



Developing a Visual Tool for Rule Definition in Role Based Access Control System for Social Networks

By:

Doaa Jawad Idais

Supervised By:

Dr. Khalil Massri

Prof. Mario Macedo

A Thesis

Submitted in Fulfillment of the Requirements for the Master of
Information Technology and Systems Management.

2021

Developing a Visual Tool for Rule Definition in Role Based Access Control
System for Social Networks

By

Doaa Idais

This thesis was defended successfully on November 2021 and approved by:

Committee Members

Signature

1- Dr.Khalil Massri

(Hebron university)

Supervisor

2- Dr.Mario Macedo

(Atlantica university)

Co-Supervisor

3- Dr.Ibrahim Ahmaro

(Hebron university)

Internal Examiner

4- Dr.Amjad Hawash

(Al-Najah National university)

External Examiner

Table of Contents:

- Chapter 1: Introduction 11
 - 1.1. Research Background..... 11
 - 1.2.Problem Statement..... 13
 - 1.3.Research Objectives..... 14
 - 1.4.Research Questions..... 15
 - 1.5.Importance of the research..... 15
 - 1.6.Short description of the research..... 16
 - 1.7.Research Outline..... 17
- Chapter 2: Literature Review 19
 - 2.1.Types of Prototyping:..... 19
 - 2.2.Types of projects Prototyping can be suitable for:..... 24
 - 2.3.The Importance of Prototyping in the product development process:..... 25
 - 2.4.How to make prototyping more effective and less costly?..... 26
 - 2.5.Access control prototype for social networks:..... 27
 - 2.6.Human centered design prototype:..... 28
 - 2.7.Access control in social networks:..... 28
- Chapter 3: User Interface (UI) Design 32
 - 3.1.Development Method (Agile Methodology Scrum)..... 32
 - 3.2.User Centered Design Methodology (UCD)..... 34
 - 3.2.1. Identification..... 37
 - 3.2.2. Users Identification Survey 37
 - 3.2.3. Online Social Network UI evaluation survey 40
 - 3.3.Requirements..... 42
 - 3.3.1. Interviewing with the UX (User Interface Expert)..... 42
 - 3.3.2. Developing Personas 44
 - 3.4.Design..... 46
 - 3.4.1. First Design (UI-1) 47
 - 3.4.2. Second Design of User Interface (UI-2) 57
- Chapter 4: Implementation 68
 - 4.1.Role-Based Access Control UI System Description..... 69
 - 4.2.General System Design..... 71
 - 4.3.Main Sub-systems..... 72

4.4.Pilot UI:.....	78
4.5.Events Handling.....	82
Chapter 5: System Usability Experiment	84
5.1.Objectives.....	84
5.1.1. Designing a test framework.....	84
5.2.Methodology.....	85
5.3.Participants.....	86
5.4.Conducting Experiments.....	88
5.5.Results.....	89
5.6.Interviews.....	92
5.7.Discussion.....	94
Chapter 6: Conclusions & Future works	97
6.1.Conclusions.....	97
6.2.Future Works.....	97
References:.....	99
Appendix A: Survey questions and results	103
Appendix B: Tasks Testing Questionnaire:	107

List of Figures:

Figure 1.1: Overall system components.	17
Figure 3.1: LAVA Scrum Board	34
Figure 3.2: UCD phases.....	35
Figure 3.3: UI design steps	36
Figure 3.4: UI design sketch.	46
Figure 3.5: Main Interface (UI-1).....	47
Figure 3.6: Add Role Interface (UI-1)	48
Figure 3.7: Add New Role (UI-1).....	49
Figure 3.8: Edit Super Admin role (UI-1)	49
Figure 3.9: Add resource interface (UI-1)	50
Figure 3.10: Add new resource (UI-1).	51
Figure 3.11: Add new field (UI-1).....	51
Figure 3.12: Add permission (UI-1).....	52
Figure 3.13: new add resource interface (UI-1).	52
Figure 3.14: Add user interface (UI-1).....	53
Figure 3.15: new user info (UI-1).....	53
Figure 3.16: Main Interface (UI-2)	57
Figure 3.17: edit role name (UI-2).....	58
Figure 3.18: add resource (UI-2)	58
Figure 3.19: add new field (UI-2).....	59
Figure 3.20: Set resource permission (UI-2).....	59
Figure 3.21: select field's actions (UI-2)	60
Figure 3.22: add role (UI-2).....	60
Figure 3.23: Set role permission icon (UI-2)	61
Figure 3.24: Set role permission.....	61
Figure 3.25: set action for role on resource (UI-2)	62
Figure 3.26: Add user (UI-2)	62
Figure 3.27: add permission hint1	66
Figure 3.28: add permission hint2	67
Figure 4.1: UI system descrbtion.....	70
Figure 4.2: General System Design.	71
Figure 4.3: Main Sub-System	72

Figure 4.4: JSON file Structure.	74
Figure 4.5: OKTOB UI Components	75
Figure 4.6: Main Pilot Interface	79
Figure 4.7: Add Resource Pilot Interface.	80
Figure 4.8: Resource Added Pitot Interface.....	80
Figure 4.9: Set Permission Pilote Interface.....	81
Figure 4.10: OKTOB Pilot Interface.....	81
Figure 4.11: React event-handling structure.	82

List of Tables:

Table 3.1: Deference between Agile Methodology and Traditional Approaches.	32
Table 3.2: Participants Tasks for UI Evaluation.....	41
Table 3.3: UX interview summary.....	43
Table 3.4: Persona One.....	45
Table 3.5: Tasks completion exp.1.	55
Table 3.6: Participants Information in exp.2.....	63
Table 3.7: task completion exp.2.....	64
Table 4.1: Comparision between React, Angular and Vue.js.	68
Table5.1: Participants Information for System Usability.	87
Table 5.2: Tasks for Usability Test.....	88
Table 5.3: Effectiveness Result for Group1.....	89
Table 5.4: Effectiveness Result for Group2.....	90
Table 5.5 Efficiency Rating.	90
Table 5.6: Efficiency Result for Group1.	91
Table 5.7: Efficiency Result for Group2.	91

List of Graphs:

Graph 3.1: Percentage of participant activities.....	38
Graph 3.2 : Percentage of user knowledge about the people who can see his information on social networks.	38
Graph 3.3 : Percentage of users who set a privacy settings before share their information.	38
Graph 3.4 : Percentage of users that don't feel safe when using social media network.	39
Graph 3.5 : Percentage of user believes that their privacy is protected by social networks.	39
Graph 3.6 : Percentage of users that Agreed to take a part of designing UI.	39
Graph 3.7 : Percentage pf users that need more flexible UI to access control their data in social networks.	39
Graph 3.8 : Percentage of users that have the ability to use new UI for access control.	39

About the research:

This research is a supplement to a published scientific article by the teamwork titled “User as a Super Admin: Giving the End-Users Full Control to Manage Access to Their Data in Social Media Networks”. This was presented at the “International Symposium on Electrical, Electronics and Information Engineering (ISEEIE 2021)” (Awad, Al-Karaki, et al., 2021) in South Korea. And the article was classified as a featured article at kudos site which is affiliated with ACM. (Awad, Al-karaki, et al., 2021)

Acknowledgements:

First of all, I would like to thank my thesis supervisors for the invaluable guidance, support, and motivation throughout the process. I am very grateful to them. I would also like to honor the experts and participants who participated in the research and contributed with compelling feedback. Without their passionate participation and input, the user research could not have been successfully conducted. Finally, I must express my deepest gratitude to my family for their unwavering support and motivation. This achievement would not have been possible without them. Thank you.

Abstract:

This thesis presents the design and evaluation of a user interface for role-based access control for any resources that can be used in social networking applications, called Lava. Lava has been designed to give end-users the ability to set permissions on user's resources based on pre-defined roles using easy and simple interfaces. Then to generate JSON files that contain the definitions of resources and role permissions schema that user sets on his resources. For that goal, a user-centered design approach was utilized to facilitate an optimal user experience and to emphasize the end-user.

The main goal of the Lava system is to make users aware of role-based access control, and let them set permissions to their own resources through interactive user interfaces in online social networks. These interfaces also let the user be a participant while designing the prototype.

The result of the research is a high-fidelity user interface prototype which has all the functionalities that enable users to set the permissions for their resources. Then based on the user interaction with the interface, a JSON file will be generated. This JSON file will be used by another system (being developed by another team partner) called ETQAN which will generate an application programming interface (API) to be used in social networks.

Chapter 1: Introduction

1.1. Research Background

User interface design is a subset of Human Computer Interaction (HCI) that focuses on understanding how humans and computers can work together efficiently. Research in creating a new user interface should consider what people need and expect, how their perception of the proposed system field, and what skillset the targeted users have (Ben Shneiderman et al., 2016). The user interface is the part of the system that people can see, touch, hear, and give commands to, then expect to see responses and results (Galitz, 2007).

Many diverse researches and studies concerning different aspects of HCI have been conducted. Ranging from dialogue techniques, gestural analysis, multimodal interfaces, computer graphics, computational linguistics, spatial cognition, robot navigation and wayfinding, input styles or devices, and monitor screens, and more. However, the ultimate goal of the studies is to contribute in improving the interaction between humans and computer systems by endowing technical systems with higher usability and satisfaction (Liu et al., 2010).

A number of different design methodologies addressing methodologies for HCI design have emerged in recent years. The rationale behind most of these design techniques is based on how user, designer and technical systems interact. Good and efficient communication between users and designers is considered the most important factor in modern design approaches (Patrício et al., 2008).

User-centered design (UCD) is a well-known design methodology that puts users at center-stage in the design processes; it also gives the end users' requirements and limitations great attention at each stage of the design process. User requirements are considered and treated as a priority in the whole design life cycle. The standard characteristics of the UCD design make

users' active participants in each iteration of the design process. Therefore, designers should not only analyze how users interact with an interface, but also test the validity of their assumptions concerning users' behavior in real-world scenarios with real users. Comparing UCD to other interface design approaches, UCD optimizes how users can, want, or need to work, rather than forcing the users to change their behavior or understanding to accommodate the designers' views. To actually achieve this design hypothesis, users, designers work together to articulate the wants, needs and limitations of the user and create a system that meets these requirements. (Galitz, 2007)

Online Social Networks (OSN) is one of the most in demand online services the Internet provides. These OSN attract users because they allow communication opportunities with new people around the world that they are acquaintances with or even unknown to them, where they can share information with their network of friends (Boyd & Ellison, 2007)

Users nowadays use online social networks (OSN) frequently. Studies show that more and more users are paying significant amount of their time to use these services (Jin et al., 2013). As these OSN grows, the threat to user's privacy and security grows. Currently OSN provide users with tools to protect their privacy and security. However users are not aware of all of the tools, and when they move from one OSN to another, they will find a complete new system and they need to adapt their mental models to the new system. Some Internet users, due to their lack of knowledge, might be unintentionally compromising their privacy. Quite often, they are not aware that their profiles are readily available to basically every other user. They often post details about their private lives, mistakenly believing that this information will remain well-protected within their friends' circle (Shillair et al., 2015) .

Role-Based Access Control (RBAC) is an approach that enables assigning permissions to different type of roles to access resources. The model has gained popularity in recent years. In RBAC the role is the basic element and the whole access control policy is formulated to

address this element. A role undertakes different presentations and can be mapped to different scenarios (Fuchs et al., 2011).

RBAC limits access to the information by allowing users to access only the part they need to perform their tasks and prevent them to have access to information that they don't need or belong to them. RBAC works on the principle of assignment of roles to the users instead of giving access to individuals. In a network, the roles refer to the level of access that users are given. Access is dependent upon many and numerous factors, such as job competency, responsibility, authority and securing confidential information. Roles consist of attributes. Attributes of the role are the properties of the job they are assigned to and the type of work they are allowed to do and the kind of information they are allowed to access. It also consists of the type of data and what data that they are not allowed to access. Role is defined in such a way that, the semantic concept around which the policy and rules and regulations of access control is made. Role brings together the user and permissions (Fuchs et al., 2011).

RBAC has been designed for admin users and people with IT knowledge. Many User Interface designs have been introduced to allow users to set permissions on resources which require technical knowledge. These interfaces are difficult and not easy to be used by normal users. If a new UI to be design to manage RBAC that target normal users, it will impose many challenges. Such user interface will need to study user awareness of RBAC and to create a model of how normal users may approach assigning roles and permissions (Alturi & Ferraiolo, 2011).

1.2. Problem Statement

Role-based access control is a complex system. Understanding it is not straightforward for regular users, in which any mistakes made could cost the users a compromise to their private data. The main problem of this research is that the OSN providers control the data and provide users with facilities to manage access to data, and the options that are available for

users to manage access control to their own data are only managed by the OSN providers (Imran-Daud et al., 2016).

If users are the owners of their own data and the only administration authority that can control access to the data, they need to have a proper understanding of managing access control and assigning roles to resources. This can be facilitated by a user-friendly interface that will guide users and help them how to assign roles properly and accurately. There are many user interfaces which have been designed for admin users who are technical persons. However, designing such a user interface for normal users is still a challenging problem (Darejeh & Singh, 2013).

This research aims to designing a user interface that enables normal users to manage access control to their data online and assign different roles on different resources.

Designing such a system involves a user center design approach, and involves background study to understand the user requirements and understanding of access control. It also designs low fidelity prototype to be directly evaluated with users through experimentation, direct observations, and interviews.

1.3. Research Objectives

1. To explore the user understanding of access control and how they can relate it to their own data.
2. To identify the most important actions users are performing when they would like to share a resource with a targeted group of people.
3. To design a good user experience for a UI that visualizes access control to user's data.
4. To develop a system works as a tool that users can learn about role-based access control and make aware of concerns of privacy to their own sensitive data.

1.4. Research Questions

- What are users understanding of access control and how they can relate it to their own data?
- What are the most important actions users are performing when they would like to share a resource with a targeted group of people?
- How to design a good user experience for a UI that visualize access control to user's data?
- How can such a system be a tool that users can learn about role-based access control and make aware of concerns of privacy to their own sensitive data?

1.5. Importance of the research

The goal of any user interface design is simple, to make working with computers easier, productive and enjoyable. The user interface is an increasingly important aspect of a product that is often at least as important as the functionality in determining its success. The user interface will guide the users to make correct decisions and get proper results. Bad user interface will allow users to make mistakes and increases the number of mistakes they probably make. On the other hand, with a good product design, the users' motivations to use the products will increase. According to a studies, the amount of code written for User Interface is more than 50% (Liu et al., 2010). The suggested UI system will enable users to control access to their data, assign roles and be aware of all concerns and issues related to the privacy of their data. The research will also contribute to the HCI, where new findings and recommendations will be made. The research will also use a UCD methodology that involve the users in the UI design to have a final product that have all the functionalities that users want and to let the UI more easy and understandable by end-users.

1.6. Short description of the research

The research idea is to develop two systems:

- **Lava system:** is a web-based system that contains user interface (UI) to give the end-users the permission to access control to their data on social media networks. The output of Lava is going to be a JSON file containing all the semantics of access control roles and permissions chosen by the user.
- **Etqan system:** is a web-based system to generate a RESTFUL application programming interface (API) from JSON files provided from LAVA system for any web sites or any user. Etqan system is going to be implemented and discussed by another thesis proposal from another member of the overall project team.

LAVA system consists of three main sub-systems:

1. User interface.
2. State management.
3. JSON file Generation.

The user interface allows the users to manage their access control to their data in social network using easy interfaces. The second sub-system is a state management which is responsible to maintain and track the overall system state, then the last sub-system is generating JSON file which will contain the users data and all permission they have and store it in the database. In the final stage, this generated file will be sent to Etqan system using POST request to generate API request for social networks.

Figure 1.1 below describes the overall system which contains LAVA and Etqan systems.

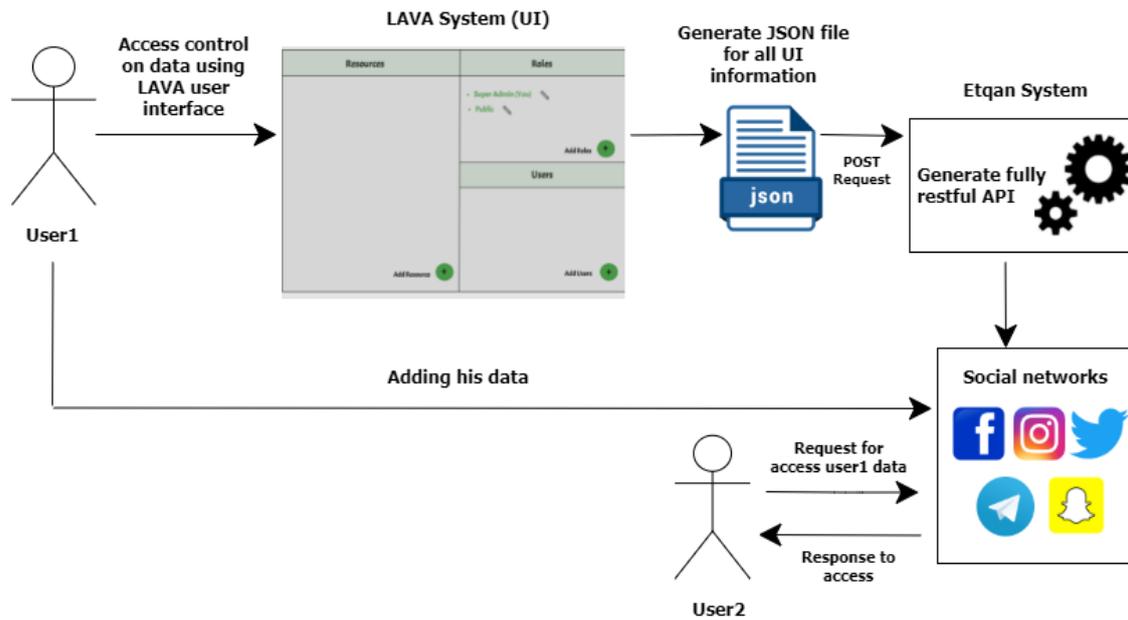


Figure 1.1: Overall system components.

In this work, the researcher discussing Lava system which will help end-users to control their data in social networks using simple and easy interfaces.

1.7. Research Outline

This thesis consists of six chapters:

1. Chapter one is an introduction to the topic, and it talks about the research problem, the objectives and introducing of the thesis itself.
2. Chapter two presents a detailed literature review and the state of the art of the topic.
3. Chapter three discusses the design iteration process with multiple usability experiments, and it explains how the final high-fidelity prototype was designed.
4. Chapter four discusses the implementation and code architecture of the proposed user interface system.
5. Chapter five presents the final usability experiment with two groups of participants and the data analysis.

6. Chapter six presents the conclusions and the future works.

Chapter 2: Literature Review

In this chapter we will describe the different types of prototyping and what is the project that can be suitable for each type, also we will talk about the importance of prototyping in the product development process, and the prototypes that used for access control in social networks, as well we will talk about human centered design prototypes.

2.1. Types of Prototyping:

The goal of the first part of this section is to present the different taxonomy of prototyping. (M. C. Yang, 2005) suggests three classifications of prototypes: throwaway, evolutionary, or incremental. A brief definition of a throwaway prototype is the one that is made in an early stage to help clarify requirements. While evolutionary prototype goes through iterative stages of development and evaluation, and finally the incremental prototype is the one that result from several stages.(Yang, 2005) Elverum and Welo mentioned the terms horizontal and vertical prototypes. In their definition, the horizontal prototype focuses on breadth and do not implement fully the product functionality to keep its main goal as a demonstration. In contrast, a vertical prototype implements selected functions of the final product (Elverum & Welo, 2015) .

Other authors classify prototypes by those that focus on form that related to the structure and those that focus on function which is the product of the structure (Camburn et al., 2017). Ulrich and Eppinger present a two dimension-classification: analytical vs physical and also focused vs comprehensive, a focused prototype accomplishes some features of the final product while most attributes of the final product are implemented by comprehensive prototypes (Ulrich, 2003) .

Another taxonomy presented in literature consists of the fidelity level of a prototype (Camburn et al., 2017). Camburn et al. categorize prototyping by a physical prototype and a virtual prototype (Camburn et al., 2017). The second part of the literature review will be the major types of prototyping, and we will focus on fidelity prototyping. Because the work will go through several iteration before developing the final product, and this will allow us to produce the best product in the shortest amount of time with the least number of design resources wasted.

2.1.1. Prototype fidelity:

Fidelity or resolution is defined as “*the levels of details presented in a prototype*”(Elverum & Welo, 2015). That is the extent to which the prototype is close to the final product. Fidelity can be linked to the availability of data and prototype functionality (Elverum & Welo, 2015). It reflects how “finished” the prototype is to the clients and users (Pyla, 2012).

The need for fidelity in prototypes depends on the project development stages. It is increasing with the project progression (Pyla, 2012). It should be taken into account that fidelity impacts the prototype design and validity of the usability tests (Zhou & Rau, 2019).

2.1.1.1. Low fidelity prototyping:

Low-fidelity prototypes are not “*faithful representations of the details of look, feel, and behaviour, but instead give rather high-level, more abstract impressions of the intended design*”. “*They do not have set the graphical design elements such as images, colour or typography*” (Pyla, 2012). Low-fidelity prototypes are used when design can be changed because they are flexible and cheap to iterate (Pyla, 2012).

Kaya et al. claims that low fidelity prototyping facilitates evaluation of different early stage ideas. They suggest that the material and tool used here should be user friendly and cheap

such as paper mock-ups (Kaya et al., 2018). During low fidelity prototyping, frequent communications must be accomplished along with a large number of product design iterations (Kaya et al., 2018).

Low fidelity prototyping is a low-cost way of collecting usability data. It gives an early opportunity to refine a design and identify problems through testing several low-fidelity prototypes quickly and presenting them to the stakeholders (Gerber & Carroll, 2012).

2.1.1.2. High fidelity prototyping:

High-fidelity prototypes are more familiar type of prototypes. They are “*effective tools to communicate and advocate a design solution to internal stakeholders and management*” (Houde and Hill, 1997, McCurdy et al., 2006) cited by (Elverum & Welo, 2015). Sauer and Sonderegger (2009) state that high fidelity prototyping is likely to give the client a positive impression and receive a better feedback (Zhou & Rau, 2019). However, high fidelity can be mistaken for a final design which causes confusions and misunderstanding between designers and clients (Gerber & Carroll, 2012).

2.1.2. “Wizard of Oz” Prototypes.

Wizard of Oz is a rarely used type of prototype according to R. Hartson and P. Pyla, it deceive the user into a high degree of interactivity. Its setup requires one slave computer which is controlled by a master one, the user uses the slave which is controlled by another person without the user knowledge (Pyla, 2012). The advantages of this prototyping are the strong user interaction, flexibility and adaptability (Pyla, 2012). It is useful to be used in areas such as testing the natural behaviour of users when interacted with the prototype (Pyla, 2012).

2.1.3. Virtual, Physical and Mixed Prototypes:

2.1.3.1. Virtual prototypes VP.

Virtual visualization systems are used such as Computer Aided Design (CAD) models. Many authors highlight the importance of these virtual models, Liker and Pereira demonstrate that they reduce the development time and cost. The authors compare the virtual models to physical models and states that the former is faster build and tested as well as less expensive compared to other prototype designs (Jeffrey K. Liker & Pereira, 2018).

Virtual Prototypes (VPs) are easy to use thanks to the advances of virtual reality technologies (Zhou & Rau, 2019) (Elverum & Welo, 2015) software tools such as CAD, computer aided manufacturing (CAM), Product Lifecycle Management (PLM) systems are used (Elverum & Welo, 2015). Different authors recommend using VP for detailed design stage such as colour, shape and dimensions (Zhou & Rau, 2019). VP can permit complex analysis and facilitate production and allow fast international communication of product design details (Camburn et al., 2017).

VPs are less expensive as well as more rapid than physical prototypes and they require less effort. Furthermore, their outcome proves to be of high performance and quality compared to physical prototypes (Camburn et al., 2017).

However, it is proven that VPs provide solely “*information of visual and auditory modality*” (Zhou & Rau, 2019) and users are not able to enjoy a physical interaction with VPs the way they are able to do with a real product (Zhou & Rau, 2019). Additionally, the authors admit that VP is categorized as having a low-fidelity level (Jeffrey K. Liker & Pereira, 2018)

2.1.3.2. Physical Prototypes.

Physical Prototypes (PP) means real direct physical interaction with a real physical hardware device (Pyla, 2012). PP can be using simple materials at hand and be crafted rapidly (Pyla, 2012). PPs are not expensive and they offer the look and feel to the product.

The authors confirm that even though design details are low fidelity, these prototypes are considered of higher fidelity in some ways as they are typically three-dimensional, embodied, and tangible. They are touchable and are manipulated physically (Liker & Pereira, 2018) (Pyla, 2012).

Many automotive companies prefer PP such as Toyota especially in the design of the primarily visual components such as car bodies (Liker & Pereira, 2018). PPs are time consuming and more demanding mainly with product with more complexity (Elverum & Welo, 2015).

2.1.3.3. Mixed prototypes

The process of combining virtual and physical elements in one model is called mixed prototyping (Camburn et al., 2017). This method is mainly used to help in usability testing (Zhou & Rau, 2019) and in more complex systems that required more than one approach at later stages of prototyping (Camburn et al., 2017). Mixed prototyping helps evaluating the behavior of the system (Camburn et al., 2017) and that is why is encouraged by many authors to be used in a product design (Zhou & Rau, 2019).

2.1.4. Augmented Reality Prototype ARP.

Augmented Reality Prototype (ARP) saves time and cost and it also provides the users during the early stage with a sensory interaction (touch, sight, sound) (Camburn et al., 2017). A study made by Bruno et al. (2010) showed that ARP like PP provides users equivalent usability

data. Examples of technologies that are used in ARP, are augmented reality, embedded screens, motion tracking, QR code tracking, and projection mapping (Camburn et al., 2017).

2.2. Types of projects Prototyping can be suitable for:

The choice of the prototype is based on the product, the situation, the requirements and the time scale. This decision will be made by the designers or programmers (Liker & Pereira, 2018) (Zink et al., 2018).

The literature review shows that if the main feature of the project is the physicality of the product or system, then prototyping that provide physical interaction is needed such as PP which affords to the user an insight of the look and feel of the product (Pyla, 2012).

Virtual prototyping is used by firms that look for high fidelity, flexibility, speed and design evaluation as well as international communication and workload balance (Liker & Pereira, 2018).

As mentioned by X. Zhou et al. virtual prototyping is an efficiently choice in finding the aesthetic and functional problems (Zhou & Rau, 2019). The authors state that mixed prototype can be used at a later stage when all requirements are mostly defined. This stage is the project design re-evaluation phase (Zhou & Rau, 2019).

Low-fidelity physical prototype can be used to help with the functional virtual prototype as it will not cost a large amount of prototyping resources and it increases the user's perception of the product (Zhou & Rau, 2019).

2.3. The Importance of Prototyping in the product development process:

Ulrich and Eppinger (p.295) declare that “prototypes enrich communication with top management, vendors, partners, extended team members, customers and investors”(Ulrich, 2003)

Previous studies proved that prototyping helps communicating and discussing ideas between the team of designers and the stakeholders. It accelerates manufacturing process technology development (Elverum & Welo, 2015).

Specifically speaking, prototyping aids validating design aspects and exploring the problems-solving between the designers and managers (Liker & Pereira, 2018) (Elverum & Welo, 2015).

Furthermore, prototyping is used in risk management and technical uncertainty. It is a tool to identify early unexpected problems (Elverum & Welo, 2015). It guarantees getting valuable feedback from the customers (Liker & Pereira, 2018). Many authors highlight that prototyping increases the speed and performance of the product development process (Liker & Pereira, 2018).

Also, prototyping encourages the customers to participate directly and involve in an early stage. This helps changing and improving the design (Pyla, 2012). It also gives an opportunity to the users to use the product model (Nam et al., n.d.).

A considerable body of literature highlights the importance of prototyping within agile product development process. It decreases uncertainty as it provides a simplified model of the final product where other details can be added at a later stage (Zink et al., 2018).

In other words, prototyping is crucial as it enables testing the user-interaction with the product and with every iteration there will be clearer insights of the project and understanding of the users' requirements (Wattanagul & Limpiyakorn, 2016) (Zink et al., 2018).

In addition to the above reasons, prototyping is an effective convincing tool used by the designers or software developer (Elverum & Welo, 2015). Prototyping is a way to reduce time and cost (Liker & Pereira, 2018).

Prototyping provides a communication medium where questions can be raised and answered, and the decisions will be supported by factual data (Gerber & Carroll, 2012). Other authors state that prototyping must be included in almost all product, service, and systems development stages (Camburn et al., 2017).

2.4. How to make prototyping more effective and less costly?

To make prototyping more effective, we should look for ways to reduce cost and time (Liker & Pereira, 2018). Clark and Wheelwright (1993) study on the appliance industry suggested that 'periodic prototyping' or using frequent prototyping to be a useful strategy that could lead to better quality and shorter lead times (Liker & Pereira, 2018).

Prototyping can be improving by using different open-source technologies and software databases. Several online sharing platforms such as Github aid downloading many repositories (Camburn et al., 2017).

(Camburn et al., 2017) suggested that hacking commercial products and reusing them in prototypes is likely to reduce the effort and the cost of prototyping (Camburn et al., 2017). X. W. Zhang et al. proposed using 3D printers and focused shape models 3DPMs (Min et al., 2019) as they are less expensive and do not need time consuming training (Camburn et al., 2017). Moreover, X. Zhou et al. thinks that designers should not choose a fully functional prototype earlier because it is likely to be time and resource consuming (Zhou & Rau, 2019).

E. Kaya et al. add that if prototyping is user-friendly and not complex, it can reduce the prototyping cost and time (Kaya et al., 2018).

Most literature does not provide a clear strategy to help designers choose the appropriate prototype taxonomy. A mixture of prototyping and techniques which have been suggested by several authors would be an efficient strategy to design a product. Each prototype has different features. Some prototypes would be efficient to be used at earlier stage whereas others can be used in a later phase in the product development process.

Recently, many authors highlighted the need to benefit for the advanced technology and encouraged designers to use online tools to reduce time and cost (Thomas C Reeves & Patricia M Reeves, 2012).

2.5. Access control prototype for social networks:

Different tools have been proposed in the literature related to access control prototyping in social networks, such as (M. Anwar et al., 2010) who provided a privacy enhanced visualization tool for Facebook style social network systems (FSNSs) to make reflective decision appraisal simpler while safeguarding the privacy of potential accessors. The users can define their access control policies by: (a) Visually displaying their expanded social graph neighborhood, (b) enabling them to inspect their profile in their expanded neighborhood from a possible accessor's point of view. This visualization strategy assists a profile owner in assessing her policies by demonstrating how prospective accessors in an expanded neighborhood are topologically connected with her and enables her to visually examine her policies using a mirror-like facility from an accessor of her choosing. This approach promotes reflective review of FSNSs' policies of entry, crossing, and contact (Anwar et al., 2010).

An approach for visualization of access control systems based on triangular matrices showed the visualization principle of access control based on the proposed model visualization,

namely: the construction of the model, interpretation of the relationship of rights by the location of the triangle and interpretation of a form of permission by color. The model suggested helps users to view permit nesting, by enabling permissions to be allocated between the whole branches of the role-tree (Kolomeets et al., 2019).

2.6. Human centered design prototype:

Human centered design prototype methodology was used by (Wardhana et al., 2017) that presented the use of content control applications on cell phones, which can enhance collaboration between children and parents. Wardhana et al. gained parental control application user interfaces that allow children to choose the content of apps and meet the demands of parents in educating children about the content of the application being used, allowing parents to educate a kid in order to let kids think about and make judgments about a problem (Wardhana et al., 2017).

2.7. Access control in social networks:

The “Solid” project (Mansour et al., 2016) has introduced a decentralized platform for social Web applications. Users' data is managed independently of the applications, where the data ownership remains for the user who created it, the user stores their data in a Web-accessible personal online data-store or as they call it a pod. One or more Pods are assigned to each user from different providers. A defined protocol is established to allow applications to access users' data with a decentralized authentication mechanism. Users in this model are given the full control over their data. In this decentralized architecture, applications can operate on users' data wherever it is stored.

An access control scheme for Online Social Networks (OSN) was suggested by (Pang & Zhang, 2015). They focused on public information, and took Facebook as an example. They

aimed to show that through user's attributes, common interests and activities within public information they can group users. They defined public information as the information exists in public domains such as Wikipedia and Bing Map. Pang and Zhang model contains information of:

1. Users and their social relationships.
2. Public information and their connections.
3. Links between users and public information.

They created a graph for users and their relationships, and another graph which depicts the links among public information. However, the links between users and public information are established with user interaction with public information, after establishing the links they used hybrid logic to define the access control scheme. The question about their policy is whether the system would generate an access control policy out of user's intention? And from a usability perspective, how it will be easily for a user to establish access control based on their formulas.

A novel user-to-user relationship-based access control (UURAC) model for OSN systems was proposed by (Cheng et al., 2016) which uses regular expression notation to create policy specifications. Authors proposed an algorithm which calculates whether the relationship between users and a given access request exists. After that the authors validated the feasibility of such approach by implementing a prototype system and evaluated the performance of the algorithm. Their model components consist of:

- Accessing user (U_a): a human user who performs activities.
- Actions: an abstract function initiated by accessing user against target.
- Target is the recipient of an action, can be either.
- Target User (u_t).
- Target Resource (r_t), which has a relationship with a controlling user (u_c).

- Access Request denotes an accessing user's request of a certain type of action against a target.
- Policy defines the rules according to which authorization is regulated.
- User to user relationship (U2U).
- User to resource relationship (U2R).

Their solution provided a solid foundation for more advanced relationship based on access control in the future. In (Sachan & Emmanuel, 2011) a bitvector transform based access control mechanism for multimedia contents in social media was applied. The study showed the efficiency of the proposed approach by means of Mathematical analysis and experimental results. Access rights associated with the contents were organized into an M dimensional space. Each dimension corresponds to a specific access policy.

A Hadoop based model for a conceptual storage to solve the security issues when transmitting data was suggested by (Gupta et al., 2016). The model is intended to be used by organizations in their own cloud systems; it allows the organization to store data ranging from confidential to public on the same cloud. In addition, it considers how valuable the data is for the organization and the number of times data have been accessed within a certain period. The model is a combination of RBAC and Hadoop distributed file system uses a normalization method to decide the access on resources by distributing the data over several servers.

An approach to facilitate the defense of shared data associated with many users in OSNs was proposed by (A. K. R. Praveena, B. S. D. Bhavani, 2013). Their proposal contained an access control replica to capture the essence of multiparty agreement requirements, along with a multiparty strategy requirement scheme and a policy enforcement mechanism. They introduced a proof-of-concept prototype of their model as part of an application in Facebook by creating an experimental social network using synthetic data which used to test

the efficacy of the semantic reasoning-based approaches. They demonstrated that a group of users could collude with one another to manipulate the final access control decision.

An automatic semantic annotation system that uses knowledge based and linguistic tools is proposed by (Imran-Daud et al., 2016). The system automatically identifies sensitive information according to the privacy requirements and the current reader by associating a meaning to the information before it is published.

An access control model which supports relation names and properties that is being defined and customized by users was introduced by (Antonio Tapiador, Diego Carrera, 2012). These customizations are unidirectional and flexible in such ways that don't constrain access control design. The key work they introduced is the application of their work to social media. Actors define their custom relations with other users in social media such as a friend or a partner. These defined relationships are equivalent to roles, after that. Actors can assign permissions to relations, which can include read wall or post to wall. Additionally, actors can make ties using those relations. They were able to test their system in a videoconference application, where they defined relationships between social entities including individuals, and social events.

This literature review presented different ways to make prototyping more effective. But the word effective was only defined by time and cost by most authors. Also, it presented the methods used in social networks for access control. The focus of the literature was on the users and stakeholders. The authors do not highlight the importance for the designers in this process. Time can be reduced, and the choice of prototyping depends mainly on the software used whether the designer is highly skilled and professional in using it.

Chapter 3: User Interface (UI) Design

In this chapter, we will discuss the design iteration process for the user interface; we used a user centered design approach which starts with user's identification that leads to design a low-fidelity prototype to the final high-fidelity prototype.

3.1. Development Method (Agile Methodology Scrum)

Currently, there are many methodologies for software development such as: Agile, Waterfall, Rapid Application Development (RAD), DevOps methodologies and others, to decide on which method to use, the traditional methods were compared with agile (STOICA et al., 2013) as table 3.1 below :

Table 3.1: Deference between Agile Methodology and Traditional Approaches.

	Traditional Approach	Agile Approach
Development life cycle	Linear; Life-cycle model (waterfall, spiral or some variation)	Iterative; The evolutionary-delivery model
Style of development	Anticipatory	Adaptive
Requirements	Knowable early, largely stable; Clearly defined and documented	Emergent, rapid change, unknown – Discovered during the project
Architecture	Design for current and predictable requirements	Design for current requirements
Management	Process-centric; Command and control	People-centric; Leadership and collaboration
Documentation	Heavy / detailed Explicit knowledge	Light (replaced by face to face communication) Tacit knowledge
Goal	Tend to be change averse	Embrace change
Team members	Distributed team of specialists; Plan-oriented, adequate skills access to external knowledge	Agile, knowledgeable, collocated and collaborative; Co-location of generalist senior technical staff;

To develop the UI the scrum agile methodology was used; Scrum development is a method of software development that consists of an iterative and incremental approach. It is used to develop software products that meet the needs of the customers.

A scrum board was created. The board was split into different tracks. For each track several tickets were added or removed, and the agreed methodology was as follow:

- A sprint of two weeks is agreed.
- A new task or bug is identified.
- A ticket is created for the task.
- The ticket is then estimated on how many points it needs to be completed, where each point equals one full working day per one person
- A deadline is set for the ticket.
- A colour code is used, which can be changed during the ticket life cycle, where the following conventions are used:
 - Green: The ticket is healthy. There is no blockage, no problems on implementing it. It is going according to the plan.
 - Amber: The ticket is getting delayed, and the amount of work may not be finished on time.
 - Red: The ticket is blocked, it is waiting for another dependency, or there is a problem that cannot be solved and needs attention.
 - Blue: The ticket is finished and ready for review and presentations.
 - Grey: The ticket was cancelled.
 - Black: The ticket was successfully completed.

- Once the ticket is completed, it should be reviewed by the product owner; it is then moved to another track and has its colour changed to blue.
- The tickets that are yet To-Do are not assigned any colour.
- The number of tickets that are selected for execution depends on the estimations given for each ticket, the length of the sprint, the availability of the researchers, and the current resources.

Figure 3.1 shows a screenshot of the agile board that was used.

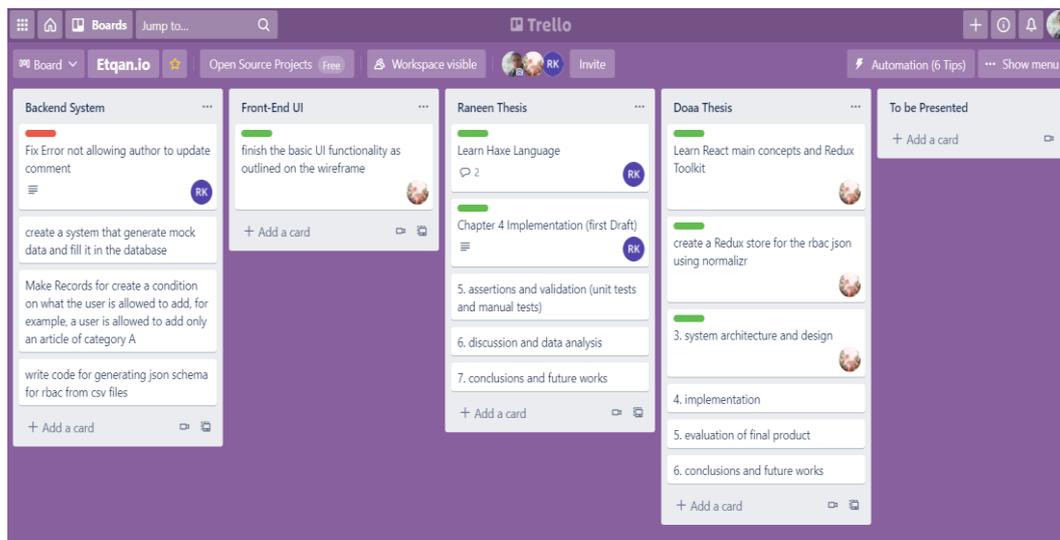


Figure 3.1: LAVA Scrum Board

3.2. User Centered Design Methodology (UCD)

This research adapts a user-centered design approach (UCD) which puts the user experience as a priority and gives the end-user the opportunity to influence how the interface is developed (Galitz, 2007). The term UCD was originated in 1980 by researcher Donald Norman and has been used since widely (Abrás et al., 2004). UCD involves end-users in the early stages of design and keeps them involved in all design iterations and phases. Developing a UI to allow normal users to manage role-based access control is a challenging problem, not

being able to address users' needs and abilities will result in frustrated users who will not use the product (Galitz, 2007).

The phases of the user-centered design approach that are used in this study are:

- 1. Identification:** in this phase, the main goal is to understand who the users are, what they will use the product for and under what conditions, and the context of the application. It is also important to understand the needs of the users to use the application, and have a concrete answer to the questions why they need this application.
- 2. Requirements:** in this phase, the main goal is to understand users' goals and the product features that will allow them to achieve these goals
- 3. Design:** this phase is conducted in stages, starting from a simple concept then building up to a complete design.
- 4. Evaluation:** several usability tests will be conducted with actual users to evaluate the designs. The evaluation is done at the end of each stage in the design phase.

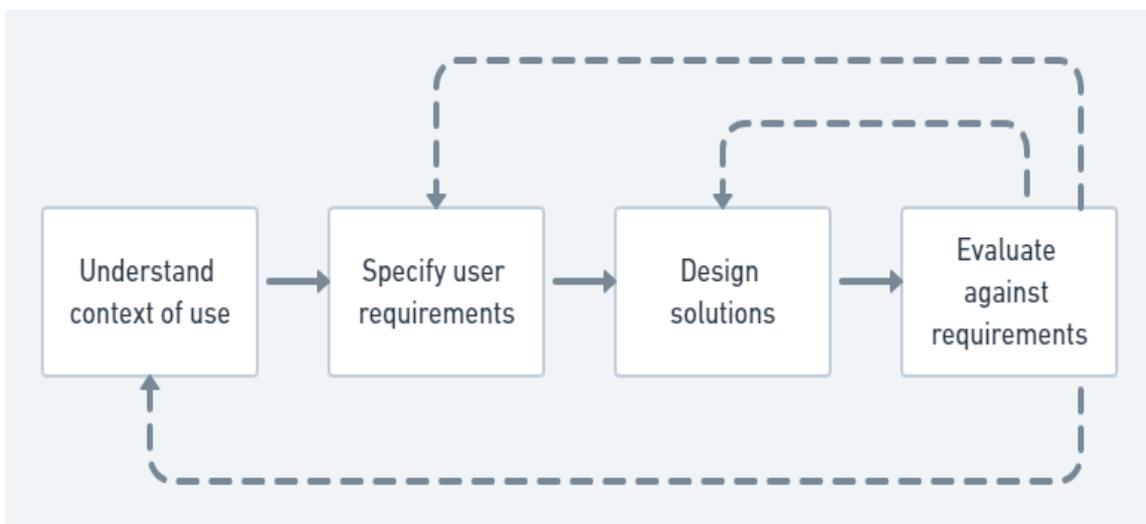


Figure 3.2: UCD phases.

The selected phases was chosen to use because UCD is an iterative approach that focuses on the users and understanding their needs in all phases in order to get a final product that reflects the user's needs (Interaction Design Foundation, n.d.).

In the first stage of the UCD, to identify the user's needs, an online survey was conducted, a random group of targeted users was selected to complete the survey, and existing online social networks (OSN) user interfaces were assessed for user awareness of privacy and security. In the second phase, to specify the requirements, an expert interview was conducted, different personas were developed, and the requirements were outlined and analyzed. In the third stage the design started by making low-fidelity prototypes and working its way towards a high-fidelity one. At the end of each design, a usability test was conducted to ensure that the application would be useful and usable for the end-user. In the final evaluation iteration, the final design was evaluated against the requirements.

To follow UCD methodology the researcher will go through the steps shown in Figure 3.3 that will be described in detail in this chapter.

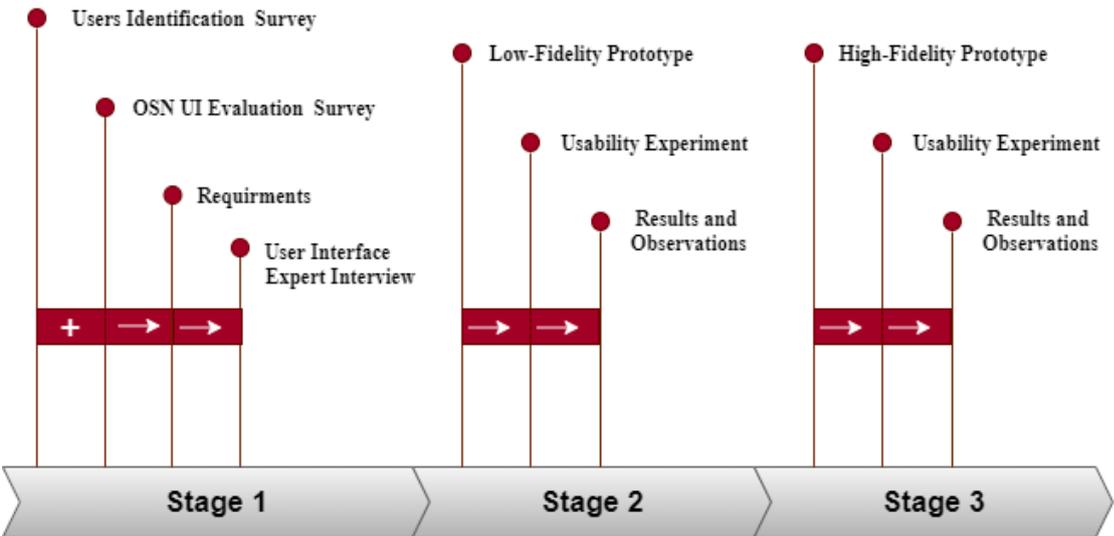


Figure 3.3: UI design steps

3.2.1. Identification

To understand users and their awareness of their privacy on social media, and whether they can understand the concept of role-based access control, an online survey was conducted. The survey aimed to collect answers about how users use social media, what they share, and whether they take their privacy into consideration. Also in order to evaluate the existing online social networks, another experiment was conducted to evaluate the effectiveness of these interfaces in providing facilities to users to control their privacy and share their data with other users.

3.2.2. Users Identification Survey

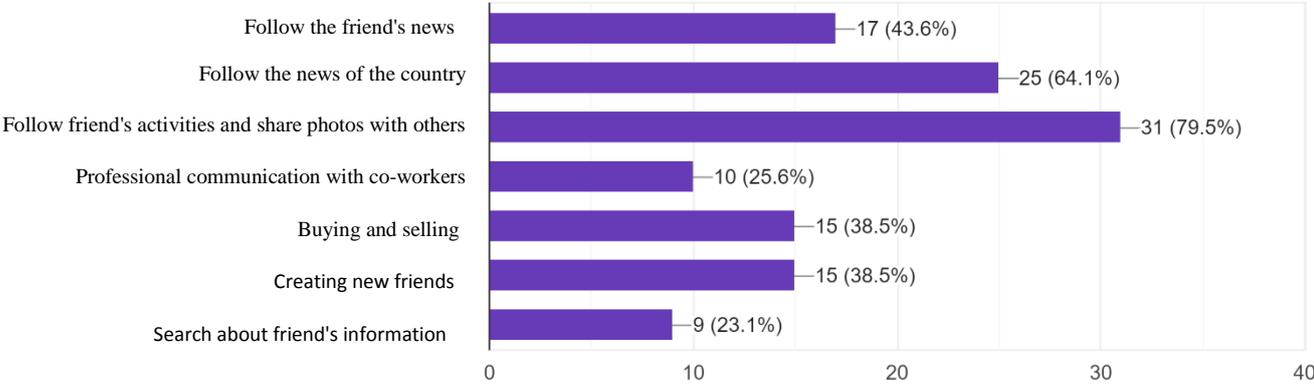
The questions of the survey were split into three sections: section one is to get the basic information about the users, section two is to identify their knowledge of protecting their online data and preserving their privacy, and the last section is to check their ability and need to use new user interfaces in order to configure their access control permissions and security settings in social networks. The targeted group (around 70 people) was regular users of OSN, who usually have more than one account with two different OSN.

The response of the survey showed that the majority of users are using social networks to follow their friends' activities and share their photos with others (79.5%) as shown in graph 3.1. However, Graph 3.2 shows that 48.7% of them do not know precisely the exact people their data might be accessed by. In this same context, 41% of them said that they set the privacy settings before sharing their information on social networks, as Graph 3.3 shows.

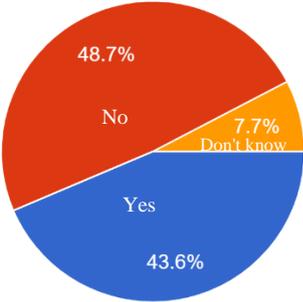
Relating to how much people trust social networks, 66.7% of them said that they do not feel safe sharing their data when using social networks, as shown in Graph 3.4. At the same time, Graph 3.5 depicts that 74.4% of users do not believe that social networks preserve the privacy

of their data since they believe that they lose ownership of their information at the moment they post it on social networks.

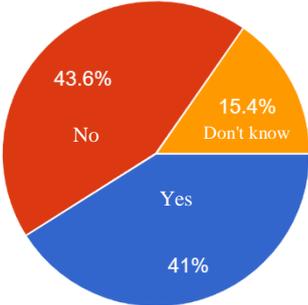
Regarding the UI of social media networks, 75% agreed that it's better to let the user take a part in designing the UI, as shown in Graph 3.6. On the other hand, Graph 3.7 shows that 70% want to use more flexible user interfaces to control the access on their data in social networks. Graph 3.8 shows that 65% have the ability to use new user interfaces for access control. For more details about the questionnaire, you can see Appendix A.



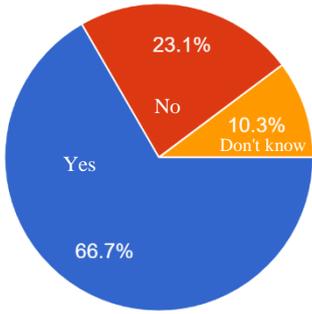
Graph 3.1: Percentage of participant activities



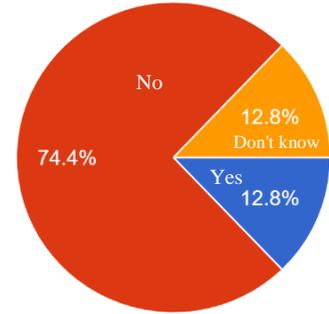
Graph 3.2 : Percentage of user knowledge about the people who can see his information on social networks.



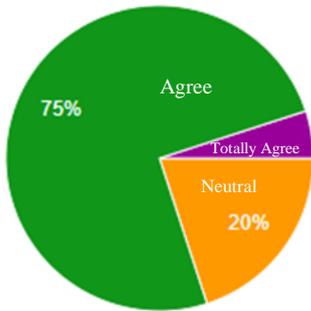
Graph 3.3 : Percentage of users who set a privacy settings before share their information.



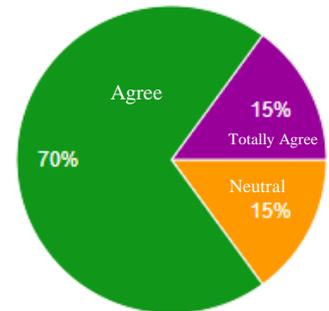
Graph 3.4 : Percentage of users that don't feel safe when using social media network.



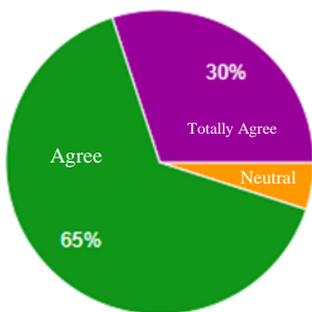
Graph 3.5 : Percentage of user believes that their privacy is protected by social networks.



Graph 3.6: Percentage of users that Agreed to take a part of designing UI.



Graph 3.7: Percentage pf users that need more flexible UI to access control their data in social networks.



Graph 3.8 : Percentage of users that have the ability to use new UI for access control.

3.2.3. Online Social Network UI evaluation survey

An experiment was conducted with randomly selected users who meet the criteria to participate in the survey. However, the criteria are that they must have two accounts, one with Facebook and the other with Twitter, to perform the tasks in the experiment. The setup was in a normal setting, inside a computer lab at Hebron University. The participants were asked to perform a set of tasks with their social media accounts, Facebook and Twitter. The researcher observed the participants while performing the tasks and took notes of what tasks were completed and how long each participant took to complete the tasks. At the end of the experiment, each participant was interviewed by the researcher in order to collect data about their performance.

To start the experiment, each participant was introduced with a scenario that may occur in real life while using any online social network. The scenario is:

"Imagine that you're on your phone and want to share your status on Facebook, but you would like some of your friends to be able to comment on this status but not all of them, you would also like to give permission for some of your friends to update your status itself. Does Facebook give you such control?"

Table 3.1 shows the tasks lists and the performance for 20 participants.

Table 3.2: Participants Tasks for UI Evaluation.

Tasks	No. Completed	Average Completion time (seconds)	No. Failed
Task 1: Share an image only to your family members	70%	101	30%
Task 2: allow only your friends to be able to comment on your post	90%	41	10%
Task 3: allow your friends to share your status.	75%	89	25%
Task 4: Share an image and except some friends from seeing it.	55%	123	45%
Task 5: allow only your best friends to be able to see your last post.	55%	112	45%
Task 6: Share an image on Facebook story and let anyone to see it.	100%	45	0%

3.3. Requirements

To outline the main requirements and features of the suggested UI system, an interview with a User Experience designer was conducted. The aim of this interview was to determine the main features the system should offer, how these features should be represented in the UI, and how the user should interact with the system. The User Experience designer who was interviewed was a person with 15 years of experience in web front-end development and user experience who worked on several projects and conducted many usability experiments.

A group of personas was developed to give a representation of the real target audience and focus the design on the needs of these personas. By the end of this phase, a set of detailed requirements were determined.

3.3.1. Interviewing with the UX (User Interface Expert)

A qualitative research method was conducted to gather further information; an interview in the form of a semi-structured was conducted with the designer expert. The semi-structured interview form typically has questions that can lead up to an open discussion and can make room for further clarifications and more questions to be added along with the interview (Jonathan Lazar et al., 2017). Table 3.2 summaries the digest conclusion of the interview.

Table 3.3: UX interview summary.

Question	Conclusion
<p>1. What main features should such a UI offer?</p>	<ul style="list-style-type: none"> • The user should be able to create groups and assign users to these groups. • The user should be able to see what each group and user can access while he edits or create the access control. • The user should be able to see as little as control as possible at one time.
<p>2. How to make users aware of the main features?</p>	<ul style="list-style-type: none"> • Add an interactive tutorial showing users how to add one basic feature step by step and they should be able to call this interactive tutorial at any time. • Every button should have a question mark next to it to give more information about this action. • Use very simple texts that can be understood by the users. • Try to have an intelligent system that gives users tips and tricks
<p>3. How to make such a system intuitive and easy to use?</p>	<ul style="list-style-type: none"> • Should have a few buttons. • Make the user focus on only one task at a time. • There should be consistency in colours. • Add controls that users are familiar with. • Use instant feedback for everything that the user does.
<p>4. What design approach should be considered?</p>	<p>Iterative and incremental design.</p>
<p>5. Give me an example of a similar use-case and how you tackled it.</p>	<p>In the electronic supermarket system, the user does the check out by himself, so he should know how the supermarket works.</p> <p>In this case, we gave the user enough time to enter the inputs, and we helped him to understand what he is doing.</p>

6. Should the system allow the user to learn about role-based access control or hide the concept from the user?	Users should learn about role-based access control while using the system, in simple language that they can understand.
7. What about mobile devices? how do you think the design should respond to the change in the viewport?	<ul style="list-style-type: none"> • It depends if we want the system to work on mobile devices or not. • If there is no need for users to see several things at the same time, we could make the system work on mobile devices.
8. Do you have any suggestions for the overall look and feel, colours, and graphics theme?	<ul style="list-style-type: none"> • Use a clear graphic theme. • Colours should be consistent. • Don't display several things on the screen, this makes the user confused. • Use a formal and clear text font.

3.3.2. Developing Personas

The researcher developed several personas and assigned them to use-cases in specific scenarios that resemble real-life settings. Personas are archetypical users whose goals and characteristics represent the needs of a larger group of users. In fact, a persona is a factionary character that represents the real target audience. Persona representations include behavioral patterns, motivations, goals, attitudes, skills, and background knowledge, in addition to the environment in which the persona operates.

With the help of the mental model methodology, the personas were more representative of the target audience. The concept of mental model refers to the users' mental representations of a specific concept and how they relate to it. It also focused on the users' motivations and needs to use a specific task (Abrams et al., 2004). Table 3.4 and table 3.5 below shows the suggested personas based of users requirements.

Table 3.4: Persona One.

Persona One		
Name	Mohammed	
Age	20	
Occupation	Student	
Skills	Communications, Collaboration.	
Motivation	Mohammed wants to share with his friends and followers updated on his personal experience as a student at Hebron University	
Scenario	Mohammed is at a lecture, he is listening to a guest speaker, he wants to add a photo, tag it with the speaker name and subject and post it to his wall in social network so that all his followers and friends can see and comment on the events	

Table 3.4: Persona Two.

Persona Two		
Name	Nour	
Age	24	
Occupation	Dentist	
Skills	Advising, Participation.	
Motivation	Nour wants to share her dental work with doctors.	
Scenario	Nour is at her clinic, she did a nice case for a patient, and she wants to share a picture of her job in her social networks accounts so that all her doctor friends can see her job.	

3.4. Design

Creating the user interface (UI) for the system will include two main stages:

- 1- Low- fidelity prototyping.
- 2- High- fidelity prototyping.

At first, we started with a simple low fidelity prototype to get feedback from users through conducting usability experiments using hand-written sketches as shown in Figure 3.4. Then a low-fidelity prototype was designed using Figma software (Figma, 2016), which is a tool for designing UI prototypes.

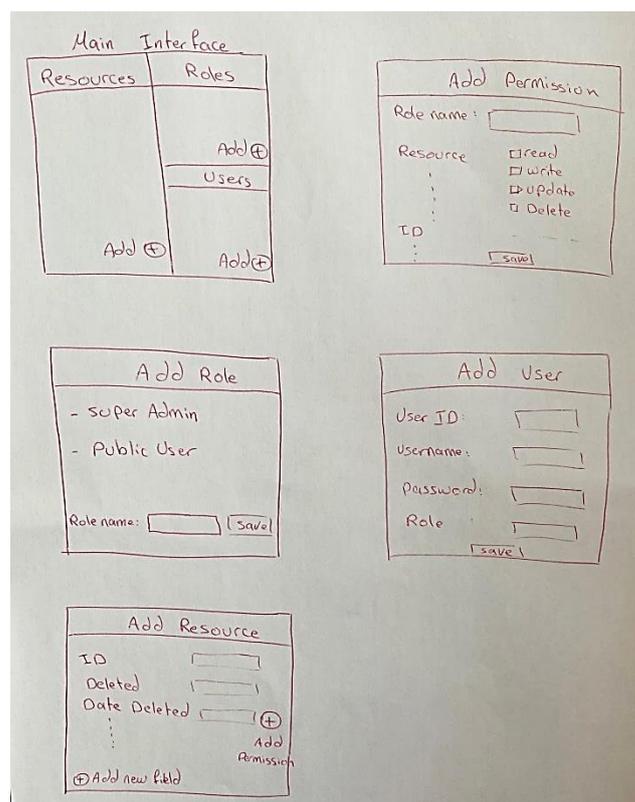


Figure 3.4: UI design sketch.

3.4.1. First Design (UI-1)

Figure 3.5 shows the main interface of the first suggested user interface (UI-1) system (LAVA). The first screen shows only three buttons, and each button is centered in its own section:

- Add Resource.
- Add Role.
- Add User

This minimalistic design makes it easy for the user to understand the system and decide on which button to click. Each button gives the user a clear goal and from the design, the user can start from any section as he/she wishes. He doesn't need to follow a specific order, nor do sections depend on each other:

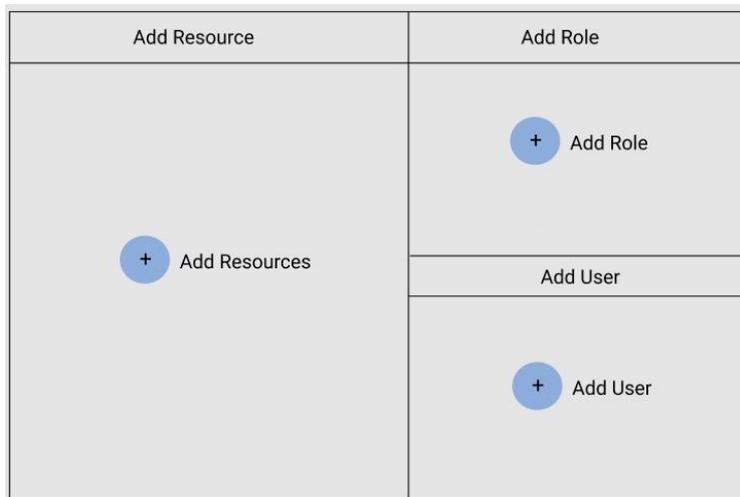


Figure 3.5: Main Interface (UI-1)

When the user clicks on the add role button, he will be presented with a new screen which is shown in Figure 3.6. The system by default has two roles that are created automatically, which are:

- Super Admin.
- Public user.

Etqan backend system requires these two roles, with the Super admin role assigned to the system's owner, who will be granted all possible permissions in the system. While the public user role is assigned to users who access the system without registering and have no role assigned to them. By default, this role is not granted any permission at all.

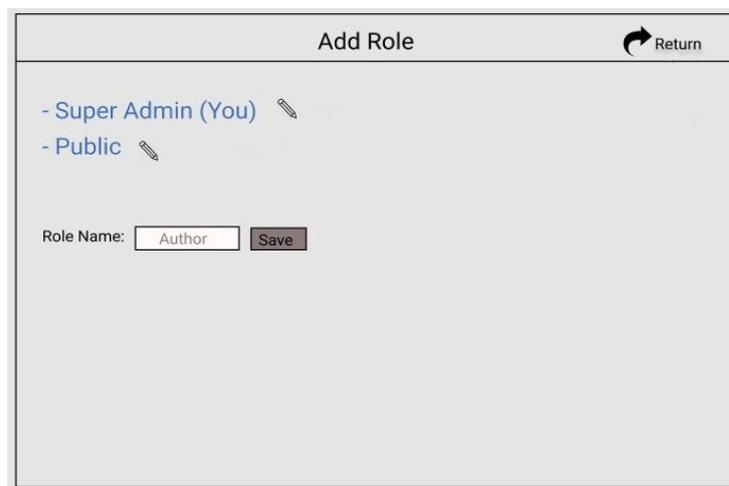


Figure 3.6: Add Role Interface (UI-1)

This screen is also easy to follow as it contains a limited number of actions. The user can add a new role by typing its name and clicking 'Save'. After that, the new role will be displayed on the interface, as shown in Figure 3.7.



Figure 3.7: Add New Role (UI-1)

Users can edit the names of roles by clicking on the pen icon next to each role name; they will be presented with a modal window as shown in Figure 3.8. The modal window shows the Super Admin edit interface which contains fields created by Etqan backend system: Role ID, Role Name, Date Created, Creator ID, and Note, the reader may note that some of these fields cannot be changed by the user as they are under the control by Etqan system.

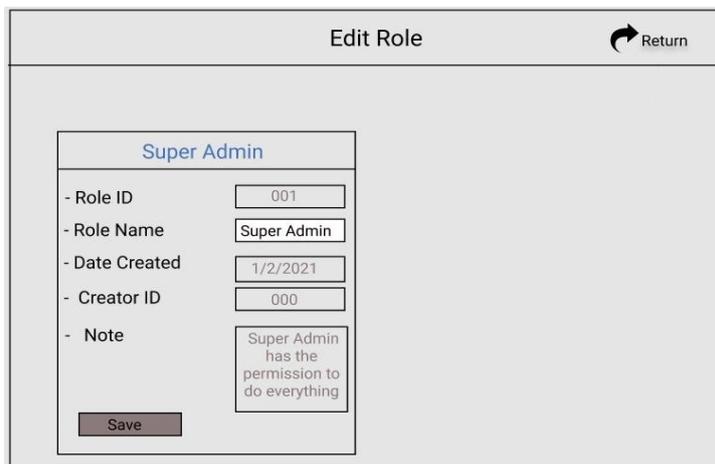


Figure 3.8: Edit Super Admin role (UI-1)

When the user clicks on the return button, he will return to the main window which is shown in Figure 3.5.

From there, the user can add a new resource by clicking on the add "new resource" button, in order to create new resource and set permissions to it. The user will be presented with the "Add Resource" screen, which is shown in Figure 3.9. The design follows the same principle of simplicity; it contains a list that displays resource names and a button to add a new one.

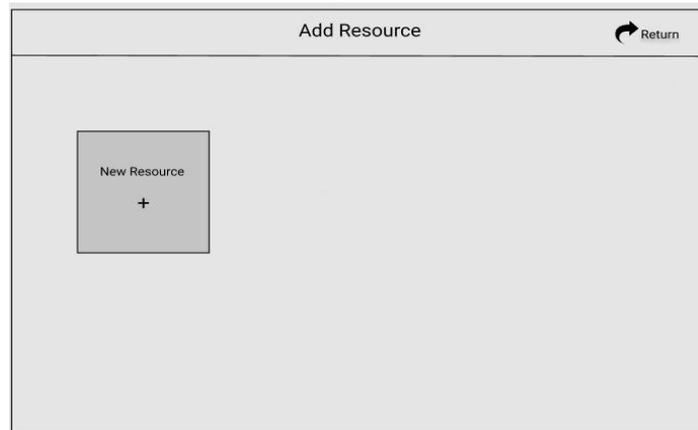


Figure 3.9: Add resource interface (UI-1)

When the user clicks on the New Resource button in Figure 3.9, the user will navigate to a new screen as shown in Figure 3.10. This screen contains all the fields that Etqan backend system uses to notify the user of their presence. The user has no permission to edit these fields, which are:

- ID: a unique integer value to identify the resource instance.
- Deleted: a Boolean flag to indicate whether the resource was deleted.
- Date Deleted: a date that will be saved as the deletion date.
- Date Created: a date to save when the resource instance was created.
- Date Modified: a date to store when the resource instance was last modified.
- Note: note added by the super admin.
- Creator ID: an integer that references the creator of the resource id.
- Owner ID: an integer that references the owner of the resource id.

From this screen, the user can add a new field command and set the field's permissions.

	Default Value	Data Type	Required
ID	021	Int	Yes
Deleted	false	bool	Yes
Date Deleted		DateTime	Yes
Date Created	2/2/2021	DateTime	Yes
Date Modified		DateTime	Yes
Note		String	Yes
Creator ID	001	Int	Yes
Owner ID	000	Int	Yes

Figure 3.10: Add new resource (UI-1).

When navigating to create a new field, a new screen will be presented which is shown in Figure 3.11. The user can set the name of the field, its default value, and its data type, and whether the field is required. The user can return from this screen back to the new resource screen.

Field Name: ▼ Select ▼ Select

- Bool
- Number
- Int
- String
- DateTime
- Resource

- Yes
- No

Save

Figure 3.11: Add new field (UI-1).

When the user asks to set permissions for the fields, she/he will be presented with the screen shown in Figure 3.12. It allows the user to set the permission at a field level, for all possible actions (Read, Write, Update, and Delete) and all existing roles.

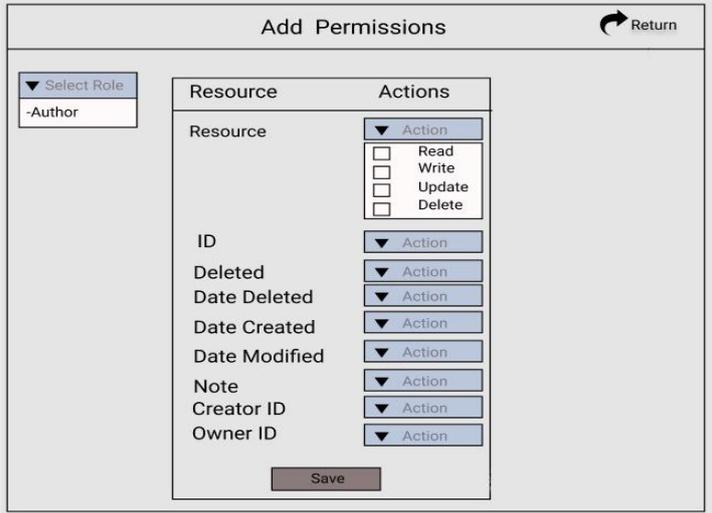


Figure 3.12: Add permission (UI-1).

When a user finishes adding the resource and clicks on save, the resource will be saved and the user will be redirected to the main resource display page, which now shows the added resource as shown in Figure 3.13.



Figure 3.13: new add resource interface (UI-1).

The interface allows the user to edit the resource name and the permissions at the resource level.

To add a new user who can have access to the resource, the user should click on “add user” from the main screen, which is shown in Figure 3.5, to navigate to the “add user” screen as shown in Figure 3.14. It also uses the same minimalistic design approach and resembles the “add resource” screen.

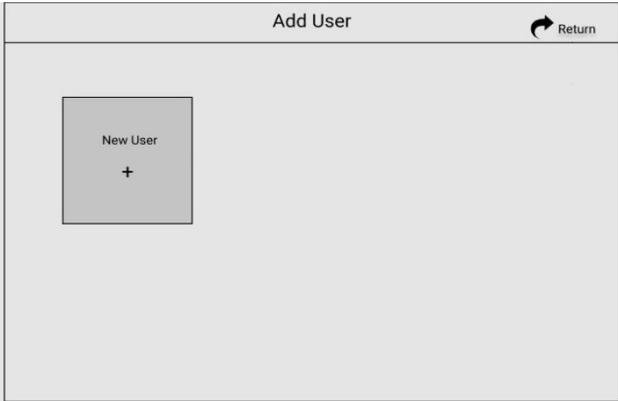


Figure 3.14: Add user interface (UI-1).

To add a new user, the buttons labeled New User should be clicked. A modal as shown in Figure 3.15 will be presented that asks to fill in the new user’s information like username, password and select a role for the new user.



Figure 3.15: new user info (UI-1).

3.4.1.1 Usability experiment

The low-fidelity prototype was evaluated in a usability experiment; the researcher used a mock-up model of the screen as a paper prototype and asked participants to perform tasks to achieve them. The participants were interviewed after the experiment to capture their feedback and talk about their experience with the design.

The goal of this experiment was:

1. Test the interface's design with regular social media users.
2. Get feedback about the UI from users who have no previous knowledge about access control.
3. Find out if the design can teach users about access control.
4. Do users find the design simple and intuitive to use?
5. Identify any design issues at an early stage.

Methodology:

We randomly selected five participants from Hebron University between the ages of 18 and 35. The selection criteria state that the person should not know anything about role-based access control.

The usability test began with a brief explanation of the system's purpose, followed by participants attempting to complete four tasks:

1. To add a resource.
2. To add a Role.
3. To add a User.
4. To set permissions for the resource.

The main researcher observed and took notes while the participants were performing the tasks.

Results:

Table 3.4 below shows how many tasks were completed by the participants.

Table 3.5: Tasks completion exp.1.

Participants	Add Resource	Add Role	Add User	Set permissions for resource
Participant 1	0	1	1	0
Participant 2	1	1	0	0
Participant 3	1	1	1	1
Participant 4	1	1	1	0
Participant 5	0	0	1	0
SUM	3	4	4	1
Success Rate	60%	80%	80%	20%

The results of the test depended on the sample space show that:

- 90% of the participants understood the main idea of the interfaces which is access control.
- 60% of participants completed the first task.
- 80% of participants completed the second task.
- 80% of participants completed the third task.
- 20% of participants completed the fourth task.

When they were asked about their experience using the interface:

- 50% of participants mentioned that they had trouble when they used the interface.
- 60% of participants indicated that they felt lost when they moved from one screen to another.

The researcher collected the following notes through observations:

1. Use the label “Add” on buttons rather than “Save” when adding a new role, user, or resource.
2. Do not suggest role names as this will make the user use the suggestion most of the time.
3. Give a more clear message on what to type in rather than giving the actual answer. Instead of displaying a suggested role name, display a message that says ‘ Enter a role name’.
4. Make it easier for users to identify the differences between users and roles.
5. Do not make the users make the mistake of assigning super-admin to some users.
6. Make the user more aware of the role that is created by default, especially the Super Admin role.
7. Change the location of the Return button and its label should be ‘Back’.
8. Give more space for the role name text control and make it obvious what makes a valid role name.
9. The location of entering the role name, the resource name, and the user name should be at the top not at the bottom, so that the controls won’t change their location when the list of entries is growing.
10. There should be careful planning for navigating between screens and presenting the users with modals. The less navigation and the fewer modals, the better, modal should

only contain tight information that needs attention to be completed otherwise they can be edited in place.

3.4.2. Second Design of User Interface (UI-2)

Based on the results from the first experiment, the interface went through a second stage of iteration. Figure 3.16 shows the new suggested interface which also includes three buttons, add resource, add rule and add users.

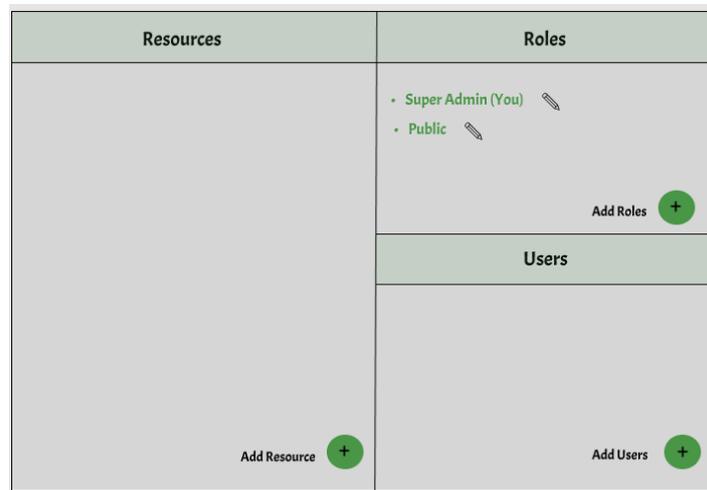


Figure 3.16: Main Interface (UI-2)

Adding resources was modified as shown in Figure 3.17 to make it easier to add a resource and reduce the number of screen navigations.

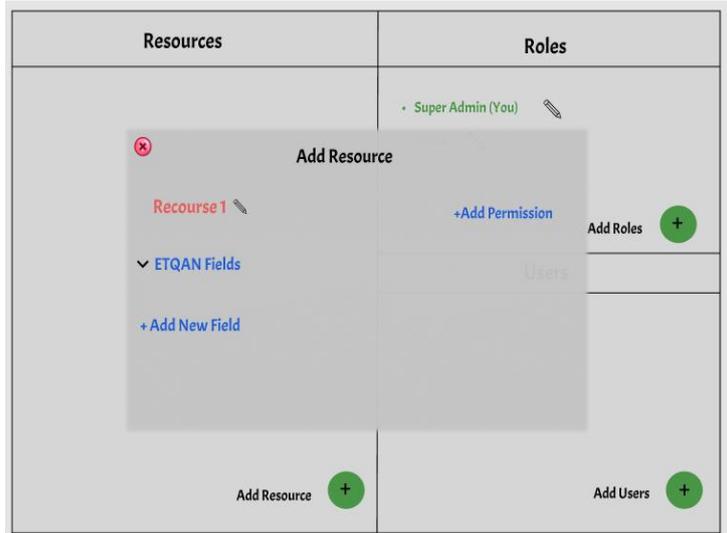


Figure 3.17: edit role name (UI-2)

The new interface keeps the main interface in the background and navigates through other actions in a wizard-style, where the user goes through them one by one. If he chooses to add a resource he will be presented with a new screen using the same new approach as shown in Figure 3.18.

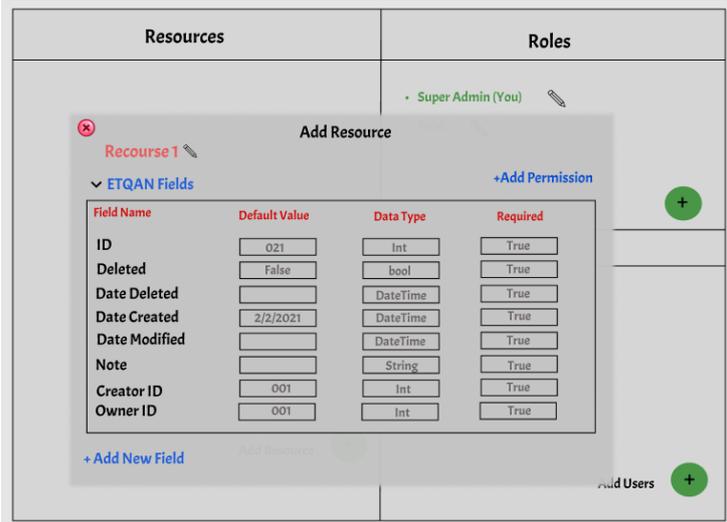


Figure 3.18: add resource (UI-2)

Adding a new field has become much simpler and can be performed in-place rather than navigating to a new screen. This makes it easier to navigate and the user will be more aware of the current location as in Figure 3.19:

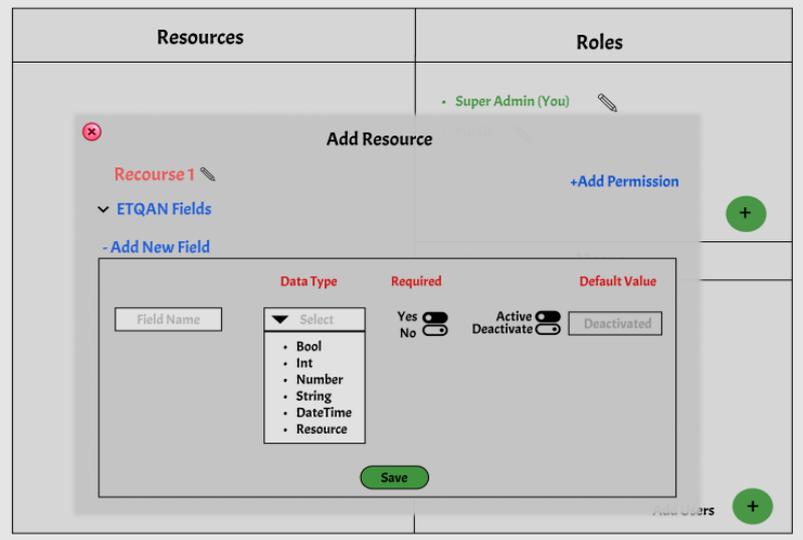


Figure 3.19: add new field (UI-2)

Respectively, adding new permissions can be accessed from the add new resource window, and then the user will be able to set the permissions on the fields level for each role:

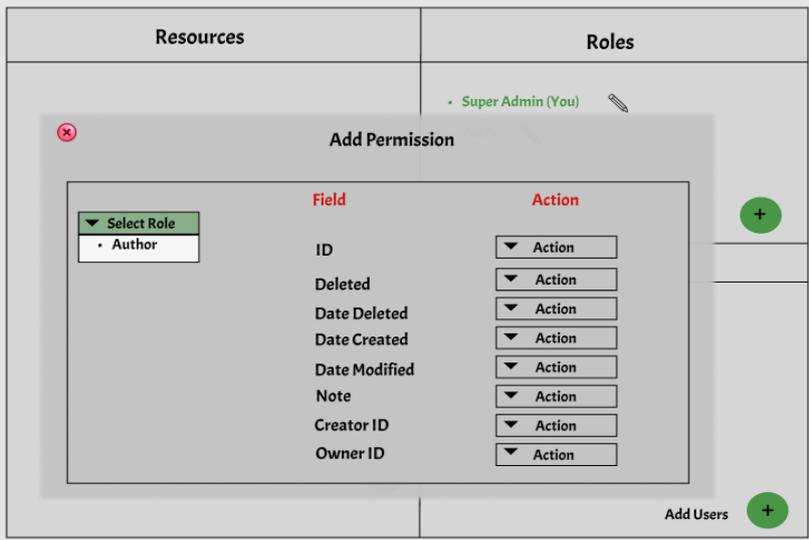


Figure 3.20: Set resource permission (UI-2)

Navigating through existing roles will cause the interface to respond and change the output according to the current role access level on the resource and as per each field as shown in Figure 3.21.

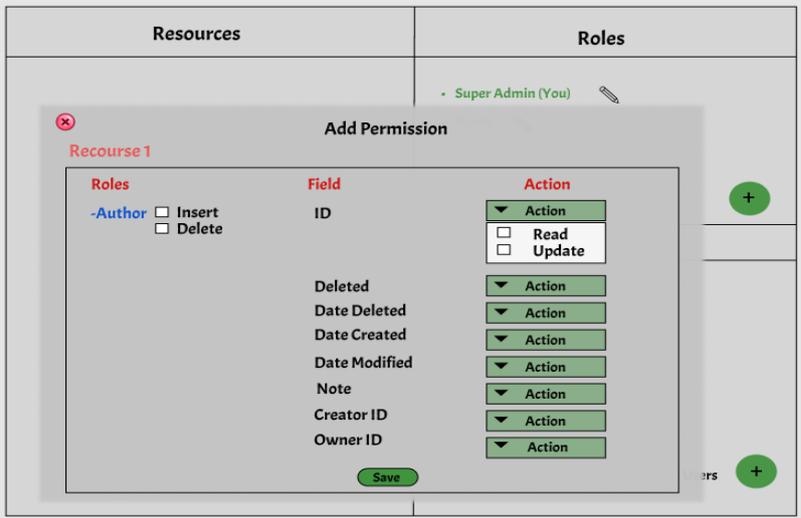


Figure 3.21: select field's actions (UI-2)

Figure 3.22 shows that adding a new role has become simpler and the user does not need to navigate away from the main window.

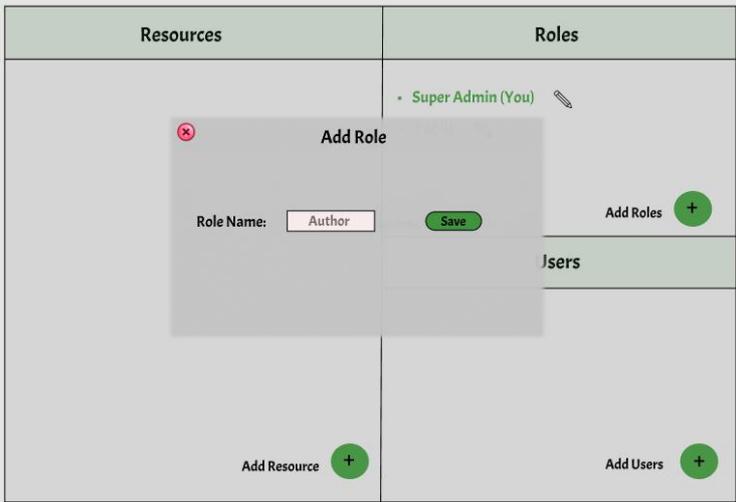


Figure 3.22: add role (UI-2)

The user can set permissions for each role on the resource level when the role is clicked on from the main screen as shown in Figure 3.23.

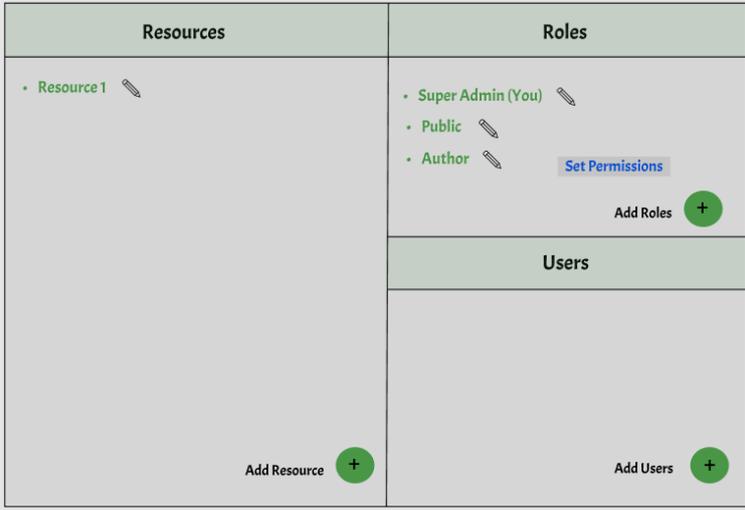


Figure 3.23: Set role permission icon (UI-2)

When the user clicks on set permission beside the permission name, figure 3.24 will display.

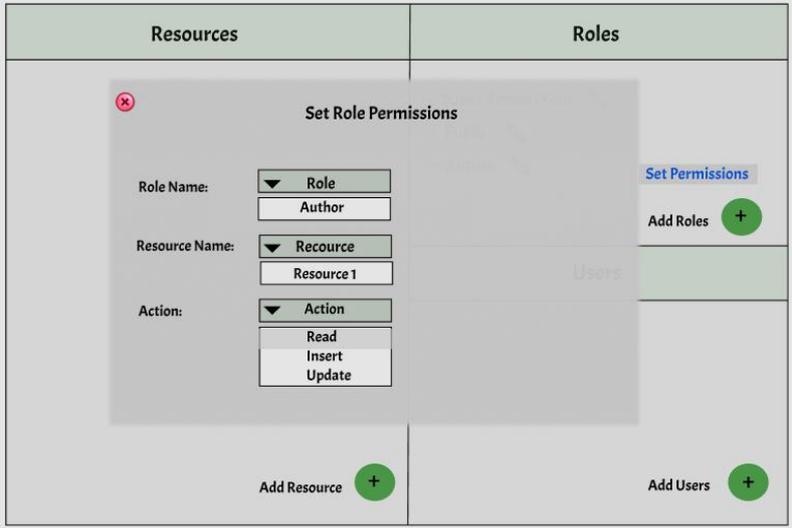


Figure 3.24: Set role permission

From the same window, for easy access, the user can also access permissions on fields level to be set for the selected role as shown in Figure 3.25.

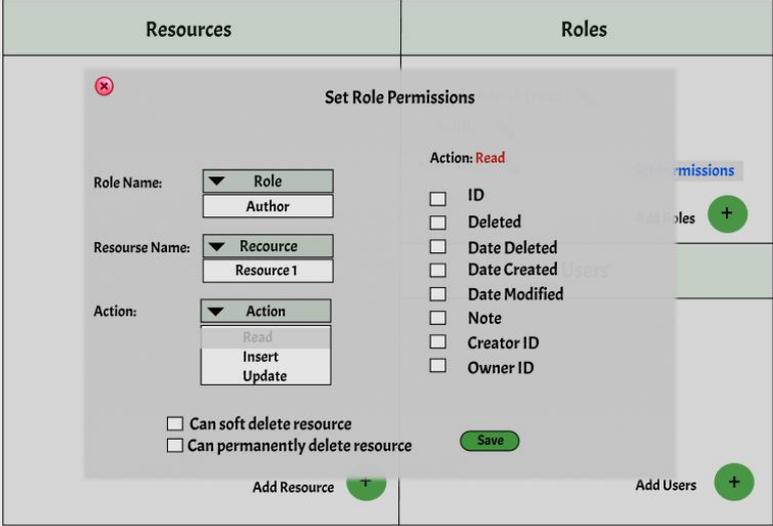


Figure 3.25: set action for role on resource (UI-2)

Adding a new user followed the same approach of reducing navigation between screens. This can also be added in a similar way as adding a new role as shown in Figure 3.26.

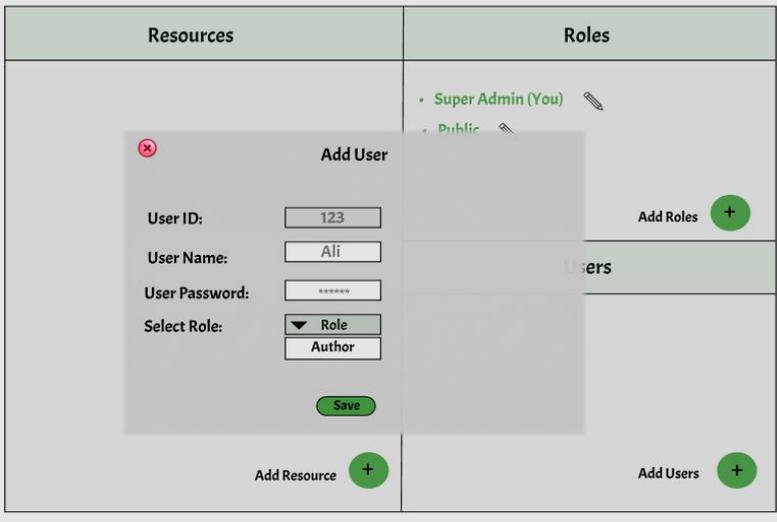


Figure 3.26: Add user (UI-2)

3.4.2.1 Second Usability (UI-2) test experiment

Similar goals and methodology were used for evaluating the second iteration of the interface. A random group of participants who are using social media between the ages of 18 and 40 years old were selected.

The usability test included an introduction, four tasks which included adding a resource, a role, a user, and a set of permissions for each resource and role. After that, the participants were interviewed to reflect on their experience and provide further feedback. The researcher observed and recorded notes while the participants were performing tasks. Each test session lasted for 30 minutes.

3.4.2.2 Second Experiment Participant Profiles

We did the second usability experiment with 10 participants; 5 of them were male and 5 were female, between the ages of 18 and 40 years. Table 3.5 below shows the details for each of the participants:

Table 3.6: Participants Information in exp.2.

Gender	Age	Job	No. of hours spent on social media	Social media used	Knowledge of access control
Male	35	Programmer	8	Facebook, Instagram, Twitter	Yes
Female	25	Dentist	5	Facebook, Instagram	No
Female	18	Accountant	7	Facebook, Instagram	No
Male	38	Teacher	4	Facebook, Twitter	Yes

Male	20	Student	6	Facebook, Instagram, Twitter	No
Female	27	Teacher	3	Facebook, Instagram	Yes
Female	28	Housewife	6	Instagram, Snapchat	No
Male	40	seller	5	Facebook	No
Female	25	Teacher	4	Facebook, Twitter	Yes
Male	21	Student	3	Facebook, Instagram, Twitter	No

The researcher collected background information about the participants to relate their performance in the experiment with their background information to detect any patterns that could give more insights into the design process.

Table 3.6 below shows the task completion for each participant. Knowing that each task (Add Resource, for example) requires many steps to be completed.

Table 3.7: task completion exp.2.

Participants	Add Resource	Add Role	Add User	Set permission for resource
Participant 1	1	1	1	1
Participant 2	1	1	1	0
Participant 3	0	1	1	0
Participant 4	1	1	1	1
Participant 5	0	1	1	0
Participant 6	1	1	1	1
Participant 7	1	1	1	1
Participant 8	1	1	1	0

Participant 9	1	1	1	1
Participant 10	1	1	1	1
SUM	8	10	10	6
Success Rate	80%	100%	100%	60%

Results of the test showed clearly that:

- All of the participants understood the main idea of the interfaces and the concept of access control.
- 80% of participants enjoyed when they tried the interfaces.
- 80% of participants completed the first task.
- 100% of participants completed the second task.
- 100% of participants completed the third task.
- 60% of participants completed the fourth task.
- 20% of participants experienced difficulty when they used the interfaces.
- 20% of participants had difficulties understanding how to add a new field for a resource and set a data type for this field.

Notable Quotes:

1. "I would not understand these Interfaces without an introduction about access control"
-User 2
2. "This idea will help me a lot to control my data on Facebook from annoying friends"
-User 8
3. "I cannot understand what data type means; I have to try this UI more than once to understand them"
-User 3

Some notes the researcher collected through observation and talking to the participants:

- Users must be educated about what access control is within the UI. An introductory video or an interactive tutorial can be played once a user uses the interface for the first time.
- Users were confused when trying to add a new field for a new resource because they didn't have a background on what data type was and how it could affect the field. This should be taken into consideration in the next design iteration.
- Some users were confused when they tried to add permissions because it was the first time and they were not sure what permissions were and what their effects were. They need training before using such a feature or they should see immediate feedback about the effect of assigning a permission to a role as shown in Figure 3.27 and Figure 3.28 below.

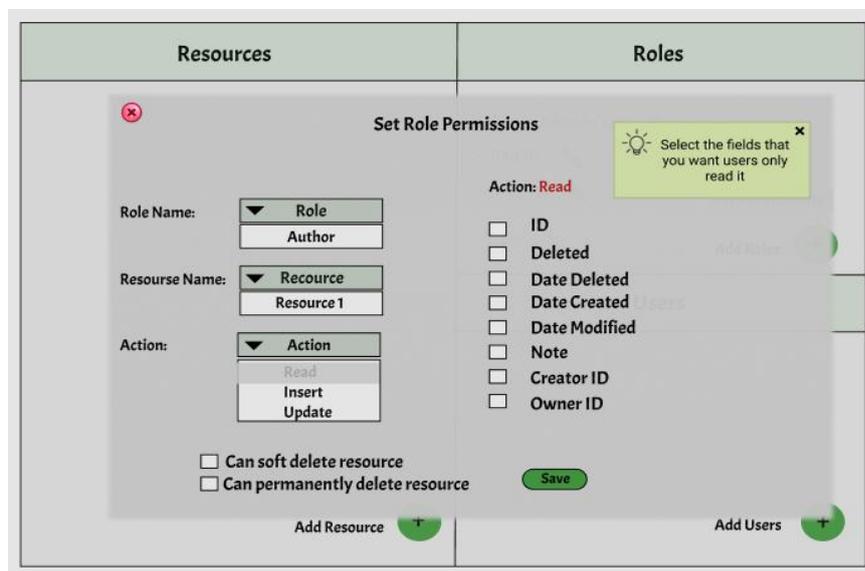


Figure 3.27: add permission hint 1

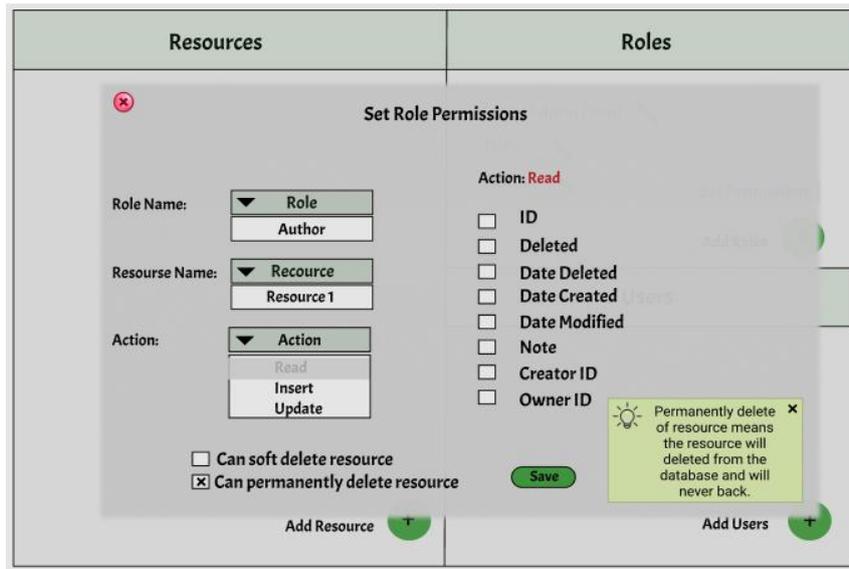


Figure 3.28: add permission hint2

Chapter 4: Implementation

In this chapter, we will describe the implementation process which starts with UI design till generating the JSON file.

As the UI is a web-based system, a framework for developing web UI must be chosen, there are many UI frameworks for the web, for the suggested UI there is a lot of interaction and the state of the UI changes frequently, it also provides many features to the end-users, and therefore a (Single Page Application) SPA framework must be chosen.

Currently, there are three main SPA frameworks leading the market: React, Angular, and Vue.js, to decide on which framework to use, each framework features were compared and analyzed as followed:

Table 4.1: Comparison between React, Angular and Vue.js.

	React	Angular	Vue.JS
Publisher	Jordan Walke	Misko Hevery	Evan You
License	MIT License	MIT License	MIT License
Community Size	109.7 KiB production 774.7 KiB development	167 KB production 1.2 MB development	30.67 KB production 279 KB development
Number of Updates	43	10	18
Documentation	The official Documentation is available.	Simple and to the point Documentation is available.	Adequate to work with Documentation.
External Libraries	React Query, Zustand, React Hook Form, React Responsive, Redux	Ng-bootstrap, Angular Google Maps, ngx-translate, ngx-charts.	AT-UIKit, Mint UI, CoreUI.
Startup Time	Quick	Longer due to its large codebase.	Quick.
Data binding	Uni-directional	Bi-directional	Bi-directional
Learning curve	Easier than Angular	A steep learning curve.	Small learning curve.
Flexibility	Most Flexible	Somewhat Flexible	Somewhat Flexible
Performance	high	Medium	High
Developer- friendly	If you want to have flexibility in the development environment.	If you want to use the structured-based framework	If you want to have separation of concerns.

Scalability	high	High	Low
Language	Java Script	Type Script	Type Script

After the comparison, the researcher decided on choosing Reach as the framework to develop the UI.

React supports a local state management system and a global state through using the Context AP feature. However, for complex states a separate library to manage the state is preferred as it gives more control over the state and allows a clear separation between the state and the UI.

Redux is a well-known management library that works with React and it was chosen as the state management system.

4.1. Role-Based Access Control UI System Description

The system enables users to control who can access their own data by a graphical user interface (GUI). They can create their roles and assign them to users. Any third-party applications or programming languages can be used with the system. All-access control logic is kept outside the code. By default; there are 2 types of access control:

1. Admin access which can have access to all resources to create, read, update and delete.
2. Public-guest that by default does not have access to any resource to perform any operation.

The user can then create their custom roles and assign permissions to them to access data. Figure 4.1 below describes the system which starts with the use of interfaces by users to access control over their data in social networks. The system then generates the JSON file which will contain the user's data and access control schema and store them in the database.

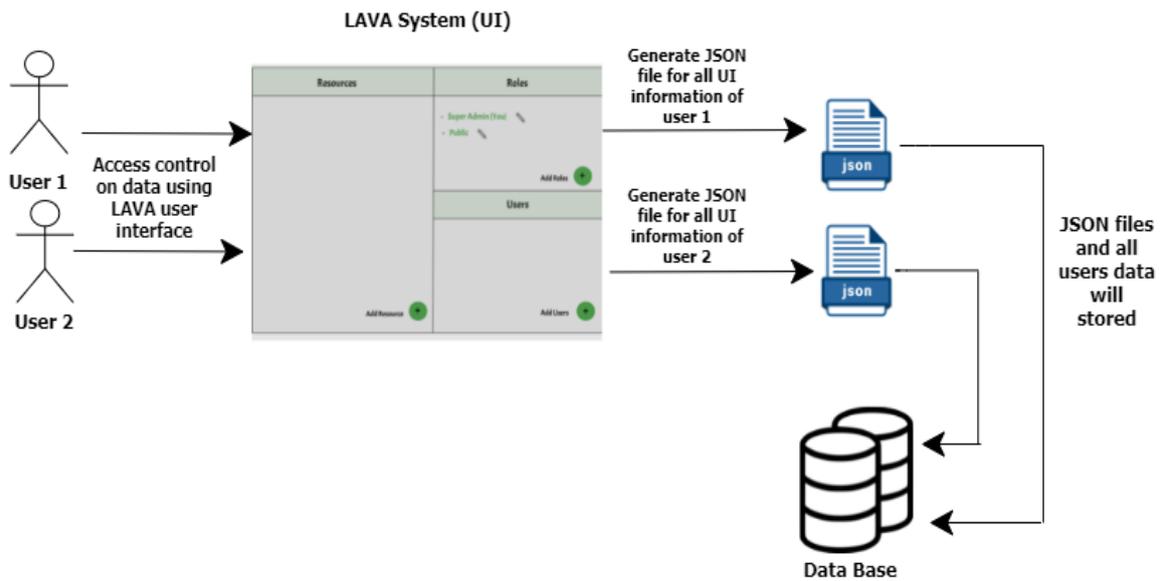


Figure 4.1: UI system description

The system's main features are:

- All RBAC logic is maintained in JSON data format. The data can be saved in a SQL database, a no-SQL database, or even as JSON text files on the server. The JSON format is friendly and easy to build and structure.
- All RBAC logic is maintained outside the OSN applications code. This enables a clear separation between RBAC and application code. RBAC can be modified without the need to access the application code at all. Decoupling RBAC roles and permissions from resources so they can be re-used.
- Fine-grained access control to any record or resource is provided through the use of conditions. Conditions are constructed by the end-users and then applied to RBAC. If another user is granted a role with a condition, he/she cannot access the resource unless the condition is met. The condition can be based on the resource itself or the user who is trying to access the resource. For example, if the user is a friend of the resource owner, then a role with appropriate permission can be assigned to the user automatically.

- Inheritance of roles is an important feature. It helps to manage the RBAC easily and offers flexibility in managing inheritance where multiple inheritances are supported. However, this may cause a conflict of access when inheriting from two different roles that have contradictory access to the same resource. This deal with conflicts based on the priority of orders and based on context. If role C has permission to access resource R based on two permissions that are inherited from Role B and A, and if role C inherits from B after A, then the permission on B will override the permission on A.

RBAC can also be assigned temporarily or altered automatically by the actions taken. For example, it is possible to add limits to policies e.g., how many times to read a resource, or to revoke a policy once it has been used.

4.2. General System Design

Each component has its own local state, and it exists within the UI hierarchy. The main component is the root of this hierarchy, with each level of sub-components being able to communicate only with their parents and direct children only. However, all sub-components, regardless of their level, can access the store to read and write to it. Figure 4.2 below shows the general system design.

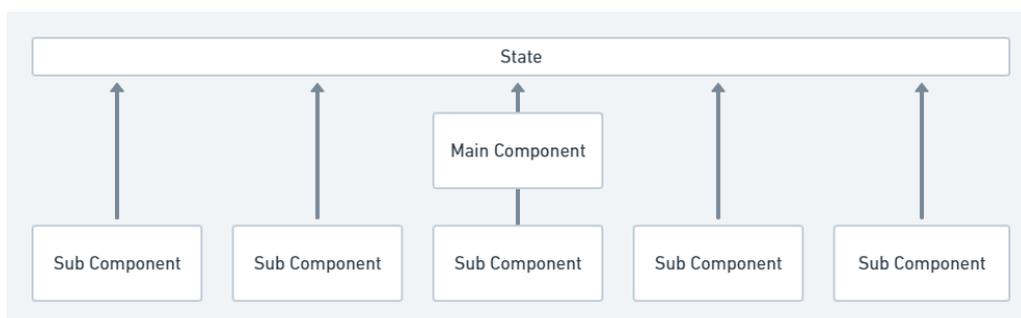


Figure 4.2: General System Design.

4.3. Main Sub-systems



Figure 4.3: Main Sub-System

There are three main sub-systems, the UI, the state management, and the API sub-systems, as illustrated in figure 4.3.

4.3.1 UI Main Components

The first one is the UI system, which is responsible for rendering the actual elements on the screen, handling responsiveness, laying out the elements and handling input and output. It is composed of subcomponents, with each one representing either a parent element that has some children components inside or a leaf component that doesn't render any further sub-components.

There are two types of UI components. The first one is the components that render some UI on screen, the other type components that act as containers for other UI and only render the child components.

4.3.2 State Management

The state management sub-system is responsible for maintaining and tracking the overall system state. It contains:

1. **Store:** which contains the current state, and also listens for any dispatched actions to update the state
2. **Reducers:** are pure functions that are used to mutate a slice of the state. Each reducer is responsible for one slice.

3. **Actions:** these are the types of actions the UI uses to update the state
4. **Actions creators:** are the functions that create the actions which are then can be dispatched to trigger an update to the state
5. **Thunks:** these are the functions that are used to communicate with API requests that can affect the state. Thunks are responsible for dealing with any asynchronous code.

4.3.3 JSON file Structure.

The last step of the sub-system is to generate a JSON file which contains three main parts:

1. System property.
2. Resources.
3. Access control.

The JSON file and its own data will be all stored in the database. After that, the file will be sent to Etqan back-end system using a POST request.

The JSON file data contain all data that can be extracted from the visual representation of the UI that gives users the ability to set permissions for every resource or role they add. These permissions will be added from a GUI that is interactive and easy to use. The main UI features represent:

1. Resources.
2. Roles.
3. Users.

This data will be parsed as JSON objects where each item contains its properties and some fields with their corresponding default values. The three main UI components have all required data which are stored in them, such as a resource name, and its fields. User and Role components have all the access control data that is needed to manage the access control, such as available usernames, their passwords, available roles, the permissions for each role on

every resource and for every possible action (READ, CREATE, UPDATE and DELETE) and any other conditions that are set in the access control policy, such as limits of access or expiration date.

Figure 4.4 shows the JSON file structure, that will be described by example in the next section.

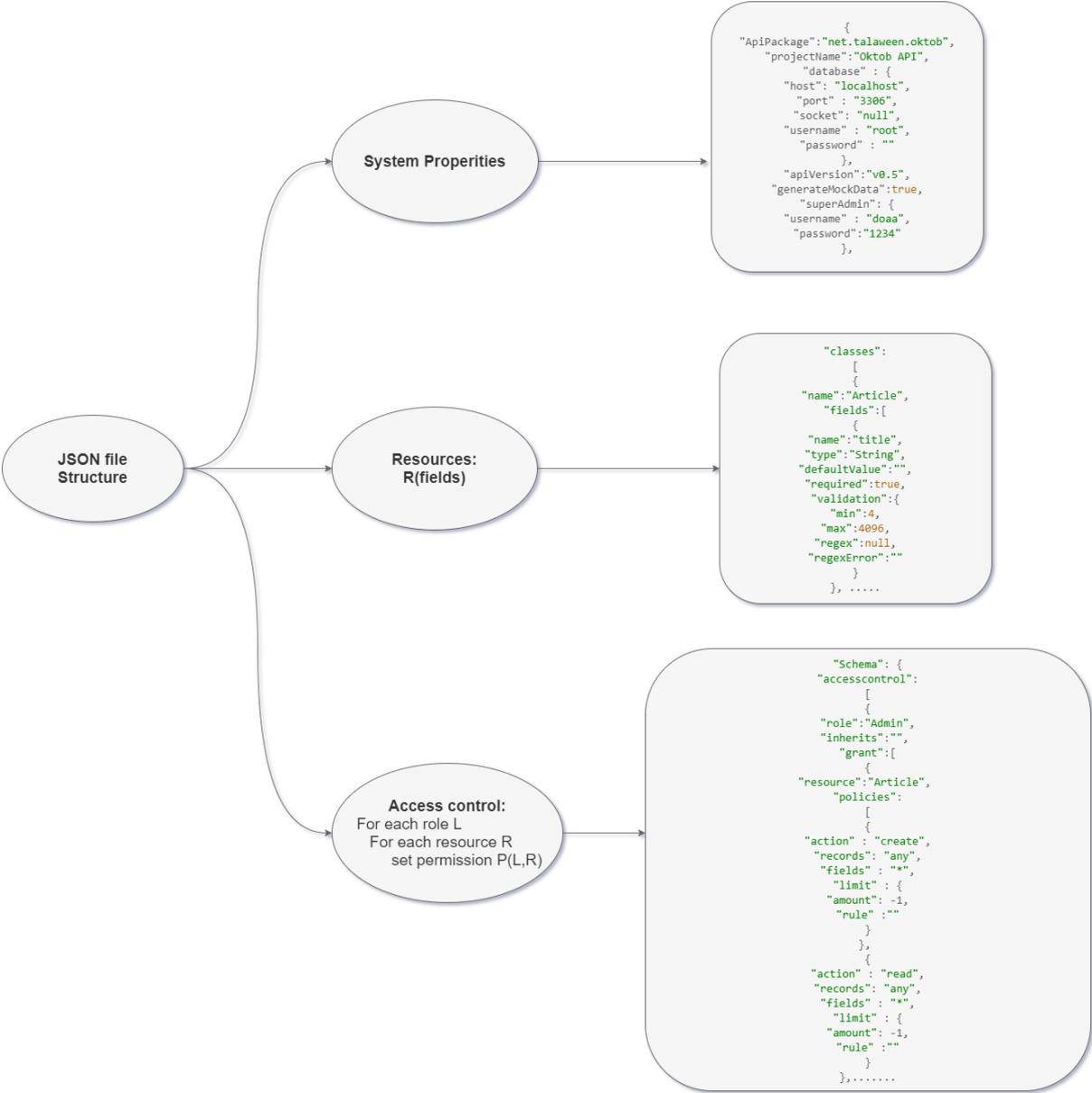


Figure 4.4: JSON file Structure.

4.3.1.1 Use-Case Example

Figure 4.5 below shows a use-case example of UI components for a suggested Blog system, which was called OKTOB (it is a simple blogging system that contains articles, comments, users and roles resources).

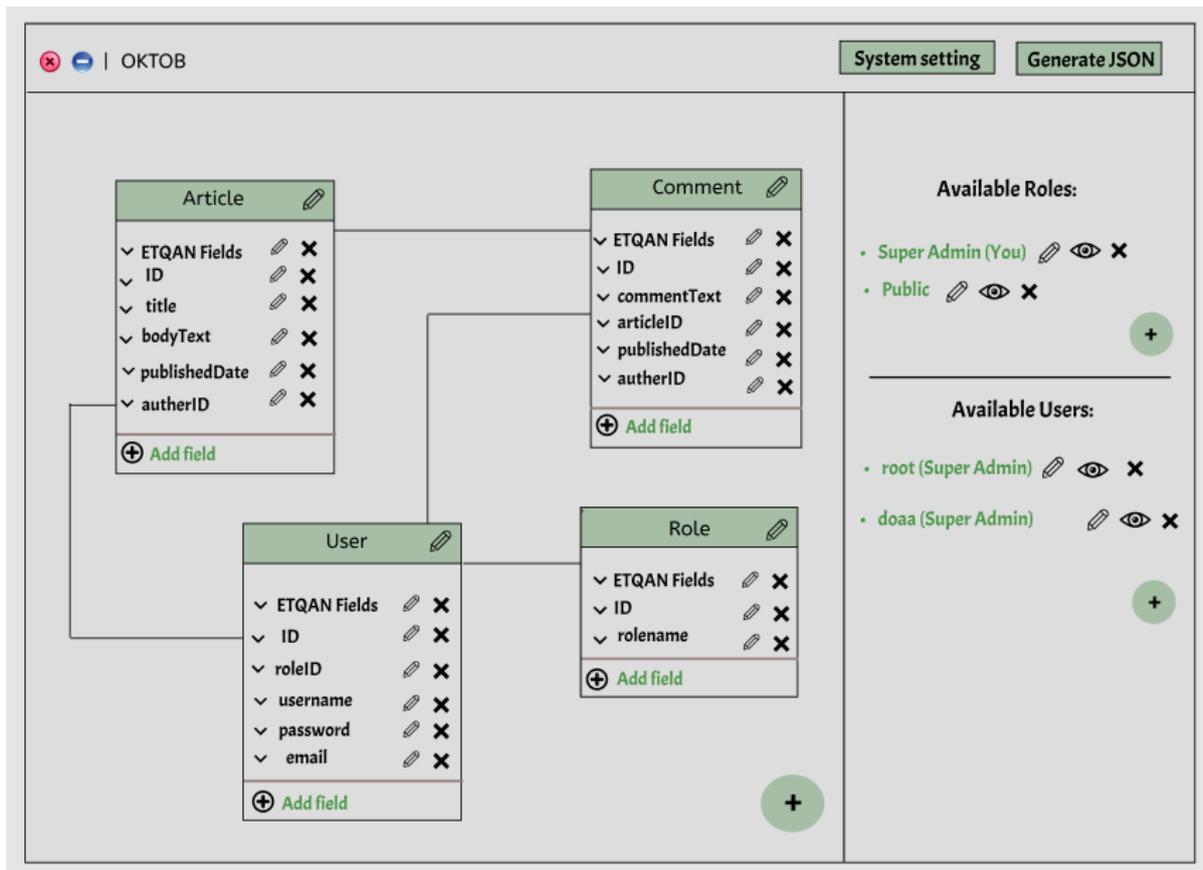


Figure 4.5: OKTOB UI Components

The figure shows four classes which are: Article, Comment, User and Role. By default two resources (Role and User) will be created, while the other two resources are created by the user. In the figure we can see that each class has fields. There are fields that are also created by default by Etqan system, while the others are added by the user. Etqan also creates two role instances by default, Super-Admin and public-guest. The user who creates the system will be assigned the super-Admin role. Then the user can create as many roles as

possible, assign permissions to them, then he can create users or allow the users to register so they can be assigned a role.

Once the Generate JSON button is clicked, the UI is converted into JSON data and will be sent to Etqan backend-system to generate the corresponding API.

The JSON file will be generated inside the UI using the command:

```
const jsonData = json.stringify(state);
```

Where the state is the actual state of the system which is built according to the required schema.

The JSON data consists of three main parts:

1. A system property, which is shown below, shows the various properties of the system.

The user can configure these from the UI using the system settings button.

```
"ApiPackage": "net.talaween.oktob",
  "projectName": "Oktob API",
  "database" : {
    "host": "localhost",
    "port" : "3306",
    "socket": "null",
    "username" : "root",
    "password" : ""
  },
  "apiVersion": "v0.5",
  "generateMockData": true,
  "superAdmin": {
    "username" : "doaa",
    "password": "1234"
  },
```

2. Resources are an array of objects. Each object represents one resource; each resource contains its own properties and an array of fields. The following code shows the code generated from the above example for the resources array:

```

"classes":
[
  {
    "name":"Article",
    "fields":[
      {
        "name":"title",
        "type":"String",
        "defaultValue":"",
        "required":true,
        "validation":{
          "min":4,
          "max":4096,
          "regex":null,
          "regexError":""
        }
      },
      {
        "name":"bodyText",
        "type":"Text",
        "defaultValue":null,
        "required":true,
        "validation":{
          "min":null,
          "max":null,
          "regex":null,
          "regexError":""
        }
      },
      {
        "name":"publishedDate",
        "type":"Date",
        "defaultValue":null,
        "required":true,
        "validation":{
          "min":null,
          "max":null,
          "regex":null,
          "regexError":""
        }
      },
      {
        "name":"dedicatedTo",
        "type":"User",
        "defaultValue":null,
        "required":false,
        "validation":{
          "min":null,
          "max":null,
          "regex":null,
          "regexError":""
        }
      }
    ]
  },
]

```

3. Access control: this section contains all the information needed to set access control on resources. The generated JSON data for the access control of the above example is shown below:

```
"Schema": {
  "accesscontrol":
  [
    {
      "role": "Admin",
      "inherits": "",
      "grant": [
        {
          "resource": "Article",
          "policies":
          [
            {
              "action" : "create",

              "records": "any",
              "fields" : "*",
              "limit" : {

                "amount": -1,

                "rule" : ""
              }
            },
            {
              "action" : "read",

              "records": "any",
              "fields" : "*",
              "limit" : {

                "amount": -1,

                "rule" : ""
              }
            }
          ]
        }
      ]
    }
  ],
}
```

4.4 Pilot UI:

To run the overall system we will implement a pilot interface that enables us to identify all the functionalities of the high-fidelity interface. The reason for that is because implementing the high fidelity UI that we have designed in chapter 3 needs experts in React programming and

UI developers and graphic designers. However, this is out of the scope of our thesis as it requires a lot of time.

The implemented UI supports the ability to drag and drop tables around the screen. This helps the user to arrange the classes for an easy access. The drag and drop feature was developed using a React library called React-Druggable. The drag and drop system update the position of a table. If this table has a relationship with another table, then a line connecting both tables is drawn on the screen. This line will need to be redrawn in an appropriate manner. A component was created to handle all cases of redrawing the line properly. Figure 4.6 below shows the main interface of LAVA Pilot UI that contains three main components (Resources, Roles and Users) and the default roles which are: (Super Admin and Public User).

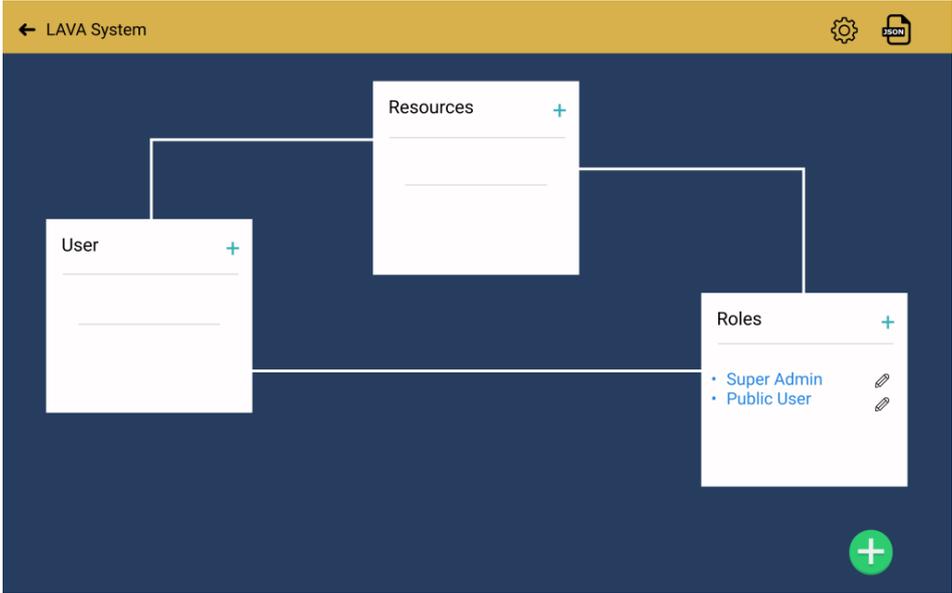


Figure 4.6: Main Pilot Interface

When the user clicks on (+) inside the resource table, it leads him to create a new resource as shown in figure 4.7 which asks the user to set a name for the resource and add new fields as he wants.

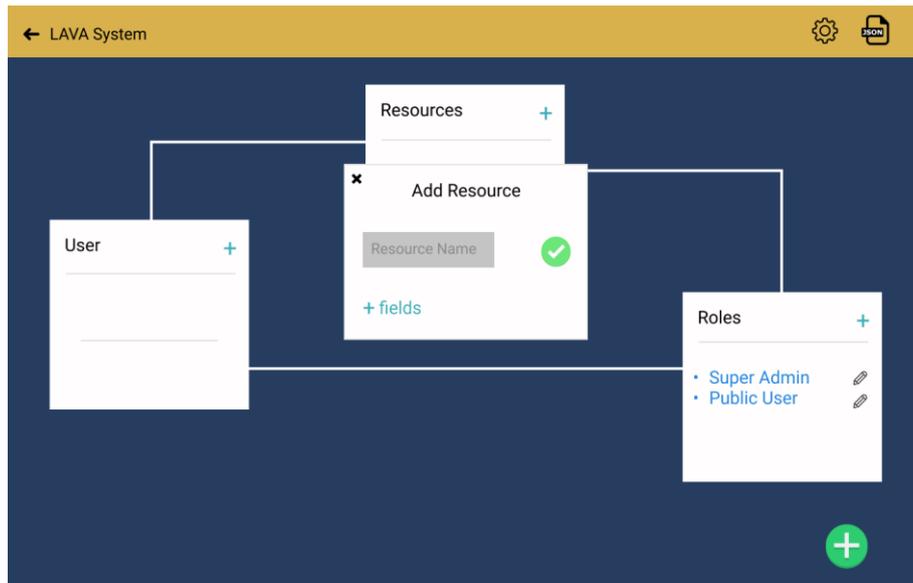


Figure 4.7: Add Resource Pilot Interface.

Figure 4.8 shows when the user adds a new resource, the resource will be shown in the main interface and the set permissions icon will also be shown beside the resource name.

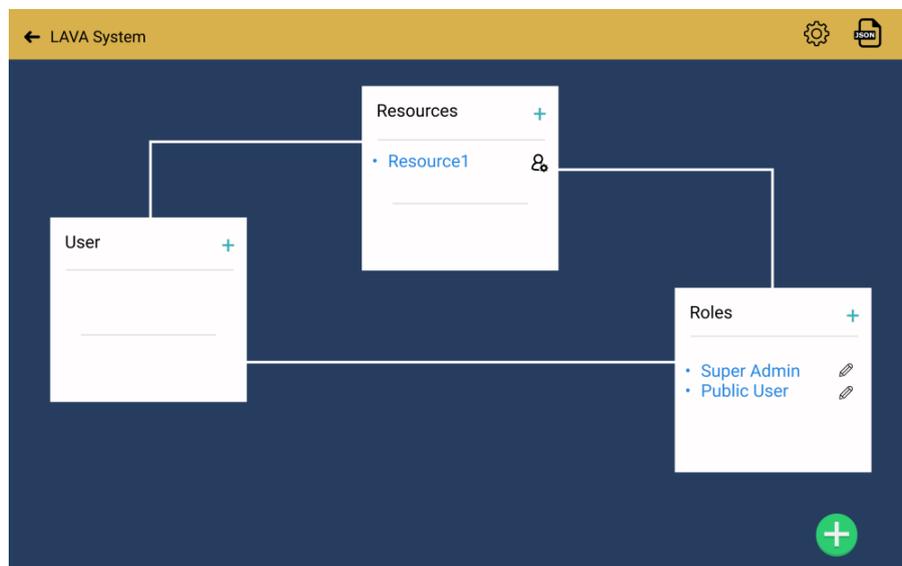


Figure 4.8: Resource Added Pitot Interface.

Figure 4.9 below describes the set permission interface that will open when the user clicks on the set permission icon for the resource. It asks the user to select the role and fields, and then sets permission for every field inside the resource.

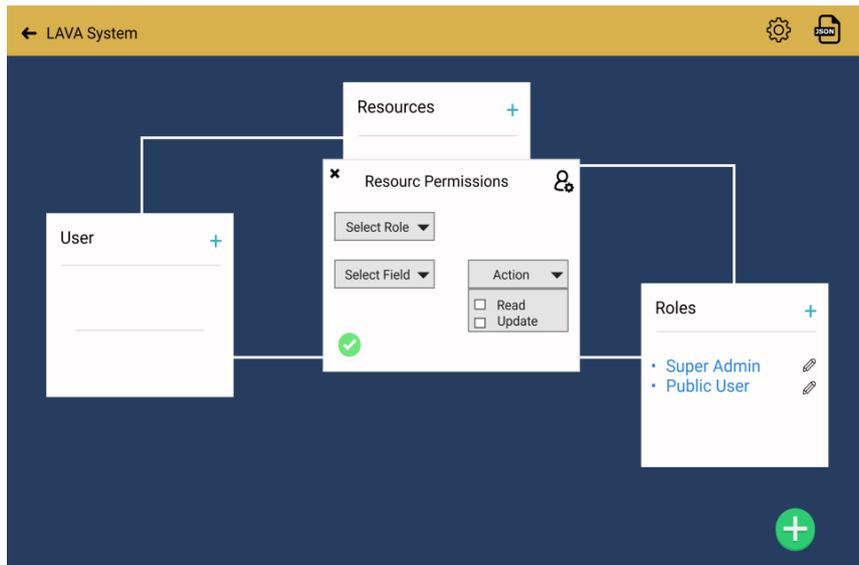


Figure 4.9: Set Permission Pilote Interface.

Figure 4.10 below shows OKTOB use-case example pilot interface that contains four resources.

- Article.
- Comment.
- Roles contain the default main roles (Super Admin, Public User).
- Users contain the users that are added from Super Admin.

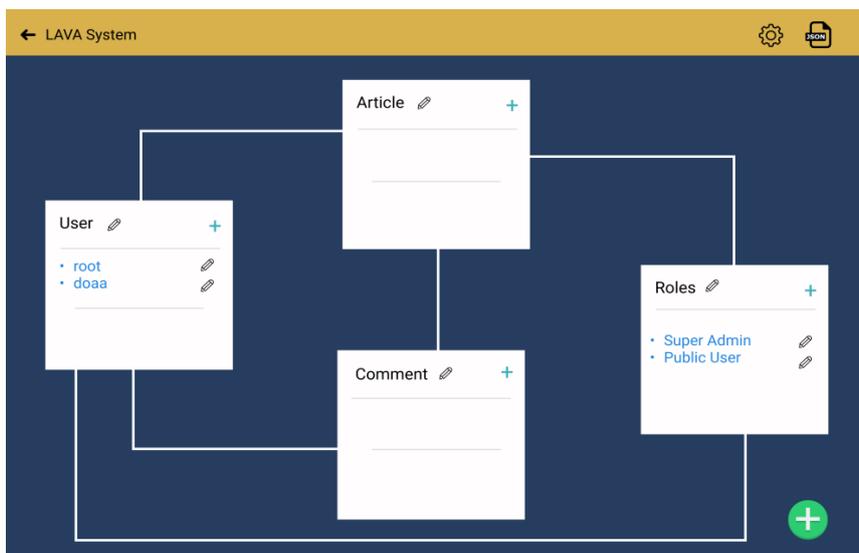


Figure 4.10 : OKTOB Pilot Interface

4.5 Events Handling

Event handling is critical because it ensures that the UI system keeps track of all actions taken by the user. Event handlers detect and process actions such as resizing a window, clicking a link, or scrolling through a website.

Declaring event handlers is a unique technique for managing events in React. The fact that event handling in React components is declarative is what sets it apart. In contrast, with jQuery, you must write imperative code that selects the appropriate Document Object Model (DOM) elements and assigns event handler methods to them. The declarative approach to event handlers has the advantage of being part of the UI structure.

The UI system used the standard React event-handling structure as illustrated in figure 4.11:

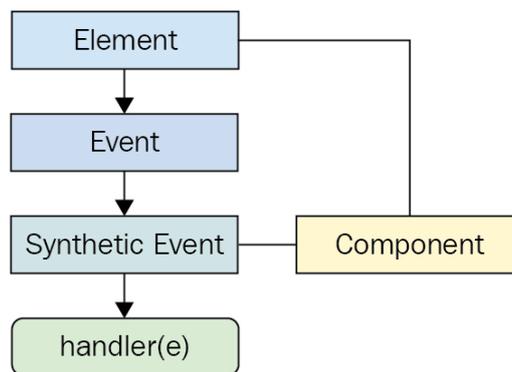


Figure 4.11: React event-handling structure.

When attaching an event handler function to a DOM element using the native `addEventListener()` function, the callback will get an event argument passed to it. Event handler functions in React are also passed an event argument, but it's not the standard Event instance. It's called `SyntheticEvent`, and it's a simple wrapper for native event instances.

Synthetic events serve two purposes in React:

1. It Provides a consistent event interface, normalizing browser inconsistencies
2. Synthetic events contain information that's necessary for propagation to work

Each component is responsible for handling the events that occur.

Chapter 5: System Usability Experiment

In this chapter, we are discussing the usability tests for the final UI of LAVA system.

5.1. Objectives

One of the main goals of the usability testing is to see how well new users understand the content arrangement and navigation on the system interface, how quickly they pick up the structure, and how efficient their performance is.

The following questions were used to define the objectives, as well as to develop and build the test cases.

- How user-friendly is the UI? Is it simple for users to locate specific information?
- Do users have a sense of command over the system? What kind of power do they think they'll have?
- Is it possible for users to foresee the outcomes of their actions?
- Is the engagement with the system fun and satisfying?

5.1.1. Designing a test framework

Usability testing was chosen as the main study approach because it is based on direct user engagement and allows researchers to monitor participants' behaviour as they complete tasks (Bruun et al., 2009). By establishing comparisons and assessments of the results, such a scenario-based method aids data analysis.

The method provides several advantages over other types of analysis. Because it is usually carried out by usability specialists and fictional consumers, it reflects less real-life circumstances. It is simpler to delve into the core of usability difficulties and watch users who are struggling during their contact with the system when actual users are involved.

In lab-testing is used as the main method of usability evaluation. Each testing session took approximately 60 minutes, and it consists of three main parts:

1. Preparation for the test which included informing the participants of the test objective, and the tasks they need to perform
2. The actual conducting of the tasks
3. Feedback session and open-ended questions where each user is given the opportunity to evaluate his/her performance and reflects on the tasks.

The participants' verbal consent was obtained at the beginning of each session, and the procedure was audio and video recorded. The researcher documented and archived both the screen and the participants' responses until the experiments were finished.

In addition, the feedback was manually recorded using pen and paper; however, because there were no observers to keep track of the notes, the moderator focused more on the test session than scripting. Unnecessary delays and miscommunication between the moderator and the participants are avoided using this technique. Furthermore, it lowers the chances of missing a crucial detail.

5.2. Methodology

A test was conducted by asking each participant to solve 8 tasks

The participants completed the activities while the researcher assessed their performance during the tests. Because the tests were completed in a short amount of time, it was necessary to videotape the sessions to allow for additional study and analysis. Subjective and objective metrics were used to acquire quantifiable data. Questionnaires were used to collect the subjective data. After each task, the participants were given questionnaires to complete. The participants used a Likert-scale based questionnaire to rate how simple or difficult it was to

complete a job. The goal was to see which tasks the participants found difficult to complete. More details about the questionnaire in Appendix B.

During the tests, a variety of objective performance data was collected. The effectiveness of the participant was the first performance metric. The effectiveness was determined by watching whether the participants were able to complete the activity. The goal of this test was to find out which tasks the participants struggled with. The time spent on each activity by the participants was calculated from the time they were given the job until they solved it, gave up, or spent more than five minutes on it. The goal of this measurement was to find out which tasks took up the most time among the participants.

Observations were used to get qualitative data. During the experiments, if a participant had any issues, they were added to a list. This list included all the issues that were discovered during all the tests, as well as the number of times they occurred.

Studies suggest keeping track of how frustrated the participants are during the tests. A researcher can determine which tasks caused frustration. However, acceptable for this research. While some of the participants expressed dissatisfaction during the exams, others did not, even while they were having difficulty completing the same activities. This could happen because some of the participants did not to reflect any negative emotions during the test.

5.3. Participants

This study was conducted with 24 participants. The participants needed to have a list of requirements to match the targeted users' profiles. So that their answers can be reliable and relevant to the research.

There were two required groups of users, each with varying levels of experience. The novice users are represented by one group, while the experienced users are represented by the other. Each group included 12 participants. This group size was selected as a prior study (Fu et al., 2002) found that 12 participants in a usability test could reveal 98 percent of usability issues. It was also critical to have an equal number of male and female volunteers to avoid skewed results.

The following table shows the participants and their basic information:

Table5.1: Participants Information for System Usability.

Participant code	Age	Gender	Occupation
Group 1			
P1G1	18	Female	History Student
P2G1	18	Female	Arabic Student
P3G1	20	Female	Accountant
P4G1	23	Male	Nurse
P5G1	27	Male	Lawyer
P6G1	25	Female	House wife
P7G1	39	Male	English Teacher
P8G1	21	Male	Secretary
P9G1	19	Male	Biology Student
P10G1	39	Male	Science Teacher
P11G1	28	Female	Dentist
P12G1	45	Female	House wife
Group 2			
P1G2	26	Male	IT Teacher
P2G2	22	Female	Programmer
P3G2	35	Male	Computer Engineer
P4G2	18	Male	Computer science student
P5G2	20	Male	Graphic designer
P6G2	20	Female	Multimedia Student
P7G2	19	Female	Multimedia Student
P8G2	24	Male	IT Teacher
P9G2	19	Male	Web designer
P10G2	33	Female	Web developer
P11G2	39	Female	Programmer
P12G2	27	Female	Graphic designer

5.4. Conducting Experiments

The tests took place in a normal setting in an ordinary room, with a PC connected to the internet, the PC can run the UI without any issue or lag.

Participants were shown a small demo video of a Think Aloud test session at the start of the session, as recommended by Nielsen (2014) (Nielsen, n.d.), to gain a better idea of what was expected of them. The testers were reassured that the goal of the session was not to test the participants and that nothing could go wrong.

All participants had to complete a list of tasks as listed in table 5.2.

Table 5.2: Tasks for Usability Test.

Task Code	Task Description
T1	Add new resource
T2	Add a new role
T3	Add a permission to update the resource
T4	Add a second resource
T5	Add a relationship between the two resources
T6	Add a new user
T7	Assign the existing role to the user
T8	Create a second role

The time they spent on each task is monitored, each participant moves to the next task if he/she completed the current one successfully, he/she spent more than 3 minutes on it, something went wrong, or he/she gave up. When a participant gave up on a task, she/he is added to a list and then he/she would be asked about it in the post-test interview session.

5.5. Results

Data from video sessions, interviews, and researcher notes were employed to find out usability issues. The evaluation of effectiveness was the initial step. A simple descriptive analysis was used to determine how many people failed the tasks. The goal was to determine which tasks the participants had the most difficulty completing. The analysis was carried out by counting the number of those who did not complete the same activities.

The following table shows the effectiveness results of group (1) the novice group and how many participants completed or failed each task:

Table 5.3: Effectiveness Result for Group1.

Task Code	No. Attempted Task	No. Completed	Average completion time (seconds)	No. Failed
T1	12	7	108	5
T2	12	9	100	3
T3	12	7	101	5
T4	12	11	60	1
T5	12	8	122	4
T6	12	10	89	2
T7	12	8	102	4
T8	12	12	58	0

The following table shows the effectiveness results of group (2) the expert group and how many participants completed or failed each task:

Table 5.4: Effectiveness Result for Group2.

Task Code	No. Attempted Task	No. Completed	Average completion time (seconds)	No. Failed
T1	12	12	58	0
T2	12	12	54	0
T3	12	11	60	1
T4	12	12	45	0
T5	12	10	78	2
T6	12	12	47	0
T7	12	12	77	0
T8	12	12	45	0

The efficiency was the subject of the second step of the analyzing process. Descriptive statistics were also used in this step. The mean task time for each group and task was calculated for the analysis. For the efficiency analysis, the participants' performance was judged to be good if they spent less than 180 seconds on a task. The assignment required too much time if they spent more than 180 seconds on it.

Table 5.5 Efficiency Rating.

Time Spent	Rating
0-60	Good
60-120	Acceptable
120+	Bad

Table 5.6: Efficiency Result for Group1.

Task Code	Average Time Spent by all participants (sec)	Average Rating
Group 1		
T1	108	Acceptable
T2	100	Acceptable
T3	101	Acceptable
T4	60	Good
T5	122	Bad
T6	89	Acceptable
T7	102	Acceptable
T8	58	Good

Table 5.7: Efficiency Result for Group2.

Task Code	Average Time Spent by all participants (sec)	Average Rating
Group 2		
T1	58	Good
T2	54	Good
T3	60	Good
T4	45	Good
T5	78	Acceptable
T6	47	Good
T7	77	Acceptable
T8	45	Good

5.6. Interviews

An interview session was conducted for each group after finishing their tasks. The questions aimed to evaluate each user experience using the system and take their feedbacks, and the questions also aim to understand the reasons why some participants failed to complete a task.

The interview included the following questions:

1. On Task x, you have not completed it successfully, what was the reason?
2. Do you understand the purpose of the whole system?
3. How did you find the interactivity with the system?
4. If there is something you suggest be adding or modifying, what would it be?
5. Would you use such a system to protect your own privacy online?

The researcher asked each participant the questions after finishing the session; the researcher used a pen and a paper to take notes of the answers, some of the notable answers regarding these questions from both groups:

Group 1 (Novice Users)

1. Question 1:
 - a. P3G1: I didn't understand the purpose of the task.
 - b. P4G1: the task is not clear to me.
 - c. P5G1: I didn't know from where to start.
 - d. P8G1: the task is too complicated and needs some technical knowledge.
2. Question 2:
 - a. P1G1: the whole idea is clear to me but I don't understand how the system actually works to achieve that.
 - b. P2G1: the idea of the system is something new to me.
 - c. P5G1: the system sounds like something I would never need or use.

- d. P7G1: the system sounds like it is made for technical people or maybe it is past my generation.
3. Question 3:
- a. P1G1: interactivity was clear and I felt the system will do something important for me.
 - b. P3G1: the system is interactive, but I could not understand the correct steps to make it respond correctly.
4. Question 4:
- a. P1G1: a tutorial.
 - b. P3G1: a video recording to explain how to do things.
 - c. P5G1: colours and font size.
 - d. P7G1: I didn't understand the system well so I cannot give an answer.
5. Question 5:
- a. P1G1: if this system would protect my privacy, I would use it.
 - b. P3G1: I would use it.
 - c. P7G1: I am not sure as I don't understand.

Group 2 (Expert Users)

1. Question 1:
- a. P2G2: I was confused about what I should do, and I was not sure where to go next.
 - b. P5G2: I think the interface did not help me find my way to accomplish the task.
2. Question 2:
- a. P3G2: the system idea is clear, and the interface helps you to know what you should do.

- b. P4G2: the system sounds familiar, and I can relate the tasks and how to perform them, but I think other non-technical people may have a problem using or understanding it.
 - c. P7G2: the general idea of the system is clear, but I think the system may be a complex system that won't be able to be used by normal users
 - d. P8G2: the system idea is great, and I think the interface is clear and easy to navigate, the interface was clearly designed
3. Question 3:
- a. P2G2: the interaction was well designed, and easy to follow
 - b. P3G2: I think you should provide more feedback to the user and give them a confirmation of each step results
4. Question 4:
- a. P2G2: a wizard to help novice users
 - b. P3G2: add a template where users don't need to always starts from scratch
 - c. P9G2: give more animation and visual effects to add a more appealing look to the application
5. Question 5:
- a. P2G2: I will use the application and evaluate it to see its actual benefits
 - b. P5G2: I will use it and recommend it to the others
 - c. P7G2: I will use the application frequently if it exists as a fully featured application.

5.7. Discussion

We note that the novice group has made more mistakes and less success rate (75% of total tasks), the expert group has made less mistakes (96.8% of total tasks).

The average rating of the novice group is (Acceptable Average) while the expert group is (Good Average).

The novice user has made the most mistakes on task (1) and task (3) which both related to the concept of a resource. This may be due to the fact the novice users may not clearly identify what a resource is. Therefore the system should provide more information and clue on what makes a resource and when to create a new one. This brings to attention a note we recorded on initial usability experiments with low-fidelity prototypes, where users sometimes had confusions between the concept of a resource and a field, and their relationships.

For the expert group, the least efficient two tasks were task (5) adding a relationship between two resources and task (7) assigning an existing role to a user. It is clear that the expert participants paid more attention on creating a relationship, while novice groups constructed the relationship quickly. The reason for that the expert group has more understanding about the concept of relationships between entities and how they affect each other's, so they give more attention on creating the relationship. The same with assigning a role to a user, they realize that this is a critical task so they give it more time than novice users. The UI should help the user create the relationship, and show the user a visual feedback about the effect of adding a new relationship. Warning messages or visual highlights with red or amber colors can be used to reflect the results of a specific action.

The interview questions show that there is a need to give more explanations and tutorials about the system, the system may not be able to reflect its purpose to novice users, therefore, an interactive tutorial or a wizard should help get them more into using the application correctly.

The expert group participants overall were satisfied with the system UI and interaction design, they were keener about novice users.

Both groups' participants indicated they would use the system on regular basis and learn more about it.

Chapter 6: Conclusions & Future works

6.1. Conclusions

Our proposed system enables the users to act as super admins over their own data, where the user is the person who can design and plan any access control policy. This will improve user privacy and security as the users are the only person who can grant access to any other user to access the data. The research focused on aware users about role-based access control using an online survey. User-centered design was used to understand user's requirements and let them be a participant in designing the UI. The system UI that designed is flexible, allowing fine-grain role-based access control to resources, in which every permission of the role set to the resource will be converted into JSON file that will be sent to ETQAN back-end system for an API request to allow social networks to use it. The experiments with users indicated that there will be a need to normalize the concept of access control to novice users through a series of interactive tutorials and visual aids.

For future works, the system UI needs more improvements and work to make it usable and ready to be used by normal users. The UI can be used by both novice and expert users and it can still be improved on the details to make it suitable for novice users

6.2. Future Works

The research has presented a high-fidelity prototype that was tested and proofed to be efficient and easy to use. However, the actual implementation of the system was merely a proof of concept and still needs further works to make the system at ready for use by a wider audience. The following are some detailed things which are recommended to be done in future works:

- More experiments are supposed to be conducted in order to apply all the feedbacks from the participants and measure the effect of that.
- Implement the high-fidelity system with all functionalities.
- The UI should always validate the username field of the user class and always make sure it is unique.
- The UI should not accept reserved words as field names.
- The UI strip special characters from project name and from field's names.
- More settings from the JSON file should be reflected in the UI such as languages, pagination size, etc.
- The UI should make sure to prevent fields with names identical to User class predefined fields.
- The UI should have a full validation system that supports flexible evaluation for dates.
- There is a need to add a web Admin Control panel to administer past and current projects and allow the user to edit settings for existing systems.
- The UI should allow users to add custom error and success messages.
- The UI should provide interactive tutorials and visual aids to the users and keep suggestions recommendations about current settings in order to help the users use the system efficiently

References:

- A. K. R. Praveena, B. S. D. Bhavani, C. K. S. B. (2013). *Online Social Networks for a Multiparty Access Control Model and Mechanisms*. 69–77.,
- Abras, C., Maloney-Krchmar, D., & Preece, J. (2004). *User-centered design*.
- Alturi, V., & Ferraiolo, D. (2011). Role-Based Access Control. In *Encyclopedia of Cryptography and Security* (Vol. 46). https://doi.org/10.1007/978-1-4419-5906-5_829
- Antonio Tapiador, Diego Carrera, J. S. (2012). *Tie-RBAC: An application of RBAC to Social Networks*.
- Anwar, M., Fong, P. W. L., Yang, X. D., & Hamilton, H. (2010). Visualizing privacy implications of access control policies in social network systems. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5939 LNCS, 106–120. https://doi.org/10.1007/978-3-642-11207-2_9
- Awad, M., Al-karaki, R., & Idais, D. (2021). How to protect the privacy and allow data sharing easily in online social networks. *ACM International Conference Proceeding Series*, 623–627. <https://doi.org/10.1145/3459104.3459204>
- Awad, M., Al-Karaki, R., & Idais, D. (2021). User as a Super Admin: Giving the End-Users Full Control to Manage Access to Their Data in Social Media Networks. *ACM International Conference Proceeding Series*, 623–627. <https://doi.org/10.1145/3459104.3459204>
- Ben Shneiderman, Catherine Plaisant, Cohen, M. S., Steven Jacobs, Niklas Elmqvist, & Nicholas Diakopoulos. (2016). *Designing the User Interface: Strategies for Effective Human-Computer Interaction* (6th ed.).
- Boyd, D. M., & Ellison, N. B. (2007). Social network sites: Definition, history, and scholarship. *Journal of Computer-Mediated Communication*, 13(1), 210–230. <https://doi.org/10.1111/j.1083-6101.2007.00393.x>
- Bruun, A., Gull, P., Hofmeister, L., & Stage, J. (2009). Let your users do the testing: A comparison of three remote asynchronous usability testing methods. *Conference on Human Factors in Computing Systems - Proceedings*, 1619–1628. <https://doi.org/10.1145/1518701.1518948>
- Camburn, B., Viswanathan, V., Linsey, J., Anderson, D., Jensen, D., Crawford, R., Otto, K., & Wood, K. (2017). Design prototyping methods: State of the art in strategies, techniques, and guidelines. *Design Science*, 3(Schrage 1993), 1–33. <https://doi.org/10.1017/dsj.2017.10>
- Cheng, Y., Park, J., & Sandhu, R. (2016). An access control model for online social networks using user-to-user relationships. *IEEE Transactions on Dependable and Secure Computing*, 13(4), 424–436. <https://doi.org/10.1109/TDSC.2015.2406705>
- Darejeh, A., & Singh, D. (2013). A review on user interface design principles to increase software usability for users with less computer literacy. *Journal of Computer Science*,

- 9(11), 1443–1450. <https://doi.org/10.3844/jcssp.2013.1443.1450>
- Elverum, C. W., & Welo, T. (2015). On the use of directional and incremental prototyping in the development of high novelty products: Two case studies in the automotive industry. *Journal of Engineering and Technology Management - JET-M*, 38, 71–88. <https://doi.org/10.1016/j.jengtecman.2015.09.003>
- Figma. (2016). <https://www.figma.com/files/recent?fuid=799772725178824156>
- Fu, L., Salvendy, G., & Turley, L. (2002). Effectiveness of user testing and heuristic evaluation as a function of performance classification. *Behaviour and Information Technology*, 21(2), 137–143. <https://doi.org/10.1080/02699050110113688>
- Fuchs, L., Pernul, G., & Sandhu, R. (2011). Roles in information security - A survey and classification of the research area. *Computers and Security*, 30(8), 748–769. <https://doi.org/10.1016/j.cose.2011.08.002>
- Galitz, W. O. (2007). *The essential guide to user interface design: an introduction to GUI design principles and techniques*. John Wiley & Sons.
- Gerber, E., & Carroll, M. (2012). The psychological experience of prototyping. *Design Studies*, 33(1), 64–84. <https://doi.org/10.1016/j.destud.2011.06.005>
- Gupta, A., Pandhi, K., Bindu, P. V., & Thilagam, P. S. (2016). Role and Access Based data Segregator for Security of Big Data. *Procedia Technology*, 24, 1550–1557. <https://doi.org/10.1016/j.protcy.2016.05.130>
- Imran-Daud, M., Sánchez, D., & Viejo, A. (2016). Privacy-driven access control in social networks by means of automatic semantic annotation. *Computer Communications*, 76, 12–25. <https://doi.org/10.1016/j.comcom.2016.01.001>
- Interaction Design Foundation. (n.d.). *User Experience (UX) design*. Retrieved December 29, 2021, from <https://www.interaction-design.org/literature/topics/ux-design>
- Jeffrey K. Liker, & Pereira, R. M. (2018). *Virtual and Physical Prototyping Practices: Finding the Right Fidelity Starts With Understanding the Product*.
- Jin, L., Chen, Y., Wang, T., Hui, P., & Vasilakos, A. V. (2013). Understanding user behavior in online social networks: A survey. *IEEE Communications Magazine*, 51(9), 144–150. <https://doi.org/10.1109/MCOM.2013.6588663>
- Jonathan Lazar, Jinjuan Feng, & Harry Hochheiser. (2017). *Research Methods in Human-Computer Interaction* (2nd ed.).
- Kaya, E., Alacam, S., Findik, Y., & Balcisoy, S. (2018). Low-fidelity prototyping with simple collaborative tabletop computer-aided design systems. *Computers and Graphics (Pergamon)*, 70, 307–315. <https://doi.org/10.1016/j.cag.2017.07.026>
- Kolomeets, M., Chechulin, A., Kotenko, I., & Saenko, I. (2019). Access Control Visualization Using Triangular Matrices. *Proceedings - 27th Euromicro International Conference on Parallel, Distributed and Network-Based Processing, PDP 2019*, 348–355. <https://doi.org/10.1109/EMPDP.2019.8671578>
- Liker, J. K., & Pereira, R. M. (2018). Virtual and physical prototyping practices: finding the

- right fidelity starts with understanding the product. *IEEE Engineering Management Review*, 46(4), 71–85.
- Liu, Y., Osvalder, A.-L., & Karlsson, M. (2010). Considering the Importance of User Profiles in Interface Design. *User Interfaces*, May. <https://doi.org/10.5772/8903>
- Mansour, E., Sambra, A. V., Hawke, S., Zereba, M., Capadisli, S., Ghanem, A., Aboulmaga, A., & Berners-Lee, T. (2016). *A Demonstration of the Solid Platform for Social Web Applications*. 223–226. <https://doi.org/10.1145/2872518.2890529>
- Min, X., Zhang, W., Sun, S., Zhao, N., Tang, S., & Zhuang, Y. (2019). VPMoel: High-Fidelity Product Simulation in a Virtual-Physical Environment. *IEEE Transactions on Visualization and Computer Graphics*, 25(11), 3083–3093. <https://doi.org/10.1109/TVCG.2019.2932276>
- Nam, S., Ko, G., Suh, K., & Kwon, J. (n.d.). User Experience- and Design- R oriented Virtual Product Prototyping System. *2019 11th International Conference on Knowledge and Smart Technology (KST)*, 116–120.
- Nielsen, J. (n.d.). *Demonstrate Thinking Aloud by Showing Users a Video*. Nielsen Norman Group. <https://www.nngroup.com/articles/thinking-aloud-demo-video/>
- Pang, J., & Zhang, Y. (2015). A new access control scheme for Facebook-style social networks. *Computers and Security*, 54, 44–59. <https://doi.org/10.1016/j.cose.2015.04.013>
- Patrício, L., Fisk, R. P., & Falcão E Cunha, J. (2008). Designing multi-interface service experiences: The service experience blueprint. *Journal of Service Research*, 10(4), 318–334. <https://doi.org/10.1177/1094670508314264>
- Pyla, R. H. and P. (2012). *Chapter 20: Prototyping. The UX Book (Second Edition)*.
- Sachan, A., & Emmanuel, S. (2011). Efficient Access Control in Multimedia Social Networks. *Social Media Modeling and Computing*. <https://doi.org/10.1007/978-0-85729-436-4>
- Shillair, R., Cotten, S. R., Tsai, H. Y. S., Alhabash, S., Larose, R., & Rifon, N. J. (2015). Online safety begins with you and me: Convincing Internet users to protect themselves. *Computers in Human Behavior*, 48, 199–207. <https://doi.org/10.1016/j.chb.2015.01.046>
- STOICA, M., MIRCEA, M., & GHILIC-MICU, B. (2013). Software Development: Agile vs. Traditional. *Informatica Economica*, 17(4/2013), 64–76. <https://doi.org/10.12948/issn14531305/17.4.2013.06>
- Thomas C Reeves, & Patricia M Reeves. (2012). *University Teaching in Focus* (1st ed.). <https://doi.org/https://doi.org/10.4324/9780203079690>
- Ulrich, K. T. (2003). *Product design and development*. Tata McGraw-Hill Education.
- Wardhana, S., Sabariah, M. K., Effendy, V., & Kusumo, D. S. (2017). User interface design model for parental control application on mobile smartphone using user centered design method. *2017 5th International Conference on Information and Communication Technology, ICoICT 2017, 0(c)*. <https://doi.org/10.1109/ICoICT.2017.8074715>

- Wattanagul, N., & Limpiyakorn, Y. (2016). Automated documentation for rapid prototyping. *ICIMSA 2016 - 2016 3rd International Conference on Industrial Engineering, Management Science and Applications*, 6–9. <https://doi.org/10.1109/ICIMSA.2016.7503998>
- Yang, M. C. (2005). A study of prototypes, design activity, and design outcome. *Design Studies*, 26(6), 649–669. <https://doi.org/10.1016/j.destud.2005.04.005>
- Zhou, X., & Rau, P. L. P. (2019). Determining fidelity of mixed prototypes: Effect of media and physical interaction. *Applied Ergonomics*, 80(December 2018), 111–118. <https://doi.org/10.1016/j.apergo.2019.05.007>
- Zink, L., Hostetter, R., Bohmer, A. I., Lindemann, U., & Knoll, A. (2018). The use of prototypes within agile product development explorative case study of a Makeathon. *2017 International Conference on Engineering, Technology and Innovation: Engineering, Technology and Innovation Management Beyond 2020: New Challenges, New Approaches, ICE/ITMC 2017 - Proceedings, 2018-Janua*, 68–77. <https://doi.org/10.1109/ICE.2017.8279871>

Appendix A: Survey questions and results.

This survey aims to find out the awareness of online social networks users, and the privacy setting that available in these sites and the possibility of developing interfaces that enable users to define privacy on their resources within social networks.

Note that the information received will be treated confidentially and will only be used for scientific research purposes. Thank you for you.

Researcher: Doaa Idais.

Supervisors: Dr.Khalil Massri, Dr.Mario Macedo.

Part 1: Basic information (Mark one box on your selected answer).

1. Sex:

Question	Answer
Male	
Female	

2. Age:

Question	Answer
Less than 18	
19-28	
29-38	
39-48	
Over 48	

3. Qualification:

Answer	
PhD	
Master's	
Bachelors	
Diploma and less	

Part 2: Identify user's knowledge about data privacy on OSN.

4. What is your goal of using OSN? (You can choose multiple answers).

Question	Answer
Follow the friend's news.	
Follow the news of country.	
Follow friend's activities and share photos with others.	
Professional communication with co-workers.	
Buying and selling.	
Creating new friends.	
Search about friend's information.	

5. Answer by (Yes, No or I don't know).

Question	Yes	No	I don't know
Do you share private information or photos on social network?			
Before you post a photo or a sensitive post, do you set the privacy settings?			

Do you know how to set permissions for people to see your information in OSN?			
Is there any site that you do not feel safe while you post private information and photos on it?			
Do you know who can see your information on OSN?			

Part 3: The ability of users to use new UI to set permissions for their data on OSN.

6. Answer by (Totally disagree, Disagree, Neutral, Agree, Totally agree).

Question	Totally disagree	Disagree	Neutral	Agree	Totally agree
It is necessary to have permissions for access control on OSN data more than the existing permissions.					
It would be better to have standalone system that provides data protection methods.					
The interfaces of privacy settings for data in social networks are easy and understandable.					
It is better to let the user to take a part of designing UI.					

<p>I prefer to set my own permissions on my data without having fixed permissions by social networks.</p>					
<p>I prefer to use more flexible UI than those currently found in social network for access control.</p>					
<p>I have the ability to use new UI for set permissions on my own data on social networks.</p>					

Appendix B: Tasks Testing Questionnaire

1. After completing task x, mark one box that best describes your reaction while doing the task.

		Strongly agree			Strongly disagree	
	Question	1	2	3	4	5
1	I think this task is unnecessary and complex					
2	I think this task is easy to use					
3	I would need technical person support to complete this task					
4	I think there too much inconsistency in this task					
5	I feel very confident using this task					
6	It was easy to find the button and information I needed					
7	I need to learn about role based access control before any task					
8	I would imagine that the most people would learn to use this system very quickly					
9	I think I would like to use the system frequently					

2. Do you have any comments about this task? Yes No
 If yes, please write the comment:

.....
