

专业群（网络）建设任务



# 广东省高水平中职学校建设项目

(1. 人才培养模式创新)

1.3 开发移动互联网应用项目 1-2 项  
佐证材料

# 目 录

<b>第一部分 四年建设概况及成效 .....</b>	<b>1</b>
<b>第二部分 分年度建设完成情况 .....</b>	<b>4</b>
<b>一、2021 年 .....</b>	<b>4</b>
1. 专家指导项目开发模式 .....	4
<b>二、2022 年度 .....</b>	<b>5</b>
1. 总结 5G 或大数据移动开发模式 .....	5
虚拟终端初始源码 1 份 .....	6
<b>三、2023 年度 .....</b>	<b>6</b>
1. 项目代码资料 .....	6
2. 开发移动互联网项目过程记录文档 .....	7
根据虚拟终端实训管理平台开发方案，规范制作过程记录文档 1 份。 .....	7
<b>四、2024 年度 .....</b>	<b>7</b>
1. 项目代码资料 .....	7
2. 移动互联网项目过程记录文档 .....	8

# 第一部分

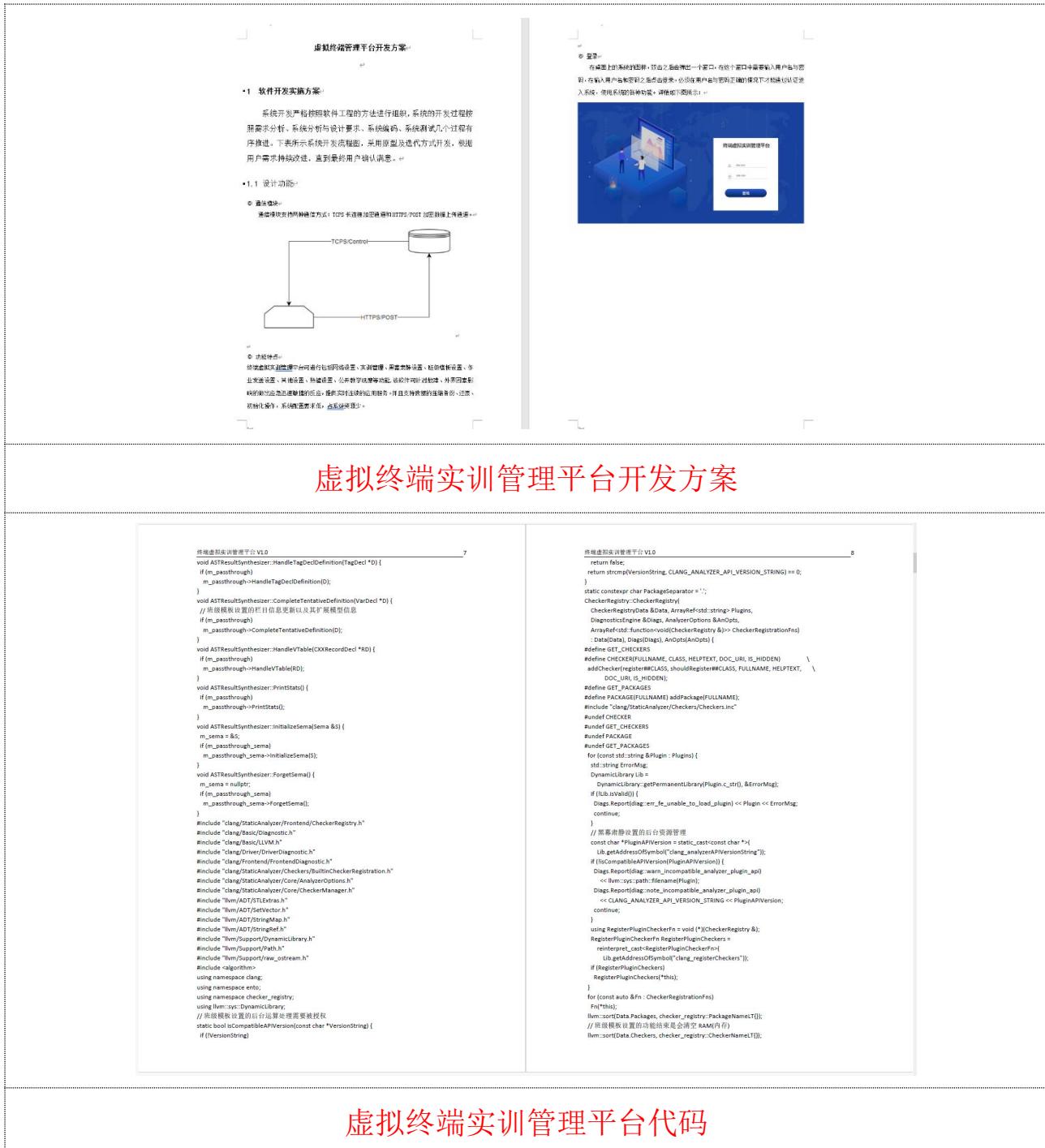
## 四年建设概况及成效

“开发移动互联网应用项目”方面共有 8 个任务要点，截止 2024 年年底，共完成 8 个建设任务点，任务完成率 100%。

为落实“多元协同、岗课证融通”人才培养模式成效，提升计算机网络技术专业群老师软件开发能力，专业群在调研产业与教学需求的基础上，针对专业实训环境不足等问题，计算机部教师设计并实现终端虚拟化实训管理平台，为学生实训提供包括网络设置、实训管理、黑幕肃静设置、班级模板设置、作业发送设置、其他设置、热键设置、公开教学观摩等功能。该软件获得软件著作权登记。

## 一、完成虚拟终端实训管理平台软件 1 个

为提升专业群软件开发水平，增强教师与企业的对接力，专业群针对当前实训条件有限的基础上，开发了虚拟终端实训管理平台 1 个，并完成软件著作权登记。





## 虚拟终端实训管理平台软件著作权证书

# 第二部分

## 分年度建设完成情况

一、2021年

### 1. 专家指导项目开发模式

为落实“多元协同、岗课证融通”人才培养模式成效，提升计算机网络技术专业群老师软件开发水平，专业群在专家指导下进行虚拟终端实训管理平台的开发设计。

XXX 系统——项目开发计划																	
<b>2 项目概述</b>																	
<b>2.1 工作内容</b>																	
移微项目功能： 为普通用户提供浏览、下载功能。 会员用户拥有普通用户权限，并可以上传文件。 系统根据用户搜索频率推荐各种文件 TOP10。 文件格式支持音频、视频、图片。																	
<b>2.2 主要参加人员</b>																	
移微项目由项目团队开发，本项目组有四人组成。 彭国贵，项目组组长，具有软件工程基础理论知识，具有项目开发经验，熟悉关系型数据库系统设计、熟悉 Java、C++ 等语言，并熟悉关系数据库 SQL 编写。 杨艳，项目成员，具有良好编程风格，并熟悉 Java、C++ 等语言，会使用 MySQL 实现数据分析师应用，各类文档的撰写。 蔡伟，项目成员，具有多年的编程经验，擅长使用流行的 Java 编程工具，熟悉 Java 编程语言。 <table border="1"><thead><tr><th>姓名</th><th>角色</th><th>工作描述</th></tr></thead><tbody><tr><td>彭国贵</td><td>组长</td><td>分配工作，编写代码</td></tr><tr><td>林汉裕</td><td>组员</td><td>编码实现，文档撰写</td></tr><tr><td>杨艳</td><td>组员</td><td>编码实现，文档撰写</td></tr><tr><td>蔡伟</td><td>组员</td><td>编码实现，文档撰写</td></tr></tbody></table>			姓名	角色	工作描述	彭国贵	组长	分配工作，编写代码	林汉裕	组员	编码实现，文档撰写	杨艳	组员	编码实现，文档撰写	蔡伟	组员	编码实现，文档撰写
姓名	角色	工作描述															
彭国贵	组长	分配工作，编写代码															
林汉裕	组员	编码实现，文档撰写															
杨艳	组员	编码实现，文档撰写															
蔡伟	组员	编码实现，文档撰写															
<b>2.3 产品</b>																	
<b>2.3.1 程序</b>																	
软件名称：移微项目 语言：Java																	
<b>2.3.2 文件</b>																	
功能：移微项目采用 Java 语言编写，对视频、音频、图片进行分类，浏览，下载，对像素数据进行处理，得到的数据存储数据库中，通过数据分析给出 topN 建议。																	
<b>2.3.3 服务</b>																	
列出给用户提供的各项服务，如培训安装、维护和运行支持等，应逐项规定开始日期、所提供支持的服务和服务的期限。																	
<b>2.3.4 非移交的产品</b>																	
软件需求说明书(软件规格说明书)：对所开发软件的功能、性能、用户界面及运行环境等做出详细的说明，它是在用户与开发人员双方对软件需求取得共同理解并达成协议的条件下编写的，也是实施开发工作的基础。该说明书能给出数据逻辑和数据采集的各项要求，为生成和维护系统数据文件做准备。 概要设计说明书：该说明书概要实际阶段的工作成果，它应说明功能分配、模块划分、程序的总体结构、输入输出以及接口设计、运行设计、数据结构设计和出错处理设计等，为详细设计提供基础。																	
<b>2.4 验收标准</b>																	
<b>2.5 完成项目的最长期限</b>																	
2022.12.18																	

项目开发计划 1 份

专家指导过程资料	
<b>一、网络安全专家指导</b> 2021 年 3 月 15 日中科院云华沈八八及相关工程师、中财利和杨艳到我校指导项目开发。	
<b>二、智能家居专家指导</b> 2021 年 3 月 18 日上海企泰信息技术有限公司何销明经理到学校指导项目开发。	
<b>三、参加实训沙盘培训</b> 2021 年 3 月 7 日至 2021 年 3 月 11 日计算机网络搭建与应用项目组师生 5 人参加了在广州神州数码公司举办的实训沙盘教学培训，为项目开发做技术准备。	

专家指导过程资料 1 份

## 二、2022 年度

### 1. 总结 5G 或大数据移动开发模式

根据前期调研及专家指导，完成虚拟终端实训管理平台开发设计，制定初始代码 1 份。

**虚拟终端管理平台开发方案**

**● 1 软件开发实施方案**

系统开发严格按照软件工程的方法进行组织，系统的开发过程按照需求分析、系统分析与设计要求、系统编码、系统测试几个过程有序推进。下表所示系统开发流程图，采用原型及迭代方式开发，根据用户需求持续改进，直到最终用户确认满意。

**● 1.1 设计功能**

**○ 登录模块**  
通信模块使用 TCP 和 HTTP 协议，通过 HTTPS POST 方式将数据上传到后台。

**○ 功能模块**  
功能模块包括：虚拟终端管理平台（包含设备管理、实训管理、实训直播、设备故障设置、报警设置、参数设置、环境设置、公屏教学设置等）；后台管理模块（包含权限管理、日志管理、故障告警、公屏管理、任务管理、统计分析、系统设置、系统配置等）。

**● 2 登录界面**

在登录界面上有统一的风格的图标，欢迎之后会增加一个窗口，在这个窗口中需要输入用户名与密码，在输入用户名和密码之后点击登录按钮，必须在用户名与密码正确的情况下才能进入实训管理系统的各种功能。操作如下图所示：

**虚拟终端应用开发方案**

**一、项目分析**

核心公司：  
1. 我们需要什么？  
我们在疫情情况下需要线上线下结合的教学支持。  
2. 我们能做什么？  
继续上实训模拟平台  
3. 我们要做成什么效果？  
能给 Windows 的课程做到线上实训。

姓名	角色	工作描述
刘国青	组长	负责总体、编写代码
林设裕	组员	编写类库、文档整理
曾伟	组员	编写类库、文档整理
苏伟	组员	编写类库、文档整理

**虚拟终端项目建设过程资料**

## 虚拟终端初始源码 1 份

三、2023年度

## 1. 项目代码资料

根据虚拟终端实训管理平台的开发，总结代码资料1份。

```

//将结果和训练逻辑写入V10
void ASTResultsSynthesizer::HandleTagDeclDefinition(TagDecl *D) {
if (m_passthrough)
    m_passthrough->HandleTagDeclDefinition(D);
}

void ASTResultsSynthesizer::CompleteTentativeDefinition(VarDecl *D) {
//从模块板设置的栏目信息更新以及其扩展模型信息
if (m_passthrough)
    m_passthrough->CompleteTentativeDefinition(D);
}

void ASTResultsSynthesizer::HandleTable(CXXRecordDecl *RD) {
if (m_passthrough)
    m_passthrough->HandleTable(RD);
}

void ASTResultsSynthesizer::PrintStats() {
if (m_passthrough)
    m_passthrough->PrintStats();
}

void ASTResultsSynthesizer::InitializeSema(Sema &S) {
m_sema = &S;
if (m_passthrough_sema)
    m_passthrough_sema->InitializeSema(S);
}

void ASTResultsSynthesizer::ForgetSema() {
m_sema = nullptr;
if (m_passthrough_sema)
    m_passthrough_sema->ForgetSema();
}

#include "clang/StaticAnalyzer/Frontend/CheckerRegistry.h"
#include "clang/Basic/Diagnostic.h"
#include "clang/Basic/LVM.h"
#include "clang/Driver/DriverDiagnostic.h"
#include "clang/Frontend/FrontendDiagnostic.h"
#include "clang/StaticAnalyzer/Checkers/BuiltinCheckerRegistration.h"
#include "clang/StaticAnalyzer/Core/AnalyzerOptions.h"
#include "clang/StaticAnalyzer/Core/CheckerManager.h"
#include "llvm/ADT/StringExtras.h"
#include "llvm/ADT/SetVector.h"
#include "llvm/ADT/StringMap.h"
#include "llvm/ADT/StringRef.h"
#include "llvm/Support/DynamicLibrary.h"
#include "llvm/Support/Path.h"
#include "llvm/Support/raw_ostream.h"
#include <algorithm>
using namespace clang;
using namespace ento;
using namespace checker_registry;
using llvm::sys::DynamicLibrary;
//从模块板设置的后台运算处理需要被授权
static bool isCompatibleAPIVersion(const char *versionString) {
if (IVersionString)

```

```
将端虚机训练管道平台 V1.0

return false;
return strcmp(versionString, CLANG_ANALYZER_API_VERSION_STRING) == 0;

static constexpr char packageSeparator = '\n';
CheckerRegistry::CheckerRegistry()
    CheckerRegistryData &data, ArrayRef<std::string> Plugins,
    DiagnosticEngine &Diags, AnalyzerOptions &AnOpts,
    ArrayRef<std::function<void(CheckerRegistry &>)> CheckerRegistrationFns)
    : Data(data), Diags(Diags), AnOpts(AnOpts) {
#define DEFINE_GET_CHECKERS
#define CHECKER(FULLNAME, CLASS, HELPTEXT, DOC_URI, IS_HIDDEN) \
    addChecker(register<#CLASS>, shouldRegister##CLASS, FULLNAME, HELPTEXT, \
              DOC_URI, IS_HIDDEN);
#define DEFINE_GET_PACKAGES
#define PACKAGE(FULLNAME) addPackage(FULLNAME);
#include "clang/StaticAnalyzer/Checkers/Checkers.inc"
#undef GET_CHECKER
#undef GET_CHECKERS
#undef PACKAGE
#undef GET_PACKAGES
for (const std::string &Plugin : Plugins) {
    std::string errorMsg;
    DynamicLibrary lib;
    if (!lib.isValid()) {
        Diags.Report(diag::err_fe_unable_to_load_plugin) << Plugin << errorMsg;
        continue;
    }
// 黑幕肃静设置的后台资源管理
    const char *pluginAPIVersion = static_cast<const char *>(
        lib.getAddressOfSymbol("clang_analyzerAPIVersionString"));
    if (!isCompatibleAPIVersion(pluginAPIVersion)) {
        Diags.Report(diag::warn_incompatible_analyzer_plugin_api)
            << ILM::sys::path::filename(Plugin);
        Diags.Report(diag::note_incompatible_analyzer_plugin_api)
            << CLANG_ANALYZER_API_VERSION_STRING << pluginAPIVersion;
        continue;
    }
    using RegisterPluginCheckerFn = void (*[&]CheckerRegistry &);
    RegisterPluginCheckerFn registerPluginCheckers =
        reinterpret_cast<RegisterPluginCheckerFn>(
            lib.getAddressOfSymbol("clang_registerCheckers"));
    if (registerPluginCheckers)
        registerPluginCheckers(*this);
}
for (const auto &fn : CheckerRegistrationFns)
    fn(*this);
ILM::sort(Data.Packages, checker_registry::PackageNameLT());
// 以便板报设置成功结果是会清空 RAM 内存
ILM::sort(Data.Checkers, checker_registry::CheckerNameLT());
```

# 虚拟终端代码资料

## 2. 开发移动互联网项目过程记录文档

根据虚拟终端实训管理平台开发方案，规范制作过程记录文档 1 份。

**移动互联网项目过程记录文档**

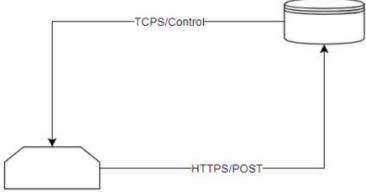
### 一、项目概述

终端虚拟实训管理平台是针对专业实训需求设计的，为提升实训的管理条件，满足虚拟化实训的管理，为学生提供终端虚拟实训功能。终端虚拟实训管理平台可进行包括网络设置、实训管理、黑幕肃静设置、班级模板设置、作业发送设置、其他设置、热键设置、公开教学观摩等功能，该软件可针对故障、外界因素影响的做出应急迅速敏捷的反应，提供实时连续的应用服务。并且支持数据的压缩备份、还原、初始化操作，系统配置要求低，占系统资源少。

### 二、项目功能设计

#### (一) 通信模块

通信模块支持两种通信方式：TCP长连接加密通道和HTTPS/POST加密数据上传通道。



**(二) 登录设计**

在桌面上的系统的图标，双击之后会弹出一个窗口，在这个窗口中需要输入用户名与密码，在输入用户名和密码之后点击登录。必须

在用户名与密码正确的情况下才能通过认证进入系统，使用系统的各种功能。



**(三) 主页设计**

操作人员点击软件的图标即可进入该软件的主界面，操作人员可以看见该软件主要包括网络设置、实训管理、黑幕肃静设置、班级模板设置、作业发送设置、其他设置、热键设置、公开教学观摩等功能。操作人员点击相应功能图标即可进入该功能的操作界面。



**(四) 网络设置设计**

在首页界面中有一个网上配置选项，点击网上配置选项，进入到

**虚拟终端项目过程记录文档**

## 四、2024 年度

### 1. 项目代码资料

根据方案，完善虚拟终端实训管理平台代码。

```
掩码和运算逻辑处理类 V1.0

void ASTResultsSynthesizer::HandleTagDef(Definition* TagDef *D) {
    if (m_pastrthrough)
        m_pastrthrough->HandleTagDef(Definition* D);
}

void ASTResultsSynthesizer::CompleteTentativeDefinition(VarDecl *D) {
    // 班级模板设置的栏目信息更新以及其扩展模型信息
    if (m_pastrthrough)
        m_pastrthrough->CompleteTentativeDefinition(D);
}

void ASTResultsSynthesizer::HandleVTable(CXXRecordDecl *RD) {
    if (m_pastrthrough)
        m_pastrthrough->HandleVTable(RD);
}

void ASTResultsSynthesizer::PrintStats() {
    if (m_pastrthrough)
        m_pastrthrough->PrintStats();
}

void ASTResultsSynthesizer::InitializeSema(Sema &S) {
    m_sema = &S;
    if (m_pastrthrough_sema)
        m_pastrthrough_sema->InitializeSema(S);
}

void ASTResultsSynthesizer::ForgetSema() {
    m_sema = nullptr;
    if (m_pastrthrough_sema)
        m_pastrthrough_sema->ForgetSema();
}

#include "clang/StaticAnalyzer/Frontend/CheckerRegistry.h"
#include "clang/Basic/Diagnostic.h"
#include "clang/Basic/LVM.h"
#include "clang/Driver/DriverDiagnostic.h"
#include "clang/Frontend/FrontendDiagnostic.h"
#include "clang/StaticAnalyzer/Checkers/BuiltinCheckerRegistration.h"
#include "clang/StaticAnalyzer/Core/AnalyzerOptions.h"
#include "clang/StaticAnalyzer/Core/CheckerManager.h"
#include "lvm/AD/TLTreeExtraction.h"
#include "lvm/AD/TLVectorFor.h"
#include "lvm/AD/TMMap.h"
#include "lvm/Support/Path.h"
#include "lvm/Support/StringLibrary.h"
#include "lvm/Support/Path.h"
#include "lvm/Support/raw_ostream.h"
#include "salg/algorithm";
using namespace clang;
using namespace entt;
using namespace checker_registry;
using lvm::sys::DynamicLibrary;
// 班级模板设置的后台运透处理需要被授权
static bool isCompatibleAPIVersion(const char *VersionString) {
    if (IVersionString)
```

```
    return false;
    return strcmp(VersionString, CLANG_ANALYZER_API_VERSION_STRING) == 0;
}

static constexpr char PackageSeparator = ':';
CheckerRegistry* CheckerRegistry::Create();
void CheckerRegistry::RegisterData(Data ArrayRef<std::string> Plugins,
DiagnosticEngine::Bundles AnalyzerOptions &AnOpts,
ArrayRef<std::function<void(CheckerRegistry &)>> CheckerRegistrationFns)
{
    (Data::Data), Diags(diags), AnOpts(AnOpts);
    #define GET_CHECKERS \
    #define GET_PACKAGE \
    #define PACKAGE(FULLNAME) addPackage(FULLNAME); \
    #include <clang/StaticAnalyzer/Checkers/Checkers.inc>
    #undef CHECKER
    #undef GET_CHECKERS
    #undef PACKAGE
    #undef GET_PACKAGES
    for (const std::string &Plugin : Plugins) {
        std::string ErrorMsg;
        DynamicLibrary Lib = \
            DynamicLibrary::getPermanentLibrary(Plugin.c_str(), &ErrorMsg);
        if (Lib.isInvalid()) {
            Diags.Report(Diagnostics::err_fe_unable_to_load_plugin) <> Plugin <> ErrorMsg;
            continue;
        }
    // 兼容静默设置的后台资源管理
    const char *PluginAPIVersion = static_cast<const char*>{ \
        Lib.getAddressOfSymbol("clang_analyzerAPIVersionString")}; \
    if (!IsCompatibleAPIVersion(PluginAPIVersion)) { \
        Diags.Report(Diagnostics::warn_incompatible_analyzer_plugin_api) \
        << IVM::sys::path::filename(Plugin); \
        Diags.Report(Diagnostics::note_incompatible_analyzer_plugin_api) \
        << CLANG_ANALYZER_API_VERSION_STRING << PluginAPIVersion; \
        continue; \
    } \
    using RegisterPluginCheckerFn = void(*)(&CheckerRegistry &); \
    RegisterPluginCheckerFn RegisterPluginChecker; \
    reinterpret_cast<RegisterPluginCheckerFn*> \
        Lib.getAddressOfSymbol("clang_registerChecker"); \
    if (RegisterPluginCheckers) \
        RegisterPluginCheckers(*this); \
    }
    for (const auto &Fn : CheckerRegistrationFns)
        Fn(*this);
    IVM::sort(Data Packages, checker_registry::PackageNameIT());
    // 班级模板被置的位置的结果将是会清空 RAM(内存)
    IVM::sort(Data Checkers, checker_registry::CheckerNameIT());
}
```

虚拟终端实训管理平台项目代码

## 2. 移动互联网项目过程记录文档

根据方案要求，完成虚拟终端实训管理平台项目过程记录文档编写。

# 移动互联网项目过程记录文档

## 一、项目基础信息

项目名称：虚拟终端实训管理平台

项目概述：终端虚拟实训管理平台可进行包括网络设置、实训管理、黑屏静音设置、班级模板设置、作业发送设置、其他设置、热键设置、公开教学观摩等功能。该软件可针对故障、外界因素影响的做出应急迅速敏捷的反应，提供实时连续的应用服务。

## 二、项目计划

开发人员：戴金辉

项目周期：2021年3月-2023年11月

## 三、项目开发阶段

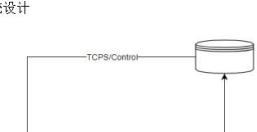
### (一) 需求分析

1. 开发目的：本平台面向教育行业，特别是职业院校和培训机构，提供实训室的网络配置、实训管理、教学观摩等功能。
2. 功能需求：
  - 通信模块：支持 TCP 和 HTTPS/POST 两种通信方式。
  - 网络设置：自定义网络参数，如数据发送速率、教学执行速率、断线检测周期。
  - 实训管理：提供实训课程的创建、管理和监控功能。
  - 黑屏静音设置：允许教师控制实训室内的显示和声音设置。
  - 班级模板设置：支持不同大小和质量的班级模板配置。
  - 作业发送设置：管理作业的发送和接收。
  - 其他设置：包括学生事件音效、系统密码设置等。
  - 热键设置：自定义快捷键，提高教学效率。
  - 公开教学观摩：支持教师进行公开教学和观摩。

### 3. 研讨照片



### (二) 系统设计



```
graph TD; Client[Client] -- "TCP/Control" --> Server[Server]; Client -- "HTTPS/POST" --> Server;
```

#### 1. 组件设计

##### 1.1 通信模块

- 实现 TCP 和 HTTPS/POST 通信协议。
- 提供数据加密和解密功能。

## 虚拟终端实训管理平台项目代码



## 虚拟终端实训管理平台软件著作权证书