MINISTRY OF EDUCATION AND SCIENCE OF THE REPUBLIC OF KAZAKHSTAN
Non-commercial joint-stock company
«ALMATY UNIVERSITY OF POWER ENGINEERING AND TELECOMMUNICATIONS»
Department of IT-engineering

APPROVED by
Head of the department
PhD, associate professor
________________Kartbayev T.S.
«____» _______________ 2018 y.

DIPLOMA PROJECT

Theme: Developing a mobile application for social assistance to those in need

Specialty 5B070400 – «Computing systems and software»

Done by: Rakhmetov R.Zh. Group: BTa-14-4
Scientific supervisor: c.t.s, associate professor Musapirova G.D.

Consultants:
on an economic part PhD, associate professor _________________Tuzelbayev B.I.
«____» _____________ 2018 y.

on a health and safety: PhD, senior lecturer _________________ Begimbetova A.S.
«____» _____________ 2018 y.

on application of
computer facilities: senior lecturer _________________________Ramazanova A.M.
«____» _____________ 2018 y.

Compliance supervisor: PhD, senior lecturer Ж. Бидахмет__________ Bidakhmet Zh.
«____» _____________ 2018 y.

Reviewer: PhD, head of automation
department of SDU ____Suliev R.N.
«____» _____________ 2018 y.

Almaty 2018

MINISTRY OF EDUCATION AND SCIENCE OF THE REPUBLIC OF KAZAKHSTAN
«ALMATY UNIVERSITY OF POWER ENGINEERING AND TELECOMMUNICATIONS»
Non-commercial joint-stock company

Institute of Control Systems and Information Technologies

Department of IT-engineering

Specialty 5B070400 – «Computing systems and software»

ASSIGNMENT

for the diploma project

Student Rakhmetov Raimbek Zhexenuly

Theme of diploma project: Developing a mobile application for social assistance to those in need

Approved by the order of university №155 from «23» October 2017

Completion date of the finished work: «01» June 2018

Data conformity of project (research to required parameters of the result (design) and the initial data of the object): PHP, MySQL, Android Studio, Java programming language

List of issues of diploma project or the summary of the project:

a) main part;
b) analytical part;
c) software implementation part;
d) economical part;
e) health and safety;
f) Appendix A. Program code.

A list of illustrations (quantity): 12 tables, 47 figures.

Main recommended references:

1 Journal of Lorenzo Snow, 1841–47, Church History Library, 39–42.

2 Jiyoung Choi, Creating an evaluation system for a mobile application design to enhance usability and aesthetics, Iowa: 2012

3 Klimov, A. Preparation for the development, Basic components of Android,2000.[Online] Available: http://developer.alexanderklimov.ru /android/android1.php (Accessed 13 Nov.2015)

4 Loren D., Shein K. ANDROID for 24 hours. Moscow: Rid Group, 2011

Consulting of related sections

Sections	Consultants	Date	Signature
Economical part	Tuzelbayev B.I.	05.05.2018- 21.05.2018	
Health and safety	Begimbetova A.S.	05.05.2018- 21.05.2018	
Computer facilities	Ramazanova A.M.	05.05.2018- 23.05.2018	
Compliance supervisor	Bidakhmet Zh.	05.05.2018- 23.05.2018	пс. Бидахмет

Diploma project preparation schedule

Name of section and list of issues	Deadlines for presenting to the supervisor	Annotation
Main part	01.11.2017-15.12.2017	done
Analytical part	20.12.2017-16.02.2018	done
Software implementation part	20.02.2018-20.04.2018	done

Appointing date of assignment «25» October 2017.

Head of department ______________________________ Kartbayev T.S.

Scientific supervisor ___________________________ Musapirova G.D.

Accepted by the student Р. Рахмет________________ Rakhmetov R.Zh.

АҢДАТПА

Дипломдық проекттің басты мақсаты мейірімділік пен қайырымдылықтың дамуына үлес қосатын мобильді қосымшаны құру.

Зерттеу объектісі - дипломдық жобада қарастырылып отырған белгілі аудитория үшін қолайлы, смартфондарға арналған операциалық жүйені зерттеу үшін қолданылатын ақпараттық технологиялар.

Зерттеу пәні – Алматы тұрғындарының арасында мейірімділікті дамыту мақсатында мобильді қосымша.

Дипломдық жобаның аясында мейірімділікті дамыту бойынша қолданыстағы бар мобильді қосымшаларға шолу жасалды.

Дипломдық жоба мазмұны, кіріспе, негізгі талдамалық бөлім, практикалық әзірлеу, қорытынды мен қолданылған әдебиеттер тізімінен тұрады.

АННОТАЦИЯ

Основная цель дипломного проекта является развитие добродетельности и благотворительности путем создания мобильных приложений.

Объект исследования - существующие информационные технологии для изучения операционной системы для смартфонов, которая наиболее подходит для данной целевой аудитории.

Предмет исследования - использование мобильного приложения в развитии добродетельности у жителей города Алматы.

В рамках дипломного проекта был сделан обзор и анализ существующих аналогов по развитию доброты.

Дипломный проект по структуре состоит из содержания, введения, основной аналитической части, практической разработки, заключения и списка используемой литературы.

ANNOTATION

The main goal of the diploma project is the development of virtue and charity in the creation of mobile applications.

The object of research is existing information technologies for studying operating system for smartphones that is most suitable for a given target audience.

The subject of the study is the use of mobile applications in the development of virtue among Almaty residents.

As part of the diploma project was made a review and analysis of existing analogues for the development of kindness.

Information about the diploma project: the structure of the work is presented introduction, the main analytical part, practical development, conclusion and a list of used literature.

Contents

List of terms and abbreviations 8
Introduction 9
1 Review and analysis of existing mobile applications charity development 10
2 Technological support and modeling application “dobro” 19
2.1 User Interface design technologies 19
2.1.1 Mobile-centered Design 19
2.1.2 UI/UX design principles 20
2.1.3 Justinmind 21
2.2 Operating system: Android technology 25
2.2.1 OS - Android 25
2.2.2 Android Studio platform 26
2.2.3 The Java programming language 35
2.2.4 Android Market 42
2.2.5 Technology review 44
2.2.6 Models 45
2.2.7 Modeling functional software 46
3 Development of mobile application "Dobro" 48
3.1 The characteristics and capabilities Software 48
3.1.1 User interface (UI) 48
3.1.2 Backendless 52
3.1.3 Firebase 53
3.2 Analysis of the mobile application “Dobro” for promoting kindness 55
3.2.1 End - User - License Agreement 55
3.2.2 Results and statistics on Android OS 57
4 Economic part 62
4.1 Stages and timing of the project implementation 62
4.2 Calculating the cost of developing an application 63
4.2.1 Calculation of material costs 63
4.2.2 Calculation of expended amounts for electricity 64
4.2.3 Calculation of labor costs 65
4.2.4 Calculation of deductions and social needs 66
4.2.5 Calculation of other costs 66
4.2.7 Budgeting 67
4.2.8 Determination of the possible (contractual) price of project 67
4.3 Assessment of the scientific and technical performance of the project 69
5 Health and safety 69

5.1 Analysis of working conditions 70
5.2 Calculation of artificial lighting 72

5.3 Development of fire safety issues 74
5.4 Calculation of air conditioning in an office building 75
Conclusion 78
List of references 79
Appendix A **Ошибка! Закладка не определена.**

List of terms and abbreviations

API	Application Programming Interface
DB	Database
DMS	Database Management System
HTTP	The Hypertext Transfer Protocol
IDE	Integrated Development Environment
IS	Information System
IT	Information Technology
JVM	Java Virtual Machine
MVC	Model View Controller
OHA	Open Handset Alliance
OS	Operating System
PC	Personal Computer
RK	Republic of Kazakhstan
SDK	Software Development Kit
UI	User Interface
UML	Unified Modeling Language
UX	User Experience
XML	Extensible Markup Language
XNU	X is Not Unix

Introduction

Today, many people need help. For certain everybody faced with difficult situations, which could not do without assistance. Citing an example, somebody have a broken car or a flat tire, somebody darn and need some kind of tool. We have a solution that provides interesting opportunities. For example, if somebody faced with the fact that his car broke down in the middle of the road. What would he do with it?

We are going to develop mobile application “Dobro”. The aim is promotion of virtue, charity development. The requirements which was given is being of charity call (challenge) of another person, group of people, companies, corporations to do good deeds, charitable events, help the needy, to hold charity events and so on.

Nowadays people who need help address to volunteers or charities or if they are out and need in little help they address to passersby. A volunteer seeks the people by social network like Instagram, VK and give people to know that they are exist like advertising. People in Almaty day by day use smartphones actively. We have many applications which really useful for citizens even they are out. And our projections that such applications like “Dobro” will also be useful.

1 Review and analysis of existing mobile applications charity development

Lorenzo Snow wrote in his diary "The provision of one service often leads to the receipt of another one" [1].

1.1 Description of the research problem

Today the phrase "to help a person" is often associated with financial assistance or with connections. But is this the only way to help our neighbor? Of course, not. After all, many of us had people who once helped with a word, advice, just listened to you at a difficult moment or helped in the performance of any necessary case. And this is much more important than simply closing the issue with money. Help is a kind of communication between people, where good words, through good deeds or actions.

Every once in life felt that joy and warmth that arise when from someone who sincerely and kindly motivates someone you help, support or save. “After a person helps someone feels a huge surge of energy” [2]. This idea has been proven in practice by the psychologists. He has the desire to live, the strength to resist his obsessive thoughts and the reason to be more often in such an inspiring state of positive and good. That is why helping others helps a person get out of a seemingly hopeless situation and turn his thoughts into a creative channel.

There is an opportunity to track how many people you helped, communicate with them, and also to whom other people helped in the application. There is also a page of ratings where you can see active virtues.

1.2 Analysis of the existing application for propagation of kindness

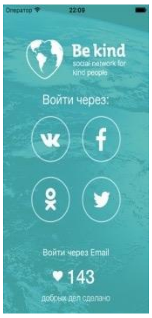
The purpose of this section is to find analogues. Analytical review ends clarifying the tasks of research, and is expected to further specify the formulation of research hypothesis.

It is well known that with every day applications for the operating system of Android OS still more and more, on the basis of these conclusions as well as developers and customers greater importance is given to quality, defaults loaded by users, and estimates of an application to be confident in your choice. The developers of each app, follow the specific innovations and regularly update the application itself, which takes into account the opinions and reviews from users.

The analysis of existing research is recommended to use for comparison and user reviews also using one of the following present application. Also, the object of study is static data obtained by observing, testing or experiments. Because each application has its own pros and cons.

The main sources that reveal comparison app store were Google's app "Android Market" (Google Play, Play Market). After each found workings was tested as a result of the review and analysis of the four applications with "Dobro", we got a comparative table of applications. Below tables it is advisable to itemize certain characteristics and comparison, that show exactly what differences exist from analogues. Table 1.1 describes the analogues of the application “Dobro”.

Table 1.1 - Review and analysis of such applications

Characteristics/Comparisons	Be kind  Be kind social network for kind people Войти через: Войти через Email 143 добрых дел сделано	Добродел	ДариДобро	Mercify
Creator	Pavel Kozlov	MoscowRegionGovernment	ITECH Mobile LLC	MerifyTechnologies GmbH
Rank out of 5	5	4	4	5
Age limits	17+	4+	4+	17+
Language of the application	Russian	Russian	Russian	Russian
Ratings	-	11	8	18
The rating scale	Free	Free	Free	Free
Application speed	Fast	Fast	Slowly	Slowly
Exit button	No	Yes, it does not always work	No	Yes, it does not always work
Confirm exit	No	Yes	No	Yes
	Does it work without access to the Internet?	Absent	Does it work without access to the Internet	Absent

Continue of Table 1.1

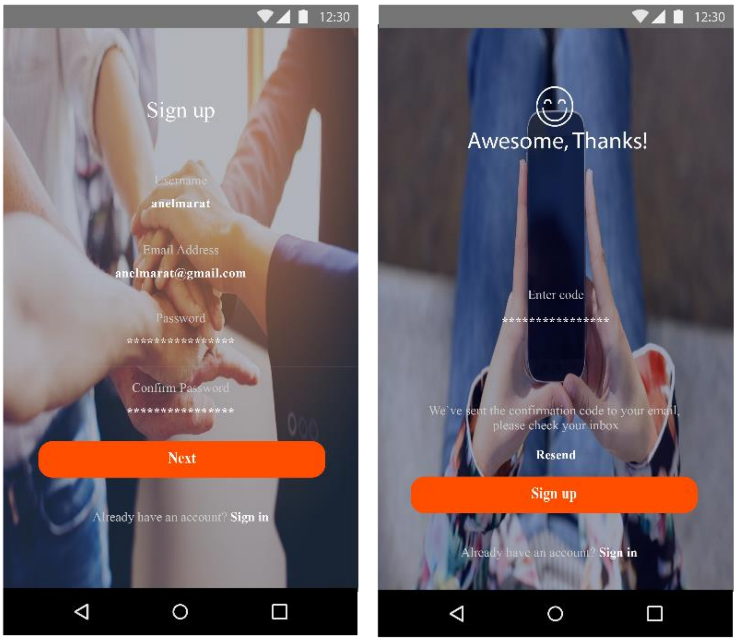
News	- Fixed problems with moving the map - Fixed problems with the operation of the button "back" - to notify - The opportunity to hide the contact details	No	Small optimization	Still image distortion
Version	1.2	2.9	1.0.4	1.0.1
Seller	Pavel Kozlov	MoscowRegionGovernment	ITECH Mobile LLC	MerifyTechnologiesGmbH
Logo	Unclear	Appropriate application	Unfinished	Simple, intuitive
Advertising	Endless, It does not close, lights bright	Distracting advertising	Available	Available
Design	Bright, pleasant to the eye	Modern user-friendly design, developed specifically easier and more pleasant	Incorrectc ombination of colors	Simple
Manufacturercountry	Russian	Russian	Mira, 26, Ulyanovsk, Ulyanovskregion., 432011	Russian
Lastupdate	February 17, 2016	October 24, 2016 г	November 03, 2014	October 10, 2016 г

Continue of Table 1.1

Description	A social network for kind and caring people, and also to solve many problems that can make our world a little kinder. The basic idea is that – in exchange for their assistance a person should perform three acts of kindness to others	You can quickly get information about what the organization or people who live in the Ulyanovsk region, need help at the moment	The application allows direct reporting to the Moscow Region Government on problems. Pits on the roads? Garbage dumps? Could not get to the doctor?	It's an online auction, where a single type of lot is meeting people and collected money goes to those who need them most
Pros / pieces	There is a map. Counts the amount of good. You can compete. You can create a list of requests for good works, from which you can select the object of assistance including on the principle of geolocation	You can ask for help «in Real time»	There are categories. List of all need help at the moment	Everything is convenient. Anyone can create his own charity event, and gives users full information about it
Minuses	No	You open "messages on proximity", but you can not read it completely. Only a piece	No	No
Numberof units	1 000–5 000	100-500	100-500	?
Size	4,1M	3,3M	2,5M	5,1M

1.2.1 Analyzing the function of analogs

The analysis of the diploma project includes comparing similar applications as foreign so local. The sphere of the analysis selected is the market of Russia. Four applications were selected on the thesis subject, such as «Добродел», «Be kind», «Добро».


12:30

Sign up

Username
anelmarat

Email Address
anelmarat@gmail.com

Password
****

Confirm Password
****

Already have an account? Sign in

12:30

Awesome, Thanks!

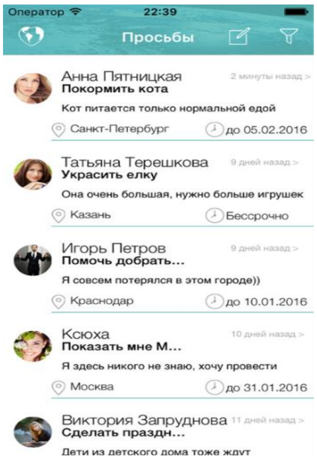
Enter code
****

We`ve sent the confirmation code to your email, please check your inbox

Already have an account? Sign in

Figure 1.1 - Homepage “Добродел”

"Messages on proximity", you cannot read it completely, only a piece of messages as shown in figure 1.1 It is not clear what is the actual problem in humans.


Оператор 22:39

Просьбы

Анна Пятницкая 2 минуты назад >
Покормить кота
Кот питается только нормальной едой
Санкт-Петербург до 05.02.2016

Татьяна Терешкова 9 дней назад >
Украсить елку
Она очень большая, нужно больше игрушек
Казань Бессрочно

Игорь Петров 9 дней назад >
Помочь добрать...
Я совсем потерялся в этом городе))
Краснодар до 10.01.2016

Ксюха 10 дней назад >
Показать мне М...
Я здесь никого не знаю, хочу провести
Москва до 31.01.2016

Виктория Запруднова 11 дней назад >
Сделать праздн...
Дети из детского дома тоже ждут

Figure 1.2 - Application interface “Be kind”

In figure 1.2 the section of my request there is no filter on cities and there is also no scale "the most important requests».

We can conclude that all applications are similar on functionality, but there are essential distinctions.

1.2.2 Evaluation table for existing applications

The beginning of a project is the time to really get to know the users we will be designing for. To start a project, user research might include evaluating existing products to find usability issues or to understand competitor products. This is a great time to talk to our users about their needs and how they organize their offline approach to your subject.

This section, analysis of analogues applications for propagation of kindness shows how the evaluation is used to analyze to analyze and evaluate the mobile application designs for kindness propagation. The existing three mobile applications were selected according to different ratings and contents. They were analyzed and evaluated with the evaluation table in appendix A and in the table 1.2 shows the score scale and evaluation for results of the evaluation table. The purpose of using the evaluation table is to rate the current mobile application designs for kindness propagation and determine the aspects of usability and aesthetics from them. To evaluate a mobile application design, we have to know who will use the application, why they want to use the application and what content is the application. Therefore, in this part a case study analyzing mobile application designs for kindness propagation will be examined.

Table 1.2 – Score Scale and Evaluation for results of the evaluation table [3]

Score scale	Evaluation
113-125	Excellent
100-112	Good
87-99	Fair
74-86	Poor
0-73	Unacceptable

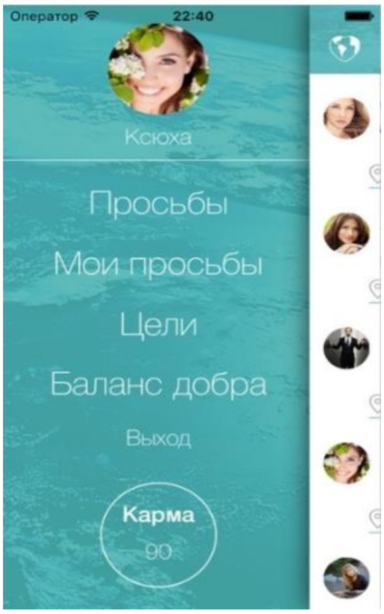
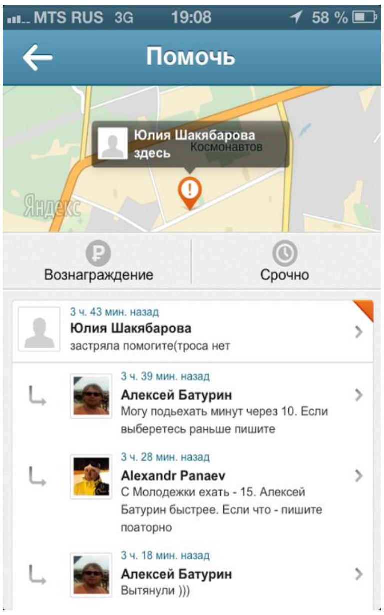


Figure 1.3 – Menu bar of the application “Be kind”

As the figure 1.3 shows the menu bar of the application “Be kind”, here the right side interface of application is designed incorrectly. Because the background main page interface should be shown fully, i.e. on the right of the menu bar there should be shown the right part of main page.



3 ч. 43 мин. назад
Юлия Шакябарова
застряла помогите(троса нет

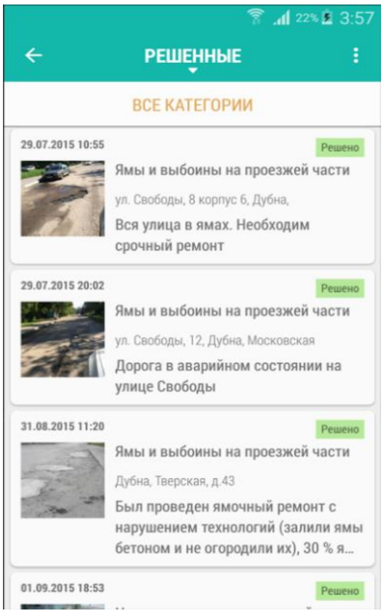
3 ч. 39 мин. назад
Алексей Батурин
Могу подьехать минут через 10. Если выберетесь раньше пишите

3 ч. 28 мин. назад
Alexandr Panaev
С Молодежки ехать - 15. Алексей Батурин быстрее. Если что - пишите поаторно

3 ч. 18 мин. назад
Алексей Батурин
Вытянули)))

Figure 1.4 – Application interface “Спасибодруг!”

In the next application interface, figure 1.4, the main thing which we noticed is an incorrect choice of icon elements and their colors for certain functionally.



29.07.2015 10:55 Решено
Ямы и выбоины на проезжей части
ул. Свободы, 8 корпус 6, Дубна,
Вся улица в ямах. Необходим срочный ремонт

29.07.2015 20:02 Решено
Ямы и выбоины на проезжей части
ул. Свободы, 12, Дубна, Московская
Дорога в аварийном состоянии на улице Свободы

31.08.2015 11:20 Решено
Ямы и выбоины на проезжей части
Дубна, Тверская, д.43
Был проведен ямочный ремонт с нарушением технологий (залили ямы бетоном и не огородили их), 30 % я...

01.09.2015 18:53 Решено

Figure 1.5 – Solved problems page of the application “Добродел”

Figure 1.5 shows the solved problems page interface of the application “Добродел”. Here the size of fonts is unreadable and the spacing between the texts and images is incorrect.

1.3 Problem statement

The main objective of this diploma work are:

a) the process of designing the user interface and the experience that results from its use. The Design process is typically four phases: user research, wireframing, visual design, production and development. To start create user interface design after research stage, it`s necessary to study the general principles in the development of mobile application for UI and UX designers. The successful and ideal user interface design of mobile application is dependent first of all, on the fully understanding the purpose of the application and the expectation of users. Moreover, for creation of UI design there are main theoretical themes to study hard and program tools primarily, to build the prototype of application and the ultimate user interface design of our mobile application UI:

1) Mobile-Centered Design;
2) UX design principles;
3) Justinmind;
4) Adobe Photoshop.

b) 3to create an application based on Android OS. The development of this task can be divided into several sub-tasks:

1) study of the Android OS;
2) introduction to Android Studio platform;
3) mastering the Java programming language.

4) conduct research and analysis of such applications;
5) implementation of the application.

It must be implemented to ensure the functional simulation, namely:

- development of a class diagram;
- organization of the access level;
- development of sequence diagram for the search function;
- development of the notification function.

2 Technological support and modeling application “dobro”

This section explains the basic technology that has been used in this project with a full explanation of functions and features.

2.1 User Interface design technologies

2.1.1 Mobile-centered Design

People use diverse interfaces in their daily lives. It is likely that most people have experienced difficulties at some point while they used it. The problem is not about the users, but about the design of the product or service. Don Norman, an academician in the field of cognitive science, design and usability engineering, has said that if users are having trouble using something, it is not their fault, but it is most likely poor design. If a product or interface design is not easy to use and makes users confused to use it, the design has not considered the factors of user-centered design.

What is “User-Centered design?” According to Stone et al., in his book, User Interface Design and Evaluation, “user-centered design is an approach to user interface design and development that involves users throughout the design and development process.” He continues, “user-centered design not only focuses on understanding the users of a computer system under development but also requires understanding of the tasks that users will perform with the system (Stone et al., 2005, p. 15)”. For example, website design with a user-centered design approach helps users to use the website effectively without obstacles. It is created by understanding the users and considering their limitations, their purpose of using the website and their expected tasks.

User-centered design also helps users have a high degree of usability. According to Stone et al., usability is defined as “the extent to which a product can be used by particular users to achieve specified goals with effectiveness, efficiency, and satisfaction in a particular context of use” (Stone et al., 2005, p. 6). If a website has a low degree of usability, users will be frustrated trying to use it, may not use it as much, or may not come back to use it again.

According to Steve Krug, in his book, don’t make me think, it is most important to make a web site that does not force the user to think in order to make the site easy to use (Krug, 2000, p. 11). A web site page that does not make users think enables users to understand the sources and contents of the page instantly without taking time. The reasons why some web sites make users think is that the sources of the sites have problems with usability. For example, if a link or button is not clickable, users come to a stop and think whether they should click it or not. Therefore, it takes some time for users to do the task distracting the users’ attention from the task at hand (Krug, 2000, p. 15).

What about user-centered design for a mobile application? The Smartphone has the characteristic of a small screen size, so the application needs to be designed to fit the smaller screen. The effective use of visual elements and concepts of usability need to be considered for making a user-centered mobile application.

2.1.2 UI/UX design principles

Before we go any further, let's define the most common user interface design principles:

- The Structure Principle. Design should organize the user interface purposefully, in meaningful and useful ways based on clear, consistent models that are apparent and recognizable to users, putting related things together and separating unrelated things, differentiating dissimilar things and making similar things resemble one another. The structure principle is concerned with overall user interface architecture.
- The Simplicity Principle. The design should make simple, common tasks easy, communicating clearly and simply in the user's own language, and providing good shortcuts that are meaningfully related to longer procedures.
- The Visibility Principle. The design should make all needed options and materials for a given task visible without distracting the user with extraneous or redundant information. Good designs don't overwhelm users with alternatives or confuse them with unneeded information.
- The Feedback Principle. The design should keep users informed of actions or interpretations, changes of state or condition, and errors or exceptions that are relevant and of interest to the user through clear, concise, and unambiguous language familiar to users.
- The Tolerance Principle. The design should be flexible and tolerant, reducing the cost of mistakes and misuse by allowing undoing and redoing, while also preventing errors wherever possible by tolerating varied inputs and sequences and by interpreting all reasonable actions.
- The Reuse Principle. The design should reuse internal and external components and behaviors, maintaining consistency with purpose rather than merely arbitrary consistency, thus reducing the need for users to rethink and remember. However, it is very important to decide which classes should be bound together by a package. Improper selection of classes can create unwanted dependencies and turn usage of that package into a nightmare.

Hannah Alvarez of User Testing gave this advice for UI beginners in their eBook, "Getting Out of the Office: Testing Mobile App Prototypes With User": "Ever heard the saying, 'Measure twice, cut once?' Well, that idea applies to building apps, too. When you're designing an app, you can verify you're on the right track and avoid doing costly rework by user testing your prototypes. You can iron out the kinks in the design before you've written any code, saving your team the time and hassle of making changes in development. Plus, you’ll already know how users will receive your product before it hits the app store”.

2.1.3 Justinmind

Justinmind is a prototyping tool that allows you to create unique, interactive and life-like simulations of your web and mobile applications. With Justinmind, you can easily add to your corporate image prototypes, export them to HTML for online presentation, and automatically create specification documents in Microsoft Word document. Do all this without programming or code line creation knowledge. Instant Modeling: by clicking on the "Simulate" button, you can see your prototype on the move and interact as if it were a real application or web site. No code required. Justinmind is an intuitive tool and you can drag and drop the required components from the floating palettes to the workspace. Automatically generated documents. Create all necessary documentation with just a few clicks.

Exported as HTML. You can export wireframes of your web sites or applications or prototypes in a variety of formats (HTML, .csv and MS Word documents), so you can see, interact with and know how your customers and users look. Avoid improving reusability and usability in your applications by including Justinmind from the very beginning of your project.

If you publish your protocols to your online Justinmind account, your clients can interpret your projects in a collaborative way. Their opinions are like chat streams in the forum and they use their comments; and if you include this step in the prototype stage, remember that you will save time in the next development stages.

Here are some tips to help you get your prototypes started with Justinmind

A prototype is a preliminary version or visual visualization of a software, application, or website. The prototype goal is to show how a program or application will behave and behave when it completes its development to better understand its capabilities and limitations early on. It can also be expressed as a model or frame depending on the level of accuracy.

Learn more about the difference between layout, frame and prototype.

A user interface is a collection of all graphical or visual components, commands and behaviors on a platform or a vehicle, and in fact, interacts between the user and the prototyping tool.

The display is a group of visual components that are grouped together in the edit field. It can also be called a page.

A functional scenario is a graph of application components that represent an application's overall behavior, such as navigation and operations.

An interpretation is a written comment made by a user who is referencing prototypes.

Getting Started with Justinmind. To start using Justinmind, you need to go to the installation folder (or any shortcut you may choose to create in the operating system / desktop menu) and double-click the icon.

Just before the actual application interface is displayed, the first image with the progress bar is displayed briefly.

If you are launching the application for the first time or if you have not disabled functionality in the "File" > "Settings" menu, a welcome window with shortcuts for the global task will appear. This window offers several options to help you get started with the program.

In the first stage of our diploma project we create the prototype of our application in this tool and below the screenshots are shown. The first page when user open the application is sign in page as shown in figure 2.1 The user can sign in or if he did not sign up, he can register through sign up button. The sign up page contains forms which include information about user like user name, nickname, email address, about user.

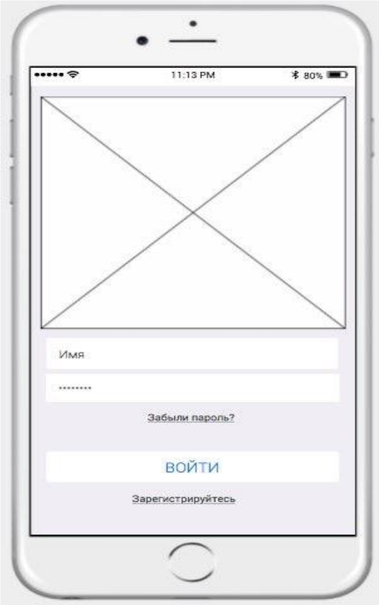
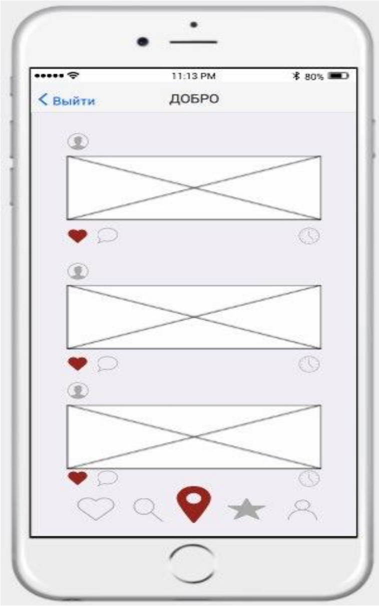
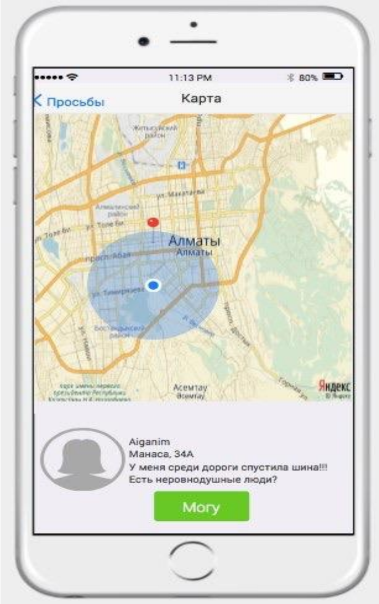


Figure 2.1 – The Sign in page



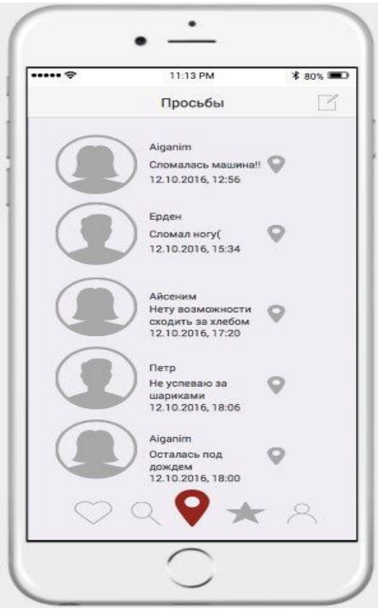
Fiqure 2.2 – The main news feed page

The figure 2.2 shows the main page that opened after sign in or sign up proces. There are new post by users of this application. New post contains user icon, photo and like, comment buttons. The main functionality of our mobile application is map and it shown in figure 2.3 There are map of the city, map pointers and the selected user request.



Aiganim
Манаса, 34А
У меня среди дороги спустила шина!!!
Есть неровнодушные люди?

Fiqure 2.3 – The map of the Almaty city



11:13 PM 80%

Просьбы

Aiganim
Сломалась машина!!
12.10.2016, 12:56

Ерден
Сломал ногу(
12.10.2016, 15:34

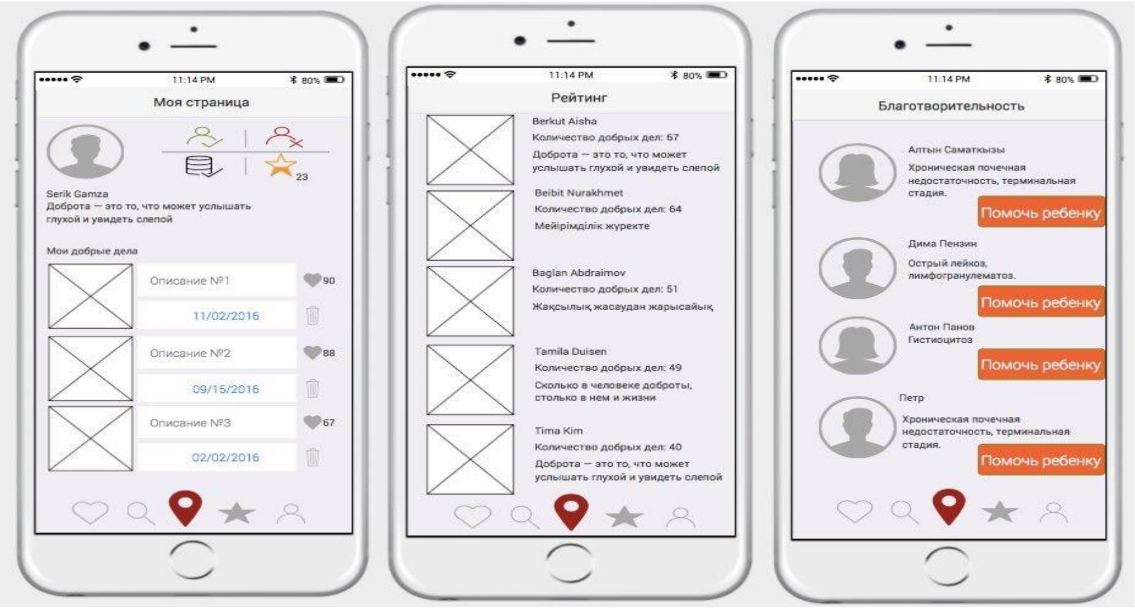
Айсеним
Нету возможности сходить за хлебом
12.10.2016, 17:20

Петр
Не успеваю за шариками
12.10.2016, 18:06

Aiganim
Осталась под дождем
12.10.2016, 18:00

Figure 2.4 – List of requests

The figure 2.4 shows the requests page. There is list of requests which users ask for help, their time interval and locations in map.



11:14 PM 80%

Моя страница

23

Serik Gamza
Доброта — это то, что может услышать глухой и увидеть слепой

Мои добрые дела

Описание №1 11/02/2016	90
Описание №2 09/15/2016	88
Описание №3 02/02/2016	67

11:14 PM 80%

Рейтинг

Berkut Aisha
Количество добрых дел: 67
Доброта — это то, что может услышать глухой и увидеть слепой

Beibit Nurakhmet
Количество добрых дел: 64
Мейірімділік жүректе

Baglan Abdraimov
Количество добрых дел: 51
Жақсылық жасаудан жарысайық

Tamila Duisen
Количество добрых дел: 49
Сколько в человеке доброты, столько в нем и жизни

Tima Kim
Количество добрых дел: 40
Доброта — это то, что может услышать глухой и увидеть слепой

11:14 PM 80%

Благотворительность

Алтын Саматкызы
Хроническая почечная недостаточность, терминальная стадия.
Помочь ребенку

Дима Пензин
Острый лейкоз, лимфогранулематоз.
Помочь ребенку

Антон Панов
Гистиоцитоз
Помочь ребенку

Петр
Хроническая почечная недостаточность, терминальная стадия.
Помочь ребенку

Figure 2.5 – User profile, Rating and Charity pages

User can manage with own profile; see the rating of users who did a lot of good deed and charity information as shows in figure 2.5. Also, he can choose activites in the right corner menu.

2.2 Operating system: Android technology

2.2.1 OS - Android

Android is an operating system (OS) for smartphones, tablets, laptops, eBooks, digital players, "smart" watches, game consoles, the Google glasses, TVs, and other devices. Based on the Linux kernel with Java programming interface. Android can create Java applications.

The Android has components such as:

- Activity - a screen of the user interface, which is represented by the Activity class and markings in the form of XML-file. Android - the application can consist of a number of active and can switch between them at runtime (i.e. the page).

- Intentions - asynchronous messages which allow the functions of the other services or activities. An application can make direct inquiries service or action (explicit intent) or request a registered Android services and applications (implicit intention). For example, an application may request an application through Intent Contact us for information or start the browser when you click links in the text of your application.

- Presentation, View, Control Component - is the user interface creating widgets classes. These are familiar to us elements: buttons, text fields, check boxes, switches, etc.

- Services perform background tasks without presentation of the user interface (for example, music playback). They can notify the user through the notification system Android - Content Provider - providing data applications. With content provider your application can exchange data with other applications.

- The receiver broadcast messages / requests - receives system messages and implicit intent can be used to respond to changes in the system status. An application can be registered as a receiver, and specific events can be triggered when an event occurs [4].

Below is a figure 2.6 show it looks like Android logo.

Figure 2.6 - Android logo

2.2.2 Android Studio platform

In 2007, a group of mobile manufacturers, wireless service providers and software developers (especially Google) formed the Alliance of open standards for mobile developers (OHA, Open Handset Alliance) to create next-generation wireless technologies. In contrast to the existing platform, a new platform should be free, based on open standards that enable reducing development costs and increasing profits. Mobile application developers should also have full access to the phone functions to create innovative applications [5].

Android Studio is an integrated application development environment for Android OS based on Intellij IDEA platform. This development environment is available for Windows, OS X and Linux. Build applications based on Gradle, i.e. the system of automatic assembly. Figure 2.7 - shows Android Studio icon.

Figure 2.7 - Android Studio icon

When developing applications in Android Studio, you can see any visual changes, which we produce in real time in the application itself. That is, in Android Studio, you can see how the developed application will look on a variety of devices running Android, with different settings and the expansion of the screen. The program also involves the drag function, with which you can move components through the user interface. Advantages of Android Studio:

- Reliable and simple development environment - It is easy to check application performance on different types of devices
- Assistants and templates for elements in common programming for Android.

When downloading and installing applications launch window opens as shown in figure 2.8.

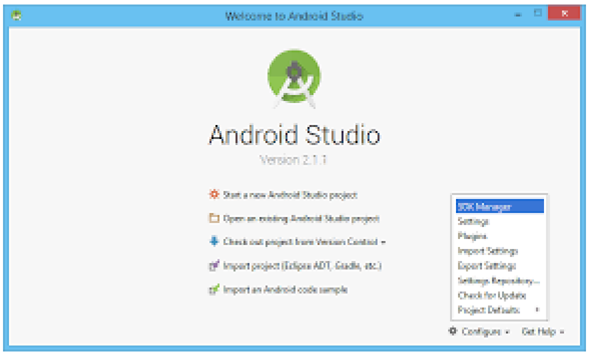

Welcome to Android Studio
Android Studio
Version 2.1.1
Start a new Android Studio project
Open an existing Android Studio project
Check out project from Version Control ▾
Import project (Eclipse ADT, Gradle, etc.)
Import an Android code sample
SDK Manager
Settings
Plugins
Import Settings
Export Settings
Settings Repository...
Check for Update
Project Defaults ▸
Configure ▾
Get Help ▾

Figure 2.8 - Welcome window Android Studio

To write the PP code in this project, there will be a used environment for developing Android Studio, and Java will serve as the language. To work in Android Studio, you must have the Java SE Development Kit (JDK) installed on the system you are using. The recommended version of JDK is at least 7. You can install it by following the link.

Java SE Development Kit is a free development toolkit and is distributed free of charge. So, after going through the above link, you need to choose the version that corresponds to your OS, Windows, Linux or Mac OS X. After downloading the executable file, you must run it and follow the instructions to install this tool.

After that, download the Android Studio installation file from the official Android Developer site. Installation files are available at.

Here you should also select the version that suits your OS and after the download of the executable file is complete, install the program following the instructions. Here it is worthwhile to pay attention to the fact that Android NDK loading for work in Android Studio is not required, since all necessary utilities are already included in the package of the application.

After launching the application, you will be greeted with the Welcome screen shown in figure 2.9.

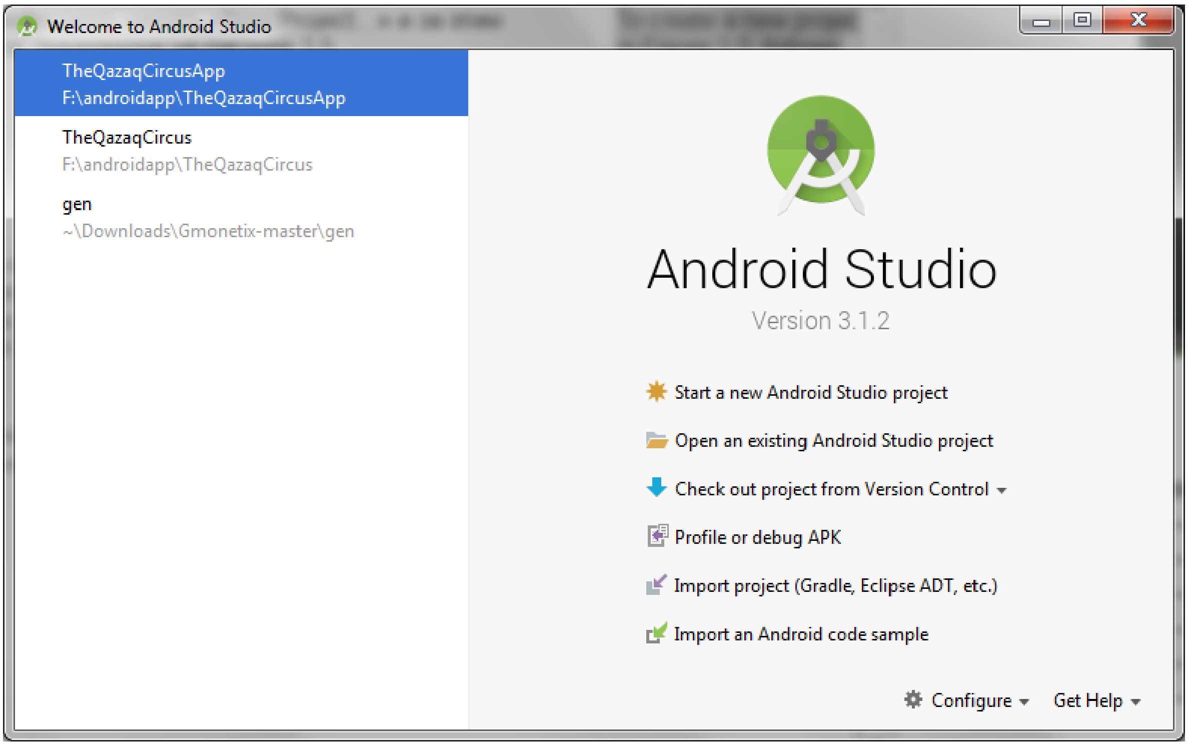

Welcome to Android Studio
TheQazaqCircusApp
F:\androidapp\TheQazaqCircusApp
TheQazaqCircus
F:\androidapp\TheQazaqCircus
gen
~\Downloads\Gmonetix-master\gen
Android Studio
Version 3.1.2
Start a new Android Studio project
Open an existing Android Studio project
Check out project from Version Control
Profile or debug APK
Import project (Gradle, Eclipse ADT, etc.)
Import an Android code sample
Configure
Get Help

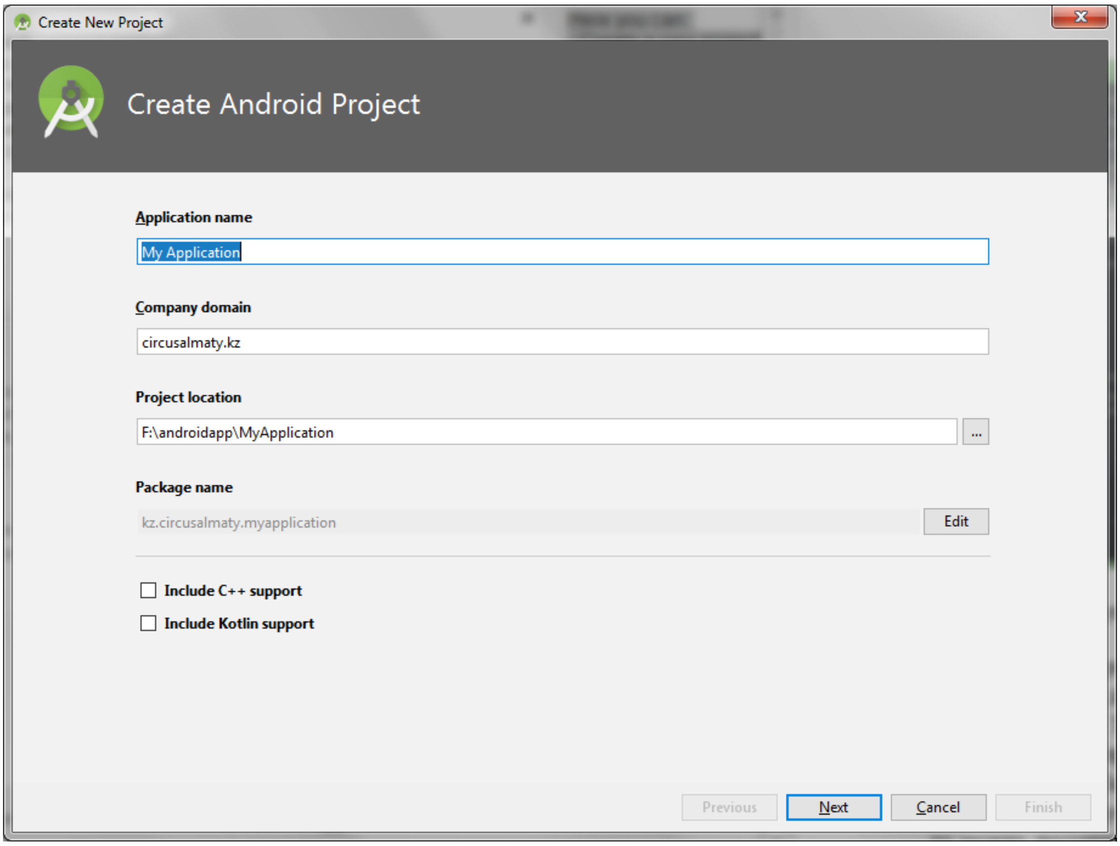
Figure 2.9 - Welcome screen of Android Studio

Here you can:

- create a new project;
- import an existing project;
- open an existing project;

- open any recent project, and perform a search among them;
- check the software version is up-to-date;
- edit Android Studio settings;
- get access to the right.

To create a new project, click "Start a new Android..." and the project creation window, shown in figure 2.10, follows.



Create New Project

Create Android Project

Application name

My Application

Company domain

circusalmaty.kz

Project location

F:\androidapp\MyApplication

Package name

kz.circusalmaty.myapplication

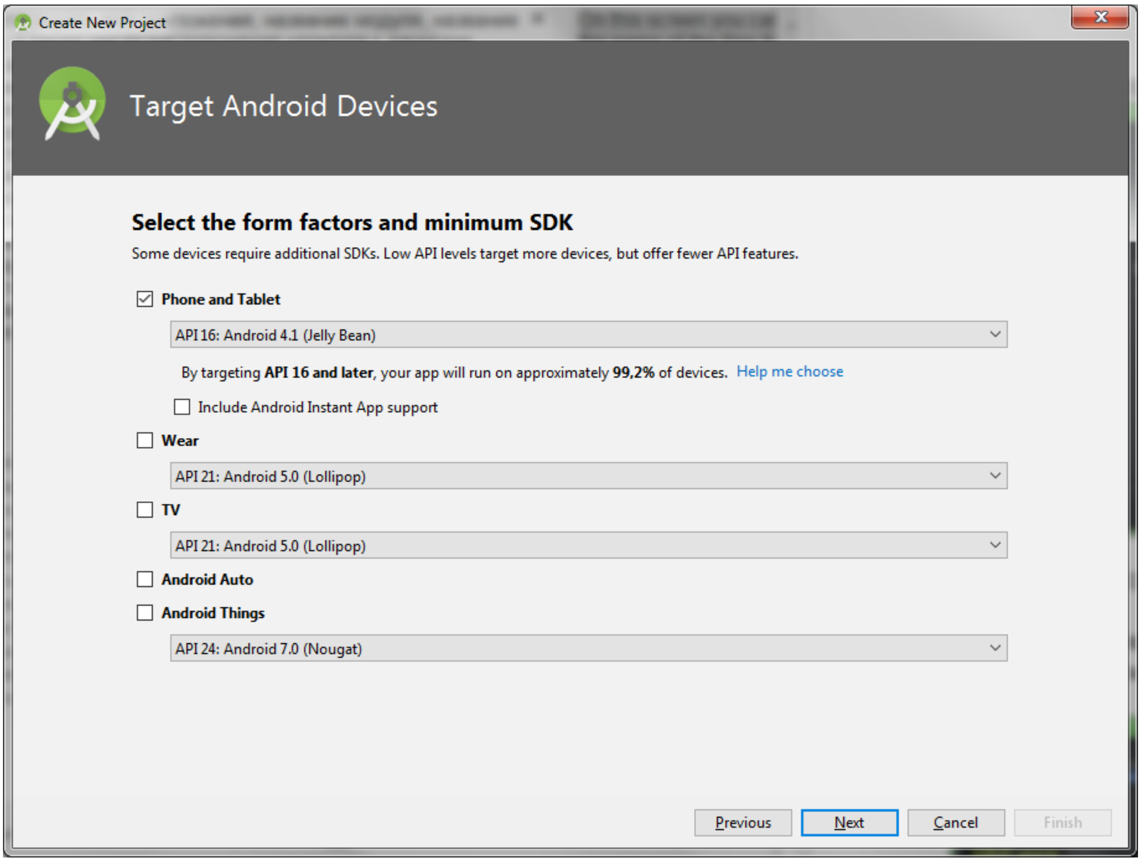
☐ Include C++ support

☐ Include Kotlin support

Figure 2.10 - Android Studio project creation window

On this screen you can specify the name of the application, the name of the module, the name of the files for the application, and the location of the directory with the project data. There is also a possibility to choose the minimum version of the OS supported by the application and the main one, on which it is oriented first. In addition, there is a choice of language level. Immediately available is the choice of the basic theme of the application, light and dark. If required, you can put a marker in the box of the "Create custom launcher icon" option, to create an application icon directly in Android Studio. And the conclusion is the choice of "Support Mode".

If the "Create custom launcher icon" was selected in the project window, after clicking the "next" button, you will see the window for creating and editing the application icon, which is also shown in figure 2.6. This editor will satisfy the needs of almost all developers, because it allows you to create quickly and qualitatively the application icons for different screen densities and immediately see the result.



Select the form factors and minimum SDK

Some devices require additional SDKs. Low API levels target more devices, but offer fewer API features.

☑ Phone and Tablet

API 16: Android 4.1 (Jelly Bean)

By targeting **API 16 and later**, your app will run on approximately **99,2%** of devices. Help me choose

□ Include Android Instant App support

□ Wear

API 21: Android 5.0 (Lollipop)

□ TV

API 21: Android 5.0 (Lollipop)

□ Android Auto

□ Android Things

API 24: Android 7.0 (Nougat)

Figure 2.11 - The window for selecting the form factor and minimum SDK

The SDK platform is the Microsoft Windows SDK for Windows 3.1x and the Microsoft Win32 SDK for Windows 9x. It was released on the market in 1999 and is the oldest SDK. The SDK platform includes builders, tools, documentation, header files, libraries and examples for software development on IA-32, x64 and IA-64 architectures. However, the .NET Framework SDK ships with the .NET Framework. Starting with Windows Vista, the SDK platform, SDK .NET Framework, SDK Tablet PC and Windows Media SDK are replaced with a new unified set called the Windows SDK. However, the .NET Framework 1.1 SDK is not included, because the .NET Framework 1.1 does not ship with Windows Vista. (Windows Media Center SDK for Windows Vista is available separately.) DirectX SDK is integrated into Windows 8 SDK and Windows SDK [3].

The Windows SDK allows the user to specify which components to install and where to install them. It integrates with Visual Studio, so several copies of the components that are both not loaded; however, there is a warning about compatibility, if one of them is not in the same period [4]. The information displayed can be filtered only by content, such as new content Windows Vista, .NET Framework only content or content for a particular language or technology.

After selecting the form factor and minimum SDK, you go to the basic activity selection screen shown in figure 2.12.

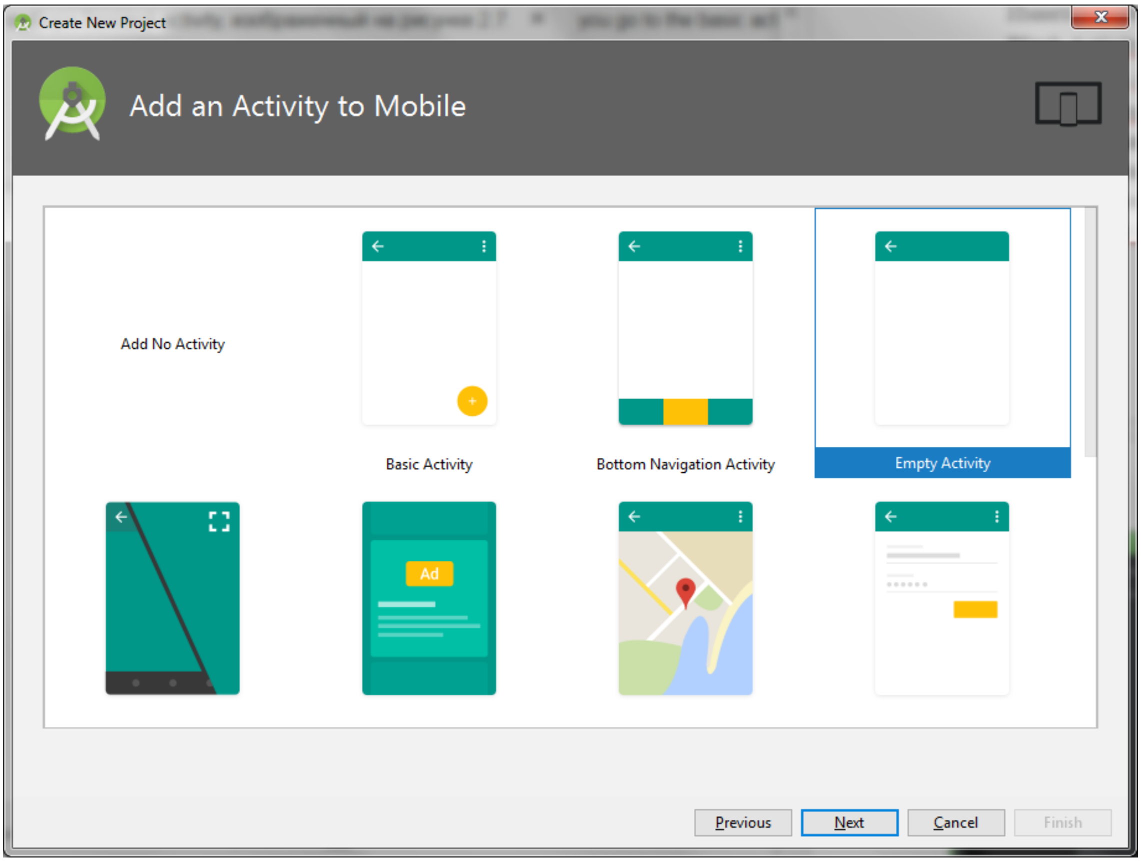
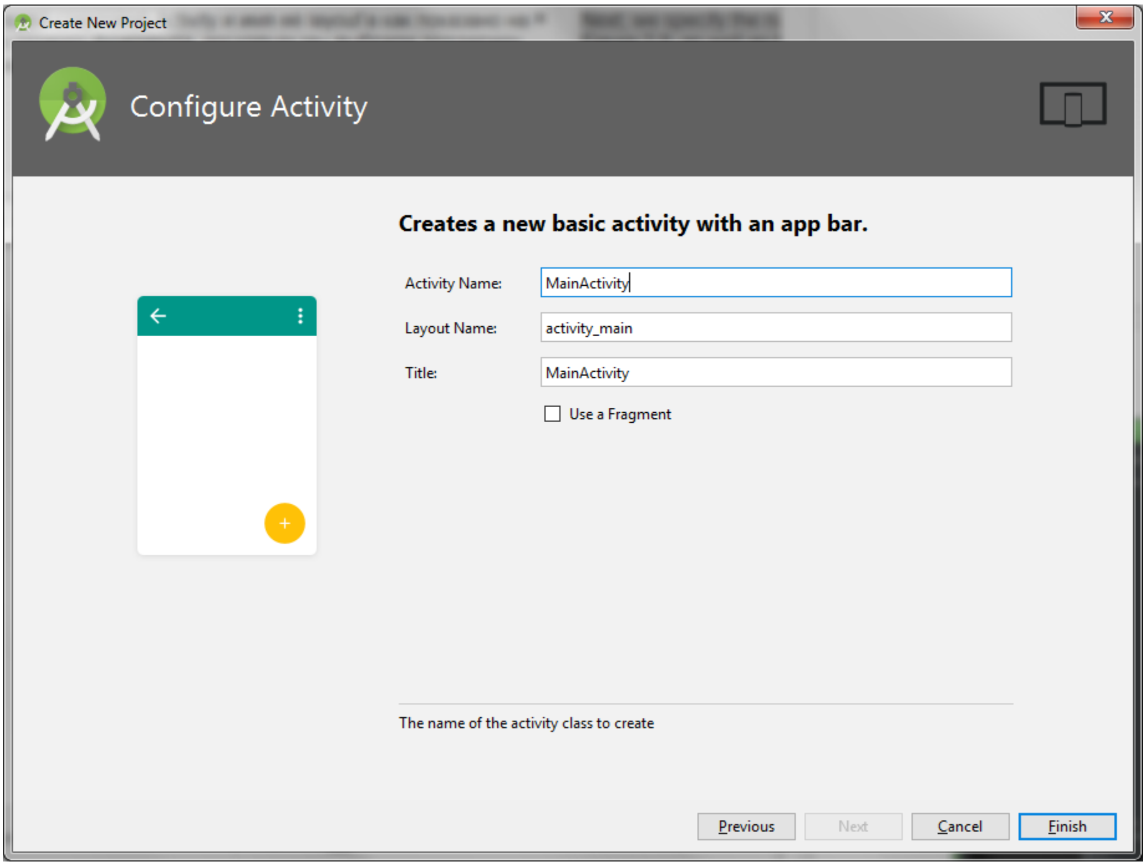


Figure 2.12 - The main activity selection window

There is a choice from a variety of options, for example:

- Blank Activity.
- Fullscreen Activity.
- Master/Detail Activity.

Next, we specify the name of our start Activity and the name of its layout's as shown in figure 2.13, as well as the name of the first fragment, as we chose to support the fragments when creating the project. You can also select the Activity navigation type.



Create New Project

Configure Activity

Creates a new basic activity with an app bar.

Activity Name: MainActivity

Layout Name: activity_main

Title: MainActivity

□ Use a Fragment

The name of the activity class to create

Figure 2.13 - Entering the name of our Start Activity and the name of its layout

After that, we are transferred to the window of the IDE itself. The first thing you should pay attention to is the structure of the project. And yes, it's not like Eclipse.

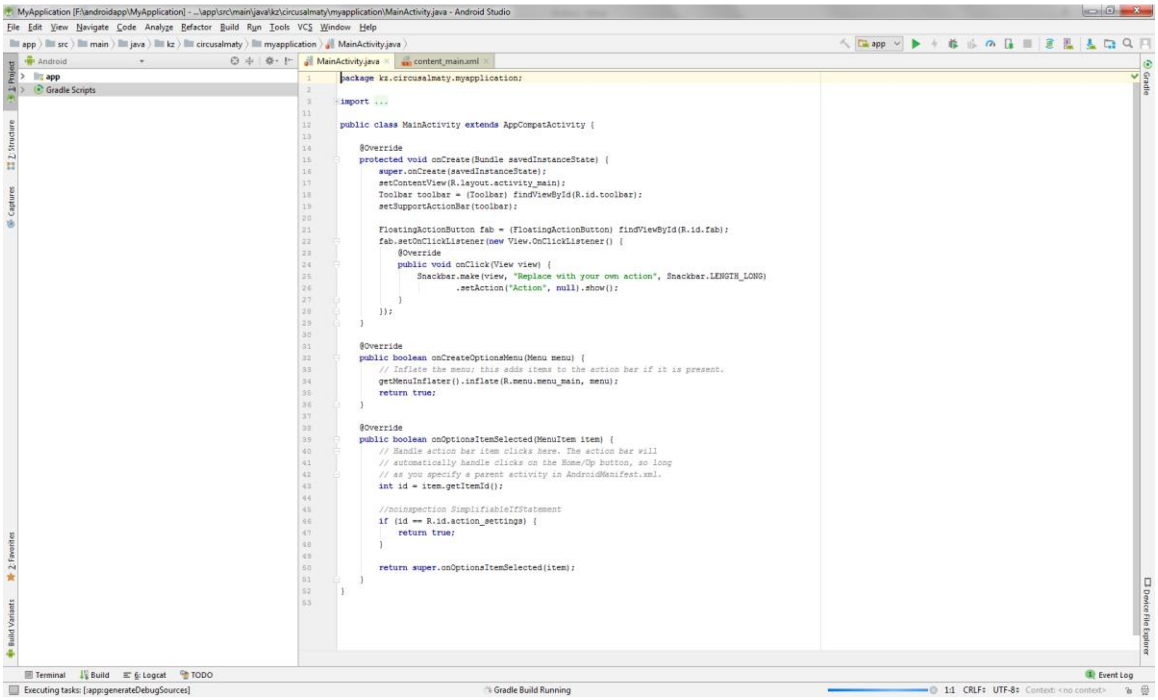
There are still familiar src and res folders, but res now lies inside the src folder, along with the new java folder into which our packages and classes migrated. This structure of the project Android Studio is due to the new system of project assembly - Gradle. It helps us manage dependencies in our project and connect external libraries, but it's not in this article that is more detailed about it.

It's worth mentioning that when you import projects with the usual structure, everything works just fine.

Figure 2.13 shows the main working window of the development environment.

The built-in profiling tools provide realtime statistics for your app’s CPU, memory, and network activity. Identify performance bottlenecks by recording method trace, inspecting the heap and allocations, and see incoming and outgoing network payloads.

Android Studio is a complete suite of software which includes an Integrated Development Environment (IDE) and the proper Android development kit (SDK) needed in order to create Android software.



MyApplication [F:\androidapp\MyApplication] - ...\app\src\main\java\kz\circusalmaty\myapplication\MainActivity.java - Android Studio

```
package kz.circusalmaty.myapplication;

import ...

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);

        FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.fab);
        fab.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Snackbar.make(view, "Replace with your own action", Snackbar.LENGTH_LONG)
                        .setAction("Action", null).show();
            }
        });
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is present.
        getMenuInflater().inflate(R.menu.menu_main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        // Handle action bar item clicks here. The action bar will
        // automatically handle clicks on the Home/Up button, so long
        // as you specify a parent activity in AndroidManifest.xml.
        int id = item.getItemId();

        //noinspection SimplifiableIfStatement
        if (id == R.id.action_settings) {
            return true;
        }

        return super.onOptionsItemSelected(item);
    }
}
```

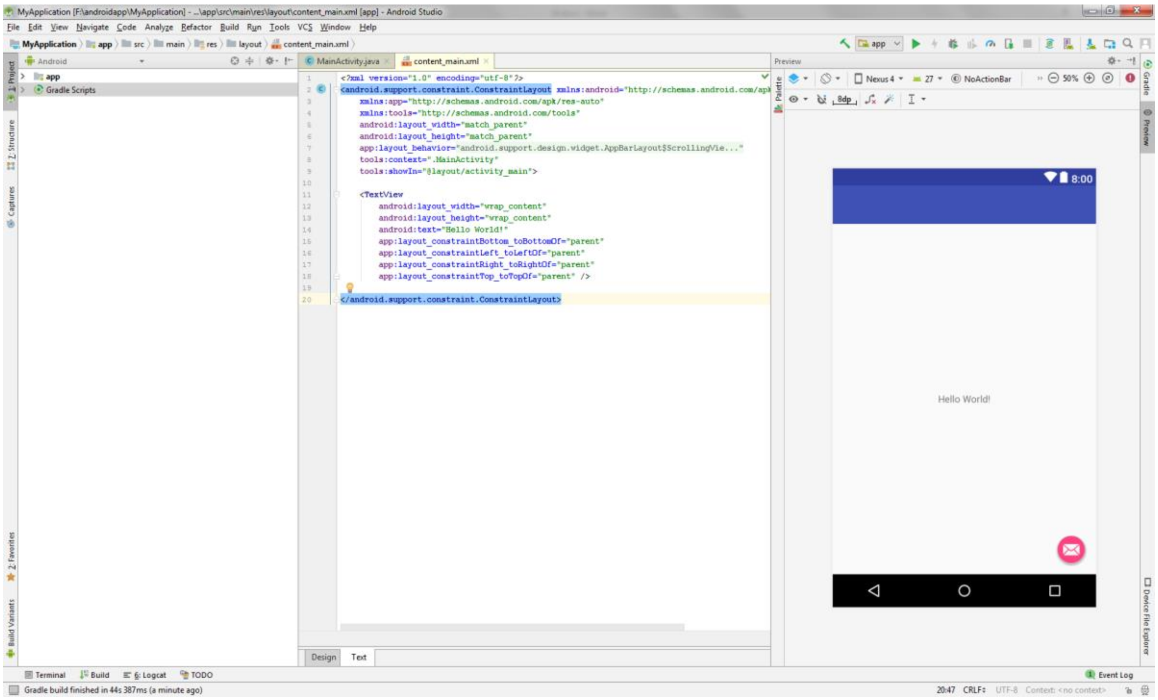
Executing tasks: [:app:generateDebugSources]

Gradle Build Running

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
app › src › main › java › kz › circusalmaty › myapplication › MainActivity.java
Android
app
Gradle Scripts
MainActivity.java
content_main.xml
1: Project
Z: Structure
Captures
2: Favorites
Build Variants
Gradle
Device File Explorer
Terminal
Build
6: Logcat
TODO
Event Log
1:1 CRLF: UTF-8: Context: <no context>

Figure 2.14 - The main IDE window

Picture below shows a working window of the xml development.



MyApplication [F:\androidapp\MyApplication] - ...\app\src\main\res\layout\content_main.xml [app] - Android Studio

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    app:layout_behavior="android.support.design.widget.AppBarLayout$ScrollingVie..."
    tools:context=".MainActivity"
    tools:showIn="@layout/activity_main">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</android.support.constraint.ConstraintLayout>
```

Gradle build finished in 44s 387ms (a minute ago)

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
MyApplication › app › src › main › res › layout › content_main.xml
Android
app
Gradle Scripts
MainActivity.java
content_main.xml
Preview
Palette
Nexus 4
27
NoActionBar
50%
8dp
8:00
Hello World!
Design
Text
Terminal
Build
6: Logcat
TODO
Event Log
20:47 CRLF: UTF-8 Context: <no context>

Figure 2.15 - Example of the xml window of the Rector

Having examined figure 2.15, the following can be noted:
- when editing xml in text mode, now there is also a preview;

- the specified colors and pictures used in the layout'e are displayed on the border in the form of small thumbnails, which easily help to understand what kind of resource you are using;

- when selecting a resource, its contents are displayed in pop-up JavaDocs, like @android: color / holo_green_dark;

- resources from dimens are automatically displayed by values, and when hovering you can always find out what kind of resource you are using;

- improved integration with Android components.

Let's try to add a new class. We get into the package where we want to place the class, and we press the wonderful combination alt-insert. I want to note that hotkey's in Android studio is sometimes quite difficult to remember, in comparison with Eclipse, but carry a much more complex and flexible functionality.

The studio offers us a choice of several objects to create. Briefly about each:

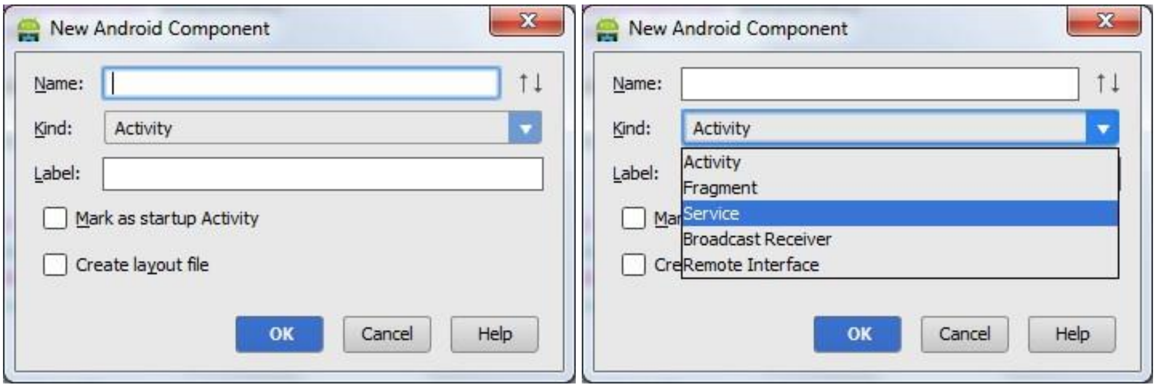
- Java Class is actually a Java Component. Allow you to create one of the main Java components: Class, Interface, Enum, Annotation and even Singleton.

- Module - creating, in fact, the module. A module is usually an auxiliary project in Android Studio. Modules in the project will be all external library projects (for example, ActionBarSherlock or Facebook SDK).

- File - a regular file of any actually with any resolution (txt, json, xml, etc.).

- Package - the package of our application.

Creating Android components. Figure 2.16 shows a window for creating a new application component.



New Android Component

Name:

Kind: Activity

Label:

☐ Mark as startup Activity

☐ Create layout file

Figure 2.16 - New component creation window

Activity - creates an Activity on one of the predefined templates, immediately registering it in the Manifest.

Android Component is a versatile thing that allows you to create any of the key components of our Android, immediately create a layout for it if necessary, and also label Activity as the starting one. By default, it registers the component in the Manifest, if necessary.

Package-info.java - a file describing the information about the package.

HTML File - actually creates an html file.

Also in the Android Studio is an advanced text editor, which has the following advantages:

- connecting lines between the beginning and end of if, while, switch constructs or methods.
- more intelligent code analysis.
- built-in Android Sources connection.
- the ability to inherit from the class, or create tests for it in 2 clicks.

Android consists of several dependent and required parts, including the following:

- in the Linux kernel operating system, which provides low-level interface to the hardware, memory management and processes, all optimized for mobile devices;
- libraries open source application development, including SQLite, WebKit, OpenGL and media manager;
- lead time is used to perform and publish Android apps, including the Dalvik virtual machine, and core libraries that provide specific functionality Android. Opening hours for the small and the coefficients to be used on mobile devices;
- availability structure application that exposes agnostically system application layer services, including window manager, content providers, location manager, telephony, and peer to peer network services;
- framework for user interface is used to host and run applications;
- preset program comes as part of the stack;
- software Development Kit to be used to build applications, including tools, plugins and documentation [6].

To develop and test applications required Android SDK-platform. In the start screen press the click Configure, and then select the SDK Manager, as shown in figure 2.17.

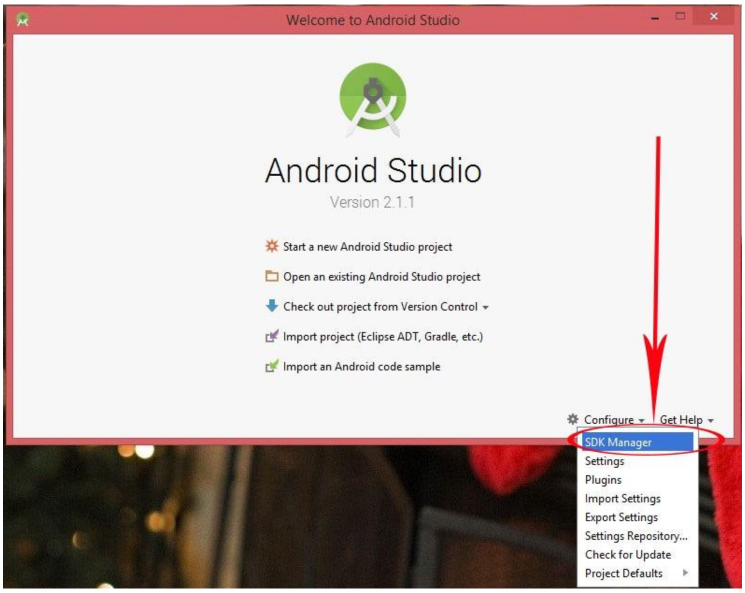

Welcome to Android Studio
Android Studio
Version 2.1.1
Start a new Android Studio project
Open an existing Android Studio project
Check out project from Version Control
Import project (Eclipse ADT, Gradle, etc.)
Import an Android code sample
Configure
Get Help
SDK Manager
Settings
Plugins
Import Settings
Export Settings
Settings Repository...
Check for Update
Project Defaults

Figure 2.17 - Configuring SDK Manager

Figure 2.18 shows the selection field of certain instruments, where we can also perform actions such as upload, update, etc.

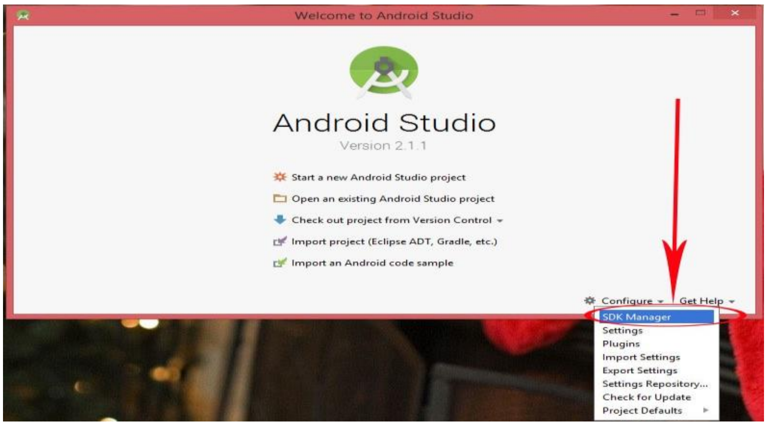

Welcome to Android Studio
Android Studio
Version 2.1.1
Start a new Android Studio project
Open an existing Android Studio project
Check out project from Version Control
Import project (Eclipse ADT, Gradle, etc.)
Import an Android code sample
Configure
Get Help
SDK Manager
Settings
Plugins
Import Settings
Export Settings
Settings Repository...
Check for Update
Project Defaults

Figure 2.18 - SDK Manager

In this menu, there are options such as download, update, or delete. There are folders, and certain versions of Android on which we want to write our application, then we choose the version that we need to develop. You can select all versions, depending on your laptop's operating system, you can also clean up and then he would not need to require in the future, as mentioned earlier. The whole process of creating and studying specific programming languages to create applications "Dobro" increasingly helped to online sources, video tutorials. Such as, the complete list of lessons studying Android [7].

2.2.3 The Java programming language

This first part is a step-by-step introduction to OOP using the Java language. The tutorial begins with an overview of the Java platform and language, followed by instructions for setting up a development environment consisting of a Java Development Kit (JDK) and the Eclipse IDE. After you're introduced to your development environment's components, you begin learning basic Java syntax hands-on.

Prerequisites: This tutorial is for software developers who are not yet experienced with Java code or the Java platform. The tutorial includes an overview of OOP concepts.

System requirements: To complete the exercises in this tutorial, you will install and set up a development environment consisting of:

JDK 8 from Oracle. Eclipse IDE for Java Developers. Download and installation instructions for both are included in the tutorial.

The recommended system configuration is:

- a system supporting Java SE 8 with at least 2GB of memory. Java 8 is supported on Linux®, Windows®, Solaris®, and Mac OS X;

- at least 200MB of disk space to install the software components and examples.

Java platform overview. Java technology is used to develop applications for a wide range of environments, from consumer devices to heterogeneous enterprise systems. In this section, get a high-level view of the Java platform and its components.

The Java language. Most Java developers constantly reference the official online Java API documentation — also called the Javadoc. By default, you see three panes in the Javadoc. The top-left pane shows all of the packages in the API, and the bottom-left pane shows the classes in each package. The main pane (to the right) shows details for the currently selected package or class. For example, if you click the java.utilpackage in the top-left pane and then click the ArrayList class listed below it, you see details about ArrayList in the right pane, including a description of what it does, how to use it, and its methods.

Like any programming language, the Java language has its own structure, syntax rules, and programming paradigm. The Java language's programming paradigm is based on the concept of OOP, which the language's features support.

The Java language is a C-language derivative, so its syntax rules look much like C's. For example, code blocks are modularized into methods and delimited by braces ({and}), and variables are declared before they are used.

Structurally, the Java language starts with *packages*. A package is the Java language's namespace mechanism. Within packages are classes, and within classes are methods, variables, constants, and more. You learn about the parts of the Java language in this tutorial.

The Java compiler: When you program for the Java platform, you write source code in .java files and then compile them. The compiler checks your code against the language's syntax rules, then writes out *bytecode* in .class files. Bytecode is a set of instructions targeted to run on a Java virtual machine (JVM). In adding this level of abstraction, the Java compiler differs from other language compilers, which write out instructions suitable for the CPU chipset the program will run on.

The JVM: At runtime, the JVM reads and interprets .class files and executes the program's instructions on the native hardware platform for which the JVM was written. The JVM interprets the bytecode just as a CPU would interpret assembly-language instructions. The difference is that the JVM is a piece of software written specifically for a particular platform. The JVM is the heart of the Java language's "write-once, run-anywhere" principle. Your code can run on any chipset for which a suitable JVM implementation is available. JVMs are available for major platforms like Linux and Windows, and subsets of the Java language have been implemented in JVMs for mobile phones and hobbyist chips.

The garbage collector: Rather than forcing you to keep up with memory allocation (or use a third-party library to do so), the Java platform provides memory management out of the box. When your Java application creates an object instance at runtime, the JVM automatically allocates memory space for that object from the *heap*— a pool of memory set aside for your program to use. The Java *garbage collector* runs in the background, keeping track of which objects the application no longer needs and reclaiming memory from them. This approach to memory handling is called *implicit memory management* because it doesn't require you to write any memory-handling code. Garbage collection is one of the essential features of Java platform performance.

The Java Development Kit: when you download a Java Development Kit (JDK), you get — in addition to the compiler and other tools — complete class libraries of prebuilt utilities that help you accomplish most common application-development tasks. The best way to get an idea of the scope of the JDK packages and libraries is to check out the JDK API documentation.

The Java Runtime Environment: The Java Runtime Environment (JRE; also known as the Java runtime) includes the JVM, code libraries, and components that are necessary for running programs that are written in the Java language. The JRE is available for multiple platforms. You can freely redistribute the JRE with your applications, according to the terms of the JRE license, to give the application's users a platform on which to run your software. The JRE is included in the JDK.

Setting up your Java development environment. In this section, you'll download and install the JDK and the current release of the Eclipse IDE, and you'll set up your Eclipse development environment.

If you already have the JDK and Eclipse IDE installed, you might want to skip to the "Getting started with Eclipse" section or to the one after that, "Object-oriented programming concepts."

Your development environment: The JDK includes a set of command-line tools for compiling and running your Java code, including a complete copy of the JRE. Although you can use these tools to develop your applications, most developers appreciate the additional functionality, task management, and visual interface of an IDE.

Eclipse is a popular open source IDE for Java development. Eclipse handles basic tasks, such as code compilation and debugging, so that you can focus on writing and testing code. In addition, you can use Eclipse to organize source code files into projects, compile and test those projects, and store project files in any number of source repositories. You need an installed JDK to use Eclipse for Java development. If you download one of the Eclipse bundles, it will come with the JDK already. Object-oriented languages follow a different programming pattern from structured programming languages like C and COBOL. The structured-programming paradigm is highly data oriented: You have data structures, and then program instructions act on that data. Object-oriented languages such as the Java language combine data and program instructions into *objects*.

An object is a self-contained entity that contains attributes and behavior, and nothing more. Instead of having a data structure with fields (attributes) and passing that structure around to all of the program logic that acts on it (behavior), in an object-oriented language, data and program logic are combined. This combination can occur at vastly different levels of granularity, from fine-grained objects such as aNumber, to coarse-grained objects, such as a FundsTransfer service in a large banking application.

Parent and child objects: A parent object is one that serves as the structural basis for deriving more-complex child objects. A child object looks like its parent but is more specialized. With the object-oriented paradigm, you can reuse the common attributes and behavior of the parent object, adding to its child objects attributes and behavior that differ.

Object communication and coordination: Objects talk to other objects by sending messages (method calls, in Java parlance). Furthermore, in an object-oriented application, program code coordinates the activities among objects to perform tasks within the context of the specific application domain.

Object summary. A well-written object:

- Has well-defined boundaries.
- Performs a finite set of activities.
- Knows only about its data and any other objects that it needs to accomplish its activities.

In essence, an object is a discrete entity that has only the necessary dependencies on other objects to perform its tasks.

It's time to see what a Java object looks like. Example: A person object. This first example is based on a common application-development scenario: an individual being represented by a Person object.

You know from the definition of an object that an object has two primary elements: attributes and behavior. Here's how these elements apply to the Person object.

As a rule of thumb, think of the attributes of an object as nouns and behavior as verbs.

Attributes (nouns)

What attributes can a person have? Some common ones include:

- Name.
- Age.
- Height.
- Weight.
- Eye color.
- Gender.

You can probably think of more (and you can always add more attributes later), but this list is a good start.

Behavior (verbs). An actual person can do all sorts of things, but object behaviors usually relate to application context of some kind. In a business-application context, for instance, you might want to ask your Person object, "What is your body mass index (BMI)?" In response, Person would use the values of its height and weight attributes to calculate the BMI.

More-complex logic can be hidden inside of the Person object, but for now, suppose that Person has the following behavior:

- calculate BMI;
- print all attributes;
- state and string.

State is an important concept in OOP. An object's state is represented at any moment in time by the values of its attributes.

In the case of Person, its state is defined by attributes such as name, age, height, and weight. If you wanted to present a list of several of those attributes, you might do so by using a String class, which you'll learn more about later.

Using the concepts of state and string together, you can say to Person, "Tell me all about you by giving me a listing (or String) of your attributes."

Principles of OOP. If you come from a structured-programming background, the OOP value proposition might not be clear yet. After all, the attributes of a person and any logic to retrieve (and convert) those values can be written in C or COBOL. The benefits of the OOP paradigm become clearer if you understand its defining principles: encapsulation, inheritance, and polymorphism.

Encapsulation. Recall that an object is above all discrete, or self-contained. This characteristic is the principle of encapsulation at work. Hiding is another term that's sometimes used to express the self-contained, protected nature of objects.

Regardless of terminology, what's important is that the object maintains a boundary between its state and behavior and the outside world. Like objects in the real world, objects used in computer programming have various types of relationships with different categories of objects in the applications that use them.

On the Java platform, you can use *access modifiers* (which you'll learn about later) to vary the nature of object relationships from *public* to *private*. Public access is wide open, whereas private access means the object's attributes are accessible only within the object itself.

The public/private boundary enforces the object-oriented principle of encapsulation. On the Java platform, you can vary the strength of that boundary on an object-by-object basis. Encapsulation is a powerful feature of the Java language.

Inheritance. In structured programming, it's common to copy a structure, give it a new name, and add or modify the attributes that make the new entity (such as an Account record) different from its original source. Over time, this approach generates a great deal of duplicated code, which can create maintenance issues.

OOP introduces the concept of inheritance, whereby specialized classes — without additional code — can "copy" the attributes and behavior of the source classes that they specialize. If some of those attributes or behaviors need to change, you override them. The only source code you change is the code needed for creating specialized classes. The source object is called the parent, and the new specialization is called the child— terms that you've already been introduced to.

Suppose that you're writing a human-resources application and want to use the Person class as the basis (also called the super class) for a new class called Employee. Being the child of Person, Employee would have all of the attributes of a Person class, along with additional ones, such as:

- taxpayer identification number;
- employee number;
- salary.

Inheritance makes it easy to create the new Employee class without needing to copy all of the Personcode manually.

Polymorphism. Polymorphism is a harder concept to grasp than encapsulation and inheritance. In essence, polymorphism means that objects that belong to the same branch of a hierarchy, when sent the same message (that is, when told to do the same thing), can manifest that behavior differently.

To understand how polymorphism applies to a business-application context, return to the Personexample. Remember telling Person to format its attributes into a String? Polymorphism makes it possible for Person to represent its attributes in various ways depending on the type of Person it is.

Polymorphism, one of the more complex concepts you'll encounter in OOP on the Java platform, is beyond the scope of this introductory tutorial. You'll explore encapsulation and inheritance in more depth in subsequent sections.

Not a purely object-oriented language. Two qualities differentiate the Java language from purely object-oriented languages such as Smalltalk. First, the Java language is a mixture of objects and primitive types. Second, with Java, you can write code that exposes the inner workings of one object to any other object that uses it.

The Java language does give you the tools necessary to follow sound OOP principles and produce sound object-oriented code. Because Java is not purely object oriented, you must exercise discipline in how you write code — the language doesn't force you to do the right thing, so you must do it yourself. You'll get tips in the "Writing good Java code" section.

Java is a programming language and software platform. It is a high-level object-oriented programming language. Java applications are also creating technology. Java helps reduce costs, implement innovation and increase the quality of service applications. The database server is the key to solving the problems of information management. In general, a server reliably manages a large amount of data in a multiuser environment so that many users can concurrently access the same data. All this is achieved while ensuring high productivity. The database server also prevents unauthorized access and provides efficient solutions for disaster recovery [8]. Without software, your computer will be useless, and it will 30 always be a black screen. Computer components are called hardware and program - software [9]. The logo of the application is shown below in figure 2.19.

Figure 2.19 - Java logo

Java technology is the byte code, that is, the platform is not stored in the machine code, but in a special code, regardless of platform. The machine language for the Java virtual machine is called Java byte code [10]. To this code could be executed on a specific processor is a special virtual machine java-JVM (Java Virtual Machine). JVM is a program that translates byte code into machine code for a particular processor. The platform includes the JVM and the application programming interface in Java (API), which is an extensive set of ready-made software components (classes) to facilitate the development and deployment of applets and applications. API Java covers many aspects of development in Java, including the manipulation of basic facilities, network programming, security, XML generation and Web-based services. API is organized as a set of libraries, called packets, which contain classes and interfaces for solutions related to each other problems.

In addition to the API, each complete implementation of the Java platform should include the following:

- Developer Tools for compiling, running, monitoring, debugging and documentation of applications;
- standard mechanisms of deploying applications in a customer environment;
- tools to create sophisticated graphical user interfaces;
- integration of software libraries for database access and manipulation of remote objects.

There are three versions of Java:

1) Java SE (Java Platform, Standard Edition). With it you can create and deploy Java-based applications to desktops and servers, as well as to develop 31 embedded software and real-time systems program. Edition Java SE includes classes needed to create a Web-services, and basic Java EE components.

2) Java EE (Java Platform, Enterprise Edition) - a corporate version of the platform that helps developers build and deploy portable, robust, scalable, and secure server-side applications in Java.

3) Java ME (Java Platform, Micro Edition), provides an environment for running applications developed for a wide range of mobile and embedded systems such as mobile phones, PDAs, set-top boxes and printers. This edition of the

platform is a means of creating flexible user interfaces, robust security model, a full range of built-in network protocols, as well as a strong support network and offline dynamically loaded applications.

Main features of Java:

- automatic memory management;
- enhanced exception handling;
- a rich set of filtering the input-output means;
- a set of standard collections: array, list, stack, etc.;
- the availability of simple tools for creating network applications;
- the availability of classes that allow you to perform HTTP-requests and process responses;
- built-in language tools for creating multi-threaded applications, which were later, ported to other languages;
- unified access to databases;
- parallel execution of programs.

2.2.4 Android Market

Google launched the Android Market way back in 2008. Just like the current Google Play Store, it allowed you to search for apps and download them to your device. As the popularity of Android smartphones grew, Google expanded its business by offering users more than just apps. You had the option of buying music, e-books, as well as movies.

At that point, the Android Market name didn't really make much sense anymore and caused quite a bit of confusion among some users. The reason was that you actually didn't have to own an Android device to buy something from Google's store and get access to various content. You could, for example, buy and download an e-book and just read it on your PC.

To avoid confusion, Google decided to rebrand the Android Market. In 2012, Google Play was launched. Along with the new name, the design of the store changed as well. The main idea behind the new look was to put more emphasis on other available content such as e-books, and music, and not just on apps. Whereas the Android Market had small tabs for e-books, music, and other content on top, Google Play made them bigger and placed them front and center, which is still the case today.

Additionally, the company also rebranded Google Music to Google Play Music, Google Books to Google Play Books, and Google Movies to Google Play Movies, essentially bringing all of the services under one name and making it clearer that you can access all of them from one place — the Google Play Store.

When Google Play launched, it made its way to Android smartphones running Android 2.2 version and higher. This means that devices with older versions of Google's operating system were left with the old Android Market, which was killed off at the end of last month, as already mentioned at the beginning of this post.

If you still have the Android Market on your device for whatever reason, it’s probably best if you upgrade it to the newer and much improved Google Play — provided you are running version or newer. As soon as you open the Android Market app, a window will pop up saying that a new version is available for download. Then just follow the directions on your screen, and in a minute or two the Android Market will be replaced by Google Play. If the update window doesn’t show up when you open the app, you’ll have to clear the cache of the Android Market.

Just open up the Settings menu, go to Apps or Application manager (this may differ depending on your device), and go to All apps. Then just find and select the Android Market app, and tap on “Clear cache”. After that, just open Android Market again on your device and the update window should appear.

When Google Play launched in 2012, it was a big improvement over Android Market, although it basically does the same thing. The online store has only gotten better over the years and is extremely simple to use with easy access to apps, movies, music, and more.

As the popularity of Android grew, so did the number of available apps. When Android Market was rebranded as Google Play, the store had around 700,000 apps available. Today, that number is close to three million!

Although Android Market still exists on older devices in certain cases, it will eventually be phased out as soon as the number of active users on certain versions of the OS decreases. In some cases, that might happen quite soon.

The worldwide smartphone market grew 1.1% year over year in 2016Q3, with 363.2 million shipments, according to data from the International Data Corporation (IDC) Worldwide Quarterly Mobile Phone Tracker. Shipments grew 5.2% QoQ, compared to the 345.5 million units in the second quarter of 2016. IDC expects to see a noticeable slowdown in smartphone shipments in 2016 with China showing a more mature growth pattern. Android dominated the market with 86.8% share in 2016Q3. Samsung continued its global leadership despite the Galaxy Note 7 recall.

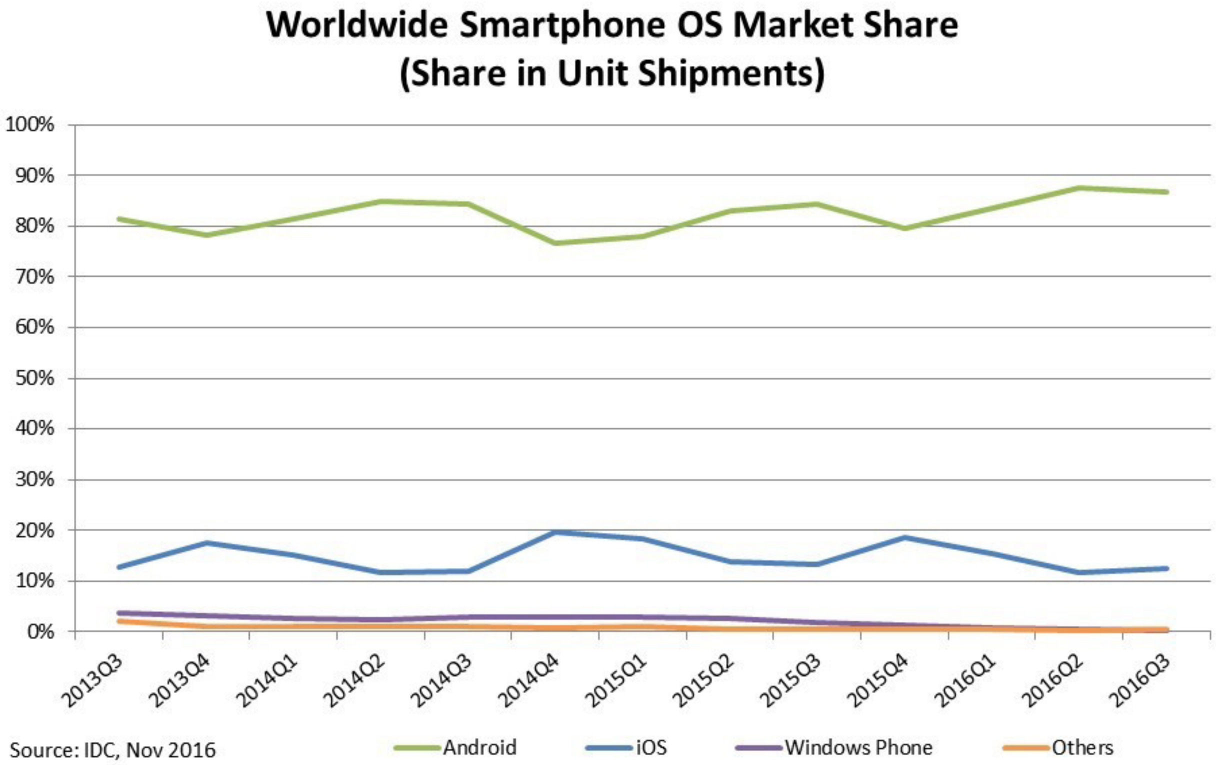


Figure 2.20 – Worldwide Smartphone OS Market Share

Android saw its market share for 2016Q3 grow by 12.7% QoQ with 45.5 million shipments as shown in figure 2.21 the Samsung S6 followed by its newest model, the Samsung s7 were the best-selling models this quarter. The Samsung S8 and 9 along with its older models accounted for 53.1% of the total Samsung shipments this quarter.

2.2.5 Technology review

This part describes the development of the mobile applications and principles of work and development of any kind of such application like this.

A design pattern is a common solution to a software problem. They are helpful for speeding up problem solving, ensuring that a developer doesn’t have to re-invent the wheel for every situation. They also give developers a common vocabulary with which to get across high-level ideas with minimal explanation and full understanding.

The Model-View-Controller design pattern assigns objects in an application one of three roles: model, view, or controller. The pattern defines not only the roles objects play in the application, it defines the way objects communicate with each other. Each of the three types of objects is separated from the others by abstract boundaries and communicates with objects of the other types across those boundaries. The collection of objects of a certain MVC type in an application is sometimes referred to as a layer — for example, model layer.

MVC is central pattern to a good design for a Cocoa application. The benefits of adopting this pattern are numerous. Many objects in these applications tend to be more reusable, and their interfaces tend to be better defined. Applications having an MVC design are also more easily extensible than other applications. Moreover, many Cocoa technologies and architectures are based on MVC and require that your custom objects play one of the MVC roles. The figure 2.21 shows the example of MVC design pattern.

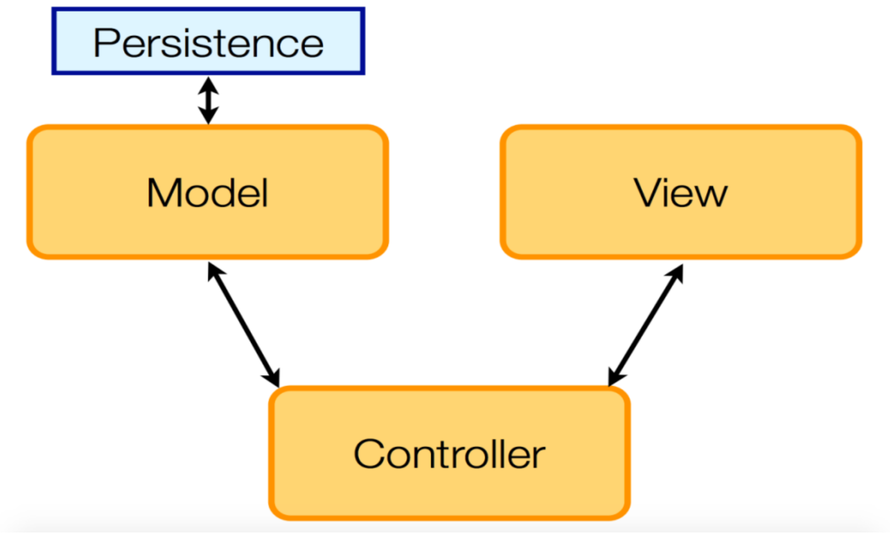


Figure 2.21 – MVC example

2.2.6 Models

Model objects encapsulate the data specific to an application and define the logic and computation that manipulate and process that data. For example, a model object might represent a character in a game or a contact in an address book. A model object can have to-one and to-many relationships with other model objects, and so sometimes the model layer of an application effectively is one or more object graphs. Much of the data that is part of the persistent state of the application (whether that persistent state is stored in files or databases) should reside in the model objects after the data is loaded into the application. Because model objects represent knowledge and expertise related to a specific problem domain, they can be reused in similar problem domains. Ideally, a model object should have no explicit connection to the view objects that present its data and allow users to edit that data — it should not be concerned with user-interface and presentation issues.

Communication: User actions in the view layer that create or modify data are communicated through a controller object and result in the creation or updating of a model object. When a model object changes (for example, new data is received over a network connection), it notifies a controller object, which updates the appropriate view objects.

2.2.7 Modeling functional software

Functional Modeling software – is the process of software project created in the application. That is, the Unified Modeling Language (UML - Unified Modeling Language) – a language for describing graphical object modeling in software development, systems engineering, and mapping organizational structures.

UML diagrams and all subsequent diagrams created on gliffy.com. Where to turn gliffy.com – is the creation of professional-quality flowcharts, wireframes, UML diagrams, etc.

The application "Dobro" used a menu item, which had the items and their attributes. Submission of application modeling "Dobro" is shown in figure 2.22.

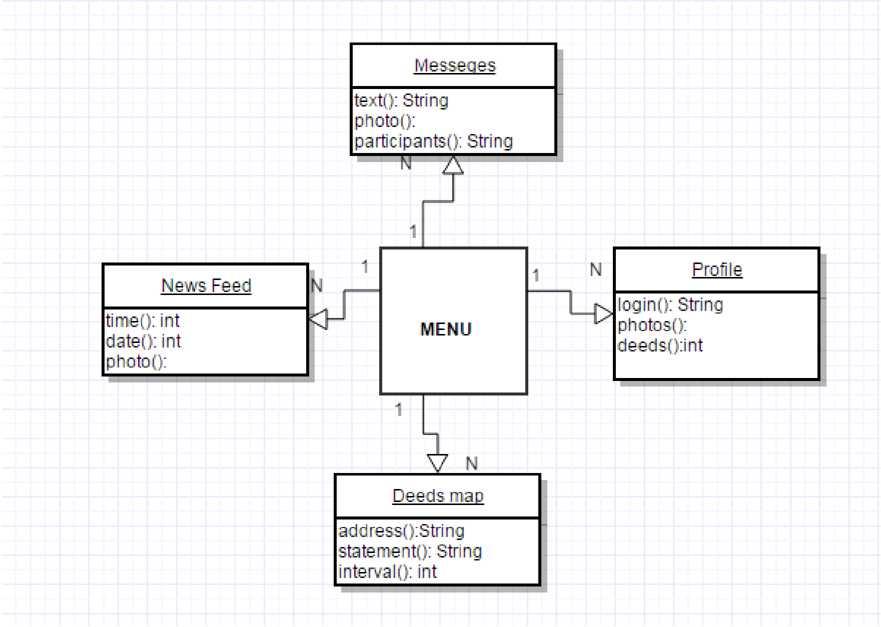


Figure 2.22 – Functional modeling software

The aim of the design is a detail of visible properties used in this application. Menu is the foundation, where it has such items as:

- Messages.
- News Feed.
- Profile.
- Deeds map.

2.2.8 The development of class diagrams

Class Diagram (Use Case - diagram) – a diagram which shows the classes and the relationships between them.

Application “Dobro” has 2 classes, namely:

- the administrator (Admin) – is the person who manages all the files;
- user – a person who can view the advertising and choose specific menu item. Working class diagram applications “Dobro” is shown in figure 2.23.

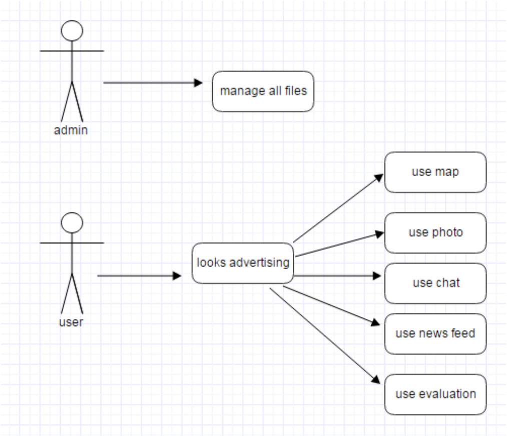


Figure 2.23 – Development of a class diagram

The Unified_Modeling_Language (UML) can help you model systems in various ways. One of the more popular types in UML is the class diagram. Popular among software engineers to document software architecture, class diagrams are a type of structure diagram because they describe what must be present in the system being modeled. No matter your level of familiarity with UML or class diagrams, our UML software is designed to be simple and easy to use.

UML was set up as a standardized model to describe an object-oriented programming approach. Since classes are the building block of objects, class diagrams are the building blocks of UML. The various components in a class diagram can represent the classes that will actually be programmed, the main objects, or the interactions between classes and objects.

The class shape itself consists of a rectangle with three rows. The top row contains the name of the class, the middle row contains the attributes of the class, and the bottom section expresses the methods or operations that the class may use. Classes and subclasses are grouped together to show the static relationship between each object.

The UML shape library in lucid chart can help you create nearly any custom class diagram using our UML diagram tool.

3 Development of mobile application "Dobro"

This section describes the process of creating an application “Dobro”.

3.1 The characteristics and capabilities Software

The application “Dobro” has several convenient features that allow the user to promptly obtain the necessary services.

3.1.1 User interface (UI)

First stage on mobile application development is User Interface. It refers to space and method to create as the junction between user and a product [14].

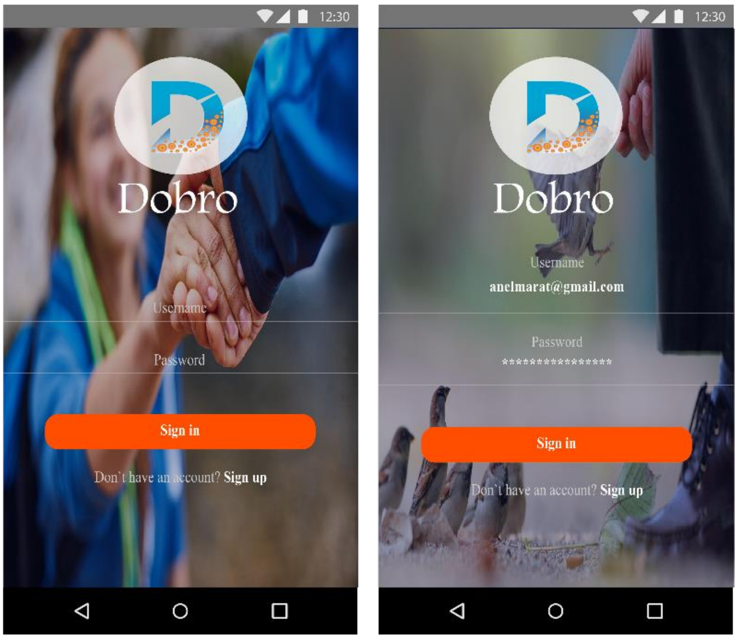
UI is a navigation, sub navigation, menus, the revealing lists, buttons, links, windows, the rounded angles, shadowing, error messages, warnings, updates, checkboxes, fields for password entry, a search box, a field of text input, a radio button, text areas, an induction status, a choice status, animation in case of swipe, scrolling, mouse clicks, an iconography, colors, lists, slide-show, the alternative text for the image, icons, notification messages, gradients, dialog boxes, a roundabout, OK/cancelling etc. [15].

Ideally application for the mobile device will work with thought speed. Moreover, the application interface will be clear even to the beginner. The goal is to facilitate effective operation of the product [16].

Chris Mears from The UX Review gave us this piece of advice on designing for mobile: “One of the main use cases for mobile is killing time. But that doesn’t mean you should waste that of your users. Make sure you understand the main tasks they want to accomplish on your app through research and make those the focus of the interface” [17].

The second thing after prototyping stage on Justinmind is UI design of application. Like graphic designers, UI designers also need to have a strong grasp of general design fundamentals like color theory and typography. UI designers do most of their work in Photoshop, Illustrator, and InDesign. But they also do a lot of work in wireframing software that allows them to show clients and team members how their design will work in practice [18].

The UI design created on Adobe Photoshop program. First trying work on Photoshop is the sign in and signs up page and the first logo of application on Illustration. There are screenshots of first trying version results for Android platform:



12:30

Dobro

Username

Password

Sign in

Don't have an account? Sign up

12:30

Dobro

Username

anelmarat@gmail.com

Password

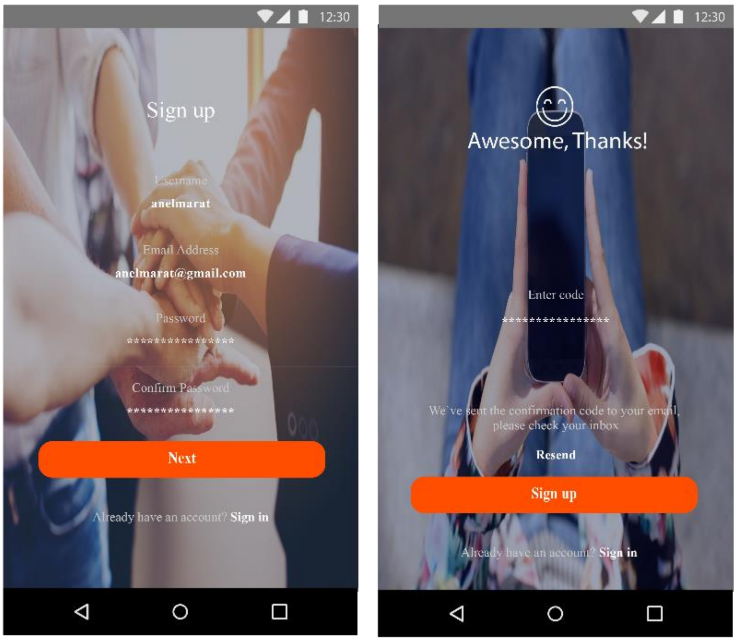
****

Sign in

Don't have an account? Sign up

Figure 3.1 – The Sign in page

The first page when user open the application is sign in page as shown in figure 3.1 and figure 3.2 The user can sign in or if he did not sign up, he can register through sign up button.



12:30

Sign up

Username

anelmarat

Email Address

anelmarat@gmail.com

Password

****

Confirm Password

****

Next

Already have an account? Sign in

12:30

Awesome, Thanks!

Enter code

****

We've sent the confirmation code to your email, please check your inbox

Resend

Sign up

Already have an account? Sign in

Figure 3.2 – The Sign up page

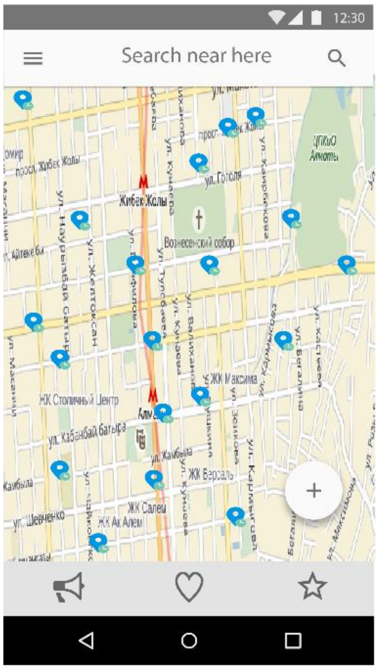


Figure 3.3 – The main - map page

There is, figure 3.3, a main page of application opened after sign in process, first in the head menu bar located hamburger-menu icon and the search engine icon on grey color. Below there is a map of Almaty city and above the map there are map pointers on blue color. In the right corner bellow the map there is located an add icon in a circle with the shadow effect for adding own help request. Last is button menu bar: news feed, feedback and rating.

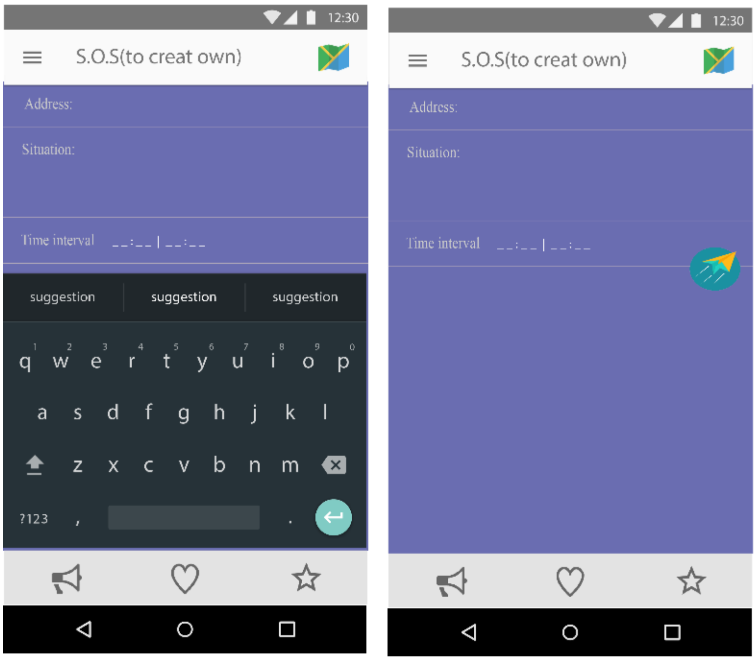
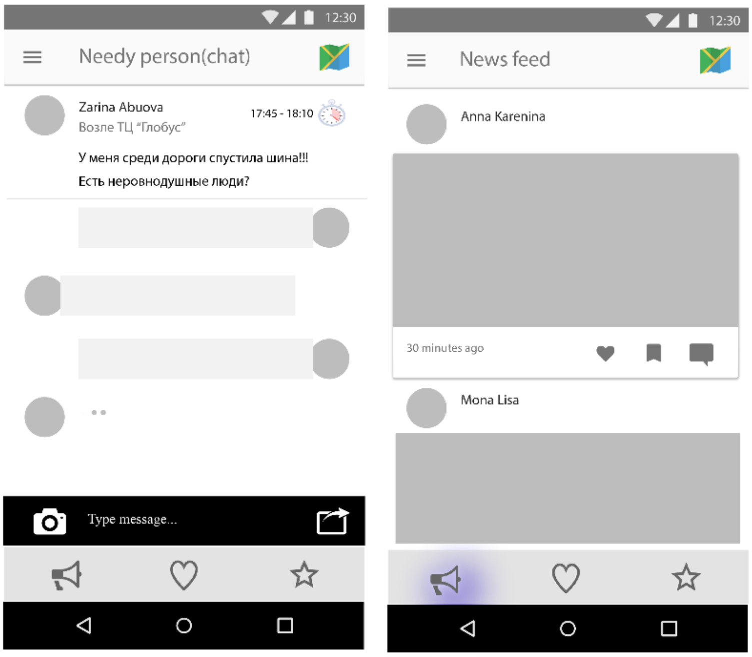


Figure 3.4 – The new request adding page

On the right corner of the head menu bar there is located a map icon for the map-main page link. The new request should be done through the forms: Address, Time interval and Situation description. On the right corner below the time interval form there is a send icon as shown in figure 3.4.



Zarina Abuova
Возле ТЦ "Глобус"
17:45 - 18:10

У меня среди дороги спустила шина!!!
Есть неровнодушные люди?

Figure 3.5 – The message and news feed pages

In the figure 3.5, there is shown the user image and near it like or comment icons with quantity or comment descriptions. The next window, figure 3.6, shows the rating of users who did a lot of good deeds.

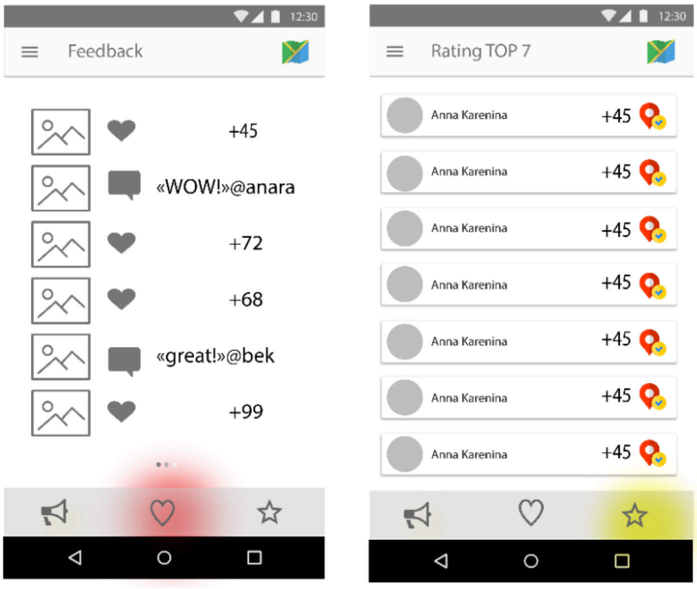


Figure 3.6 – The Feedback and Rating pages

There are also user profile images and the login names with quantity of good deeds and map pointer-done icons. (figure 3.7)

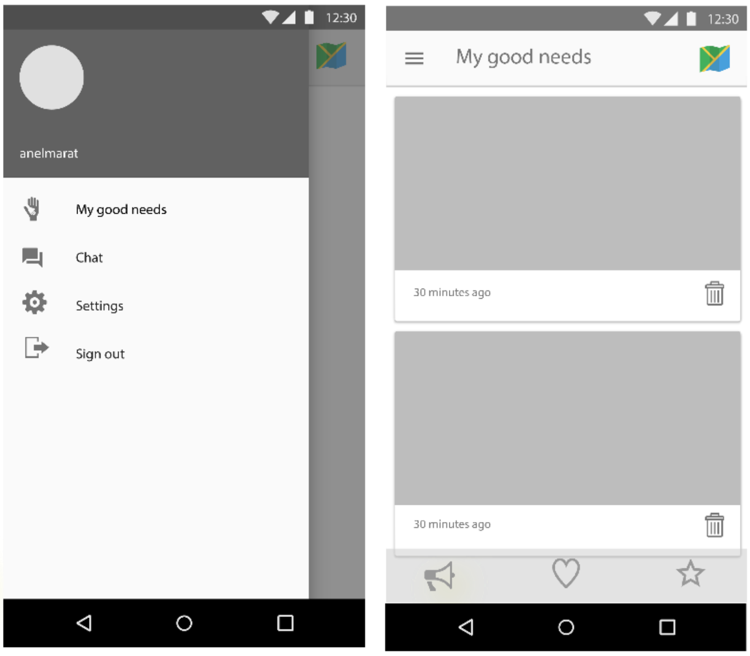
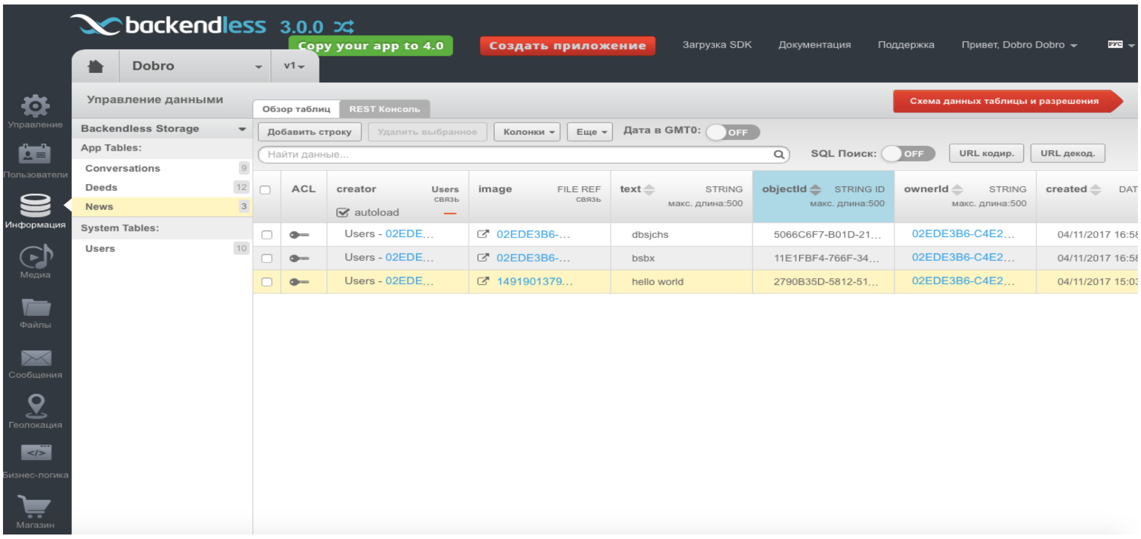


Figure 3.7 – Menu bar and user good deeds

The first feedback from users shows us a negative result, moreover the first UI for our application was created like prototype and the redesign version of application UI created for iOS platform with new logo and new interface design.

3.1.2 Backendless

Backendless is a leading mBaaS and powerful Mobile Application Development Platform that enables rapid development of feature-rich mobile, desktop and browser-based applications. Backendless provides a reliable and secure server infrastructure, broad server-side functionality of data persistence, user management, publish/subscribe messaging and push notifications, media streaming, geolocation and other mobile backend services. Backendless offers a variety of unique features: cross-platform data interchange, unlimited API calls, client-side debugging of custom business logic, client code generation, geofencing, centralized logging, and multi-tier security as shown in figure 3.8 Furthermore, absolutely all default API behavior can be modified or extended with custom business logic. The platform is available as an Online service with a free registration and a generous free plan and as an on-premise version which can run on a developer’s own machine, company’s data center or a private cloud [19].



□	ACL	creator Users связь ☑ autoload	image FILE REF связь	text STRING макс. длина:500	objectId STRING ID макс. длина:500	ownerId STRING макс. длина:500	created DAT
□		Users - 02EDE...	02EDE3B6-...	dbsjchs	5066C6F7-B01D-21...	02EDE3B6-C4E2...	04/11/2017 16:5
□		Users - 02EDE...	02EDE3B6-...	bsbx	11E1FBF4-766F-34...	02EDE3B6-C4E2...	04/11/2017 16:5
□		Users - 02EDE...	1491901379...	hello world	2790B35D-5812-51...	02EDE3B6-C4E2...	04/11/2017 15:0

Figure 3.8 – Backendless (Dobro)

3.1.3 Firebase

At the I/O Conference in May, Google unveiled the latest iteration of Firebase – a software development platform that reduces the hassle of establishing proper infrastructure, provides analytics about user behavior, and facilitates rapid application deployment.

Firebase isn't new. It was originally an independent company that Google acquired in 2014. Since then, the company has put its employees to work to establish Firebase as a branded Google property that will give other deployment platforms, such as Amazon Web Services, stiff competition.

That's because Firebase not only provides users with a software developer's kit (SDK), but it also gives them the infrastructure they need to quickly produce robust solutions.

As a bonus, much of the features of Firebase are free. Even better: Google promises to make those services free *for life*. That's also seen as shot across the bow to Amazon, which currently offers a single free EC2 instance to developers, but just for just one year.

Multiple Platforms Supported. With Firebase, Google isn't just offering free technical goodies to the development community; it's also delivering a solution that works with multiple platforms.

Firebase is a development framework that supports web applications, iOS applications, and, unsurprisingly, Android applications. That means developers not only have access to a broad spectrum of tools to make their lives easier, they also have the ability to use Firebase to produce applications that run on multiple operating systems.

That's going to save some companies big bucks. It can cost a fortune to hire a development team to produce an app that runs on both iOS and Android. That's because the skill set required for the two technologies is noticeably different. The cost would increase even further for companies that want a web solution similar to their mobile apps, because once again a different skill set is involved.

All those different development streams result in an IT expense that's too high for a lot of struggling startups. It's likely, then, that many small businesses will look to Firebase as a cost-effective solution for delivering apps.

Analytics. There must have been some clamoring from the development community for more analytics in apps because Google goes out of its way to make the case that Firebase has your back if you're interested in app analytics. It's the very first feature listed on the "Features" page of Firebase and Google claims that Firebase Analytics is the "heart" of the Firebase platform.

If you're familiar with Google Analytics, then the learning curve for Firebase Analytics should be pretty small. It's not only built by the same company, it also presents itself with a similar user-interface that's fairly similar to Google Analytics.

The biggest difference between Google Analytics and Firebase Analytics is the metrics that are tracked. Instead of page views, impressions, and session lengths, developers will see how many users downloaded the app and their behavior within it. That way, designers can improve their apps to enhance the user experience and, of course, increase conversions.

Firebase Analytics provides developers with access to more than 500 metrics and enables audience segmentation based on devices, events, and other user properties.

Cloud Messaging. One of the reasons that you know Google is taking the Firebase brand name seriously is because the company actually renamed Google Cloud Messaging to Firebase Cloud Messaging.

That's right. Google took away its own name from one of its own properties.

FCM is a cross-platform messaging solution that enables developers to deliver messages to client apps. Even better: it's offered by Google at no cost.

Authentication. It's likely that an app will need to know the identity of its user. Fortunately, Firebase offers a solution for that.

Firebase Authentication delivers a suite of tools that make it easy for developers to authenticate and authorize users. It also offers single sign-on support with identity providers like Facebook, Twitter, and, obviously, Google.

The Firebase authentication solution also integrates with well-known security standards, such as OpenID Connect and OAuth 2.0.

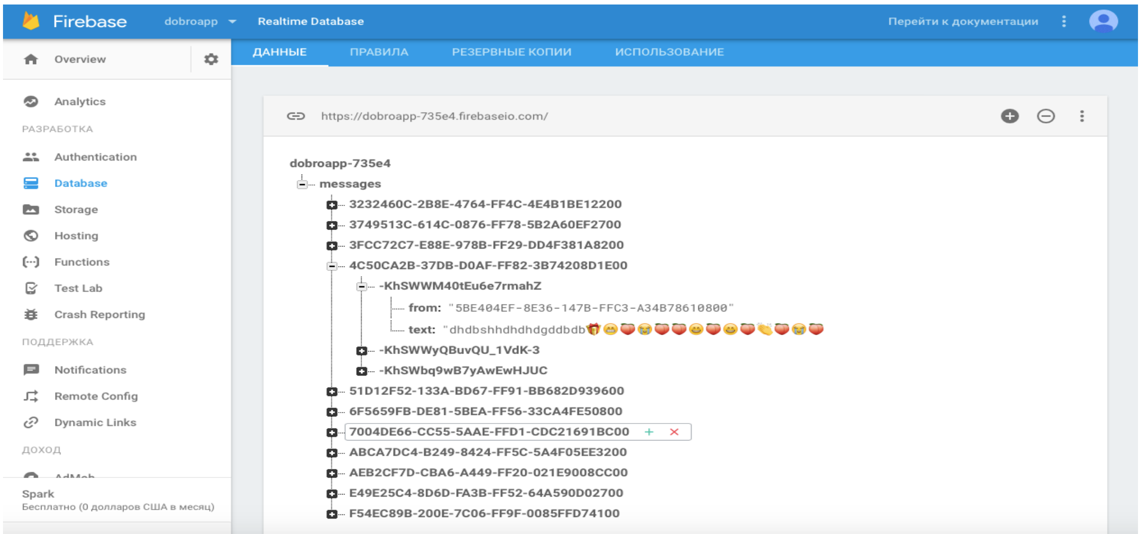
Database. Developers who've struggled with finding a solution for persisting user data will appreciate the Firebase Realtime Database.

It's a cloud-based NoSQL database solution. Data is stored in JavaScript Object Notation (JSON) format for easy retrieval without the complicated hassles of inner and outer joins associated with SQL queries.

The Firebase Realtime Database also performs up-to-the-milisecond synchronization. That means when data changes, all devices running the app immediately receive the update. It's a solution that gives users in different geographic regions the opportunity to communicate, collaborate, and coordinate more effectively.

At the heart of Firebase is Firebase Analytics, a free and unlimited analytics solution. See user behavior and measure attribution from a single dashboard (figure 3.9):

- unlimited reporting of 500 event types, each with up to 25 attributes;
- one dashboard to view user behavior and cross-network campaign performance;
- demographic segmentation, including age, gender, and location, available out of-the-box;
- export raw data to BigQuery for custom querying;
- the Firebase Analytics Partner database includes leading mobile advertising technology platforms, which have been validated to measure and optimize app campaign performance.



```
https://dobroapp-735e4.firebaseio.com/

dobroapp-735e4
  messages
    3232460C-2B8E-4764-FF4C-4E4B1BE12200
    3749513C-614C-0876-FF78-5B2A60EF2700
    3FCC72C7-E88E-978B-FF29-DD4F381A8200
    4C50CA2B-37DB-D0AF-FF82-3B74208D1E00
      -KhSWWM40tEu6e7rmahZ
        from: "5BE404EF-8E36-147B-FFC3-A34B78610800"
        text: "dhdbshhdhdhdgddbdb🎁😬🍑😭🍑🍑😄🍑😄🍑👏🍑😭🍑
      -KhSWWyQBuvQU_1VdK-3
      -KhSWbq9wB7yAwEwHJUC
    51D12F52-133A-BD67-FF91-BB682D939600
    6F5659FB-DE81-5BEA-FF56-33CA4FE50800
    7004DE66-CC55-5AAE-FFD1-CDC21691BC00
    ABCA7DC4-B249-8424-FF5C-5A4F05EE3200
    AEB2CF7D-CBA6-A449-FF20-021E9008CC00
    E49E25C4-8D6D-FA3B-FF52-64A590D02700
    F54EC89B-200E-7C06-FF9F-0085FFD74100
```

Figure 3.9 - Firebase (Dobro)

3.2 Analysis of the mobile application “Dobro” for promoting kindness

3.2.1 End - User - License Agreement

These Rules are an official document of individual developer "Raimbek Rakhmetov" (here in after the application Administration) and determine the method of processing and protecting information on the individuals using DOBRO mobile application (here in after the application) and its services (here in after the Users).

Please read this End-User License Agreement carefully before registering in application. By registering, you are agreeing to be bound by the terms and conditions of this Agreement. If you do not agree to the terms of this Agreement, do not register.

License: Application Administration grants you a revocable, non-exclusive, non-transferable, limited license to download, install and use the Application solely for your personal, non-commercial purposes strictly in accordance with the terms of this Agreement.

Modifications to Application. Application Administration reserves the right to modify, suspend or discontinue, temporarily or permanently, the Application or any service to which it connects, with or without notice and without liability to you.

Terms and conditions. By making available the application and its services, the application administration, acting reasonably and in good faith, believes that the User:

- has all necessary rights to register on and use this application;
- provides true information about himself (herself) to the extent necessary for use of the application Services;
- understands that the personal information posted by the User may become available to other application Users, be copied and disseminated by such users;
- understands that some types of information transferred by him / her to other Users cannot be deleted by the User himself / herself;
- is aware of and accepts these Rules and assumes the rights and obligations provided by these Rules.

The Application Administration does not check the user information received (collected).

Users' information protection measures. The application Administration takes technical, organizational and legal measures to ensure that the User’s personal data are protected from unauthorized or accidental access, deletion, modification, blocking, copying, dissemination as well as from other unauthorized actions.

Access to the application is authorized using the User’s login (e-mail) and password. The User is responsible for keeping this information confidential. The User may not transfer his / her login and password to third parties and is also obliged to take measures for keeping them confidential.

Purposes of information processing. The application Administration processes information on the Users, including their personal data, in order to fulfill the application Administration’s obligations to the Users regarding use of the application and its services.

Term and Termination. This Agreement shall remain in effect until terminated by you or application. Application Administration may, in its sole discretion, at any time and for any or no reason, suspend or terminate this Agreement with or without prior notice. This Agreement will terminate immediately, without prior notice from Application administration, in the event that you fail to comply with any provision of this Agreement. You may also terminate this Agreement by deleting the Application and all copies thereof from your mobile device. Upon termination of this Agreement, you shall cease all use of the Application and delete all copies of the Application from your mobile device.

Severability. If any provision of this Agreement is held to be unenforceable or invalid, such provision will be changed and interpreted to accomplish the objectives of such provision to the greatest extent possible under applicable law and the remaining provisions will continue in full force and effect.

Amendments to this Agreement. My Company DOBRO reserves the right, at its sole discretion, to modify or replace this Agreement at any time. If a revision is material, we will provide at least 30 days' notice prior to any new terms taking effect. What constitutes a material change will be determined at our sole discretion.

Contact Information. If you have any questions about this Agreement, please contact us. raimbek.rakhmetov@gmail.com

3.2.2 Results and statistics on Android OS

How to view application statistics?

Play Console. The statistics for each application can be viewed in the mobile and web versions. See statistics. Available reports. With reports in the Play Console, you can see the number of installs and deletions of applications, its estimates, and its revenue and crash data.

- in the Control Panel, you will find a brief overview of the key application indicators;
- for more information, see the Statistics page.

Control Panel. Here's how to familiarize you with the key indicators:

- go to the Play Console;
- select the application;
- in the left menu, click Control Panel. The data for a certain period of time will be displayed. To learn more about a particular parameter, click Details.

Note. To learn more about a particular metric, hover your mouse over the icon.

Statistics. On the Statistics page, you can compare metrics, select date ranges, and view data based on various parameters.

How to set up a report

a) go to the Play Console;

b) select the application;

c) in the new menu, click Statistics;

d) choose a date range. To compare the data in two time intervals, check the "Compare data for different periods" box and select the second range;

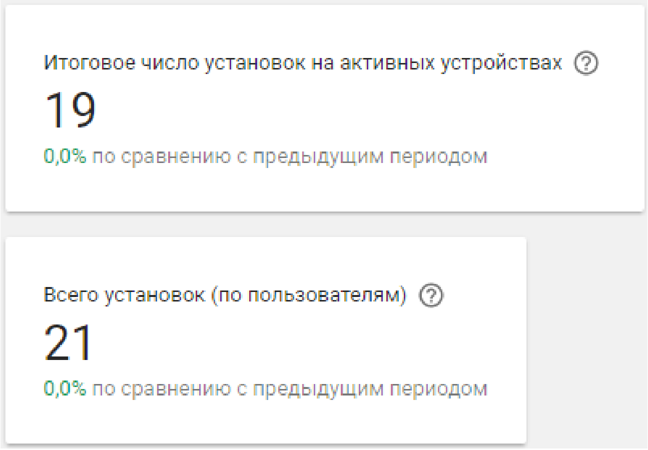
e) at the top of the screen, click Configure report. You can:

1) compare indicators (no more than two). Some of them cannot be selected at the same time;

2) view the data for the parameter, the pointer of the desired parameter on the right side of the screen;

f) click Finish;

In the figure 3.10, we can see the statistics of a mobile application “Dobro” for whole testing period.



Итоговое число установок на активных устройствах ?

19

0,0% по сравнению с предыдущим периодом

Всего установок (по пользователям) ?

21

0,0% по сравнению с предыдущим периодом

Figure 3.10 – The Android OS feedback statistics

The total number of settings on active devices is 19 downloads. The total number of installations (by users) is 21 (Figure – 3.11).

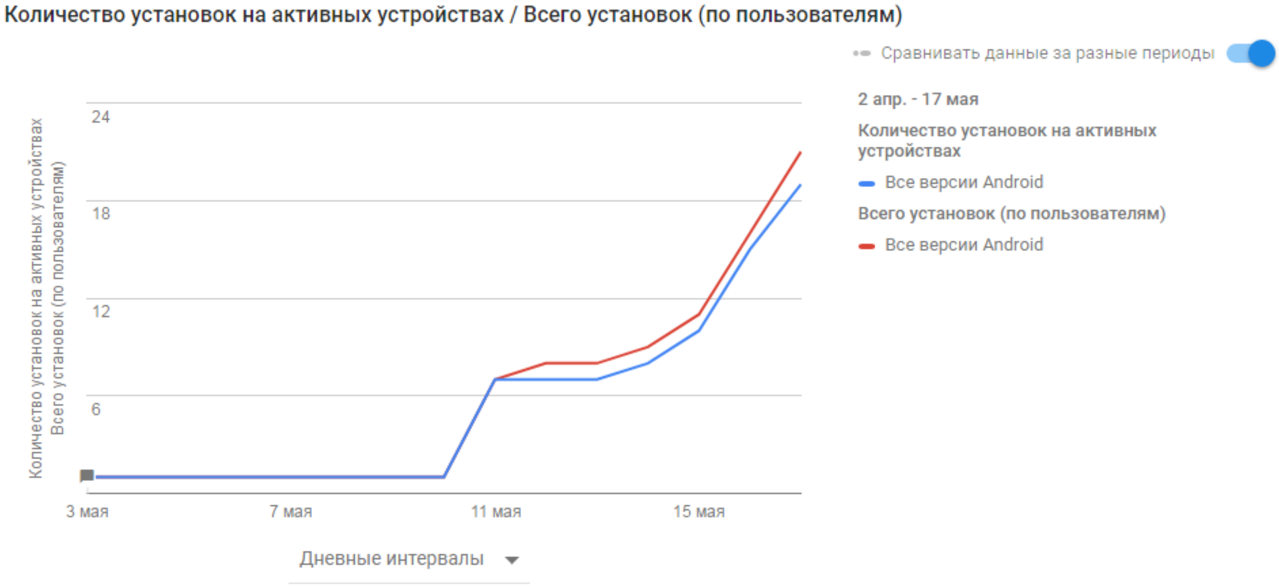
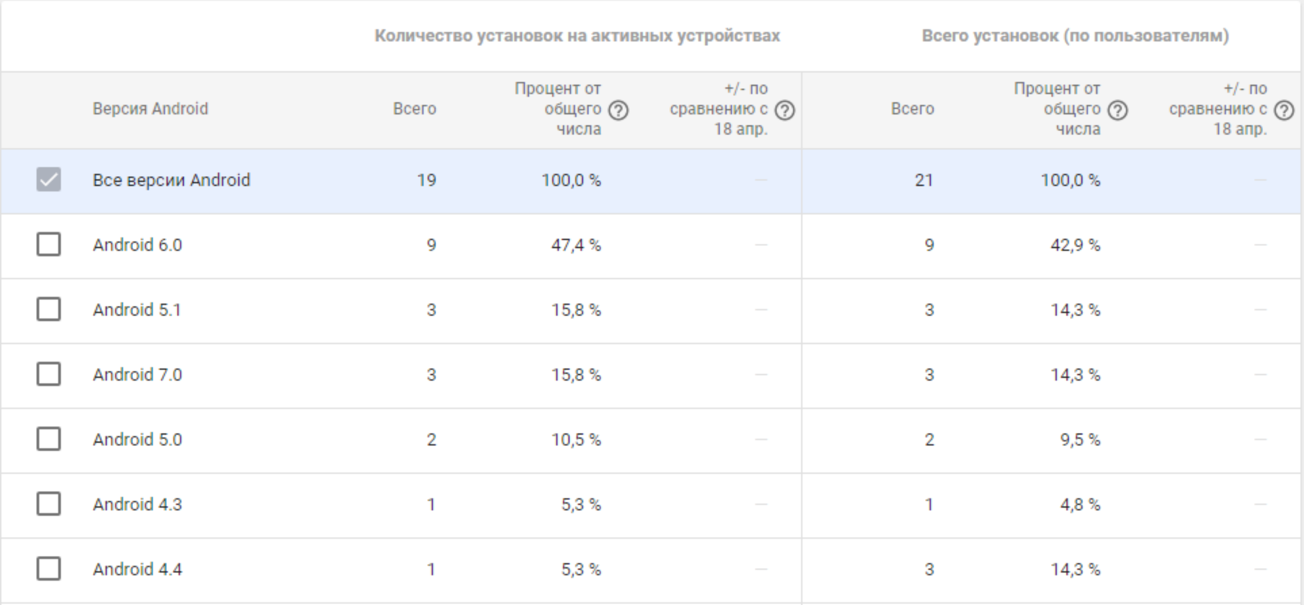


Figure 3.11 – The total number of installations

The number of installations on active devices 19, with versions of Android 6.0 - 9, accounting for 47.7%; Android 5.1 and Android 7.0-3, accounting for 15.8%; Android 5.0 -2, accounting for 10.5%; Android 4.3 and Android 4.4 -1, accounting for 5.3% (Figure 3.12).



		Количество установок на активных устройствах			Всего установок (по пользователям)		
	Версия Android	Всего	Процент от общего числа	+/- по сравнению с 18 апр.	Всего	Процент от общего числа	+/- по сравнению с 18 апр.
☑	Все версии Android	19	100,0 %	—	21	100,0 %	—
□	Android 6.0	9	47,4 %	—	9	42,9 %	—
□	Android 5.1	3	15,8 %	—	3	14,3 %	—
□	Android 7.0	3	15,8 %	—	3	14,3 %	—
□	Android 5.0	2	10,5 %	—	2	9,5 %	—
□	Android 4.3	1	5,3 %	—	1	4,8 %	—
□	Android 4.4	1	5,3 %	—	3	14,3 %	—

Figure 3.12 – Results by Android OS versions

The number of viewers was 8, and 75.0% of them installed the application (Figure 3.13).

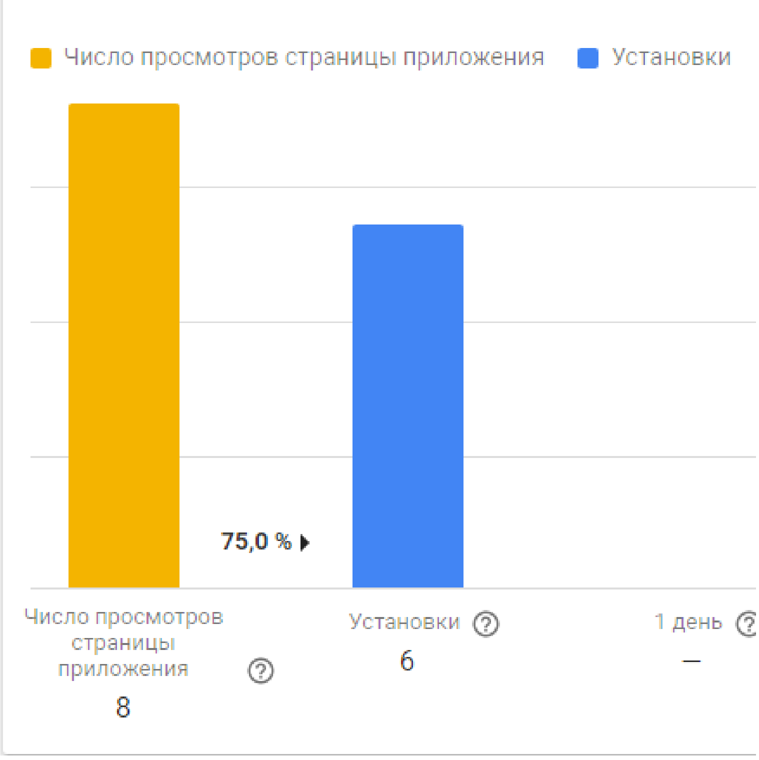
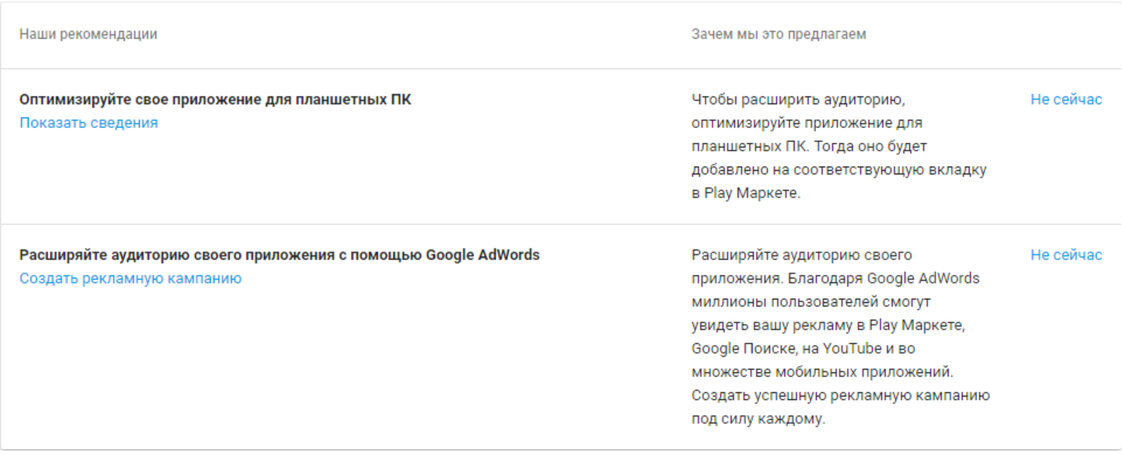


Figure 3.13 – Diagram of views and installers

Google play Console offers us a number of recommendations:

1) optimize the application for tablet PCs to expand the audience. Then it will be added to the appropriate tab in the Play Store;

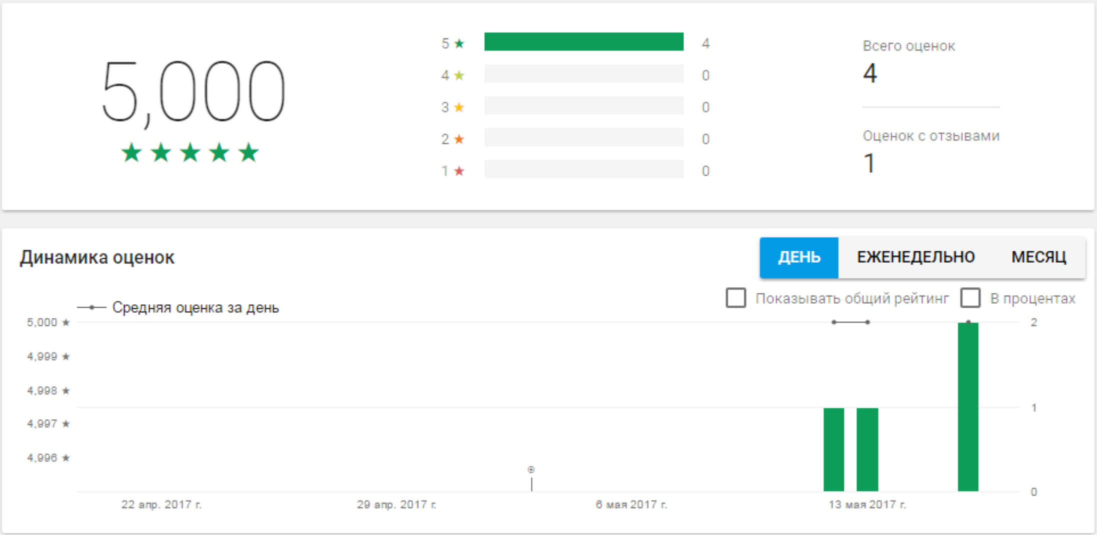
2) to expand the audience of your application using Google AdWords, just to expand the audience (Figure 3.14).



Наши рекомендации	Зачем мы это предлагаем	
Оптимизируйте свое приложение для планшетных ПК Показать сведения	Чтобы расширить аудиторию, оптимизируйте приложение для планшетных ПК. Тогда оно будет добавлено на соответствующую вкладку в Play Маркете.	Не сейчас
Расширяйте аудиторию своего приложения с помощью Google AdWords Создать рекламную кампанию	Расширяйте аудиторию своего приложения. Благодаря Google AdWords миллионы пользователей смогут увидеть вашу рекламу в Play Маркете, Google Поиске, на YouTube и во множестве мобильных приложений. Создать успешную рекламную кампанию под силу каждому.	Не сейчас

Figure 3.14 – The Google play recommendation

In the following figure 3.15 shown the dynamics of assessments. There are four excellent marks and an excellent mark with comment.



5,000
★★★★★

5 ★	4
4 ★	0
3 ★	0
2 ★	0
1 ★	0

Всего оценок
4

Оценок с отзывами
1

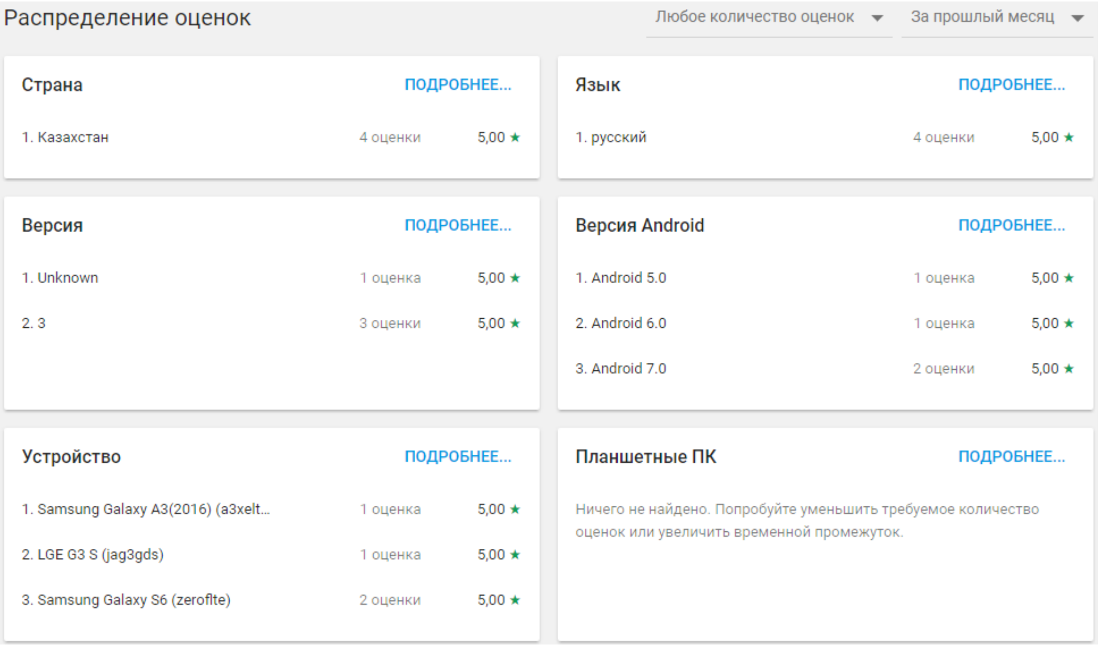
Динамика оценок

ДЕНЬ ЕЖЕНЕДЕЛЬНО МЕСЯЦ

□ Показывать общий рейтинг □ В процентах

Figure 3.15 – The dynamics of assessments

The last figure 3.16 shows us the distribution of assessments. In the figure we can see the results of user’s installation by region, version of Android OS, language and the through what kind of devices user installed the application.



Распределение оценок

Любое количество оценок ▾ За прошлый месяц ▾

Страна ПОДРОБНЕЕ...

1. Казахстан	4 оценки	5,00 ★

Язык ПОДРОБНЕЕ...

1. русский	4 оценки	5,00 ★

Версия ПОДРОБНЕЕ...

1. Unknown	1 оценка	5,00 ★
2. 3	3 оценки	5,00 ★

Версия Android ПОДРОБНЕЕ...

1. Android 5.0	1 оценка	5,00 ★
2. Android 6.0	1 оценка	5,00 ★
3. Android 7.0	2 оценки	5,00 ★

Устройство ПОДРОБНЕЕ...

1. Samsung Galaxy A3(2016) (a3xelt...	1 оценка	5,00 ★
2. LGE G3 S (jag3gds)	1 оценка	5,00 ★
3. Samsung Galaxy S6 (zeroflte)	2 оценки	5,00 ★

Планшетные ПК ПОДРОБНЕЕ...

Ничего не найдено. Попробуйте уменьшить требуемое количество оценок или увеличить временной промежуток.

Figure 3.16 – The distribution of assessments

4 Economic part

The theme of diploma project is the creating a platform based on the web and mobile application for organizing debates in the cultural sphere.

The purpose of this work is to develop an automated website and mobile application to create surface for organizing, which allows you to receive information about it. And if it is interest to user, start a blog and write about the actual.

The software developed in the framework of the thesis was originally designed to help provide the information to small audience and to improve its performance.

4.1 Stages and timing of the project implementation

Table 4.1 - Stages of application development and estimation of monetary time and labor costs

Stages of application development	The name of the stages	Executor	Labor input person/hour	Labor input days
1	Formulation of an idea	Manager	15	3
2	Development of the technical task	Manager	27	6
3	Examining the necessary materials for application development	Developer	204	22
4	Analysis of market place and future of the project	PR-manager	25	5
5	The first prototype of the project	Programmer-developer	142	15
6	Executi on of work for the presentation	Developer	73	10
7	Results of the work done	Manager	64	9
Total complexity of the completed project			550	70 (3 month)

4.2 Calculating the cost of developing an application

Determining the cost of developing an application is made by drawing up an appropriate estimate, which includes the following articles:

- expenses for necessary materials;
- costs for employees;
- deductions for social needs;
- depreciation of fixed assets;
- other expenses.

4.2.1 Calculation of material costs

"Material costs" include the costs of basic and auxiliary materials, the energy needed to develop the application.

The calculation of expenditures for material resources is given in table 4.2.

Table 4.2 - Expenditure of material resources

Name of material resources	Unit. of measurements	Quantity of consumable material	Cost per unit, tg	The total amount, tg
Paper A4	Package	1	900	900
USB 3.0 drive	Piece	1	3000	3000
Pen	Piece	4	150	600
Pencil	Piece	5	45	225
Outcome of material inputs				4725

The calculation of hardware and software costs is shown in table 4.3.

Table 4.3 - Equipment and software costs

Name of	Unit of measurements	Quantity	Price per unit, tg	Sum of, tg
Laptop Lenovo G510	Pc.	1	150 000	150 000
Xiaomi Redmi 4A	Pc.	1	70 000	70 000
OS Windows 10	Licensed copy	1	33 000	33 000
MS Office 2007	Licensed copy	1	20 000	20 000
Android Studio	Licensed copy	1	Free	Free
Total hardware and software costs				279 000

The total cost of material resources (C_m) is determined by the formula:

$$C_m = \sum_{i=1}^{n} E_i * P_i \qquad (4.1)$$

where E_i - consumption of the i-th type of material resource, tg;

P_i - price per unit of the i-th type of material resource, tg;

i – type of material resource;

n – number of types of material resources;

4.2.2 Calculation of expended amounts for electricity

To develop the application used a variety of technical equipment, and certainly spent energy.

The calculation of electricity costs is carried out according to the form given in table 4.2.

Table 4.4 - Electricity costs

Names	Equipment power kW	Coefficient of power use	The operating time of the equipment for the development of the project, h	Operating time for the development of the project, days	The price of electricity, $\frac{tg}{h * kW}$	The amount, tg
Laptop Lenovo G510	0.075	0.9	535	78	16.65	601.23
Smartphone Xiaomi Redmi 4A	0.02	0.9	21	3	16.65	10.584
Mouse Logitech M100	0.045	0.9	50	7	16.65	56.7
TOTAL energy costs						668

The amount of electricity costs (E_c) is calculated by the following formula:

$$E_c = P_i \cdot K_i \cdot T_i \cdot P \qquad (4.2)$$

where P_i – passport power of the i-th electrical equipment. kW

K_i – coefficient of power use of the i-th electrical equipment (it’s assumed Ki =0.7; 0.9)

T_i - time of operation of i-th equipment for the entire period of application development, h;

P – Price of electricity, tg/kWh

i – Type of electrical equipment

n – Quantity of electrical equipment

Here are the calculations for each equipment:

1) Laptop Lenovo G510

E_c = 0.075·0.9·535·16.65 = 601.273125.

2) Smartphone Xiaomi Redmi 4A

E_c = 0.02·0.9·21·16.65=6.2937.

3) Mouse Logitech M100

E_c = 0.045·0.9·50·16.65=33.716

Total amount of depreciation deductions amounted to 668.

4.2.3 Calculation of labor costs

"Labor costs" includes the costs of remuneration of all employees engaged in the development of mobile applications.

Labor costs are calculated according to the form given in table 4.4.

Table 4.5 - Labor costs.

Position of employee	Salary for month, tg	Rate of hour, tg/h	Labor intensity of work execution, people*h	Amount, tg
Developer	200 000	1190	329	391 510
PR-manager	180 000	1071	33	35 343
Manager	250 000	1488	156	232 128
Total labor costs				658 981

The total cost of labor (C_L) is determined by the formula:

$$C_L = \sum_{i=1}^{n} H_i * L_i \quad (4.3)$$

where H_i – hour rate of the i-th employee, tg;

L_i – labor input of research, people·h;

i – category of the employee;

n – number of employee involved in the development of the application.

The hourly rate of the employee is calculated by the following formula:

$$H_i = \frac{SM_i}{FWT_i} \tag{4.4}$$

where SM_i – monthly wage of the i-th employee, tg;
i – monthly fund of working time of the i-th employee, FWT_i = 168h

Let’s perform calculations for each employee:

$$H_1 = 200\,000/168 = 1190 \text{ tg/h}$$

$$C_{L1} = 1488 \cdot 329 = 391\,510 \text{ tg}$$

$$H_2 = 180\,000/168 = 1071 \text{ tg/h}$$

$$C_{L2} = 893 \cdot 33 = 35\,343 \text{ tg}$$

$$H_2 = 250\,000/168 = 1488 \text{ tg/h}$$

$$C_{L2} = 1607 \cdot 156 = 232\,128 \text{ tg}$$

4.2.4 Calculation of deductions and social needs

The costs under this item are deductions for the single social tax (UST).

According to the Tax Code of the Republic of Kazakhstan, the social tax is 11% of the WFP (wage fund). It should be noted that pension contributions are not subject to social tax. Payroll deductions are determined by the following formula:

$$P_D = (WFP - PF) \cdot 0.11 \tag{4.5}$$

where PF – deductions to the pension fund, which is 10% of the WFP,

$$PF = 658\,981 \cdot 0.1 = 65\,898$$

So, deductions from wages amounted to:

$$P_D = (658\,981 - 65\,898) \cdot 0.11 = 65\,239$$

4.2.5 Calculation of other costs

"Other costs" include expenses for rent, for various services and other expenses.

The cost of rent is determined depending on the cost of renting 1 square meter of occupied space.

Table 4.6 - Rent for rent of premises

The area, sq.m	Cost per square meter, tg	Cost per month, tg	Term, month	Amount, tg
25	4 000	100 000	3	300 000

To develop the mobile application, we used the service of the ID net provider from Kazakhtelecom. And the ID Net 120 tariff plan was selected, its cost and the total amount calculation is indicated in table 4.7.

Table 4.7 - Internet usage costs

Price per month, tg	Term, month	Amount, tg
3 500	3	10 500

Total other costs amount to 310 500 tenge.

4.2.7 Budgeting

Based on the data obtained for individual articles, a cost estimate for the development of the mobile application was compiled according to the form given in table 4.8.

Table 4.8 - Cost estimates for the development of the application

Cost items	Amount, tg
Material costs, including: - Material - Energy	 4725 668
Labor costs	658 981
Social needs	65 239
Other costs	310 500
Equipment and software costs	279 000
TOTAL	1 319 113

After the calculation of material costs, including materials and energy consumption, labor costs, social expenditures, depreciation of fixed assets and other costs, the total cost of implementation was 1 319 113 tg.

4.2.8 Determination of the possible (contractual) price of project

The value of the contractual price of the mobile application and web should be established taking into account the efficiency, quality and timing of its implementation at a level that meets the economic interests of the customer or the consumer and the performer.

The contract price (Pc) for an applied mobile application is calculated by the formula:

$$P_C = C_I \cdot (1 + \frac{R}{100}) \qquad (4.6)$$

Where C_i – costs for the implementation of research (from table 4.8), tg;

R – average level of profitability of R & D, % (is accepted in the amount of 20% in consultation with the economic consultant).

Next, the selling price is determined, taking into account the value-added tax (VAT), the VAT rate is established by the Tax Code of the Republic of Kazakhstan. For 2017, the VAT rate is set at 12%.

The selling price, including VAT, is calculated by the formula:

$$P_s = P_c + P_c \cdot VAT \qquad (4.7)$$

Taking into account all previous calculations, we will determine the possible (contractual) price of the mobile application

P_c = 1 319 113 · (1 + 20/100) = 1 582 935

Price including VAT:

P_s = 1 582 935 + 1 582 935 · 0.12 = 1 772 887 tg.

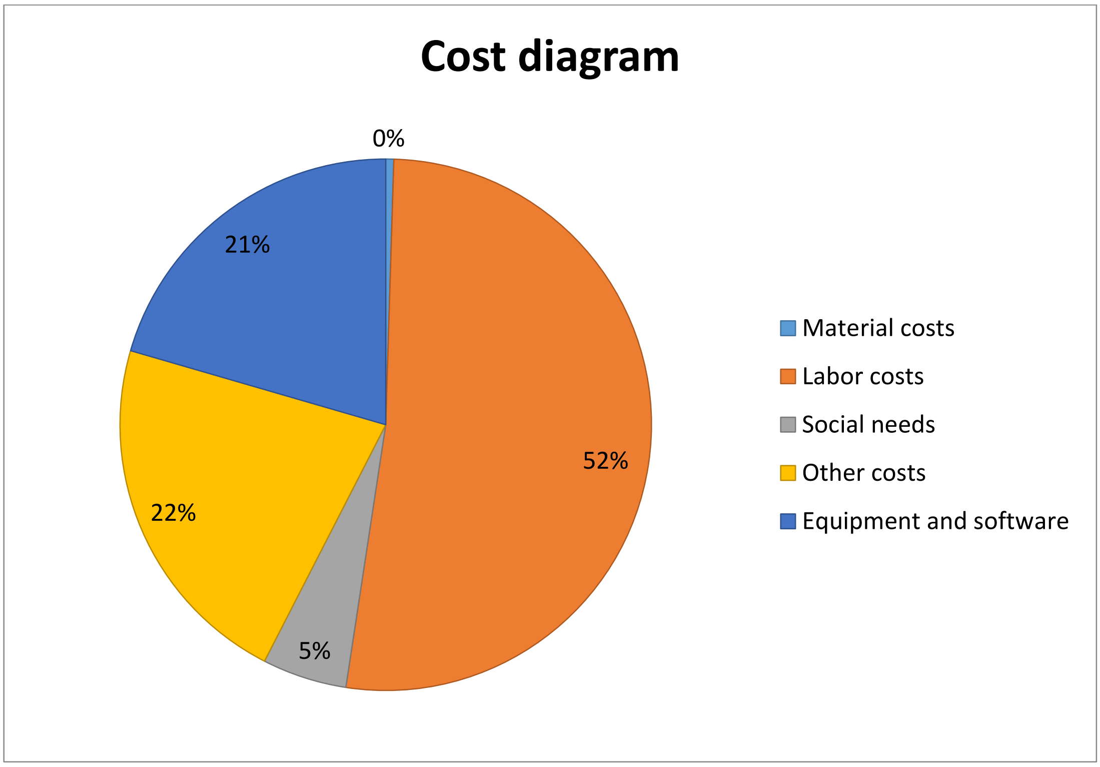


Figure 4.1 – Cost Diagram

4.3 Assessment of the scientific and technical performance of the project

For companies, the growth of the company's profit indicators is very important. The value of mobile applications is estimated by several indicators. It:

- number of active users;
- income from mobile application;
- uniqueness of the application functionality;
- the effectiveness of the work done on the application;
- and other important indicators;
- there are several ways to earn from a project.

The total cost of the project was 1 319 113 tg, which included various costs for the implementation of the project.

By calculations above, the most part of the share was for labor payment - 49%. Further, other expenses follow from 23% (Figure 4.1).

This calculation showed us how much resources were spent and on what. And in terms of efficiency, looking at the report and making a couple of calculations, we can reduce the unnecessary costs of developing the application; increase the speed of releases of new versions. Assess the possibility of hiring new staff. It is more reasonable to allocate money for various expenses.

5 Health and safety

Theme of the diploma project is "Developing a mobile application for social assistance to those in need".

The purpose of the work is the development of an instant messenger, which allows potential buyers to communicate with the operator and receive reliable information.

The main task is to create an easy-to-understand chat window for the average user and a convenient administration panel.

This section contains BD components for the implementation of the project under development, calculation of artificial lighting, microclimate, and analysis of the production premises, equipment and mode of work of employees.

5.1 Analysis of working conditions

Consider an example of a typical production room. In the room, in addition to equipment and workplaces, there may be different heaters and air conditioners in the room where the computer was installed. The complex of computer equipment additionally includes: a plotter, a laser printer, a matrix printer. A web developer as a program user and a PC at his workplace is exposed to the following harmful factors:

- unfavorable microclimatic conditions, which are determined by the air temperature, its composition and pressure, relative humidity, airflow velocity;
- improper lighting of the workplace due to the non - optimal location and use of lighting devices;
- PC working noise, ventilation devices, air conditioners and external noises;
- electromagnetic radiation, the main source of which is the monitor;
- psychophysiological factors, i.e., mental overstrain of visual and auditory analyzers, monotony of work, emotional overload and others;
- danger of electric shock U = 220 V, f = 50 Hz; When using electrical appliances, in particular a computer;
- risk of fire due to improper connection of electrical appliances.

Long-term presence of a person in the zone of combined exposure to various adverse factors can lead to occupational disease. It should be noted that the personnel of the computer laboratory works in conditions of constant danger

The layout of the working area is shown in figure 5.1.

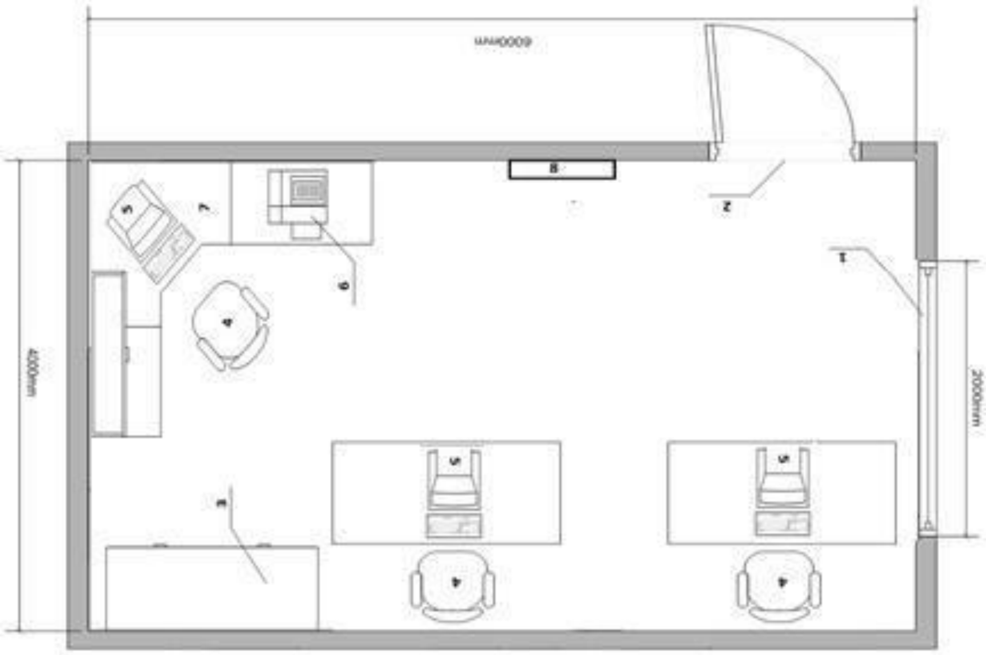


Figure 5.1 - Working area

.

where 1 is the window;

2 - the door;

3 - the conditioner;

4 - chairs;

5 - computers;

6 - the printer;

7 - cabinet;

8 - tables.

The room has the following parameters: - working class dimensions: height - 3.2 m, width - 4 m, length - 6 m; - the total area of the room is 24 m^2; - area occupied by equipment and furniture ≈ 15 m^2; - The room for visual working conditions belongs to the category of light work (light physical, category I_a, work done by sitting and does not require physical stress) (GOST 12.2.032-78); - artificial lighting - lamps: 2 lamps, each with 2 fluorescent lamps. When working with a PC, potentially dangerous and harmful factors can occur, the impact of which on the human body can cause harm to it and lead to injuries. The main factors with their origin and norms are set out in GOST 12.1.003-74 / 80 and are given in table 5.1.

Table 5.1 - Major hazards and hazards

Factor name	Place of occurrence	Remote control, MPC	Possible consequences
Increased value of the electrical circuit voltage	Developer Workplace	GOST 12.1.038-82. In the emergency mode, $U_{pr} \leq 36V$ alternating current for prolonged exposure (more than 1s).	Electrical Trauma
Electric arc	Switchboard	GOST 12.2.007.3-75 GOST 12.2.007.4-75 GOST 12.1.004-85	Burns, fire
Name factor	Place of occurrence	Remote control, MPC	Possible effects
Insufficient illumination of the working area	The room	SNiP 23-05 E=300 Лк	Fatigue, the risk of injury, blurred vision
Increased noise level	Workplace	GOST 12.1.003-88 Level on frequency bands: less than 75 dB.	Nervous-psychological overload, diseases of the hearing organs
Monotone labor	Workplace	GOST 12.1.003-80	Neuropsychiatric overload

5.2 Calculation of artificial lighting

Calculation of artificial illumination of the development department by the use analysis method. During execution of the calculation part, you must select the lighting, light source, type of lamp for the specified area or office. It is necessary to determine the number of luminaires that are necessary to provide the E_{min} and the power of the lighting installation, which are necessary to provide working classes with normal lighting. Reset of visual work - V. Nominal illumination according to table.

Table 5.2 - Technical characteristics of fluorescent lamps T8

Rated power, W	Rated light flux of T8 type lamps	Lamp size, m	
		Diameter	Length by pins
36	3250	25,78	12

As a luminaire is taken T9 Luxxline Pilus F56W / 828. The length of the luminaire is 1400 mm; width 150 mm. Calculation of artificial lighting is done using the coefficient of utilization. Coefficients of reflection from the ceiling of walls and floor are equal:

- ρCEI = 70%;
- ρWA= 50%;
- ρFL = 30%.

Calculate the height of the luminaire over the work surface

$$H = h - h_P - h_c, \tag{5.1}$$

where h_C- distance from the luminaire to the ceiling, $h_c = 0.11$; h_p - height of the working surface above the floor;

$h_r = 0,8$ m. h-height of the room, h = 3,2 m.

H = 3,2 – 0,8 – 0,11 = 2,29м

The advantageous distance from the window to the luminaire is defined as

$$L = \lambda \cdot H \tag{5.2}$$

The distance from the wall to the nearest lamp, when the work at the wall is not carried out, is determined by the formula

$l_1 = (0,4 \div 0,5) \cdot L$ (5.3)

$l_1 = 0,4 \cdot 2,748 = 1,1$ м

The utilization factor in this case is n = 49, the safety factor is $k_3 = 1.2$.

Let us determine the number of fluorescent lamps according to the formula where

- S is the area of the room.
- E-given minimum illumination, E = 400 lux.
- Z-coefficient of unevenness of illumination, Z = 1,1; n-number of lamps in the luminaire.
- F_l - light flux of the selected lamp, $F_l = 3250$ lm.
- utilization factor, n = 410.

In total, to create a standard illumination of 400 lux, 4 lamps with 2 fluorescent lamps are needed each, the power of each lamp should be 36 W, which means that it is not necessary to increase or decrease the number of lamps available to meet sanitary standards.

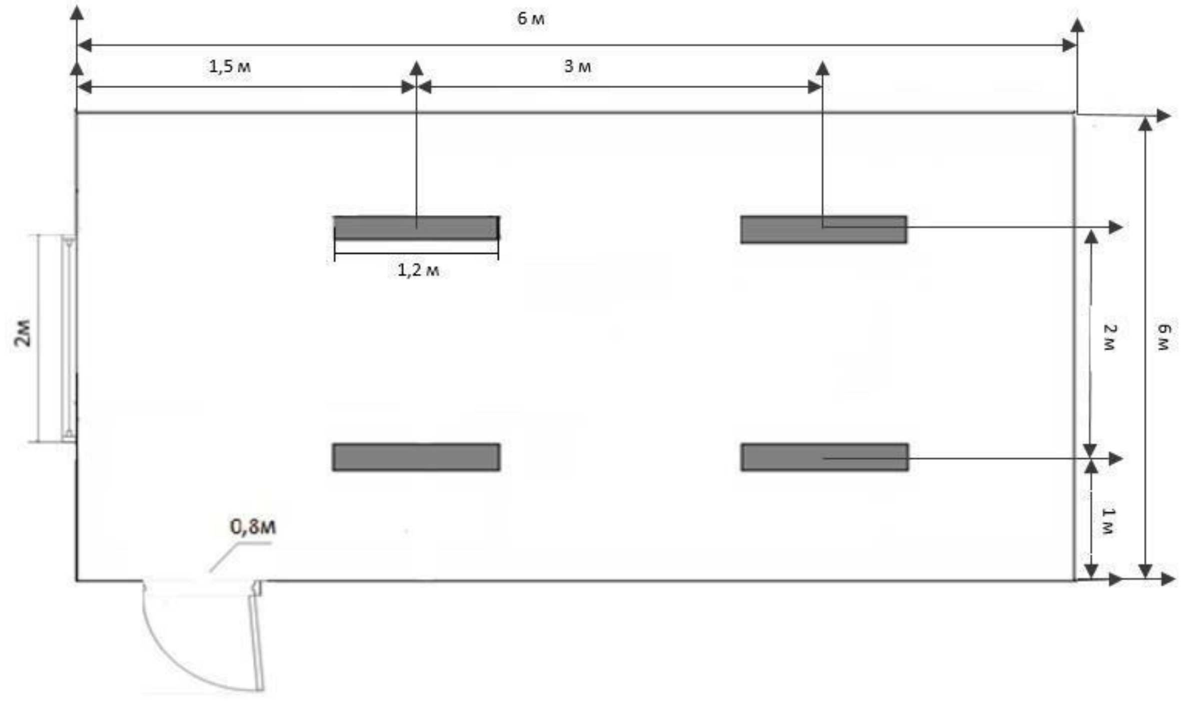


Figure 5.2 - Layout of lighting fixtures

5.3 Development of fire safety issues

According to SNiP 2.04.09-84 and RNTP-01-94, the building, depending on the degree of danger of fire development, from the functional purpose and the fire load of combustible materials, belongs to the group of the 1st category.

Causes of a fire can be:

- ignition of equipment elements;
- ignition of finishing materials from faulty switches, sockets;
- non-observance of operating conditions of equipment, improper operation of personnel.

In the event of a fire, not only the room, but also expensive equipment can be affected, which leads to human casualties. Therefore, it is necessary to take measures to identify and eliminate fires. The sources of ignition can be electronic circuits of computers, devices used for maintenance, power devices, air conditioners, in which as a result of various violations, superheated elements are formed, etc.

In accordance with the requirements of fire safety of premises for storage of fire-extinguishers OU-5, taking into account - one fire extinguisher per 100 m2. The total area of the premises is 76.8 m2, so a fire extinguisher is installed. As a fire extinguishing agent, a combined mixture of carbon dioxide-chloride is used. The calculated mass of the combined carbon dioxide-chlondon md composition for bulk fire extinguishing is determined by the value of 10.6

$$m_d = k \cdot g_n \cdot V \quad (5.6)$$

where k = l,2 – coefficient of compensation for not taken into account loss of carbon dioxide-coolant composition;

g_n = 0,04 - the normative mass concentration of the carbon dioxide-chladone composition.

V - the volume of the room is determined by the formula 5.7.

$$V=A \cdot B \cdot H \quad (5.7)$$

where A = 6м – length of room;

B = 4м – width of the room;

H = 3,2м – height of the room.

5.4 Calculation of air conditioning in an office building

Define the type of air conditioner to create a comfortable working environment for the employee. Excess heat generation and heat transfer can be is produced from windows, doors, walls, equipment, etc.

Excess heat is determined by the formula:

$$\mathrm{Q_{red}} = (\mathrm{Q_{equip}}+\mathrm{Q_{light}}+\mathrm{Q_{per}}+\mathrm{Q}_{p})-\mathrm{Q}_{td} \quad (5.8)$$

where Q_{equip} - heat, allocated by the production equipment;

Q_{light} - the artificial lighting system of the room;

Q_{per} - working people;

Q_p - heat introduced into the room by the sun (solar radiation);

Q_{td} - heat transfer by natural way.

Heat emitted by production equipment is determined according to the formula:

$$Q_{equip} = 860 \cdot P_{equip} \cdot n$$

where 860 is the thermal equivalent of 1 kBT · h, i.e. Heat equivalent to 1 kW of electrical energy;

P_{equip} - power of consumed equipment, kW / h;

η - the coefficient of heat transfer to the room. The value is equal 0.95 is the rate of heat loss personal computer.

For the 1st computer:

$$Q_{equip} = 860 \cdot (3 \cdot 0.25) \cdot 0.95 = 204.25 \text{ kcal / h}$$

The value of η = 0.95 is the rate of loss of power consumption for heat dissipation of computer equipment. Heat generated lighting installations.

$$Q_{equip} = 860 \cdot N_{light} \text{ x } \eta \qquad (5.9)$$

where N_{light} - consumed power of lamps, kW;

η = 0,55 - norm of losses of power consumption for heat release fluorescent lamps.

$$Q_{equip} = 860 \cdot 0.55 \cdot 0.52 = 246 \text{ kcal / h}$$

The heat emitted by workers is calculated by the formula:

$$Q_{people} = K \cdot (q_{ep} \cdot q_{pc}) \qquad (5.10)$$

where K - quantity of working personnel;

q_{equiop} - explicit heat, kcal / h;

q_{pc} -heat release for one person for this category works I-III, kcal / h.

Work performed in the room, belongs to the I category of work.

Those q = 100 W, or 0.1 kW for office space

$$Q_{people} = 1 \cdot (860 \cdot 0.1) = 86 \text{ kcal / h}$$

Heat input from the sun is calculated by the following formula:

$$Q_p = m \cdot F \cdot q_{rest} \qquad (5.11)$$

where m - the number of windows in the room;

F - the area of one window, m

q_{rest} - solar radiation through a glazed surface, i.e. the amount of heat introduced per hour through the glazed surface an area of one meter square.

For a double glazed window with wooden bindings q_{rest} = 105 (the windows face north, Almaty is at latitude 43 ° north latitude). The number of windows is 2.

The window area is equal to

$$F = 2.4 \cdot 2 = 4.8 \text{ m}^2$$

$$Q_p = 2 \cdot 4.8 \cdot 105 = 1008 \text{ kcal / h}$$

For a warm period of the year, in calculations, $Q_{td} = 0$ can be assumed.

$$Q_{red} = 204.25 + 246 + 86 + 1008 = 1544.25 \text{ kcal / h}$$

In the presence of excess heat, the amount of air that must be removed from the premises.

$$L_b = \frac{Q_{red}}{C_p + p_b(t_{out} + t_{inc})} \quad (5.12)$$

where Q_{red} - the excess heat, kcal / h;

C_p - heat capacity of air (0,24 kg / °C)

p_b - 1,206 kg / m3 is the specific mass of the supply air.

t_{in} - temperature of the air leaving the room, °C;

t_{out} - temperature of air entering the room, °C;

Then we got these calculations:

$$L_b = 1544.25 / (0.24 + 1.2(22-20)) = 584.94 \text{ m}^3 \text{ /hours}$$

The existing air conditioner has an airflow of 655m3 / h., that and is sufficient to ensure comfortable microclimate.

The parameters of the air conditioner:

- productivity for the cold -3155 (W);
- heat output -3700 (W);
- power consumption in cooling mode -1250 (W);
- power consumption in heating mode -1250 (W);
- indoor unit noise level -45 (dB (A));
- supply voltage -1/220/50 (P_h / V / Hz);
- the weight of the indoor unit is -13 (kg).

Conclusion

As a result of the diploma work there was created an application based on Android OS which propagates virtue and charity among Almaty citizenships. Designed as an application in the first place to allow people to share with the confronted difficult situation and ask for help. The application is free and with useful advertising, which shows the ratings of the active users who have helped a lot of people and thus which will be the motivation to do well to people.

The main sources revealing a comparison app store was for Android OS users Google’s app “Android Market” (Google play, Play Market).

The implementation of an application was made by a high-level object-oriented Java programming language. The Java helps reduce costs, implement innovation and increase the quality of service application.

After studying architecture, Android OS, we discussed means of developing an application for the operating system and complete study of the creation of an application for propagation virtue.

The main objective of research was to create Android-based application for propagate virtue. Where did the tasks such as:

- study of the Android OS;
- introduction to Android Studio platform;
- the development of the Java programming language;
- the studies and analysis of such applications;
- implementation of the application itself.

Also, the functional simulation software has been implemented, namely:

- develop a class diagram;
- organization of the access level;
- develop a sequence diagram for the search function;
- development of the notification function.

Scientific work has achieved its goal and was launched in the store Google Play (Android Market) and AppStore applications.

These objectives and the results have been justified, but that does not finish the project, because the world is in the move, as the introduction of innovations in human relations do. And in the future we plan to update our application after feedback from the users.

List of references

1 Journal of Lorenzo Snow, 1841–47, Church History Library, 39–42.

2 Помощь людям – это открытие нового себя, 2012 [Online] Available: www.spasizhizn.com (Accessed: 12 Feb.2016).

3 Jiyoung Choi, Creating an evaluation system for a mobile application design to enhance usability and aesthetics, Iowa: 2012.

4 Klimov, A. Preparation for the development, Basic components of Android,2000. [Online] Available: http://developer.alexanderklimov.ru /android/android1.php (Accessed 13 Nov.2015).

5 Loren D., Shein K. ANDROID for 24 hours. Moscow: Rid Group, 2011.

6 Reto Meier. Professional Android Application Development. USA, 2009

7 Damager82. Full list of lessons, Start Android list all the lessons, 2011. [Online] Available: http://startandroid.ru/ru/uroki/vse-uroki-spiskom.html (Accessed 13 Nov.2015).

8 Cyran M. Oracle Database Concepts, 10g Release 2 (10.2). USA, 2005

9 YakovF,. Programming in Java for children, parents and grandparents. 3rd edition, Moscow: 2011.

10 https://www.lifewire.com/what-is-ios-1994355.

11 Savov, Vlad (June 13, 2016). "Apple announces iOS 10 with 10 major features and redesigns". The Verge. Vox Media. Retrieved January 17, 2017.

12 Clover, Juli (September 13, 2016). "Apple Releases iOS 10 With Redesigned Lock Screen, New Messages Features, Third-Party Siri, and More". MacRumors. Retrieved January 17, 2017.

13 StoneRiverElearning. Mobile UI and UX Design Tutorial, 2015. [Online] Available:https://www.youtube.com/playlist?list=PL5eJgcQ87sgfTfa339_OSuAcdS yPSDSjJ (Accessed 25 Nov.2016).

14 Golden Krishna, The best interface is no interface, USA: 2015.

15 Stepan, S. “Design of a mobile application. How to achieve optimum result?”, 2015. [Online] Available: https://habrahabr.ru/company/ appodeal/blog/260095/.

16 Eric Bieller, How to design app using user interface deign principles, 2016. [Online] Available: https://www.careerfoundry.com/en/blog/ui-design/how-to-design-a-mobile-app-using-user-interface-design-principles/.

17 Cameron Chapman, “Visual design vs. Graphic design: What`s the difference?”, 2017. [Online] Available: https://skillcrush.com/2016/05/31/what-is-visual-design/.

18 https://connect.redhat.com/partners/backendless.

19 The Labor Code of the Republic of Kazakhstan. Astana, Akorda, on May 15, 2007, No. 251-III ZRK (with amendments and additions as of January 17, 2014).

20 Sanitary rules "Sanitary and epidemiological requirements to working conditions with sources of physical factors (computers and video terminals) that

have an effect on a person", approved by the order of the Minister of National Economy of the Republic of Kazakhstan dated January 21, 2015 No. 38.

21 http://online.zakon.kz/Document/?doc_id=33585984#sub_id=1
http://online.zakon.kz/Document/?doc_id=38910832#pos=30;-132
https://egov.kz/cms/ru/law/list/V100006438.

22 Article “Охрана труда. Организация наиболее оптимального рабочего места инженера разработчика.” http://webcache.googleusercontent.com/search?q=cache:http://vv399.narod.ru/BJD.doc.

23 Law of the Republic of Kazakhstan "On Civil Protection" of April 11, 2014 No. 188-V (with amendments and additions as of April 8, 2016).

24 Catalog of ventilation equipment “ Вентоборудование-Тверь” http://www.tverclima.ru/texts/ventilator.pdf.

25 Article “Расчет систем вентиляции” http://www.rfclimat.ru/htm/vent_ft.htm.