SCHOOL OF SCIENCE AND ENGINEERING

Capstone Design: CSC4402

DIORR: Real Estate Agency

Web Application

By: Ismail Qarchli

Supervised by:

Dr. Nasser Assem

MAI 5th 2015
Acknowledgement:

Diorr is the name of the project I worked on during the semester of spring 2015, in order to accomplish the requirements of my capstone class at Al Akhawayn University. Before starting this project, I would like to thank Dr. Assem Nasser for his supervision during this whole semester and his support he was offering during his office hours and by email. He gave us a great opportunity to understand, use new tools and discover the world of real estate agencies. As I would like to thank my teammate Sawsane Ouchtal for her persistence, patience, excellent ideas and outstanding contribution to this project. It would not have been the same without their contribution.
CONTENT:

1. Abstract ............................................................................................................. v
2. Introduction ........................................................................................................... 1
3. Methodology ......................................................................................................... 2
4. Main Text
   4.1. Application Requirements .......................................................................... 3
       4.1.1. Functional Requirements ................................................................. 3
       4.1.2. Non-Functional Requirements ......................................................... 5
       4.1.3. Domain Requirement ........................................................................ 6
   4.2. Entity Relationship Diagram ........................................................................ 7
   4.3. Description of the Entities ........................................................................... 8
   4.4. Description of the relationships between the entities .............................. 10
   4.5. Sequence Diagram ..................................................................................... 12
   4.6. Steeple Analysis ......................................................................................... 14
   4.7. System Architecture ................................................................................... 16
   4.8. Technology enablers ................................................................................. 18
   4.9. Database Synchronization .......................................................................... 22
   4.10. Implementation .......................................................................................... 24
         4.11. Implementation results ....................................................................... 28
5. Future Work ......................................................................................................... 32
6. Conclusion ............................................................................................................ 33
7. References ............................................................................................................ 34
8. Appendix ............................................................................................................. 35
1. Abstract:

As the name of the project indicated, Diorr has the main aim of developing a system that is meant to provide an online platform estate agency that will help users to sell, buy, or rent houses. This web application will be as an alternative to the traditional estate agents encountered in some place in the street. In fact, many house occupants find difficulties to find the right, trustworthy, and honest estate agent who will help them to find the perfect client, and vice-versa. Moreover, it is difficult sometimes to find the location of these agents. Therefore, this application will be an intermediary between both clients and house sellers/renters as it is going to be along with both parts during the whole process of house renting, selling, or buying. The project is composed of two parts: a web application as well as a mobile application. This document will be covering all the steps followed, from preliminary requirements to the final steps of implementation and maintenance of the web application.
2. Introduction:

Diorr is a project that I worked on, with my teammate Sawsane Ouchtal, during the spring 2015. Diorr is a web application that will play the role of a real estate agent. In fact, the idea of this project came from the fact that the real estate agent cannot be available every day at any time. The outdated real estate agencies have many disadvantages especially for Moroccans. Therefore, we thought of a new way of doing real estate agency work in Morocco. We thought of developing a system that is meant to provide users with all the information they need in order to buy, sell or rent houses.

This kind of applications is still not very common in Morocco. In fact, people still like to deal with traditional real estate agents and this is due to many reasons. For instance, Moroccans are still not familiar with new technologies and the digital world in general. However, we would like to replace this traditional way since finding a trustworthy agent is still a difficult task for some. Therefore, this application will, hopefully, facilitate this task of selling, buying, or finding the appropriate house by providing a platform that is available with all the necessary information that users might need, as well as a user friendly interface that is easy to use. This web application will save a lot of time and money for the users. In fact, it will be available 24/7. Therefore, this application is a tool that will allow, on one-hand property owners to publish and their properties: fields, houses, or villas and advertise their possessions. On the other hand, people will be able to view these goods and to choose the ones that most suit their needs.

However, instead of being like Avito or any normal website of announcements, a revenue must be generated after each transaction made between both parts within the website. In fact, it is worth to mention that the application will make profit thanks to the transaction between clients, who make most of the transaction steps through the application.
3. Methodology:

At some point in this semester, I had to choose a methodology in order to develop the idea of the project I am working on. In fact, we studied many types of models to be followed in projects like this one in software engineering classes. However, the difficulty or the challenge remains on choosing the most reliable and consistent one. I hesitated between the waterfall model and the incremental model. However, after many discussions and thoughts, I decided to follow the incremental model. In fact, this model is based on gathering solid and concrete specification requirements from the client; however, it allows the flexibility and adaptability at the design and the implementation of the application. The engineer has this flexibility in terms of the design, implementation, and maintenance. Real estate world will always face changes in the future. Therefore, in terms of the design and implementation, the engineer will have to get adapted to those changes. I decided that the right model that should be used in this project is the incremental model one. In this case, I will always have the ability to improve the quality of the application by changing some specific substances in the design and implementation phase so that the final version of the product meets the requirements set by the client at the first place.

The second point that I would like to discuss is the way I obtained the functional, non-functional and the domain requirements. Requirements’ gathering is a crucial step in every project. The help of my supervisor Dr. Assem Nasser, as well as the help of national and international real estate websites, that are mentioned in the references part, were the main reasons I succeeded in gathering solid requirements.
4. Main Text:

4.1. Applications Requirements

In this step of the project, my teammate and I gathered the requirements of the application. This was done through meetings with our supervisor Dr. Nasser Assem. This step came after studying the feasibility of the project as a whole and see whether it is feasible or not under time and tools constraints. After these meetings, we were able to decide on the functional, nonfunctional as well as the domain requirements of the application. It is worth to mention that, regardless of the methodology used in a software engineering project, requirements specifications are very crucial for the development of any project.

The following list is a summary of all the functional, nonfunctional and domain requirements regarding real estate agencies in general and my project, “Diorr,” specifically.

4.1.1. Functional Requirements

4.1.1.1 For the User

a. Register

➢ Registration will be based on the First and Last name, CIN, Email address, Password, Secret Question, and Secret Answer (for security matters). In addition to the date of birth and gender.

b. Login

➢ Login will be based on the Email address along with the Password. An error message will be displayed in case one of the two does not match.

c. Manage Announcements:

   i. Add Announcement
   ii. Modify announcement
   iii. Delete announcement
   iv. Search for announcement

       1. By Address or city
       2. By Category
       3. By Price

4.1.1.2 For the Admin/Employees

a. Manage Users

   i. Add User
   ii. Modify User
   iii. Delete User
iv. Search for User
   1. By First Name
   2. By Last Name
   3. By Date of Registration

b. Manage Testimonials:
   i. Add Testimonial
   ii. Delete Testimonial
   iii. Modify Testimonial
   iv. Search for testimonial
       1. By User
       2. By Date

b. Manage Announcements:
   i. Add Announcement
   ii. Modify announcement
   iii. Delete announcement
   iv. Search for announcement
       1. By Address or city
       2. By Category
       3. By Price

c. Manage Employees: (Exclusive for the admin)
   i. Add Employee
   ii. Modify Employee
   iii. Delete Employee
   iv. Search for Employee
       1. By First Name
       2. By Last Name
       3. By ID

d. Manage transactions:
   i. Add Transaction
   ii. Modify Transaction
   iii. Delete Transaction
   iv. Search for Transaction
       1. By Date
       2. By Registration
e. **Manage Payments:**
   i. Add Payment
   ii. Modify Payment
   iii. Delete Payment
   iv. Search for Payment
      1. By Date
      2. By Type
      3. By Transaction

f. **Manage Visitations:**
   i. Add Visitation
   ii. Modify Visitation
   iii. Delete Visitation
   iv. Search for Visitation
      1. By Registration
      2. By Employee
      3. By Date

### 4.1.2. Nonfunctional Requirements

After all the meetings, we realized that there are only three nonfunctional requirements. The web-application must be coded using Php5, along with CSS, HTML5, and JavaScript. For the database, it should be managed using MySql. The deployment must be by the end of the spring 2015. The application must have a user-friendly interface so that it can provide its services in a way that will be clear and easy for the user to follow. The application must be **secure, reliable, easily accessible** to users, as well as **scalable and portable.** So that we can run the software on other platforms easily. For the security part, and since the user will have the ability to insert some sensitive data to our database, I must add some strong PHP code behind that will prevent from any kind of SQL injections or any similar attacks. All of this can be encapsulated by using a **framework** that will provide me with particular functionalities that will make the task of developing the web application easier. This is by including compilers, new libraries, tool sets, and different APIs that will help me develop this application in an easy way. However, for the moment, I will focus on implementing the application without using the framework as a beginning, and I will leave this part as a future work since the tools I am using in this project are already new for me and need some time to grasp. Another important point that should be mentioned, we thought that the web application must be available in French since our clients and users of the application are Moroccans who are fluent in French.
4.1.3. Domain Requirements:

“Diorr” must follow the requirements of the real estate business. This is why we need to note that this application will have to serve the needs of different users within the real estate field. Moreover, we need to note everything that might be against the law in the real estate business. All the articles and websites that helped us at this point are included in the references section.
4.2. Entity Relationship Diagram

After finishing the collection and the analysis of the functional, nonfunctional, and domain requirements, I was able to draw an entity relationship diagram with my teammate Sawsane Ouchtal. As a first step, we had to identify the entities that this project requires based on the modules of the application that we already identified. During this semester, we drew many versions of the entity relationship diagram. In fact, we had to modify it many times since we discovered many anomalies as we are progressing in the project. After many discussions, we came up with the following, which is the latest version of the entity relationship diagram:
4.3. Description of the entities:

The database of our project “Diorr” will contain the following tables: User, Account, Announcements, Registration, Property, Property_description, Price_History, Visitation, Employee, Transaction, Payment, and Testimonials, Remark.

The following part is a description of each entity:

**User:** This table refers to the users using the system of the real estate agency. Each user is defined by his/her CIN, which will be the Primary key of this entity since it should be unique between users. Moreover, a user is defined by a first name, last name, gender, a date of birth, and date of registration.

**Account:** This table will keep track of all accounts in the system. Each user will have an account that will be defined by an Email, which will be its primary key. Therefore, if a user wants to get connected, he/she has to enter his/her Email and password. For security issues, an account will be defined as well by a secret question and its answer in case the user forgets the password. The last attribute is registration_date, which will help to know the date of registration of each user.

**Property:** This table is used to refer to the different users’ properties that are ready to get rented or sold. It is defined by an ID, type, name, city, and a specific address. Each property is going to be linked to a specific user.

**Property description:** This is used in order to know the category of the property to be entered into the system (property_type attribute). This table will be identified by the id of the property description, property area, number of rooms, number of façades as well as a little description of it.

**Testimonials:** This table is for gathering the testimonials of different users who used the application. Each testimonial will be recognized by an ID, text, and the date of release. Each testimonial will be linked to the user who made it by using his/her CIN.

**Announcement:** This table will keep track of all announcements made by the users of the application. In fact, each user having a property to sell or rent, can create an announcement that will be defined by an announcement ID, date of the announcement, the type of the announcement, as well as the status. The user who will create the announcement must be a member of the website. Therefore, he/she has to create an account before getting into the step of managing (adding, modifying, deleting) announcements.
**Registration:** This registration table is known by a registration_ID, date, and description. This table is going to be used by the users of the application who are interested in a property posted by another user.

**Employee:** An employee will be present during the visitation date in order to take remarks about the visitation between the two users. Each employee is identified by a CIN, ID, First and last names, date of birth as well as his/her status.

**Visitation:** an ID, date, and type identify each visitation. One of our employees will schedule each visitation. Therefore, it is the employees’ responsibility to manage those visitations and give remarks on each one of them.

**Remarks:** A remark is a note identified by a primary key Remark ID, date, along with its content. A visitation ID will play the role of the foreign key in this table.

**Transaction:** This table is going to be used in order to keep track of all transactions made between users of the application. Each transaction is identified by an ID, an amount, as well as the date of the transaction. This table contains registration ID as a foreign key in order to keep track of the users behind these transactions.

**Payment:** The payment table is identified by an ID, amount, date as well as its type. The table contains as well transaction ID that represents a foreign key coming from the transaction table. This foreign key is the one that will help us identify the commission to take from every transaction.
4.4. Description of the relationships between the entities:

**Account-User:**

This is a one-to-one relationship since every user should have exactly one account by which he/she will perform the different possible operations on the application.

**User-Testimonial:**

Each user can write many testimonials where he/she expresses their opinion about the application and report any bug or unsatisfaction.

**User-Announcement:**

A user can create an announcement to find buyers/renters for his/her property, and create as many announcements as he/she needs. A user can also be registered just to view the announcements of others, which justifies the cardinality of many side; zero or more.

**User-Registration:**

Once an announcement is created and posted on the application, users who are interested in the property being announced can show their interest to the owner by creating a registration into that specific announcement. A user can create as many registrations as he needs, and might create none if he is a property owner rather than a buyer or a renter. This justifies the cardinality of the many side, zero or more.

**Announcement-Property:**

As an announcement serves to allow a property owner to publish this property for sale or rent, every announcement should be related to a property. Also, a user can create as many announcements for the same property depending on his/her needs.

**Price_History – Property:**

A property has many price histories as the price is changeable by the owner depending on how other users respond to his/her announcement. The property should have at least a price history which is the one set at the beginning in case of no change.

**Property-Property description:**
Each property will have only one description, since each property description will be defined by an id. Moreover, because of the reason that each property will have its structure and thus, a specific number of rooms, floors, façades, garages etc.

**Registration-Transaction:**

A registration might lead to a successful transaction between both parties, or might not lead to any transaction at all if the parties do not agree on its details. This is why there is a one to one relationship between Registration and Transaction entities with a cardinality of zero or more.
4.5. Sequence Diagram:

As it has been specified in the interim report, a sequence diagram is an illustration that will show us clearly the scenario of events that the users/classes involved in the system will go through. We included it because we realized that a sequence diagram might be helpful for us to implement as well as understand the application as a whole. It will represent all the functionalities within the application and their succession before performing a specific service by the user.
The diagram shows that buyers, sellers or renters who are using our application need to register first by creating an account. Then, the owner of a specific property creates an announcement to publish details about it in order to look for interested users. If a user registers into one of the announcements, it means that he/she is interested into that property. Then, the administrator schedules a visitation depending on the availability of both parties and specifies which employee of the real estate agency will accompany them to the property. Once the visitation is over, the employee has to record any remarks related to the meeting between the buyer and the seller. If the real estate transaction happened successfully, the employee has to record the transaction and any detail about the legal process between the parties of the transaction. These have to pay the commission in favor of the agency, which made their transaction happen. This payment is recorded by the administrator to keep track of the income.

The users who registered in our application, once the visitation is over and the employee recorded the remarks, have to pay the fees in favor of the agency which made their property being sold or rented. It is important to mention that this payment is calculated according to the percentages set by the AMAI, Association Marocaine des Agences Immobilières [1]. Indeed, after some research, we found that real estate agencies can make profit out of each real estate transaction depending on the type of that deal: sale, goodwill, rent for a short term or long term. Payments are recorded by the employee who supervised the transaction to keep track of the income of the agency. The percentages of the commission are included in figure (4.5.1), Appendix A.
4.6. STEEPLE ANALYSIS:

The STEEPLE analysis consists of looking at seven external aspects that might influence or get affected by our application. Once we will get in touch with the external environment, we will have to know how to deal with the possible risks that we might face.

1. **Social:**

We would like our users to save time and money through this application. Therefore, an application like this one will help them find an intermediary, who is available 24/24 and all days of the week, and will help them rent, buy, or sell their properties in a fast and legal way. Therefore, an application like ours will help them reduce risks of conflicts and misunderstandings before getting their transactions done.

2. **Technology:**

‘Diorr’ is a modern way to find a renter, buyer, or seller of your property in a minimum amount of time. This is because our application will be available 24/7 and accessible from different devices. Therefore, we think that this application will push the Moroccan community to get more familiar and in touch with technology in the field of real estate.

3. **Economic:**

Since this application makes it more easy for users to find others interested by their property, or the other way around, we think that the application will help the Moroccan economy in the field of real estate. Moreover, the application would be affected directly if the real estate activity if affected by a national crises.

4. **Environment:**

Users of this application will not have to move in order to find a seller, buyer, or renter for their property unless for a meeting with the other user who’s interested. Therefore, reducing mobility is a way to say that our application will reduce a little bit of the environmental pollution in the Moroccan streets.

This application would have an indirect effect on the environment. In fact, people will not have to look for intermediaries and real estate agencies in the streets to buy, rent or sell their properties. Our application will make them find solutions for their needs while reducing their mobility, and so the environment pollution.
5. **Political:**

“Diorr” is an application that follows the Moroccan rules and policies of the real estate activity. If there is a change in the commissions that the real estate agency gains from every transaction, then the income of the real estate agency is affected directly by the political environment.

6. **Legal:**

“Diorr”, as every other application, needs to respect the rules of the field of activity. Therefore, it needs to respect the real estate law in Morocco, that is regarding the commissions and the percentage that the application will gain from every real estate transaction made through it should not exceed what is allowed by the government.

7. **Ethical:**

The last point is the ethical context, where the confidential information of the clients must be respected. For instance, we should not use the localization of the clients without informing them. All users must be using the application in a safe context where their profiles are not used or shared with external environment.

The design phase is as well an opportunity for us to get a description of the construction of the application, which is “Diorr”, to be implemented. It will give us an answer on how the application is going to be made. This is to say that we will be answering questions about the technology enablers, appropriate algorithms implemented within the application, and every detail that will help the process of implementing the application. The design phase starts with defining the technology enablers. It means that we need to choose carefully the programming language to be used, the DBMS (data-management system), along with the operating system on which the application will run and the GUI (graphical user interface tools). The next section will describe deeply each one of these.
4.7. System Architecture:

- **Process used to schedule visitations:**
  
  Before discussing the system architecture, I would like briefly to comment on the process that I followed in order to help the administrator to schedule visitations between the property holders and users who are interested in that specific property. This will help getting an idea about the free services that I made use of within my application. The process starts by making a registration on a specific property, the administrator will check the availability of the property owner, who already inserted it as part of the announcement information. This availability will help scheduling a visitation with the interested person, using **Doodle**. In fact, the free scheduling service was used in order to help our users to get a meeting as soon as possible without any complication. In fact, Doodle will help scheduling in a simplified way and will prevent us from reinventing the wheel. Meaning that we won’t have to implement a functionality that’s already available. Once the interested person enters his/her availability on doodle, the administrator will be able to email both persons and inform them about the meeting details. For this matter, I used **Gmail** as a free service that will facilitate the communication between the administrator and clients.

- Once the meeting is held, the administrator will have to note the remarks. If the real estate transaction is made and both concerned clients are satisfied, an invoice will be generated and sent to both clients on **Gmail**.
The architecture of this system, as shown in the figure above, is composed of a web browser or client that allows the application’s users to make use of the different functionalities of the system. All of the requests will start through a web-browser (google chrome f.e). Once the request is made, it is going to be sent to the web-server using HTTP requests. The request will as well reach the database through PHP language. In this step, as I described in the non-functional part; I am aware of the sensitivity of the data that will be entered by the users and that must be checked before inserting it into my database. This will be done by adding PHP code that will prevent from SQL injections. Moreover, a solid framework must be chosen carefully so that we can have a secure and professional application.
4.8. Technology enablers:

As stated in the previous paragraph, the technology enablers are the tools to be used in order to implement the application. These tools are the programming language, the database model system, the operating system that the application will work in and finally the graphical user interface. Choosing these tools is a crucial step in our project. Project requirements, client or supervisor requirements, simplicity, robustness, and programming paradigm are the main constraints that make us stick to one technology enabler and not the other. In fact, we need to take into consideration all of these restrictions and see what technology enablers are more suited for the project. The following are the ones that we ended up by choosing to implement this project:

a- Programming paradigm:

We had to opt for the object oriented paradigm since “Diorr” is web application for the management of users, announcements, accounts and registrations made by users. Therefore, it will be easier for us to treat these data as objects. Since we have worked with the object-oriented paradigm previously, it was obviously the best alternative at this point. The object-oriented paradigm will help us to use the concept of objects, and thus, data structures that encompasses data. This is the most suitable way to do it because managing and linking these to our database would be easier.

b- Programming Language:

Since we have specified the programming paradigm to follow, now it is time to look for a programming language that supports it. We have opted for PHP Version 5.5.12, because of two reasons: The first one is that I had to respect the requirements of my supervisor. The second reason is that since I have never had the chance to code an application using PHP, it was the perfect opportunity for me to learn a new programming language. For the version, I opted for the latest one available. However, Java was as well one of the alternatives since it respects the programming paradigm I talked about in the previous paragraph. Along with PHP, I had to choose HTML 5, CSS, and JavaScript. For the JavaScript part, I had to include JQuery library, which is free and open source, for the simplification of the client side scripting. The reason behind choosing these complementary programming and markup languages is that we need an application that will be user friendly. Therefore, in order to have an interactive webpages that will make the experience better for the users using our web application.

A slight point that I want to mention in my report is that I included Bootstrap in my project. It is “the most popular HTML, CSS, and JS framework for developing responsive,
mobile first projects on the web.” This way, if a user would like to use our services in his/her mobile phone. The webpage will be adapted to his/her device’ screen he/she is using.

In order to reassemble all of this code, I needed a text editor. After few discussions with some friends who coded before their projects using PHP, I have chosen to work with SublimeText 2 which is a sophisticated text editor for code and markup. With a user friendly interface and many shortcuts that helped me a lot at this point of the project.

c- Database Management System (DBMS):

Alongside PHP, I opted for MySQL as a relational database management system. This choice was based on the fact that MySQL is the most widely used open source DBMS and the second most widely used relational database management system. The other reason to use MySQL is the fact that I am using WAMP server as a windows web development environment [9]. This free service offers as well PhpMyAdmin that allowed me to manage, easily and in a short period, my database. It might have been more difficult to manage in case I have chosen another DBMS.

d- Browsers:

The application is compatible with Google Chrome, Safari, Mozilla Firefox, or Internet explorer. The users can be connected to us no matter what browser is used. As stated before, the Bootstrap file included and used within the CSS code is a way to make the web application compatible with any browser, even in smartphones. Bootstrap is a magnificent way to adapt your web application to different browsers by including “a responsive, mobile fluid grid that appropriately scales up to 12 columns as the device of viewport size increases” [2]. This is by using the predefined classes in the CSS file that are easy to use. The following are screenshots from both browsers, google chrome and Mozilla Firefox, and the results are quite the same:

a. Google Chrome
b. Mozilla Firefox:

c. Google Chrome on a Smartphone:
The 12 columns grid functionality in the Bootstrap file is the best way to make the web application compatible with any browser.
4.9. **Database Synchronization between the mobile application and the web application:**

One of the biggest issues that my teammate and I encountered in this project is the database synchronization issue. In fact, at some point in this semester, we decided to synchronize both databases that we are working on in order to keep the consistency of data. We thought, with the coordination of our supervisor Dr. Assem Nasser, that it would be better if we can synchronize and link both databases in order to get rid of duplicated information. It is important to keep in mind that what made this synchronization somehow challenging is the use of two different database management systems. In fact, my teammate is using SQLite for the mobile application, and on the other hand, I am using MySQL as described previously in the technology enablers section. As a result, we faced many issues in terms of coding. The next paragraph summarizes exactly the processes that my teammate and I followed as well as the issues encountered during each process.

a- **First alternative:**

We thought that each one of us keeps the same database management system he started with at the beginning. This means that I will keep the MySQL in my web application part and Sawsane, my teammate, will keep using SQLite in the mobile side. When we looked up for solutions, we found that it will involve the usage of two external libraries: Gson and android-async-http. The process will go as follows:

The mobile or Android app will send HTTP requests to the web application. This latter will respond by sending back unsynchronized data in JSON format. This means that the web application will send back all the information that still needs to be synchronized and therefore changed in the mobile application. These data are finally fetched by the mobile app and added to SQLite db. However, we noticed that this alternative is very heavy in terms of code and will cost too much for the web application and mobile application. In fact, there is a need to include many PHP functions in the web application side, and many Android classes on the mobile application side. Here are the issues that we faced once we included the code in both applications:

- **Issue 1:** The external library Android-Async-http does not allow HTTP requests to be sent to the web application in a correct way.
- **Issue 2:** Once this code is compiled by Android Studio, the whole project crashes. This is because of the difficulty to compile using Android Studio. In fact, the IDE used by
teammate takes a lot of time to compile pieces of code in a short period.

b- Second alternative:

Since we faced many issues when we wanted to go for the first alternative, we thought of a second alternative that is easier in terms of realization. We thought of a way where we can use the same database management system. In fact, we thought that it might be interesting that both of us. This means that we will use MySQL under the web application as well as the mobile application. The process goes as follows:

1- Go to PHPMYAdmin and grant all privileges to mobile database. This will allow my teammate to access all the tables that I built in my side along with their contents.
2- Go to the command line and use GRANT ALL PRIVILEGES command.
3- Mobile database established connection with web database
4- Access data smoothly.

This was supposed to be the process to follow in order to get into the synchronization of data between both applications. However, we faced an issue as the following paragraph describes:

- Issue: the connection cannot be established between both databases. Therefore, the connection cannot be established between both machines and thus, unable to exchange data between both applications.
- Reason: after few discussions and research, we think that it might be an issue caused by the ITS department at the university. They might have blocked this remote access privilege from a mobile to a computer. Meaning that accessing a computer resource remotely cannot be done within AUI’s network, due to some constraints made by the department for security issues.

Therefore, we tried our best to synchronize both databases and make the applications having the same content in terms of data, but we failed to do so. However, as I will describe in one of the next paragraphs, entitled future work, our thoughts of some ways that we might try in the future and make the synchronization happen, so that we can get rid of redundancies in our project.
4.10. **Implementation:**

As each software engineering project, the implementation is one of the important project’s sections that we need to take seriously. It is a section where the engineer has to translate or transform what he/she defined as functionalities or operations, into a programming language. This part of the project consists of converting what we ended up with in the design phase to an executable software using programming tools that must be chosen correctly and with care. According to software engineering A Practical Approach to Software Engineering for Students and Apprentices, the implementation phase consists of integration, testing, coding and deploying [5].

Since we agreed on new tools and technology enablers to work with, we were obliged to start the implementation early this semester. I was able to implement the final web application that I will present during the day of the defense. The process started with organizing the project into folders so that I can easily call files from different locations [6]. The implementation consisted of building web pages that are linked with the database, and finally link those webpages together. Each action from the user needs to lead to a result. This result might be an error message, a satisfaction message, or a redirection to another webpage.

During this project, and after drawing the entity relationship diagram, I had to start by building the database from my side, using MySQL as described in the previous section, with all the entities and attributes we agreed on, in addition to filing it with few elements for testing. The next step was to start working on the interface of the web application. This means that I had to build an empty interface, using HTML, JavaScript and CSS only. Then comes the link between webpages and the database. This was the most difficult part since I needed to include PHP code inside HTML in order to write queries and get or send information from or to the database. With the collaboration of my teammate, Sawsane Ouchtal, and my supervisor, Dr. Assem Nasser, I was able to implement all the functionalities that we agreed on at the beginning of the semester.

In this section, I would like to give some code snippets and describe what it does. As every web application, I had to make sure that the user has to register before benefiting from all our services. This snapshot will help me to explain how the registration process is done.
Under the Function.php file, which is a PHP file where I put all my functions that I use in my webpages, we can find a function called register [8]. After getting all the information from the user who would like to register, I connect to the database and put all the information in both tables, User and Account, with inserting a 0 into the Account_Active attribute. This is to know what accounts still need to be verified. Once registered, I need to make sure that the user activates his/her account. It is done by sending him/her an email on his/her inbox with a randomly generated link. Once he/she clicks on the link, he/she will be directly redirected to a new webpage where he/she can see his/her profile, add announcements, or search for other announcements. Thus, the Account active attribute within the account table will be switched from a 0 to 1. This last step is done through a function that I called update, and here is a screenshot of the function:

```
function update ($code)
{
    global $db;

    $select = $db->prepare("UPDATE account SET Account_active=1 WHERE Account_activation='$code'");
    $select->execute();
    $count = $select->rowCount();
    if($count>0) return(0);
    else return 1;
}
```

- I would like to mention that this process of registering users, sending an email, and activating their accounts might be thought of as a Trigger. Even if it is not the same syntax that we learned to do in the database course, this piece of code is automatically executed as a response of certain events on the user and account tables.
Sending emails while working on a localhost is not an easy task. In fact, since I am working on my own computer and using it as a localhost, I had to find a way to send emails to users. After few days of research, I found a way to do it. This was by using the PHPMailer [7]. Indeed, it is one of the most known libraries used by developers in order to send emails from a localhost. It does that by the integration of SMTP support. Once I downloaded the folder available for free in the website that I mentioned in the references section, I needed to include a specific PHP file in my code. The following snippet is a line of code that’s responsible for it:

```php
<?php
require_once 'include/PHPMailer-master/PHPMailerAutoload.php';
@session_start();
?>
```

- Places-autocomplete: API provided by Google for developers:

At some point, the user will have to enter the address of his/her property that he/she wants to promote. Therefore, I have chosen to include the free places-autocomplete API provided by Google so that the user enters the address in a correct form [3]. This functionality has been used as well once one of the users wants to search for a specific property. This piece of code is divided into two sections: JavaScript Code and HTML code. Here is a snippet of it:
These functions are used in order to select an address from the dropdown/combobox that will be displayed for the user. Next, it will get the place details from each address entered: city, street name, and country. Then, it will populate some specific text fields coded in HTML. The result of this code included will be shown in the next section.
4.11. Implementation results:

a- Registration page:

This section will be focusing on the implementation results. This means that I will be posting and commenting on each one.

First of all, a user might need to register in order to get the privilege to post announcements and promote his/her property. Therefore, I had to work on a registration page that will help me to get information about each user.

![Registration Page]

b- Login/add announcement:

Once registered, the user will be able to login and promote his or her property.
In this figure, we can see that the user is already logged in and a session was opened on his name (upper right corner) and now is able to enter information about his property. We can see as well that once he wanted to fill the address field, suggestions were given to him in order that he can enter the correct address.

- **History of prices:**

Each user, after publishing his/her property, has the write to modify it. However, one of the important records that we need to keep track of are prices. In fact, we need to keep a record of all the prices that were given before to one specific priority. Therefore, I thought of implementing a section where the user can see all the previous prices, he/she has entered in the system. Moreover, he/she will have the possibility to share those prices with the public. This means that once a random user search for an announcement, he/she will be able to see the record of prices related to that property so that to get an idea on how the price of the property is progressing.
Search for properties:

This figure banner is located in the index page to help the users to search for properties according to the type of property, address, city, price, or type of announcement.
Once he/she clicks on the search button, he/she will be redirected to a page where he/she can see all the results related to what he/she is looking for. I am using a default image just for the sake of the demonstration.
5. Future work:

As you have understood, this is a group project that is composed of a web application and a mobile application. This report was made to describe all the major features of the web application as a first part of the project, and my teammate Sawsane Ouchtal, took care of the second part of it, which is a mobile application. I described earlier in this report that at some point of this project we faced a database synchronization issue that we could not finish implementing it, which made us go for an alternative that consists of creating database duplicates. Therefore, as a future work, we would like to synchronize our databases in order that the project becomes more professional and well equipped to be deployed.

Moreover, we thought of finalizing the payment section. This is by integrating Payzone, a Moroccan platform for online payment, or just create a Paypal account in order to make the payment through our platform.

The last thing that we thought of is to include the localization of our users from their IP address. In fact, it would be interesting to detect the localization of our user in order to suggest for them all the properties we have in our database that are localized in their surroundings.
6. Conclusion:

This web application was done in the context of my capstone project. It would be a very useful for people who are no longer interested in the traditional real estate agent and do not want to waste their time because of the unavailability of the outdated real estate agent. This system will provide all the necessary information for a user about different properties available for sale or rent around Morocco. It is as well a great opportunity for different users to promote their properties and find the accurate buyer or rented. This is as well an opportunity for Moroccans to use technologies and get more familiar with the online applications related to real estate agencies.

This capstone project was a great opportunity for me to know more about the process behind real estate agencies as well as using new technology enablers and tools to implement this application. This was as well an opportunity for me to recapitulate all what I have learned at Al Akhawayn University in order to get to the results needed. Moreover, it was an opportunity for me to manage my time between classes, office hours with our supervisor Dr. Assem Nasser, and progress in the project. On the other side, it was great occasion for sharing and developing teamwork skills with my teammate, Sawsane Ouchtal, who without her the results of this project would not have been the same. Finally yet importantly, it was the first time that I needed to write diaries and keep my supervisor updated with all the progress I did each week.
7. References:


8. **Appendix:**

The table below shows the percentage of each transaction, which happens between users, I can make once the web application is deployed.

<table>
<thead>
<tr>
<th>Transaction</th>
<th>Commission</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vente/Achat</td>
<td>2.5% H.T (3%TTC) sur la valeur totale du bien payable par chacune des parties : Vendeur et Acheteur.</td>
</tr>
<tr>
<td>Location</td>
<td>1 mois H.T du montant de la location payable par chacune des parties : Locataire et Bailleur.</td>
</tr>
<tr>
<td>Location Courte Durée</td>
<td>25% H.T du loyer mensuel.</td>
</tr>
<tr>
<td>3 mois</td>
<td>50% H.T du loyer mensuel.</td>
</tr>
<tr>
<td>6 mois</td>
<td>1 mois H.T du loyer mensuel.</td>
</tr>
<tr>
<td>12 mois</td>
<td></td>
</tr>
<tr>
<td>Gestion locative</td>
<td>10% H.T de la valeur mensuelle du loyer</td>
</tr>
<tr>
<td>Fond de commerce</td>
<td>10% H.T de la valeur du fond de commerce</td>
</tr>
</tbody>
</table>