学号_____ 密级_____

东北大学本科毕业论文

基于安卓平台的消防管理系统的设计与实现

学 院 名 称：软件学院

专 业 名 称：软件工程

学 生 姓 名：[redacted]

指 导 教 师：[redacted] 教授

[redacted] 工程师

2019 年 6 月

Design and Implementation of Firefighting Management System Based on Android Platform

by [redacted]

Supervisor: Professor [redacted]
Associate Supervisor: Engineer [redacted]

Northeastern University

June 2019

基于安卓平台的消防管理系统的设计与实现

作者姓名：[redacted]

校内指导教师：[redacted] 教授

校外指导教师：[redacted] 工程师

单位名称：软件学院

专业名称：软件工程

东 北 大 学

2019 年 6 月

毕业设计（论文）任务书

毕业设计（论文）题目：

基于安卓平台的消防管理系统的设计与实现

基本内容：

本课题要求对消防器材管理进行调研，分析管理过程中的薄弱环节，开发一款辅助消防人员进行生产管理的安卓应用，并实现器材信息的云端存储和多人共享。另外，本课题要求以增强现实辅助消防器材管理为形式创新点，以本科软件工程领域的相关知识作为支撑，立足于消防部门职员的现实需求，开发出一套结合学习测验、指令传达和器材的识别管理监控为一体的移动端应用。

翻译一篇与毕设内容相关的外文资料，译文汉字字数不少于 4000 字。

毕业设计（论文）专题部分：

题目：

基本内容：

学生接受毕业设计（论文）题目日期

第 1 周

指导教师签字：

2019 年 3 月 9 日

Solemn Statement

The graduation thesis submitted by me is the result of independent research work under the guidance of the tutor. All data and pictures are true and reliable. As far as I know, the research results of this thesis do not include the content of copyright enjoyed by others, except as indicated in the text. Other individuals and groups that have contributed to the research work involved in this paper have been clearly identified in the text. The intellectual property rights of this thesis are attributed to the training unit.

Signature: Date:

ABSTRACT

With the maturity of software technology and the improvement of hardware foundation, mobile Internet technology has been rapidly applied to many fields, which has greatly changed the way people live and work. Therefore, in order to meet the actual needs of employees in the work process, designing a mobile phone application that assists employees to work will have great practical value. However, there are few firefighting management applications on the market and those applications' user experience is poor. The main functions of the system include querying device information, knowledge learning, sending messages, and identifying firefighting equipment. The main goal of the system is to provide system users with a series of practical functions, assist relevant personnel to manage firefighting equipment and improve the automation of the equipment management process.

This topic mainly studies the firefighting management system based on the Android platform. By analyzing the actual needs of the firefighting department, the system allows users to query all devices in a certain area or information on a single device. In the system, users can study different types of learning materials, such as e-books or video lessons; after learning, users can test their understanding of specific topics and review their results; they can receive commands and announcements from managers, manage their task lists, and schedule their tasks. Finally, QR code, AR and other technologies are used to identify specific devices and display related information. The identification part is the key to the system implementation and the core of the firefighting department's needs. The efficiency and accuracy of device identification are directly related to the user experience and implementation of the entire system. The system realizes the basic functions of the firefighting department management platform and reduces operational difficulties during the actual inspection through AR technology, which efficiently meets the functional requirements of different types of users in the management platform.

After analyzing the actual needs of the firefighting department, the system has designed a corresponding system architecture and implemented a series of functional modules. After sufficient use case testing, the system basically met the project requirements of the firefighting department. The system can operate correctly and has been applied to some

firefighting departments, which greatly simplifies the management of firefighting equipment and provides great convenience for the staff.

Keywords： Android; Augmented Reality; Firefighting Management; Unity3D; Vuforia

摘 要

随着软件技术的成熟和硬件基础的提升，移动互联技术被迅速应用于多个领域，极大地改变了人类的生产生活方式。因此，针对员工在工作过程中的实际需求，设计一款辅助员工工作的手机应用，将具有极大的实际价值。然而，市场上的消防管理应用数量较少，且用户体验较差。本系统的主要功能包括：查询设备信息、知识学习、发布命令和识别消防器材等。本系统的主要目标是为系统用户提供一系列具有实际价值的功能，并辅助相关人员对消防器材进行监控和管理，提高器材管理过程的自动化程度。

本课题主要研究了基于安卓平台的消防部门管理系统，通过分析消防部门的实际需求，该系统允许用户对某一区域内的所有设备或对单一设备的信息进行查询。在系统中，用户可以学习不同类型的学习资料，如电子书或视频课程；在学习之后，用户可以对某一知识点进行检测，并查看对知识点的掌握情况；用户可以接收来自管理者发布的命令和公告，也可以对当前的任务列表进行管理并制定日程；最后，综合运用二维码、AR 等技术，对特定设备进行识别，并显示相关信息。设备识别部分是系统实现的关键，也是消防部门需求的核心。设备识别的效率和准确度，直接关系到整个系统的用户体验和实现效果。该系统实现了消防部门管理平台的基本功能，并通过 AR 技术降低了实际巡检时的操作难度，使之更加高效地满足了消防部门不同类型用户对于管理平台的功能需求。

经过对消防部门实际需求的分析，本系统设计了一套相应的系统架构，并实现了一系列功能模块。经过充分的用例测试，本系统基本达到了消防部门的项目需求。该系统能够正确运行并且已经运用到消防部门中，极大地简化了消防器材的管理过程，为职员提供了极大的便利。

关键词：Android，增强现实，消防管理，Unity3D，Vuforia

Content

Chapter 1 Introduction

1.1 Research Background

As the world's most popular mobile platform, Android has developed for more than ten years. The first Android smartphone was released in October 2008. As time progressed, Google continued to improve the version of Android, and various hardware manufacturers also continuously launched new Android-based mobile phones. At present, Android is gradually expanding to tablet computers, TVs, cars, watches, digital cameras, game consoles, smart home appliances and other fields. In addition to the version number of the Android operating system, the Android API also has a version number, which is called the Android level. As the Android operating system is upgraded, the API level changes.

In order to provide users with better user experience and more flexible functionalities, the Android application in this paper embeds a Unity project, in which Augmented Reality Engine is used to recognize images. Augmented reality technology [1] is an emerging technology that superimposes computer-generated virtual information into the real world where users are located. The virtual information such as taste and touches, through computer technology, is simulated and then applied to the real world, perceived by human senses, thus achieving a sensory experience beyond reality. It is an important branch of virtual reality technology. It enhances the user's perception of the real world and provides a new way for humans to communicate with the world [2].

1.2 Research Significance

Due to the improved performance and price reduction of mobile phone and software devices, more and more diversified applications will be developed by technology companies, allowing augmented reality technology to quickly enter the firefighting department. Relying on its superior experience, such as interactivity, immersion, conception, real-time interaction, virtual and real-time combination and three-dimensional

registration, the mode of staff sending or receiving messages and the way of employees managing firefighting equipment will be changed. The use of augmented reality technology in the inspection process of equipment will subvert the traditional equipment management mode, which has great application prospects.

With the expansion of the firefighting station scale, some hidden danger problems have gradually emerged. Due to the large number of firefighting equipment, extensive spaces to monitor and great difficulty of management, the information of equipment is not displayed and updated in time, which leads to the existence of low efficiency in the work process. Many dangers are hidden in the back. Therefore, the significance of this research is to make the function of the new firefighting management system meet the construction requirements of the "smart firefighting" system.

(1) Allow video display function to achieve fully dynamic display of area. The new firefighting management system can visually monitor firefighting equipment and automatically check the current status of the equipment. The system can help to grasp the situation of daily patrol work of the staff and maintenance personnel.

(2) Realize the function of firefighting with mobile end. The traditional firefighting system does not have the technical means of information management, and the firefighting management of the staff relies entirely on the traditional management methods of people. Especially for firefighting equipment, materials consumption and stocks, traditional management methods require a lot of manpower, and it is difficult to avoid the omission of grassroots staff negligence. In daily management, the traditional firefighting management system cannot automatically upload online tasks through mobile phones, automatically record the inspection, rectify the automatic tracking function and upload the hidden danger rectification record. Therefore, daily equipment inspection and hidden danger investigation work cannot be mastered in time, and supervision of these tasks requires additional labor costs.

(3) It is beneficial to strengthen the learning situation of employees and analyze the data of employees' learning history. It is conducive to the Group's comprehensive analysis of the overall firefighting management, equipment status, hidden danger rectification and so on.

1.3 Related Research

At present, the research and construction of "smart firefighting" system in China are at the initial stage, although many firefighting stations have built a set of so-called "smart" system. However, in their case, it can be found that the "smart firefighting" systems that have been built are all in the stage of "Internet of things data display" and "urban fire remote monitoring system" that are operated manually, and the fire control room still needs two people and two ports to be on duty 24 hours a day. The current intelligent fire control system can only provide part of the functions of monitoring instead of controlling. Therefore, after the deployment of the system, the same number of watchmen should not only take care of the newly deployed platform, but also the traditional fire control equipment, which has not achieved the goal of reducing staff and increasing efficiency of the group.

However, there are many other studies similar to the case of "smart firefighting" system. In 2013, a group in University of Aveiro, Portugal developed an Android system of Firefighter team breathing management [4]. In 2018, a team in Department of Electronics Engineering, University of Gezira developed a warning firefighting system using Internet of Things [5], which has influenced the career of firefighting greatly. Related work includes mobile team monitoring [7] and studies of firefighting and respiratory physiology [6,8,9].

Using Augmented Reality in the field of firefighting management is a brand-new approach. Augmented reality is defined as a system that satisfies three characteristics [10,11,12]: first, combining real and virtual worlds; second, allowing real-time interaction; and third, aligning real and virtual objects in three-dimensional space. The system characteristics of AR technology make it applicable to many fields. In the field of education, Kokaew et al. attempted to support hearing-impaired students by using mobile phone applications to superimpose virtual reality scenes in the real world, and achieved good results in teaching evaluation [13,14]. In the medical field, Rashed et al. proposed an augmented reality imaging system for minimally invasive orthopedic surgery that maps X-ray images onto real objects, significantly reducing the number of X-ray

exposures during surgery, saving time and improving safety [15]; Seongpung Lee et al. studied the method of calibrating an endoscope to an optical tracking system (OTS) for use in surgery based on high-precision augmented reality, improving the accuracy of AR visualization for surgical navigation [16]. In the field of corporate training, training technicians to acquire new maintenance and assembly skills is important for all industries. Because maintenance and assembly tasks can be complex, it is challenging to train technicians to effectively implement new skills. In response to this situation, Mitrovic et al. attempted to combine augmented reality with Intelligent Tutoring System (ITS) to assist in the training of manual assembly tasks and achieve significant efficiency gains in corporate practice [17, 18, 19]. In the retail shopping arena, Dacko et al., by mobile augmented reality (MAR) applications, attempt to create additional value for customers and benefit retailers to describe the extent to which MAR applications contribute to the smart retail environment [20].

1.4 Research Content

The goal of this graduation project is to design a firefighting management application that will provide a series of functions, which are useful during the working process. The innovation and original intention of this system is to assist user to perform augmented reality immersive interactive experience on video streams, and to enable leaders and staff with more new forms of communication, knowledge learning and managing firefighting equipment. This design utilizes augmented reality technology to extend the universally applied static picture recognition to the orderly recognition of video streams, and to superimpose interactive information with real-world video content to create a more immersive viewing experience. The interactive system based on augmented reality technology makes the management method of the equipment more diverse, and the way of communicating and learning makes it more convenient for users to have their daily work experience.

The main research contents of this graduation project include technical solutions for AR recognition of video, interaction process and interaction mode design and software function display forms. Therefore, the system is divided into account module, study

module, exam module, message module and equipment management module.

1.5 Thesis Organization

The structure of this thesis is arranged as follows:

Chapter 1 Introduction. It will introduce the technical background and project significance of this graduation design project, explain the technical principles and practical application values used in this project. This chapter introduces the current research related to this graduation design at home or abroad. This chapter will also summarize the content of this paper and introduce the organizational structure of this article.

Chapter 2 Key Technologies. It will introduce the key technologies and solutions for applying the technology in the development of firefighting management systems based on Android platform and Augmented Reality technique.

Chapter 3 Requirement Analysis. It will introduce the content of requirement analysis, including use case analysis, the economic possibilities and technical possibilities of firefighting management system based on Android Platform.

Chapter 4 System design. The detailed design of firefighting management system will be introduced in this chapter. Sequence diagram will be used to introduce the key modules design and E-R diagram will be used to introduce the database design.

Chapter 5 System Implementation. It describes how the firefighting management system based on augmented reality technology is implemented step by step according to the system design plan and elaborates on the implementation of several core modules or core functions.

Chapter 6 System Testing. Test procedure and test results of a firefighting management system based on Android platform and augmented reality technology are described in this chapter.

Chapter 7 Conclusion. It will summarize development of the firefighting management system based on Android Platform and the application prospects of this project will be presented.

Chapter 2 Key Technologies

2.1 System Architecture of Android

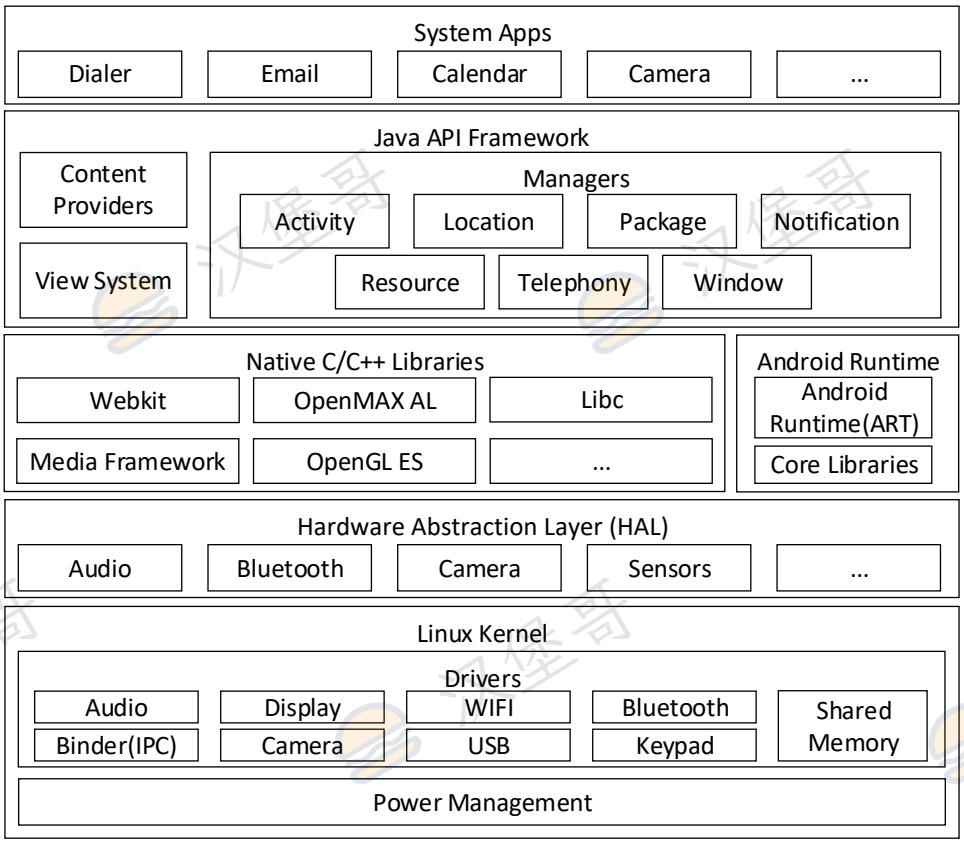

Figure 2.1 Android System Layers

As shown in Figure 2.1, Android is implemented in the form of a software stack architecture consisting of a Linux kernel, many hardware abstractions, a runtime environment and corresponding libraries, a Java API Framework and a set of applications.

The Linux kernel is the foundation of Android platform, which provides many underlying functionalities such as low-level memory management and threading.

The hardware abstraction layer (HAL) exposes device hardware capabilities to the higher-level Java API framework by providing many standard interfaces.

Android Runtime (ART) provides core libraries for an application to run. In addition, many native libraries are provided.

Java API Framework provides the entire feature-set of the Android OS. These APIs form the building blocks you need to create Android apps by simplifying the reuse of core, modular

system components and services. System Apps layer is on top of the Java API Framework.

By using Java API frameworks, applications can access libraries such as SQLite and Media Framework in the application developed in this paper. Besides, since the system is required to scan video streams and present information through different types of media, many hardware drivers such as Camera and Audio in the Linux Kernel need to be called through Hardware Abstraction Layer.

2.2 Core Components of Android

2.2.1 Activities

An activity provides a screen in Android, similar to a frame or window in Java. UI components or widgets can be placed on this screen.

An activity facilitates the following key interactions between system and app:

(1) Keeping track of what the user currently cares about (what is on screen) to ensure that the system keeps running the process that is hosting the activity.

(2) Knowing that previously used processes contain things the user may return to (stopped activities), and thus more highly prioritize keeping those processes around.

(3) Helping the app handle having its process killed so the user can return to activities with their previous state restored.

(4) Providing a way for apps to implement user flows between each other, and for the system to coordinate these flows. (The most classic example here is sharing)

2.2.2 Services

Service is an approach of running programs in the background in the system of Android. It is suitable for performing tasks that require long-term operation instead of too much interaction. By default, a service runs in its application process, but it can run in a separate process configured with android:process. The services can only interact with other components in the background. Another component, such as an activity, can start the service and let it run or bind to it in order to interact with it. There are two very distinct semantics services tell the system about how to manage an app: Started services tell the system to keep them running until their work is completed. This could be to sync some data in the background or play music even after the user leaves the app. Syncing data in the background or playing music also represent

two different types of started services that modify how the system handles them:

Music playback is something the user is directly aware of, so the app tells the system this by saying it wants to be foreground with a notification to tell the user about it; in this case the system knows that it should try really hard to keep that service's process running, because the user will be unhappy if it goes away.

A regular background service is not something the user is directly aware as running, so the system has more freedom in managing its process. It may allow it to be killed (and then restart the service sometime later) if it needs RAM for things that are of more immediate concern to the user.

Bound services run because some other app (or the system) has said that it wants to make use of the service. This is basically the service providing an API to another process. The system thus knows there is a dependency between these processes, so if process A is bound to a service in process B, it knows that it needs to keep process B (and its service) running for A. Further, if process A is something the user cares about, then it also knows to treat process B as something the user also cares about. Because of their flexibility (for better or worse), services have turned out to be a useful building block for all kinds of higher-level system concepts. Live wallpapers, notification listeners, screen savers, input methods, accessibility services, and many other core system features are all built as services that applications implement and the system binds to when they should be running.

A service is implemented as a subclass of Service. For more information about the Service class, see the Services developer guide.

2.2.3 Content Providers

A content provider is a component to manage application data that is stored in the file system, in a cloud database or any other storage places that users' application can connect with. Other applications can access data if corresponding content providers allow it. For example, the Android system provides a content provider that manages the user's contact information. Any application with the proper permissions can query the content provider, to read and write information about a person. It is tempting to think of a content provider as an abstraction on a database, because there is a lot of API and support built into them for that common case. However, they have a different core purpose from a system-design perspective. To the system,

a content provider is an entry point into an app for publishing named data items, identified by a URI scheme. Thus, an app can decide how it wants to map the data it contains to a URI namespace, handing out those URIs to other entities which can in turn use them to access the data. There are a few things this allows the system to do in managing an app:

(1) Assigning a URI doesn't require that the app remain running, so URIs can persist after their owning apps have exited. The system only needs to make sure that an owning app is still running when it must retrieve the app's data from the corresponding URI.

(2) These URIs also provide an important fine-grained security model. For example, an app can place the URI for an image it has on the clipboard but leave its content provider locked up so that other apps cannot freely access it. When a second app attempts to access that URI on the clipboard, the system can allow that app to access the data via a temporary URI permission grant so that it is allowed to access the data only behind that URI, but nothing else in the second app.

A content provider is implemented as a subclass of ContentProvider and must implement a standard set of APIs that enable other apps to perform transactions. For more information, see the Content Providers developer guide.

2.3 Important Concepts of Unity3D

2.3.1 Scenes

Scenes contain the environments and menus of your game. Think of each unique Scene file as a unique level. In each Scene, you place your environments, obstacles, and decorations, essentially designing and building your game in pieces.

2.3.2 GameObjects and Components

The GameObject is the most important concept in the Unity Editor.

Every object in your game is a GameObject. This means that everything you can think of to be in your game has to be a GameObject. However, a GameObject can’t do anything on its own; you need to give it properties before it can become a character, an environment, or a special effect.

A GameObject is a container; you add pieces to the GameObject container to make it into a character, a light, a tree, a sound, or whatever else you would like it to be. Each piece

you add is called a component.

Depending on what kind of object you want to create, you add different combinations of components to a GameObject. You can think of a GameObject as an empty cooking pot, and components as different ingredients that make up your recipe of gameplay. Unity has lots of different built-in component types, and you can also make your own components using the Unity Scripting API.

2.3.3 Prefabs

Unity's Prefab system allows users to create, configure, and store a GameObject complete with all its components, property values, and child GameObjects as a reusable Asset. The Prefab Asset acts as a template from which users can create new Prefab instances in the Scene.

When users want to reuse a GameObject configured in a particular way – like a non-player character (NPC), prop or piece of scenery – in multiple places in users' Scene, or across multiple Scenes in users' Project, users should convert it to a Prefab. This is better than simply copying and pasting the GameObject, because the Prefab system allows users to automatically keep all the copies in sync.

Any edits that users make to a Prefab Asset are automatically reflected in the instances of that Prefab, allowing users to easily make broad changes across users' whole Project without having to repeatedly make the same edit to every copy of the Asset. Users can nest Prefabs inside other Prefabs to create complex hierarchies of objects that are easy to edit at multiple levels. However, this does not mean all Prefab instances have to be identical. Users can override settings on individual prefab instances if users want some instances of a Prefab to differ from others. Users can also create variants of Prefabs which allow users to group a set of overrides together into a meaningful variation of a Prefab. Users should also use Prefabs when users want to instantiate GameObjects at runtime that did not exist in your Scene at the start - for example, to make powerups, special effects, projectiles, or NPCs appear at the right moments during gameplay.

Some common examples of Prefab use include:

(1) Environmental Assets - for example a certain type of tree used multiple times around a level.

(2) Non-player characters (NPCs) - for example a certain type of robot may appear in your game multiple times, across multiple levels. They may differ (using overrides) in the speed they move, or the sound they make.

(3) Projectiles - for example a pirate's cannon might instantiate a cannonball Prefab each time it is fired.

(4) The player's main character - the player prefab might be placed at the starting point on each level (separate Scenes) of users' game.

2.4 Vuforia Engine

Vuforia Engine is the most widely used platform for AR development, with support for leading phones, tablets, and eyewear. Developers can easily add advanced computer vision functionality to Android, iOS, and UWP apps, to create AR experiences that realistically interact with objects and the environment.

The Vuforia Engine Library provides technical documentation to help developers use the SDK and create AR applications. The documentation is organized by the platforms supported by Vuforia Engine. These include the Unity Engine and the three native platforms: iOS, Android and UWP.

The Vuforia Engine Library is also organized by major feature categories. The major feature categories are Images, Objects and Environments.

The Vuforia Engine API documentation includes C# APIs for Unity, C++ APIs for iOS and UWP, and Java APIs for Android.

2.5 Summary

This chapter is based on the research of Android platform and application development. First, the architecture of the Android system is introduced, then the important components of Android are introduced, and finally the characteristics of the Android platform are summarized. This chapter lays a theoretical foundation for the design and implementation of the system. The next chapter describes requirement analysis through functional analysis, use case analysis, and feasibility analysis.

Chapter 3 Requirement Analysis

The management system aims to increase work efficiency in the firefighting department by using Android and AR technique. Currently, the client of this project, a local firefighting department, lacks an application to manage all kinds of equipment and to integrate many useful functions for their staff. Therefore, the main functions of the system are detecting equipment, publishing and receiving commands, studying through different ways, etc.

In this section, the overall project will be introduced and some methods including use case analysis and feasibility analysis will be provided to analyze the system from different aspects. The system analysis is as follows.

3.1 Project Overview

The firefighting management system is designed to provide users with user-friendly software based on the Android platform. The main plan is to provide various functions and especially present the information of the scanned equipment to the user through image recognition, which greatly simplifies the process of staff managing the equipment.

This system uses AR technology to assist quality inspectors in managing fire equipment. Using this system, quality inspectors can obtain equipment information more conveniently and intuitively and master the overall situation of firefighting equipment. The system is a subsystem of the fire department software, which has practical application value. The system can greatly reduce labor costs, improve the office efficiency of the fire department, and has great social benefits.

There are several problems lying in the past workflow of firefighting department.

(1) Lacking a unified and convenient way to learn professional knowledge. It is time-consuming for staff to collect study resources including books and videos that are scattered across the Internet.

(2) Publishing and receiving commands are inefficient. Like many other administration departments, leaders usually publish commands by making a phone call or passing a document. Those approaches are sometimes unreliable and complicated.

(3) Managing firefighting equipment manually consumes a lot of manpower and time. Furthermore, since equipment is monitored and recorded by people, some flaws and errors are

inevitable.

Therefore, in order to solve those problems, an application based on Android platform is developed in this paper. Staff in the firefighting department can conveniently use this app when they are working. Detailed function requirements are described in Table 3.1.

Table 3.1 Function description

No.	Function Requirements	Description
1	Register	New users need to register and confirm their identities. A leader can publish commands while staff can receive commands
2	Login	A user can log in the system with correct username and password
3	Read Book	The system provides a list of book resources; A user can find target book quickly by using filter and sort functions
4	Watch video	Professional video lessons are displayed to staff with brief introduction
5	Do exercises	Staff can test their mastery of knowledge by doing exercises.
6	Scan Equipment	Detect equipment and display information bound to this equipment
7	Publish Command	A leader needs to send message or command to a single person or to a group of staff at the same time.
8	Receive Command	Common users can read messages sent by a leader
9	View Announcement	A type of message that broadcasts to all staff
10	View Regional Equipment status	Display all equipment’s information in a specific area
11	Schedule tasks	Users can make tasks according to the schedule and mark the completion level of different tasks

3.2 Use Case Analysis

The use case diagram is used to describe the customer's need, which is what specific

function that staff in a firefighting department want the application to achieve. As the starting point of the requirement analysis phase, the common understanding of the use case is the functional module of the software. Therefore, in order to give an overall glimpse of the whole system, the use case diagram, Figure 3.1, is described as follows.

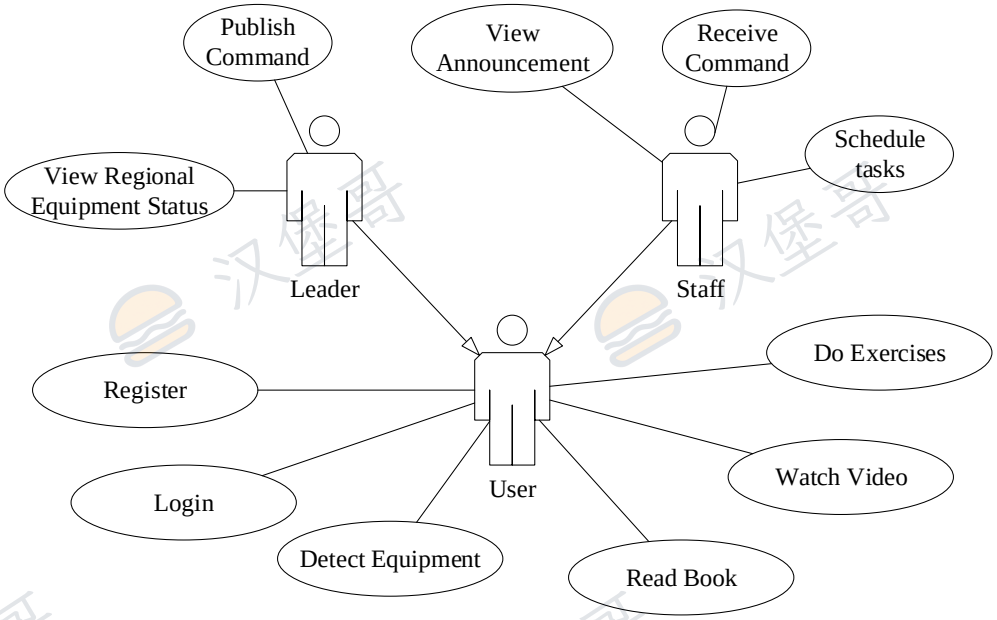

Figure 3.1 Overall use case diagram

3.2.1 Account Module

When a leader or staff member opens this application, they need to register an account first. A user needs to provide his username, password, gender, phone number, e-mail address and identity code. If the identity code belongs to a leader, a leader account will be created; otherwise, a staff account will be created. Every user can log in the system by providing correct username and password. Table 3.2 shows the description of Login.

Table 3.2 Use case description of Login

Use case No.	1
Brief Description	Log in the system
Actors	User
Pre-Condition	Input username and password
Post-Condition	Enter main page or display failure information
Flow of Event	1. User inputs username and password 2. User clicks login 3. System checks the account credentials and responds

3.2.2 Study Module

Study module contains two parts: reading books and watching videos. A user can view a list of professional books and users can sort books by different standards such as popularity and created time. Users can also filter out some books if they don't want to view books that they have already read, for example. The system should provide a PDF reader to display the book resources that a user chooses. A user can also view a list of videos. There should be a picture and a brief introduction of each video in the list. After a user chooses to watch one video, a set of operations on this video such as pause, play, forward should be provided. The user should be allowed to make comments on videos.

The Use case diagram of Study modules is as follows.

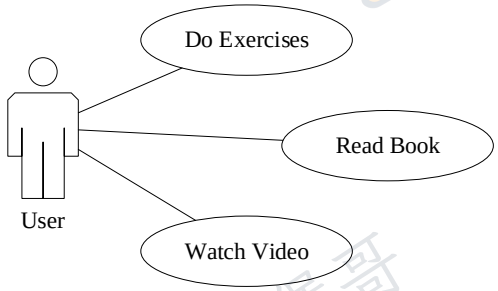

Figure 3.2 Use case diagram of study module

Table 3.3 Use case description of submitting an exam

Use case No.	2
Brief Description	User submits a set of questions
Actors	User
Pre-Condition	All questions have been completed
Post-Condition	Display scores and question analysis
Flow of Event	1. User clicks to submit an exam 2. System compares user's answers with correct answers 3. System calculates the user's score and records the user's exam history 4. System displays scores and question analysis

3.2.3 Exam Module

The exam module is used to help users review knowledge that is learned through study module. Therefore, the user should be allowed to see where a set of questions comes

from (which book or video). Users can view a list of exams and basic information of these exams will be presented. After choosing one exam, the first question will be displayed to users. The user will be required to complete the question according to the question type (Multiple choices, Single Choice, etc). After a question has been completed, the user can choose to continue to the next question or jump to a random question. During the process of doing exercise, the user should be allowed to view the overall completion degree and collect some questions. The exam history should be graded and recorded if the exam is submitted by a user. Table 3.3 shows the description of submitting an exam.

3.2.4 Message Module

In message module, a leader can send a command to one person or a group of people. A leader can also release an announcement that will notify every staff member of this department. Staff can view announcements and receive commands. Besides, Staff can schedule tasks according to received commands. Based on the deadline of each command and personal timetable, a schedule can be made, and the system will notify users to achieve tasks on time.

Table 3.4 describes the use case of Publish Command from the aspect of a leader.

Table 3.4 Use case description of Publish Command

Use case No.	3
Brief Description	Leader publishes command to selected staff
Actors	Leader
Pre-Condition	The leader has logged in the system
Post-Condition	Display commands' status
Flow of Event	1. Leader clicks to publish commands 2. System provides the list of available staff 3. Leader selects staff and edits the command 4. System sends the command to the accounts of selected staff

3.2.5 Equipment Management Module

A leader can view all equipment's status in an area. All information will be grouped by equipment's type. The status of a piece of equipment should contain device number, introduction and some specific parameters of this equipment.

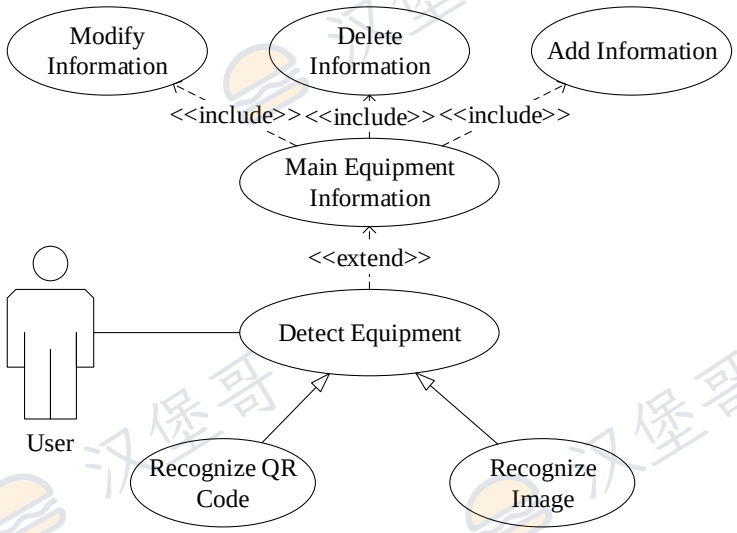

Figure 3.3 Use case diagram of equipment management module

Table 3.5 Use case description of Recognize Image

Use case No.	4
Brief Description	user chooses to detect equipment by recognizing target images
Actors	User
Pre-Condition	The user has logged in the system
Post-Condition	display equipment's status
Flow of Event	1. User clicks to detect equipment 2. System requests the camera video stream 3. System processes video and detects target images 4. System requests equipment data from the database 5. System displays equipment's information

Except for viewing regional equipment information, every user of this system can view specific equipment by camera detection. There should be two ways to recognize a device. The first approach is recognizing QR code. In this circumstance, one piece of equipment is bound to a specific QR code and if this QR code is detected, the information of this equipment will be displayed to users. The second method is using AR technique to recognize target images. In this situation, the camera interface needs to be called to provide video stream. After target images are detected, information of corresponding equipment should be displayed. The user can add information, as well as delete and modify it. In Figure 3.3, the

function of detecting equipment through AR or QR technique will be described.

In Table 3.5, event flow of recognizing image through AR technique will be described, which is like the process of recognizing QR code. The user can edit the information of Equipment after the system detects an equipment.

3.3 Feasibility Analysis

3.3.1 Technical Feasibility

Based on Android platform, this system can use many plugins such as ZXing QR scanning framework and video player framework. ZXing QR scanning frame can implement the function of detecting QR code and enable software developer to focus on steps after QR code being detected. The video player framework encapsulates the main operations on a video. The system can embed a subsystem of Unity project. Unity is a multi-platform, integrated game development tool that allows players to easily create interactive content such as 3D video games, architectural visualizations, real-time 3D animations, and more. Through Unity engine and Vuforia AR engine, the system can develop a subsystem to recognize target images by using AR technique. Besides, user interface of the subsystem can be designed in Unity in order to present better user experience. After target images are recognized, some virtual elements can be displayed on the mobile phone's screen and 3DMAX can be used to design those virtual elements.

The system has the advantages of high scalability and compatibility. The software is more convenient to upgrade later and can be flexibly configured according to the actual situation of the measurement. New functional modules and sub-software can be added according to different needs to meet the individual needs of users. Software provides data interfaces and application interfaces for easy software integration and secondary development to ensure efficient use of existing resources.

3.3.2 Economic Feasibility

From the input cost of the firefighting management system based on Android platform and augmented reality technology, tools such as Unity 2018.4.0f1, Adobe Photoshop, Vuforia AR SDK and NGUI used in system development have freely available versions. The required computer specifications do not need to be high, so the development cost of the firefighting management system based on Android Platform is extremely low.

The market prospects of the firefighting management system based on Android and AR are described below. In the recent past, AR identification of goods in warehouses is very popular, and this process is very similar to identifying equipment in the firefighting department, so it has a large market. Augmented reality technology can solve the problem of the current firefighting department's efficiency in equipment management. This technology provides great convenience for quality inspectors, providing staff with a variety of forms of digital content and a combination of virtual and real contextualized work environment, enhancing the sense of presence and immersion of staff at work. Through augmented reality technology, virtual scenes can be combined with the real world, and communication and interaction through time and space can enhance the perceptual knowledge and real experience of the staff. The cost of this project is not high, and it is easy to get started. It is very suitable for the popularization of the firefighting department in the country. The market prospects are considerable.

3.4 Summary

This chapter conducts specific technical feasibility analysis and economic feasibility analysis of firefighting management system based on Android platform through detailed overall project function introduction, use case diagram analysis, functional analysis and non-functional analysis. This shows that the project is feasible and meaningful in terms of technology development and market prospects.

Chapter 4 System Design

In the previous chapter, system analysis has described the basic requirements of this application. The system is divided into five modules: account module, study module, exam module, message module and equipment management module. In this section, architecture design and detailed design of five function modules will be described.

4.1 Architecture Design

As shown in Figure 4.1, the architecture of the system is mainly divided into two modules: Unity3D and Android. The Unity3D Project part mainly binds preset images and model information through C# scripts. The model resources are provided by the modeling tool 3DMAX; Android Application part designs the user interface, and writes the background logic through Java code, realizes the interaction between Android and Unity3D by calling unity-classes.jar and Vuforia.jar files, and finally realizes the cloud database storage of application information through MySQL.

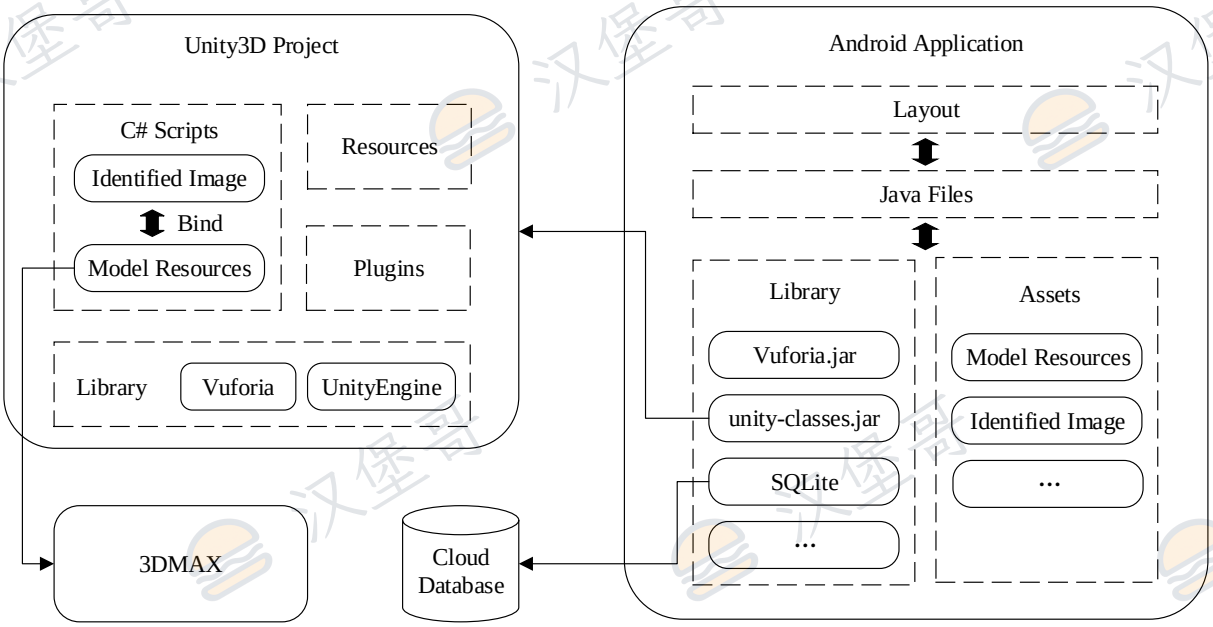

Figure 4.1 System architecture

4.2 Module Design

As shown in Figure 4.2, the firefighting management system contains five function modules, including Account Module, Study Module, Exam Module, Message Module and Equipment Management Module.

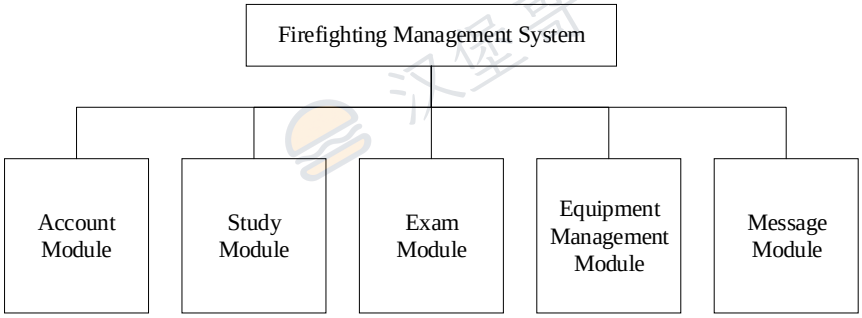

Figure 4.2 Function module diagram of the system

Firefighting management systems must be designed with very clear ideas and a very clear framework. Based on the analysis of the requirements of all systems, this application needs to achieve the following goals when designing. First, employees can use the system to learn and do exams conveniently and quickly. Second, the commands and announcements issued by the leaders can be timely transmitted to the employee's mobile application, ensuring that employees can complete the tasks in a planned manner after receiving instructions. Third, the firefighting equipment administrator can view the three-dimensional information board when scanning the firefighting equipment. The information board has information about the equipment. As the user moves the mobile phone camera, the three-dimensional information can be stably displayed on the object to be identified; Fourth, the system implements various functions of the human-computer interaction system, and allows the user to further manage the device information through some small functions of the auxiliary experience; The interface is simple and the user experience is good. In addition, the system should also follow the following design principles when designing:

(1) The overall display environment is coordinated, setting the atmosphere and improving the user experience;

(2) Pay attention to the orderly division of functional modules, which can meet the requirements of users, and not too complicated and redundant;

(3) Design human-computer interaction function as humanized as possible, so that users can use the system easily, and commit to simple and easy operation methods;

4.2.1 Account Module

Account management is the most basic function of the system. There are two types of users, one is the leader, that is, the person who can issue commands; the other is the ordinary employee, that is, the person who executes commands. New users need to register, and the

registration information is stored in the MySQL database. The user needs to enter a username and password when logging in.

Define the login logic and system response as follows:

(1) If the account name entered by the user does not exist or the password is incorrect, the system fails to log in accordingly.

(2) When the number of login failures reaches three, the system requires a verification code. The user must enter the correct code, username and password to log in. When the failure count reaches five, the account will be locked.

4.2.2 Study Module

The study module is divided into two parts, one is the reading part, and the other is the video course part.

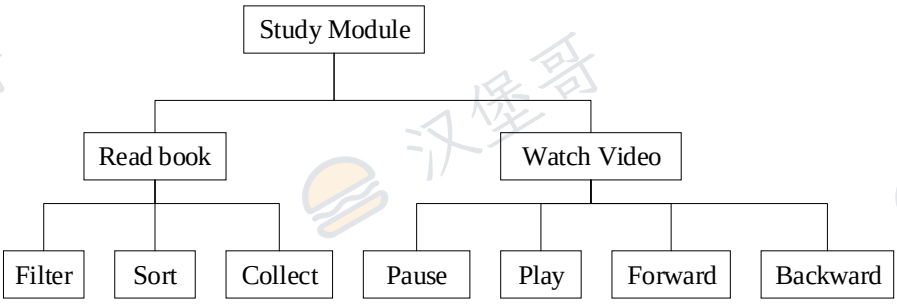

Figure 4.3 Function module diagram of study module

The attributes that each book needs to contain are type, theme, introduction, title. The behavior of the user opening a book will be recorded. In the list of books, you can view the number of clicks of a book. The user can also view the cover photo of the book. The photo resource will be stored in the database as a URL. The system provides the user with the ability to sort and filter. The user can also collect or cancel a book, and the user will send a request to the system after clicking the button of collecting, and the system will change the value of the corresponding field in the database.

As shown in Figure 4.3, the functional design of the video course part is like the reading part. The name, description and screenshot URL of the video will be stored in the database. After the user clicks on a video in the form, the corresponding video stream will be parsed by the video player of the system and presented to the user.

4.2.3 Exam Module

The exam module provides the user with the ability to do exams related to firefighting knowledge. The user can see the title and number of questions for each exam in the list. Each exam holds a set of questions of different types. Question types include single-choice, multiple-choice, and fill-in-the-blank. Each question should have multiple attributes: content, question type, answer, analysis, and so on. When the user is doing the question, he can choose to collect this question or cancel collecting. The user can view the status of current exam, such as the completion degree of the topic. Users can choose to view the question's analysis or submit an exam. After submitting the exam paper, the user's work record will be saved. The function module diagram is in Figure 4.4.

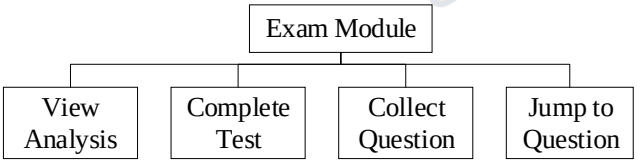

Figure 4.4 Function module diagram of exam module

The relationship between classes in the exam module is a bit complex, so the class diagram is presented in Figure 4.5 in order to describe this part in detail. A user can own multiple reports and every report corresponds to one single exam and every exam contains many questions. Questions exist independently in the database and therefore the relationship between Question and Exam is aggregation.

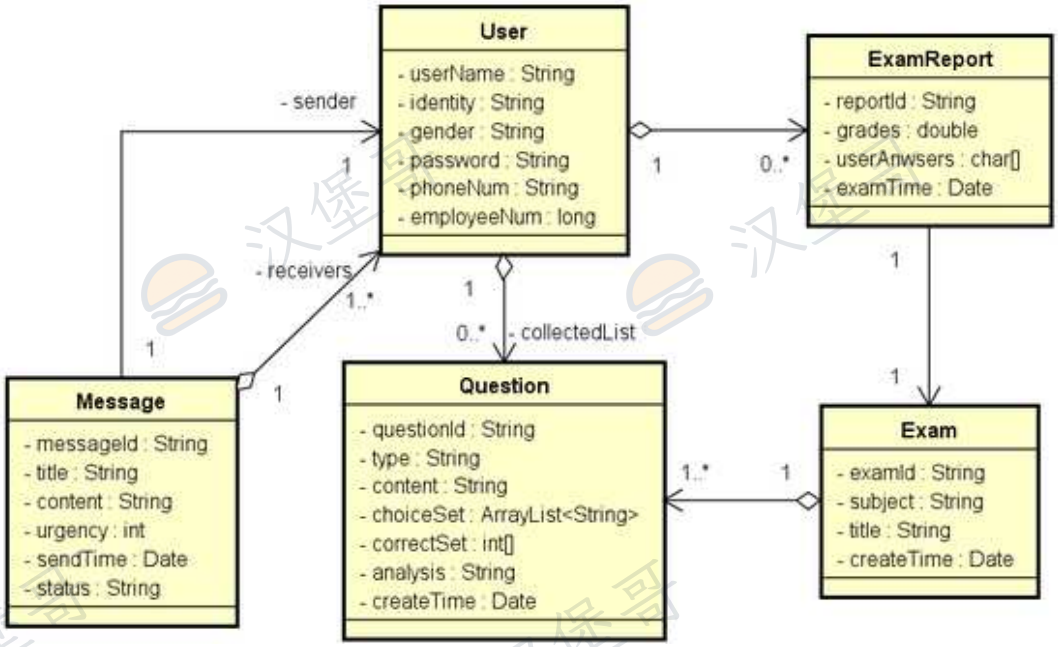

Figure 4.5 Class diagram of message and exam module

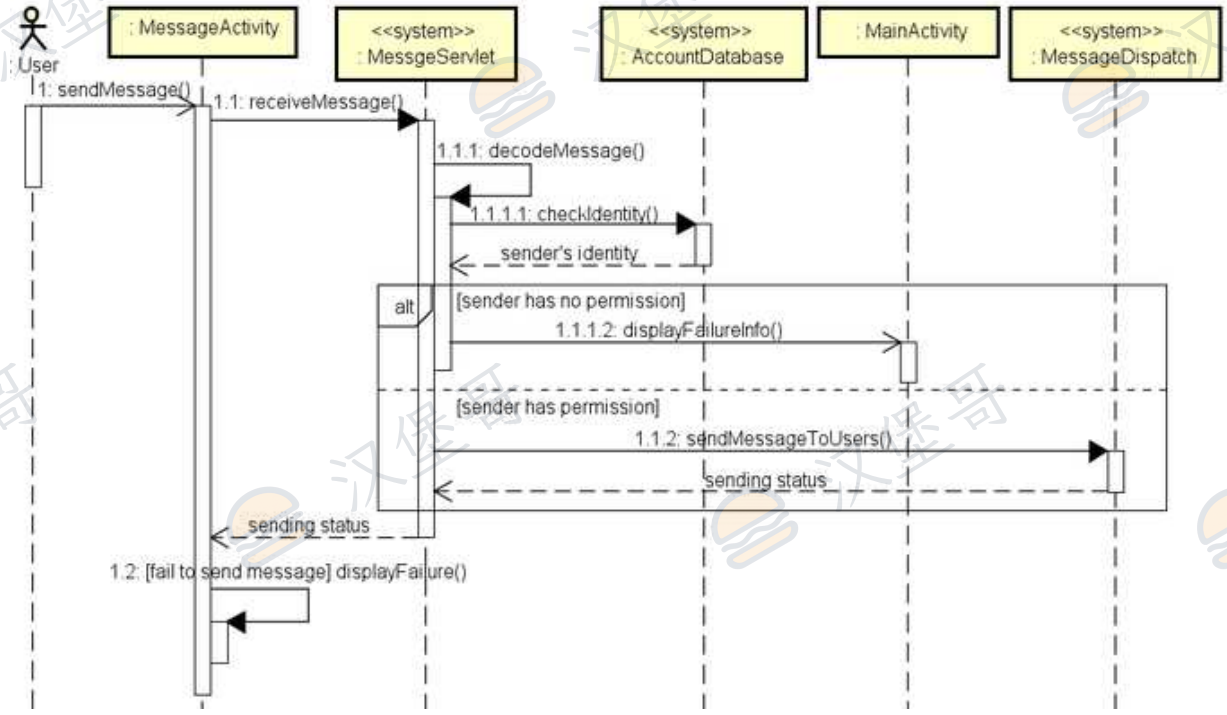

Figure 4.6 Sequence diagram of sending messages

4.2.4 Message Module

In the message module, the set of functions that users can perform is different.

(1) For the leader.

The leader can view a list of employees who are currently executing the task, as well as view the personal information of each employee in the list, such as name, work location, contact information, identity, and so on. Leaders can also view a list of tasks each employee is currently performing, as well as details of each task, such as task content, task location, task release time, and task completion. The leader can choose to give a message to an employee or some employees, and the content of the message includes the subject, content, and so on. The sequence of sending messages is described in Figure 4.6.

(2) For ordinary employees.

Ordinary employees can see announcements issued by different leaders in the announcement area, including the title of the announcement, the name of the publisher, the time of publication, the urgency of the announcement, and the content of the announcement. A more urgent announcement will be displayed on the main interface, reminding each system user in a carousel. The private instructions received by the employee will be presented in the

message bar. The instructions in the message bar have the following attributes: the name of the leader, the time of the instruction, the content of the instruction, the location of the task, and so on. The command is unread when it is not clicked and becomes read after being clicked. After the employee completes a task, he can mark the task as completed. The function module diagram of command is shown in Figure 4.7

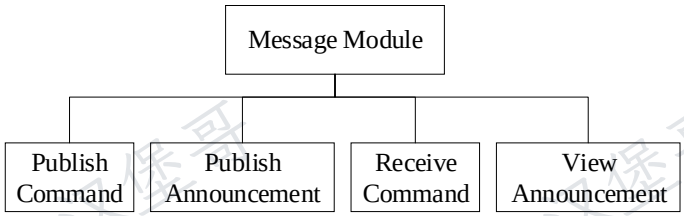

Figure 4.7 Function module diagram of message module

4.2.5 Equipment Management Module

The equipment management module has three main functions, which will be described in Figure 4.8.

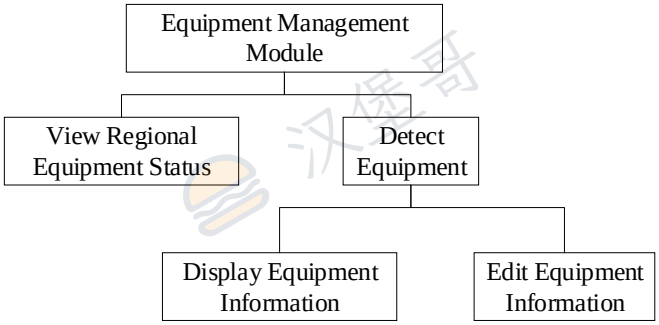

Figure 4.8 Function module diagram of equipment management module

The first function is to view device information in an area. Leaders can search for all devices in a region, and the results retrieved are sorted by category. Different types of equipment will have different types of attributes. For example, a fire extinguisher includes the equipment number, type, equipment area, equipment pressure range, and current pressure value.

The second function is to scan a single device using QR code technology. The technology uses the open source framework, such as ZXingframework, which is an efficient and good framework for Android to write two-dimensional code programs. In this framework, developers can develop their own QR code application. After scanning to a single device, the information of the device can be operated, for example, the current pressure value of the fire extinguisher can be modified.

The third function is to scan the firefighting equipment using AR technology. First, the system will retrieve the video stream obtained by the camera, and then use the AR engine to scan and match the video stream information. After the photo of the firefighting device is recognized, the system will display the information of the device to the firefighting device administrator. The device administrator can edit the device information and view the operational video of the device.

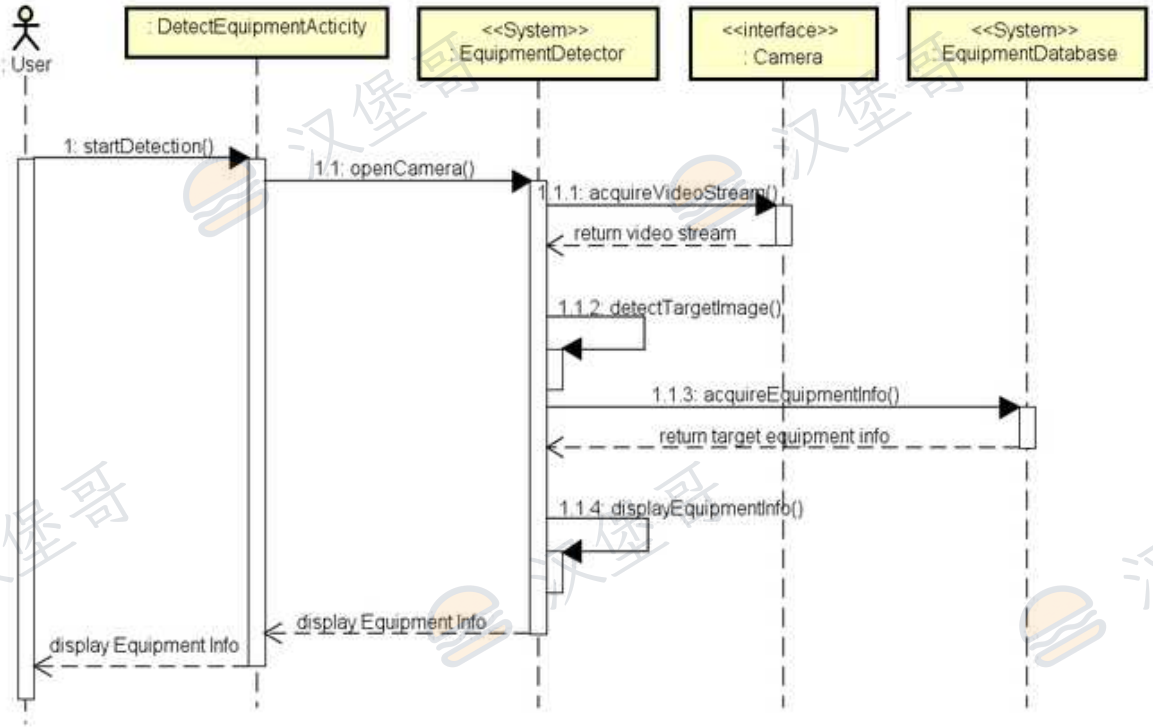

Figure 4.9 Sequence diagram of detecting equipment

The user manages the firefighting device using a mobile phone with a camera, and the device administrator needs to turn on the camera mode to generate augmented reality information board on the mobile phone screen by scanning to identify a specific target. The user can operate information of equipment on the information board, and the device information will be saved in the database. This project is mainly developed using the Unity3D engine. The system mainly identifies key frames of the video stream. The sequence diagram of detecting equipment is described in Figure 4.9.

4.3 UI Design

The user interface design of the firefighting management system based on Android platform generally follows several standards:

(1) Guarantee uniformity. The design of the start interface, main interface and other

functional interfaces of the firefighting management system based on Android platform should be consistent and unified, giving users a clear, clean and consistent feeling.

(2) Provide clear feedback. Interactive interface information such as buttons in the firefighting management system based on Android platform should provide clear feedback after the user performs operations.

(3) Provide clear instructions when the task is completed. To prompt the user after completing a task or operation, for example, after the user has modified the information of a certain setting, the user will get feedback from the system to know whether the modification operation is successful.

(4) Mis-operation prevention. For example, some buttons that are currently not clickable are set to gray, which makes the user clear when the system is executed.

(5) Enable the user to cancel the operation. The application software has the function of undoing and restoring, which avoids the user's mis-operation which may have a serious impact on their experience. In general, users want to control system interactions themselves. In performing tasks, users should be able to stop or quit at any time, rather than helplessly watching the system continue.

4.4 Database Design

The system uses MySQL to store data including account information, equipment information, exam information and so on. As the most widely used relational database management system in the world, MySQL can satisfy the requirement of general individual users and small or medium enterprises. Currently, data entities involved in this system are relatively simple, but with consideration of the application scale and time cost, MySQL is adopted, and many stored procedures are programmed in the database.

The description of account table is described in Table 4.1. All fields should be input during the process of registering. All fields except username can be modified in a user’s profile center. There are several stored procedures, including checking validity of an account, checking request permission, because all these procedures will be executed multiple times and therefore coding stored procedures can speed up the process of accessing account-related data from the database.

Table 4.1 Data structure of account table

Column Name	Data Type	Description
username	varchar(20)	using username to log in the system
password	varchar(20)	password of the user
identity	varchar(20)	identity of the user, including leader or common staff
phoneNum	varchar(11)	the user's phone number
gender	varchar(4)	the user's gender
employeeNum	varchar(20)	the employee number of the user

Study module includes two tables in the database, which are the tables of book and video. The system uses these two tables in a similar way, since both contain a resource URL and some basic information. The resources of books and videos must be stored on the server. In the database, the URL or the file path of resources is needed for convenience of acquiring it in the future. Detailed design of two tables is described in Table 4.2 and Table 4.3.

Table 4.2 Data structure of book table

Column Name	Data Type	Description
bookId	varchar(20)	ID of the book
bookTitile	varchar(20)	title of the book
author	varchar(20)	the list of authors' names
type	varchar(20)	the list of the book's types
url	varchar(50)	the url of the book resources or file address on the server
createdTime	date	date of the book being published

Table 4.3 Data structure of video table

Column Name	Data Type	Description
videoId	varchar(20)	ID of the video
videoName	varchar(20)	name of the video
author	varchar(20)	the list of authors' names
duration	varchar(20)	duration time of the video
url	varchar(50)	the url of the video resources or file address on the server
createdTime	date	date of the video being published

The system allows a leader to send messages to one person or a group of staff. Every request to send a message will be recorded in the database for the need of being retrieved in the future. In the design of message table, the system mainly uses a JSON string to store a list of message receivers. Besides, the column of urgency is designed to show importance degree of the message and if the number is greater than a specific number, it will automatically notify corresponding users. Details are described in Table 4.4.

Table 4.4 Data structure of message table

Column Name	Data Type	Description
messageId	varchar(20)	Id of the message
title	varchar(30)	title of the message with maximum number of 30
content	varchar(500)	content of the message with maximum number of 500
sender	varchar(20)	username of the message sender
receivers	varchar(500)	list of receivers in the form of JSON String
urgency	int(1)	urgency number from 0 to 9
sendTime	date	date of the message being sent
status	varchar(50)	current status of this message, such as Completed and Failed

Table 4.5 Data structure of equipment table

Column Name	Data Type	Description
equipmentName	varchar(20)	name of the equipment
status	varchar(20)	status of the equipment
type	varchar(20)	type of the equipment
equipmentInfo	varchar(500)	detailed information of equipment in form of JSON
imageUrl	varchar(50)	the url of the equipment's image

The relationships in exam module are described in Figure 4.10. A user can hold reports of exams, which record the history of the user doing exams. Every report has a set of question indices and a set of indices of errors. The system uses indices to retrieve a question's details. The database holds a list of exams and every exam has a set of questions. Every question includes a set of choices and a set of correct indices. The question can be

retrieved by the questionId.

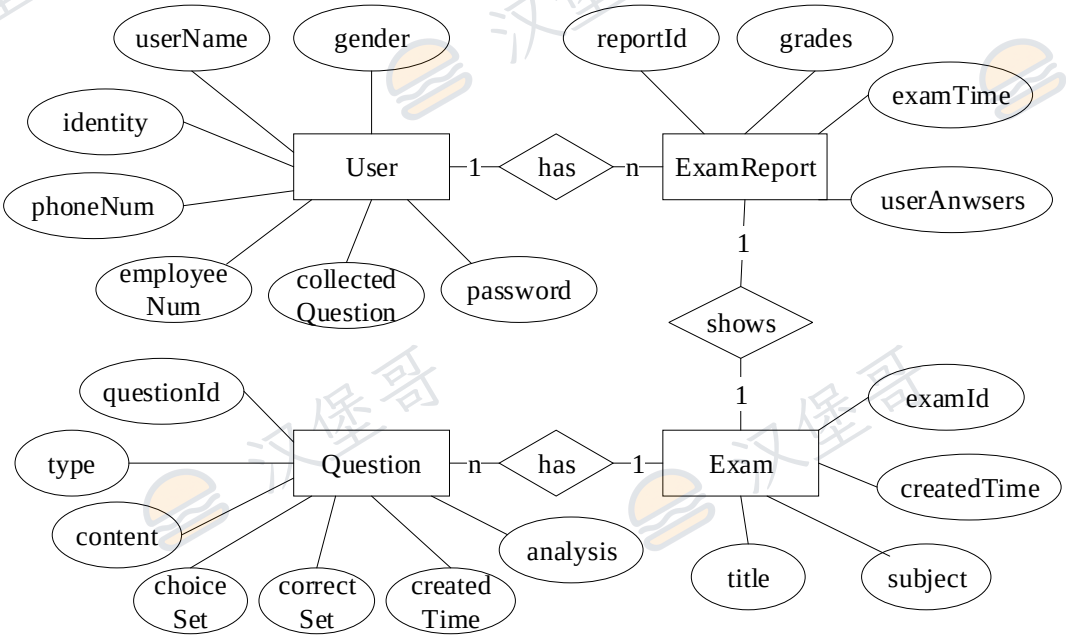

Figure 4.10 E-R diagram of exam module

The data structure of equipment information is described as follows in Table 4.5. Since the system uses Vuforia engine to detect equipment, images are required to recognize equipment and URLs of image resources should be stored on the server.

4.5 Summary

This chapter elaborates on the design content of the firefighting management system based on Android platform. Overall design of the system is presented, as well as specific account module, study module, exam module, equipment management module and message module. Moreover, user interface design of the application and database design of five functional modules are described in the later sections. This chapter uses a variety of design drawings to design the system and build the entire application in detail.

Chapter 5 System Implementation

5.1 Environment Configuration

The development environment of the fire management system based on augmented reality technology is as follows:

Installed Memory(RAM): 16.0GB

Operating system: Windows 10

Java Version: JRE 1.8.0_191-b12

Unity Version: Unity 2018.4.0f1 (64-bit)

Vuforia AR Version: Vuforia Engine 8.1.11

MySQL Version: MySQL Community Server 5.5.19

5.2 Development Tools

Android development tools use Android Studio. This is an Android integrated development tool launched by Google, based on IntelliJ IDEA. Android Studio provides integrated Android development tools for development and debugging. It has the following features: Gradle-based build support; Android-specific refactoring and quick fixes; prompt tools to capture performance, availability, version compatibility, etc.; support for ProGuard and application signatures; template-based wizards to generate common Android application design and components; a powerful layout editor that lets you drag and drop UI controls and preview effects.

Unity3D bundle is embedded in the Android application in order to implement the function of detecting equipment. Unity3D is a multi-platform integrated game development tool developed by Unity Technologies that allows players to easily create interactive content such as 3D video games, architectural visualizations, real-time 3D animations, etc. It is a fully integrated professional game engine. The model exported by 3DMAX is presented by the game engine, such as information boards and indicators in the system.

Use the Vuforia SDK as an augmented reality development tool. This is a set of AR development tools that are not required for mobile device configuration. It can identify and track planar images and 3D objects in real time, which greatly simplifies the development process of AR. It uses computer vision technology to identify and capture planar images or

simple three-dimensional objects (such as boxes) in real time, and then allows developers to place virtual objects through the camera viewfinder and adjust the position of the object on the physical background in front of the lens.

5.3 Implementation of Key Function Modules

5.3.1 Account Module

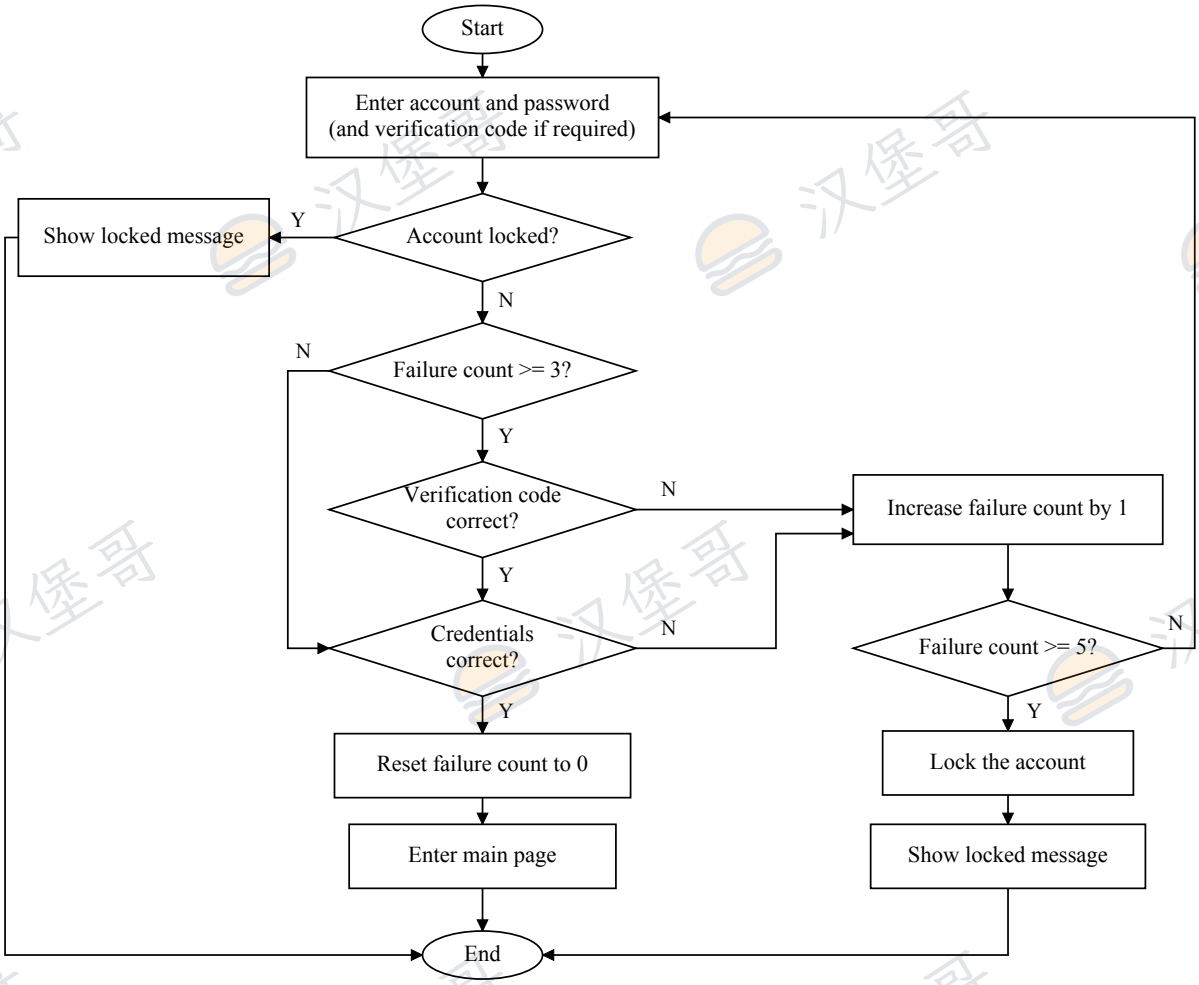

Figure 5.1 Workflow diagram of login

A new user can register with account name, password and identity code. Identity code is used to decide the type of the account, which is leader or staff.

After the user registers his account successfully, he can log in with his account name, password and possibly together with an identification code. The number of login failures for each account will be recorded. When the failure count reaches 3, the system requires a verification code. Every failed authentication attempt increases the count. When it reaches 5, the account is locked; a successful login resets the count to 0. The user can only log in after the account is unlocked by the system administrator. Flowchart of the logic is described

in Figure 5.1. The user interfaces of login and profile management center are displayed in Figure 5.2.

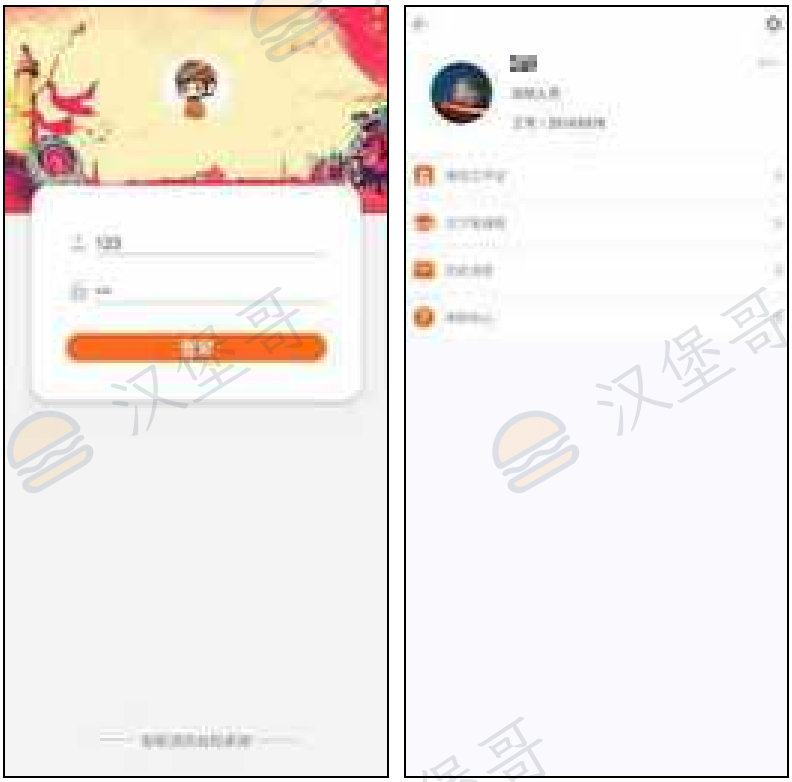

Figure 5.2 Interface of account module

5.3.2 Study Module

Study Module is divided into two parts, reading e-books and videos. The design of e-book list and video course list is similar. The UI type of e-book list is ListView while the UI type of video course list is GridView. Both lists use BaseAdapter to load data into their own UI elements.

As shown in the left picture of Figure 5.3, there are functions of sorting and filtering in the top of book list. At first, the sequence of books displayed in the list is based on the relative value of ID, but if a user clicks different sorting or filtering standards, the BaseAdapter will be notified to change the sequence or hide some books. The interface of reading books is also displayed in Figure 5.3. The system uses an existing framework to implement the PDF viewer. The system uses JCVideoPlayer framework to play a video course, whose features include: being able to completely customize the UI and any functions; perfect detection list sliding; can achieve full-screen playback, small window playback; can enter full screen and exit full screen in various states such as loading, pause, play, etc.; Home button

exits interface to pause playback, return to interface to continue playing; double speed playback. Figure 5.4 displays the interfaces of video list and playing videos.

Figure 5.3 Interface of reading books

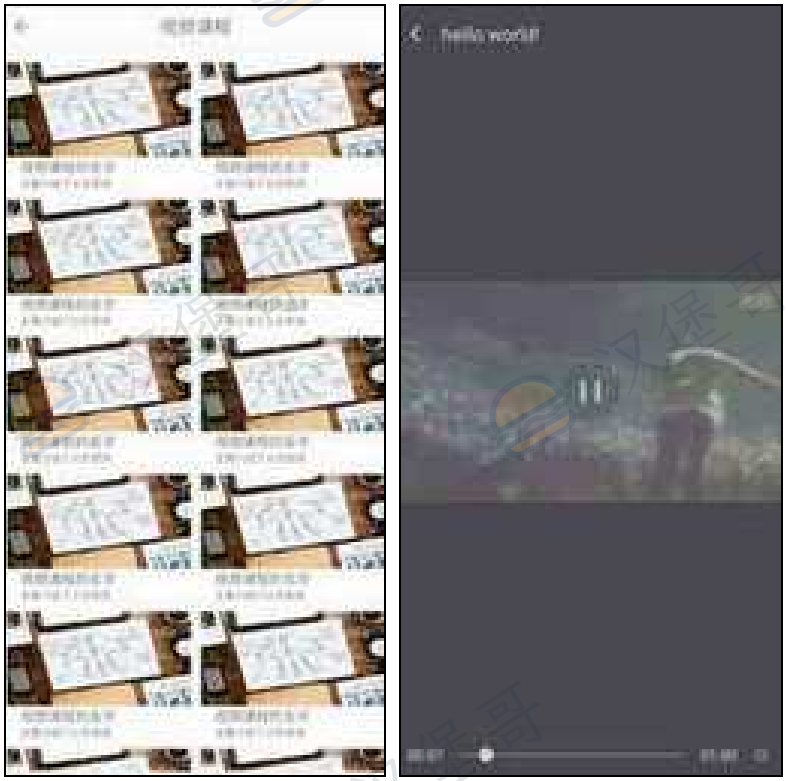

Figure 5.4 Interface of watching video

5.3.3 Exam Module

The system uses ListView to implement the list of exams. After clicking one of the exams in the list, the first question will be displayed to the user. The system uses ViewPager to hold a question's elements such as question content, question choices and analysis. PagerAdapter will be used to load question data into ViewPager. There is a status bar in the bottom of the screen, in which basic information of this exam will be displayed. Figure 5.5 shows the workflow of completing an exam, and Figure 5.6 displays the interface of the exam module.

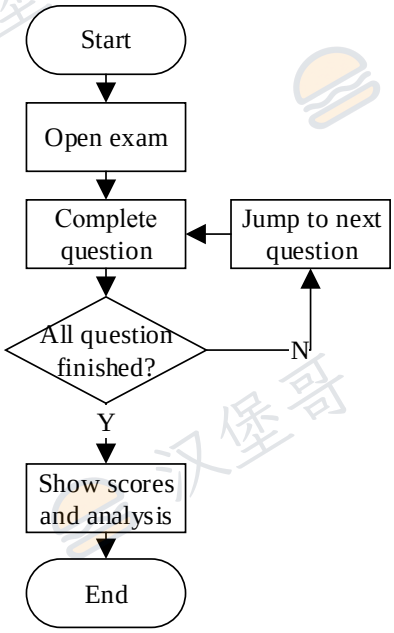

Figure 5.5 Workflow diagram of completing an exam

5.3.4 Message Module

Use ExpandableListView to implement the list of available staff. A leader needs to select a group of staff that the command needs to be sent to. Then the system will forward this command to corresponding users and will record the command in the history. As shown in the Figure 5.7, this is the page where a leader publishes a command. Since the main color tone of this application is orange, most elements such as button and text boxes are designed to be orange.

The system uses ListView to implement the lists of personal messages and announcements. After a leader sends a command message, the staff will receive a system notification which will notify the staff that there is a new message to check.

After a staff member clicks a message, it is marked as read.

Figure 5.6 Interface of exam module

Figure 5.7 Interface of sending messages

The announcement part contains a list of announcements that leaders have published. The urgency of the announcement is divided into two types, urgent and normal. Urgent announcements will be displayed in the carousel of main page as shown in left picture of

Figure 5.8. The system allows 5 urgent announcements to coexist in the main page. If the number of urgent announcements is more than 5, the system will only display 5 items based on the created time of the announcement. Normal announcements will be displayed in announcement center, which is also presented in the Figure 5.8.

Figure 5.8 Interface of viewing messages and announcement

5.3.5 Equipment Management Module

The ID of each device will be displayed at the top, and other information will be displayed below the ID. At the top of the interface is a list of types of firefighting equipment. When you click on one of the firefighting devices, all firefighting equipment of that type will be displayed.

Figure 5.9 shows the function of scanning firefighting equipment using QR code technology. The ZXing QR code scanning framework is used here. After scanning out the device, different properties are displayed according to the device type, and these properties can be edited by users. Taking a dry powder fire extinguisher as an example, after scanning the QR code of the device, the current pressure value, pressure range and working life of the device are displayed, and the user can also report fault of the equipment.

Figure 5.9 Interface of scanning equipment by QR technique

As shown in Figure 5.10, this module uses Unity and Vuforia AR to recognize equipment images and display some information. In the main page of this Unity application, there will be a Fireman. If the target object is recognized, the virtual figure will raise his hand, otherwise he will not raise his hand. If we click on the Fireman with the raised hand, the relevant information will be displayed.

Mobile application development has always had two biggest pain points: one is performance optimization, the other is UI adaptation; For the first point, as the development of technology, mobile phone hardware is constantly upgrading, performance problems have been improved a lot. Therefore, during development, developers usually pay more attention to code efficiency, as well as the rational use of resources. Besides, there is a wide variety of mobile phones in the market, especially Android phones, and correspondingly because of its open source features, mobile phone resolution is numerous, which is impossible to generate separate set of resolutions for each mobile phone. Therefore, the system tries to perform UI elements' automation by placing FireMan's anchor point in the lower right corner of the entire UI. Regardless of how the model changes, the button will only change in proportion to the distance in the lower right corner of the screen, and its size will not change to prevent stretch

deformation.

Figure 5.10 Interface of detecting equipment through AR technique

Another problem is that Vuforia doesn't autofocus. To solve this problem, this system registers Vuforia's startup and pause callback methods, and resets Vuforia's camera focus mode in these two callbacks. Besides, this application uses the most common resolution, 800*1200, but height and width of elements are dynamically adjusted in running time.

Regarding the storage of data, the system chose PlayerPrefs. This component is a class provided by Unity3D for saving and reading persistent local data. The component works very simply, saving the data in a file in the form of a key-value pair, and the program can take the last saved value based on this name.

Staff with permissions can view information of equipment in a specific region. Implemented by ViewPager, this part contains a list of equipment’s types. After clicking a specific type bar, all corresponding items will be displayed with their status information including location, equipment ID, equipment type and so on. Staff can also add tasks to their schedule. After entering schedule center, a calendar and a list of tasks will be displayed. The system can notify users automatically based on their task table. All of these functions will be displayed in Figure 5.11.

Figure 5.11 Interface of viewing regional information and scheduling tasks

5.4 Summary

This chapter introduces the system implementation process of the software from the account module, study and exam module, equipment management module and message module of the firefighting management system based on Android platform. The in-depth development process and key technology implementation processes were carried out for the main functions included in several major modules.

Chapter 6 System Testing

Testing is an indispensable part of development. It runs through the entire firefighting management system. Its main purpose is to find and eliminate as many hidden errors as possible in the current software system at every stage of software development. Whenever a feature is developed, it is necessary to perform certain tests such as stability tests and compatibility tests. The following is a comprehensive and detailed description of the content of system testing in the development of the project.

6.1 Testing Purpose

The first purpose of testing is to confirm the quality, which contains what users expect and whether it is done in the right way. Errors in the virtual display system can be discovered and adjusted in time. The second is to provide information, such as feedback to developers, and information for risk assessment. Testing is not only about testing the product itself, but also about the development process. The third purpose of the test is to ensure that the entire development process is of high quality. When testing, of course, it must be carried out step by step according to the requirements analysis, so the demand analysis should be as detailed as possible. The entire system is tested to verify that it meets the user's needs.

6.2 Basic Test Method

6.2.1 Black Box Testing

Black box testing, also known as functional testing, looks at the external features of the program, regardless of the internal logical structure of the program. The tester views the program under test as a black box and does not care about the internal structure of the program. The black box test is performed at the program interface. It only checks whether the program function can be used normally according to the specification. The program can properly receive the input data to generate correct output information and maintain the integrity of the external information.

6.2.2 White Box Testing

White box test is a type of test case design method. It follows the structure test procedure inside the program to verify that each path in the program can work correctly according to the

predetermined requirements regardless of its function. Test methods for white-box testing include code checking, static structural analysis, static quality metrics, logical overlays, basic path testing, domain testing, symbol testing, path coverage, and program variation.

6.3 Test cases

The account module includes registration and login functions. The test cases for this module are described below.

Table 6.1 Test case of account module

No.	Operation Name	Testing Method	Expected results	Test result
1	Register	enter correct account, password, identity code	show register success	Same as expected
2	Enter wrong password three times	enter correct account but wrong password	require user to input identification code	Same as expected
3	Login	enter correct account and password	enter main page	Same as expected

As shown in Table 6.2, the study module contains reading books and watching video. The pre-condition of this test case is that the user has logged in the system.

Table 6.2 Test case of study module

No.	Operation Name	Testing Method	Expected results	Test result
1	Enter Book Center	Click the button of Book Center in the main page	show a list of books	Same as expected
2	Filter different type of books	click all tags in the filter bar	book list contains only books of specific type	Same as expected
3	Sort books	choose to sort books by all standards	books in the list are sorted in specific order	Same as expected
4	Enter Video Center	Click the button of Video Center in the main page	show a list of videos	Same as expected
5	Play a video	choose one video and click buttons of pause, play, forward	the video is played normally, and other functions work well	Same as expected

The test cases of the exam module are shown in Table 6.3.

Table 6.3 Test case of exam module

No.	Operation	Testing Method	Expected results	Test result
1	Enter Exam Center	click the button of Exam Center in the main page	show a list of exams	Same as expected
2	Complete question	select choices and click the button of Yes	jump to next question automatically	Same as expected
3	Submit the exam	click the button of submit	show scores, analysis	Same as expected

The equipment management module contains several functions including detecting equipment through QR technique and AR technique. Related test cases are as follows.

Table 6.4 Test case of equipment management module

No.	Operation	Testing Method	Expected results	Test result
1	QR code scanning	open camera, perform function of QR scanning toward target item	Recognize target, display related information	Same as expected
2	AR scanning	open camera, click the button of AR scanning	Recognize target, display related information	Same as expected
3	Edit information	click the button of Edit, input some words and click the button of Save	close the application and reopen it, then same words are displayed.	Same as expected

As shown in Table 6.5, in the test case of message module, there are two operations including publishing commands and viewing announcement.

Table 6.5 Test case of message module

No.	Operation	Testing Method	Expected results	Test result
1	publish command	select staff and click Sent Command	Corresponding staff receive message	Same as expected
2	view announcement	click Announcement Center	jump to the page of Announcement Center, show all announcements	Same as expected

6.4 Summary

This chapter mainly introduces the testing process of firefighting management system based on augmented reality technology. The correctness and stability of several major functional modules and user interface interaction modules in the development process of this system and the technical feasibility of key technologies were tested in detail from three aspects: test purpose, test method and test case. The test results of the system in each test session were reported and analyzed.

Chapter 7 Conclusion

7.1 Summary of the Work

This paper studies the design and implementation stages of firefighting management system based on Android platform and elaborates on every aspect of the system development stages from requirements analysis to system design, system implementation and system testing. The workflow and development process of the study module, exam module, message module, equipment management module and all sub-modules contained in the firefighting management system based on the Android platform are introduced. The graduation design content discussed in this paper proposes a set of feasible solutions for using Android platform and AR technology to assist staff to manage firefighting equipment, and carries out detailed and specific testing process, and finally completes the development of the firefighting management system based on the Android platform, in the whole system. The main work of the development process is as follows:

(1) Android Programming

This system has designed many practical functions for firefighters, including learning functions, exam functions, command transceiver functions, and device management functions. The design of the functional interface considers the aesthetics of the interface and the ease of operation of the user. The Android main program optimizes the thread to ensure that the program can be smoothly carried out on the user's mobile phone.

(2) Embed the Unity program into Android

Package the Unity3D project as a JAR file, embed it into the Android application, add dependencies and modify the AndroidManifest.xml configuration file to implement the interactive communication between Unity3D and Android.

The application scenario of the system is a mobile phone, which detects the current environment through a camera, identifies different types of firefighting equipment, and displays corresponding information according to actual conditions, and assists the quality inspection personnel to manage the firefighting equipment. After obtaining enough information, integrate and analyze the data, form a report, score the current management of fire-fighting equipment, and implement other auxiliary functions.

The above information is visually displayed on the user's mobile phone screen through augmented reality technology, taking into consideration the aesthetics and user friendliness.

7.2 Future Work

The next phase of the work plan is divided into three parts:

First, improve the UI interface, mainly to improve the integrity and aesthetics of the program. When connecting with the background data, the front-end interface may be fine-tuned. At the same time, the UI interface can consider adding new features, such as real-time communication between users.

Secondly, in terms of connecting the background data, it is necessary to add corresponding real data of firefighting equipment to the database, as well as data of exams, study resources and so on. Besides, the account information could be bound with users' social account, which will help users to register more conveniently.

Finally, the current AR auxiliary firefighting subsystem is relatively simple, only to identify some objects, and to present static information bound to such objects, and the interface design is relatively simple. The next stage is to implement more complex business logic, so that users can achieve some simple interactions in the AR subsystem and improve the aesthetics of the interface.

References

[1] Eleonora Bottani,Giuseppe Vignali. Augmented reality technology in the manufacturing industry: A review of the last decade[J]. IISE Transactions,2019,51(3).

[2] Sabrina Romina Sorko,Magdalena Brunnhofer. Potentials of Augmented Reality in Training[J]. Procedia Manufacturing,2019,31.

[3] Nazri N.I.A.M, Rambli D.R.A: Current limitations and opportunities in mobile augmented reality applications. ICCOINS.2014.6868425

[4] Marques Fábio, Azevedo Paulo, Cunha João Paulo Silva, Cunha Manuel Bernardo, Brás Susana, Fernandes José Maria: Firefighter team breathing management system using android. ISWC 2013, p 133-134, 2013

[5] Eltom R.H., Hamood E.A., Mohammed A.A., Osman A.A.: Early Warning Firefighting System Using Internet of Things. 2018 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE), p 7 pp., 2018

[6] Wultz T. J., et al., SCBA, Essentials on firefighting and Emergency Response, 2nd edn, 2004

[7] Márcio M. F. Colunas, et al., Droid Jacket: a mobile monitoring system for a team, In Proc. of: 7th International Wireless Communications and Mobile Computing Conference, IWCMC 2011

[8] Gary C. White, Basic Clinical Lab competencies for respiratory care: An integrated Approach, 5th edn.,Connecticut: Delmar Cengage Learning, 2009

[9] Xing-Guo Sun, et al., Ventilatory Efficiency during Exercise in Healthy Subjects, American Journal of respiratory and Critical Care Medicine, vol. 166, 2002

[10] Azuma, Ronald T: A Survey of Augmented Reality[J]. Teleoperators and Virtual Environments, 1997,6(4):355-385

[11] Blum J.R, Greencorn D.G, Cooperstock J.R: Smartphone sensor reliability for augmented reality applications. In: Zheng, K., Li, M., Jiang, H. (eds.) MobiQuitous 2012.LNICST, vol. 120, pp. 127–138. Springer, Heidelberg (2013).

[12] Huamaní J.A: Augmented reality: a digital resource between the real and the virtual world. Hamut'Ay 2, 50–57 (2016)

[13] Kokaew U, Luangrungruang T: Applying universal design for learning in augmented reality education guidance for hearing impaired student. ICAICTA, 250-5, 2018

[14] Dabran I, Avny T, Singher E, Ben H: Augmented reality speech recognition for the hearing impaired. COMCAS, p 4 pp., 2017

[15] Rashed E.A, El-Seoud S.A: An interactive augmented reality imaging system for minimally invasive orthopedic surgery. ICKEA, 138-41, 2017

[16] Seongpung Lee, Hyunki Lee, Hyunseok Choi, Sangseo Jeon, Jaesung Hong: Effective calibration of an endoscope to an optical tracking system for medical augmented reality. Cogent Engineering, v 4, n 1, 1359955 (11 pp.), 2017

[17] Westerfield G, Mitrovic A, Billinghurst M: Intelligent Augmented Reality Training for Assembly Tasks. AIED: LNCS 7926, 542-51, 2013

[18] Edison A. Chicaiza, Edgar I. De la Cruz, Edgar I. De la Cruz: Augmented Reality System for Training and Assistance in the Management of Industrial Equipment and Instruments. 13th International Symposium, ISVC 2018, Proceedings

[19] Chenlin Zhang et al: Application of Augmented Reality Technology in Workshop Production Management. 2018 IOP Conf. Ser.: Mater. Sci. Eng. 466 012048

[20] Dacko S G: Enabling smart retail settings via mobile augmented reality shopping apps. Technological Forecasting and Social Change, v 124, 243-56, Nov. 2017

Appendix

本文研究了基于 Android 平台的消防管理系统的设计与实现过程，阐述了系统开发过程中从需求分析到系统设计、系统实现到系统测试的各个方面。介绍了基于 Android 平台的消防管理系统的账户模块、学习模块、测试模块、信息模块、设备管理模块及各子模块的工作流程和开发过程。毕业设计内容中讨论并提出了一套可行的使用基于增强现实技术的协助消防部门工作人员管理设备的解决方案，并对该解决方案进行了系统架构设计和模块设计，完成了消防管理系统的开发和测试。

账户模块主要包括注册和登录两个步骤。根据对消防部门工作人员的角色分析，发现消防部门人员主要分为两类：一类是领导类，即发布命令的群体；另一类是普通职员类，即执行命令的群体。本系统要求用户在注册时提供实际工号，并需要通过短信验证。

学习模块主要分为查看消防类书籍和查看相关视频课程。书籍和视频在数据库中的表单结构类似，都存有相关文件的链接。对于书籍来说，数据库中会存有类型为 pdf 的文件类型的链接。该文件通过系统内置的 pdf 阅读器进行解析，呈现给系统用户。视频资源的解析过程与书籍资源的解析过程类似。测试模块主要为消防部门人员提供进行测验的功能模块。用户可以在考试列表中看到每份试卷的名称和题目数量。一份试卷由若干题目组成。题目的类型分为单项选择题、多项选择题和填空题。每个题目应该有多个属性，即内容、问题类型、答案、分析等等。用户可以选择查看题目解析或提交考试。提交试卷后，将保存用户的测试记录。信息模块主要为领导和普通职员提供收发消息的功能模块。领导者可以看到当前能够执行任务的员工列表，还可以查看列表中每个员工的个人信息和工作列表。领导者可以选择给一个或几个员工下命令，命令的内容包括主题、内容等等。普通员工可以在公告区看到不同领导发布的公告，还可以接收私人指令。设备管理模块包含三个主要功能：查看区域设备信息，使用二维码技术识别特定设备，使用增强现实技术识别特定设备。二维码识别部分使用 ZXing二维码识别框架，增强现实识别部分使用 Vuforia 引擎，特定设备被识别后会显示相关信息，用户可以对设备信息进行修改。

整个系统架构方面，将 Unity3D 项目打包成 JAR 文件，嵌入到 Android 应用程序中，添加依赖项，修改 AndroidManifest.xml 配置文件，实现Unity3D 与Android 的交互通信。本系统的应用场景为手机，手机通过摄像头检测当前环境，识别不同类型的消防设备，并根据实际情况显示相应的信息，协助质检人员对消防设备进行管理。在获得足

够的信息后，对数据进行整合分析，形成报告，对消防设备的当前管理进行评分，并实现其他辅助功能。

Acknowledgment

The four-year period of the undergraduate course is about to pass, and I hope that this article will be a perfect ending for university life. In this paper, the author has practiced and applied the knowledge he has learned in Northeastern University for four years. The author tries to use the graduation design as a test to see if he meets the training objectives of the Northeastern University Software College for students.

The production of the graduation project has gone through several processes, such as demand analysis, system design, system implementation, system testing, etc., each process is full of challenges and difficulties. It is in the united and friendly learning atmosphere of the school of software of Northeastern University that the complicated and heavy work of graduation design becomes no longer boring and tedious. In the process of completion of graduation design, I have gained more harvest and valuable development experience. The author would like to thank Professor Yu for his tireless guidance during the graduation project. He not only helps to solve many technical difficulties, but also provides much useful advice in the process of developing a firefighting management system based on Android platform.

Finally, "Self-improvement, unity of knowledge and practice" is the most important and vivid lesson Northeastern University gave me. Although the time in college will soon pass, Northeastern University left me good memories, which are unforgettable for me; The wealth of knowledge Northeastern University has given me will benefit me for the rest of my life. The spiritual strength endowed by Northeastern University will also point out the way for my future life. I would like to thank the good teachers and helpful friends of Northeastern University for their help during my four years of study and life.

东北大学

本科生毕业设计（论文）答辩小组

表 决 票

学生姓名_____ [redacted] _____ 学院（部） 软件学院

论文题目 基于安卓平台的消防管理系统的设计与实现

一、成绩评定（五级分制）

优秀 （90-100 分）	良好 （80-89 分）	中等 （70-79 分）	合格 （60～69 分）	不合格 （59 分及以下）

二、答辩评定

同意通过答辩	不同意通过答辩	弃权

答辩委员签名：_____

说明：1、学生姓名、学院、论文题目栏，由答辩委员会秘书填写；

2、成绩评定栏目请在合适的空格内**填上具体的分数**；

3、答辩评定栏请在确认项目空格内画“〇”；

4、没按规定投票的视作弃权。

东北大学

本科生毕业设计（论文）答辩小组

表 决 票

学生姓名_____[redacted]_____ 学院（部） 软件学院

论文题目 Design and Implementation of Firefighting Management System Based on Android Platform

一、成绩评定（五级分制）

优秀 （90-100 分）	良好 （80-89 分）	中等 （70-79 分）	合格 （60～69 分）	不合格 （59 分及以下）

二、答辩评定

同意通过答辩	不同意通过答辩	弃权

答辩委员签名：_____

说明：1、学生姓名、学院、论文题目栏，由答辩委员会秘书填写；
2、成绩评定栏目请在合适的空格内**填上具体的分数**；
3、答辩评定栏请在确认项目空格内画“〇”；
4、没按规定投票的视作弃权。

学 生 工 作 记 录

设计工作与图纸绘制
完成日期： 5 月 15 日（第 11 周）
（实验研究、论文工作）

设计说明书
完成日期： 5 月 28 日（第 13 周）
（论 文）

专题部分完成日期： 月 日（第 周）

毕业设计
送交评阅人日期： 5 月 30 日（第 13 周）
（论 文）

指导教师签字： 年 月 日

上机（实验）地点：[redacted]

累计上机时数：692 小时

答辩日期： 月 日（第 周）

答辩委员会主任签字： 年 月 日