МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
Факультет кібербезпеки, комп'ютерної та програмної інженерії
Кафедра комп'ютерних систем та мереж

"ДОПУСТИТИ ДО ЗАХИСТУ"
Завідувач кафедри

_____ Жуков І.А.

"_____"_____2020 р.

ДИПЛОМНА РОБОТА

(ПОЯСНЮВАЛЬНА ЗАПИСКА)

випускника освітнього ступеня "МАГІСТР"

спеціальності 123 «Комп'ютерна інженерія»

освітньо-професійної програми «Комп'ютерні системи та мережі»

на тему: **"Програмний модуль системи для проектування інтер'єрів"**

Виконавець: _____ Гришко Н.С.

Керівник: _____ Кудренко С.О.

Нормоконтролер: _____ Надточій В.І.

Засвідчую, що у дипломній роботі
немає запозичень з праць інших авторів
без відповідних посилань
Гришко Н.С.

Київ 2020

MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE
NATIONAL AVIATION UNIVERSITY
Faculty of Cybersecurity, Computer and Software Engineering
Computer Systems and Networks Department

"PERMISSION TO DEFEND GRANTED"

The Head of the Department

_____ Zhukov I.A.

"_____"_____2020

MASTER'S DEGREE THESIS

(EXPLANATORY NOTE)

Specialty: 123 Computer Engineering

Educational-Professional Program: Computer Systems and Networks

Topic: "**Software module of the system for interior design**"

Completed by: _____Hryshko N.S.

Supervisor: _____ Kudrenko S.O.

Standard's Inspector: _____Nadtochii V.I.

Kyiv 2020

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет кібербезпеки, комп'ютерної та програмної інженерії

Кафедра комп'ютерних систем та мереж

Освітній ступінь: «Магістр»

Спеціальність: 123 «Комп'ютерна інженерія»

Освітньо-професійна програма: «Комп'ютерні системи та мережі»

"ЗАТВЕРДЖУЮ"
Завідувач кафедри
_____ Жуков І.А.
"_____"_____2020 р.

ЗАВДАННЯ
на виконання дипломної роботи

Гришко Наталії Сергіївни
(прізвище, ім'я та по-батькові випускника в родовому відмінку)

1. Тема дипломної роботи: "Програмний модуль системи для проектування інтер'єрів" затверджена наказом ректора від 25.09.2020 р. № 1793/ст

2. Термін виконання роботи (проекту): з 1 жовтня 2020 р. до 25 грудня 2020 р.

3. Вихідні дані до роботи (проекту*): Система для проектування інтер'єрів*

4. Зміст пояснювальної записки: *Вступ, огляд теми, огляд існуючих технологій побудови програмного модуля системи для проектування інтер'єрів, розробка програмного модуля системи для проектування інтер'єрів, висновки по роботі.*

5. Перелік обов'язкового графічного (ілюстративного) матеріалу: *Графічні матеріали результатів дослідження надати у вигляді презентації у форматах .ppt, .pdf.*

NATIONAL AVIATION UNIVERSITY

Faculty of Cybersecurity, Computer and Software Engineering
Department: Computer Systems and Networks
Educational Degree: "Master"
Specialty: 123 "Computer Engineering"
Educational-Professional Program: "Computer Systems and Networks"

"APPROVED BY"
The Head of the Department
_____Zhukov I.A.
"_____"_____2020 p.

Graduate Student's Degree Thesis Assignment

Hryshko Nataliia Serhiivna

1. Thesis topic: "Software module of the system for interior design**"**

approved by the Rector's order of 25.09.2020 p. № 1793/ст

2. Thesis to be completed between 01.10.2020 and 25.12.2020

3. Initial data for the project (thesis): *system for interior design*

4. The content of the explanatory note (the list of problems to be considered): *Introduction, overview of the topic, review of existing technologies for developing the software module for the interior design, design of the software module for the interior design, conclusions on work*

5. The list of mandatory graphic materials: *Graphic materials are given in MS Power Point presentation.*

6. Календарний план-графік

№ пор.	Завдання	Термін Виконання	Підпис керівника
1	Узгодити технічне завдання з керівником дипломної роботи	1.10.20-8.10.20	
2	Виконати пошук та вивчення науково-технічної літератури за темою роботи	9.10.20-15.10.20	
3	Опрацювати теоретичний матеріал	16.10.20-18.10.20	
4	Огляд існуючих технологій для розробки програм-ного модуля для системи проектування інтер'єрів	19.10.20-03.11.20	
5	Розробка програмного модуля для системи проектування інтер'єрів	04.11.20-15.12.20	
6	Тестування роботи програмного модуля та виправлення знайдених помилок	16.12.20	
7	Оформлення пояснювальної записки	06.12.20-12.12.20	
8	Оформити графічні матеріали проекту та подати матеріали роботи на антиплагіатну перевірку матеріалів	13.12.20-14.12.20	
9	Отримати рецензію та відгук керівника та надати матеріали роботи на кафедру.	15.12.20-18.12.20	

7. Дата видачі завдання: “1” жовтня 2020 р.

Керівник дипломної роботи _____ Кудренко С.О.
(підпис керівника)

Завдання прийняв до виконання _____ Гришко Н.С.
(підпис випускника)

6. TIMETABLE

#	Completion stages of Degree Project (Thesis)	Stage Completion Dates	Signature of the supervisor
1	Technical task coordination with the supervisor	1.10.20-8.10.20	
2	Selection and study scientific literature on the topic	9.10.20-15.10.20	
3	Working with theoretical materials	16.10.20-18.10.20	
4	Review of existing technologies for developing the software module for the interior design	19.10.20-03.11.20	
5	Designing of software module for the interior design	04.11.20-15.12.20	
6	Testing of the designed software module and fixing the found bugs	16.12.20	
7	Making the explanatory notes	06.12.20-12.12.20	
	Preparation of the graphical materials and filing materials of work to antiplagiarism checking of materials	13.12.20-14.12.20	
	Receiving the reviews from reviewer and supervisor and providing the materials to the department	15.12.20-18.12.20	

7. Assignment issue date: 1.10.2020

Diploma Thesis Supervisor _____Kudrenko S.O.
(Signature)

Assignment accepted for completion _____Hryshko N.S.
(Student's Signature)

ABSTRACT

The Explanatory Note to on Master's Degree Graduation Project – "Software module of the system for the interior design": 92 pages, 41 figures, 2 tables, 20 references.

The Goal of Graduation Project. The goal of the graduation project is to design a software module than can help customers calculate the quantity of material they need to buy.

Main Tasks. To compare the technologies that can be used for the web application construction, analyze the input and output data and select the technologies that are the most appropriate; design and test the software module of the system for interior design that allows to calculate the quantity of material the user needs to buy based on the user's room size and create the orders.

The Subject of Project. Designing and testing of software module of the system for the interior design.

Practical usage. Can be used in real world by real customers after signing up the contract with the real shops and integration with their database.

Main Metrics and Results. As a result, there was designed software module which help people to design and buy in more convenient way. It consists of the two portals: User and Admin Portals. The first one is for customers and the second one was design to manage the orders and add goods.

The main idea of the application is to make it easy to calculate the number of materials needed and create the order. A user enters the parameters of the room and adds the item to cart. Then he in she can simply clicks on the "Calculate" option near item on the Cart and the Calculation pages will be open. Here the user can select on which wall he or she want to apply the wallpapers and click on the corresponding option. The results will be displayed in a second and the user can update the cart via the "Update" option.

The software module was manually tested and covered with the end-to-end test scenarios.

CONTENT

LIST OF SYMBOLS, ABBREVEATIONS, TERMS

BIM	Building Information Modeling/Management
PC	Personal Computer
AEC	Architecture Engineering Construction
PWA	Progressive Web Application
DB	Data base
OOP	Object-oriented programming
TS	TypeScript
NPM	Node Package Manager
HTML	Hypertext Markup Language
CSS	Cascading Style Sheets
UI	User Interface
DOM	Document Object Model
JSX	JavaScript XML
API	Application Programming Interface
URL	Uniform Resource Locator
SaaS	Software-as-a-Service
MVP	Minimum Viable Product
SDK	Software Development Kit
IDE	Integrated Development Environment
SQL	Structured Query Language
NoSQL	Non Structured Query Language
IAM	Identity and Access Management
JSON	JavaScript Object Notation
HTTP	Hypertext Transfer Protocol
SSL	Secure Sockets Layer
CDN	Content Delivery Network

INTRODUCTION

Nowadays the interior design is a very important. Modern people spend a lot of time at home and the good home design is increase the productivity and mode. However, sometimes it is complicated to create good design if the person is not a designer. The interior design is very long and complicated process. It is difficult to imagine how the picture in the head will look like in a real room.

According to the Houzz&Home survey in 2020 only 19% of people hiring the professional designer to renovate the home. Moreover, according older generations drive renovation activity defined in the survey: Baby Boomer and Gen X homeowners accounted for 85% of those renovating on Houzz in 2019, up from 83% in the prior year. The rest were largely Millennials, whose share declined from 14% in 2018 to 12% in 2019. the Baby Boomer and Gen X homeowners accounted for 85% of those renovating on Houzz in 2019, up from 83% in the prior year.

The main idea of this system for creating interior is to help people design the interior of their home by their own. The user can design the interior he or she wants via the system for interior design from the items that stored in the catalog. The only one thing he or she needs to do is to give the parameters of the room and set where the windows and doors are located. Then the items can be simple drag and drop from the catalog and the application automatically add them to the 3D model of the room.

The software module that was design in this diploma project do not have a full functionality of the system for interior design. The possibility to work and review the interior in 2D and 3D will be added in future. For now, the module allows to create the catalog on the Admin portal, review and add the items from the catalog on the User Portal and create the orders on the User Portal and manage them on the admin portal.

In addition, the project has a very useful functionality. It is possible to calculate the numbers of items he or she need to buy before proceeding to the checkout page. The user can do this for the items in the cart. After the calculation there will be displayed the calculation results and button to update the quantity of item in the cart if they are different.

Currently, the application supports calculation of four categories of items: wallpapers. tiles, parquets, and laminates.

Successful interior designers use programs that are based on BIM technologies. The apartment is assembled in 3D, like a constructor from walls, windows, doors and furniture, and then automatically get ready-made drawings based on the model. In the diploma work there were considered some of them. There are AutoCad, Revit and Archicad. All of them allows to create detailed plans, sections, elevations, legends, and schedules. The user can add their own modules. However, they cannot be used without good knowledge of programs and designer skills.

There are also more simple programs: SketchUp, Roomsketcher, Planner 5D and Ikea Home Planner. All of them have catalog of items and supports 3D visualization. The user can interior designers all the tools they need to visualize their work. The clients are impressed with the ability to translate their vision into something they can see on the screen before they see it in their living room.

But there is some disadvantaged. Most of them are not fully free and the functionality is very limited. For instance, the SketchUp has a lot of modules but they cannot be used in free account. Even for using the real items from the shops the client should pay.

The difference of this interior system from its analog is that it will be free for the ordinary customers and use the real catalogs of the shops. The shops will be interested to pay for allocation of their goods on the portal and pay for usage of the system. Also, the system for interior design is not usually connected with the real shops databases and it is hard to find the desired items in the shop.

All items of the designed interior system will be from real shops and can be ordered via the web application. Also, it is not needed to install any program on the personal computer, and it is very useful. The design can be stored online, and the user can no worry that it will be missed or deleted.

To give the user ability for convenient work there was decided to develop a web application, not the desktop program. The desktop program requires additional steps to install and it is not user friendly nowadays. In web development, many different

technologies and off-the-shelf products are used: these are programming languages and markup in a "pure" form, databases and other information stores, software products to simplify development (frameworks and site management systems), and server software. Modern web applications consist of the four main components: Interface, Software Part, Database and Hosting.

Interface of this application was designed via the Framer prototyping tool. Framer is an IDE where it is possible to build a prototype for the project for free. It runs Framer.js, an open-source Javascript library that creates the Layer abstraction, which is familiar to many designers.

The UI is developed using the React.js library. React is a declarative, efficient, and flexible TypeScript library for building user interfaces. It lets a user to compose the complex UIs from small and isolated pieces of code called “components”. React makes it painless to create interactive UIs. It designs simple views for each state in the user’s application and will efficiently update and render just the right components when the data changes. In addition, there was used the Material UI framework. It is a popular React framework which take care about injecting the CSS where needed. Material-UI components work without any additional setup, and don't pollute the global scope.

For controlling the states within application there were selected the Redux Library. Redux is a small library that work with data. It helps developers to understand where and how the data should be stored. All components are given callback functions as props and they call them when UI-event is happened. Those callbacks create and dispatch action based on the event. Reducers process the actions and computing the new state. The new state of the whole application is located into one store. Components receive the new state as a props and re-render themselves if needed.

To control routing within the application the React Router package is user. It is designed with intuitive components to let a user build a declarative routing system for the application. With declarative routing, it is possible to create intuitive routes that are human-readable, making it easier to manage the application architecture. With routers, the user experience of the app can simplify site navigation. To make the responses for the search event fast it was decided to use Algolia API. It provides convenient RESTful API

for the web sites and applications for instance search. Most of web-services and mobile applications such as Spotify, Salesforce and Amazon should guarantee fast and useful access for the objects of the databases through the simple search window.

The application has a large database, and the application should be stable and reliable. So it is very important to select the database for the application. In the diploma project there were compared three Realtime databases that can be used for the application: DynamoDB from Amazon, Firebase from Google and MongoDB from the MongoDB Inc. All of them has advantages and disadvantages. At the end of research the Firebase cloud platform was selected for the project.

Firebase is a database not a backend. Firebase does not support complicated SQL queries. It is NoSQL database that provides the mechanism for storage and retrieval the data that is different than tabular relations used in relational databases. The solution allows to quickly start the MVP development and has also a lot of advantages. Firebase has many options such as Realtime and Firestore that both are cloud-hosted, NoSQL databases flexible and scalable in terms of size. All data are stored in JSON format and synchronized for all connected clients in real time.

The platform doesn’t charge for most of its services and requires choosing a pricing plan only after reaching a certain amount of database memory. It’s great for beginners who want to validate if the platform is good enough for their product and don’t want to pay for all the services upfront.

The project was developed in the Visual Studio Code. It was design by Microsoft and comes with build-in support for JavaScript, TypeScript, and Node.js. It also has plenty of extensions for other languages (such as C++, C#, Python, and PHP). VS Code has many features such as syntax highlighting, autocompleting with IntelliSense based on variable types, function definitions, imported modules, custom hotkeys, integrating with GitHub and ect.

There is nothing ideal in the world. Everything should be changed from time to time for continuous growing. In the near future the software module designed in this project can be extended with additional modules. At the end of the third part there were described seven main improvements for the system. There are some of them: advance filtering,

different room shapes and 3D Visualization, designers support, sharing and export the planning to files, vacation tracker for the internal users, mobile application and mail notification module in the Admin Portal.

The first improvement is to design convenient filtering of the catalog items, users and orders on the Admin portal. Also, the same filtering functionality should be applied for the Catalog page of the User Portal. For now, there is only a Search field where the user can search by name or number. It is not convenient when there are so many objects in the database. The filter will be created and added for each column of the tables of the Admin portal. The user can simply click on the column name and additional popup will be displayed like the filter window in the Microsoft Excel.

Currently, the web application is limited to the rectangular and square shapes. In future the functionality that allows user to draw the room by his own or use some default templates (circles, corners, ellipses, open room solutions etc.) should be added. There will a page where the room will be displayed in 3D form and a user can edit the walls, ceil, and floor size by selecting the side and entering the size in the corresponding form. There will be options to navigate thought the 3D room model, add new walls, and update the existing ones. Also, it will be possible to delete the unnecessary walls. The design can be created for one or for several rooms simultaneously. So, it will be convenient to plan the flat and see how it will look like in general.

PART 1
OVERVIEW OF THE INTERIOR DESIGN PROCESS

1.1Analysis of the Modern Approaches in the Interior Design

The interior design is very long and complicated process. It is difficult to imagine how the picture in the head will look like in a real room. However, only 19% of people hiring the professional designer to renovate the home according to the Houzz&Home survey in 2020. Overall, pro hiring has been stable in the last three years and homeowners are continuing to hire more than one pro, on average.

Design-related pros, often hired by homeowners irrespective of renovations, were hired by 1 in 5 renovating homeowners. Among homeowners who renovated their homes, specialty service providers were the most hired renovation professionals, followed by construction professionals [1]. So, the users need to have a useful and simple to study application to clear explain their idea to the construction professionals.

Older generations drive renovation activity: Baby Boomer and Gen X homeowners accounted for 85% of those renovating on Houzz in 2019, up from 83% in the prior year. The rest were largely Millennials, whose share declined from 14% in 2018 to 12% in 2019. Renovation Pace Continues with Reduced Spend: overall home renovation activity remained stable year over year, with 54 percent of homeowners reporting a renovation project in 2019 and tackling nearly three interior rooms on average. The mix of renovation activities and project types remained consistent with recent years, with kitchens and bathrooms continuing to lead in popularity. Median renovation spends, however, declined from $15,000 in 2018 to $13,000 in 2019, owing to a reduction in average project scope as well as slightly fewer projects per homeowner.

Home offices get to work: while kitchens and guest bathrooms remained the most popular rooms to renovate (27% and 25%, respectively), home offices were added or upgraded by 1 in 10 homeowners in 2019. Millennials and Gen Xers were more likely to pursue a home office project (11% each) than were Baby Boomers (9%).

Coronavirus pandemic’s impact: Subsequent surveys have shown that over half of homeowners who were in the midst of a project at the start of the pandemic were able to continue with renovations. That said, some homeowners have opted to delay certain elective renovations due to implications related to social contact, labor and material availability and personal discretionary spending. Maintenance and repairs, on the other hand, are more likely to proceed, especially when the need is urgent. Deferred maintenance will accrue during this period, setting the stage for a renewed burst of activity following the pandemic.

Successful interior designers use programs that are based on BIM technologies. The apartment is assembled in 3D, like a constructor from walls, windows, doors and furniture, and then automatically get ready-made drawings based on the model.

1.2 Advantages and Disadvantages of the Existing Programs

1.2.1 The Professional Designer’s Tool

The one of the first program for the interior design was AutoCad. The first released was in 1982. It helps to create design project with all needed drawings. AutoCad can create any 2D drawing and 3D model or construction that can be drawn by hand. The program also allows the user to group or layer objects, keep objects in a database for future use, and manipulate properties of objects, such as size, shape, and location [2].

However, AutoCad is a program that is difficult to study for the ordinary user and it is not free. Also, the 3D view is not close to the look of the future room. There are a lot of symbols and notations that are hard to read at first sight.

The "giants" in the field of design are such companies as AUTODESK and GRAPHISOFT, and specifically their main BIM products - Revit and ARCHICAD. The main feature of working with BIM technologies is the ability to make corrections to one view of the model and observe how all related drawings and specifications will instantly be recalculated by themselves.

Revit

Revit is a single file database that can be shared among multiple users. Plans, sections, elevations, legends, and schedules are all interconnected, and if a user makes a change in one view, the other views are automatically updated. Thus, Revit drawings and schedules are always fully coordinated in terms of the building objects shown in drawings (fig. 1.1).

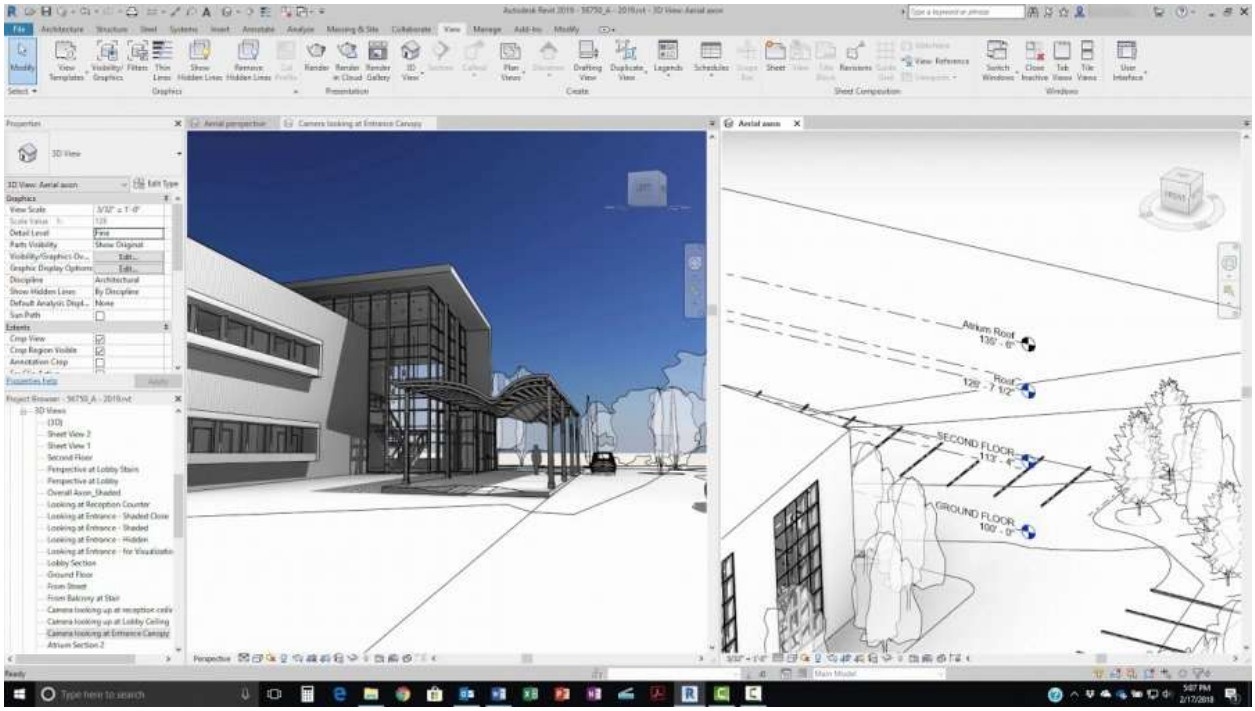

Fig. 1.1 Screenshot from the Revit program

The base building is drawn using 3D objects to create walls, floors, roofs, structure, windows, doors and other objects as needed. Generally, if a component of the design is going to be seen in more than one view, it will be created using a 3D object. Users can create their own 3D and 2D objects for modeling and drafting purposes.

Archicad

Archicad is a software tool for architects working in the architecture-engineering-construction (AEC) industry for designing buildings from the conceptual phase all through to the construction phase (fig. 1.2). It is a BIM software developed by GRAPHISOFT, enabling architects to work in a BIM (Building Information Modeling/Management) environment. Archicad works on two platforms: macOS and Windows - with no significant differences between them [3].

1.2.2 3D visualization software

Interior designer software for 3D visualization has come a long way in the past 10 years. There are some programs that are perfect for someone looking to make that transition and should be considered in all interior design offices.

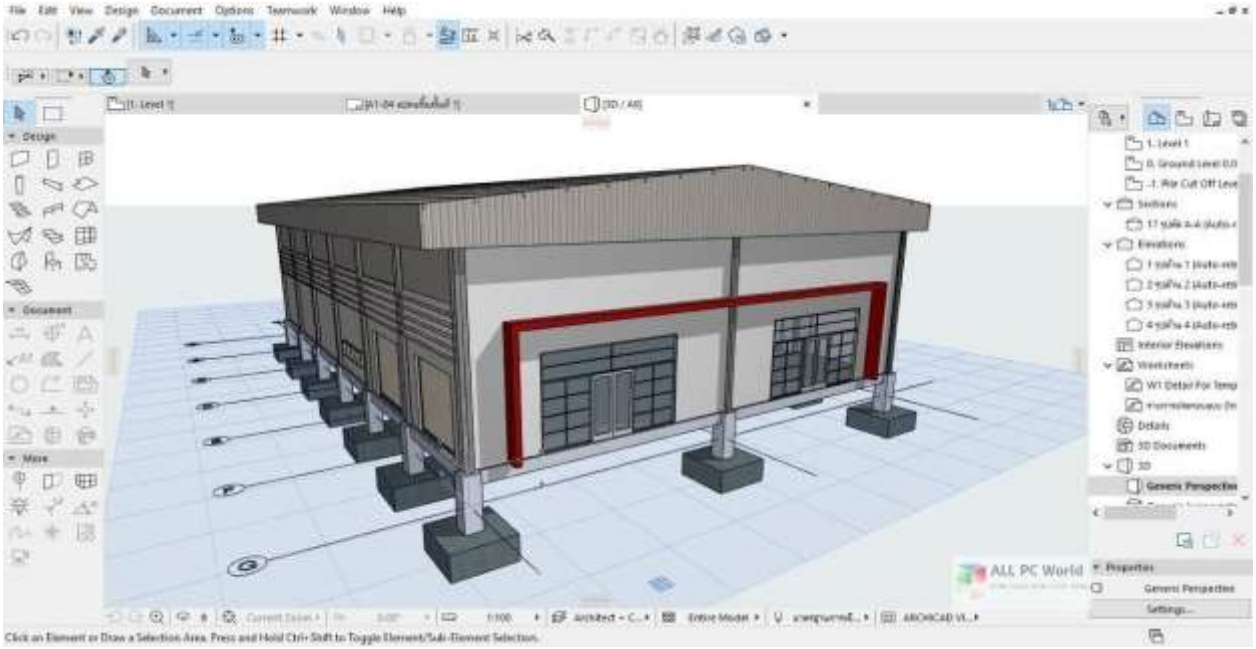

Fig. 1.2. Screenshot of from the Archicad program

SketchUp

SketchUp is a program for modeling relatively simple three-dimensional objects: buildings, furniture, interior. It was acquired by Google in May 2006 together with the small firm @Last Software. In April 2012, Google sold SketchUp to Trimble Navigation.

While not exclusively tailored to interior designers, SketchUp is a 3D modeling and visualization program that just about any designer can find value in. Not only is the learning curve as shallow as they come, SketchUp comes loaded with a vast suite of 3D models ready to be picked and added to the interior scene. Chairs, light fixtures, appliances, and textures are available to anyone who downloads the base program (which is free, by the way).

In a matter of hours, it is possible to have a beautifully realized interior model to start reinforcing the internal feedback loop and communicating with the client to figure out what is working and what is not. SketchUp works well with just about every piece of rendering software on the market.

Roomsketcher

Roomsketcher is a tool to create basic floor plans, interior elevations, and eventually fully realized 3D visualizations. This free 3D modeling and visualization program comes with preset plans, or the ability to quickly create a new one. Roomstetcher has one of the most wonderful features to move from 2D floor planning to a 3D representation of the space. The clients are impressed with the ability to translate their vision into something they can see on the screen, before they see it in their living room.

Planner 5D

Planner 5D was developed to be the one-stop-shop for interior design visualization. There aren't many particularly jaw-dropping features here, just an efficient and user-friendly design program that gives interior designers all the tools they need to visualize their work.

The tool allows to create professional looking drawings and diagrams that do exactly what they are designed to do: communicate with clients. Planner 5D is a great tool for the beginners. It is convenient and simple in use [4].

Ikea Home Planner

All three programs are not free and require special knowledge in design and software products themselves. There is one more tool that is free and do not require special skills. It is design by famous brand – IKEA. The application is called Ikea Home Planner (fig. 1.3).

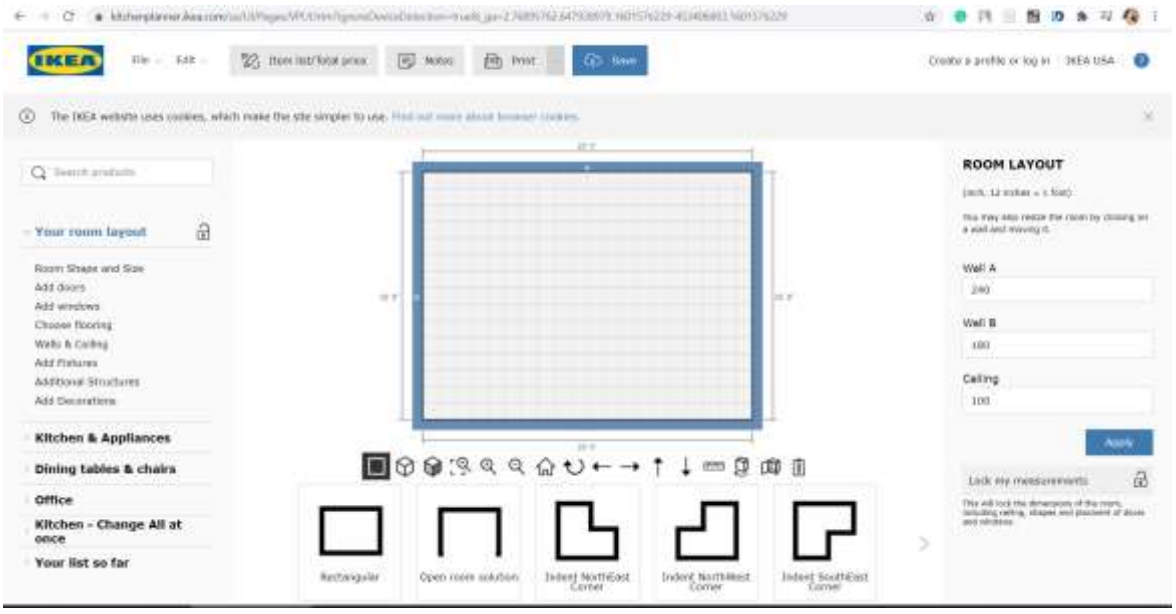

Fig. 1.3. Screenshot of the Ikea Home Planner web-application

It allows users to create dream kitchen, bathroom or office with the IKEA Home and Kitchen Planner. The user can save the plan to IKEA.com account and edit it later or access it from another computer. When the plan is complete, print it together with the product list and visit the local IKEA store to complete the purchase [5]. The IKEA Home Planner is only compatible with the Safari 7.0+, Firefox 35 and Chrome 40 browsers. The link to the web-application is https://kitchen-planner.ikea.com/us/UI/Pages/VPUI.htm. The user can also use the help of specialist or even use some existing design. And it is totally free for the user.

1.3 Problem Definition

The software application will be used by users who have a PC with an Internet access. Mobile devices and tablets will be not supported currently. The programming module helps users to calculate the number of rolls of the wallpaper, quantity of tiles, laminate or parquet. Also, in the future, the users can build the 3D model of theirs rooms they want to design. The catalog with items from the real shops will be added and the user can select the item from the catalog and the 3D model of item will be displayed in the user's 3D room.

The purpose of the application is to make it easy to calculate the number of materials needed and create the order. User do not need to worry how the wallpaper he or she many wallpapers it is needed to buy. The application can successfully calculate the quantity based on the room size entered by user.

Also, the customer does not need to go to the shop and waste a lot of time here. He can do this from home that is very important in the time of pandemic. In the future the possibility to add the 3D model of room and furniture will be added and the customer can fully design the room and review how the room will look like before buying all items.

In addition, there is no need to hire the professional designer because with this application anybody can be a designer. The application can be used even just for fun to practice the designing skills.

However, the professional designer also can find the application useful. For instance, they do not need to waste the time and search for the item that want to add to design in the real shop. The catalog will contain the real items from the real shop and the designer can just ordered item he or she need.

There will be two tariff plans for designers. Designers can use it for free if they use the application for creating design only, but they also can buy the ability to search client via the application. In this case designer's profile will be added to the Designers page. Then an ordinary customer can find a designer on the Designers page and contact him or her to create a design for his room online.

The one of the benefits of such communication that there is no need in offline meeting for customer and designer. A user can share the room size and shape, then designer will create planning for the user and share the solution after it is ready. Also, the user can start to create planning by his own and sent the invitation to the designer to work on the project together.

The primary objectives of the project are mentioned below:

- To fulfill the requirement for achieving the Master's degree of Computer Engineering;
- To know the fundamentals of the React.js Technology and Firebase Cloud System.

The secondary objectives of this project are mentioned below:

- to develop an application that helps any person to design the interior;
- to develop the convenient interface and management for the customer an admin users;
- to provide details information about the items in the catalog;
- to create an intuitive checkout page to create an order;
- to create an order confirmation page.

The program must be composed of two parts - the server, which represents the "backend" with the "firebase" cloud service, and the client, used "React.js". The server part should provide the catalog of all items and stored all customer's orders.

The client part should have an intuitive interface for viewing data from the server, downloading data, displaying information from queries, etc.

1.4 Functional Requirements

The first thing before building any system us to clearly declared the functional requirements of it. Definition of specific requirements is a crucial stage of a project. Such approach helps to reduce the development time and prevent a big number of bugs.

The web application should have the following parts:

- login and registration forms
- forgot password form
- page to update the user details
- tutorial page
- form to enter room size;
- catalog of wallpaper, laminate, tile and parquet;
- module to select the items from the catalog;
- cart page
- page to calculate the number of materials according to the room size;
- Checkout page to create an order
- Order confirmation page

In addition, the user should be able to change the selected items and deleted them from the order and the system should correctly recalculate the needed.

The system should also have the admin portal. The admin user should have the ability to add, update and delete the items in the catalog. The design module will be not displayed for him or her. Also, the admin user will see the orders of the customers and can manage them. The admin user should have the ability to ship the order, cancel the order and update the order.

1.4.1 User workflow

The user navigates to the site. There is a login form displayed. If the user does not have an account he or she can register. After login welcome page is displayed. The user can navigate to the any of the following pages: Tutorial, Calculations or Catalog.

On the Tutorial page a user can find information about the web-application and how to use it. Also, there will be some history about idealHome and contacts displayed.

The Room size form will be displayed after clicking on the “Calculations” block on the Welcome page. Here it will be possible to enter the room size which will be used in future to calculate the quantity of the items.

The Catalog page displays all items that are available to buy. A user can open the full page with all item details or just hover mouse over the item in the list and open the popup to review and add to the cart. After the item is added to cart the user can navigate to the Cart page and click on “Calculate” button near the item. It the user does not manage to enter the room size in the system the room size page will be displayed. After completing the form, a user will be redirected to the calculation page. Here he or she can select number of walls to apply the wallpapers and calculate the quantity of item that is needed. In the case if a user wants to calculate the laminate or parquet pages then there is no need in additional selections of the wall. The quantity will be calculated in the background and there will only results displayed with an update option to update the quantity of item in the Cart. For the tile item category, the form will consist of the list of walls and floor and the tile can be applied for the floor and walls as well.

The user can change the quantity of items in the cart at any time. After the design will be completed the user can go to checkout page, complete all required fields, and confirm the order. After completing the order, the order confirmation email will be sent.

1.4.2 Admin workflow

The user navigates to the Admin Portal and log in. The admin portal does not have the registration form. The user can be created by the Superadmin user of the Admin portal. The user can recover the password if he or she forgot it via the “Forgot password” option.

The Admin Portal consists of the three main tabs: Users, Catalog and Orders.

On the Catalog tab admin can add the item to the catalog, edit or delete item. The required fields to create an item should be the following: name, type, quantity, length, width, and item image. Also, it should be easily to search the items by name or type and sort them by adding date.

Mechanism of completing the admin operations for catalog:

Add wallpaper – this would consist of the function which will add new wallpaper to the database.

Add laminate – this would consist of the function which will add new laminate to the database.

Add tile – this would consist of the function which will add new tile to the database.

Add parquet – this would consist of the function which will add new parquet to the database.

Update item – this would consist of the function which will update the name, size and quantity of the selected item.

Delete item – this would consist of the function which will delete the item.

On the Orders tab admin can review all orders. Sort the orders by status and manage them. The admin should be able to review the order, cancel the order, update the orders and of course ship the orders. By default, only active (i.e. not shipped and not cancelled) orders are displayed on the Order tabs. Moreover, the admin user can create new orders. Such option is useful for instance in the case if the customer cannot create an order via the application and calls to the call center directly to make an order.

Mechanism of completing the admin operations for orders:

Add new order - this would consist of the form to create an order. The opened page should display a form to create an order. The fields should be customer name and surname, shipping address, fields for adding the items and quantity of the items.

Review – this would consist of the function which will be called to display the details of the customer's orders. The page opened should contains the proper information from the customer's checkout page: name, surname, address and list of the ordered items, method of shipping. The order can be opened by double clicking on the row in the list.

Edit – this would consist of the function which will be called to update any details of the customer's order. The admin user can change the number of items in the order or simple remove them, customer's personal details or shipping method.

Cancel – this would consist of the function which will be called to change the status of the order to Cancelled.

Complete – this would consist of the function which will be called to change the status of the order to Complete. The order is formed, and it is in process of delivering.

Ship – this would consist of the function which will be called to change the status of the order to Shipped. The order is successfully delivered.

The Users tab is visible only for the Superadmin user. It is divided into two tabs: Internal Users and Customers. On the Internal Users tab Superadmin can create, update, or delete (disable the login for the user) a user for the Admin portal. The Customers tab allows to add, review and disabled users from the User portal. In the future there will be design functionality to track the vacations of the workers. Also, the number of roles will increase to provide convenient and secure work for everyone.

Mechanism of completing the admin operations for users is the same for Internal Users and Customers tabs:

Add – this would consist of the form to create a new user. The fields of the opened form should be the following: name and surname, date of birthday, address, phone number, email. The form for creating Internal Users should have additional fields: hiring date, role, position, user role and status.

Edit – this would consist of the function which will be called to update any user details.

Disable – this would consist of the function which will be called to disable any selected user in the system.

Conclusions on the First Part

The interior design is very long and complicated process. It is difficult to imagine how the picture in the head will look like in a real room. However, only 19% of people hiring the professional designer to renovate the home according to the Houzz&Home survey in 2020.

Coronavirus Pandemic’s Impact: Subsequent surveys have shown that over half of homeowners who were in the midst of a project at the start of the pandemic were able to continue with renovations.

Successful interior designers use programs that are based on BIM technologies. The apartment is assembled in 3D, like a constructor from walls, windows, doors and furniture, and then automatically get ready-made drawings based on the model.

The one of the first program for the interior design was AutoCad. It helps to create design project with all needed drawings. AutoCad can create any 2D drawing and 3D model or construction that can be drawn by hand. However, AutoCad is a program that is difficult to study for the ordinary user and it is not free. Also, professional designers use Revit to creating plans, sections, elevations, legends, and schedules are all interconnected, and if a user makes a change in one view, the other views are automatically updated.

Archicad is a software tool for architects working in the architecture-engineering-construction (AEC) industry for designing buildings from the conceptual phase all through to the construction phase.

The purpose of the application design for the diploma work is to make it easy to calculate the number of materials needed and create the order. User do not need to worry how many wallpapers rolls he or she needs for the room. The application can successfully calculate the number based on the room size entered by user.

In future the idealHome can help designers to plan and implement the interior using just one program and do not waste time for real shopping to found desired items. All items will be saved in the catalog and available to buy because the application will be using real catalog of real shops.

PART 2
OVERVIEW OF USED TECHNOLOGIES, ANALYSIS OF DATA

To be successful, effective in the web industry it is essential to keep pace with modern technologies. There is no absolute response what tool is chosen, so as it really depends on set goals. Online business owners may face a problem: whether to choose a web application or a website for their future activities. At the same time, the online trading industry is growing rapidly. According to Statista portal, in 2021 the total online sales will reach 638,051 million US dollars (fig. 2.1).

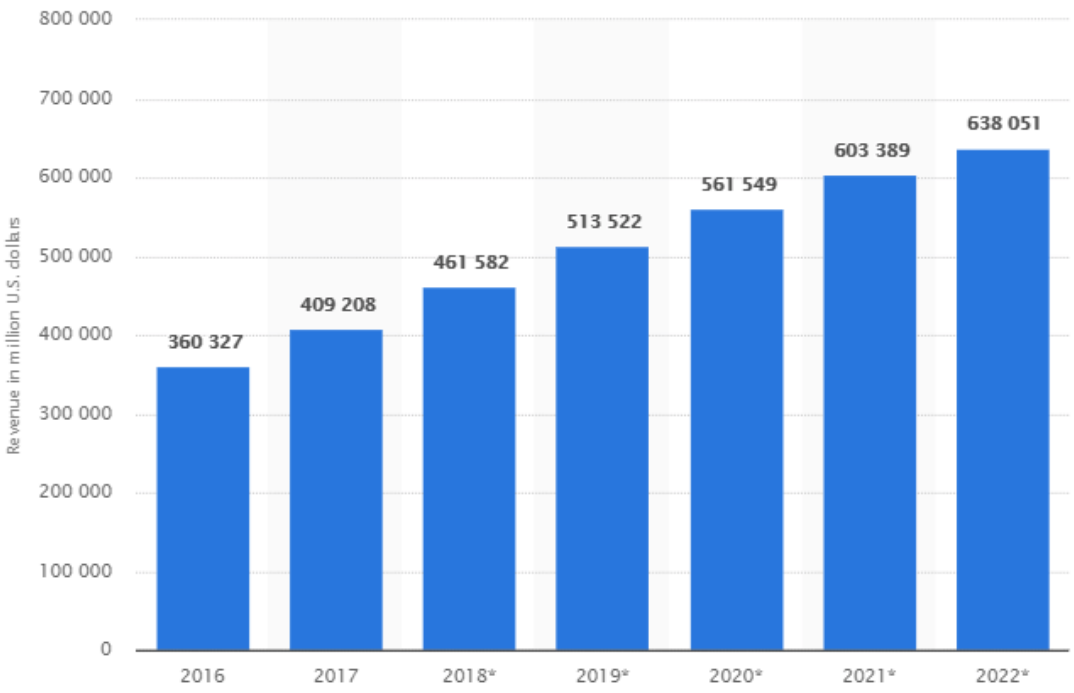

Fig. 2.1. Diagram of total online sales per year

A website is a group of interlinked and well-structured webpages that exists on the same domain. The main goal of the website is to display the information to the visitors. A web application is a software program that exists on the server and runs using a website browser through a web page.

It is created using a combination of the programming languages and web application frameworks. The main goal of the web application is to interact with the users and response for a user's request.

The future of the web application is progressive web application (PWA). A PWA is a web application built using specific technologies to achieve a given target. Targets are deciphered as follows:

- Reliable - the application is loaded and displayed immediately, regardless of the status and quality of the network connection.
- Fast - data exchange over the network is fast, the UI is smooth and responsive.
- Engaging - makes the user experience with the application comfortable and pleasant, encouraging him to want to experience it again, and again, and again ...

From Google's perspective, this is what separates the look and feel of websites from native apps now.

2.1 Analysis of Input and Output Data

To start working on any project firstly it is needed to analyze the data that will be stored and managed withing the application. In future it will be easier to design the database and project structure. In the design system there are a lot of data that can be filled in the form and stored in the database.

2.1.1 Input data

The project has two portals: Admin and User. Both have different tables with the users in the DB.

To sign up for the User portal a customer should provide the following data:

- Name and Surname
- Date of birth
- Contact details (email address and phone number)
- Password
- Shipping address: country, city, street, house and flat numbers, postal code (optional fields).

To sign in to the user portal customer should enter login (email address) and password. If a user forgot the password than he or she can restore it by entering the email address.

On User portal customer can enter the size of his or her room and the amount of selected item will be calculated for this customer. To do this a customer should enter the following data:

- length and width of the room in meter;
- height and length of each wall in the room in meters;
- numbers of doors in the room;
- heigh and width of the doors;
- numbers of windows in the room;
- size of the windows.

In addition, a user can create an order. To complete the order is the needed to select the desired item from the catalog, update the calculated quantity if it is needed and enter the shipping address and desired delivery date.

The user for the Admin portal can be created only by Superadmin user. To create a user, Superadmin user should enter the following data:

- name and surname;
- date of birth;
- contact details (email address and phone number) and address;
- position and hiring date;
- user role
- temporary password
- photo

The procedure to sign in is the same as for the User portal except the first login. A user should enter the email address and temporary password and then create a new password after that for his or her account. To do this it is needed to enter the new password two times. Also, a user can recover the account via “Forgot password” option. The email address is required to do this. Also, the Admin portal have Catalog section.

To create a new item is needed to enter the following data:

- item’s name;
- brand and country where the item was made;
- quantity;
- price;
- size (length and width);
- color (select from the list);
- image;

There are 4 types of products that can be added to the Catalog now: wallpaper, laminate, parquet, and tile. All of them have different form for adding to the catalog. The wallpaper form has additional field: wallpaper rapport. Parquet, laminate and tile forms have field to set the quantity of items in package. The number is autogenerated for the items. The last section of Admin panel is for Orders. Here a manager can set the order statuses: Cancel, Return or Shipped.

2.1.2 Output data

Each input data should have outcome messages. All possible events are described in table 2.1.

Tab. 2.1

Description of the output data

Income event	The output events
1	2
User Registration	Notification of successful registration or error is displayed
User login	Notification of successful login or error is displayed
Password reset	The password reset email is if there is a user in the system with such email address
User profile	The user information (name, surname, date of birthday, email, phone number, shipping address) is displayed
User profile update	Result of updating the user profile. The data is saved to the DB and notification of successful update is displayed.

Disable the user	Request to disable the possibility to log in to any portal
Request for calculation	The user set the room size and select an item from the catalog. The calculation results are quickly displayed for him/her

Continuation of table 2.1

1	2
Create an order	The order is created and recorded to the DB. The message is displayed
Add user for the admin portal	Notification of successful registration of new user is displayed. The email with temporary password is sent
New password for the first login	The new user is successfully created a new password and can log in with it
Add/update/delete wallpaper	The wallpaper is added/updated/deleted in the DB (message about successful operation or error is displayed)
Add/update/delete laminate	The laminate is added/updated/deleted in the DB (message about successful operation or error is displayed)
Add/update/delete tile	The tile is added/updated/deleted in the DB (message about successful operation or error is displayed)
Add/update/delete parquet	The parquet is added/updated/deleted in the DB (message about successful operation or error is displayed)
Request for viewing Catalog items	Result of getting Catalog items from the DB (message about successful getting or error)
Request to search the item in the Catalog	Result of searching item in the DB (message about successful searching or error)
Request for viewing the list of orders	Result of getting orders from the DB (message about successful getting or error)
Request to change order status	Notification of successful order update or error is displayed
Request to filter the Orders	Result of searching orders in the DB (message about successful searching or error)
Request to save the	The room size that is entered by customer on the User Portal is

user's room size	saved and recorder into the DB. The room's size is not available to displayed for the internal user of the Admin Portal. The size should be displayed on the Room size page after login once it was saved

2.2 Typescript Programming Language

Over 20 years after its introduction to the programming community, JavaScript is now one of the most widespread cross-platform languages ever created. Starting as a small scripting language for adding trivial interactivity to webpages, JavaScript has grown to be a language of choice for both frontend and backend applications of every size. While the size, scope, and complexity of programs written in JavaScript has grown exponentially, the ability of the JavaScript language to express the relationships between different units of code has not. Combined with JavaScript's rather peculiar runtime semantics, this mismatch between language and program complexity has made JavaScript development a difficult task to manage at scale [6].

TypeScript is an open-source programming language developed and maintained by Microsoft. It is a strict syntactical superset of JavaScript and adds optional static typing to the language. TypeScript is designed for development of large applications and transcompiles to JavaScript [7]. As TypeScript is a superset of JavaScript, existing JavaScript programs are also valid TypeScript programs. The first public release of the TypeScript was in October 2012 (at version 0.8) after two years of development at Microsoft. The last version of the typescript today is Typescript 4.0

TypeScript is the programming language that is compiled in JavaScript and is developed specially for the big applications. It takes a lot from the Java and C# languages that are stricter than dynamically typing JS. The code in TS is more readable and convenient to debug. Due to good support of OOP, modules and name spaces the project will be more structured event if it is a big one. The compiler can find more errors for the TS than for JS before they happen in the runtime and broke something.

The easiest way to install the TS is via the NPM. This is the packet manager of the Node.js and it should be installed before installing of the TS [8].

Application of static typing TypeScript with the indicated functions:

```
function add (left: number, right: number): number {
  return left + right;
}
```

Primitive types that can be victorious when annotated are number, boolean and string (numeric, boolean and row, apparently). To avoid the type checking in the Typescript the any type can be used. It is useful when the developer does not the type if the variable. Moreover there is one more type – void. It is opposite to any and can be used as a return type of function that do not return any value.

2.3 Technologies and Platforms Used for Creating the System

In web development, many different technologies and off-the-shelf products are used: these are programming languages and markup in a "pure" form, databases and other information stores, software products to simplify development (frameworks and site management systems), and server software. A modern website consists of the four main components: Interface, Software Part, Database and Hosting.

Interface. This is what the user sees when they visit the site. Implemented with HTML, CSS and JS.

Software part. These are algorithms that serve to process user requests. In fact, the software part creates or receives information from the database that the user requests, and receives and processes data from users. The software part is written in server-side programming languages and, as a rule, based on some development platforms.

Database. This is the storage location for the information that is used on the site. The database stores the content of all pages of the site and their relationship, user orders in online stores and much more. At the request of the software part, the database finds and gives the required records (for example, information displayed on a page) or records new data (for example, the content of an order).

Hosting. This is a server (computer) on which the software part of the site runs, and the database is stored. Hosting can be different; the speed and reliability of the site depends on its type.

The actual requirements for the components are quite predictable: the interface should be beautiful and convenient, displayed correctly in different browsers and on different devices; the software part must perform its tasks quickly and without errors; the database must reliably store information and quickly process requests and the server should work without interruptions, not slow down and withstand many simultaneous calls.

2.3.1 React.js Library

React is a declarative, efficient, and flexible TypeScript library for building user interfaces. It lets a user to compose the complex UIs from small and isolated pieces of code called “components”. React makes it painless to create interactive UIs. It designs simple views for each state in the user’s application and will efficiently update and render just the right components when the data changes. React was released in 2013 and quickly became popular among developers. Nowadays it is used by Facebook, Instagram, Trello, AirBnb, PayPal etc.

React has a few different kinds of components. It builds the encapsulated components that manage their own state and then composes them to make complex UIs. The components are written in Typescript, so the developer can easily pass reach data through the app and keep state out of DOM. The library has a lot of advantages:

- updates process is optimized and accelerated
- JSX makes components/blocks code readable. It displays how components are plugged or combined with.
- prompt rendering. Using comprises methods to minimize number of DOM operations helps to optimize updating process and accelerate it.
- reach has own tool for testing and debugging the code

However, React.js does not have the official documentation. It is developing too fast and there is no time to document everything. In addition, React library is too big for so little functionality [9].

There are few tips to create a React application. All components should be small. Such approach reduces efforts for the maintaining the code. Also, it is better to create stateless components. It is very hart to test it in the future because it is impossible to check each state separately. Of course, sometimes it makes sense for a particular component to totally own a particular piece of state. In which case, fine, go ahead use this.setState. It's a legitimate part of the React component API. For example, if a user is typing into a field, it might not make sense to expose every keypress to the whole app, and so the field may track its own intermediate state until a blur event happens, whereupon the final input value is sent outwards to become state stored somewhere else.

Working with React it better to use separate frameworks that can be used to control the states of the components. The most popular is Redux.js.

2.3.2 Redux Library

Redux is a small library that work with data. It helps developers to understand where and how the data should be stored. Redux was created in 2015 by Dan Abramov and Andrew Clark for the conference to prove the concept of Flux where a developer could change the logic. Here is the short description of the principles of Redux work:

1. Components are given callback functions as props and they call them when UI-event is happened.

2. Those callbacks create and dispatch action based on the event.

3. Reducers process the actions and computing the new state.

4. The new state of the whole application is located into one store.

5. Components receive the new state as a props and re-render themselves if needed.

The most part of the properties above are valid not only for the Redux but it has simple realization and tiny API. There are some advantages of the Redux:

- Reducer is a clear function that execute the following: oldStat + action = newState. Each reducer computes separate part of state which are united then. It simplifies testing of the business logic and states.
- API is smaller and has better documentation.
- If Redux is used as it should be only few components will be dependent from it, the other one will receive the states and callback functions as props.

There are some libraries that can complement Redux extremely well:

- Immutable.js – Immutable data structures in TypeScript. I it better to store state in it to avoid data mutating in future
- redux-thunk - this is used for when the actions need to have a side effect other than updating the application state. For example, calling a REST API, or setting routes, or even dispatching other actions.
- reselect - Use this for composable, lazily-evaluated, views into your state. For example, for a particular component it is needed to inject only the relevant part of the global state tree, rather than the whole thing or inject extra derived data, like totals or validation state, without putting it all in the store [10].

There is an alternative for the Redux. MobX is a standalone library for managing the front-end state of an application. MobX ensures the consistency and consistency of the internal state of the front-end application, providing convenient tools for changing it. Simplified, MobX allows you to implement the chain: "Action" → "Change state" → "Change view". In this case, changes occur atomically and automatically - as a result, it is guaranteed that there will not be a moment when the state is inconsistent.

Both libraries are used to manage state in JavaScript/Typescript applications. They are not necessarily linked to a library like Angular. They are also used in other libraries like ReactJs and VueJs. If one of the state management solutions is chosen, a vendor lock will be not encounter. A developer can upgrade to a different state management solution at any time. Also, a developer can switch from MobX to Redux or Redux to MobX.

Michel Weststrat's MobX is influenced not only by object-oriented programming, but also reactive programming. It wraps your state in observable objects. This way you have all the "Observable" capabilities in your state. Data can have simple setters and getters, but the observable allows updates to be received after the data changes. In Mobx, the state is fluid. This way it is possible to change the state directly.

2.3.3 React-Router Package

React Router is one of the most popular routing frameworks for React. The library is designed with intuitive components to let a user build a declarative routing system for the application. With declarative routing, it is possible to create intuitive routes that are

human-readable, making it easier to manage the application architecture. With routers, the user experience of the app can simplifying site navigation.

React-router package provides the core routing functionality for React Router. If an application will be run in the browser, the react-router-dom should be installed instead. Similarly, if it is a React Native application, it is needed to install react-router-native. Both of those will install react-router as a dependency.

To install the react-router package it is needed to execute the following in terminal:

$ npm install –save react-router

If the Redux is used to manage application states and React Router is used to do routing one more library need to be installed. The first two libraries do not coordinate. A user wants to do time travel with the application state but React Router does not navigate between pages when a user replays actions. It controls an important part of application state: the URL.

This library helps to keep that bit of state in sync with your Redux store. There is a copy of the current location hidden in state. When a developer rewinds the application state with a tool like Redux DevTools, that state change is propagated to React Router so it can adjust the component tree accordingly. A developer can jump around in state, rewinding, replaying, and resetting as much as he would like, and this library will ensure the two stay in sync at all times.

To install the react-router-redux package it is needed to execute the following in terminal:

$ npm install --save react-router-redux

This library allows to use React Router's APIs as they are documented. And, it is possible to use redux like a developer normally would, with a single app state. The library simply enhances a history instance to allow it to synchronize any changes it receives into application state [11].

history + store (redux) → react-router-redux → enhanced history → react-router

2.3.4 Algolia API

Algolia provides convenient RESTful API for the web sites and applications for instance search. Most of web-services and mobile applications such as Spotify, Salesforce

and Amazon should guarantee fast and useful access for the objects of the databases through the simple search window. People desire to find music and products in just a few keystrokes.

Algolia was founded in 2012 by Nicolas Dessange and Julien Lemoine in France. This technology is already successfully used all over the world by such companies as LVMH, Decathlon, Lacoste, Dior, Under Armor, Slack, Stripe, Discovery, Medium, Zendesk and many others. The platform's functionality includes:

- Search results, sorting and relevance. Customize search rankings based on a very wide range of criteria: typos, geolocation, filters, underweight words, word proximity, attribution, exact match, and more.
- Search rules. Setting up search results for specific user requests: display rules, boosting goods / products, fixing issue slots, etc.
- Instant search. Instant dynamic search as you enter text in the search box.
- Search personalization. Using many user attributes to customize search results.
- A/B testing. Instantly launch and test new experiments right from the platform dashboard to increase search conversions. The functionality allows you to understand which data you need to focus on, and which one to lower the weight when building search results.
- Search API. A powerful and convenient API enables quick integration of the solution and its subsequent customization, taking into account the business requirements and the specifics of the client's business.

The software module has a large database that is why the Algolia API is needed to quicky search the items and filter them. The Algolia Search API returns full results in less than 10ms on average, allowing to display results as a user types, just like Google, but 200 times faster than other search technologies.

Algolia is a SaaS solution. It has paid but has a free plan for usage. It can be integrated with Firebase with the help of official JS library. As a result of integration, a user gets a search index in Algolia which contains all objects from the Firebase. The main disadvantage is that SaaS is not free however for simple MVP applications the free traffic should be enough. At the end a developer obtains convenient admin panel with access to

analytics, search index and search queries peculiarities. An important plus is the presence of an SDK for everything and everyone - from mobile platforms to frameworks for the backend.

Also, ElasticSearch can be used instead of Algolia but is not easy in use for the beginners. In addition, it has open port that is used by Elastic Search for external queries as well as, for recording and management of search indexes. In future such feature can cause the deletion of the search index or adding data to it. Therefore, initially this port is open only for requests from the same machine on which ElasticSearch is installed [12].

2.3.5 Material UI

Material UI is a popular React framework. It can be installed via npm and take care about injecting the CSS where needed. Material-UI components work without any additional setup, and don't pollute the global scope. The first version of Material Design had one drawback that was important for brands that want to adapt it to their style. All Material Design applications were extremely similar. So, in 2018 Google launches Material Design 2.0. This version was an answer to the above criticism. With multiple styles, layouts, depth effects, cool animations, they are empowered to make truly unique branding themes.

Material UI can be integrated with Figma, Skech and Framer tools for UI design. The designer can use A set of reusable components for design tools and do not think about styling the components. However, the integration with Figma and Skech is not free but the beginner can user Framer and reduce time on creating mockups of the Reach application.

A developer can use default elements or customize them to align the style with the company identity (design system) and products [13].

The library can be used for small projects if developers are limited in time and it is better to focus on the business logic. It was design by Google to help teams build high-quality digital experiences for Android, iOS, Flutter, and the web. Material Components are interactive building blocks for creating a user interface, and include a built-in states system to communicate focus, selection, activation, error, hover, press, drag, and disabled states. Components cover a range of interface needs, including:

- Display: Placing and organizing content using components like cards, lists, and sheets.
- Navigation: Allowing users to move through the product using components like navigation drawers and tabs.
- Actions: Allowing users to perform tasks using components such as the floating action button.
- Input: Allowing users to enter information or make selections using components like text fields, chips, and selection controls.
- Communication: Alerting users to key information and messages using components such as snack bars, banners, and dialogs [14].

The color, typography, and shape of Material Components like buttons can be easily modified to match the brand. Material has an organized approach to apply colors to the UI. There are primary, secondary (for brands), surface, background, and error colors. In addition, there are complementary colors used for elements placed “on” top of it to promote consistency and accessible contrast.

Material Design has some disadvantages. It limits the effectiveness of other branding while using the design system. Yes, designers can incorporate logos, color palettes (within the Material Design guidelines), and other differentiating factors to support the brand identity, but a product following the Material Design specifications will almost always also be associated with Google.

Also, if any kind of animation is required for the product it is needed to incorporate motion in designs according to the Material Design specification. But it will increase the size of the product as well as the data usage for the customer.

Material UI has some analogs but all of them are not so powerful as Material Design. Eva Web is a customizable UI Library for Sketch that makes it easy to create awesome designs for web applications, admin platforms, and dashboards. With an ability to quickly adapt and integrate into your brand, Eva is an ideal design assistant. This version of Eva was developed specifically for web applications and contains many ready-made components that significantly speed up the design process. But it is not free and demo functionality is so limited [15].

Carbon Design System is an open source project from IBM's. The system consists of working code, design tools and resources, human interface guidelines, and a vibrant community of contributors. The design system is built React first. Also is supports core parts of the system in vanilla JS, Angular, Vue, and Svelte. If a developer is using a different framework, he still builds components by following the guidelines for other frameworks. Carbon Design System is similar to Material UI but the last one has better styling for the elements.

Atlassian is a software design language that provides design principles, code elements, and a library of UI assets in Sketch. Using clear guidelines (e.g., on typography, motion), pre-built templates, and presentation kit, you can design products faster.

Atlassian comprises various components including icons, fonts, buttons, badges, banners, pagination, tabs, tags, tables, spinners, etc. that help build outstanding experiences. It is noteworthy that 83% of Fortune 500 features companies use Atlassian products and there are 150,000 Atlassian customers in more than 190 countries

On September 12, 2017, Atlassian became a second product of the day and the fifth of the week on ProductHunt. By now, it has received 2,301 upvotes [16]. The disadvantages of this system is that it is limited to the Atlassian's products only.

2.3.6 Framer Design Tool

Framer is a free prototyping tool for the teams. Framer is an IDE where it is possible to build a prototype for the project for free. It runs Framer.js, an open-source Javascript library that creates the Layer abstraction, which is familiar to many designers. The Framer IDE is still release with new features each 40 days.

There are 4 main workspaces (fig. 2.2): Design tab, code tab, preview and present mode.

Design tab allows to create artboards and layouts.

Code tab adds interaction on created design

Preview displayed the results on the simulated device

Present mode allows to share the work with teams in presentation.

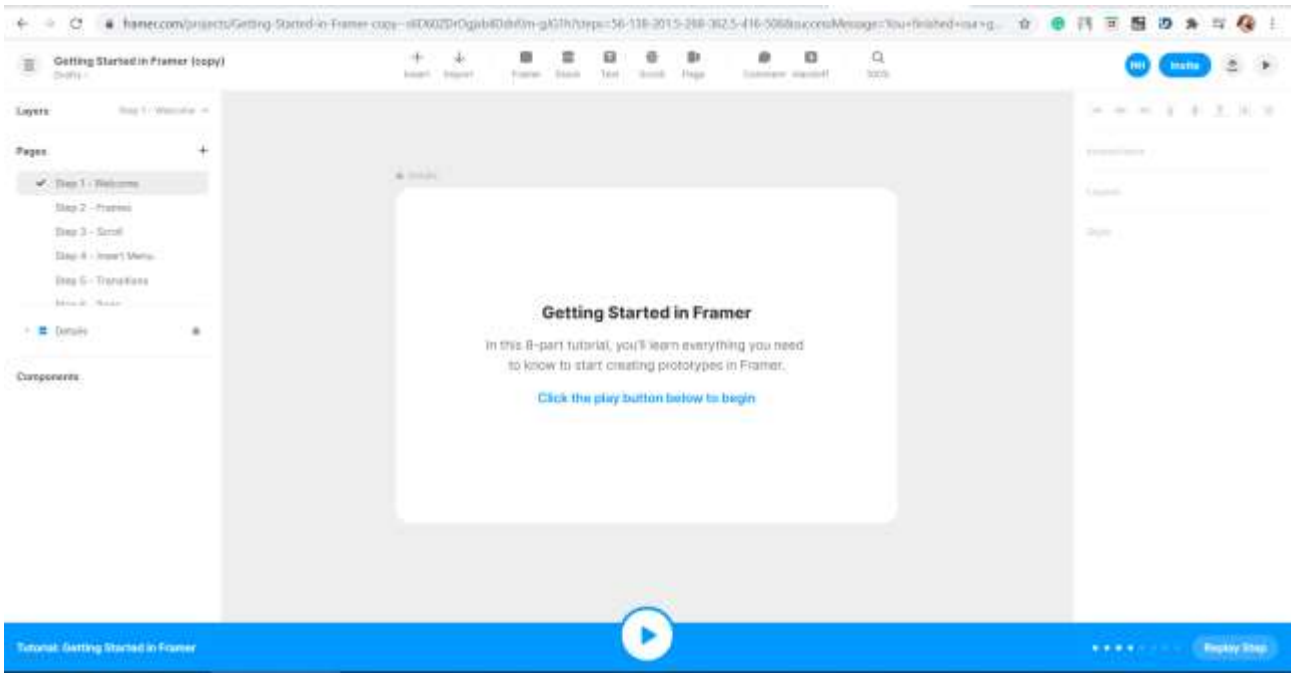

Fig. 2.2. Framer Interface

Framer provides creativity freedom because It inherits all visual features of modern CSS. Also, a user can open Framer on any device running in the web browser: desktops, smartphones, tables and smart-TVs. In addition Framer offer customers Framer modules integrated with After Effects and Lookback Integration and VR prototyping. Also, the Framer handoff feature provides specific production code for developers.

However, there are some disadvantages. Framer requires some programming skills and for many designers it seems unachievable reality. Many designers struggle to achieve code literacy, as tutorials often skip explaining basic rules of coding to get the right results. Also, interoperability with mainstream tools is pretty much absent, rendering every Framer effort as duplicated work. Framer has been implementing features to become a final destination like Sketch, but it is still a couple of laps behind. And the last but not the least Framer handoff and Export cover animations and styles but leave much of the developer information to be manually delivered. By focusing on internal spec generation, Framer looses desirable speed on delivering improvements into overall IDE [17].

There are few analogs of Framer. They are more powerful then Framer but they cannot integrate with Material Design for free. Figma is a graphic editor for web design. In Figma. It is possible to create interactive prototypes of sites and mobile applications; interface elements - icons, buttons, menus, windows, feedback forms; vector illustration.

In Figma, all documents are stored in the cloud. Thanks to this, in the editor, a developer can collectively work on layouts and open them by reference, without downloading.

A user can enter Figma through a browser or download the program to the computer. It is suitable for both Windows and Mac. In the desktop version, there is ability to work offline, and when access to the Internet appears, the changes are synchronized.

Sketch is a vector graphics editor for macOS developed by the Dutch company Bohemian Coding. It is used to design interfaces for mobile applications and websites. Supports the ability to create interactive prototypes. It was first released on September 7, 2010 and won the 2012 Apple Design Awards. Since 2016, Sketch switched to an annual license renewal. Within a year from the date of purchase, users receive program updates, after which they can continue to use the latest version published before the license expires, or renew their license to continue receiving updates for another year.

2.4 Database for the Web Application

Every modern web application uses one or more databases to store information. Databases provide tools for organizing, adding, searching, updating, deleting, and performing calculations on data. In most cases, the web application servers communicate directly with the job servers. In addition, each back-end service can have a corresponding database, isolated from the rest of the application. There are two types of the databases: SQL and NoSQL.

SQL stands for "Structured Query Language" (structured query language). It was invented in the 1970s. SQL databases store data in tables that are linked by shared keys. Such keys are usually represented by integers.

NoSQL stands for Non-SQL and is a newer set of database technologies. It was designed to handle very large amounts of information that can be generated by large-scale web applications. Most SQL variants do not scale well horizontally, and can scale vertically only up to a certain point.

The design module will be stored a very large amount of items in the Catalog. Also, there will be number of orders from customers and customer’s profiles. In addition, all

data are not deeply related among each other. That is why for the given project there will be used the NoSQL database.

2.4.1 Comparison of the Database Solutions

There were consider three different databases: DynamoDB from Amazon, Firebase Realtime Database from Google and MongoDB from MongoDB, Inc. All of them have advantages and disadvantages. The first two was released in 2012 and the last one in 2009.

DynamoDB is hosted, scalable database service by Amazon with the data stored in Amazons cloud. Firebase is cloud-hosted realtime document store. iOS, Android, and JavaScript clients share one Realtime Database instance and automatically receive updates with the newest data. MongoDB is one of the most popular documents stores available both as a fully managed cloud service and for deployment on self-managed infrastructure.

All databases have document store primary database models. In addition, DynamoDB has Key-value store.

DynamoDB and Firebase have commercial licenses while MongoDB are open source project. Also, the first two databases are only available as cloud servers. The MongoDB can be run also locally on Linux, OS X, Solaris and Windows. The DynamoDB supports the following programming languages: .Net, ColdFusion, Erlang, Groovy, Java, JavaScript, Perl, PHP, Python, Ruby. Firebase supports Java, JavaScript, Objective-C and MongoDB has the largest number of supported programming languages: Actionscript, C, C#, C++, Clojure, ColdFusion, D, Dart, Delphi info, Erlang, Go, Groovy, Haskell, Java, JavaScript, Lisp, Lua, MatLab, Perl, PHP, PowerShell, Prolog, Python, R, Ruby, Rust, Scala, Smalltalk, Swift.

The DynamoDB uses RESTful HTTP API access method only. The Firebase uses Android, iOS, Javascript API and RESTful HTTP API access methods. The MongoDB has proprietary protocol using JSON.

DymanoDB provides access rights for users and roles can be defined via the AWS Identity and Access Management (IAM). Firebase gives access control based on authentication and database rules and MongoDB grand access rights for users and roles.

The comparison table of three databases are displayed in table 2.2.

Tab. 2.2

Comparison of the databases

Name	Amazon DynamoDB	Firebase Realtime Database	MongoDB
1	2	3	4
SQL	no	no	Read-only SQL queries via the MongoDB Connector for BI
License	commercial	commercial	open source
Cloud-based only	yes	yes	no
Data scheme	schema-free	schema-free	schema-free
Server operating systems	hosted	hosted	Linux, OS X, Solaris Windows

Continuation of table 2.2

1	2	3	4
APIs and other access methods	RESTful HTTP API	Android iOS JavaScript API RESTful HTTP API	proprietary protocol using JSON
Server-side scripts	no	limited functionality with using 'rules'	JavaScript
Implementation language	-	-	C++
Triggers	yes	Callbacks are triggered when data changes	yes
MapReduce	no	no	yes
Consistency concepts	Eventual Consistency Immediate Consistency	Eventual Consistency, Immediate Consistency	Eventual Consistency Immediate Consistency

Foreign keys	no	no	no
Transaction concepts	ACID	yes	Multi-document ACID Transactions with snapshot isolation

2.4.2 Firebase Cloud Platform

Firebase is a database not a backend. Currently it is possible to use Firebase without server side, but this can cause a lot of problems in the future. Firebase is NoSQL database with all advantages and disadvantages. That is why Firebase does not support complicated SQL queries. NoSQL databases provide the mechanism for storage and retrieval the data that is different than tabular relations used in relational databases. To choose the solution for data storage it is needed to analyze the data and its structure. Firebase has the following disadvantages:

- its scope is much smaller than that of a NoSQL solution;
- Firebase severely restricts you when fetching data and, if necessary, writing data to several places at the same time;
- not all data structures are convenient to work with in Firebase.

The solution allows to quickly start the MVP development and has also a lot of advantages. Firebase has many options such as Realtime and Firestore that both are cloud-hosted, NoSQL databases flexible and scalable in terms of size. All data are stored in JSON format and synchronized for all connected clients in real time. In addition, Realtime provides developers with offline access and real-time updates that enable them to work on responsive apps without internet connectivity. The service allows deploy web applications in a few seconds. It provides static hosting of activities, usage of protected connection using SSL and network infrastructure CDN.

The platform doesn’t charge for most of its services and requires choosing a pricing plan only after reaching a certain amount of database memory. It’s great for beginners who want to validate if the platform is good enough for their product and don’t want to pay for all the services upfront. For those who want to estimate the total price they need to pay for the customized plan, there is a price calculator that ease the process.

Firebase has well-prepared technical documentation that eases the work with offered services and makes them more accessible for users. A user can find all the necessary information about integrations, availability and supported technologies. What's more, there are about 1,5M apps around the world based on Firebase. It means that the community around the product and the number of resources will only benefit developers trying to find an answer to any problem.

Finally, the ease of integration and quick setup. The pre-made APIs that the platform offers enable you to add new functionalities within just a few clicks. Firebase requires little to no technical knowledge to start working on your product. The simple UI makes it possible to implement functionalities like authentication to your app without major problems. Working with Firebase there is no need for complex configurations, so nearly anyone can set up the app – web or mobile [18].

Firebase has 4 methods of recording data to the Firebase Realtime database: set, update, push, transaction support.

Set – write or replace the data in the set destination like messages/ users/<username>.

Update – update some keys for define path without replacing all data.

Push – add to the list of data in the database. Each time when new node is push to the list the database generate new unique key like messages/users/<unique-user-id>/<username>. The unique key is based on a timestamp, so list items will automatically be ordered chronologically.

Transaction support – it is used when working with complex data that could be corrupted by concurrent updates.

The basic database write operation is a set which saves new data to the specified database reference, replacing any existing data at that path. The data for the app is stored at this database reference:

```
// Import Admin SDK
var admin = require("firebase-admin");
// Get a database reference to the blog
var db = admin.database();
```

```
var ref = db.ref("server/saving-data/fireblog");
```

To save a user object user's username, full name, and birthday to the database the set() / setValue() methods can be used. The code for this is listed below:

```
var usersRef = ref.child("users");
usersRef.set({
  neteli: {
    date_of_birth: "January 27, 1998",
    full_name: "Natalia Hryshko" },
  chyter: {
    date_of_birth: "February 6, 2005",
    full_name: "Vlad Hryshko"
  },
  referi: {
    date_of_birth: "March 8, 1988",
    full_name: "Natalia Anderson" }});
```

When a JSON object is saved to the database, the object properties are automatically mapped to database child locations in a nested fashion.

To update the saved data update() method can be used. It is used when it is needed to write to multiple children without overwriting the other child nodes. In addition, the Firebase Realtime Database also supports multi-path updates. The usage of it is shown below:

```
usersRef.update({
  "neteli/nickname": "Tine Smile",
  "chyter/nickname": "Vladko"
}
);
```

In addition, the Firebase Realtime Database has a security language that lets you define which users have read and write access to different nodes of your data. You can read more about it in Secure Your Data.

The firestore database structure is similar to object in Javascript programming language. In addition to managing the database, the management of privacy, analysis and authentication are provided.

There is an application control panel with a database structure is shown on the figure 2.3.

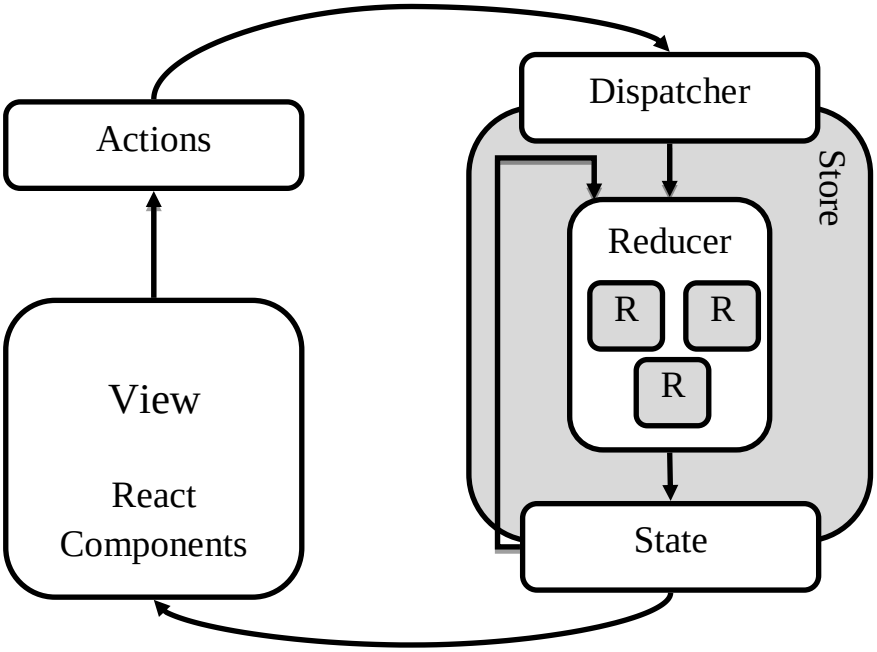

Fig.2.3. Redux data flow

2.5 Description of the Development Environment

A simple HTML page with some CSS styling can be easily created in any text editor. But the larger the project is, the harder is to maintain everything in the text editor. So, there are a lot of IDE and extensions for convenient web development. Using a quality HTML editor is critical for building respectable websites. Below there will be considered some of them.

Sublime Text 3 is a flexible, free IDE for Windows, Mac, and Linux. It supports number of different programming and markup languages, including Python, C, HTML, JavaScript, and CSS. The interface is known to be clutter-free and fast. Sublime Text includes the following features:

- code autocompletion - snippets and macros;
- can be tweaked to boost productivity;

- enhanced pane management;
- go to definition;
- go to symbol;
- multiple selections;
- command palette;
- split editing;
- instant project switch;
- customizable key bindings, menus, snippets, and more

The disadvantage of the Sublime Text 3 is that it is not free. It has some functionality available for free, but the user can often see the annoying popup to buy the application. Also, Sublime does not have the graphic interface for changing the theme and settings. Sometimes developer can see the crashes of the application due to installing some third-party plugins.

Atom is an IDE for JavaScript programming. It was created by Github and works with Mac, Windows and Linux. Atom is highly customized but it possible to use it with default settings. Atom helps to find, preview, and replace text as the user types in a file across all projects, easily browse and open files and projects in one window. Also, it has teletype tool that enables collaboration with other developers from within the editor. In addition, Atom is very customizable and hackable.

Atom uses telemetry by default. It can be turned off, and unlike VS Code, there are no extensions restrictions, so it is not needed to change the workflow due to privacy concerns. Atom is definitely a useful tool for developers. Moreover: it is basically suitable for anyone who uses a computer. It can be used as a good general-purpose text editor.

However, Atom is very slow to run. It is very inconvenient if it is just need to update few lines in the code. Typically, text files over 10MB crash or freeze, making it less useful as a regular text editor.

Visual Studio Code was the most popular in 2018 survey by the State of JS. It was design by Microsoft but it not the same as Visual Studio. The IDE comes with built-in support for JavaScript, TypeScript, and Node.js. It also has plenty of extensions for other languages (such as C++, C#, Python, and PHP). Developed by Windows, Visual Studio

Code is great for new programmers as it explains everything from HTML tags to syntax and error handling. VS Code has many features. It supports:

- syntax highlighting;
- autocomplete with IntelliSense based on variable types;
- function definitions;
- imported modules;
- custom hotkeys;
- integrating with GitHub;
- IntelliSense, which provides smart completions based on variable types;
- customizable themes etc. [19].

VS Code functionality can be extended via different plugins. They can be installed withing the VS Code and there is no need to search them though the internet and manually install and connect with the IDE. Initially, VS Code was created as an open source project. However, all it builds are shared under Microsoft license. However, the user can build his own version which is called “Code-OSS”. The difference between VS Code and Code-OSS is very small. VS Code works with telemetry which mean that Microsoft can collect the data on how the product is used.

That is why Visual Studio Code IDE was selected for development of this project. The screenshot of the VS Code interface is displayed below in figure 2.4.

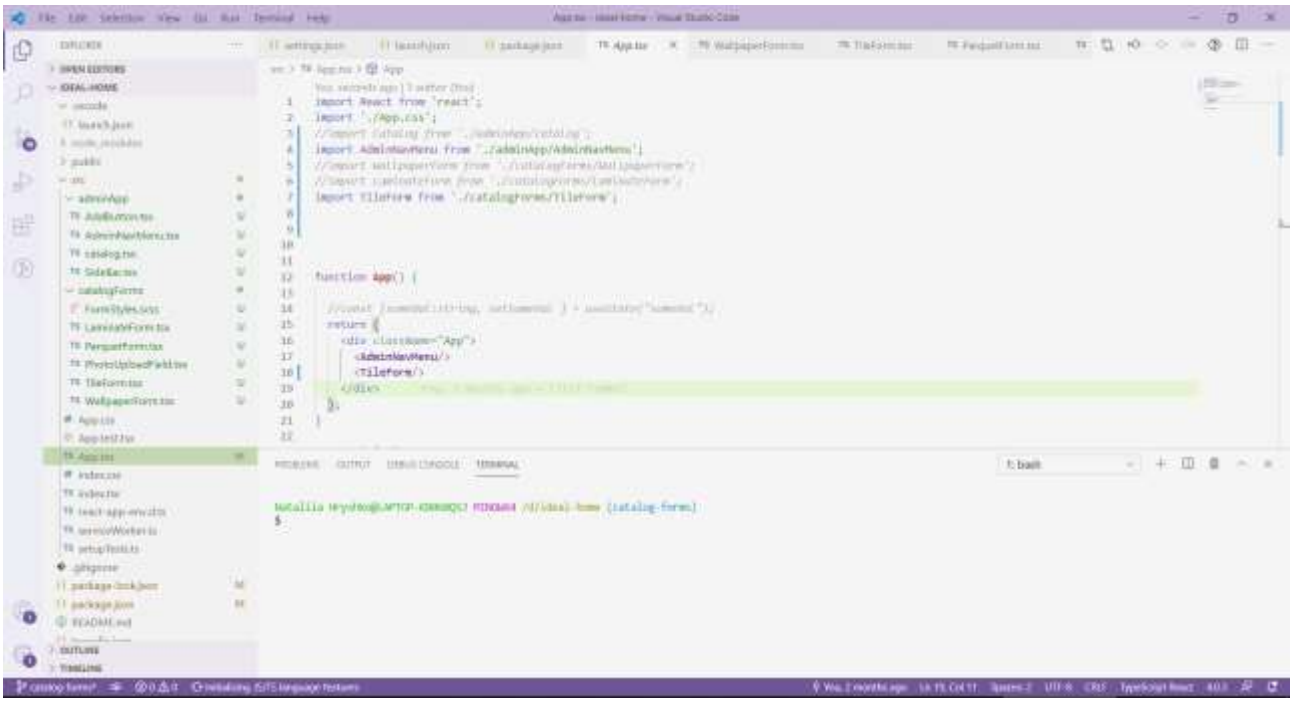

Fig. 2.4. Screenshot of the Visual Studio Code interface

VS Code also has some disadvantages, but they are pretty small. It is similar to all other electron apps; VS Code's memory and battery usage is pretty bad. Better plug in when use it. It doesn't have git merging, which many people have come to expect as Sublime Text 3 and Atom have the ability to do that. In addition, the universal search in the application is not user friendly and is a little bit confused.

Conclusions on the Second Part

A web application is a software program that exists on the server and runs using a website browser through a web page. It is created using a combination of the programming languages and web application frameworks. To start working on any project firstly it is needed to analyze the data. In this part there were analyzed input and output data that will be stored and managed within the application.

It this part there were described technologies that will be used in the development. The product will be developed in the VS Code environment. The programming language is TypeScript. TypeScript is an open-source programming language developed and maintained by Microsoft. It is a strict syntactical superset of JavaScript and adds optional

static typing to the language. Also, to simplify the development process React, Redux and React-Router libraries will be used.

React is a declarative, efficient, and flexible TypeScript library for building user interfaces. It lets a user to compose the complex UIs from small and isolated pieces of code called “components”. Redux is a small library that work with data. It helps developers to understand where and how the data should be stored. React Router is one of the most popular routing frameworks for React. The library is designed with intuitive components to let a user build a declarative routing system for the application. With routers, the user experience of the app can simplify site navigation.

For the backed side there were selected Algolia. Angolia provides convenient RESTful API for the web sites and applications for instance search. The software module has a large database that is why the Algolia API is needed to quicky search the items and filter them. The Algolia Search API returns full results in less than 10ms on average. For the database the Firebase database will be used.

To speed up the UI styling Material UI framework will be used. A developer can use default elements or customize them to align the style with the company identity (design system) and products. To design the mockups for the application Framer was selected as prototyping tool. Framer is an IDE where it is possible to build a prototype for the project for free.

PART 3

WEB-APPLICATION INTERFACE AND LOGICAL DESIGN

3.1 Application Interface Design

An interface is a contract between a system and an external environment. Interface on any system should be understandable, convenient and user friendly. User should be able to understand whether his data was saved, processed, or delete successfully. Also, if there were any errors occurred it should be displayed on the screen for the user. The error text should clearly describe the problem.

3.1.1 Interface of the Admin Portal

There are two portals in the system: Admin and User portals. Both of them have login screens. The login page for the Admin portal is displayed in figure 3.1

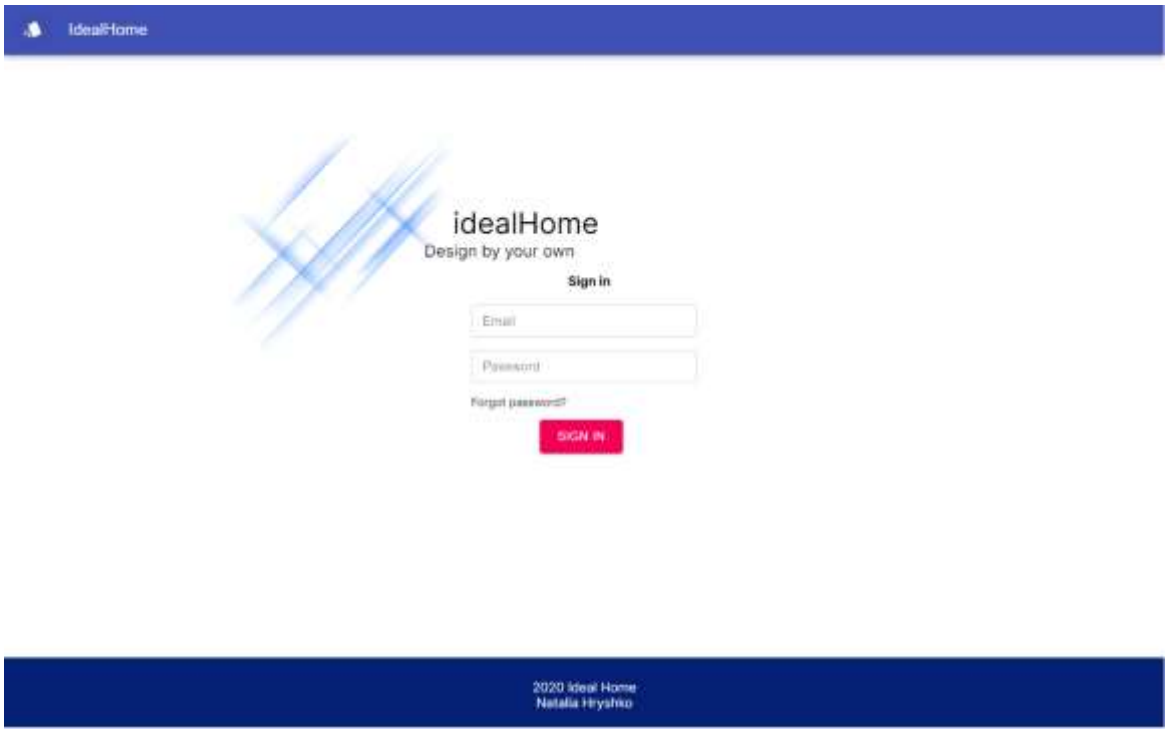

Fig. 3.1 Screenshot of the Login form of Admin portal

A user of the admin portal cannot sign up by his own. The Superadmin user should create an account for him. After the account is created the email will be sent to the user with temporary password and the user should change the password after first login. The

window for password change is displayed in the figure 3.2. The same window will be displayed if a user forgot the password and clicedk on the link from the password recovery email.

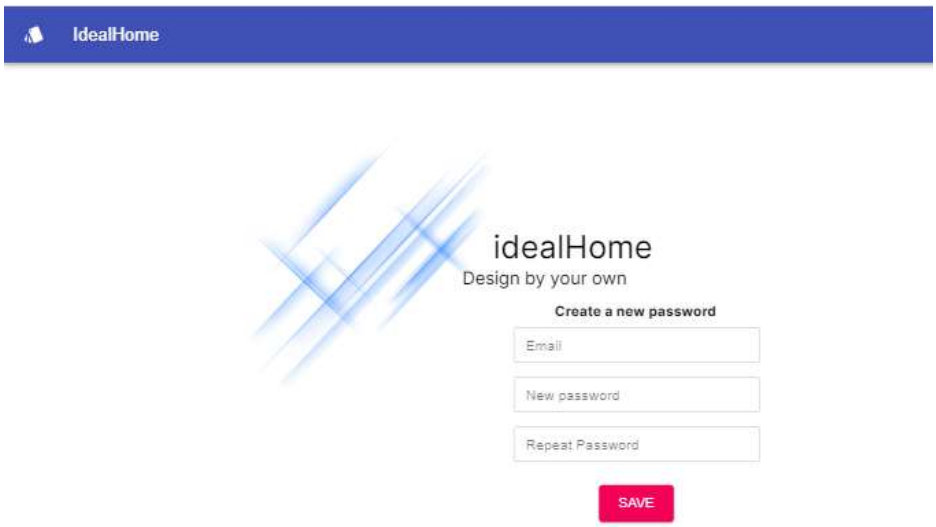

Fig. 3.2. Screenshot of the form for creating new password.

If the admin portal user forgot the password, he or she can recover it via clicking on the ‘Forgot password?” option on the login page. The window that will be displayed is shown in figure 3.3.

Fig. 3.3. Screenshot of password recovery screen

After successful login there is a Catalog page displayed. The screenshot of the catalog page is displayed in the figure 3.4.

The main responsibility of this page is to manage the items in the Catalog: add, edit and delete from the list. Currently there are four types of items in the system: wallpapers,

tile, laminate and parquet. So, the drop-down list for the “Add” option was designed (fig. 3.5).

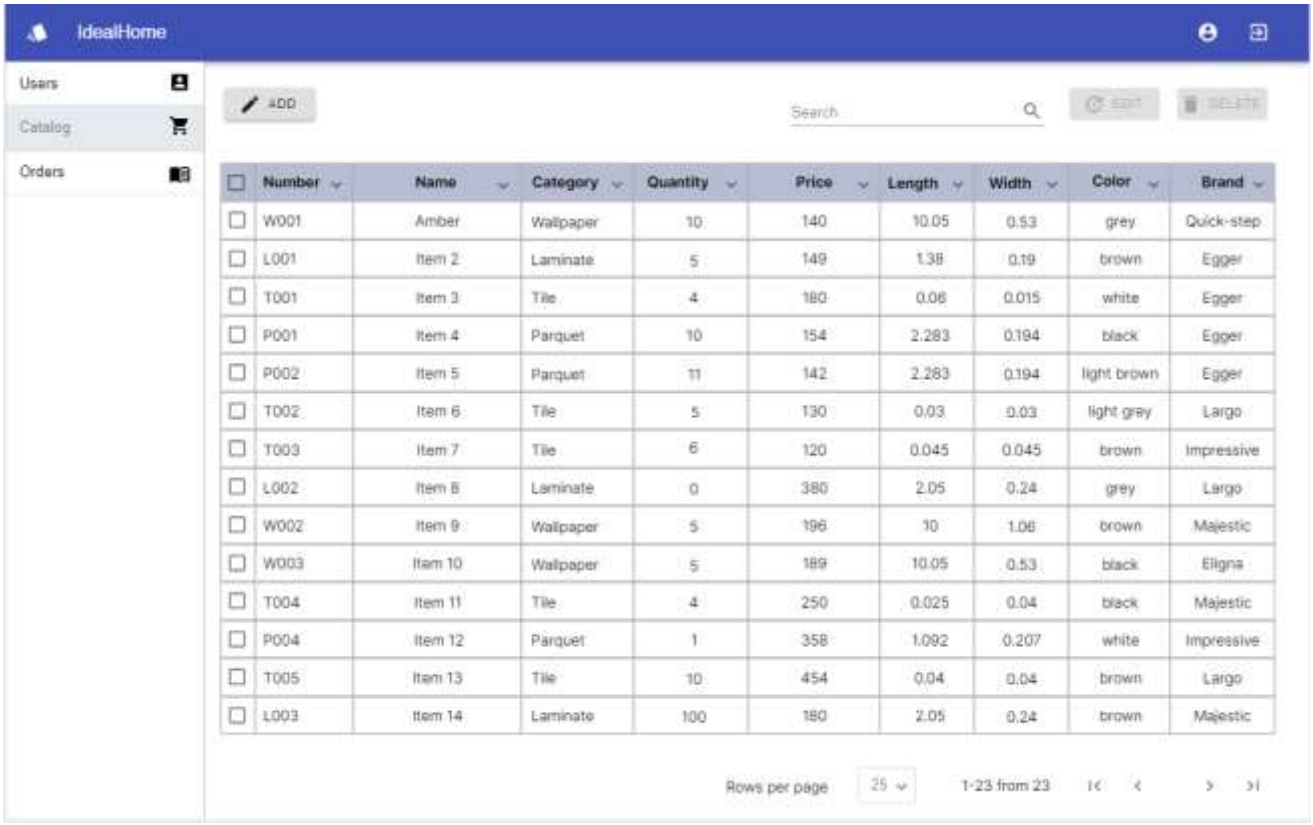

Fig. 3.4. Screenshot of the Catalog screen

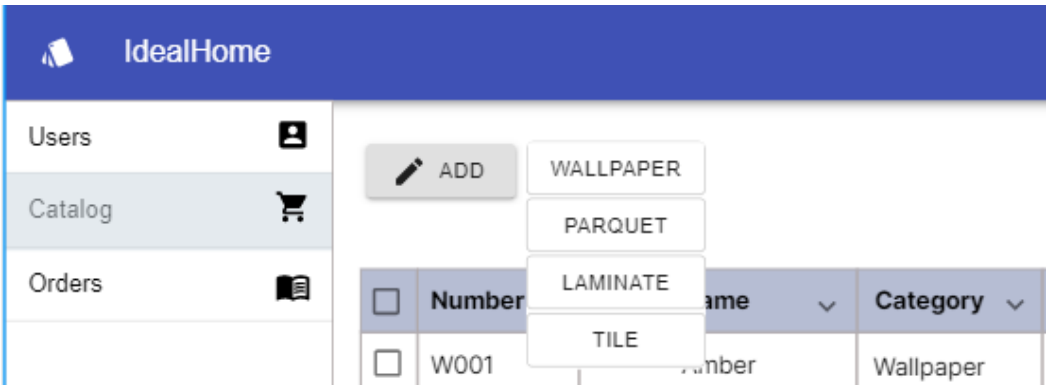

Fig. 3.5. Drop-down list for the Add option of the Catalog page

When a user clicks on the “Add” button and selects Wallpaper then the form for adding a wallpaper is displayed in new page. The Number field is prepopulated (for Wallpaper the number starts from W and follows with the number of wallpaper in the catalog). The user can save a new item and close the window via “Save and Close” option or click “Save and add a new one” to save the item and open a new form for the new wallpaper. The example of filled form is displayed in figure 3.6. The “Save and Close”

and “Save and Add New” options are disabled until all required fields are filled. For the user convenience save buttons are added at the top and bottom of the window.

After a user adds images then they are displayed in the Image preview section. The image that was uploaded as the last one is autoselected in the Preview section and zoomed in.

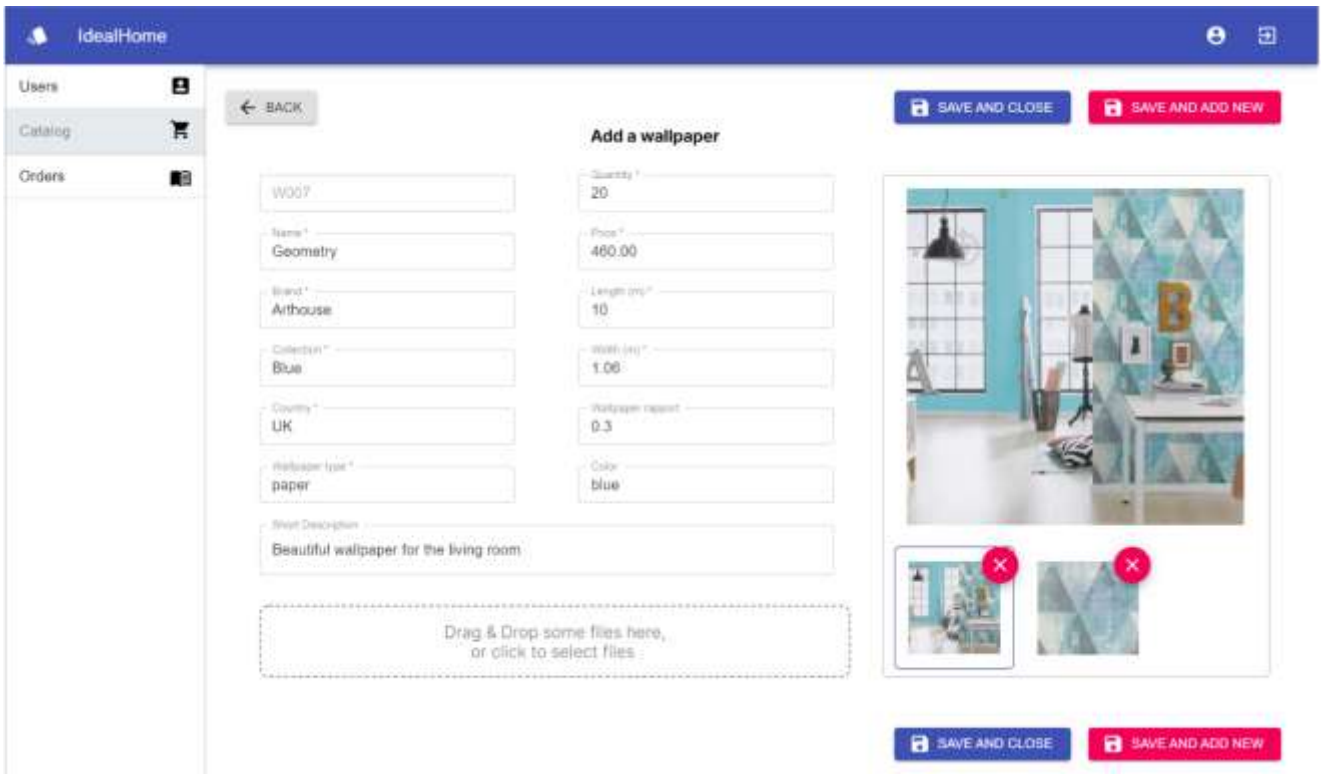

Fig. 3.6. Wallpaper form

The design of forms for adding tiles, laminate and parquet are the same as for the wallpapers. However, they have more fields to fill. The form for adding the items is the same as for adding but it has different header (for instance, the form for editing the wallpaper item has “Edit a wallpaper” header), and all item details are prepopulated and can be edited except the number field.

The catalog contains a lot of items, so the pagination was design for convenient navigation among pages (fig. 3.7).

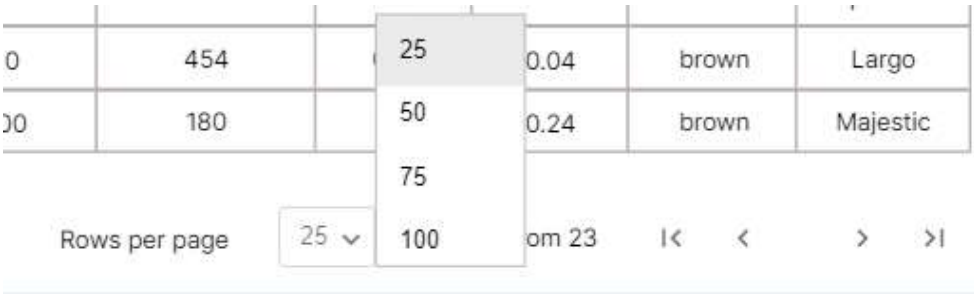

Fig. 3.7. Pagination design of the Catalog page

The maximum number of items on the page should be not more than 100. The user can navigate to the next or previous page or to the last or to the first page via the pagination buttons.

The Edit and Delete buttons should be disabled by default. The Edit and Delete button should be enabled if a user selects an item from the catalog (fig. 3.8). The Edit button will be disabled if there were multiple items selected. The Delete button should be enabled for multiple items selected (fig. 3.9).

Fig. 3.8. One option selected in the Catalog

Fig. 3.9. Two options selected in the Catalog

Orders tab (fig. 3.10) is designed to manage the orders from the customers and create a new one if it is needed. It can be useful when the client call to the internal user and order the items through the call center.

To create an order, it is needed to click on the “Add” button. The “Create an order” form will be displayed (fig. 3.11). To update the order and/or change the order status it is needed to click on the “View Details” option. In addition, there is a search field available and the customer can search the order by order number, name and surname of the customer, order status, order total or the order creation date. The pagination is designed the same as for the Catalog page. The maximum number of items on the page should be not more than 100.

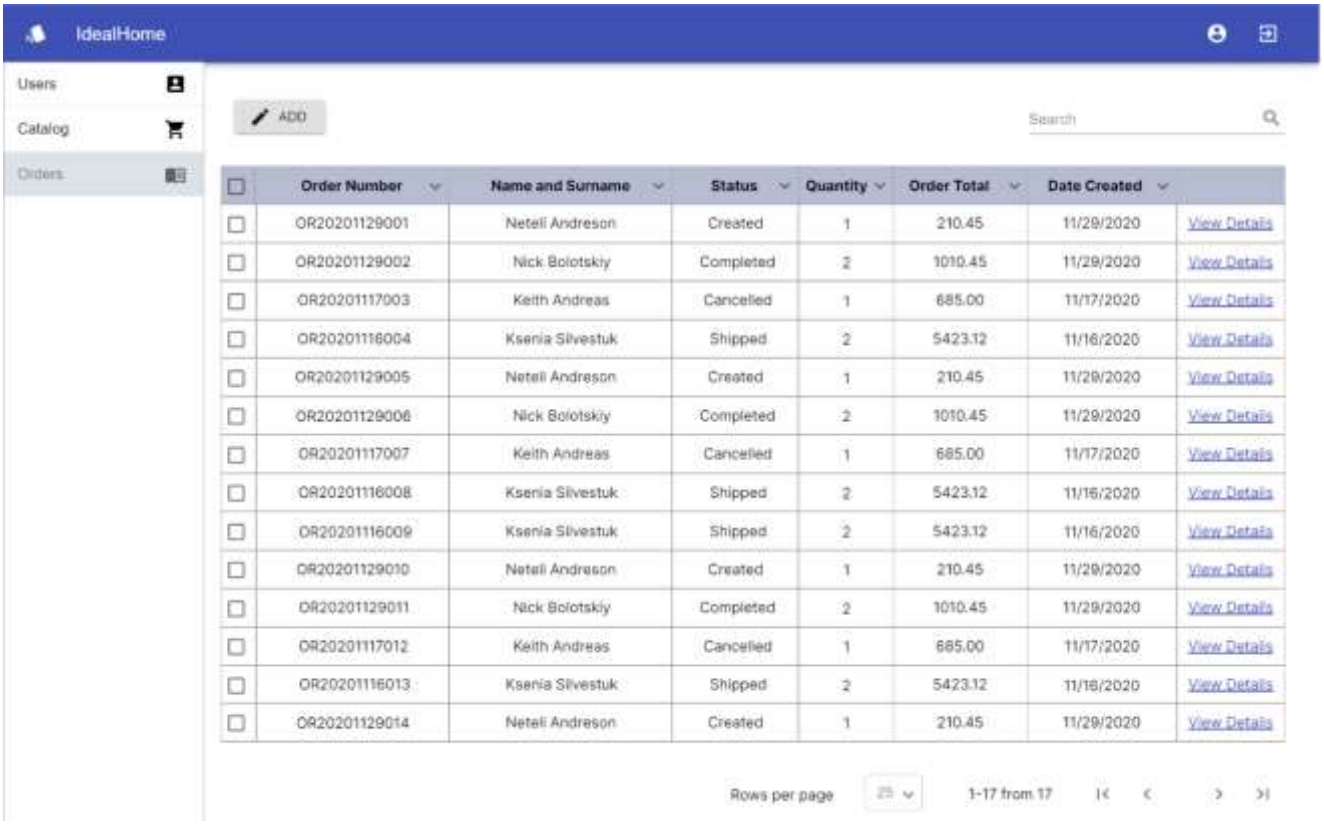

Fig. 3.10. Admin Orders Tab

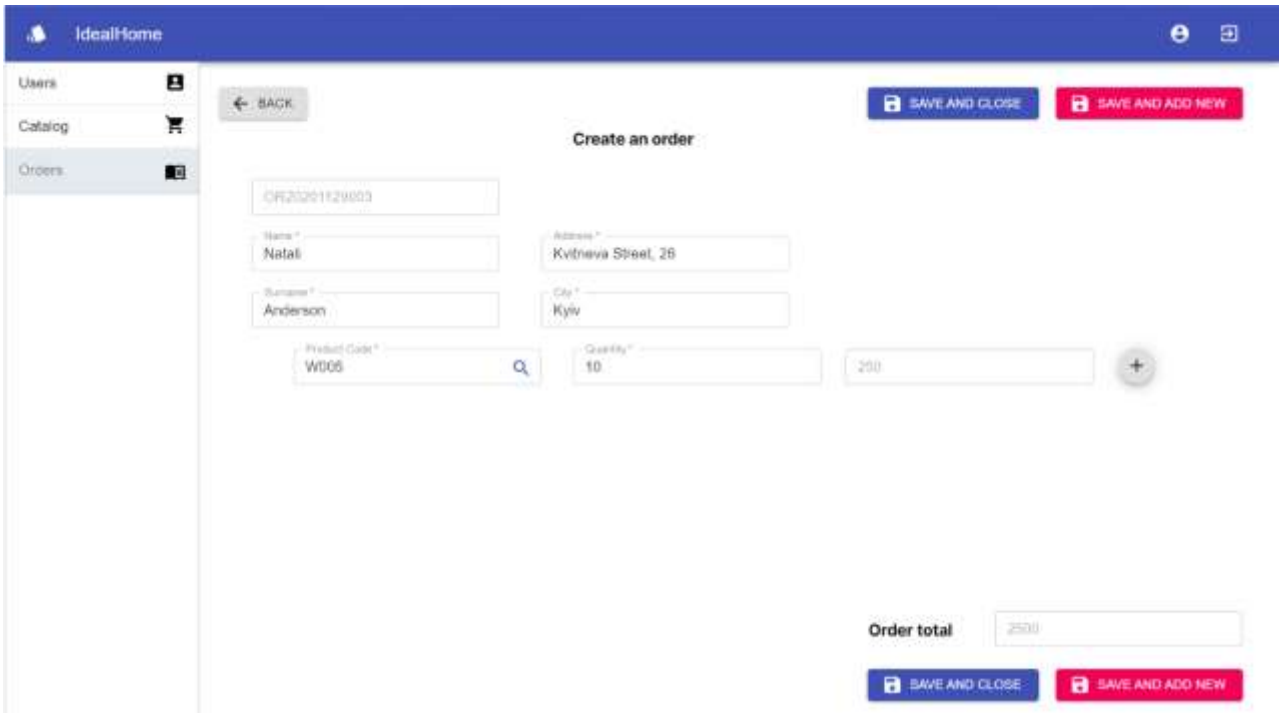

Fig. 3.11. Create Order form

The form for order editing is displayed in figure 3.12. The “Ship” order button is enabled only for the orders in “Completed” status. Completed status means that the order is ready for delivery. The status can be changed during updating the order.

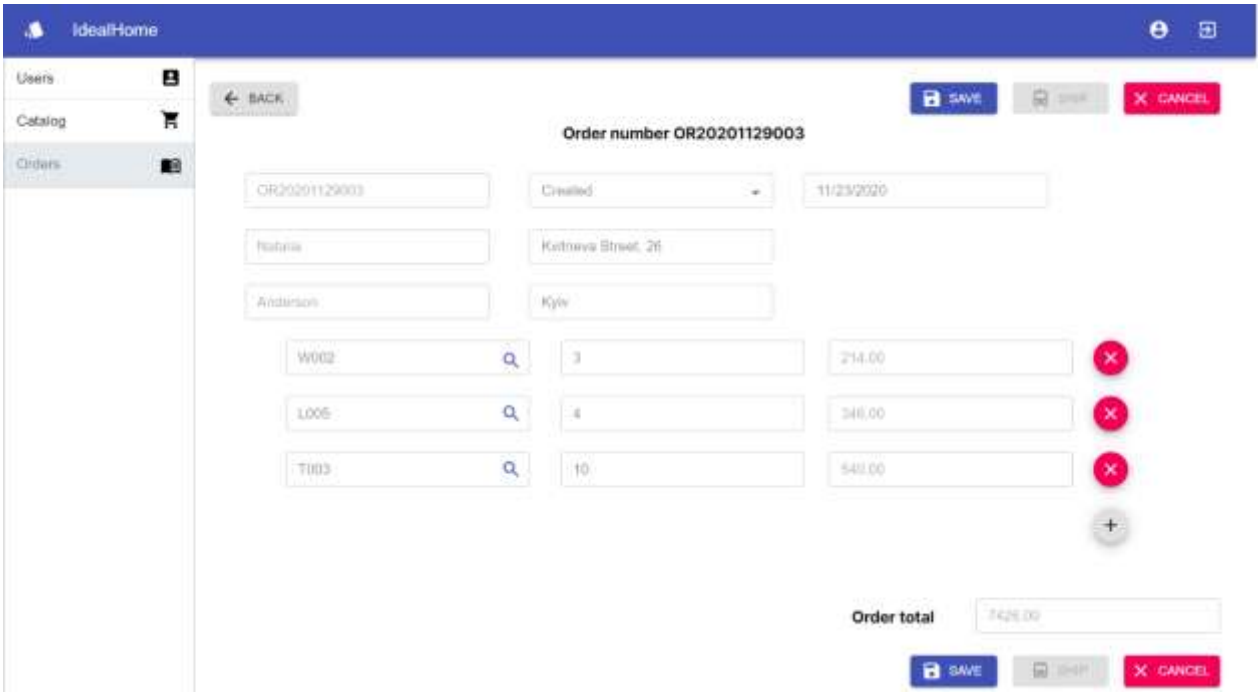

Fig. 3.12. Update Order form

Users tab is designed for administration of the users in the system. It can be accessed by the Superadmin only. The Users tab should be hidden is a user is not a Superadmin. There are two options within the Users tab: Internal Users and Customers. The Internal users tab is designed for managing the internal users that can access the Admin Portal (fig. 3.13).

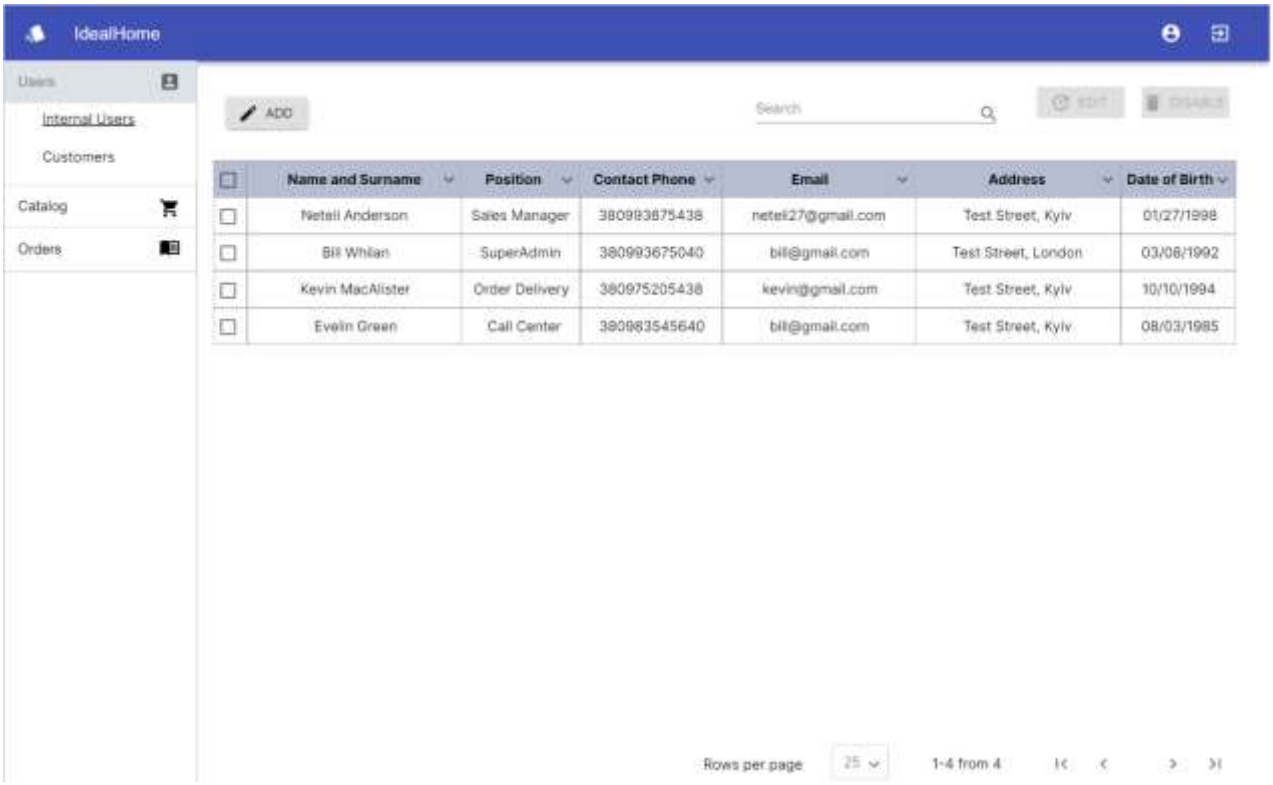

Fig. 3.13. Internal users tab in the Admin Portal

At the bottom of the page there is a pagination design for convenient navigation among users. The Edit and Disabled button are disabled by default. The Edit and Disabled buttons will be enabled if any checkbox near the user will be enabled. The Edit button should be displayed if there are multiple users selected.

Customers tab is designed to manage the users of the User Portal. The design is similar to the Internal Users page. There is only one difference the Position field is replaced by City field. Here it is possible to create a new customer, update existing ones and disabled if it is needed. The Customers tab has the same functionality as Internal Users tab for the Edit and Disabled buttons (fig. 3.14-3.15).

Fig. 3.14. One option selected on the Customer tab

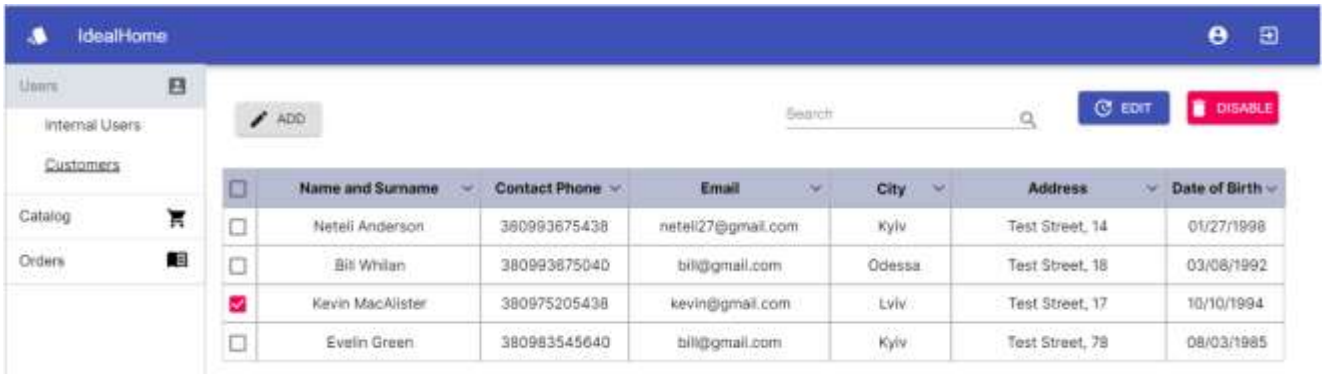

Fig. 3.15. Two options selected on the Customer tab

The form for Creating an internal user is displayed in the figure 3.16. The form for updating the user is pretty the same. There are several user roles for a user: Superadmin and Manager. The user status can be Active or Disabled. In addition, the user that is not a Superadmin user can edit his own profile after clicking on the Profile icon at the top of the page. The form for adding a new customer does not contain the following field: position, hiring date, user role and status. Also, it is not needed to upload a profile picture.

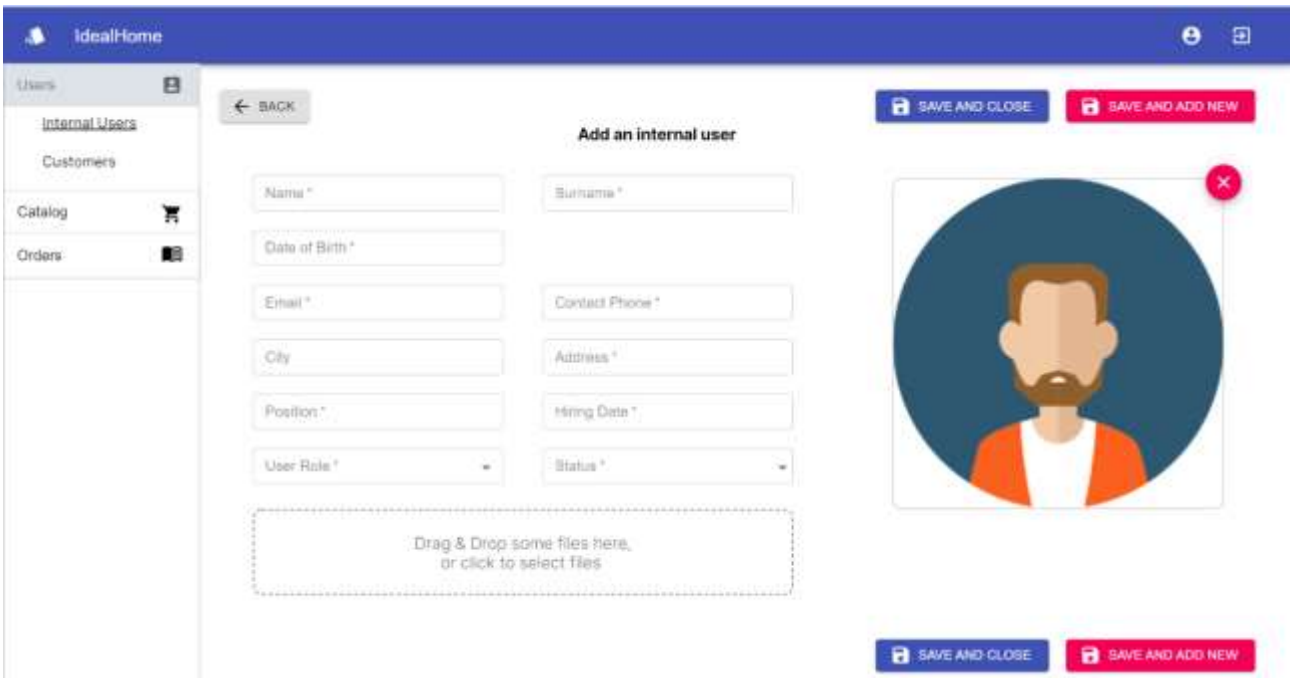

Fig. 3.16. Internal user adding form

3.1.2 Interface of the User Portal

User Portal is designed for the user to create orders and calculate the needed quantity of materials from the catalog. The access a catalog the user should be registered. The registration and login forms are displayed in the figures 3.17 and 3.18.

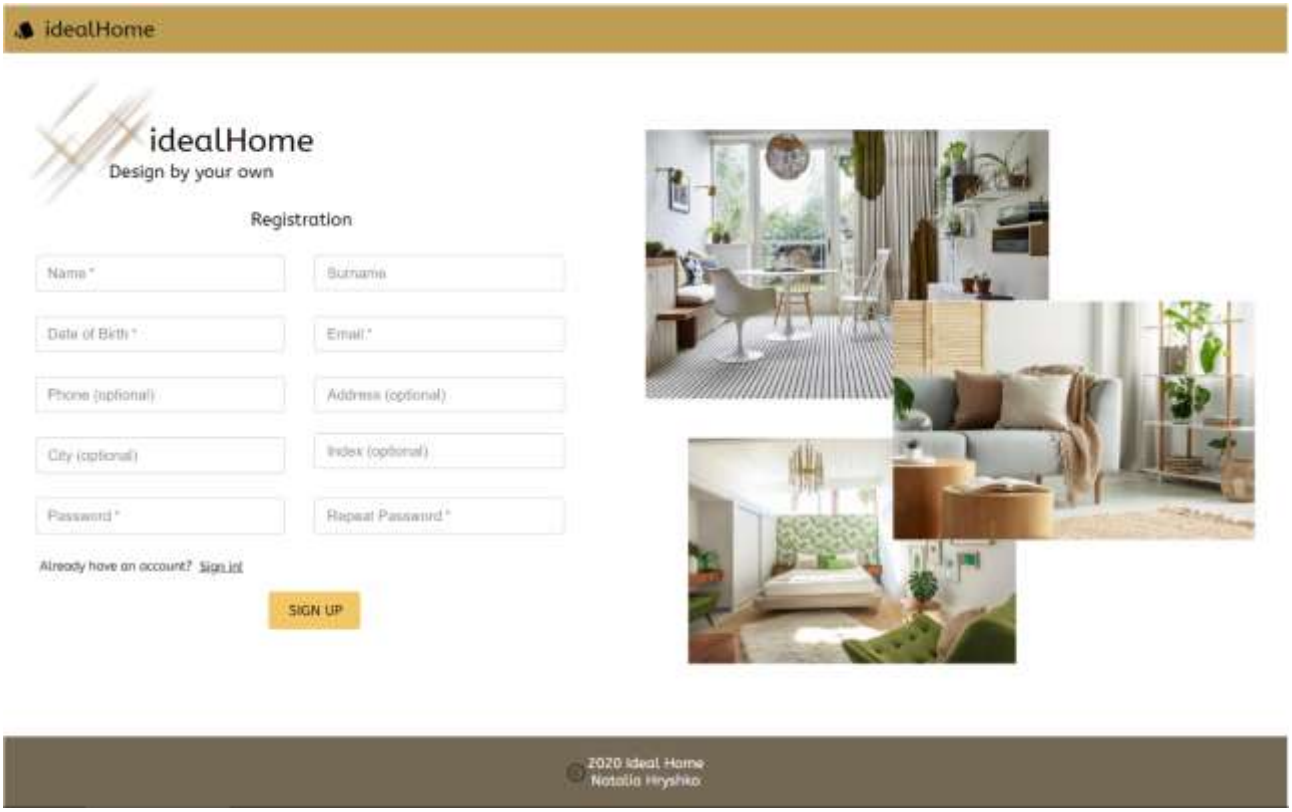

Fig. 3.17. Registration form of the User Portal

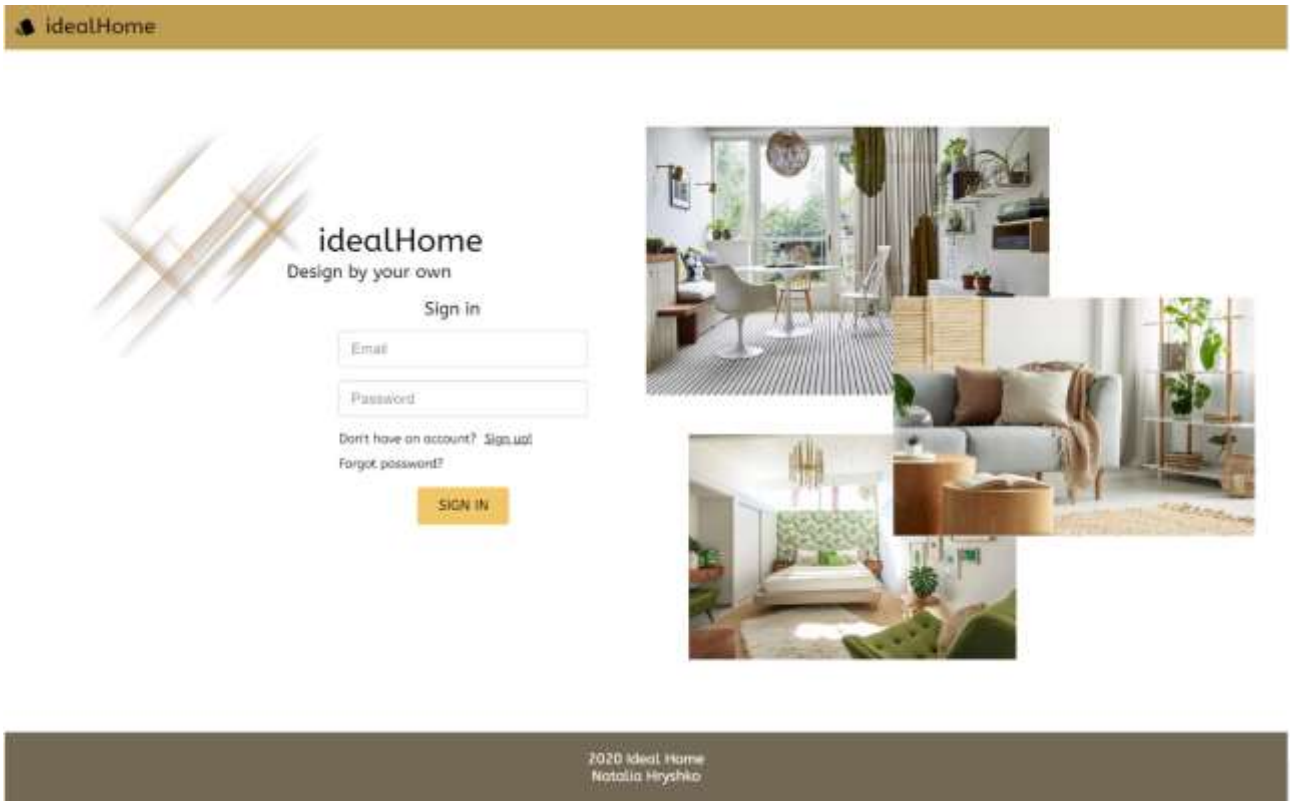

Fig. 3.18. Login form of the User Portal

Similar to the Admin portal a user can recover his or her password via “Forgot password?” option. The design of the forms for password recovering is displayed in figures 3.19-3.20.

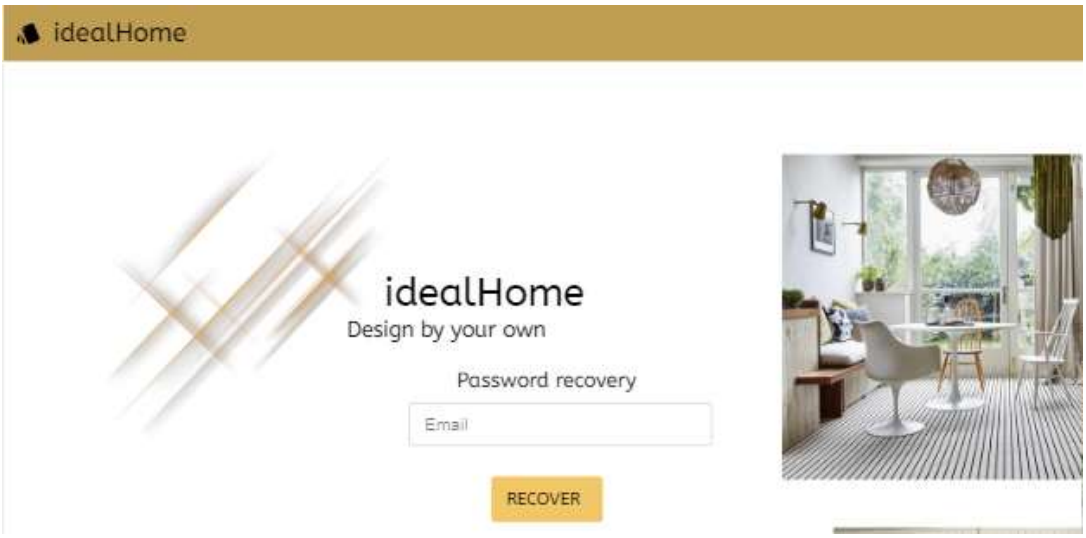

Fig. 3.19. Screenshot of password recovery screen

After the user logged in to the User Portal the Welcome page is displayed. The Welcome page is shown in figure 3.21. There are three options to navigate from the user welcome page: Tutorial, Calculations and Catalog.

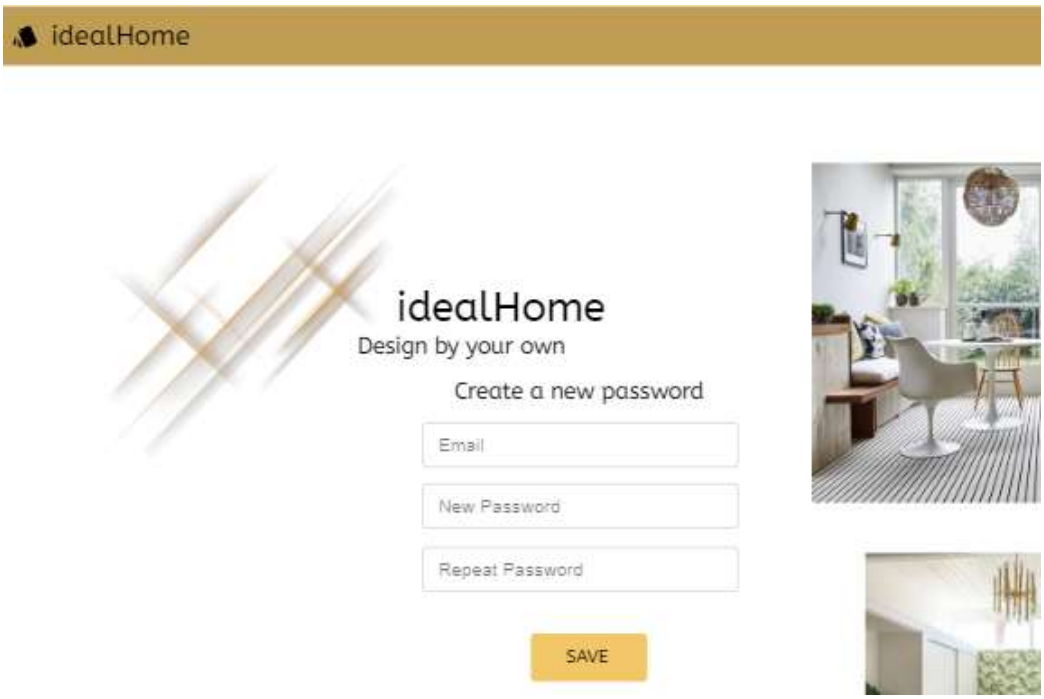

Fig. 3.20. Screenshot of creating new password on the User Portal

Fig. 3.21. Welcome page of the User Portal

The Tutorial page is a page about idealHome and user manual on how to use the portal. On the Calculations tab the user can enter the room size and calculate the amount of material which he or she selected from the Catalog. The best way to use the web application is to enter the room size and then select the desired items from the catalog. The Catalog tab displays all available items: wallpapers, laminate, parquet and tile. It is possible to review the page with item details and add it to the cart.

Please note that the application calculates only the number of items that is needed for the room that was added to the Cart.

The Calculations tab has two sections: Room size and Cart. The first one is a form for adding the room parameters (fig .3.22). The user can adds size of room, size of wall, adds the doors and windows parameters and set their quantity via the "+" button.

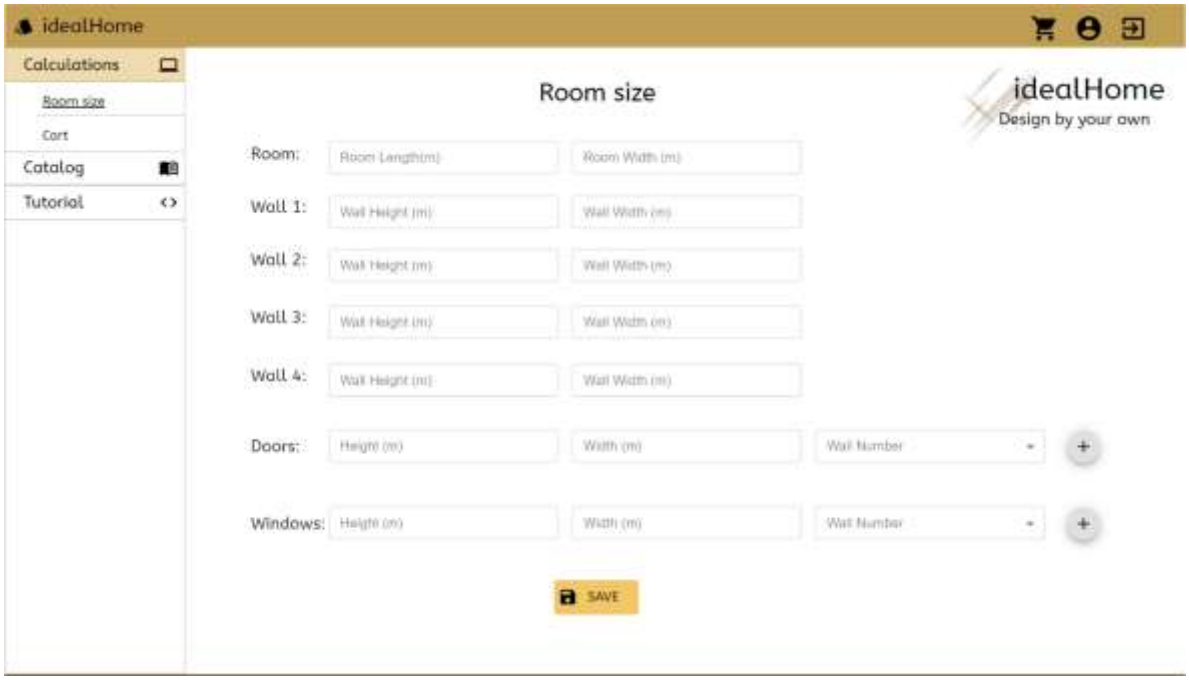

Fig. 3.22. Room size form of the Calculations tab

On the Tutorial page (fig. 3.23) a user can find information about the web-application and how to use it. Also, there will be some history about idealHome and contacts displayed.

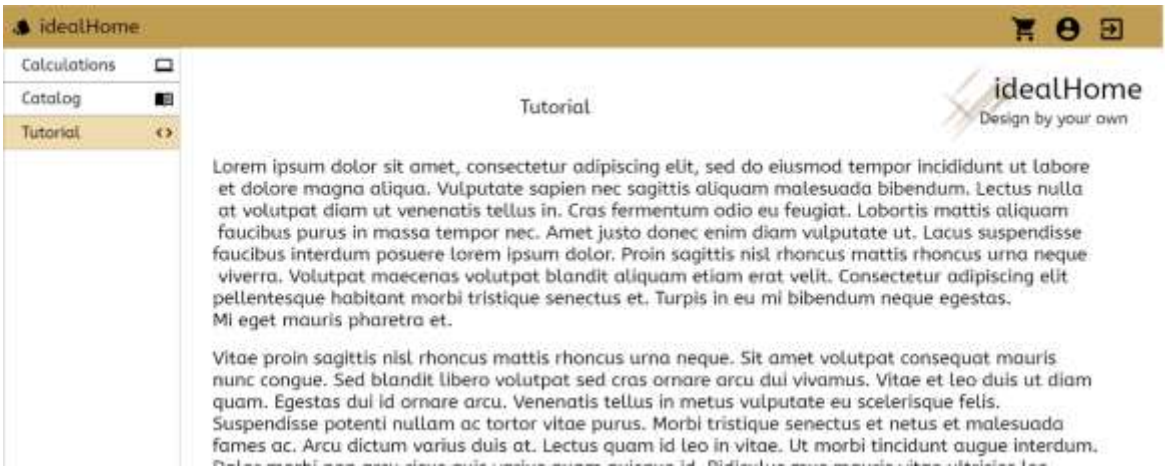

Fig. 3.23. Tutorial Page

The Catalog page was design to list all available items and add them to the cart (fig. 24). Each item has its own page with the item's details (fig. 3.25) or just hover mouse over the item and open the popup to review and add to the cart.

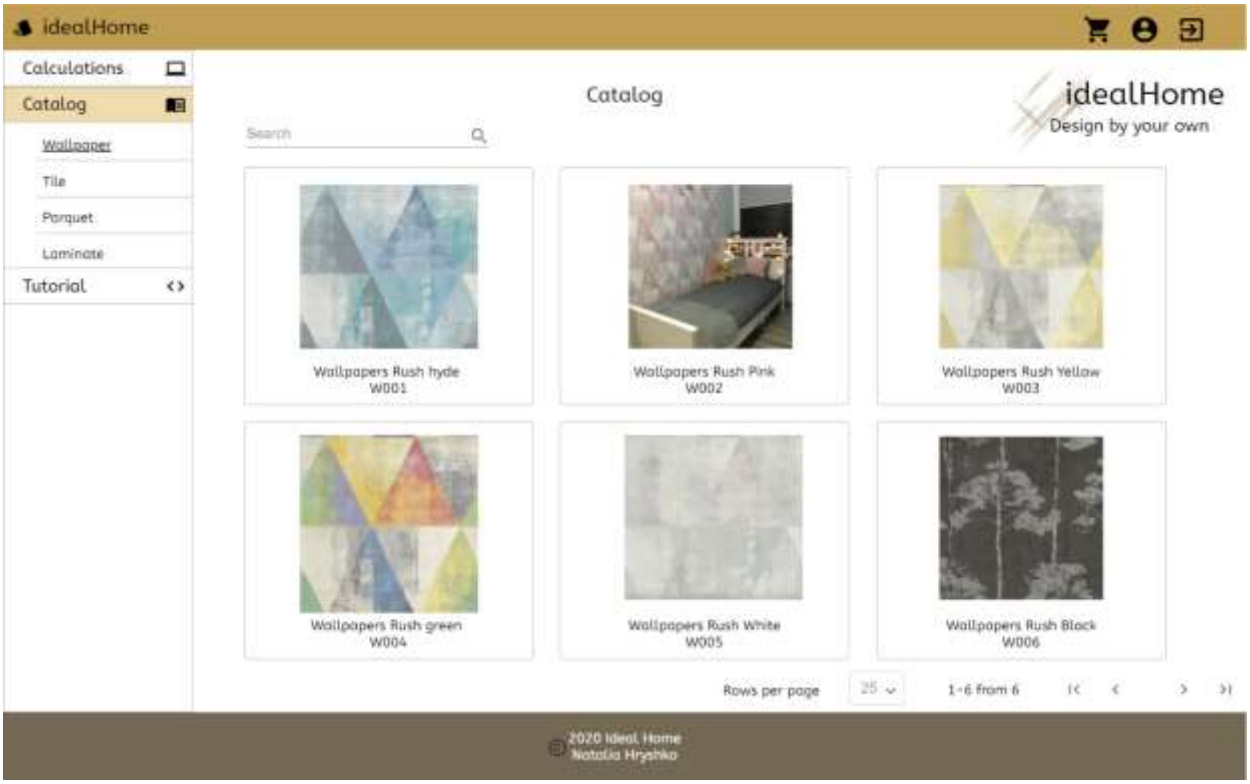

Fig. 3.24. Catalog List

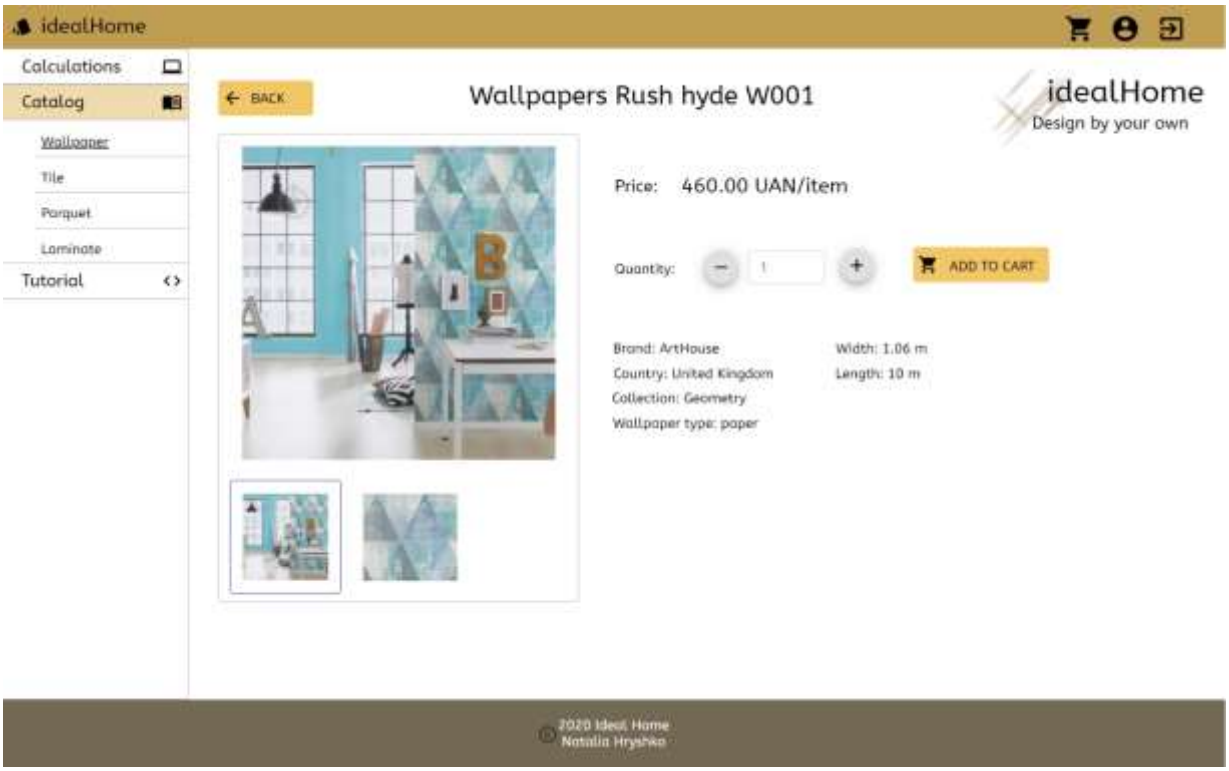

Fig. 3.25. Catalog Item page

Cart page is displayed in figure 3.26 and it displays all items that a user added from the catalog. Here it is possible to navigate to the calculation page and calculate the quantity of items needed specific per the room sizes.

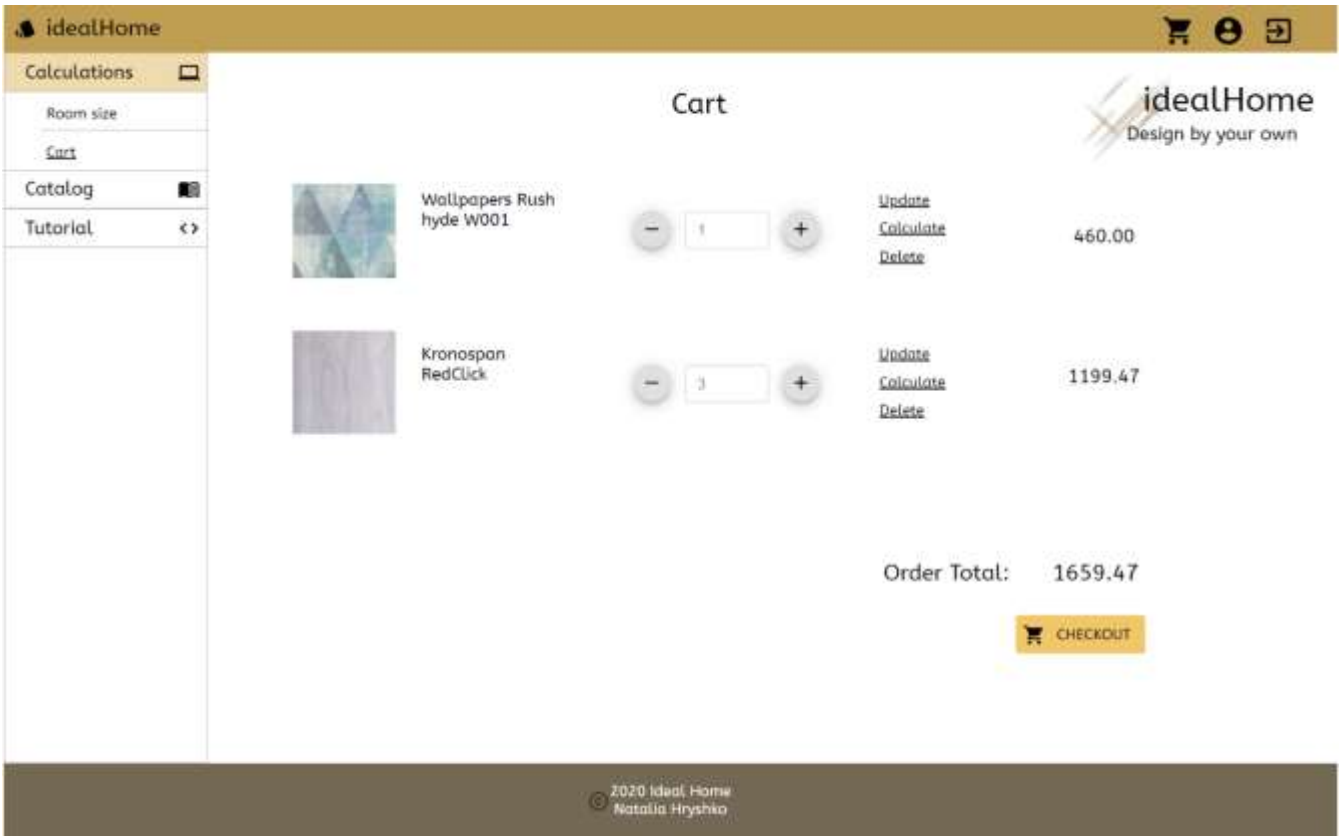

Fig.3.26. Cart Page

In addition, a user can update the quantity and delete the items from the cart. If the User updates the quantity of the item, it is needed to click on the “Update” button to save the changes and recalculate the total price of the order.

If a user clicks on the “Calculate” option, then he or she will be redirected to the Calculations page. It is different for the items of different categories. In the figure 3.27 the calculation page for the wallpaper is displayed. The calculations can be made for the whole room. In the case a user should enable the “Do you want to apply the wallpaper to the whole room?” checkbox and all 4 wallpaper checkboxes will be automatically selected. For now, the web-application calculations are limited to the rectangular and square rooms.

After a user click on the “Calculate” button the calculated quantity will be displayed. Also, there will be “Update” button to update the quantity of the item in the Cart. If a user clicks to update the quantity of items in the cart, then the table with item details and prices will be displayed. The screenshot with results of calculations and updating the cart is displayed in figure 3.28.

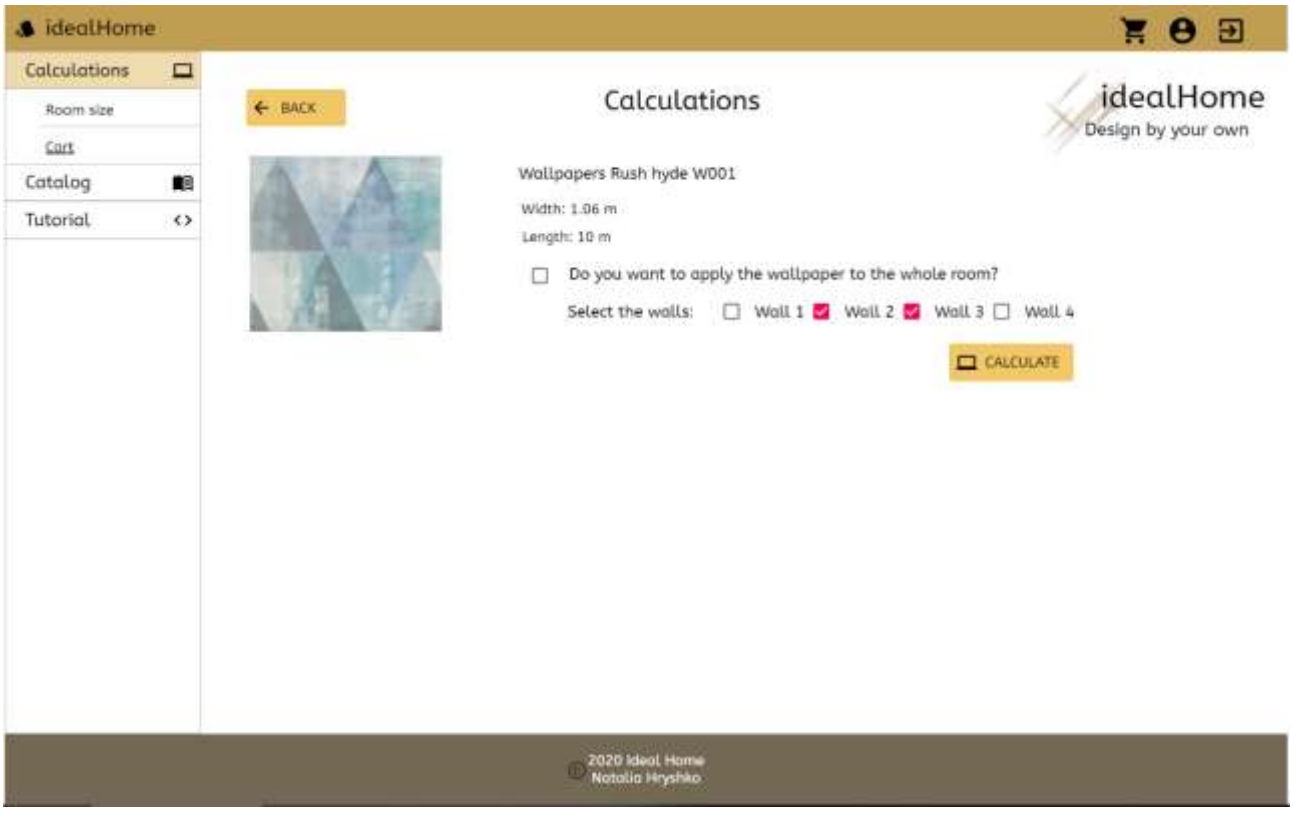

Fig. 3.27. The Calculation page before computing the quantity of the wallpapers

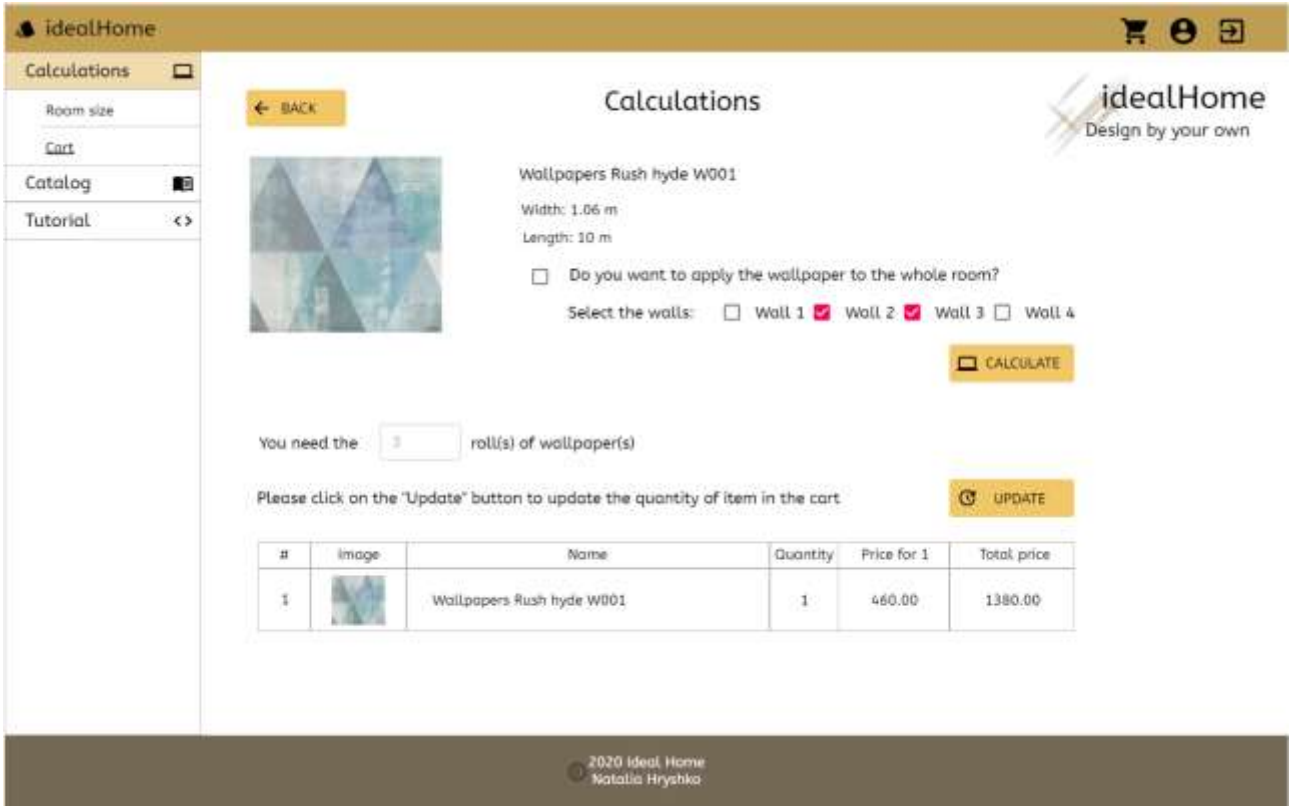

Fig. 3.28. Calculations pages for the wallpapers after updating the cart

Checkout page is designed to complete the order (fig. 3.29). Here a user needs to fill all field with valid data to get the items from the order. The name, surname and email will be prepopulated on the checkout page.

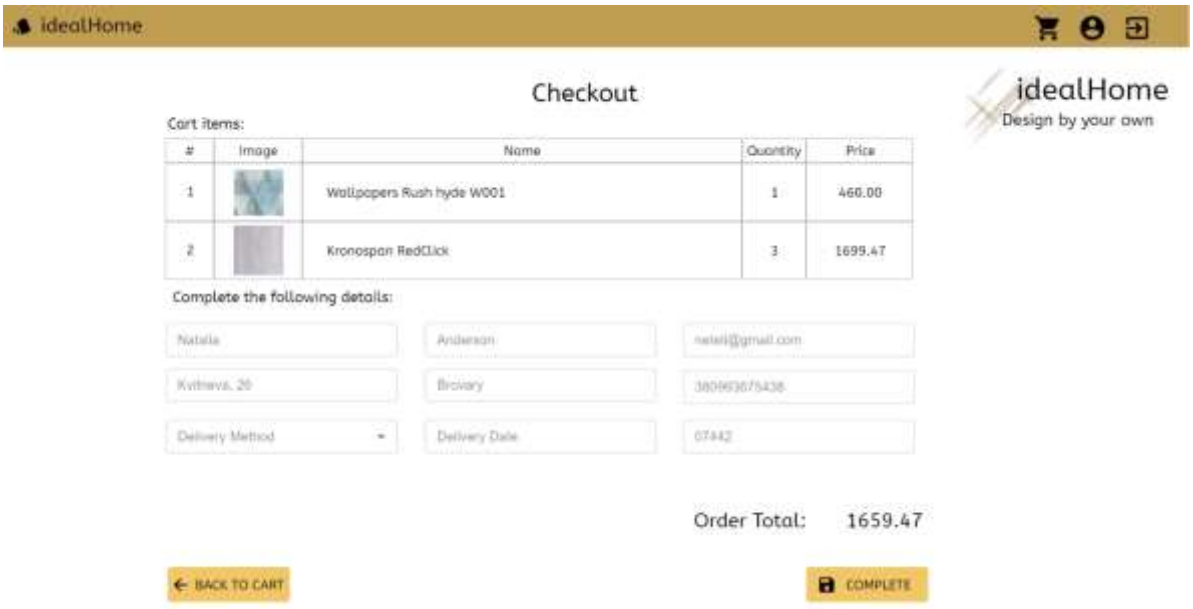

Fig. 3.29. Checkout page

The phone number, shipping address, city and index fields will be also auto filled if a user added them during registration or updated the User Profile page (fig. 3.30).

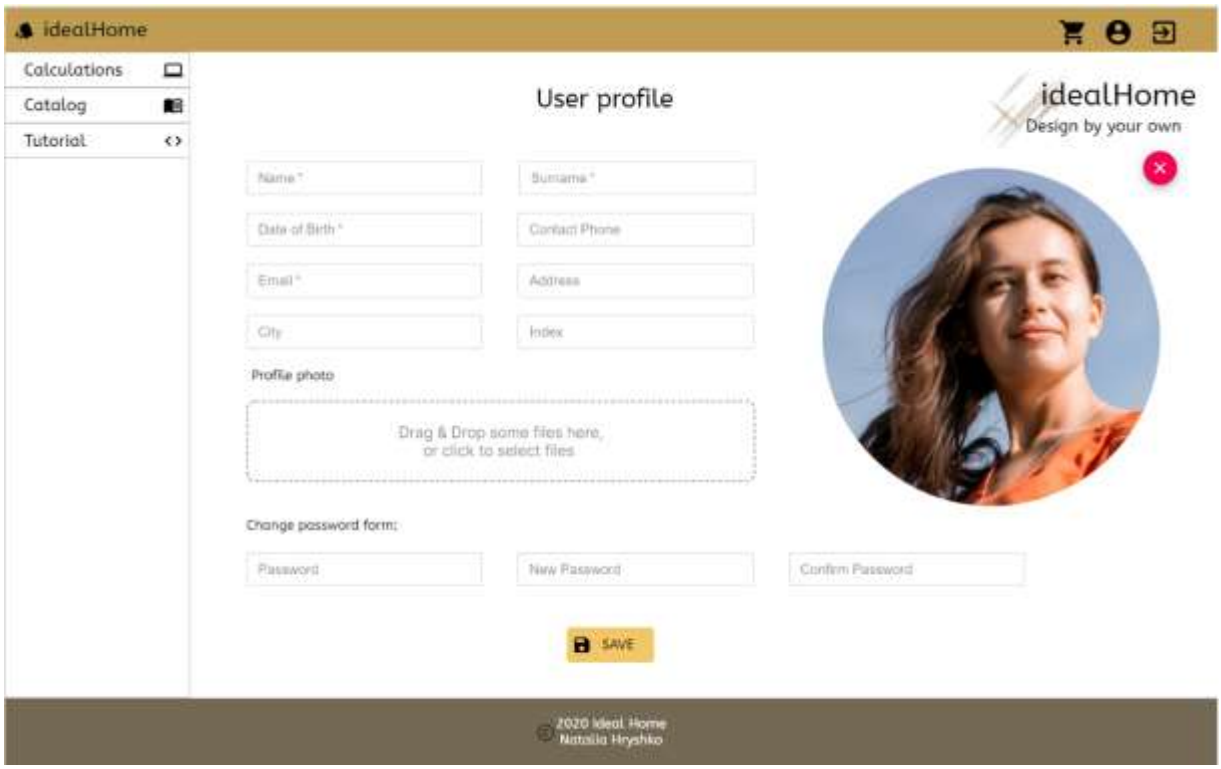

Fig. 3.30. The User Profile update page

After the order is completed the Order confirmation page is displayed (fig. 3.31).

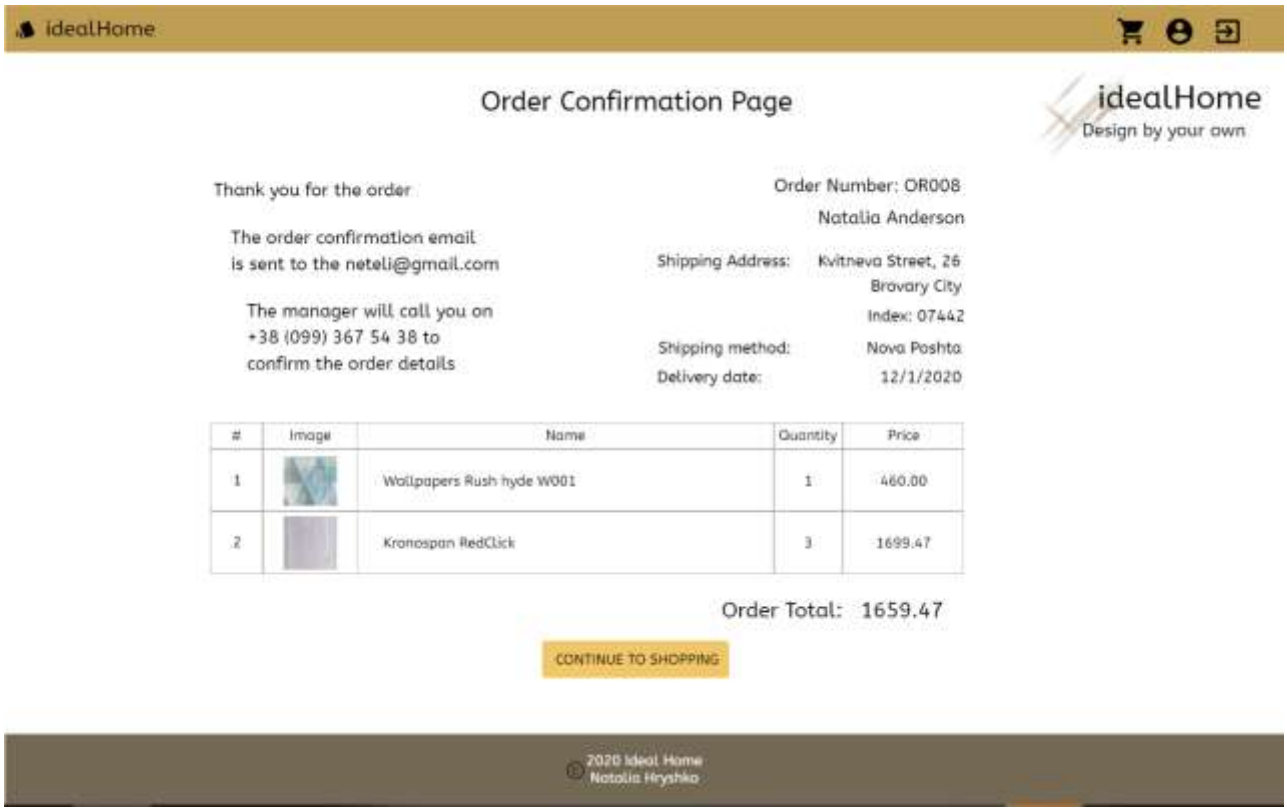

Fig. 3.31. Order Confirmation page

3.2 Database Design

The purpose of any database (DB) is to provide and guarantee the storage of the large amount of data. Record of any data in the database is called data record. Also, a user should be able to query the information from the database, add data records and update or delete the existing ones.

The Firebase Cloud Firestore Database is a cloud-hosted database. Data is stored as JSON and synchronized in real-time to every connected client. Firestore is NoSQL database and does not support complicated SQL queries. To add the Firebase to the Typescript project it is needed to execute some steps.

There is one more dabases release by Firebase Realtime database. However, Firestore uses documents and collections which makes queries easier to use compared to Firebase. While the Realtime Database is just a giant JSON tree, Cloud Firestore is a little more structured

Step 1. Create a Firebase project.

To create a Firebase project, it is needed to navigate to the Firebase console (https://console.firebase.google.com/u/0/) and click on the “Add project” button. To access the page the user should be registered.

Step 2. Register the application in Firebase.

To Register the application, it is needed to click on the “<>” button, enter the name of the application and click on the “Register application” button. After registration, the firebase generates the script which should be added at the end of the body tag before the Firebase services:

```
<script
src="https://www.gstatic.com/firebasejs/8.1.2/firebase-app.js">
 </script>
<script>
  // The web app's Firebase configuration
```

```
  var firebaseConfig = {
    apiKey: "AIzaSyCBZdicxmEtvuPkNhIT9MFI0RiRWaXTQhM",
    authDomain: "neteli-324a2.firebaseapp.com",
    projectId: "neteli-324a2",
    storageBucket: "neteli-324a2.appspot.com",
    messagingSenderId: "250134876335",
    appId: "1:250134876335:web:cff890f58c85343c93045a"
  };
  // Initialize Firebase
  firebase.initializeApp(firebaseConfig);
</script>
```

Step 3. Add Firebase SDK and initialize Firebase.

To install the Firebase SDK it is needed to tun the following command in terminal: npm install --save firebase. Firebase App (the core Firebase SDK) is always required and must be listed before other Firebase SDKs. If some modules are required for any Typescript file they can be required in the code:

```
const firebase = require("firebase/app");
// Add the Firebase products
require("firebase/auth");
require("firebase/firestore");
```

Also, the import module can be used as well. To import any module it is needed to enter the following at the top of the Typescript file:

```
//Import the Firebase SDK
import firebase from "firebase/app";
//Adding firebase modules. The module names can be different.
import "firebase/auth";
```

Step 4. Install CLI and deploy to Firebase Hosting

This step is optional. If the Firebase Web App is linked with a Firebase Hosting site, it is possible to deploy the site's content and configuration when setting up the Web App or anytime later.

To deploy to Firebase it is needed to use the Firebase CLI (command-line tool). After the CLI is installed developer should initialize the Firebase project. To do this it is needed to run the following command from the root of the local app directory: firebase init

By default, every Firebase project has free subdomains on the web.app and firebaseapp.com domains (project-id.web.app and project-id.firebaseapp.com). To deploy to the site the following command should be from the root app's directory:

firebase deploy

The site can be viewed on project-id.web.app or project-id.firebaseapp.com.

When the project and app is set up a developer should create a database. To do this it is needed to navigate to the Cloud Firestore tab and click on the "Create database" button. Then a developer should select server and a starting mode for the Firebase Security Rules. All documents are grouped in collections and global relationships are already established between collections.

To work with Cloud Firestore collections, developer must follow the rules below:

- collections may contain only documents. The document cannot contain fields directly with values, and cannot contain other collections;
- documents may not contain other documents, but they may indicate parts other documents;
- the root database can only be from collections and cannot contain separate documents.

To add the data to the database the form components are created on the UI. The states are controlled by the Redux JS library. The idealHome system has large database. It has three main collections: AdminCatalog, User and Orders. The AdminCatalog collection has four documents – one per each category: wallpapers, tiles, laminates and parquets (fig. 3.32) The full structure of the Admin Catalog collection is displayed in APPENDIX A.

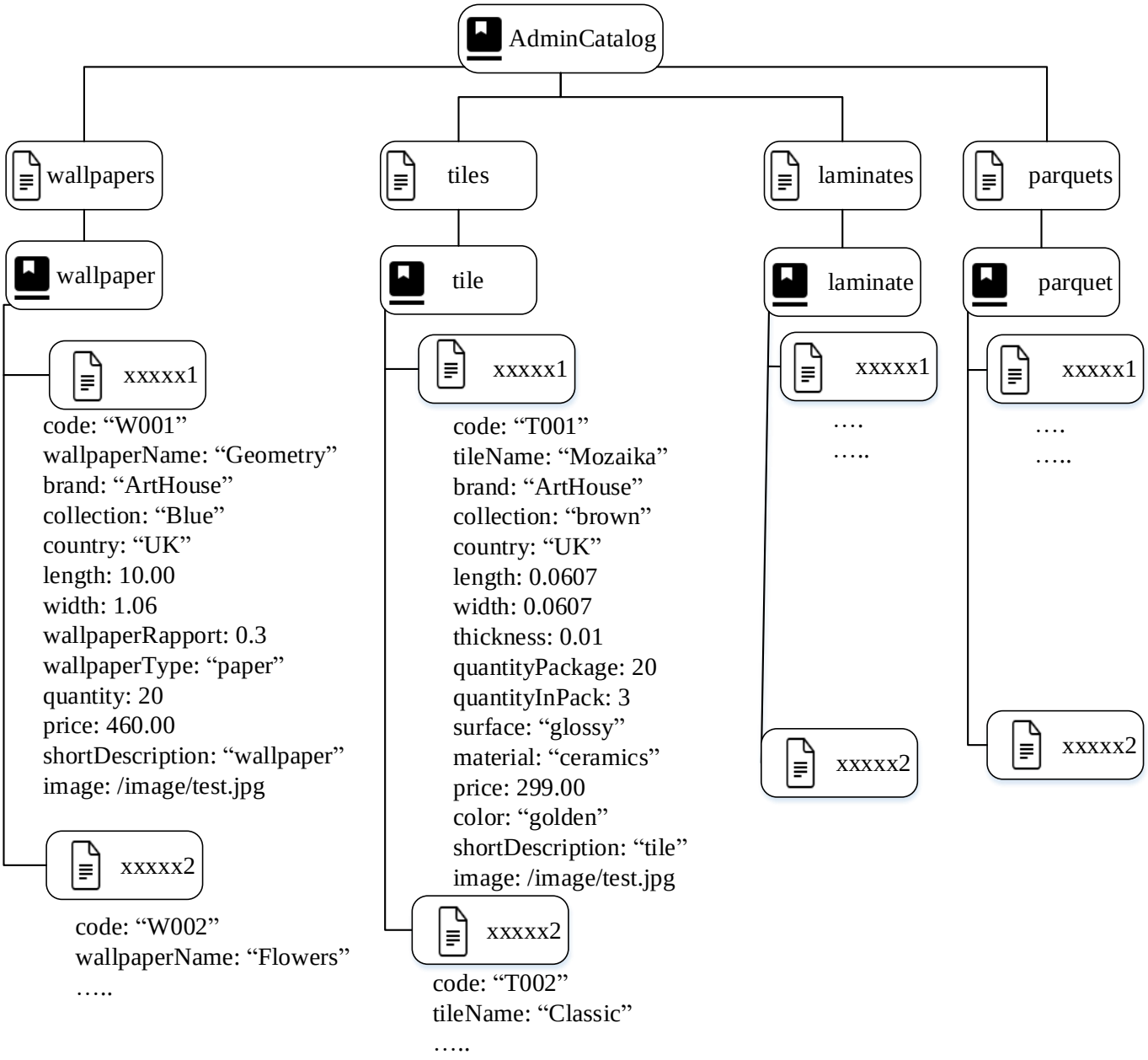

Fig. 3.32. Structure of the AdminCatalog collection

The User collection is designed to store the information about users. It has two documents: internalUsers and customers. The User collection structure and list of fields are displayed in figure 3.33.

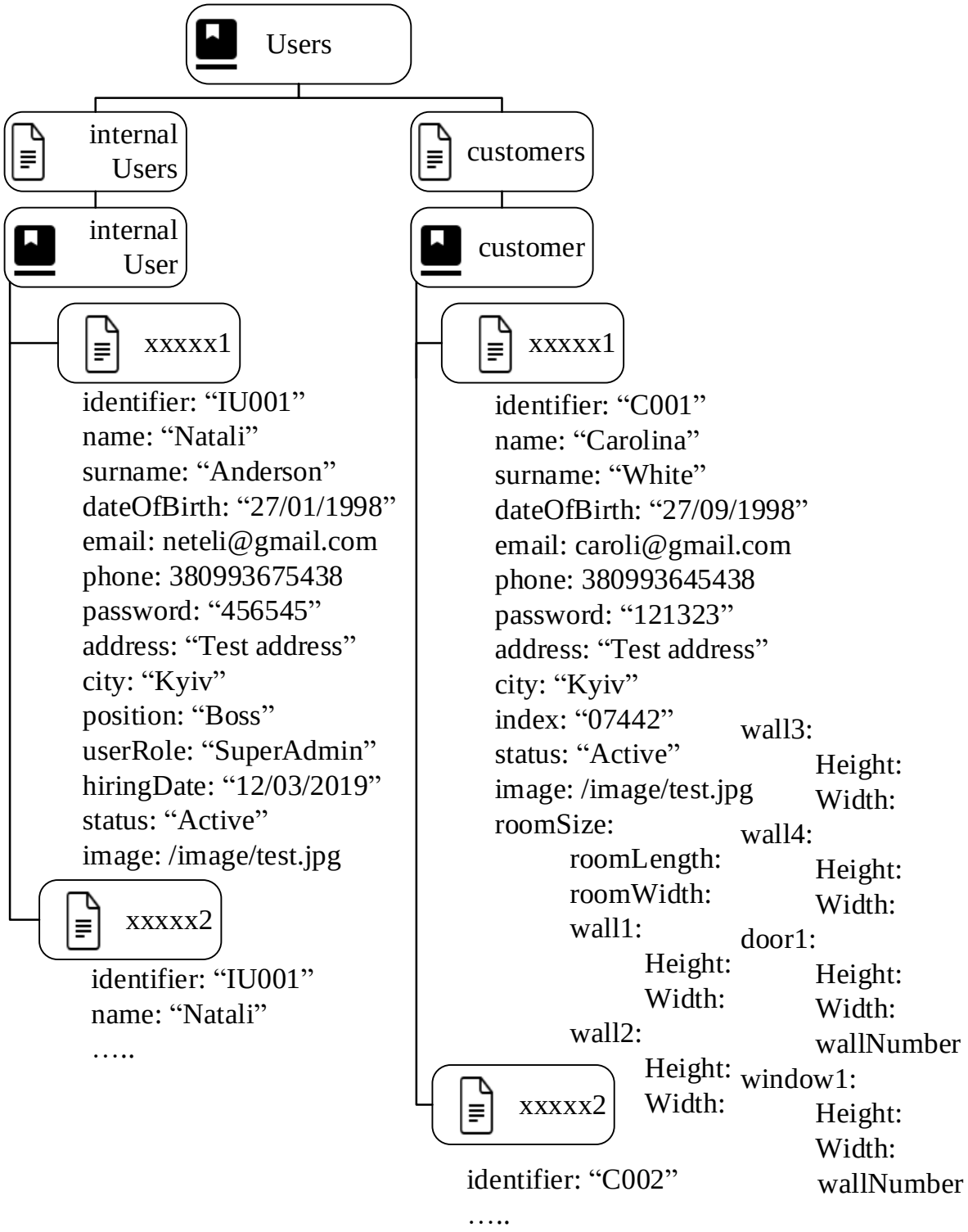

Fig. 3.33. Structure of the Users collection

The internalUsers document store the collection internalUser in which all users of the Admin portal are saved. Each user has the following fields in the own document: identifier, name, surname, dateOfBirth, email, phone, password, address, city, position, userRole, hiringDate, status and image. The customers document stores the customer collection.

The Orders collection structure is displayed in figure 3.34.

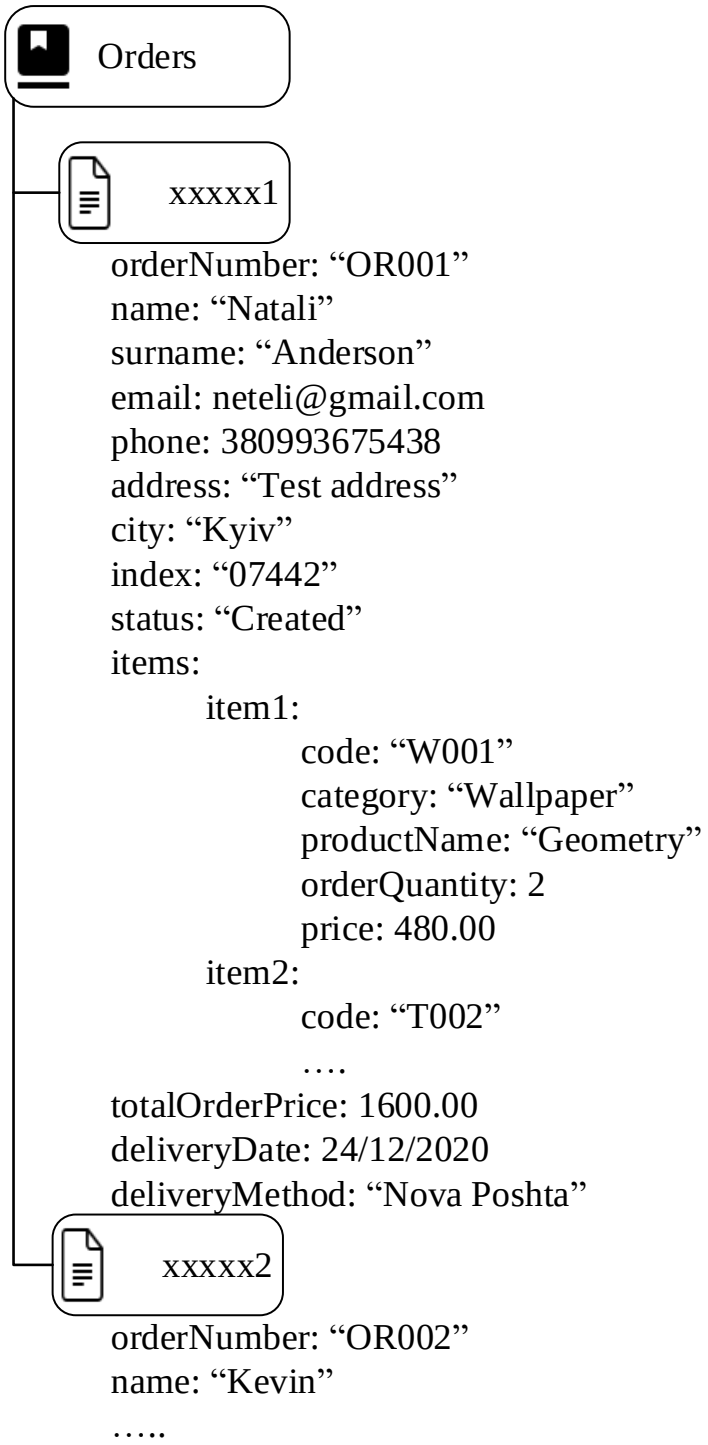

Fig. 3.34. Structure of the Orders collection

The Orders collection has multiple documents for storing the orders. Each document for the order has the following fields: orderNumber, name, surname, email, phone, address, city, index, status, items details, totalOrderPrice, delivery date and delivery method.

3.3 Test Plan

A Test plan is a document that describes the entire scope of testing, from a description of the test object, strategy, schedule, criteria for starting and ending testing, to the equipment required in the process of operation, special knowledge, as well as risk assessment with options for their resolution.

The test plan is usually created for: settling testing processes, prioritizing tasks, resource planning and accounting software and human resources. The quality test plan should include the clear answers on the following questions:

1. What should be tested?
2. How testing will be done?
3. When the testing will be executed?
4. Criteria to start testing
5. Criteria to complete the testing

There are two templates to create the test plan: IEEE and RUP. The IEEE template is more convenient to read that is why it will be used to create the test plan for the designed project. The purpose of this test plan is to cover all aspects of testing and describe the main test scenarios.

There are two Portals that should be tested: Admin Portal and User Portal. The main functions of these two portals are to provide the customers with an up to date Catalog of goods and supply the ability to create an order; the calculation page should function and list the correct calculation results; internal users should be able to see all customers order and successfully managed them. The basic end to end test scenarios will be described in the next subparagraph.

The testing process should be manual and started on the first stage of the product implementation. It helps to avoid a lot of bugs later and reduce the costs of testing efforts. The tasks for the implementing the project are created in Git. All found bugs should be also reported in the git bug tracking system. The ticket should have clear summary, detailed steps, actual and expected results and screenshot or video of the issue added to the description.

The Frontend and Backend tasks should be tested separately. This should help to reduce errors that can be found during integration. The testing should be repeated after integration these two tasks.

The work of the designed web application is supported for the Firefox and Chrome browsers. That is why the testing should be done in both browsers. The browsers version should be the latest ones.

The regression testing should be done after integration of the different modules, especially if the modules are for different portals. The regression test results should be put in the excel sheet and shored.

The task is done when it is implemented as per the specification and function as expected. Also, the functionality is covered with the unit tests and test cases. The lower issues can be accepted for instance the UI styling (e.g. test alignment).

In the future when the main modules of the web application will be done the basic functionality should be covered with automation tests. It will reduce the testing efforts for the regression testing and help to focus on the quality of the new functionality.

3.4 End to End Test Scenarios

End-to-end testing is a Software testing methodology to test an application flow from start to end. The purpose of end- to-end testing is to simulate the real user scenario and validate the system under test and its components for integration and data integrity.

Nobody wants to be known for their mistakes and their negligence, and same is the case with the Testers. When the Testers are assigned an application to test, from that moment, they take the responsibility and the application also acts as a platform to show their practical and technical testing knowledge.

So, to describe it technically, to ensure that testing is done completely, it is necessary to perform End to End testing. [20]. All tests perform software checks, verifying that the actual result obtained corresponds to the specification requirement with different inputs. The test scenariocan consists of a lot of small checks which are called test case. All

test scenarios described below test the system as a whole, emulating a real user environment.

3.4.1 Catalog Test Scenarios

Scenario 1

The new item was added to the Catalog on the Admin portal. When the customer logs in to the User Portal and navigates to the Catalog page then the new wallpaper is displayed and can be added to the cart.

Scenario 2

The quantity of item in the Catalog on the Admin Portal is zero. When the customer navigates to the User Portal and open the item's details then the "Add to cart" button is not available to click and message that the item is out of stock is displayed under the "Add to cart" button.

Scenario 3

The item was updated in the Catalog on the Admin Portal. When the customer logs in to the User Portal, navigates to the Catalog page and open the updated items then all item's details are correctly displayed on the item's details page.

Scenario 4

The item was deleted in the Catalog on the Admin Portal. When the customer logs in to the User Portal, navigates to the Catalog page and searches for the deleted item then there is no item displayed on the page. If the item was in the customer's cart, then the message that the item was deleted from the Catalog and no longer available will be displayed under the item. Also, the item should be displayed and calculated to the total order price on the checkout and order confirmation pages.

3.4.2 Order Creation and Management Test Scenarios

Scenario 1 – Verify the Order creation Workflow

The customer logged in to the User Portal. The customer navigated to the Catalog and added the desired products to the cart. Then he or she clicked on the "Checkout" button and after that the checkout page is displayed. All items' names, images, quantities, and prices are correctly prepopulated on the checkout page. When a customer filled all required fields and click on the "Confirm" button then the Order Confirmation page is

displayed. All item's and orders details are correctly displayed on the page. In addition, the Order Confirmation email should be sent to the customer and internal user. When the internal user of the Admin Portal navigates to the Orders tab and search for the order from the customer then the order is listed in the Orders table and all order's details are correctly prepopulated on the order's details page. The Order is displayed in the Created status.

Scenario 2 – Verify the Order Shipment Workflow

The customer created an order, and the order is displayed on the Admin Orders tab in created status. When the internal user of Admin Portal opened the order's details and changed the order's status to "Completed" then the "Ship" button becomes enabled and the order can be shipped via it. The Shipment Confirmation email should be sent to the customer.

Scenario 3 – Verify the Order Cancellation Workflow

The customer created an order, and the order is displayed on the Admin Orders tab in created status. When the internal user of Admin Portal opened the order's details and clicked on the "Cancel" button then the Order status is changed to "Cancel" and Order Cancellation email is sent to the customer.

Scenario 4 – Verify the Order Details Update Workflow

The customer created an order, and the order is displayed on the Admin Orders tab in created status. When the internal user of Admin Portal opened the order's details and updated any order detail then the Order Details Update email is sent to the customer.

3.4.3 User Management

Scenario 1 – Verify registration on the User Portal

A customer navigated to the Registration page and completed all fields of the Registration form. After he or she clicked on the "Sign up" button the Welcome page should be displayed. When the customer navigates to the User Profile page then all information from the registration form are correctly displayed in the User Profile. Also, the customer should be added to the Customers tab of the Admin Portal. All customer's details should be correctly prepopulated on the customer's details page.

Scenario 2 – Verify possibility to disable the User

The SuperAdmin user is logged in to the portal, opened the Customers or Internal Users tab, selected any Active user from the table. When the user clicked on the Disabled button then login possibility should be disabled for the selected users. If the admin disabled, the customer than he or she cannot log in to the User Portal and the appropriate message is displayed. If the disabled user is internal user than he or she cannot log in to the Admin portal.

Scenario 3 – Verify the update of the user's details

A customer logged in to the User Portal and opened the User Profile page. When the customer updated any details in the profile and saved then the customer's details are updated in the Admin Portal on the Customers tab. The same test scenario should be performed for the internal users as well.

Scenario 4 – The User's Profile is updated with address details after completing the order if the address details were not added during the registration.

A customer registered on the User Portal and filled only required fields during registration. The address, city, phone, and index fiedls are empty in the User's Profile. When the customer created an order then the User's profile is updated with the address details from the order. The address, city, phone and index should not be empty anymore. The customer's details should be also updated in the Admin Portal on the Customers page.

3.5 Future System Improvement

There is nothing ideal in the world. Thus, continuous improvement is essential process for long-term business success. Continuous improvement (CI) is an ongoing effort to improve products, processes, or services by reducing waste or increasing quality. This continuous effort drives a competitive advantage for organizations that get it right but, as with many things in life, consistency is not easy to achieve. To make an interesting product for customers it is every time needed to improve the system and add more and more interesting features. This section describes the ideas on how the system can be improved and what features can be added to it.

3.5.1 Advance Filtering

The first improvement is to design convenient filtering of the catalog items, users and orders on the Admin portal. Also, the same filtering functionality should be applied for the Catalog page of the User Portal. For now, there is only a Search field where the user can search by name or number. It is not convenient when there are so many objects in the database.

The filter will be created and added for each column of the tables of the Admin portal. The user can simply click on the column name and additional popup will be displayed like the filter window in the Microsoft Excel. It will contain filtering options depend on the type of data in the column.

In addition, there will be the “Filter” button and the Filtering popup will be opened after clicking on it. Here all possible parameters for filtering will be displayed. For instance, some application limits the quantity of parameters for filtering, but it is inconvenient for so users so in the IdealHome application there will be realized filtering by all parameters. The same “Filter” button should be design for the Catalog page of the User Portal.

3.5.2 Different Room Shapes and 3D Visualization

3D visualization is a perfect way to demonstrate how the room will look like in reality. It allows to see the room from different perspectives and change if there is something not suite the design. Even designers who’ve spent their lives bringing their work to life with pens, colored pencils, and markers are coming around to the idea that things can simply be done faster, and more effectively in the digital realm.

Currently, the web application is limited to the rectangular and square shapes. In future the functionality that allows user to draw the room by his own or use some default templates (circles, corners, ellipses, open room solutions etc.) should be added. There will a page where the room will be displayed in 3D form and a user can edit the walls, ceil, and floor size by selecting the side and entering the size in the corresponding form. There will be options to navigate thought the 3D room model, add new walls, and update the existing ones. Also, it will be possible to delete the unnecessary walls. The design can be created for one or for several rooms simultaneously. So, it will be convenient to plan the flat and see how it will look like in general. The room planer will have three views: floor view, 3D

line front view and Front 3D view. The floor view allows to create detailed and precise floor plans. It is useful to draw the room shape and set the room size. 3D line front view and Front 3D view are similar but the first one shows the room schematically in 3D while the second one shows the details of each item added from the catalog.

To make it easily to work with 3D model the button for Locking the 3D will be added. If a user clicks on the “Lock” button the possibility to move to spin the room view in any direction will be disabled and a user can continue the work in 2D mode. Sometimes it is really useful to have switch to 2D mode to reviews the design from various angles, arrange, edit and apply custom surfaces and materials. After the button is clicked it is renamed to “Unlock” and a user can click the button again and unlock the 3D functionality at any time.

When a wall is selected then it is possible to apply the wallpaper, tile or just paint it in any color. If it is needed to apply different material for the same wall a user can divide the wall on separate areas and apply the material for different areas independently. The same functionality will be design for the floor and ceil. For the floor there will be available the following items: laminate, parquet, tile, and paint.

The cart will be updated with new categories: kitchen and appliances, bedroom furniture, dinner tables, office planner etc. To add the item to design a user has to drag and drop the items from the catalog and place anywhere in the room. A user can relocate the item at any time by clicking on the item and dragging and dropping it to a new place. When the item is select in the room then there will be an option available to add it to the cart. In the case if some calculation is required like for the wallpaper then there will be a form displayed with an option to calculate the quantity of item per the room planning and option to add it to the cart after calculation will be performed.

Moreover, there will be a feature to create the furniture. A user can navigate to the Create furniture page and design a sketch of the desired object. After the sketch will be completed, he or she can select company from the list that can made the furniture for the user. Each furniture company has own catalog of the materials for the furniture and it will be available for the user to applied for the designed sketch. When the design will be ready

a user can contact an assistant of selected company and discuss price and details or he or she can create an order and wait for the call from company manager to confirm the order.

3.5.3 Designers Support

The web application might be helpful for the designers especially those who are working remotely. The idealHome with 3D rooms can help designers to build the interior and do not waste time for real shopping to found desired items. All items will be saved in the catalog and available to buy because the application uses real catalog of real shops.

There will be two tariff plans for designers. Designers can use it for free if they use the application for creating design only, but they also can buy the ability to search client via the application. In this case designer's profile will be added to the Designers page. Then an ordinary customer can find a designer on the Designers page and contact him or her to create a design for his room online.

The one of the benefits of such communication that there is no need in offline meeting for customer and designer. A user can share the room size and shape, then designer will create planning for the user and share the solution after it is ready. Also, the user can start to create planning by his own and sent the invitation to the designer to work on the project together.

3.5.4 Sharing and Export to Files

Nowadays people like share what they did in social media. It is good advertise for the product if users share information about it with friends. So, one of the future features that should be added as well is sharing the design directly to the social media and possibility to save it as images or into the PDF, DXF or SVG.

Also, the product should have integration with professional tools for planning. There are a lot of designers that use Archicad and Revit software tool. Hence, planning should be easily be exported and imported from these two products. It allows to attract more designers start to work with the idealHome web application. However, the furniture in the room will not be saved and integrated to the separate program due to security measures. It is design by real furniture companies and the sketch should not be openly shared with third-party software tools. The room sketch can be easily shared by customers.

Moreover, the user will be able to capture the snapshot of his or her work. So, it will be possible to create a snapshot of the work and continue the design process. It can be useful when some design experiment has to be implemented. The user can restore the previous design from the snapshot at any time. The snapshot functionality will be similar to the one that the VM Ware Workstation has. It will be limited to not more than three snapshot for one room.

3.5.5 Vacation Tracker for the Internal Users

One more tab should be added to the Users tabs of the Admin Portal. The tab will display all vacation requests for the workers to a manager. The manager can approve or cancel the vacation request of the worker. Also, there is will be additional tab in the Internal user profile of the Admin Portal to create a vacation request. The field of request will be the following: Start Date, End date and Comments. The vacation days will be calculated automatically.

In additional, Admin Portal users can report a sick leave on the Profile page. The fields will be the same as for the vacation requests. Also, there will be a field to add the photo of the document from the doctor's appointment that confirm the sick leave of the worker.

3.5.6 Mobile Application

Current edition doesn't allow to work with idealHome via mobile devices. It is because the application is created to work with interior, so the users have to use large screens to work with detailed visual information. But some of features can be also delivered in mobile application to allow customers to manage their deals anytime everywhere. To track the orders, to communicate with shops and designers, also User portal can be a part of such mobile application (or created as a separate one). So, Android edition and IOS edition can be developed in the future and offered to customers to make user experience more diverse and flexible.

It goes without saying that all the data should be connected to personal account and synchronized between mobile and desktop devices. This should be implemented according to the best practices, that what became a standard for today.

3.5.7 Mail Notification Module in the Admin Portal

Initially in the project all emails templates are hardcoded. So, to update any text in the email it is needed to use the developer's efforts. So, the purpose of this improvement is to give the internal user an ability to create and update the templates by his or her own. This should be developed separately as an improvement. Custom email template can be used for notifying customers about their orders statuses (and also just for summarizing after order is created as it works in most of the shops), for delivering private messages in conversations with designers, for different promo campaigns. Moreover, the template can be customized and use the brand colors of the specific shop if the order consists only of the items of one shop.

The idea is to create a mechanism which will allow to adjust mails (their content, delivery conditions, frequency) without specific technical skills, so every manager can adjust it according to individual needs. It can be possible to update the colors in the emails, font family and font size and test. The project can be used in different countries so in future the functionality to add support of the different languages in the emails will be also developed.

Conclusions on the Third Part

In this part there were described the user interface for the Admin and User Portals, database set up and structure, created test plan for the application testing with covering basic end to end test scenarios. I addition there were described 7 improvements that should be added in future for the designed program module.

An interface is a contract between a system and an external environment. Interface on any system should be understandable, convenient and user friendly. The feature of the application interface is big font size. There are a lot of systems that have so small size, and it is not convenient to work with them. So, in this application this problem is resolved. All interface design was done in the Framer tool.

For the database the Cloud Firestore Realtime database is used. The idealHome system has large database. It has three main collections: AdminCatalog, User and Orders. The AdminCatalog collection has four documents – one per each category: wallpapers,

laminates, parquets, and tile. The User collection is designed to store the information about users. It has two documents: internalUsers and customers. The Orders collection has multiple documents for storing the orders.

The testing process should be manual and started on the first stage of the product implementation. It helps to avoid a lot of bugs later and reduce the costs of testing efforts. The Frontend and Backend tasks should be tested separately. This should help to reduce errors that can be found during integration. The testing should be repeated after integration these two tasks.

The functionality of the program module is covered via the end-to-end test scenarios. The purpose of end-to-end testing is to simulate the real user scenario and validate the system under test and its components for integration and data integrity.

There is nothing ideal in the world. Thus, continuous improvement is essential process for long-term business success. That is why there are 7 improvements that can be added to the web application: advance filtering, different room shapes and 3D Visualization, designers support, sharing and export the planning to files, vacation tracker for the internal users, mobile application, and mail notification module in the Admin Portal.

CONCLUSIONS

The interior design is a very important. Modern people spend a lot of time at home and the good home design increases the productivity and mode. Design-related pros, often hired by homeowners irrespective of renovations, were hired by 1 in 5 renovating homeowners. Among homeowners who renovated their homes, specialty service providers were the most hired renovation professionals, followed by construction professionals So, the users need to have a useful and simple to study application to clear explain their idea to the construction professionals.

There are a lot of programs for interior and exterior design. Some of them are for the professional designers like AutoCad, Revit or ARCHICAD and there also with clear interface for the ordinary users like Ikea Home Planner, SketchUp, Roomsketcher, Planner. However, most of them are not free and use not the real item that can be bought in the shop. For instance, the SketchUp has a lot of modules but they cannot be used in free account. Even for using the real items from the shops the client should pay.

In this diploma work there was designed software module which help people to design and buy in more convenient way. The main idea of the application is to make it easy to calculate the number of materials needed and create the order. A user enters the parameters of the room and adds the item to cart. Then he in she can simply clicks on the "Calculate" option near item on the Cart and the Calculation pages will be open. Here the user can select on which wall he or she want to apply the wallpapers and click on the corresponding option. The results will be displayed in a second and the user can update the cart via the "Update" option.

The difference of this interior system from its analog is that will be free for the ordinary customers and use the real catalogs of the shops. The shops will be interested to pay for allocation of their goods on the portal and pay for usage of the system. Also, the system for interior design is not usually connected with the real shops databases and it is hard to find the desired items in the shop.

All items of the designed interior system will be from real shops and can be ordered via the web application. Also, it is not needed to install any program on the personal computer, and it is very useful. The design can be stored online, and the user can no worry that it will be missed or deleted.

For development of the software module there were user one of the most popular modern technologies. Before selectin the technologies in the second part of the diploma work there were listed and compared a lot of libraries, frameworks and databases solution. Also, there were analyzed the type of input and output data that will be stored within the application.

The product is developed in the VS Code environment. The programming language is TypeScript. TypeScript is an open-source programming language developed and maintained by Microsoft. It is a strict syntactical superset of JavaScript and adds optional static typing to the language. Also, to simplify the development process React, Redux and React-Router libraries will be used.

React is a declarative, efficient, and flexible TypeScript library for building user interfaces. It lets a user to compose the complex UIs from small and isolated pieces of code called “components”. Redux is a small library that work with data. It helps developers to understand where and how the data should be stored. React Router is one of the most popular routing frameworks for React. The library is designed with intuitive components to let a user build a declarative routing system for the application. With routers, the user experience of the app can simplify site navigation.

The application has two portals: User and Admin. To access any of this portals the user should be authorized. The User portal has a registration form where user can create an account in few minutes by his or her own. For adding the user to the Admin portal, the Superadmin user should log in to the portal and create an account for the internal user. After the account is created the email with details will be sent to the user’s email.

The User portal is used by customers to review the catalog, calculate the quantity of items added to the card per room’s size and to complete the order. The Admin Portal can be used for management of users from both portals, created orders and catalog. Here it is

possible to add new user or disable an existing one, create, update, cancel and ship the order and add, update and delete the items in the catalog.

The application has a lot of objects that need to be searched so for the backed side there were selected Algolia. Angolia provides convenient RESTful API for the web sites and applications for instance search. The software module has a large database that is why the Algolia API is needed to quicky search the items and filter them. The Algolia Search API returns full results in less than 10ms on average. For the database the Firebase database will be used.

To speed up the UI styling Material UI framework will be used. A developer can use default elements or customize them to align the style with the company identity (design system) and products. To design the mockups for the application Framer was selected as prototyping tool. Framer is an IDE where it is possible to build a prototype for the project for free.

For the database, the Cloud Firestore Realtime database is used. The idealHome database has three main collections: AdminCatalog, User and Orders. The AdminCatalog collection has four documents – one per each category: wallpapers, laminates, parquets, and tile. The User collection is designed to store the information about users. It has two documents: internalUsers and customers. The Orders collection has multiple documents for storing the orders. Each document is one order in the system. The structure of the main database collection is detailed described in the third part of the diploma work.

For now, the software module is not covered via the automation tests. All testing was done manually. The testing process was started on the first stage of the product implementation. It helped to avoid a lot of bugs later and reduce the costs of testing efforts. The Frontend and Backend tasks was tested separately and then the testing was repeated after integration these two tasks.

In addition, the basic functionality of the product was covered via the end-to-end test scenarios that should be tested after adding new features to the web application. End-to-end testing is a Software testing methodology to test an application flow from start to end. The purpose of end-to-end testing is to simulate the real user scenario and validate the system under test and its components for integration and data integrity. The test scenarios

are divided into three categories: Catalog test scenario, Orders creation and management test scenarios and user management test scenarios.

There is nothing ideal in the world. Thus, continuous improvement is essential process for long-term business success. That is why there are 7 improvements that can be added to the web application. There are seven improvements for the near future:

- advance filtering,
- different room shapes and 3D visualization;
- designers support;
- sharing and export the planning to files;
- vacation tracker for the internal users;
- mobile application;
- mail notification module in the Admin Portal.

.

REFERENCE LIST

1. Overview of U.S. Renovation in 2019 & 2020 [Electronic resource]: – URL: https://st.hzcdn.com/static/econ/2020-US-HouzzandHome--Study.pdf (retrieved date: 05.11.2020)

2. What Is AutoCAD? [Electronic resource]: – URL: https://study.com/what_is_auto_cad.html (retrieved date: 06.11.2020)

3. What is Archicad? [Electronic resource] URL: https://helpcenter.graphisoft.com/knowledge-base/86314/ (retrieved date: 06.11.2020)

4. Interior Designer Software for 3D Visualization [Electronic Resource]: URL: https://www.easyrender.com/a/interior-designer-software-for-3d-visualization (retrie-ved date: 08.11.2020)

5. IKEA Planning tool [Electronic resource]: URL: https://www.ikea.com /us/en/planners/ (retrieved date: 08.11.2020)

6. The TypeScript Handbook [Electronic resource]: URL: https://www.typescriptlang.org/docs/hand-book/intro.html (retrieved date: 09.11.2020)

7. Bright, Peter (3 October 2012). "Microsoft TypeScript: the JavaScript we need, or a solution looking for a problem?". Ars Technica. Condé Nast. Retrieved 26 April 2015.

8. TypeScript in 30 minutes [Electronic resource]: URL: https://code-guida.com/post/475 (retrieved date: 10.11.2020)

9. Tutorial: Intro to React [Electronic resource]: URL: https://reactjs.org/tutorial/tutorial.html (retrieved date: 10.11.2020)

10. 9 things every React.js beginner should know [Electronic resource]: URL: https://camjackson.net/post/9-things-every-reactjs-beginner-should-know Cam Jack-son 2016 (retrieved date: 10.11.2020)

11. How To Handle Routing in React Apps with React Router [Electronic resource]: URL: https://www.digitalocean.com/community/tutorials/how-to-handle-routing-in-react-apps-with-react-router (retrieved date: 15.11.2020)

12. Kelly Burke, Timur Newport, WA [Electronic resource]: SaaS Boilerplate URL: https://builderbook.org/book (retrieved date: 20.11.2020)

13. Material UI [Electronic resource]: URL: https://material-ui.com/ (retrieved date: 21.11.2020)

14. Introduction to Material UI [Electronic resource]: URL: https://material.io/ design/intro-duction#theming (retrieved date: 25.11.2020)

15. Eva Design Assets [Electronic resource]: URL: https://eva.design/?utm_ source=medium&utm_medium=article&utm_campaign=top10_DS (retrieved date: 26.11.2020)

16. Top 10 best Design Systems for 2020 [Electronic resource]: URL: https://medium.com/akveo-engineering/top-10-best-design-systems-for-2020-3994d4 66f56f (retrieved date: 28.11.2020)

17. Prototyping in Framer: Pros and Cons [Electronic resource]: URL: https://blog.framer.com/ framer-pros-and-cons-346778e091f8 (retrieved date: 28.11.2020)

18. Pros and Cons of Firebase [Electronic resource]: URL: https://redvike.com/pros-and-cons-of-firebase/ (retrieved date: 30.11.2020)

19. Best Web Development IDE [Electronic resource]: URL: https://hackr.io/blog/web-develop-ment-ide (retrieved date: 30.11.2020)

20. What Is End To End Testing [Electronic resource]: E2E Testing Framework With Examples. URL: https://www.softwaretestinghelp.com/what-is-end-to-end-testing/ (retrieved date: 05.12.2020)

APPENDIX A

Structure of the AdminCatalog Collection in the Database