



東南大學



中國高科集團

CHINA HI-TECH GROUP

《智能防偽考勤系統》 概要設計說明

V1.1

版本/ 状态	作者	日期	备注
1.0	陈嘉博、安典坤、蔡林希、董睿、涂煜洋	2020-7-12	创建
1.1	陈嘉博、安典坤、蔡林希、董睿、涂煜洋	2020-08-01	修订

目 录

第一部分 引言.....	6
一 编写目的.....	6
二 读者对象.....	6
三 术语解释.....	6
XML	6
Web Service.....	6
MVC.....	7
MAC 地址.....	7
第二部分 项目概述.....	8
一 项目描述.....	8
二 项目功能描述.....	8
1、签到和身份验证.....	8
2、员工工作日修改.....	9
3、查看修改个人考勤情况.....	9
4、管理者查看修改日（月）签到表.....	9
5、签到申诉.....	10
6、添加删除员工信息.....	10
7、查看工作日历.....	11
8、出差请假申请.....	11
9、请假审核.....	11
10、考勤时间修改.....	11
11、导入国家法定工作日、节假日	12
12、管理员注册登录.....	12
13、合法考勤机器管理.....	12
第三部分 设计约束.....	13
一 需求约束.....	13
遵循的技术标准.....	13
软硬件环境标准.....	13
接口/协议标准	13
用户界面标准.....	13
软件质量.....	13
二 隐含约束.....	14
第四部分 功能单元和功能模块设计.....	14
一 系统相关业务流程.....	14
数据库访问模块.....	14
1.1 类图结构.....	14
1.2 类说明.....	14
1.3 功能时序图.....	15

日历访问并显示功能.....	15
修改日历相关信息功能.....	16
员工信息访问功能.....	17
员工考勤数据修改功能.....	17
签到模块.....	18
2.1 类图.....	18
2.2 类说明.....	19
2.3 功能时序图.....	19
2.4 时序流程说明.....	20
三、业务功能概要结构.....	20
1、员工端功能模块.....	22
1.1 注册登录.....	22
1.2 签到.....	22
1.3 查看考勤信息.....	22
1.4 出差请假申请.....	22
1.5 签到申诉.....	23
2、管理员端功能模块.....	23
2.1 管理者注册（管理员权限操作）	23
2.2 管理者登录.....	24
2.3 添加员工（管理员权限操作）	24
2.4 删除员工（管理员权限操作）	24
2.5 查看整体签到情况（管理员权限操作）	24
2.6 员工申诉审核（管理员权限操作）	25
2.7 出差请假审核（管理员权限操作）	25
2.8 设置工作日（管理员权限操作）	25
2.9 出差请假审核（管理员权限操作）	26
2.10 考勤时间修改（管理员权限操作）	26
2.11 导入国家工作日（管理员权限操作）	26
2.12 信息导出.....	27
2.13 合法考勤机器添加（管理员权限操作）	27
2.14 合法考勤机器删除（管理员权限操作）	27
第六部分 总体设计.....	28
一、逻辑架构设计.....	28
软件系统采用 MVC 设计模式	28
业务处理与数据库间采用分层架构思想.....	28
各客户端同服务器间采用 C/S 风格	29
二、物理架构设计.....	29
服务端.....	29
具体应用.....	29
客户端.....	29
管理员端.....	29
三、技术架构设计.....	29
MVC 模型设计	29
智能身份鉴定技术.....	31

第七部分 界面设计.....31

 一、风格设计.....31

 1、主桌面风格.....31

 2、业务界面风格展示.....32

第八部分 运行环境和部署.....33

 一、运行环境.....33

 1、服务器.....33

 2、客户端环境.....33

 二、系统性能要求.....34

第一部分 引言

一 编写目的

本篇概要设计的主要目的是把需求分析得到的用例模型转换为软件结构和数据结构。设计软件结构的具体任务是：将一个复杂系统按功能进行模块划分、建立模块的层次结构及调用关系、确定模块间的接口及人机界面等。数据结构设计包括数据特征的描述、确定数据的结构特性、以及数据库的设计。

本设计是指导详细设计和项目实施的重要指导性文件，也是进行系统集成测试和重要依据。

二 读者对象

该文档的读者为该系统的软件分析人员、开发管理人员和测试人员。

三 术语解释

XML

可扩展标记语言（Extensible Markup Language, XML），用于标记电子文件使其具有结构性的标记语言，可以用来标记数据、定义数据类型，是一种允许用户对自己的标记语言进行定义的源语言。XML 是标准通用标记语言（SGML）的子集，非常适合 Web 传输。XML 提供统一的方法来描述和交换独立于应用程序或供应商的结构化数据。

Web Service

Web Service 是基于网络的、分布式的模块化组件，它执行特定的任务，遵守具体的技术规范，这些规范使得 Web Service 能与其他兼容的组件进行互操作。Internet Inter-Orb Protocol (IIOP) 都已经发布了很长时间了，但是这些模型都依赖于特殊对象模型协议，而 Web Services 利用 SOAP 和 XML 对这些模型在通讯方面作了进一步的扩展以消除特殊对象模型的障碍。Web Services 主要利用 HTTP 和 SOAP 协议使商业数

据在 Web 上传输，SOAP 通过 HTTP 调用商业对象执行远程功能调用，Web 用户能够使用 SOAP 和 HTTP 通过 Web 调用的方法来调用远程对象。

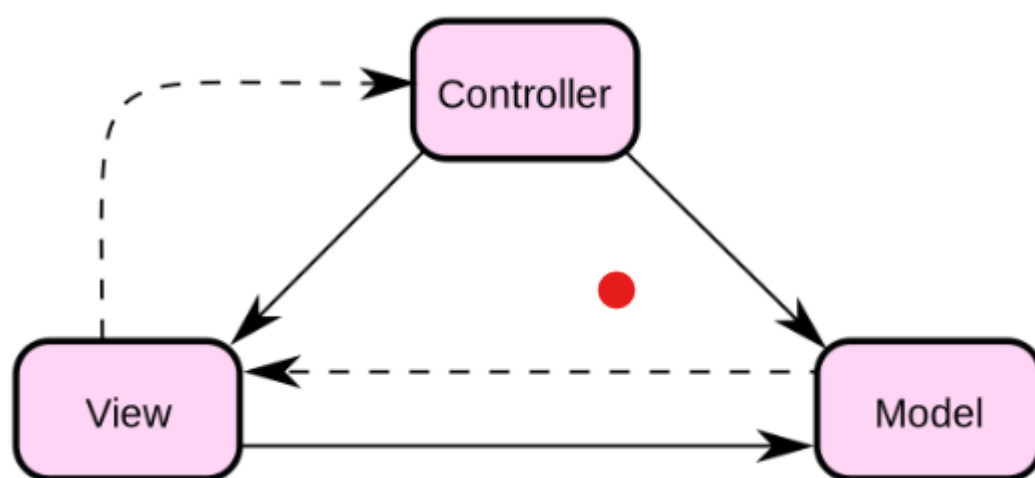
MVC

MVC 模式代表 Model-View-Controller（模型-视图-控制器）模式。这种模式用于应用程序的分层开发。

Model（模型） - 模型代表一个存取数据的对象或 JAVA POJO。它也可以带有逻辑，在数据变化时更新控制器。

View（视图） - 视图代表模型包含的数据的可视化。

Controller（控制器） - 控制器作用于模型和视图上。它控制数据流向模型对象，并在数据变化时更新视图。它使视图与模型分离开。



MAC 地址

MAC 地址（英语：Media Access Control Address），直译为媒体存取控制位址，也称为局域网地址（LAN Address），MAC 位址，以太网地址（Ethernet Address）或物理地址（Physical Address），它是一个用来确认网络设备位置的位址。在 OSI 模型中，第三层网络层负责 IP 地址，第二层数据链路层则负责 MAC 位址。MAC 地址用于在网络中唯一标示一个网卡，一台设备若有一或多个网卡，则每个网卡都需要并会有一个唯一的

MAC 地址。

第二部分 项目概述

一 项目描述

随着社会的飞速发展以及各行业对快速有效的身份验证的迫切要求，生物特征识别技术在近十几年中得到了飞速的发展。与此同时，随着企业规模的扩大和企业信息化程度的提高，企业对高效准确的打卡考勤的需求也与日俱增，比如能通过多种技术手段确定打卡人员的身份，防止顶替打卡，同时要注重效率与便捷性等。为了满足企业的考勤需求，提高考勤的准确率和效率，提升企业的效益，我们小组准备开发一个智能防伪考勤系统，这个系统可以部署在公司员工的台式机上，可以完成员工考勤、申请请假，管理员查看考勤信息、员工信息、修改工作日等方面的功能。

二 项目功能描述

用户业务主要包含考勤信息收集管理、工作日程表设置、员工信息收集管理以及其他业务。

考勤信息收集管理包含九个主要业务：签到和身份验证、员工工作日修改、查看修改员工个人考勤信息、员工考勤信息导出、管理者查看日（月）签到表、签到申诉。

工作日程表设置包含五个业务：查看工作日历、出差请假申请、请假审核、考勤时间修改、导入国家工作日。

员工信息收集管理业务包含一个业务：添加删除员工信息。

其他业务包含三个业务：管理员注册登录、合法考勤机器管理。

1、签到和身份验证

主要是验证当前员工的身份并更新考勤信息。

需要验证的信息有：

- 1) 员工工号
- 2) 员工面部特征比对
- 3) 员工声纹特征比对

员工输入工号点击确定后开始比对身份信息，通过后提示签到成功并且更新数据库中的信息。

2、员工工作日修改

主要是根据节假日以及企业安排的需要，管理员动态地调整员工的工作日和休息日的安排。

管理员在工作日历界面上点击某一天，右侧会出现按钮“设为工作日”和“设为休息日”。

3、查看修改个人考勤情况

员工和管理者可以通过工号来查看该员工的相关考勤信息，与员工不同的是，管理员拥有修改某天的考勤情况的权限。

考勤信息主要有：

- 1) 缺勤次数
- 2) 考勤评价
- 3) 请假情况

管理员和员工可以点击查看个人考勤情况，输入工号点击确定之后即可显示相关界面，并可以选择月份来查看当前以及过去某个月的考勤数据。

4、管理者查看修改日（月）签到表

主要是管理员查看具体某一天或者某一月的总的考勤数据。

主要考勤信息有：

1) 该日（月）缺勤人次

2) 该日（月）迟到人次

3) 该日（月）请假人次

管理员点击查看签到表按钮即可显示当日考勤信息，并可以选择查看其他日期的考勤信息。

5、签到申诉

主要是员工在身份验证失败或者觉得自己的考勤信息出现了问题时，通过填写申诉表向管理员说明相应信息。

申诉表需要填写的信息有：

1) 申诉原因

2) 具体原因/情况说明

员工可以在主界面或者身份验证失败后点击签到申诉按钮，出现申诉界面，填写表格后点击确定提交按钮进行提交，等待管理员处理。

6、添加删除员工信息

主要是员工入职或者是离职时，管理员要录入或者删除相应的员工信息。

需要添加或者删除的主要信息有：

1) 员工姓名

2) 员工工号

3) 员工手机号

4) 员工面部信息

5) 员工声音信息

管理员在员工信息管理界面点击添加按钮，填写相应信息并让员工录入面部和声音

信息，该信息经过模型训练以后存入到数据库中；或者按下删除按钮，输入员工工号删除该员工信息。

7、查看工作日历

员工和管理员可以查看当前或者之前某个月份的工作日历，员工可以看到该月自己的部分考勤信息，与员工不同的是，管理员拥有其它的操作权限。

日历上显示的信息有：

- 1) 基本的日历信息（日期、周几、节日等）
- 2) 某一天是工作日还是休息日
- 3) 员工界面上还要有考勤信息

员工和管理员可以点击查看工作日历按钮显示相应界面。

8、出差请假申请

主要是员工在有出差或者请假时进行申请，这样系统会更改其相应的考勤信息。

需要填写的信息有：

- 1) 原因（请假/出差）
- 2) 开始日期和结束日期

员工点击出差请假申请，填写完表格等待管理员处理。

9、请假审核

主要是管理员处理员工的请假申请，同意或者驳回，并填写理由

管理员点击待处理请假请求按钮，显示出当前待审核的请假信息，逐条处理，并进行回复，把结果反馈给员工。

10、考勤时间修改

主要是管理员针对季节时令变化以及公司需求，改变考勤的时间、次数等。

一天中可能要多次考勤，所以管理员要添加删除某次考勤、设置某次考勤的截止时间。

管理员点击考勤时间管理显示相应界面，界面上有添加按钮可以添加一条新的考勤，并设置考勤的截止时间等。每条考勤后还应该删除和修改按钮，完成相应功能。

11、导入国家法定工作日、节假日

主要是系统从互联网中导入国家规定的工作日、节假日信息，帮助管理者安排公司的工作日和休息日，减少手工操作。

管理员点击导入国家工作日按钮，系统自动抓取相关信息并可视化显示。

12、管理员注册登录

主要是新的管理员为了获取系统的管理权限进行新账号注册或者登录。

新的账号由已有的管理员新建，需要填写的信息主要有：

- 1) 新管理员的姓名
- 2) 新管理员的工号
- 3) 新管理员的初始密码

管理员点击新增管理员按钮填写相应信息并保存到数据库中。

13、合法考勤机器管理

主要是管理员管理公司内部的考勤机器。

管理员点击考勤机器管理，通过身份验证后进入管理页面，当前存在的每条考勤机器信息后面有修改和删除按钮，对当前机器进行修改和删除操作；界面上部应该有添加按钮，添加新的考勤机器的物理地址。

第三部分 设计约束

一 需求约束

遵循的技术标准

- (1) 开发过程中遵循 Qt 和 Python 以及 MySQL 的基本语法。
- (2) 命名规则遵循《Objective-C Style Guide》中相关的规定。

软硬件环境标准

本系统采用 MVC 架构。Python + Qt 编写，数据库采用 MySQL。软件的开发环境为 windows 桌面端。

接口/协议标准

通过 socket 协议实现终端和服务端之间的数据通信

用户界面标准

使用 Qt 标准 UI 库对界面进行开发，并且负责最后的界面优化和美化。

软件质量

- (1) 正确性：员工的打卡签到、管理员的处理员工信息、处理考勤表信息等功能需要被正确处理。
- (2) 安全性：员工和管理员不能直接访问数据库，需要通过功能模块的接口完成对于数据库的查看和修改操作。
- (3) 可用性：该软件能保证在长时间内（7*24h）正常使用，并且保证出现了错误后可以迅速恢复。

(4) 可测试性：系统是的各部分易于单独测试，并能方便地进行整体测试。

二 隐含约束

(1) 使用者具有基本的业务技能和基本的电脑知识，对于软件的使用应在经过学习后进行操作。

(2) 软件可以流畅地运行在 Windows 桌面的环境中。

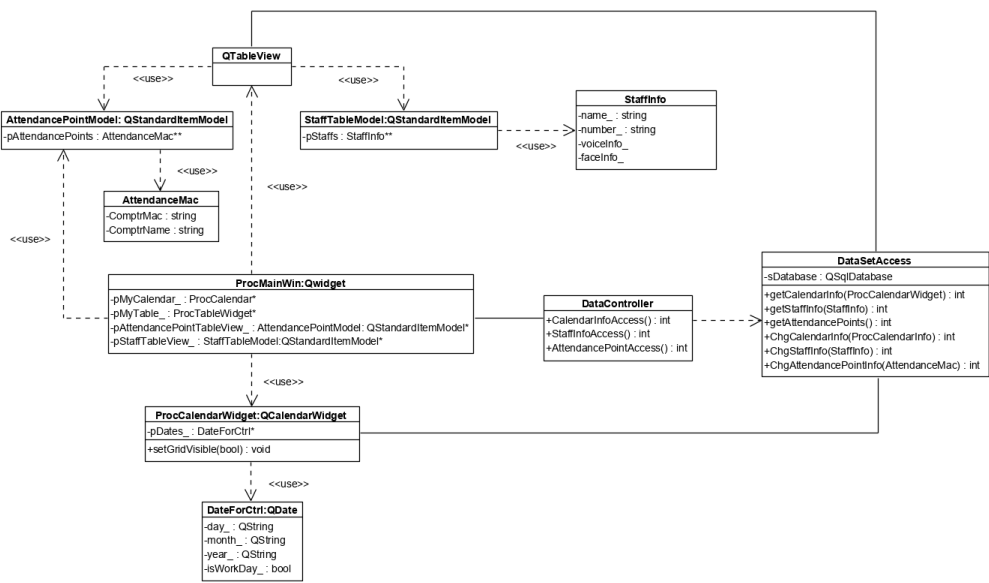
(3) 应该把有可能变动的参数存放到配置文件或数据库中，保证修改参数的灵活性。

第四部分 功能单元和功能模块设计

一 系统相关业务流程

数据库访问模块

1.1 类图结构

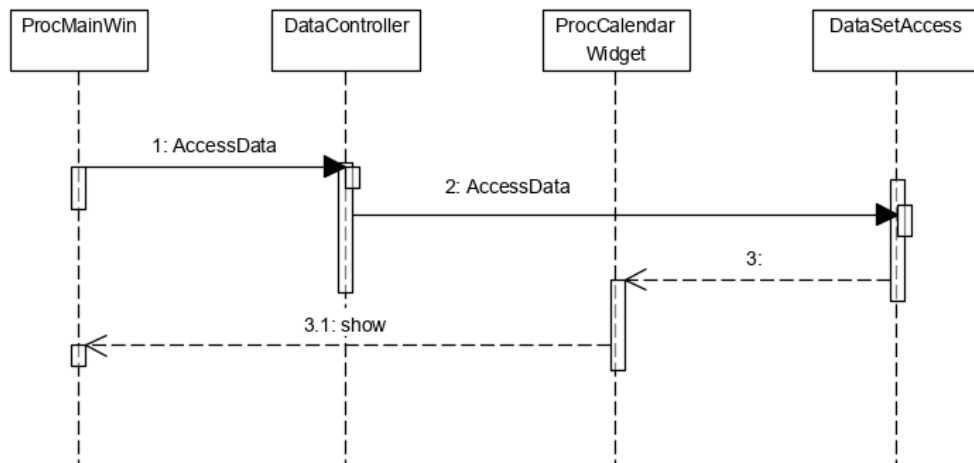


1.2 类说明

1. ProcMainWin 类：负责创建应用环境，作为展现给用户的程序主窗体对象。
2. DataController 类：控制器，用于在软件系统的视图和模型（业务处理模块）间进行数据和信息的传递工作。
3. DataSetAccess 类：作为系统和相应数据库的交互模块，完成系统与数据库间数据的交互工作。
4. ProcCalendarWidget 类：完成日历视图、控制和模型的集合模组，用于完成日历的显示和模型处理功能。
5. AttendanceMac 类：考勤机器类，用于存储考勤机器的物理地址和机器名称（编号）的数据结构。
6. Staffinfo 类：员工信息类，用于存储员工的工号姓名以及他的声音和面部信息等数据的数据结构。
7. StaffTableModel 类：用于帮助使用表格形式组织员工相关信息的数据结构。
8. QTableView 类：Qt 模组，用于以表格形式向用户呈现相应的员工或考勤机器信息，并可以通过表格模型对其中的数据进行组织与处理。
9. AttendancePointModel 类：被 QTableView 类所使用的用于组织考勤机器相关数据的数据结构。
10. Server 类：用于实现直接对于数据库的访问，提供统一的接口。

1.3 功能时序图

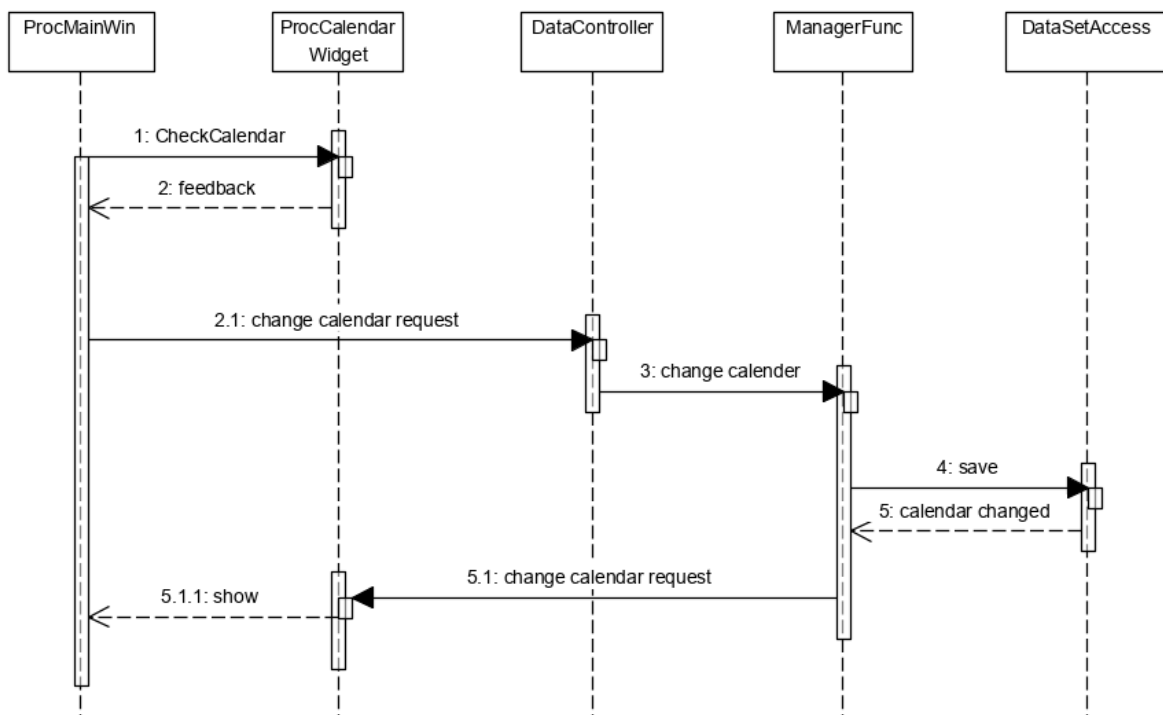
日历访问并显示功能



时序流程说明：

- (1) 管理员请求访问数据，数据控制器将访问请求传递给数据库访问类
- (2) DataSetAccess 访问数据库获得相应数据。
- (3) DataSetAccess 类将相应数据传输给相应的日历显示模组，显示在主界面上

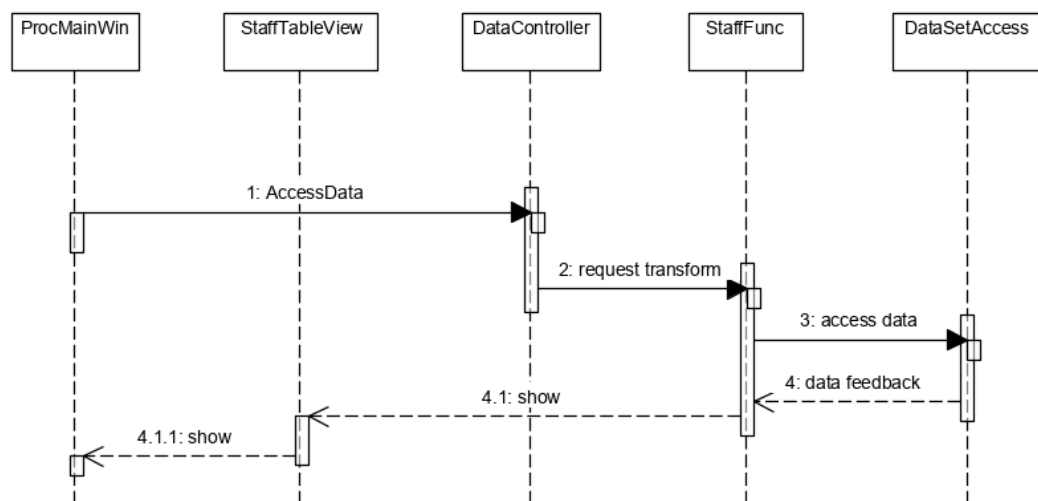
修改日历相关信息功能



时序流程说明：

- (1) 用户在日历界面上请求修改数据
- (2) 数据控制器将修改请求和期望修改数据传递给传递到 ManagerFunc 类开启业务处理
- (3) ManageFunc 执行相应业务并将完成的数据通过 DataSetAccess 类保存到数据库中
- (4) 在获得保存成功反馈之后 ManagerFunc 类将修改后的数据反馈到显示模組上，呈现给用户。

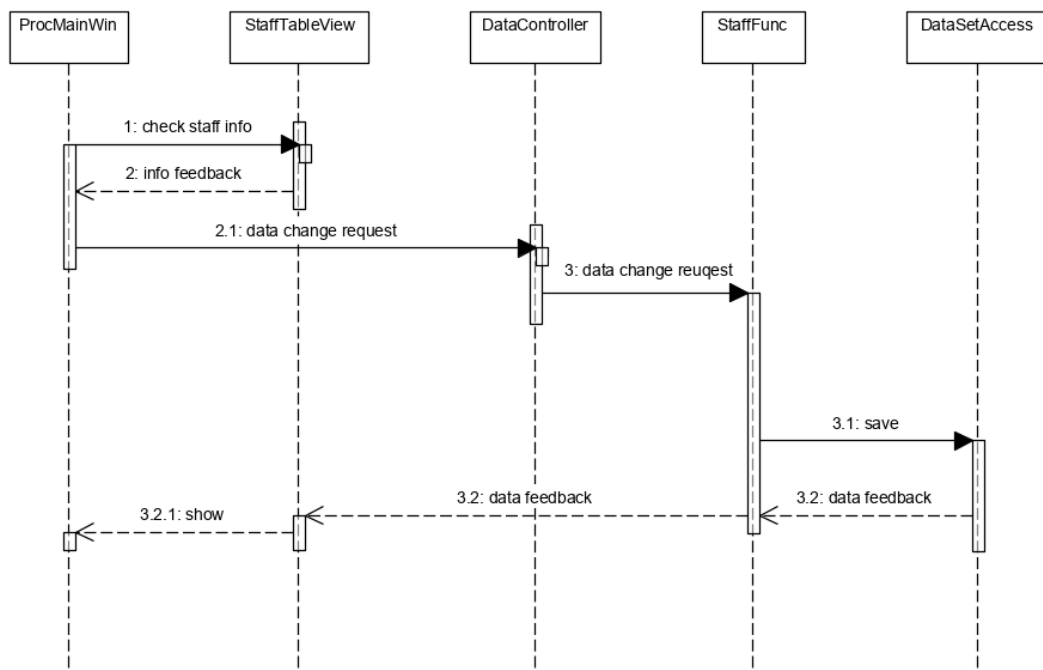
员工信息访问功能



时序流程说明：

- (1) 用户在主界面上请求访问数据。
- (2) 数据控制器将访问数据请求传递给 StaffFunc 类启动访问业务。
- (3) StaffFunc 类通过 DataSetAccess 类从数据库获得相应信息。
- (4) StaffFunc 类将获取的信息反馈到相应的显示模組上呈现给用户。

员工考勤数据修改功能

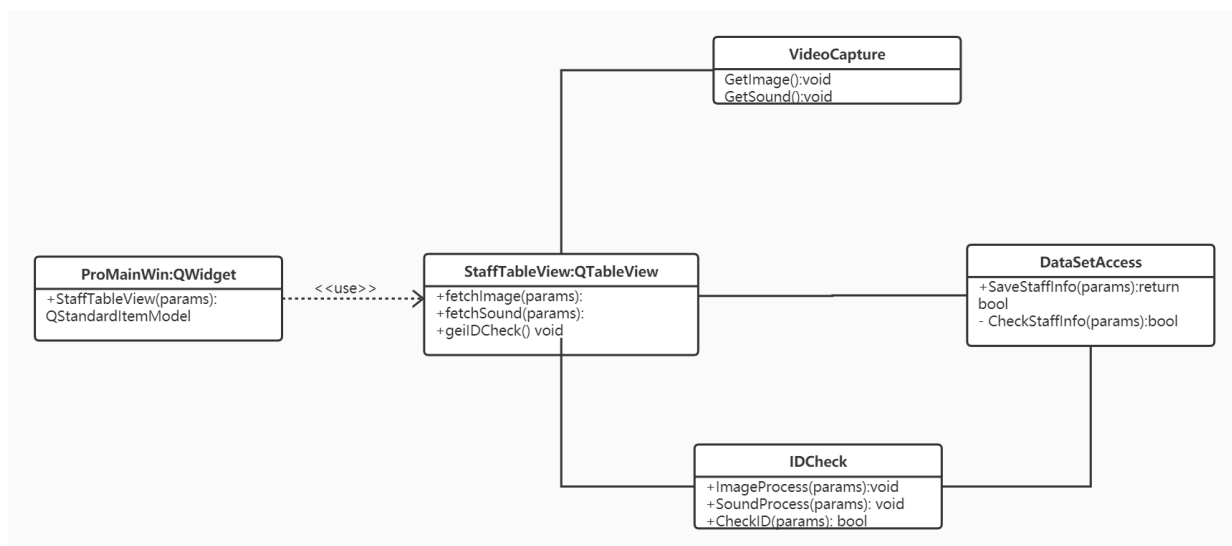


时序流程说明：

- （1）用户在考勤信息界面上请求修改数据。
- （2）数据控制器把需要修改的数据和请求传递到 StaffFunc 类中进行处理。
- （3）StaffFunc 将修改完毕的数据通过 DataSetAccess 类保存到数据库中。
- （4）在获得成功反馈之后将数据反馈到数据显示模型中显示给用户。

签到模块

2.1 类图

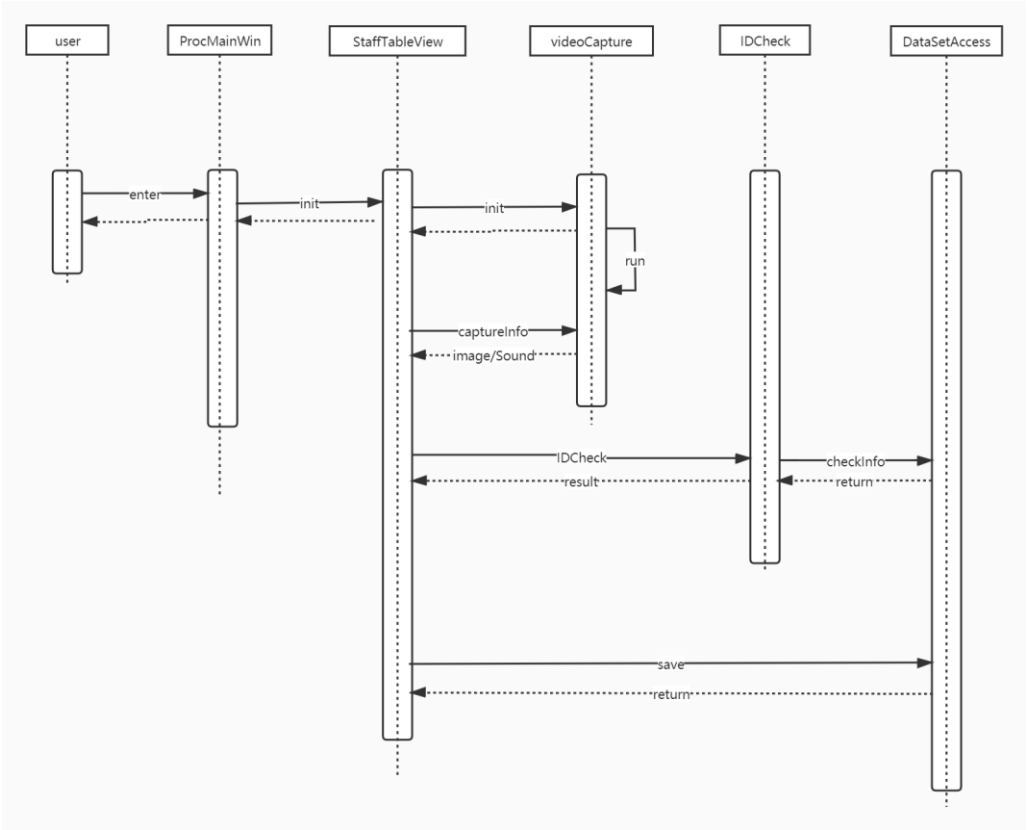


2.2 类说明

ProcMainWin 类：创建应用环境，启动程序主窗体对象。

StaffTableView 类：在 ProcMainWin 中启动，向设备发起请求，从 videoCapture 处获取音频图像信息后返回，发送给 IDCheck 模块识别，将识别结果保存至数据库 DataSetAccess。

2.3 功能时序图



2.4 时序流程说明

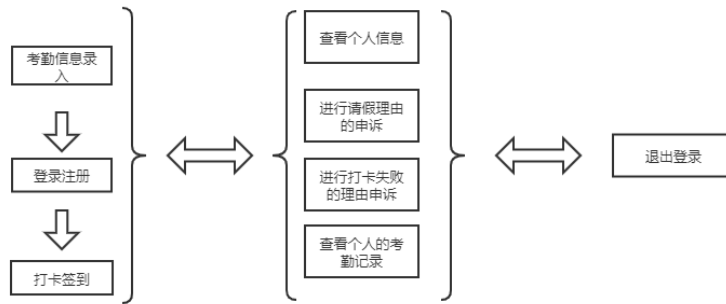
（1）用户登录界面，获取权限后通过摄像头等设备获取图像等信息，返回界面，同时交给身份验证模块进行处理。

（2）身份验证模块对数据处理后与数据库中信息比对，查找成功则返回，同时保存考勤记录。识别失败则直接返回。

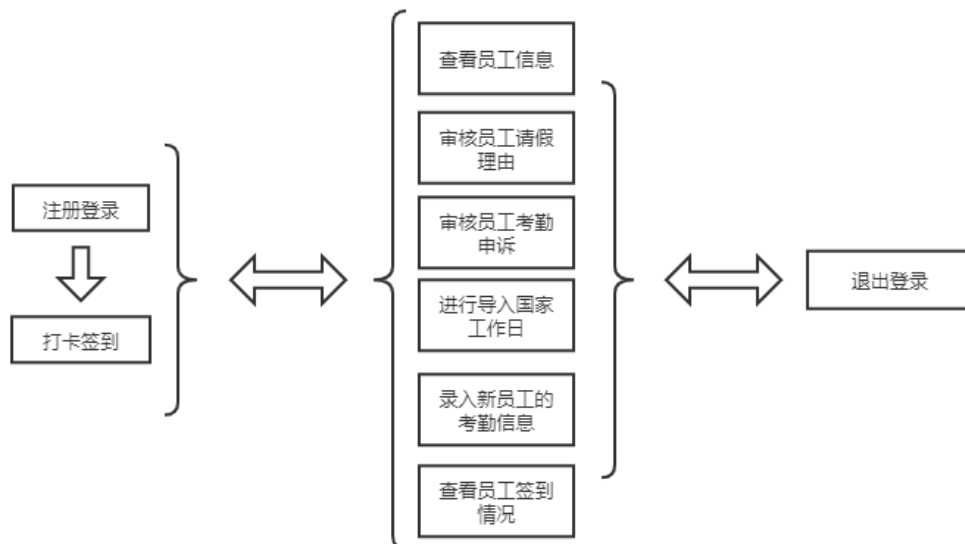
三、业务功能概要结构

本平台作为应用系统的终端，业务需求概要结构图如下：

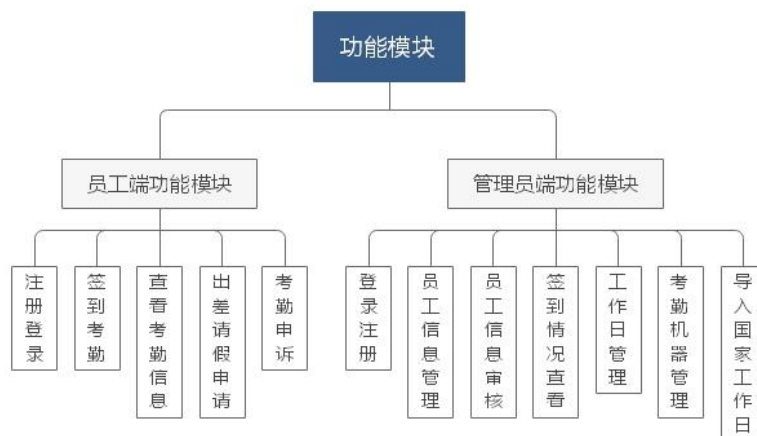
员工端：



管理员端：



业务功能模块如下：



1、员工端功能模块

1.1 注册登录

- 1、用户输入工号和姓名进行登录
- 2、用户忘记密码的，可以通过找回密码重设
- 3、对于初次登录的用户，点击注册进行用户的注册

1.2 签到

- 1、点击签到按钮
- 2、输入工号
- 3、员工看着摄像头念 0-9 十个数字
- 4、将员工工号、姓名、人脸、声纹等数据和请求传至服务器
- 5、服务器判定签到是否成功，返回签到成功与否的信息
- 6、返回签到成功状态，并将其显示在界面上

1.3 查看考勤信息

- 1、用户点击查看考勤信息
- 2、初始化工作日历界面
- 3、用户输入工号
- 4、工作日信息从服务器请求得到
- 5、显示员工的考勤数据

1.4 出差请假申请

- 1、点击出差请假申请按钮

- 2、初始化申请表界面
- 3、填写出差请假申请表
- 4、点击按钮发送申请
- 5、服务器将申请表写入数据库中
- 6、返回申请发送结果

1.5 签到申诉

- 1、点击签到申诉按钮
- 2、初始化签到申诉界面
- 3、填写申诉表
- 4、点击发送
- 5、将申诉信息写入数据库中
- 6、返回信息发送结果，并显示在界面上

2、管理员端功能模块

2.1 管理者注册（管理员权限操作）

- 1、管理员已登录（防止非法注册）
- 2、点击注册按钮
- 3、初始化注册界面
- 4、输入新管理员相关信息
- 5、系统向数据库写入新管理员信息，写入成功之后返回注册成功信息
- 6、界面收到注册成功信息之后显示注册成功，点击功能列表返回其他界面

2.2 管理者登录

- 1、左侧登录按钮被点击
- 2、初始化登录界面
- 3、管理员在输入用户名和密码
- 4、使用用户名、密码信息向服务器发送登录请求
- 5、服务器返回登录结果
- 6、如果登录成功，则显示登录成功并解锁左侧所有按钮。否则提示登录失败

2.3 添加员工（管理员权限操作）

- 1、点击添加员工按钮
- 2、管理员输入新员工工号、姓名
- 3、新员工看着摄像头念 0-9 十个数字
- 4、将员工工号、姓名、声纹、唇语、人脸、做该操作的管理员账号等数据和请求传至服务器
- 5、服务器将训练好的模型写入数据库，返回
- 6、返回添加员工成功与否的信息，并将其显示在界面上

2.4 删除员工（管理员权限操作）

- 1、点击删除员工按钮
- 2、管理员输入员工工号
- 3、将员工工号和删除请求数据传至服务器
- 4、服务器在数据库中找到对应数据并删除
- 5、返回删除成功状态，并将其显示在界面上

2.5 查看整体签到情况（管理员权限操作）

- 1、用户点击查看整体签到表按钮
- 2、初始化签到表界面
- 3、签到表信息从服务器请求得到
- 4、显示该管理员所有员工签到记录

2.6 员工申诉审核（管理员权限操作）

- 1、初始化申诉列表，将所有申诉显示在界面上
- 2、点击某条申诉
- 3、将申诉表从数据库读出，显示在界面上
- 4、点击驳回按钮，删除此条申诉记录；点击同意，删除词条申诉记录并且修改对应的考勤信息
- 5、返回至申请列表界面

2.7 出差请假审核（管理员权限操作）

- 1、初始化申请列表，将申请显示在界面上
- 2、点击某条申请
- 3、将申请表从数据库读出，显示在界面上
- 4、点击驳回按钮，删除此条申请记录；点击同意，删除词条申请记录并且修改对应的工作日为休息日。
- 5、返回至申请列表界面

2.8 设置工作日（管理员权限操作）

- 1、初始化员工列表

- 2、管理员选择员工并点击确定
- 3、初始化工作日列表，选择员工工作日
- 4、将员工数据、工作日数据和考勤时间数据提交至服务器
- 5、服务器将这些数据写入数据库中对应位置，返回写入状态
- 6、界面显示设置成功与否

2.9 出差请假审核（管理员权限操作）

- 1、初始化申请列表，将申请显示在界面上
- 2、点击某条申请
- 3、将申请表从数据库读出，显示在界面上
- 4、点击驳回按钮，删除此条申请记录；点击同意，删除词条申请记录并且修改对应的工作日为休息日。
- 5、返回值申请列表界面

2.10 考勤时间修改（管理员权限操作）

- 1、初始化员工列表
- 2、管理员选择指定日期设置为工作日/考勤日
- 3、将员工数据、工作日数据和考勤时间数据提交至服务器
- 4、服务器将这些数据写入数据库中对应位置，返回写入状态
- 5、界面显示修改成功与否

2.11 导入国家工作日（管理员权限操作）

- 1、初始化员工列表

- 2、管理员点击导入国家工作日按钮

- 3、将这些数据传入服务器，服务器将这些员工的工作日设置成国家工作日，同时相应的结果

2.12 信息导出

- 1、在查看签到表、工作表等查看操作之后之后，点击导出按钮

- 2、信息被导出至默认文件夹中的新 excel 表格

2.13 合法考勤机器添加（管理员权限操作）

- 1、该操作为管理员权限操作

- 2、点击添加该机器为合法考勤机器按钮

- 3、获取本机 MAC 地址

- 4、将该请求和 MAC 地址参数发给服务器，服务器写入数据库

- 5、服务器返回添加结果

- 6、界面上显示是否添加成功

2.14 合法考勤机器删除（管理员权限操作）

- 1、该操作为管理员权限操作

- 2、点击删除该合法考勤机器按钮

- 3、获取本机 MAC 地址

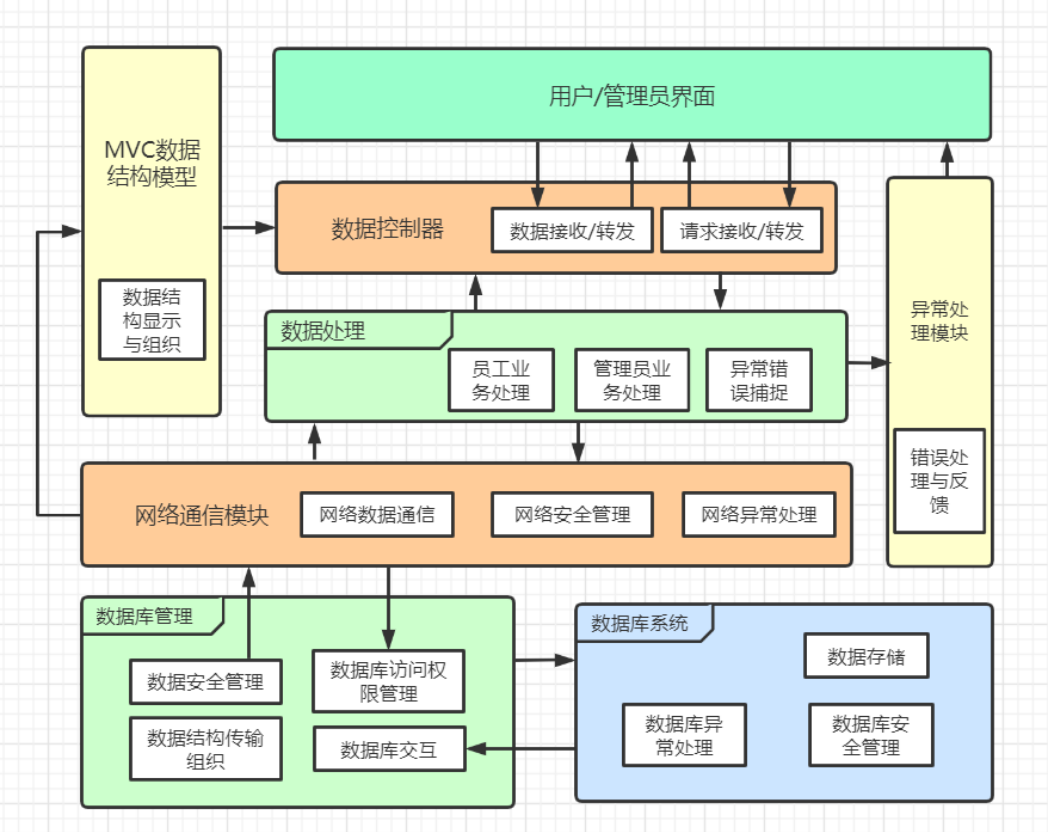
- 4、将该请求和 MAC 地址参数发给服务器，服务器删除数据库中的对应数据

- 5、服务器返回删除结果

- 6、界面上显示是否删除成功

第六部分 总体设计

一、逻辑架构设计



软件系统采用 MVC 设计模式

基于模块化和解耦合的角度分析，整个软件采用 MVC 设计模式，利用 UI 视图界面和平台提供的相应 MVC 视图模型的视图部分构成整体的视图部分。平台提供的 MVC 数据结构和软件系统中的界面，数据控制器和数据处理模块一同构成了整体的 MVC 软件结构。并且在各模块间集成一个完整的异常处理模块以处理业务处理中可能出现的各种异常情况。

业务处理与数据库间采用分层架构思想

使用分层架构构成业务处理模块、网络通信模块、数据库交互模块的整体体系。各个模块只与其相邻层进行交互，以实现软件整体结构的解耦合划分。

各客户端同服务器间采用 C/S 风格

使用 C/S 风格保证软件业务在安全性，事务性和交互性上的要求。为保证客户交互间的性能和质量需求，本软件在网络通信上使用 C/S 风格结构来实现。

二、物理架构设计

服务端

系统服务端可以部署到云平台上，系统管理员通过服务器端实现业务管理操作；

具体应用

系统服务端可以部署到云平台上，作为实际数据的支持者和提供者。

客户端

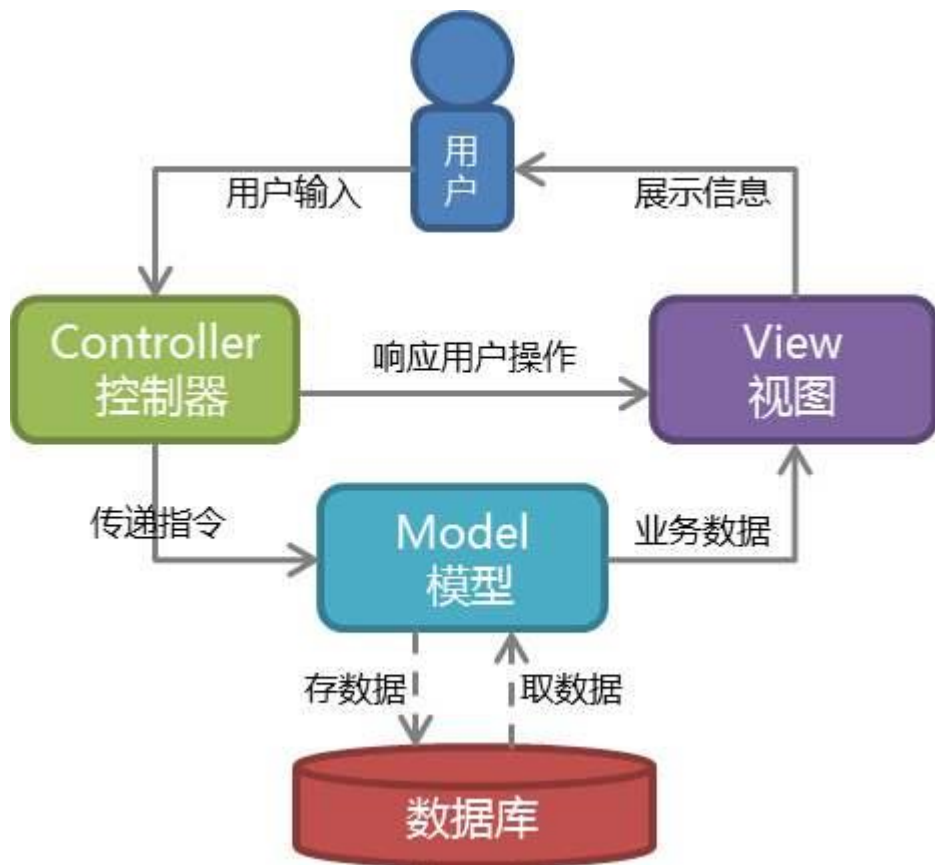
因为系统采用 C/S 模式，考勤机器需要下载相应考勤客户端进行考勤。

管理员端

管理员端需要下载软件的服务端进行对相应管理员账号的人员仓库的人员管理。

三、技术架构设计

MVC 模型设计



本系统主要使用 Qt 平台，利用 MVC 设计模式组织设计并实现整个软件系统。其中，使用 Qt 中提供的视图模型作为基础形成子模组，并且在整体的设计行为中也使用 ui 类、控制器和模型（业务模块）构成整个软件架构体系。

（1）View 模块：

View 负责数据显示与数据输入。数据的输入通过事件交给 Model 层处理，数据显示通过 Model 层的托管实现刷新。

（2）Controler 模块：

每个业务功能的主操作场景采用一个核心控制器类，利用基于 Qt 平台的组件设计实现快速的场景切换设计，由核心控制器控制不同业务处理视图的切换。

View 层通过事件、或者异步线程模式调用控制器的函数，达到把客户响应汇集到控制器。Model 层通过 Controller 中实现的协议方法把数据汇集到控制器。

（3）Model 模块：

Model 层服务数据处理，负责向服务器请求数据，并把数据处理好传递给托管。交

给 View 层完成。

智能身份鉴定技术

（1）人脸识别技术：

本软件将使用人工智能领域的机器学习和深度学习技术，结合人脸的特征图像进行使用者的考勤身份鉴定。

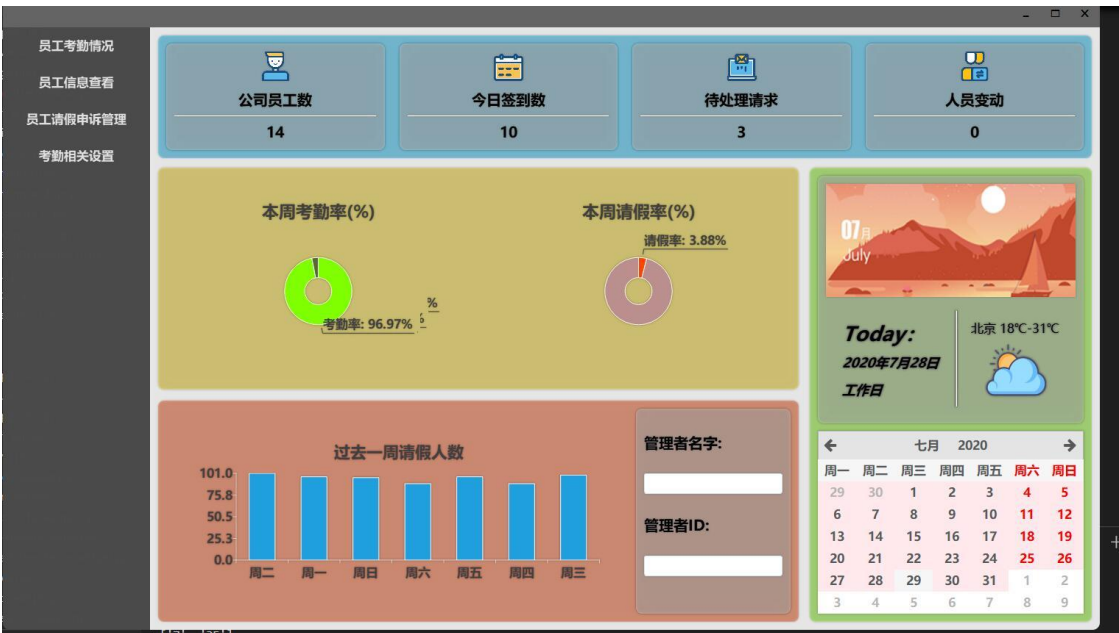
（2）声纹识别技术：

本软件将使用人工智能领域的机器学习和深度学习技术，结合人脸的声纹特征在考勤时使用者的身份进行鉴定。

第七部分 界面设计

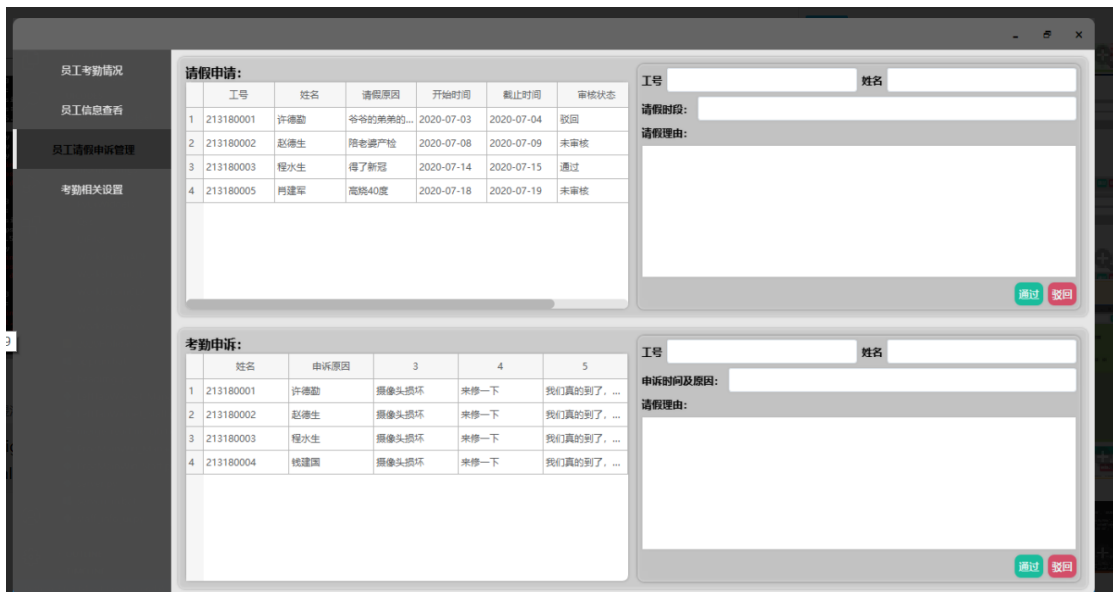
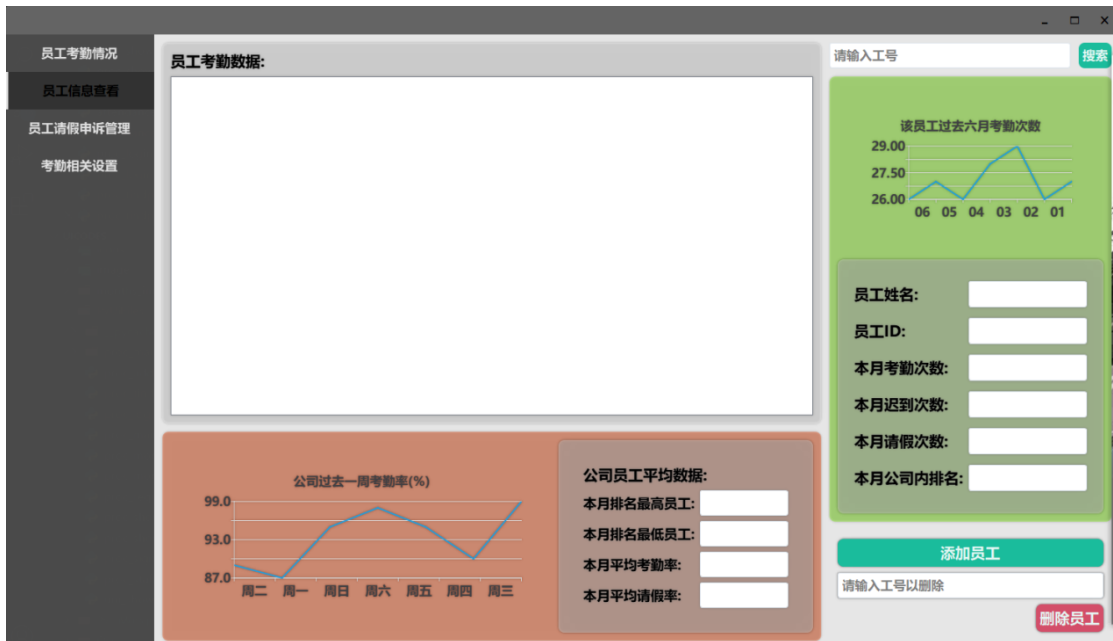
一、风格设计

1、主桌面风格



1、 将按钮排列在界面左侧，管理员未登陆时为员工模式，没有管理员权限，部分功能不能使用，只有在管理员登录后才能使用这些功能

2、业务界面风格展示



统一风格采用左侧的功能列表操作风格。理论上大体分成四区：功能列表区、条件查询输入区、详细信息显示区、信息列表摘要显示区。

员工端采用如下风格设计，其中不同的按钮对应不同的功能，主界面的日历负责对应的考勤数据的可视化。理论上也分为如下区域：功能列表区、条件查询输入区、详细信息显示区、信息列表摘要显示区。



第八部分 运行环境和部署

一、运行环境

1、服务器

Windows Server 10

8GB 以上内存，16GB 内存以上较优；

500GB 以上的硬盘空间，不包括数据存储空间

DB: MySQL 8.0

2、客户端环境

Pentium III 或以上微处理器 (CPU)

Microsoft Windows 8 及更高版本操作系统

1GB 以上内存，建议使用 2GB 内存

摄像头 800 万像素及以上

48db 及以上灵敏度的麦克风

二、系统性能要求

#	项目	模块	级 别	技术参数
1	设计 实现技术 指标	系统架构	A	采用顶端 MVC 和底端分层架构
2		面向对象 开发语言与框 架	A	采用主要采用 python 和 C++ 语言进行开发
3		注释和文 档	A	符合 CMMI 软件开发过程标准文档（至少提供：需求、概要、详细设计、测试报告、部署和环境、用户手册），代码注释量>=30%。
4		模块化和 适合实训	A	SOA 设计、模块化，保证系统各模块单元较强的独立性适合实训教学。
5		测试覆盖 率	A	功能覆盖率>=100%，业务覆盖率>=100%，语言覆盖率>=100%，逻辑覆盖率>=80%。
6	资源 利用率要 求	CPU 占用率	B	<=30%的系统占用率
7		内存使用 率	B	<=75%利用率（附近标准配置）

8	响应时间要求	服务器	B	≤100ms（附近标准配置）
9		网络	B	≤100ms（附近标准配置）
10		客户端	B	≤5s（附近标准配置）
11	系统稳定性要求	成熟性	A	真实的用户，成功使用本系统
12		稳定性	A	无故障运行时间≥365 天，系统恢复时间≤2 小时。
13		先进性	A	采用目前体验最好、最流行的 iPhone 与 iPad 移动终端。
14		典型意义	A	案例项目要有典型意义，有推广价值。
17	Web 服务接口要求	Web 平台端	A	
18	集成部署环境	服务器	A	Microsoft Windows XP/2003 Server
19		数据库	A	MYSQL BD 100g 以上

1	2		考勤端	A	Microsoft Windows XP 及以上
---	---	--	-----	---	--------------------------