ANNOTATION

SYMBOLS AND ABBREVIATIONS LIST

IoT — Internet of Things;

OOD — Object Oriented Design;

IoC — Inversion of Control.

CONTENTS

INTRODUCTION

The developed mobile application is a part of a system that belongs to a class of home automation systems - systems capable of performing actions and solving certain everyday tasks without human intervention. Home automation is seen as a separate case of the Internet of things, it includes home-based Internet access, while Internet of things include any device connected via the Internet at all.

The most common examples of automatic actions in a "smart home" are automatic switching on and off, automatic adjustment of the heating system or air conditioner and automatic notification of intrusion, flash or water leakage. The system will play the role of "smart intercom", allowing the owner of the home to interact with visitors through a mobile phone program anywhere in the world where there is the Internet.

The purpose of this work is to write a mobile application for the automatic control system "Smart Intercom" using modern technology stack.

The mobile application is an important component of the system, since it represents the interface of the end-user interaction with the system. The quality of its implementation depends on the convenience of use, and thus the demand for the system as a whole.

Due to the diversity of mobile platforms on the modern market, the mobile application will be developed with the ability to expand on different platforms. This project will use Xamarin technology. This technology allows developers to create a common code base with the use of C # and .NET right away for all three platforms - Android, iOS, Windows Mobile, because Xamarin is a technology for cross-platform mobile application development, and allows to qualitatively and, most importantly, optimally to meet market needs in mobile applications.

1 ANALYTICAL REVIEW OF THE LITERATURE AND OF EXISTING METHODS OF SOLVING THE PROBLEM

* 1. General description of the problem and existing approaches to solve it

Nowadays it is very hard to imagine the life of a modern people without smartphones. They makes our lives more interesting, easier and performative. The functionality of modern smartphones is growing very intensively and they are covering more and more functional arias of their usage. The most interesting direction of their development is home automation systems where the smartphone takes place of the control panel, the face of such system at all.

The developed system relates to a class of home automation systems - systems capable to perform actions and solve certain everyday tasks without human involvement. Homemade automation is seen as a separate case of the Internet of things, it includes home devices and applications accessible via the internet.

The most widely spread kind of home automation systems are “Smart home" systems.

A smart home is an indispensable attribute of any modern home, in which there are so many different engineering systems: lighting, power electric, heating, ventilation, air conditioning, home theater technology, security and fire alarm.

A smart house is an intellectual management system that unites all equipment in a single complex that solves various tasks in the sphere of security, life support, entertainment and communication. Any smart house system consists of sensors, through which information is received, and actuators.

A smart house is a system of intelligent automation for controlling the engineering systems of a modern building.

The Smart home system provides a mechanism for centralized control and intellectual management in residential, office or public premises. With the installation of such a system at home or at work, each user gets next opportunities:

1. To set the parameters of his own individual environment (light, air temperature, sound, etc.);
2. To manage the necessary system (lighting, climate, video surveillance, etc.)
3. To get access to the information about the state of all life support systems at home (whether inside or remote);

The general scheme of the control system is as follows:

1. Central processing unit / main control unit
2. Sensors (temperature, light, smoke, traffic, etc.)
3. Control devices (dimmers, relays, IR emitters, etc.)
4. Control interfaces (pushbuttons, IR and radio control panels, touch panels, web / WAP interface)
5. Own management network that combines the above elements
6. Managed devices (lamps, air conditioners, home theater components, etc.)
7. Auxiliary networks (Ethernet, telephone network, distribution of audio and video signals)
8. Project Software

The Smart Home system provides a real-time management with using a mobile device or a PC located in a local area network or that has an Internet access.

The main goal of this work is to analyze and design a mobile application and all support services for a system "Smart Intercom" using modern technologies and equipment. So, according to the requirements, the developed system must be user friendly, secure, must consist of a cloud server API mobile application will work with, the support services that will provide corresponding communication channels between server and mobile application and mobile application itself as a presenter.

The developed application should be designed independently from a mobile platform so cross platform technologies should be used.

* 1. Technical task analysis.

The main document, which contains the main initial data and requirements for automatic control system “Smart intercom” is specification, when system is designing. Careful analysis of existing specification for the presence of full required information for successful implementation of the system gives required specification items.

Specification contains the main information about automatic control system modes, about requirements for reliability and safety, about devices composition, about characteristics, disturbances, electrical parameters etc.

Automatic control system “Smart intercom” is intended to help householders to interact with their visitors being in any point on our planet where the Internet connection is available

This work should be implemented in 3 stages:

1. Specification implementation
2. Detailed design
3. Automatic controls system design

The main aim of this automatic control system is to provide the user friendly interface for communication with the visitors of the house using a mobile phone application. Householder should be able to see who has come to him, to communicate with the visitor, to see him and to lock or unlock the door using different ways depending on the situation.

The declared functionality of the automatic control system “Smart intercom” is next:

1. user should be able to unlock the door using the smartphone application from any point on the planet where internet connection is available;
2. user should be able to unlock the door being in the building using the smartphone application even without internet connection using wireless technologies;
3. user should be able to unlock the door without smartphone using his credentials;
4. user should be able to provide the special authorization token (QR-Code) to guest, which can be used to unlock the door without interaction with householder. Token should have the expiration date and time which could be set up by householder when the token is creating;
5. door should be automatically closed after the timeout period has ended;
6. householder should be able to check the current state of the door (locked/unlocked) and change it manually on demand using mobile application;
7. all events happened in the system should be logged by centralized logging system and stored in remote storage system;
   1. Analysis of existing software capabilities

Home automation and the Internet of Things is new modern direction in software development that grows very fast nowadays. Many of modern companies specialized on the electronics and internet technologies has their own projects in this area. It is a good practice to analyze existing solution before the new system design and implementation.

Among the home automation systems and “Smart home” systems there are a lot of examples of competitive solutions. Let’s overview the most relevant of them.

Xiaomi MI Smart Home Suite (Fig. 1.1.)

This system represents a decentralized home automations system. It consists of a main device called Gateway which provides the entry point of the system and allows user to interact with other components of the system using special mobile application via internet or using wireless technologies locally.



Figure 1.1 — Xiaomi MI Smart Home Suite

The functionality of this system is determined by the system composition. Each device or sensor can be set up to the system (through the Gateway device) independently using the wireless technologies depending on the user requirements for the system itself. Some of devices are included to the initial set of the system, and some could be bought separately. The main advantage of this system is very simple set up process. Everything that is needed to configure the system is to install the mobile application, enable power on the Gateway device, pair required sensor and devices with Gateway and system is ready for usage.

This product provides a special mobile application for system control called Xiaomi mi home (Fig.1.2.).

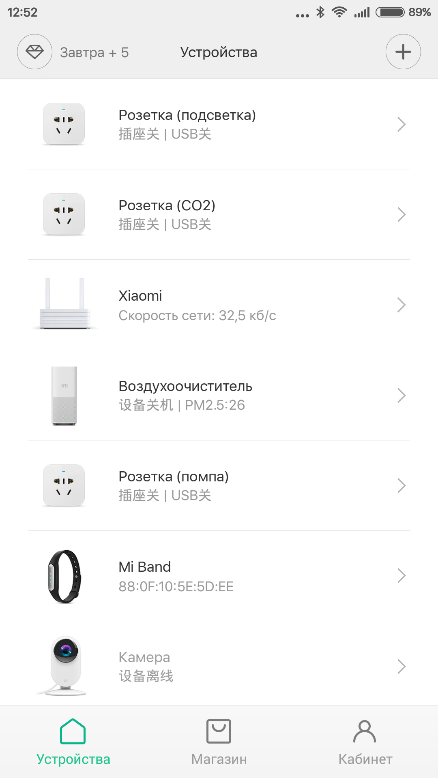


Figure 1.2 — Xiaomi Mi Home application

This application allows use to control every device connected to the system from a single place – from a smartphone.

But despite all the advantages of this system it does not provide the functionality that developed system provides. Also applications on different platforms have some differences in functionality and this is not so user friendly.

Ring Wi-Fi Enabled Video Doorbell (Fig. 1.2)

This system of home automation provides two-way audio communication, motion sensing, cloud recording etc. This system also provides very convenient mobile application (Fig. 1.4) to end users.

But as the previous system it also does not cover the main functionality the developed system provides.



Figure 1.3 — Ring Wi-Fi Enabled Video Doorbell

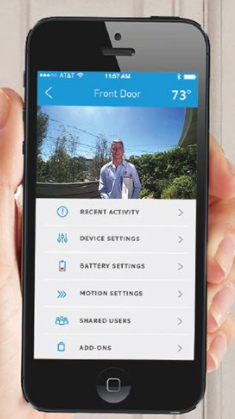


Figure 1.4 — Ring Wi-Fi Enabled Video Doorbell mobile application

Saful Smart Home Apartment Wireless Wifi Video Door Phone Intercom System (Fig. 1.3.)

This is the most similar home automation system to the developed system. It covers the most of functionality required from the developed system – it supports Wifi, Android & IOS applications, real time video and audio chat, movement detecting, unlocking. But it is not enough secure. First of all this system is centralized and this fact reduces system tolerance, stability and availability.

The provide mobile application (Fig. 1.6) is designed very well and covered the main required functionality but the stability of work on different platforms was not enough to satisfy the requirements for developed system, first of all because of previously described issues.



Figure 1.5 — Saful Smart Home Apartment Wireless Wifi Video Door Phone Intercom System

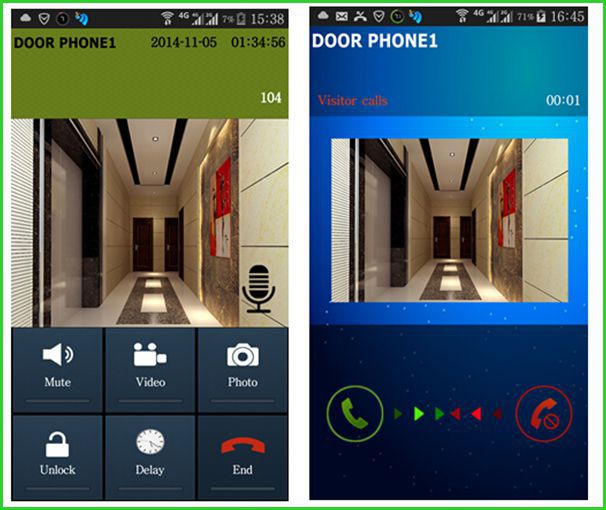


Figure 1.6 — Saful Smart Home Apartment Wireless Wifi Video Door Phone Intercom System mobile application

After exiting solutions overview and analysis next the most reasonable question is what software infrastructure and software technology stack is better to use to implement home automation system. From the requirements described in the technical task it is clear that cloud technologies should be used. After the analysis of the cloud services market Microsoft Azure Cloud was chosen because it is one of the most convenient and functional cloud service. Also Microsoft Azure provides the set of services especially for the Internet of Things and Home Automation systems development called Azure IoT Suite. This suite provides special services that allows home automation system developers perform next actions:

1. establish bi-directional communication with billions of IoT devices;
2. authenticate per device for security-enhanced IoT solutions;
3. register devices at scale with IoT Hub Device Provisioning Service;
4. manage your IoT devices at scale with device management;
5. extend the power of the cloud to your edge device.

If the Microsoft Azure Cloud is chosen as the software infrastructure and environment provider it is better to use full Microsoft software development stack.

For mobile development Microsoft provide the cross platform development framework called Xamarin. The main advantage of this framework is that developers can use the common code base for developing the applications for deferent platforms. This fact increases performance of mobile application development and reduces time for testing developed applications on different platforms.

* 1. Designing tasks set up

Automatic control system “Smart intercom” is a complex distributed software system that requires very wide spectrum of details that should be taken into account during its design.

After detailed analysis of the technical task, existing solutions and software capabilities there is an appropriate time to formulate the main goals and tasks that should be done in this paper.

When the requirements are defined, the main thing should be implemented is the functional scheme of the system. This scheme should reflect the main modules of the system and relations (signals) between them. All requirements should be considered while the scheme is creating.

After the functional scheme is created the software capabilities should be chosen, specific applications designed taking into account their features, protocols of communication between them should be defined.

When the software is designed, there is turn to choose the hardware for the automatic control system implementation. It is required to do this step before the software implementation to take into account all capabilities, features and issues introduced by selected devices.

To make developed automatic control system resistant to different kinds of disturbances, stable, high available cloud technologies should be used as the environment for server software deployment. It will reduce the costs for supporting such system and will increase the convenience of automatic control system usage at all.

The final stage of this work presents the implementation of the prototype of the automatic control system itself. It should be the special stand that represents the main functionality of the system. This prototype should demonstrate that the automatic control system works as expected and provides all declared functionality.

When the automatic control system prototype is ready there is time to test it. Different kinds of test suites should be developed to cover the system with tests. This work is intended to verify the correctness of the automatic control system work and to evaluate the system stability and stability stocks.

Main tasks which should be done in this paper:

1. requirements analysis should be performed;
2. existing solutions and capabilities should be analyzed;
3. functional scheme of the automatic control system should be created according to the requirements;
4. software infrastructure should be chosen for the automatic control system software implementation;
5. hardware infrastructure should be chosen for the automatic control system software implementation;
6. the automatic control system software should be designed and implemented;
7. the automatic control system prototype should be created and tested
8. the automatic control system prototype correctness of work should be verified and corresponding conclusions should be made.
9. PROBLEM INVESTIGATION AND MATHEMATICAL MODELS SYNTHESIS
   1. Definition and justification of the system functional scheme

Functional diagram is a scheme explaining the processes taking place in the individual functional circuits of the product or the product at all. The functional diagram is an explication of certain types of processes that occur in integral functional blocks and circuits of the device. To develop the functional scheme correctly it is required to understand what actions system will perform and what processes will occur inside it. Knowing the requirements declared in technical task it becomes clear that the system will consist of two main subsystems – server side part (also called as backend) and the user portal part which will be represented as mobile phone application and the Web API for interaction with server side.

The goal of this paper is to design and implement the mobile phone application for developed system. As required, the whole system will be distributed, so designing the presenter part this fact should be taken into account.

According to the previously mentioned condition the frontend subsystem should consist of two parts – the mobile application which will be used by a householder to interact with the system itself and the Web API service which will provide the REST-full service for mobile phone application communication with backend side.

The gateway device should be able to process and transfer data from sensor screen and video camera and also should be able to control and manage the state of the door locker. This device should be protected from physical vulnerabilities and secured from programing threats. And the main requirement is that this device should be able to work independently from other parts of the system, even when other parts of the automatic control system are broken or unavailable (e.g. when there is no internet connection).

When required data is received from sensor devices it should be transferred to the cloud service. This service should be designed as an asynchronous stateless application that can be automatically scaled depending on the network load.

The communication between these parts of subsystem should be performed using special service called IoT Hub. This service is intended to provide two-way communication between cloud hosted service and IoT devices without directly configured IP addresses of the IoT Devices. Each IoT Device uses the special connection string with configured address to the IoT hub, listens the incoming messages from it, and sends messages to it when needed.

When the visitor makes a request to ask the householder to open the door (pressing the special button on the touch screen) this signal comes to the gateway device, gateway device should transfer it to the cloud service using the IoT Hub. Then Cloud service should notify the householder with incoming requests. So developed system should contain the special communication channel between Cloud service and mobile application. Such service is called Notification Hub and is intended to send one way signals from the backend to the mobile phone. When the mobile phone receives the request signal, depending on the householder decision, it could request additional data, like video and audio streams, send the signal to open the door or just send reject to incoming request. This actions should be performed from the mobile application through the special interface application called Web API. This application provides public interface for operating with data managed by the cloud service.

The whole system should log every action happened inside, also statistics data should be collected so the cloud service should store required information in the remote database. Web API should provide the interface to retrieve this data from a mobile application.

Web API service should be implemented as a REST-full service. It also should contain an authentication and authorization layers inside to refuse unauthorized control request. It will make the developed system secure at all.

Depending on the primary description of the automatic control system “Smart intercom” from mobile development perspective its functional scheme will take the next form (Fig. 2.1.):

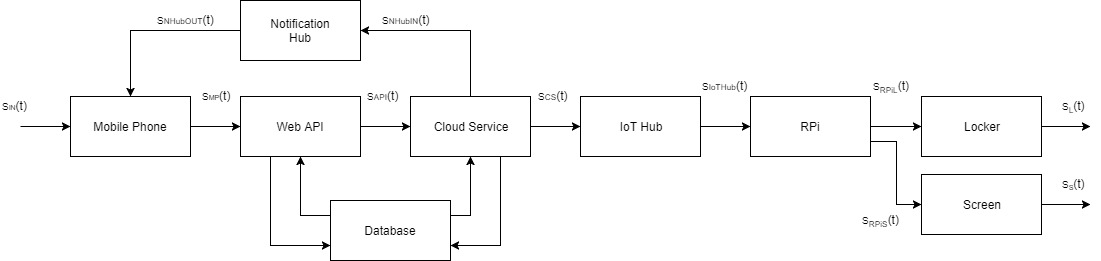


Figure 2.1 — Functional scheme of the automatic control system “Smart intercom”

There are next components on the scheme:

Mobile Phone – smartphone with a developed mobile application that provides a user friendly interface that allows householder to interact with the developed system.

Web API – application that provides a public interface for interaction with the system from the mobile application, represented by a REST-full service;

Cloud service – service intended to manage the data received from the gateway device;

Notification Hub – service intended to provide one way communication between cloud service and mobile application;

IoT Hub – service that provided a communication channel between the gateway device and cloud service;

RPI – a gateway device, represented by a single board computer;

Screen – device, intended to display required information to the visitor;

Locker – device that manages the state of the door locker;

Internal storage – device intended to store the internal configuration of the gateway device and to store temporary information until it will be transferred to the cloud service;

Database – storage engine for storing logs and statistics data;

There are next signals on the scheme:

— input signal from a user that uses mobile application to control the developed system;

— HTTP request to the Web API controller with request data;

– output signal from the Web API to cloud service represented by a request over http protocol;

– output signal from the cloud service represented by a request over http protocol;

– output signal from the IoT Hub represented by a request over http protocol;

– output signal from the gateway device to change the door locker state;

– output signal from the gateway device to display required information on the screen;

— output signal of a screen;

— output signal to change the door locker state;

– output signal from the cloud service to Notification Hub represented by a request over http protocol;

– output signal from the Notification Hub to mobile application represented by a request over http protocol;

* 1. Mathematical models

“Smart intercom” is a complex distributed automatic control system that requires a detailed design before the implementation. As a modern software system it should be designed according to a common software development principals and approaches.

The most important design principals in software development are Object Oriented Design principle and SOLID principle. This principles allows software developers to design complex system correctly and make it maintainable in the future.

In case of developed system the most reasonable will be design according to the SOLID principle, because this principle has already included the main ideas of OOD. In computer programming, the term SOLID is a mnemonic acronym for five design principles intended to make software designs more understandable, flexible and maintainable. The principles are:

1. The Single Responsibility Principle

There should never be more than one reason for a class to change. Basically, this means that classes should exist for one purpose only. Responsibility is the heart of this principle, so to rephrase there should never be more than one responsibility per class.

1. The Open Closed Principle

Software entities (classes, modules, functions, etc.) should be open for extension, but closed for modification. Developer can make an object behave differently without modifying it by using abstractions, or by placing behavior (responsibility) in derivative classes. In other words, by creating base classes with override-able functions.

1. The Liskov Substitution Principle

Functions that use pointers or references to base classes must be able to use objects of derived classes without knowing it. In other words, if developer is calling a method defined at a base class upon an abstracted class, the function must be implemented properly on the subtype class. Or, when using an object through its base class interface, the derived object must not expect such users to obey preconditions that are stronger than those required by the base class.

1. The Dependency Inversion Principle

Depend on abstractions, not on concretions or High level modules should not depend upon low level modules. Both should depend upon abstractions. Abstractions should not depend upon details. Details should depend upon abstractions. This is very closely related to the open closed principle. By passing dependencies to classes as abstractions, developer remove the need to program dependency specific.

1. The Interface Segregation Principle

Clients should not be forced to depend upon interfaces that they do not use. In other words, when a client depends upon a class that contains interfaces that the client does not use, but that other clients do use, then that client will be affected by the changes that those other clients force upon the class. This principle sounds like the inheritance specific single responsibility principle.

According to principals described above the first thing that should be done during the system design is defining the contracts which will be used for communication between modules of the system. In programming the mathematical representation of the module is the interface it provides.

According to the functional scheme and requirements description the interfaces for each module of the system are:

1. Screen module

This module is responsible for displaying the information containing in the buffer byte array to the screen. Its interface is:

*public interface IScreen*

*{*

*Void DisplayData(byte[] buffer);*

*}*

1. Locker module

This module is responsible for 3 actions – to unlock the door, to lock the door and to verify door’s state, so the interface will be next:

*public interface ILocker*

*{*

*void Lock();*

*void Unlock();*

*DoorState GetDoorState();*

*}*

DoorState object is an enumeration that can has several states: “Locked”, “Unlocked”, “Unavailable”.

1. Gateway module

Gateway module is intended to perform recurring actions for processing data from different sources and transferring this data to the cloud. Its interface should contain operations for primary initialization of aggregated devices and for starting the processing loop.

*public interface IGateway*

*{*

*void Initialize();*

*void Start();*

*}*

1. IoT Hub module

IoT Hub module should provide operation for sending the message to the cloud and also an event for receiving messages from the cloud service in asynchronous way in its interface:

*public interface IIotHubClient*

*{*

*void SentMessage(IotHubMessage message);*

*event EventHandler<IotHubMessageEventArgs> OnMessageReceived;*

*}*

According to this interface parent module that uses this client should subscribe on provided event and implement an event handler method that will be able to process *IotHubMessageEventArgs* object which contains the incoming message.

1. Cloud service module

The public interface of this module is represented by a Web API module which is a wrapper around the cloud service itself and its interface design is out of scope of this paper and will be described in other.

1. Media service module

This module is intended to transfer the stream captured by gateway from camera module to the mobile application. So the main operations it should provide are operations to open stream and to close the stream by stream identifier.

*public interface IMediaService*

*{*

*Stream OpenStream(Guid id);*

*bool CloseStream(Guid id);*

*}*

Guid is a global identifier that is a 128-bit number used to identify information in computer systems. It uses a special algorithms to generate random sequence of bytes to create a globally unique identifier in a distributed system and in the world at all.

1. Repository module

This module will provide the general interface for CRUD operations on all modules of developed system that works with data storing engines. It does not make sense what kind of storage is used – this module should implement the general interface which has operations for retrieving data from storage, for inserting data to storage, for updating existing data and for deleting specific data from storage. The interface will be next:

*public interface IRepository<TEntity>*

*{*

*TEntity[] GetAll();*

*TEntity Get(Guid id);*

*TEntity Get(Expression<Func<bool, TEntity>> filter);*

*TEntity GetFirst(Expression<Func<bool, TEntity>> filter);*

*TEntity Create(TEntity entity);*

*TEntity Update(TEntity entity);*

*void Delete(TEntity entity);*

*}*

Here TEntity is a generic type of the entities stored in database, Expression<Func<bool, TEntity>> is a type of the predicate which will be used to filter the result set of data on the storing engine side, so the client using the implementation of this interface will receive the filtered data passing the filter represented by a lambda expression to the corresponding method.

8)Web API module should be designed as a REST-full service. RESTful web service is a way of providing interoperability between computer systems on the Internet. REST-compliant Web services allow requesting systems to access and manipulate textual representations of Web resources using a uniform and predefined set of stateless operations. It works over HTTP protocol and uses GET, POST, PUT and DELETE Http method to retrieve, create, modify and delete the information on backend side. In developed system RESTful service should provide the endpoints for requesting video and audio stream, for changing the door state, for viewing and managing statistics and logging data and also for configuring the system at all.

When the contracts are defined it is required to design each module dependent on the provided interfaces but not on the specific implementation of these interfaces in case when one module is dependent on another. To resolve which implementation for each interface is created there is special tools called IoC (Inversion-of-control) containers where all dependencies should be registered. Using such containers gives possibility to change the implementation of a specific interface without making changes to the code base, and it makes supporting of developed system more convenient and easier.

* 1. Evaluation of simulation results