# YUKE 普拉提小程序详细开发文档 V1.0

本文档基于《YUKE 普拉提小程序产品文档 (PRD) V1.1》，旨在为开发人员提供详细的技术与功能实现指导。

## 1. 需求规格说明书 (Software Requirements Specification - SRS)

本部分将PRD中的功能需求，转化为更精确、无歧义的开发规格。

### 1.1 功能性需求

FR-1: 用户模块

FR-1.1: 系统应支持通过微信授权一键登录，并自动创建用户账户。

FR-1.2: 用户首次登录后，系统应为其生成一个唯一的、非自增的YUKE用户ID（例如：2507029845133D）。

FR-1.3: 用户可以在“我的”页面绑定/解绑手机号码，手机号需作为可选的登录凭证。

FR-1.4: 用户可以查看自己的账户余额、会员卡（类型、剩余次数/有效期）、代金券（面额、使用条件、有效期）。

FR-1.5: 用户可以对账户余额进行充值（需集成微信支付）。

FR-2: 课程与预约模块

FR-2.1: 系统应能按日期展示课程表，课程信息应至少包含：课程名称、等级（星级）、起止时间、授课教练、价格/消耗课时、容量、当前预约人数。

FR-2.2: 用户可以选择一个未满员且未开始的课程进行预约。

FR-2.3: 预约时，系统应优先检查用户是否拥有适用的会员卡（次卡），若有则扣除课时；若无，则检查并扣除账户余额。

FR-2.4: 用户可以在“我的预约”中查看所有状态为“已预约”的课程，并可在课程开始前X小时（后台可配置，默认2小时）取消预约。

FR-2.5: 取消预约后，系统应自动返还扣除的课时或金额至用户账户。

FR-2.6: 课程结束后，系统应自动将“已预约”的记录状态更新为“已完成”。

FR-3: 训练与社区模块

FR-3.1: 系统应自动记录用户每次“已完成”的训练，并累加总训练次数和总时长。

FR-3.2: “训练”页面的日历应高亮显示所有有“已完成”训练的日期。

FR-3.3: “社区”页面的排行榜应能按“本月”和“上月”两个维度，展示用户的累计消耗卡路里（卡路里数据需在后台课程中预设）。

### 1.2 非功能性需求

NF-1 (性能): 95%的API请求响应时间应在500ms以内；小程序页面主要内容加载时间应在2秒以内。

NF-2 (安全性): 用户敏感信息（如手机号）在数据库中需加密存储；所有API接口需进行身份验证，防止未授权访问；需防范SQL注入、CSRF等常见Web攻击。

NF-3 (可用性): 系统应保证7x24小时可用，可用性不低于99.9%。

NF-4 (兼容性): 小程序应兼容近两年主流的iOS和Android系统版本的微信客户端。

## 2. 交互与 UI 设计文档

本部分描述核心页面的用户流程与视觉规范。

### 2.1 设计理念与规范

风格: 现代、简洁、专业、有亲和力。

主色调:

品牌色/强调色 (用于按钮、选中状态): #4A55A2 (沉稳的深蓝色)

主背景色: #F7F8FA (非常浅的灰色，避免纯白刺眼)

主文字颜色: #1A202C (深灰色)

辅助文字颜色: #718096 (中灰色)

成功/提示色: #28A745 (绿色)

错误/警告色: #DC3545 (红色)

字体:

中文: PingFang SC (iOS), Microsoft YaHei (Windows)

英文/数字: Montserrat 或类似的无衬线字体，保证数字清晰易读。

圆角: 所有卡片、按钮均使用 8px 的圆角，营造柔和的视觉感受。

### 2.2 核心用户流程 (User Flow)

示例：新用户首次预约课程流程

进入小程序:

-> 微信弹出授权请求（头像、昵称）。

-> 用户点击“允许”。

-> 进入【预约】首页。

浏览与选择:

-> 在【预约】页查看今日课表。

-> 左右滑动切换到“明天”的日期。

-> 看到一节“FULL BODY Level1”团课，点击【预约】按钮。

预约确认:

-> 弹出一个从底部滑出的确认面板（Action Sheet），显示课程详情（名称、时间、教练、扣费方式）。

分支1 (无会员卡/余额不足): 提示“会员卡/余额不足”，按钮变为【去购卡/充值】，点击后跳转到相应页面。

分支2 (有可用资产): 提示“将扣除1次课时 / ¥298.00”，按钮为【确认预约】。

完成预约:

-> 用户点击【确认预约】。

-> 系统显示一个短暂的“预约成功”的Toast提示。

-> 页面上该课程的按钮变为“已预约”的灰色状态。

-> 用户可在【我的】->【我的预约】中看到该条记录。

## 3. 技术架构与开发规格

### 3.1 技术架构图

+------------------+ (HTTPS) +------------------------+ +--------------------+

| | <---------------> | |---->| 腾讯云TencentDB |

| 微信小程序端 | API请求 | Node.js 后端服务器 | | (MySQL) |

| (Vant Weapp) | | (Express / Koa) | +--------------------+

| | <---------------> | |

+------------------+ +------------------------+ +--------------------+

|-------------------->| 腾讯云COS |

| | (图片/静态资源) |

| +--------------------+

|

| +--------------------+

|-------------------->| 微信支付API |

+--------------------+

### 3.2 数据库核心表结构设计 (ER Model)

-- 用户表

CREATE TABLE `users` (

`id` BIGINT AUTO\_INCREMENT PRIMARY KEY,

`user\_id\_str` VARCHAR(20) UNIQUE NOT NULL COMMENT 'YUKE用户ID',

`openid` VARCHAR(128) UNIQUE NOT NULL,

`nickname` VARCHAR(255),

`avatar\_url` VARCHAR(1024),

`phone\_number\_encrypted` VARCHAR(255) COMMENT '加密后的手机号',

`balance` DECIMAL(10, 2) DEFAULT 0.00 COMMENT '账户余额',

`created\_at` TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

`updated\_at` TIMESTAMP DEFAULT CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP

);

-- 课程定义表

CREATE TABLE `classes` (

`id` INT AUTO\_INCREMENT PRIMARY KEY,

`name` VARCHAR(255) NOT NULL,

`level` INT COMMENT '难度等级1-5',

`description` TEXT,

`duration\_minutes` INT NOT NULL,

`calories\_burned` INT COMMENT '预估消耗卡路里'

);

-- 课程安排表 (排课表)

CREATE TABLE `schedules` (

`id` BIGINT AUTO\_INCREMENT PRIMARY KEY,

`class\_id` INT,

`instructor\_id` INT,

`start\_time` DATETIME NOT NULL,

`end\_time` DATETIME NOT NULL,

`capacity` INT NOT NULL,

`booked\_count` INT DEFAULT 0,

`price` DECIMAL(10, 2), -- 现金价格

`session\_cost` INT DEFAULT 1, -- 消耗课时

`status` VARCHAR(20) DEFAULT 'SCHEDULED', -- 可选: SCHEDULED, CANCELED

`version` INT DEFAULT 0, -- 乐观锁版本号

FOREIGN KEY (`class\_id`) REFERENCES `classes`(`id`)

);

-- 预约记录表

CREATE TABLE `bookings` (

`id` BIGINT AUTO\_INCREMENT PRIMARY KEY,

`user\_id` BIGINT,

`schedule\_id` BIGINT,

`status` VARCHAR(20) NOT NULL, -- 可选: BOOKED, CANCELED, COMPLETED

`payment\_type` VARCHAR(20), -- 可选: BALANCE, CARD

`amount\_paid` DECIMAL(10, 2),

`sessions\_deducted` INT,

`created\_at` TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (`user\_id`) REFERENCES `users`(`id`),

FOREIGN KEY (`schedule\_id`) REFERENCES `schedules`(`id`),

UNIQUE KEY `user\_schedule\_unique` (`user\_id`, `schedule\_id`)

);

### 3.3 API接口规格 (示例)

获取指定日期的课表:

Endpoint: GET /api/v1/schedules

Query Params: date=YYYY-MM-DD

Success Response (200 OK):

{

"code": 0,

"message": "Success",

"data": [

{

"scheduleId": 101,

"className": "FULL BODY Level1",

"level": 3,

"startTime": "2025-07-03T10:00:00",

"endTime": "2025-07-03T11:00:00",

"instructorName": "Eileen",

"capacity": 10,

"bookedCount": 8,

"price": 298.00,

"userBookingStatus": null // null, "BOOKED", "COMPLETED"

}

]

}

创建预约:

Endpoint: POST /api/v1/bookings

Request Body:

{

"scheduleId": 101

}

Error Response (400 Bad Request):

{

"code": 4001,

"message": "课程已满员"

}

## 4. 功能模块详细说明

模块：课程预约后端逻辑 (使用乐观锁)

接收请求: 后端API POST /api/v1/bookings 接收到前端发来的 scheduleId 和从Token中解析出的 userId。

启动事务: 为了保证数据一致性，启动一个数据库事务。

数据校验 (第一部分): a. 查询 schedules 表，获取 scheduleId 对应的课程信息，包括 version 字段。 b. 检查课程状态: 验证该课程是否为 SCHEDULED 状态，时间是否晚于当前时间。若否，回滚事务并返回错误。 c. 检查用户是否重复预约: 查询 bookings 表，检查该 userId 是否已有该 scheduleId 的 BOOKED 记录。若有，回滚事务并返回“请勿重复预约”错误。

扣费/扣课时: a. 检查用户是否有适用的会员卡（次卡）。若有且次数 > 0，则扣减一次。记录支付方式为 CARD。 b. 若无可用次卡，检查用户 users 表中的 balance 是否足够支付课程 price。若足够，则扣减相应余额。记录支付方式为 BALANCE。 c. 若均不满足，回滚事务并返回“余额/课时不足”错误。

更新数据 (核心): a. 在 bookings 表中插入一条新的预约记录，状态为 BOOKED。 b. 执行带乐观锁的更新: sql UPDATE schedules SET booked\_count = booked\_count + 1, version = version + 1 WHERE id = ? AND booked\_count < capacity AND version = ?; 将 scheduleId 和第3步中查到的 version 作为参数传入。 c. 检查更新结果: 检查上述UPDATE语句影响的行数。如果影响行数为0，意味着在你查询和更新的间隙，数据已被其他线程修改（满员或version变化）。此时应回滚事务并返回“预约失败，请重试”（或“课程已满员”）的错误给用户。

提交事务: 如果更新影响行数为1，说明预约成功，提交数据库事务。

返回成功: 向前端返回成功的响应。

## 5. 测试与验收标准

### 5.1 测试分类

单元测试: 针对后端关键业务逻辑（如预约事务、计费算法、乐观锁实现）编写测试用例。

集成测试: 测试小程序端与后端API接口的联调，确保数据请求和响应格式正确。

用户验收测试 (UAT): 由非开发人员（或模拟用户）根据以下场景进行测试。

### 5.2 UAT验收场景 (Checklist)

| 测试用例ID | 模块 | 测试场景 | 预期结果 | 测试结果 (Pass/Fail) | | YUKE-UAT-01 | 用户 | 新用户首次打开小程序 | 弹出微信授权 -> 授权后进入首页 -> “我的”页面能看到自己的微信头像和昵称。 | | | YUKE-UAT-02 | 预约 | 在课表页预约一节未满员的课程 | 弹出确认框 -> 确认后提示“预约成功” -> 该课程按钮变为灰色“已预约”。 | | | YUKE-UAT-03 | 预约 | 预约一节已满员的课程 | “预约”按钮为灰色不可点击，或点击后提示“课程已满员”。 | | | YUKE-UAT-04 | 我的 | 在“我的预约”中取消一节未开始的课程 | 提示“取消成功” -> “我的预约”列表中该课程消失 -> “预约”页面该课程恢复可预约状态 -> 课时/余额返还。 | | | YUKE-UAT-05 | 训练 | 完成一节课后第二天查看训练日历 | “训练”页面的日历上，昨天对应的日期被高亮标记。 | | | YUKE-UAT-06 | 支付 | 在“我的”页面点击“充值” | 成功调起微信支付 -> 输入密码后完成支付 -> “我的”页面余额正确增加。 | | | YUKE-UAT-07 | 并发 | (需工具模拟)两人同时预约最后一个名额 | 只有一人预约成功，另一人收到“课程已满员”或“预约失败”的提示。数据库中booked\_count不会超过capacity。 | |