sup>6</sup>。因此我们的系统将支持`\ce{}`语法：如`\ce{H2O}`渲染为 $\text{H}_{2\text{O}}$ ，`\ce{CO2}`渲染为 $\text{CO}_{2\text{}}$ ，并能处理化学方程的箭头符号、催化剂条件等<sup>5</sup>。这样学生在做化学题时，能直观看到规范的化学式表示。若有机化学结构式等复杂图形，可通过图片上传的方式辅助呈现。
- **多媒体与其他学科**：英语和人文科目主要是文字为主，平台会确保富文本显示正常（支持粗体、斜体、段落等基础格式）。如果需要展示音频（如听力题）或视频（如科学实验小视频），平台可集成多媒体播放控件，以增强学习体验。所有内容将保证在PC和iPad等终端上自适应显示，字号和布局针对不同屏幕优化，确保学生不论在大屏还是平板上都能清晰阅读题目与材料。

- **响应式与排版：**采用响应式网页设计，使内容展示能根据设备屏幕调整版式。例如，在平板竖屏时，可能需调整公式换行或图片大小，以避免左右滚动。数学公式较长时在小屏设备上自动缩放或换行显示，避免溢出。表格、图表在窄屏上可水平滚动或缩略图预览。通过CSS和前端框架的栅格系统，确保不同内容元素（文字、公式、图片、交互组件）都布局合理，不会相互遮挡或错位。
- **本地化与课程标准：**内容展示还体现课程体系的本地化特点。根据澳洲课程标准，某些年级会使用特定术语（如年级对应术语Year 7, Year 8等）和符号约定（例如用小数点还是逗号等）。平台将遵循澳大利亚国家课程规范，保证用词和符号符合当地教材习惯。例如数学中的小数点用“.”表示，度量单位采用公制，货币符号为"\$"等符合澳洲习惯。平台的内容库也将按澳洲课程大纲组织，各年级各科目的知识点架构清晰，与官方标准对齐<sup>15</sup>。IXL等成熟产品已证明对接本地课程标准的重要性——IXL针对澳洲课程对每个年级的内容做了完全的对对应覆盖<sup>16</sup>。我们的平台亦将每道题、每项练习关联到澳洲课程的知识点，以便教师和家长了解孩子的掌握程度是否达到课程要求。

总之，在内容展示方面，本平台注重**精准、规范和清晰**。借助成熟的公式渲染技术和合理的设计，确保科学学科内容不因数字化而失真，学生获得的体验尽可能接近课堂纸质资料的质量。同时兼顾多媒体和交互元素，让学习内容生动丰富。

## 文件上传机制与格式要求

为了方便教师和家长批量导入内容，平台提供文件上传功能，支持**CSV文件**批量导入以及其他类型文件的上传，具体机制和要求如下：

- **CSV批量导入：**平台允许管理员、教师通过CSV（Comma-Separated Values）文件一次性导入多条学习内容记录。典型应用包括：导入单词列表、习题集、试卷题库、用户名单等。使用CSV能极大提高内容录入效率，实现跨系统数据迁移<sup>10</sup>。为确保数据正确导入，平台提供各类CSV模板下载，并严格定义格式要求：
- **单词表CSV格式：**第一行作为表头，包含字段名称，例如 word, meaning, example\_sentence, image\_url, audio\_url 等。每行代表一个单词的数据，其中**word**为英文单词，**meaning**为释义（全英文，因为网站内容全英文），**example\_sentence**为示例句子（英文，最好包含该单词空缺让学生填词），**image\_url**为该词图片的链接或文件名，**audio\_url**为发音文件路径（如有）。导入时系统根据image\_url从已有媒体库匹配图片（或在上传CSV同时附带图片文件压缩包），根据audio\_url匹配音频文件。CSV需保存为UTF-8编码以兼容特殊字符。导入后系统反馈成功导入多少单词，如有错误（如必填字段缺失、格式不符）会记录日志提示用户修正。
- **题目题库CSV格式：**根据题型不同提供几种模板：
  - **选择题模板：**字段包括 question\_text, optionA, optionB, optionC, optionD, correct\_option, explanation, subject, grade 等。其中question\_text支持一定的Markdown或LaTeX用于公式，option列填入选项内容（如果少于4项，多余列可留空或不写），correct\_option填写正确答案选项标号（A/B/C/D），explanation为答案解析，可选填。subject和grade用于标记题目归属科目和难度。
  - **填空题/简答题模板：**字段包括 question\_text, answer, explanation, subject, grade 等。其中answer填写标准答案，如果允许多个答案（同义词等），可用分号分隔。
  - **题库导入**支持在CSV中指定题目分组或标签，方便后续按章节出卷。比如增加字段 tag 或 chapter 来归类。

CSV中的题干和解析字段内如包含逗号等特殊字符，需用引号括起避免格式错乱。对于需要插入图片的题目，可在question\_text里用占位符（如 <img1>）表示，然后在附带的图片压缩包中提供对应命名的文件，系统导入时会占位符替换为实际图片的引用链接。

- **用户账户CSV格式（管理员专用）**：可批量创建学生或家长账户。字段如 `username`, `password`, `role`, `associated_student` 等。出于安全考虑，初始密码可以由系统随机生成或使用统一默认密码（如后续要求首次登录修改）。

CSV上传界面将提供预览功能：用户选择文件后，系统会解析前几行数据供预览，用户确认无误再提交正式导入。导入过程在后台执行，可能需要几秒至几分钟（视数据量而定）。完成后向用户展示导入结果报告，包括成功条数和失败条数，失败的行及原因（如格式错误或数据冲突）。管理员应有权限撤销导入操作（在误导入时可一键删除最近一次导入的数据）。

- **媒体文件上传**：对于图片、音频、视频等非文本内容，平台支持通过专门的媒体管理模块上传。用户可在相应模块中批量选择文件上传到服务器。上传的文件存储在服务器的文件系统或云存储，并生成唯一URL。在内容编辑或CSV中即可引用该URL，以在前端显示相应媒体。为保持系统性能和安全，限制单个文件大小（如图片不超过5MB，音频不超过10MB，视频不超过100MB等），并限制文件类型（图片限JPEG/PNG，音频限MP3/WAV等）。上传时系统会对文件进行病毒扫描，图片做适当压缩处理。对于通过CSV引用的外部URL图片，系统可选择在夜间任务中批量抓取缓存这些图片到本地，以免第三方链接失效影响显示。
- **上传权限与流程**：并非所有用户都可以随意上传文件和内容。一般而言：
  - **教师/管理员**有权限上传题目内容和单词表。家长通常也允许上传作业题目，但可能受到管理员审核机制约束（防止不适当内容）。学生无权上传内容（除非在回答题目时上传作答附件，见下）。
  - 上传的所有内容建议走**审核流程**：管理员可设置某些用户上传内容需后台审核通过后才对学生可见。例如家长自己创建的练习题，在未审核前仅对自家孩子可见，对公共题库不可见；教师上传的公共题库由学科管理员审核后发布。审核流程可简化为由管理员定期检查新上传内容，勾选通过或删除。
  - **学生作答附件上传**：对于非选择题类作业，学生可能需要上传图片或文件作为答案（如数学过程草稿、作文文档等）。平台允许学生在作业提交时附加文件上传。每题可限制最多上传文件数和大小，总提交大小也有限制。服务器接收后将文件与作业答案关联存储。教师/家长批改时可点开查看附件。这样的文件上传也归属到作业模块内部的机制，有专门存储目录并权限控制，仅相关教师和本人的可访问。
  - **错误处理与日志**：所有上传行为都会记录日志，包括上传者、时间、文件名/CSV名称、结果等。如发生错误（如CSV格式不对、文件超过大小限制），系统及时反馈错误信息。同时后台持续监控上传服务的运行，防止大文件上传导致服务器性能问题。必要时对批量上传操作进行排队限流，保证服务器稳定。

通过以上机制，本平台能够灵活接收各类学习内容输入。一方面教师和管理员可以方便地批量构建资源库，另一方面学生的提交也得到良好支持。文件和数据上传功能在提供高效性的同时，注重规范格式和权限审核，确保平台内容质量与安全。

## 后台与数据存储需求

平台后台架构设计需要满足多用户并发使用和数据安全可靠的要求。主要的存储需求包括用户数据、学习内容、作业成绩、日志等。下面概述后台架构和各类数据的存储方案：

- **总体架构**：后端采用分层架构，提供RESTful API或GraphQL接口与前端通信。考虑到平台功能丰富，采用模块化设计，将不同业务（用户管理、内容管理、作业批改等）划分为独立服务或模块。后台可部署在Tender环境下的服务器集群上（Tender环境为本项目的部署环境，已配置相应运行支撑）。为保证可扩展性和稳定性，后端应用可能采用容器化（Docker）部署，并结合CI/CD进行持续交付，使后续更新更平滑。

- **数据库选型**：选用关系型数据库（如MySQL或PostgreSQL）作为主要数据存储，用于保存结构化的数据。这是因为教育平台的数据（用户、课程、成绩等）具有清晰的关系结构，关系型数据库方便执行复杂查询和保证事务一致性。数据库主要表设计如下：

- 用户表（Users）：存储用户基本信息（ID、姓名、邮箱、加密密码、角色等）。
- 角色及权限表（Roles, Permissions）：定义角色类型及对应权限列表，实现细粒度权限控制（也可用RBAC框架内置功能实现）。
- 学生-家长关联表（StudentParent）：映射学生和家長账号的对应关系，实现家长查看孩子数据权限。
- 单词表（Words）：存储词汇内容（单词、本地图片路径或URL、释义、例句、发音文件路径等）。
- 题目表（Questions）：存储练习/作业题目（题干文本、题型、答案、解析、科目年级标签等），可能进一步分成ChoiceQuestions, FillBlankQuestions等子表，以适应不同字段。
- 作业发表表（Assignments）：记录每次作业的发布信息（发布者、发布时间、包含题目列表、针对班级/学生范围、截止日期等）。
- 作业提交表（Submissions）：记录学生对每份作业的作答（关联学生ID和Assignment ID、提交时间、每题作答内容或选项、得分、是否完成等）。
- 错题本表（WrongBook）：记录每个学生的每道错题（学生ID、题目ID、首次错误时间、最新练习状态、已消灭标记等）。
- 成绩日志表（Scores/Progress）：存储学生各科目的累计学习数据（练习次数、平均分等）用于统计分析。
- 奖励记录表（Rewards）：记录学生每日是否达成奖励条件及奖励时长、对应的解锁时间窗口等，以便后台校验与家长确认。

除上述核心数据表外，还会有系统配置表、日志表（记录用户登录、内容变更等）、审核记录表（记录内容审核情况）等辅助表。采用**规范化**数据库设计，保证数据一致性，必要的地方建立索引优化查询速度。对于高并发的读操作（如学生练习时频繁读取题目或提交答案），可考虑使用只读数据库副本或分片提高性能。

- **文件存储**：媒体文件（图片、音频、附件等）通常不存入关系数据库，而是存于文件系统或云存储服务（如AWS S3、Azure Blob等）。本方案可使用Tender环境提供的文件存储资源，将文件按照类别分目录保存，例如：
  - `/media/words/images/` 存放单词图片，文件名以单词或ID命名；
  - `/media/words/audio/` 存放单词发音音频；
  - `/media/questions/` 存放题目插图（按题目ID分子目录）；
  - `/media/submissions/` 存放学生提交的附件（按提交ID或学生ID区分文件夹）；
  - `/media/others/` 其他杂项文件。

数据库中则存储这些文件的相对路径或URL引用。例如Words表中image\_path字段保存 `words/images/word123.jpg`。为提高访问速度，前端请求这些静态文件时，可通过Nginx等静态服务器直接提供服务，或使用CDN加速（视用户规模决定）。

- **缓存**：为了提高常用数据的读取效率，后端可使用缓存系统（如Redis）。缓存适合存储一些变化不频繁但读取频繁的数据，如课程列表、词汇库索引、配置项等。对于每个学生的当前练习进度也可暂存于缓存，以便即时更新页面显示。例如学生做题时的计分、定时器，都可以存在Redis里，以实现实时性。同时Redis也可用于会话管理（如JWT之外进一步存储登录态）或队列服务（如处理异步的批量导入CSV任务）。
- **搜索与索引**：如果平台需要全文搜索功能（比如按关键词查找题目、查单词），可以引入ElasticSearch等搜索引擎，对题库和词库建立全文索引，提升搜索速度和功能。这属于优化层面需求，可在后期实现。
- **数据备份与安全**：教育数据具有重要性，需要定期备份和保护。数据库设置每日自动备份，至少保留近30天的备份快照。媒体文件亦需定期备份到异地（或利用云存储的多副本机制）。所有与学生有关的个

人数据遵循隐私法规（如澳大利亚Privacy Act），仅用于学习目的，不向无关方提供。传输过程中使用HTTPS加密，数据库对敏感字段（如密码）进行散列存储，对某些敏感数据（学生个人信息）可考虑加密存储。管理员应有一套权限配置，普通管理员不能导出或查看所有敏感数据，必要时做操作留痕审计。

- **后台管理界面**：提供给管理员和授权教师使用的管理后台，可以是一个Web界面或直接使用前端admin模块。通过后台界面，可执行数据管理和系统配置，例如查看用户列表、审核内容列表（新上传题目、词汇）、查看日志、手动调整学生奖励时间或分数规则等。后台界面也会连接上述数据库和存储服务，权限上仅管理员角色可访问。
- **扩展与集成**：后台应考虑未来可能的集成需求，例如与学校现有信息系统（LMS等）的数据同步。可能需要支持通过API与外部系统交互，如获取学生名册，推送成绩到学校系统等。采用模块化API设计将使这些集成更容易。同时，为支持移动端App开发（若未来需要推出移动应用），后端API要设计良好，可被不同客户端复用。

综上，后台的数据存储方案以关系数据库为核心、文件系统为辅助，辅以缓存和搜索技术以优化，形成**可靠可扩展**的后端支撑。注重数据**的一致性、安全性**，保证学生的学习记录和成绩不丢失、不错乱。随着用户量增长，后端架构可通过水平扩展数据库读写分离、增加缓存节点等手段来支撑更高的并发和数据量。

## 自动批改与评分逻辑

自动批改是本平台的一大特色和难点所在。针对不同题型和科目，系统需设计合理的评分逻辑，实现客观、公正且高效的自动评分。以下分题型说明系统的批改与计分机制：

- **客观题（选择题、判断题等）**：这类题型有标准唯一答案，系统批改较为直接。数据库存有正确答案选项，学生提交后系统即时比对：
- 对于单选题/判断题：如果学生选项与标准答案匹配，则判定此题正确，得满分；不匹配则得0分。
- 对于多选题：可以采用部分评分或全对才给分。根据需求设定：一般教育场景可要求“全选对才给满分，少选/错选均不得分”，以避免猜测取巧。但也可按比例给分（如每个正确选项占一定分值，错选则该题得0）。此规则在题库中标记，可在题目属性中设置是否部分给分。
- 批改实现上，系统将正确答案与学生答案进行集合比较，并按照预设规则计算分值。结果存入Submissions表并在前端实时显示正确或错误标记。
- **填空题（数值计算、简答词填空）**：
- 对于**数值计算填空**（如数学题让填写结果）：系统在题库中存储标准答案，可能包含多种等价形式。例如答案可以设定为`1/2 or 0.5`，表示分数或小数均正确。学生提交的答案先经过预处理：去除多余空格，统一格式（如小写转大写或反之），再与标准答案比对。如果答案是数值，系统可将其解析为浮点数或有理数表示，与标准值比较差值在允许误差范围内（考虑四舍五入误差）则判为正确。例如物理计算题可允许±5%的误差。对于分数等精确值则要求**严格匹配或等价**（通过计算两分数之差是否为0判断）。必要时集成数学计算库检查等价性，如`Sympy`或`math.js`，以支持更复杂的代数式对比。
- 对于**文本填空**（如单词拼写题、问答题中的关键字）：标准答案可支持多个同义表达。系统采用**关键词匹配或字符串比对**。简单场景下，学生答案去掉标点符号后，与标准答案（或答案列表）逐一比对，完全匹配则给分。如果答案需要包含多个关键词（例如问答题要求提及两点要素），可在题库为标准答案设置关键词列表及评分规则，如每提及一个关键词给一半分，两者都提及才满分等。这种情况更复杂，当前MVP实现时可以限制为单个词/短语的精确匹配题。大小写可忽略，但对拼写严格要求（除非特别注明近似也可）。

- **答案正则**：对于某些填空答案，可能用正则表达式来判断（如日期格式、化学方程等的平衡判断）。管理员可在标准答案中设置正则匹配模式，系统使用该模式测试学生答案以确定正确性。这适用于答案形式变化多但都有共同模式的情况。
- **简答题/主观题**：这类题开放性强，自动批改难以覆盖所有情况。在MVP中，我们可采取**半自动方式**：系统无法直接给分，但可辅助提供相似答案对比、AI初判建议等供人工参考。例如：
  - 平台可使用AI自然语言处理，对简答题生成一个初步评分建议或将答案与标准答案关键点进行匹配程度评分，然后由教师/家长最终裁定分数。这类似ASSISTments等工具利用AI辅助老师快速评分<sup>17</sup>。
  - 由于本方案主要强调自动批改，我们可暂定MVP阶段对纯主观题不自动评分，而是由教师/家长批改。系统提供界面让批改者阅览学生答案，并给出分数和评语。后续优化阶段再考虑NLP模型自动评分的可行性。
- **数理化公式题**：学生可能提交**公式**作为答案（如解方程给出  $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ ）。比较两个公式等价非常具有挑战性。我们采用的方法：
  - 要求学生按照平台提供的公式编辑器或格式来输入答案，尽量结构化（例如使用LaTeX格式输入）。然后利用计算引擎（如SymPy）尝试对学生答案和标准答案进行规范化（simplify）处理，再比较符号表达是否等价。如果等价则判正确<sup>7</sup>。SymPy可以解析代数表达式并化简为标准形式，从而检测如  $(x^2 + 2x + 1)$  与  $(x + 1)^2$  是否相等。
  - 对于化学方程式，判定正确需要检查元素计量是否一致。可通过字符串解析各元素数目比对，或者引入专门的化学公式比对逻辑（mhchem提供了一些解析能力）。MVP阶段化学方程判分可限定为判断是否完全匹配标准答案字符串，暂不考虑同价异构等复杂情况。
  - 需要注意的是，公式自动批改的覆盖面有限，某些情况可能误判。因此平台会注明对复杂推导题建议上传过程供老师人工评分。自动批改只对**最终结果**进行判断，即便学生过程不同，结果正确即可给分。这也意味着如果题目要求填写最后结果，学生只要最终表达式正确就判对，否则判错。
- **评分反馈**：无论何种题型，系统应在学生提交后即时提供反馈。对于客观题，页面上直接标出哪些题对错，并显示正确答案和解析（可设置提交后或截止后才显示解析）。即时反馈有助于学习<sup>18</sup>：“及时了解题目做错的原因”，从而学生可以立刻反思<sup>4</sup>。对于需要人工批改的题，学生提交后状态显示为“待批改”，由老师批改完成后再通知学生查看得分和评语。平台支持在家长/教师批改后自动通知学生，例如在学生端显示“你的作文已批改，点击查看”。
- **成绩计算**：每份作业或练习的总得分按照各题权重汇总计算。通常每题分值均等或根据难度不同赋分，题库中会存储每题的分值权重。系统累加所有题目的得分得到总分，并计算百分比。此百分比用于判定奖励是否达标（如 $\geq 90\%$ ）。此外，对于背单词等练习模块，可采用积分或熟练度而非百分制评分。例如IXL使用SmartScore动态分数机制<sup>19</sup>：学生答题得分会随着正确率变化而非线性增长，但为简化，我们可使用每练习题若全部正确则计为100%达标，否则根据正确率计算百分比。
- **错题收集**：评分过程中，一旦发现学生某题错误，系统将自动将该题记录到错题本模块<sup>4</sup>。如果学生之前已经错过此题，则更新错题本中的统计（错误次数+1等）。当学生后来在错题重练中答对该题，也更新错题状态。如果题目内容有版本更新（管理员修改了题目或答案），错题本中引用的题目也需同步更新或标记可能失效，提示学生。
- **特殊情况**：如学生中途退出练习再回来，系统会保留已作答题目的状态，不重复计分。若作业允许多次尝试提交最高分（比如练习赛），系统记录每次得分并以最高分为准，或者记录最后一次，这些规则由发布者设定。对于开放时间较长的作业，可以让学生在截止前多次修改答案，但自动批改结果只有在截止后才计入成绩，避免学生通过试错获取正确答案。



通过上述逻辑，平台力求对客观题实现高度自动化准确批改，对主观题提供半自动辅助批改，在保证公平性的前提下**减轻人工批改负担**。值得一提的是，目前已经有成熟系统如Gradescope、Khan Academy等在实践中验证了自动评分的效率与准确<sup>20</sup>。我们参考其经验，同时结合自身需求制定这一套评分规则体系。随着技术发展，未来可进一步引入AI技术提升批改质量，例如OCR识别学生手写答案<sup>7</sup>、NLP评估开放问答等，但在MVP阶段，本方案以稳妥的规则算法为主，以确保**评分结果可信可解释**。

## App解锁逻辑与对接建议

“每日学习成果解锁iPad指定App”功能旨在将学习表现与娱乐时间挂钩，形成激励。其实现涉及平台与iPad设备上应用使用权限的联动。以下详细阐述逻辑和可能的技术对接方案：

- **规则判定：**每日针对不同科目设置奖励判定规则。在本方案中，以数学和英语科目练习为例：
  - 每日学生完成数学在线练习（或作业）后，系统计算其得分百分比。如果 $\geq 90\%$ 且未达100%，判定达到“优秀”标准；如果100%满分，判定达到“满分”标准。英语练习同理。
  - 系统每天可对每个科目各计算一次奖励资格。例如学生当天进行了多次数学练习，以最高成绩或最后一次练习计当天成绩。或者明确规定某套题为当天的考核。
- **达标后奖励时长：**优秀（90%以上）对应15分钟娱乐App解锁，满分对应30分钟。如果数学和英语都达标，是否累计奖励（如分别15分钟，总共30）可以由家长配置决定。本方案建议**各科奖励独立**，例如学生数学90分且英语90分，则分别获得15分钟，两项可叠加为30分钟，但同一科目多次满分不重复奖励。
- **奖励记录：**当学生满足条件时，系统后台生成一条奖励记录，写入Rewards表，包含学生ID、日期、科目、获得时长、使用状态等。此记录同时触发通知机制，告知相关家长。例如家长端收到提示：“Michael今日数学练习得分95%，已获得15分钟YouTube使用权”。学生端也可收到动画或奖章提示，以示鼓励。
- **解锁机制：**核心问题是**如何让iPad上指定的App在限定时间内可用**。由于iPad本身的家长控制（Screen Time）功能需要家长设置，web平台无法直接更改设备设置。但可考虑以下方案：
  - **借助第三方App：**例如使用类似ScreenTreat或Kidslox的家长控制应用<sup>11 21</sup>。这些应用允许家长预先设定孩子设备上的任务与奖励。一种思路是我们的平台通过与此类应用的API集成，当学生达标时，由我们的服务器调用家长控制应用的接口，在孩子iPad上解锁对应App一段时间。以ScreenTreat为例，它在孩子设备上控制App使用，家长可以在其中配置任务<sup>11</sup>。我们平台可以通过约定，在ScreenTreat中将每日数学练习映射为任务，由我们来更新任务完成状态，从而ScreenTreat自动执行解锁。这需要ScreenTreat提供开放API并允许第三方触发任务完成。目前不清楚这些App是否提供此能力，但如果有合作可能，这是最自动化的方案。<sup>11</sup>的描述表明ScreenTreat在孩子设备上本地运作，无需远程，但Kidslox等有云端控制，可研究对接他们的“时间奖励”API<sup>22</sup>。
  - **iOS捷径或配置文件：**苹果的屏幕使用时间功能可以通过配置描述文件和Apple MDM（移动设备管理）来远程控制。当设备受监控（Supervised）时，MDM可以下发命令允许或禁止特定App。我们的平台可与家长的MDM服务整合：当奖励触发时，通过家长的管理账号向MDM发送解锁某App的指令，该指令通常可以包含定时（比如15分钟后重新锁定）。这种方式需要家庭使用Apple的“Family”管理或学校使用Apple School Manager设定设备，此实现较为复杂且一般个人用户难以自行架设。不过在学校管理的iPad上，此方案是可行的，由学校IT管理员在Tender环境中部署MDM服务，与学习平台联动。
  - **短信/验证码通知家长手动解锁：**一种相对简单但需要家长配合的方法：当孩子获得奖励时，平台向家长发送一次性解锁码或链接。家长在孩子iPad上运行一个事先安装的“解锁App”（我们可以开发一个简单iOS家长端App），输入该验证码后，即临时取消对某些娱乐App的限制计时15分钟。这个“解锁App”本身需要配置Device Management权限或使用苹果的Guided Access API来管理其他App，这在苹果生态中有诸多限制。但至少可以通过启动一个计时器，时间到后提醒家长关停或自动锁定（若有权

限)。此方案需要我们开发额外应用，增加复杂度，但用户体验较直观，由家长执行解锁确认，安全性高（家长有最终控制）。

- **直接依赖家长监督：**在没有技术对接的情况下，平台可以简单地将奖励信息展示，由家长据此采取行动。例如家长设置iPad的屏幕使用时间密码，约定孩子只有在获取奖励时才请求家长输入密码解锁指定App，并遵守时间限制。平台在家长界面提供一个**计时工具**：点击“开始15分钟YouTube”，平台会开始倒计时并在时间到时给出提示音/通知提醒家长锁定。这实际上是**人为控制**，平台仅辅助计时和记录，不直接干预iPad设置。该方式实现简便，适用于小规模家庭使用，但不算真正技术自动化。作为过渡，MVP阶段可以采用这一方式，因为它无需深度技术集成，只要在UI上做好提示即可满足功能。
- **推荐实现：**综合考虑实现难度与用户体验，建议**MVP阶段采用方案4**（家长手动+平台计时提示）。具体流程：当学生达到奖励条件，家长端会出现“解锁奖励”按钮。家长点击后，选择要解锁的App（下拉列表可配置如YouTube、Minecraft等），然后点击“开始计时”。平台开始倒计时，并显示“X应用已解锁，还有XX:XX分钟”。家长这时手动允许孩子使用iPad上的该App。倒计时结束时，平台发出声音提醒并发送通知提醒家长时间已到，建议关闭App。平台将此奖励使用情况记录（如Michael使用了15分钟YouTube，结束时间几点几分）。若家长希望暂停或提前终止，也可在界面点击“结束解锁”，平台立即提醒收回设备。

在后期优化阶段，我们可以朝**方案1或2**努力，与成熟的家长控制系统集成以实现全自动解锁。比如联系ScreenTreat厂商合作，将我们平台学习数据接入它的任务模块，实现真正“完成任务自动发放屏幕时间”<sup>11</sup>的闭环体验。Kidslox也提供“Screen Time Rewards”功能鼓励做作业<sup>22</sup>，未来也可探讨API对接。届时家长几乎无需手动操作，孩子达到条件后iPad自动弹出通知“你已获得15分钟游戏时间”，点确认后直接允许游戏App运行15分钟，到点再锁定——这将是体验最佳的形态。

- **安全与滥用防范：**需考虑防止学生通过作弊获取额外奖励时间。例如学生可能反复练习简单题以多次获得奖励。对此，系统应**限制奖励频率**：每科每天只有一次奖励机会，不累积超过上限（比如每天最多30分钟，即使多科满分也不超过30，或数学英语各不超30这样设置需明确）。还可以增加判定：必须完成规定的练习任务（题量不少于N道，或用时不少于M分钟）才有资格获取奖励，以防学生跳过大量题直看答案拿高分。管理员可调整这些策略参数。平台日志也会记录每次奖励的获得和使用情况，供家长审查。家长若发现异常（比如孩子短时间内多次“满分”不正常），可介入调查。
- **扩展性：**虽然目前例子是解锁娱乐App时间，未来可扩展为多种奖励形式，例如：在平台内给予虚拟奖励（徽章、排行榜加分），或解锁现实奖励（比如打印一张奖状）。技术上可以加丰富的奖励体系。而对于设备管控奖励，也不仅限于iPad，理论上安卓平板、手机等都可考虑，但各自对家长控制的实现不同。由于本方案聚焦iPad，我们先把iOS生态跑通。

综上，App解锁功能将在MVP中以**家长参与的形式**实现，确保规则和计时到位；在未来优化时，力争做到**自动触发和收回**。无论哪种形式，关键是将其与学习平台紧密结合，让孩子明白“努力学习就能赢得玩乐时间”，达到寓教于乐的目的。这一功能也将成为平台的一大卖点，迎合很多家长希望孩子先学习再娱乐的教育理念，有助于提高用户黏性和参与积极性<sup>11</sup>。

## UI/UX设计风格建议

平台的用户界面设计需要兼顾3年级（约8岁）到12年级（18岁）这一**广泛年龄段**用户的使用体验。UI/UX风格既要**对低龄学生有吸引力**，又不能让高中生觉得幼稚，同时满足家长和管理员的专业性需求。在设计时应遵循以下原则和建议：

- **简洁直观的导航：**整个网站应具有清晰的导航结构，让不同用户都能快速找到所需功能。采用**顶栏或侧边栏菜单**区分主要模块，例如“单词练习”、“作业”、“错题本”、“奖励”、“统计”等。对学生而言，提供**仪表盘**概览本日任务和进度；对家长，提供切换孩子视图、查看报告的清晰入口；对管理

员，有独立的管理菜单。遵循“三次点击”原则：用户在不超3次点击内应能触达核心功能页面<sup>23</sup>。使用下拉菜单、面包屑导航等辅助定位，并提供搜索功能（如题目搜索）。高年级学生和成人用户对效率要求高，简洁的导航能减少迷惑和挫败感。

- **响应式与跨设备优化**：UI设计需充分考虑iPad等平板的使用。采用响应式布局，确保在平板竖屏、横屏以及不同分辨率电脑上呈现良好。特别地，**移动端友好**非常关键：调查显示约75%的学生倾向于用移动设备访问学习网站<sup>24</sup>。按钮和点击区域要足够大，方便触屏操作；字号和间距适配小屏阅读（正文字体在移动端至少14px以上以保证可读性）<sup>25</sup>。首页和登陆页在移动端应简洁居中。测试UI在常见的iPad尺寸（10.2寸等）和浏览器上确保没有布局错乱。
- **面向儿童的设计元素**：对于低年级学生（小学阶段），界面可以加入一些**趣味和卡通元素**来增强吸引力。例如使用活泼的配色（明亮的蓝、绿、橙等）、可爱的图标或吉祥物形象。可以设计一个友好的**虚拟助手角色**（如一只卡通考拉或袋鼠，呼应澳洲特色），在界面中引导孩子操作，如第一次使用时由角色气泡提示点击哪里<sup>26</sup>。动画和音效也可适当运用：比如答对题目时弹出笑脸动画、积分增加的声音，完成学习目标时播放小彩带动画庆祝等。这样增加互动性，帮助孩子保持兴趣<sup>27 28</sup>。不过，需把握度，不要让动画分散注意力或过于嘈杂。儿童用户还喜欢**收集和成就**，UI上可以体现诸如积分、徽章、连续学习天数等激励元素，鼓励孩子持续使用<sup>29</sup>。
- **针对青少年的专业性**：对于高年级学生（中学生），界面应相对**成熟简洁**，避免过度卡通化以免让他们觉得幼稚。可使用稍偏稳重的配色和更简约的图标风格，同时保留现代感（例如扁平化设计）。比如整体配色可以采用学校常见的蓝色或绿色主调，并通过深浅搭配来区分模块区域（背单词模块可能色调活泼些，作业模块则稍严肃）。**一致性的设计模式**很重要：青少年和成人一样欣赏一致且熟悉的界面<sup>30</sup>。不要为迎合孩子而设计完全另类的UI，常见的按钮样式、页面布局仍然适用于他们。设计时可以**像对待成年人一样对待青少年**，给予他们简洁、有逻辑的界面，同时用细节体现关怀（例如可更换的主题颜色以满足个性）。高中生界面上不妨减少卡通元素，改为展示学习数据的**图表**、知识结构树等，让他们感觉这个平台是帮助学业的严肃工具。
- **色彩与字体**：选择清新明快但不过于花哨的配色方案。可采用**扁平化的色彩搭配**，例如主色选用蓝色（信任、安全之意）或橙色（活力），辅以中性色（白、灰）做背景。不同科目可以用不同颜色区分（如数学绿、英语蓝、科学橙等），方便学生快速联想。字体方面，正文使用易读的无衬线字体，英文字体如 Arial, Verdana 等，对儿童可考虑稍圆润的字体。中文界面下（虽然平台内容主语言为英文，但若界面提供中文翻译给家长使用也需考虑）可用苹方等清新字体。儿童界面字体建议稍大，14-16px以上<sup>25</sup>，年长学生可采用12-14px。标题和重要文字可以适当加大或加粗突出。需要注意对**色盲友好**和**高对比度**：避免红绿组合辨识答案正确/错误（可加图标辅助），背景和文字颜色对比充足以提升可读性。尽量遵循WCAG无障碍准则，虽然主要用户是儿童，但也可能有视觉障碍学生。
- **布局与控件**：界面布局应该**简洁不繁杂**。每个页面只突出当前任务，减少干扰项。例如在背单词页面，只显示图片、单词和选项，不夹杂额外广告或过多按钮，让孩子专注学习。用**卡片式布局**展示内容块，使界面模块化清晰。控件选择上，考虑儿童的操作特点：年幼孩子可能操作精度差，因此按钮、选择框要大而明显<sup>31</sup>；拖拽等复杂交互尽量简化或在教程中教给他们。加入**持续反馈机制**<sup>28</sup>：孩子每完成一个小任务，界面给予即时的正面反馈，比如进度条前进、获得小星星奖励等，强化成就感。对于需要等待的操作（如CSV导入或题库加载），给出进度提示动画防止孩子不耐烦。
- **多用户界面切换**：因为家长也使用本平台，UI需在**儿童模式**和**家长管理模式**之间切换良好。这可能采用不同的配色或布局。家长和管理员界面应更偏向**工具型**风格，重点在数据展现和管理功能，减少卡通元素。比如家长查看报告页面，用图表、表格清晰呈现成绩和错题统计，色调可以更稳重商务风（深蓝、灰）。而当家长切换至孩子练习界面时，又可以看到孩子视角下的学习内容预览。实现上，可以做**角色识别**：登录后根据角色加载不同CSS主题或布局方案。确保家长使用时不会觉得像儿童玩具界面，也确保孩子使用时界面有趣而非枯燥。

- **本土化风格**：考虑澳大利亚的文化元素融入设计。例如界面插画风格可以使用澳洲特色的卡通动物（考拉、袋鼠）或风景，让本地用户产生亲切感。语言方面虽然主要内容英文，但界面操作提示可以用简单易懂的英文，针对低年级可配图标解释。若需要，也可提供中英文界面切换供华人家长使用，但默认仍以英文为主以符合澳洲环境。
- **示例借鉴**：可以借鉴一些优秀教育类产品的设计。比如IXL的网站设计较简洁，使用了明亮的色块和插画但整体专业有序<sup>32</sup>；百词斩App界面卡通感强，采用了情景插图和大字号的单词显示，非常直观<sup>2</sup>。我们可以融合二者优点：**保持功能布局的简明**（不让用户迷失），同时**注入视觉趣味**（不让孩子觉得枯燥）。也可参考通用的儿童网站设计原则，如素马设计总结的5条儿童UX原则：<sup>26</sup> <sup>33</sup>——针对年龄设计、精选色彩字体、引入友好角色、持续反馈奖励、极致直观。这些原则我们都已在上文体现，如分年龄风格、用卡通助手、小挑战小奖励不断刺激，以及尽量采用通用UI模式不要让孩子迷惑<sup>33</sup>。

总之，UI/UX设计应时刻围绕**提高学习投入度**和**降低操作难度**两个目标展开。让孩子在使用平台时感觉操作顺畅、有趣、有成就感<sup>33</sup>；让家长 and 教师觉得界面专业、易于获取所需信息。通过精心打磨视觉和交互，本平台将不仅功能强大，而且令人“爱用”、“愿用”，真正成为师生家长日常离不开的学习伙伴。

## 技术栈推荐（网页端）

考虑到平台需要在网页端实现丰富的交互功能，并兼顾性能和可维护性，下面推荐适合本项目的技术栈：

- **前端（Web 客户端）**：使用现代的前端框架提高开发效率和用户体验。推荐使用 **React** 框架搭配 TypeScript。React拥有大量成熟的组件库和社区支持，非常适合构建动态交互界面和SPA（单页应用）。TypeScript则提供类型检查，提高代码健壮性。在UI组件库上，可选用 **Ant Design** 或 **Material-UI** 这类成熟组件库，方便构建响应式布局和常用控件（表单、表格、图表等）。Ant Design风格简洁且有中文支持，也贴和管理后台需求；Material-UI符合谷歌Material Design规范，在教育场景中也常用。考虑用户主要在iPad浏览，框架需适配触屏事件，React在这方面通过HTML5标准事件即可支持。React Router用来管理多视图页面的路由。状态管理可用 **Redux** 或 **React Context** 结合 **Hooks** 简化数据流。对于富文本和公式输入，可以集成开源的 **富文本编辑器**（如 **Draft.js** 或 **TinyMCE**）以及 **MathQuill**（用于所见即所得地输入数学公式），确保教师录入题目方便。前端还需引入 **MathJax/KaTeX** 用于渲染公式<sup>5</sup>。此外，为实现实时交互（如作业定时、即时反馈），可以使用 **WebSocket** 或采用 **React Query** 轮询接口更新数据。
- **后端（服务器端）**：后端需要处理API请求、业务逻辑和数据库操作。可采用 **Node.js** 平台，搭配 **Express.js** 或 **Nest.js** 框架。Node.js使用JavaScript/TypeScript，与前端共享语言，团队协作和代码复用方面有优势。Express是轻量灵活的Web框架，Nest则提供更结构化的模块和依赖注入机制，适合大型项目。如果团队有Python经验，也可考虑 **Django** 或 **Flask** 框架。Python在科学计算和AI方面优势明显，将来做自动批改AI可直接利用。但综合看，Node.js生态在实时应用、JSON处理、并发上表现出色，更贴合web服务场景。可以在Node后台集成一些现成服务，如 **Passport.js** 做用户认证、中间件处理权限校验等。
- **数据库**：采用 **MySQL** 或 **PostgreSQL** 作为关系数据库管理系统。MySQL广泛使用且Tender环境可能已有支持，PostgreSQL功能更强大支持地理和JSON等，但本项目关系型特征明显，两者均可胜任。使用 **Sequelize**（针对Node的ORM）或 **TypeORM** 做数据层封装，将关系表映射为对象，方便操作。同时可利用数据库的存储过程或事务机制确保比如发布作业和写入题目这些复杂操作的完整性。对于高并发读写，可以配置数据库连接池并优化常用查询。后期若数据量暴增，可考虑分库分表或读写分离策略，但MVP阶段单一数据库实例足够。
- **缓存/消息队列**：使用 **Redis** 内存数据库。Redis一方面可缓存热点数据（如热门题目、配置），另一方面可以用它的Pub/Sub或List实现**简易消息队列**，用于处理异步任务（如CSV批量导入的后台任务、发

送通知邮件/短信等)。Redis还可用于存储用户会话(如果不用JWT而用sessionID的话)、记录在线用户状态等。

- **搜索引擎(可选)**: 如需支持站内搜索大量内容,建议引入 **Elasticsearch**。比如按关键词搜题目、知识点等,用搜索引擎能更快返回结果并支持模糊查询。初期可不启用,当内容量上来后再集成。
- **文件存储与CDN**: 部署上,可以直接在Tender环境的服务器文件系统存媒体文件。如果Tender有内置对象存储服务,则直接对接。否则推荐使用 **AWS S3** 或等价的云存储,结合 **CloudFront** 等CDN加速文件分发,提高图片音频加载速度。前端通过域名访问CDN资源,后端仅负责签名上传或提供文件URL。不使用云服务的话,也可自己配置Nginx做静态文件服务器并启用缓存头。
- **第三方服务集成**:
  - 邮件发送:使用 **SendGrid** 或 **Mailgun** API发送注册验证、重置密码、报告邮件等。
  - 短信/通知:可用 **Twilio** 发送短信通知家长奖励信息等。苹果Push通知需要有移动App才可用,如果后期做iOS家长App可申请APNs推送。
  - AI算法:如需简答题自动评分,可调用 **OpenAI API** 或谷歌的 NLP 模型服务,将学生答案和标准答案交给AI判分。但短期内可能不考虑接入。
- **开发与部署**: 开发时使用 **Git** 进行版本管理,前后端分别仓库或monorepo都可。构建工具对前端用Webpack/Vite来编译打包React代码。测试方面,前端可用 **Jest + React Testing Library** 做单元测试,后端用 **Mocha/Chai** 或 **Jest** 进行API和逻辑测试。部署时,在Tender环境下如果支持容器,可将前后端打包为Docker镜像部署。前端也可构建后作为静态文件,由Nginx托管。使用 **PM2** 或 **forever** 等工具守护Node进程防故障。数据库部署考虑主从或云数据库服务(如AWS RDS)提升可靠性。整套系统上线前需要在Tender环境配置好域名、HTTPS证书等,保证访问安全。
- **安全与优化**: 全站强制HTTPS;后端防范常见攻击如SQL注入(ORM默认处理参数化查询)、XSS(对用户输入字符实体转义)、CSRF(对于敏感操作要求Token或使用SameSite Cookie策略)。启用HTTP缓存策略,减少静态资源重复加载。合理利用前端的状态管理降低API请求频率,比如错题本本地缓存等。对算法复杂的操作(如批改表达式等)在后端实现时注意性能,可用C++扩展或WebAssembly(例如KaTeX就是高性能渲染引擎)。随着用户量上升,可以考虑使用负载均衡分发请求、多台服务器部署,以及CDN缓存教材内容等手段扩展。

**技术栈小结**: 本方案推荐 **前端: React + AntD/Material-UI + MathJax, 后端: Node.js (Express/Nest) + MySQL**, 辅以 **Redis** 和若干第三方服务。这套栈在业界较为常见,成熟稳定,团队容易招聘到相关开发人员。同时,它完全支持网页端,用户无需安装客户端,仅通过浏览器即可获得流畅的交互体验。

## 项目开发阶段规划(MVP→测试→优化)

为了稳步推进项目建设,我们将分阶段实施开发。各阶段目标如下:

**1. 最简可行产品(MVP)开发**: 预计用时~3-4个月。目标是尽快构建核心功能可用版本,供小范围用户试用反馈。

- **MVP范围**:
  - 用户注册/登录和基础角色管理。预先创建示例用户Michael、August、Grace、Dan等用于测试。
  - 学生主页展示当日任务列表(如需完成的练习、作业)。

- **背单词模块初版**：支持学生查看单词卡片，带图片、释义、例句<sup>1</sup>。实现基本的学习流程：随机抽取一定数量单词做选择题，提交后显示正确单词释义。记录学生的掌握情况（可简单记录正确/错误，下次再练时重复错词）。暂不实现复杂的记忆曲线算法，但可做好接口以备后续引入。
- **作业布置与提交**：教师/家长能通过Web表单创建一份简单作业，填写题目（先支持选择题和填空题），发布给指定学生。学生登录后看到作业，在线作答提交。系统对选择题自动判对错并给出每题得分，填空题简单字符串匹配。总分统计并存储。
- **自动批改初步**：实现对选择题100%自动判分，对数值填空题支持基本的数值比对（精确或固定误差范围），对文本填空支持大小写不敏感匹配。主观题在MVP中可以不出现或当作附件形式由家长线下批改。**公式渲染**在此阶段需打通，让题目和答案显示正确公式符号<sup>5</sup>。
- **错题本初版**：学生提交作业后，将答错的客观题记录到错题表。提供一个界面列表展示错题（题干、正确答案、本人答案），暂不实现在线重练功能，但至少能查看整理的错题集。
- **CSV导入**：MVP中可以实现**单词表**的CSV导入功能作为验证。提供后台管理界面让管理员上传单词CSV，解析入库<sup>10</sup>。题目和用户的批量导入可以暂缓，用手动方式添加几条测试数据即可。
- **奖励机制**：实现基础的奖励规则判定和提示。不涉及真实App控制，仅在学生提交练习/作业后，如果满足条件，在界面弹出“恭喜达标XX，可让家长解锁15分钟XX应用”的提示，并在家长端生成一条通知。家长端界面提供一个按钮“确认奖励已兑现”，点击后记录奖励发放（具体发放过程由家长线下执行，MVP不强制控制iPad）。相当于此阶段先把奖励**显示**层面完成，技术对接留到后期。
- **UI设计**：初版界面完成基本布局和导航。遵循之前的设计思路，做到主要页面可用但不求完美细节。先以简单清新的风格为主，插画和卡通元素可后续补充。保证移动端自适应正常。提供英语界面文本。
- **技术验证**：在Tender环境实际部署跑通端到端流程。确保用户可以通过浏览器访问网站，完成从登录到练习提交的闭环。验证公式渲染在浏览器端OK<sup>5</sup>、数据库读写正常、文件上传（如单词图片）正常显示等关键点。
- **MVP交付**：在内部或小范围试用。邀请2-3名目标年级学生及其家长试用（用Michael等账户），收集基础体验反馈。重点关注功能是否正确、易用性问题、任何严重bug。如错题本是否正确记录错题，自动批改是否有明显错误判定，UI在iPad上操作是否流畅等。

**2. 测试完善阶段**：预计用时 ~1-2 个月。目标是基于MVP反馈修复问题，完善细节，并进行更大范围测试和性能调优。

- **问题修复与改进**：整理MVP试用反馈，修复发现的bug。例如：公式渲染错误情况修正；自动判分误判情况调整匹配逻辑；导航不清晰处调整；移动端样式错乱问题修复等。完善错题本界面，增加错题解析显示<sup>4</sup>；实现错题重练的循环出题功能<sup>8</sup>（至少基本可以让学生重新练一遍之前错题）。
- **扩充功能**：在MVP基础上补充一些次要但重要的功能：
- 作业模块增加定时和截止控制（过期不能提交等）。
- 加入**家长审批**流程：如家长上传的公共题需要管理员审核通过的功能框架（后台界面简易实现）。
- **用户账号管理**前端界面：允许管理员/家长修改密码、学生修改头像等个人设置。
- **日志和通知**：实现基本的通知中心，让家长可以收到例如“孩子提交了作业”“获得了奖励”等消息提示（前端弹窗或通知列表）。
- **性能与安全测试**：模拟一定量用户并发提交，看看系统负载。优化SQL查询（加索引）和前端渲染性能。测试文件上传的大文件情况。进行安全扫描，修补漏洞（如尝试SQL注入、XSS在文本框输入，确保被正确过滤）。
- **更多测试**：开展更大范围的用户测试。如果有条件，可让一个小型补习班的老师家长参与测试，获取不同用户视角的意见。高中生和小学生各找一两位测试，看界面风格是否都能接受，并根据反馈在UI上做针对性的调整（比如字体过小则调大，低龄模式卡通元素是否需要更多等）。
- **文档完善**：在此阶段，完善项目的文档，包括开发文档和用户文档。开发文档记录各模块设计细节和接口，方便后续开发者。用户文档（或帮助指南）则供家长老师参考，例如如何上传CSV、如何查看错题本等。

通过测试完善阶段，我们希望达到**功能完整、稳定性提升**的版本，可对外小规模上线试运行。

**3. 优化提升阶段：**预计用时 ~2-3 个月（可迭代进行）。目标是针对上线后的真实使用情况，进行功能增强和优化，使产品更加智能好用，具备一定领先优势。

- **功能优化：**
  - **自动批改优化：**引入更智能的判分机制。探索将 **AI技术** 融合进来：例如调用机器学习模型对简答题进行评分建议，对主观题可以训练一个评分模型参考。对于数学步骤题，调研可否识别学生的多步推导过程给予部分分（这可能需要OCR+CAS，技术复杂度高，酌情考虑）。另外，进一步完善公式比对的算法，使更多等价情况判为正确 <sup>7</sup>。
  - **错题本优化：**增加错题分类和知识点标签。系统对每道题关联课程知识点（澳洲课程标准）<sup>16</sup>，错题本可以按知识点聚合，方便学生重点复习薄弱环节。实现错题本的统计可视化，如错题数量随时间的下降曲线，错题击败率等 <sup>34</sup>。另外，可以加入**错题提醒**：一段时间后（例如一周）提醒学生回顾尚未消灭的错题，形成**循环复习**。
  - **奖励机制自动化：**与**家长控制App**对接。在优化阶段重点攻关与ScreenTreat或Kidslox的技术合作 <sup>11</sup><sup>22</sup>。如果能拿到API，则开发服务器到第三方的对接模块，实现真正的一键远程解锁。也可以开发**家长移动端App**：该App与网站账户联动，收到奖励信息后通过本地配置文件修改实现临时放行应用。这需要iOS开发技能和苹果审核，可根据项目资源决定是否执行。如果移动端App暂不做，也可以优化网站在手机Safari上的体验，让家长可以直接用手机浏览器操作（Screen Time很多设置只能App内做，这里可能有限）。
  - **学习资源扩充：**将内容规模扩大，比如引入**澳洲本地课程教材**对应练习题。可与内容供应方合作或由教研团队编写题库，利用CSV批量导入功能将上千道题目、上万词汇导入平台。在功能上也相应增强浏览查找：实现**题库管理**界面，老师可以根据年级和知识点选择题目组卷。词汇模块增加分级词汇书选择（小学词汇、中学雅思词汇等）<sup>35</sup>供不同程度学生使用。
- **社交和互动：**为了提升学习动力，可考虑加入一些社交互动元素，例如：
  - **排行榜：**每周展示练习积分排行榜，激发竞争（可按班级或全站）。
  - **班级功能：**支持教师创建班级邀请学生加入，班级成员间可以适度竞争或合作。家长之间也可有社区交流（比如分享学习心得论坛）。
  - **成就系统：**设计若干徽章，如“单词达人：背完1000单词”、“满分之星：连续5次作业满分”等，学生达成条件即获得徽章或称号，显示在个人主页，增加荣誉感 <sup>19</sup>。
  - **学习提醒与计划：**增加日历或打卡功能，提醒学生每日学习并记录连续学习天数等，配合奖励制度形成自律习惯 <sup>36</sup>。
- **性能扩展：**如果用户量上升，需优化系统架构：
  - 部署方面，上线多台服务器，前端与后端分离部署，通过Nginx做负载均衡；数据库如有大量读请求，可增设从库分担查询；缓存进一步用来减轻数据库压力（缓存常查询的字典数据，首页统计等）。
  - 前端优化方面，利用代码分拆和懒加载减少首屏加载时间；对大量数据的界面（如错题列表、题库列表）加入分页或虚拟列表技术，避免一次性渲染太多DOM。
  - 使用监控工具持续跟踪服务器CPU/内存，前端页面性能，并针对瓶颈优化代码或升级服务器配置。
- **迭代发布：**优化阶段可按**2周一迭代**方式渐进发布新功能。每次小版本更新先在测试环境验证，然后发布生产环境。保持与用户的沟通，告知新功能使用方式。收集用户反馈持续改进。比如推出自动解锁功能后，要收集家长对其可靠性的反馈；推出排行榜后关注是否带来负面压力等。
- **最终目标：**在优化阶段结束时，平台应达到较高的完善度：**功能全面**（覆盖从内容学习、练习、评测、巩固全学习环节），**体验优秀**（UI友好、响应迅速），**效果显著**（学生成绩和参与度提升，有数据支持）。平台与澳洲课程深度对齐 <sup>16</sup>，赢得教师家长信任，具备推广到更多学校或个人用户的条件。

- **长期规划：**虽然不属本方案范围，但值得一提的还有未来可以扩展的方向，如增加更多学科（科学、历史等）内容、推出移动App供随时练习、甚至考虑AI个性化辅导（根据错题与薄弱点推荐练习）。这些可以作为更长期的roadmap，为平台预留接口和扩展性。

按照以上阶段划分推进，能够确保项目**有序进行**：先搭建基础，再打磨细节，最后增强竞争力。每阶段都有明确的验收标准和目标用户反馈，从而降低开发风险。项目管理上将采用敏捷开发方法，每两周Sprint评审迭代成果，确保团队始终朝着最终愿景迈进。相信通过科学的规划和执行，本平台能如期实现，从MVP逐步发展为一个成熟的在线学习与作业平台，帮助澳洲3-12年级学生更好地学习成长。 2 11

---

1 35 36 App Store 上的“百词斩-学外语、背单词必备”

<https://apps.apple.com/cn/app/%E7%99%BE%E8%AF%8D%E6%96%A9-%E5%AD%A6%E5%A4%96%E8%AF%AD-%E8%83%8C%E5%8D%95%E8%AF%8D%E5%BF%85%E5%A4%87/id557545298>

2 3 大学生背单词，百词斩的效果如何？ | 人人都是产品经理

<https://www.woshipm.com/evaluating/3923886.html>

4 8 9 18 34 在线考试系统的“错题集”有多重要？ - 优考试\_在线考试系统\_新闻资讯

<https://www.youkaoshi.cn/news/yksabn/>

5 6 13 14 Writing Formulas and Equations (LaTeX) for Assignments – Gradescope Guides

<https://guides.gradescope.com/hc/en-us/articles/22236676385165-Writing-Formulas-and-Equations-LaTeX-for-Assignments>

7 AI-Assisted Grading and Answer Groups – Gradescope Guides

<https://guides.gradescope.com/hc/en-us/articles/24838908062093-AI-Assisted-Grading-and-Answer-Groups>

10 How to import content from CSV | Taalhammer

<https://www.taalhammer.com/how-to-import-content-from-csv/>

11 Parental Control - ScreenTreat on the App Store

<https://apps.apple.com/lt/app/parental-control-screentreat/id6701426470>

12 19 32 What is IXL and How Does it Work? What's New? | Tech & Learning

<https://www.techlearning.com/how-to/what-is-ixl-and-how-does-it-work>

15 16 IXL - Australia maths curriculum

<https://au.ixl.com/standards/maths>

17 Using AI to Provide Student Feedback in Math Education

<https://new.assistments.org/blog-posts/the-future-of-mathematics-education-using-artificial-intelligence-to-provide-efficient-feedback>

20 What's the best math auto grading solution, is anyone using it? : r/matheducation

[https://www.reddit.com/r/matheducation/comments/tsxxvt/whats\\_the\\_best\\_math\\_auto\\_grading\\_solution\\_is/](https://www.reddit.com/r/matheducation/comments/tsxxvt/whats_the_best_math_auto_grading_solution_is/)

21 22 Parental Control App - Kidslox on the App Store

<https://apps.apple.com/us/app/parental-control-app-kidslox/id914825567>

23 24 Best School Websites (2025) – Top Design Examples & Features

<https://morweb.org/post/best-school-websites>

25 26 27 28 29 30 31 33 面向儿童：如何创建孩子们会喜欢的UX设计

<https://www.sumaarts.com/share/1326.html>