**PAYROLL SYSTEM**



Table of Contents

[2 Table of figures 4](#_Toc40167092)

[3 Table of Tables 4](#_Toc40167093)

[4 Environment/Technology Architecture 5](#_Toc40167094)

[4.1 server 5](#_Toc40167095)

[4.2 network 5](#_Toc40167096)

[4.3 Programming Language 5](#_Toc40167097)

[4.4 Database 6](#_Toc40167098)

[4.4.1 Features 6](#_Toc40167099)

[4.4.2 Costs 6](#_Toc40167100)

[4.4.3 Reliability 6](#_Toc40167101)

[4.4.4 Upgradability 6](#_Toc40167102)

[4.5 Security 7](#_Toc40167103)

[4.6 Storage system 7](#_Toc40167104)

[5 Use case employee record 8](#_Toc40167105)

[5.1 Use case description 9](#_Toc40167106)

[5.2 Network diagram 10](#_Toc40167107)

[6 Use case for attendance 11](#_Toc40167108)

[7 Use case for salary 14](#_Toc40167109)

[8 sequence diagram for request payment salary 15](#_Toc40167110)

[9 State diagram 15](#_Toc40167111)

[10 Component diagram 16](#_Toc40167112)

[11 class diagram 16](#_Toc40167113)

[12 Entity Relationship diagram 17](#_Toc40167114)

[13 Navigation diagram 18](#_Toc40167115)

[14 Website’s frontend 18](#_Toc40167116)

[15 WEB LINK 20](#_Toc40167117)

# Table of figures

[**Figure 1 Use case employee data** 8](file:///C:\Users\1999m\Desktop\KUWAIT%20PAYROLL%20SYSTEM%20UPDATED.docx#_Toc40167075)

[**Figure 2 Network diagram** 10](#_Toc40167076)

[**Figure 3 Use case attendance** 11](#_Toc40167077)

[**Figure 4 Use case salary** 14](#_Toc40167078)

[**Figure 5 Sequence state diagram** 15](#_Toc40167079)

[**Figure 6 State diagram** 15](#_Toc40167080)

[**Figure 7 Component diagram** 16](#_Toc40167081)

[**Figure 8 Class diagram A** 16](#_Toc40167082)

[**Figure 9 Class diagram B** 17](#_Toc40167083)

[**Figure 10 ERD** 17](#_Toc40167084)

[**Figure 11 Navigation diagram** 18](file:///C:\Users\1999m\Desktop\KUWAIT%20PAYROLL%20SYSTEM%20UPDATED.docx#_Toc40167085)

[**Figure 12 Image A** 18](#_Toc40167086)

[**Figure 13 Image B** 19](#_Toc40167087)

[**Figure 14 Image C** 19](#_Toc40167088)

[**Figure 15 Image D** 19](#_Toc40167089)

[**Figure 16 Image E** 20](#_Toc40167090)

[**Figure 17 Image F** 20](#_Toc40167091)

# Table of Tables

[**Table 1 Employee use case Description** 9](#_Toc40167067)

[**Table 2 Employee Actor's Description** 9](#_Toc40167068)

[**Table 3 Attendance use case description** 12](#_Toc40167069)

[**Table 4 Attendance actor's description** 13](#_Toc40167070)

# Environment/Technology Architecture

## server

For the server we choose:

* Model: P06420-B21
* Description: HPE ProLiant DL380 Gen 10 4110 2.1GHz
* 8-core 1P 16GB-R P408i-a 8SFF 500W PS
* Unit price ~= $ 6,000++

## network

For the network we choose:

VPN, or Virtual Private Network, allows you to create a secure connection to another network while using the Internet.

VPNs forward all our network traffic to the network, which is where the benefits arise:

* accessing local network resources remotely.
* bypassing Internet censorship.
* numerous benefits from the functionality, security, and management of the private network.
* data encapsulation and encryption to ensure data is secured.

This type of connection works best for private networks.

## Programming Language

For the programming language we choose Python since it is the best suited for our needs. It is a simple programming language with an easily readable syntax. Due to its simplicity developers can focus on actually solving problems instead of wasting both time and energy understanding and analyzing the technical nuances of the language.

1. Python focuses on code readability. It is versatile, easy to use and learn, well-structured, and fast to develop.
2. It is an open source language. Meaning it is free to use.
3. Python’s community is one of the best in the world. It is large, active,  and some of the best programmers contribute to both the language itself and the community.
4. It is great for prototypes because trying new ideas is much quicker in python that in other languages. Hence it saves time and is less costly.
5. Python can run on Mac, Windows, and Unix systems and has also been ported to Java and .NET virtual machines.

## Database

For the database we choose:

### Features

1. ACID – compliance, helps in maintaining data integrity and reliability.

2. Support cross-network communication, it has a networking stack which allows communication between applications that run on different platforms such as windows and oracle databases running on UNIX.

3. It is cross-platform, it can run on different hardware supported by various operating systems such as the wide distribution of GNU/Linux, UNIX and windows.

4. Back-up and recovery, It has a powerful tool, Recovery Manager (RMAN) which performs cold, hot and incremental backups that ensure the integrity of data in case of system failure.

5. Clustering, it has real usage regarding the Real Application Cluster (RAC) which ensures the system to run without interrupting the services when one or more servers on the cluster fail.

### Costs

1. Real Application Clusters ( ~ $20,000 per processor).
2. Taking into account other additional costs (the cost per PC in the US ~ $ 110,000).

### Reliability

1. It employs Real Application Clusters (there is maximum availability).
2. It has a powerful back-up and recovery manager which ensures data integrity in case of a system failure.

### Upgradability

1. It offers several methods of upgrading the database based on the complexities of the enterprise.
2. Oracle has doubled the speed of MySQL query handling. It has made some changes to boost the speed of the database management system.

## Security

For security we choose:

Total AV, it is a powerful security tool that is very easy to use. Total AV is the #1 software vendor throughout Trustpilot. the pro version has more features. TotalAV Ultimate Antivirus is an advanced version of the software which is priced for $59.95 a year. Some of its features include:

1. Ability to detect malware and viruses.
2. Automation, it can find and delete duplicates without the user even knowing about it.
3. Cross-Platform, TotalAV works with Windows, Mac, Android, and iOS.
4. bonus features like Cleanup and System Boost helps the computer perform well.

## Storage system

For storage we choose:

Lenovo ThinkSystem DB620S. Some of its features include:

1. Increase in performance for demanding workloads.
2. Simple end-to-end management by automating repetitive daily management tasks.
3. Runs Fabric OS, which delivers distributed intelligence throughout the network and enables a wide range of value-added features.

# Use case employee recordA close up of a map Description automatically generated

**Figure 1 Use case employee data**

## Use case description

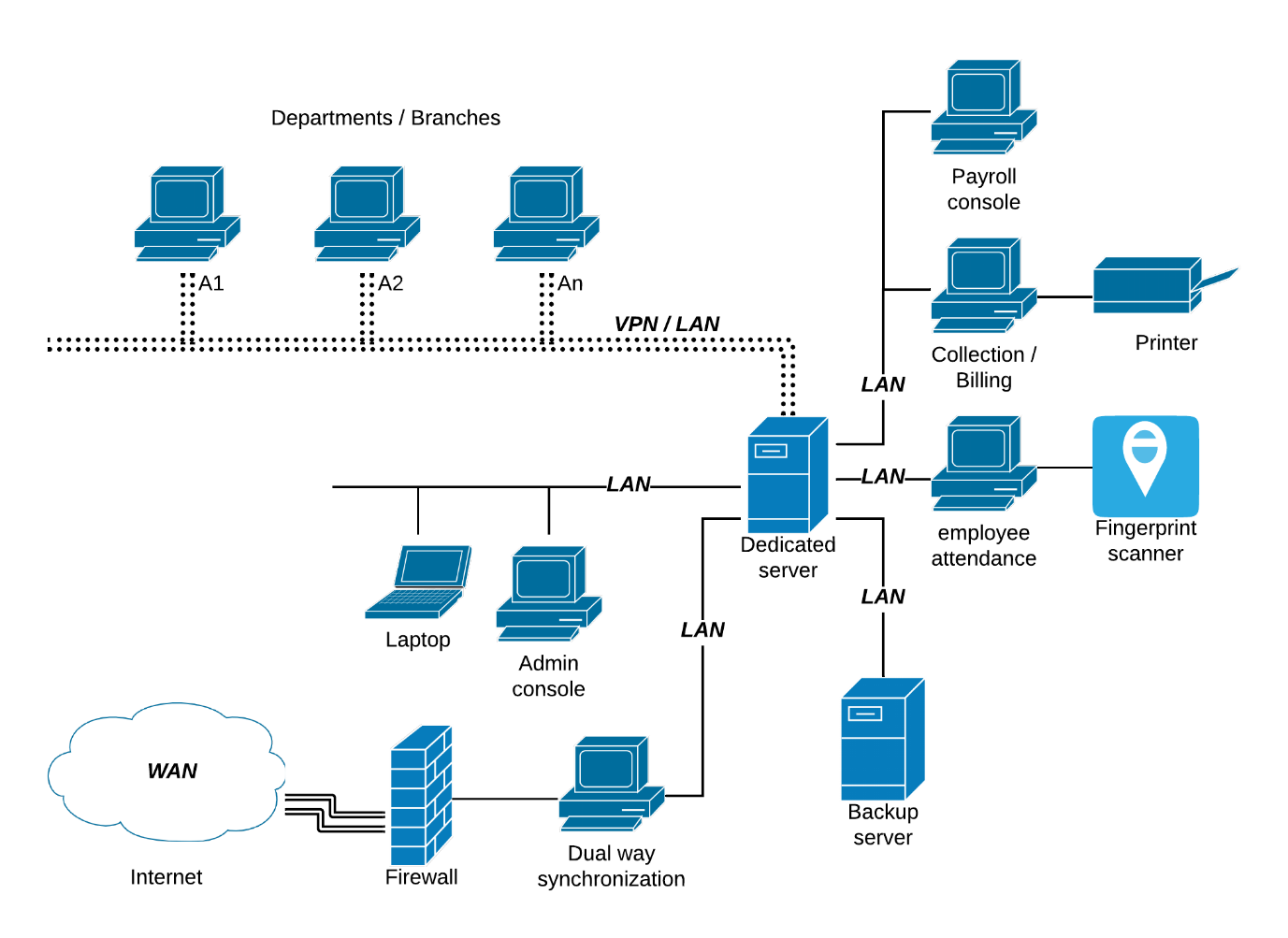
|  |  |
| --- | --- |
| Use Case | Description |
| Enter Data | When a worker applies for the company HR staff will enter the Name, day of birth, ID number, and day of starting the job. |
| HR manager will authorize the data. |
| Add the job title | HR manager decides the job title for the new worker. |
| Then the HR staff will add the job title. |
| Salary amount | HR manager chooses salary amount. |
| HR staff will add it. |

**Table 1 Employee use case Description**

|  |  |
| --- | --- |
| User | User target |
| HR manager | Add job title. |
| Decide salary amount. |
| HR staff | Enter worker data. |

**Table 2 Employee Actor's Description**

## Network diagram



**Figure 2 Network diagram**

# Use case for attendance

**A close up of a logo

Description automatically generated**

**Figure 3 Use case attendance**

|  |  |
| --- | --- |
| Use Case | Description |
| Enter Fingerprint | When an employee enters his/her fingerprint, the system will save the employee’s check-in, check-out and time. |
| Create attendance report | Process of viewing attendance report for each employee, for each day, month and year. |
| Mark Attendance | Process of HR employee entering the attendance data. |
| Request attendance |  |
| Add user |  |
| Request update attendance |  |

**Table 3 Attendance use case description**

|  |  |
| --- | --- |
| User | User goal |
| Employee | Entering fingerprint. |
| Request attendance . |
| Request update attendance. |
| HR staff | Request attendance. |
| Mark attendance. |
| Create attendance report. |
| Add user. |
| Request update attendance. |

**Table 4 Attendance actor's description**

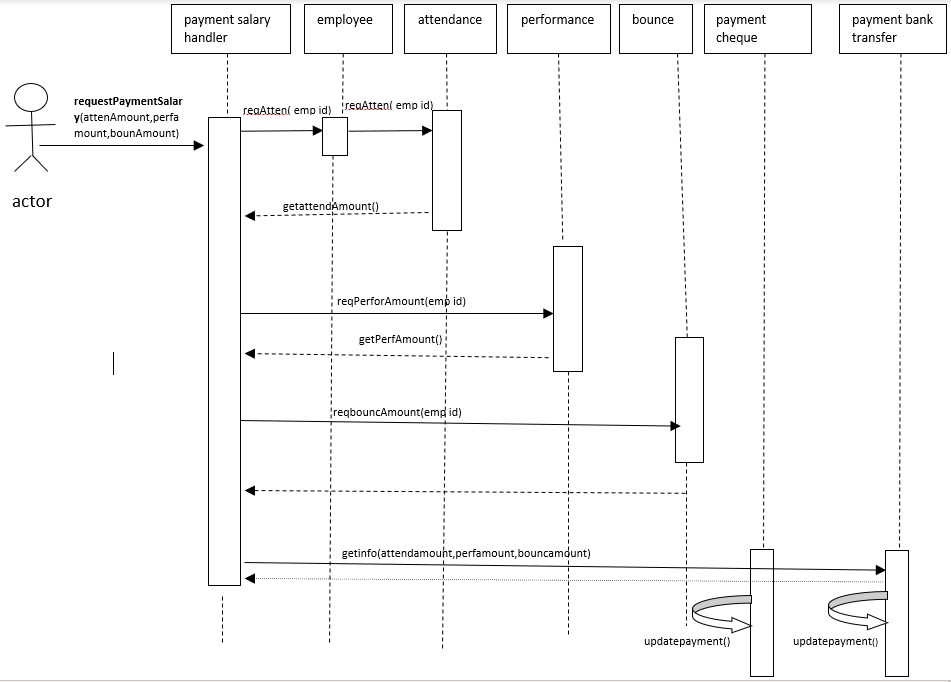
# Use case for salary

**A close up of text on a white background

Description automatically generated**

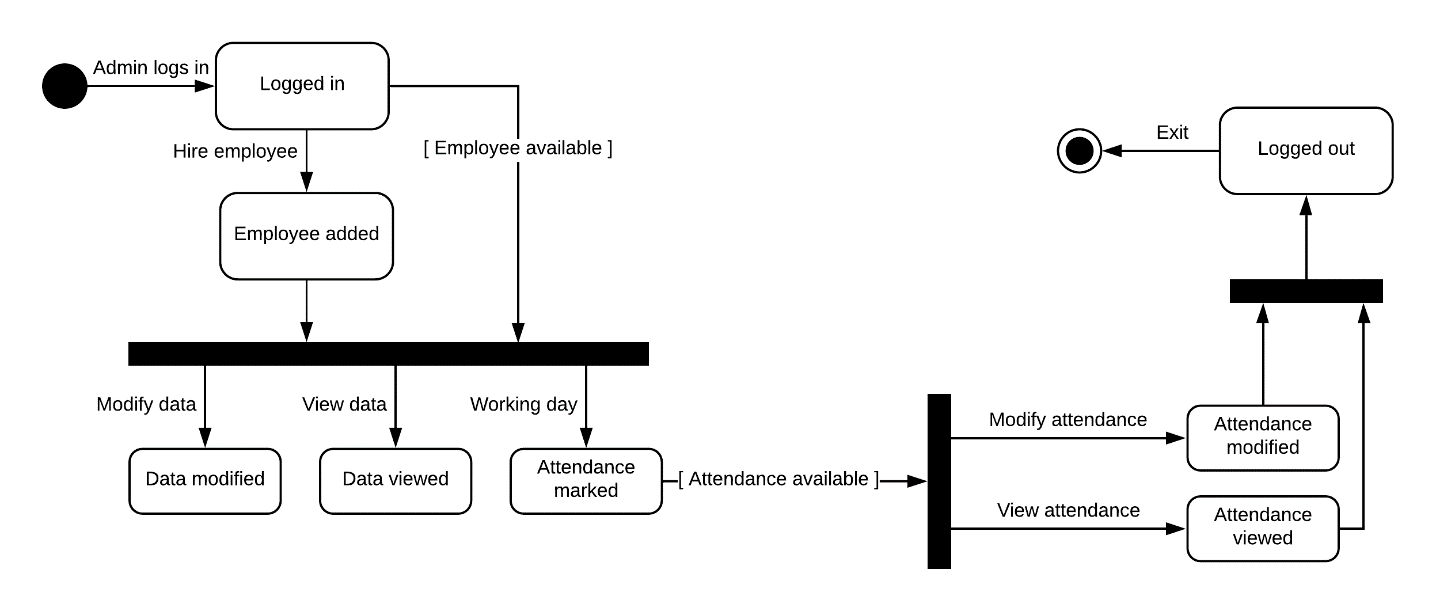
**Figure 4 Use case salary**

# sequence diagram for request payment salary



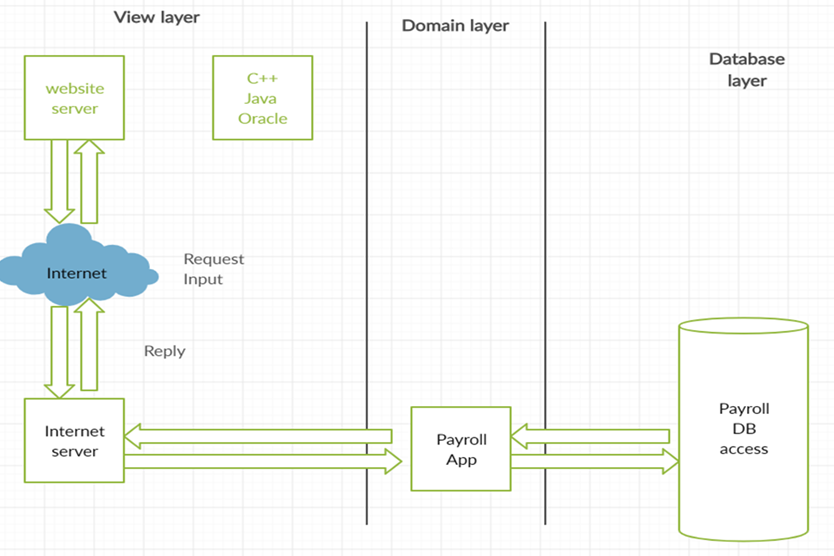
**Figure 5 Sequence state diagram**

# State diagram



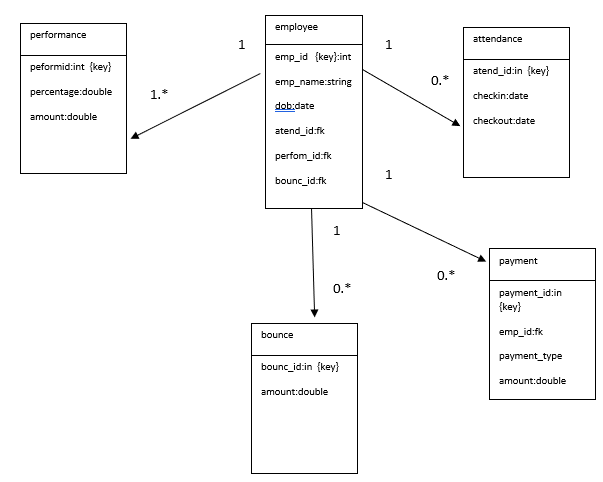
**Figure 6 State diagram**

# Component diagram

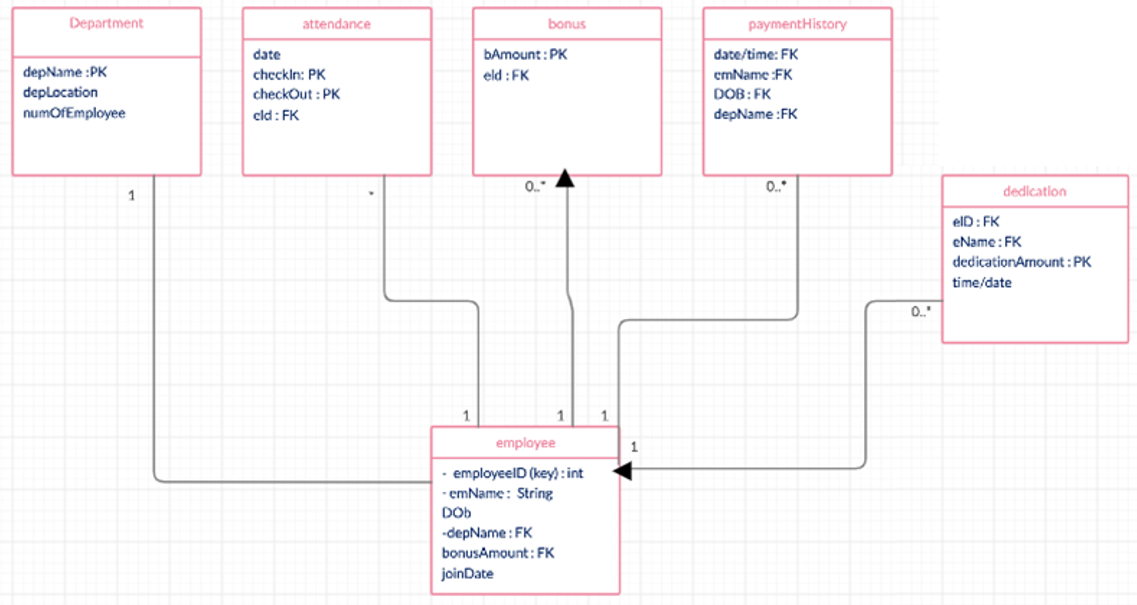


**Figure 7 Component diagram**

# class diagram

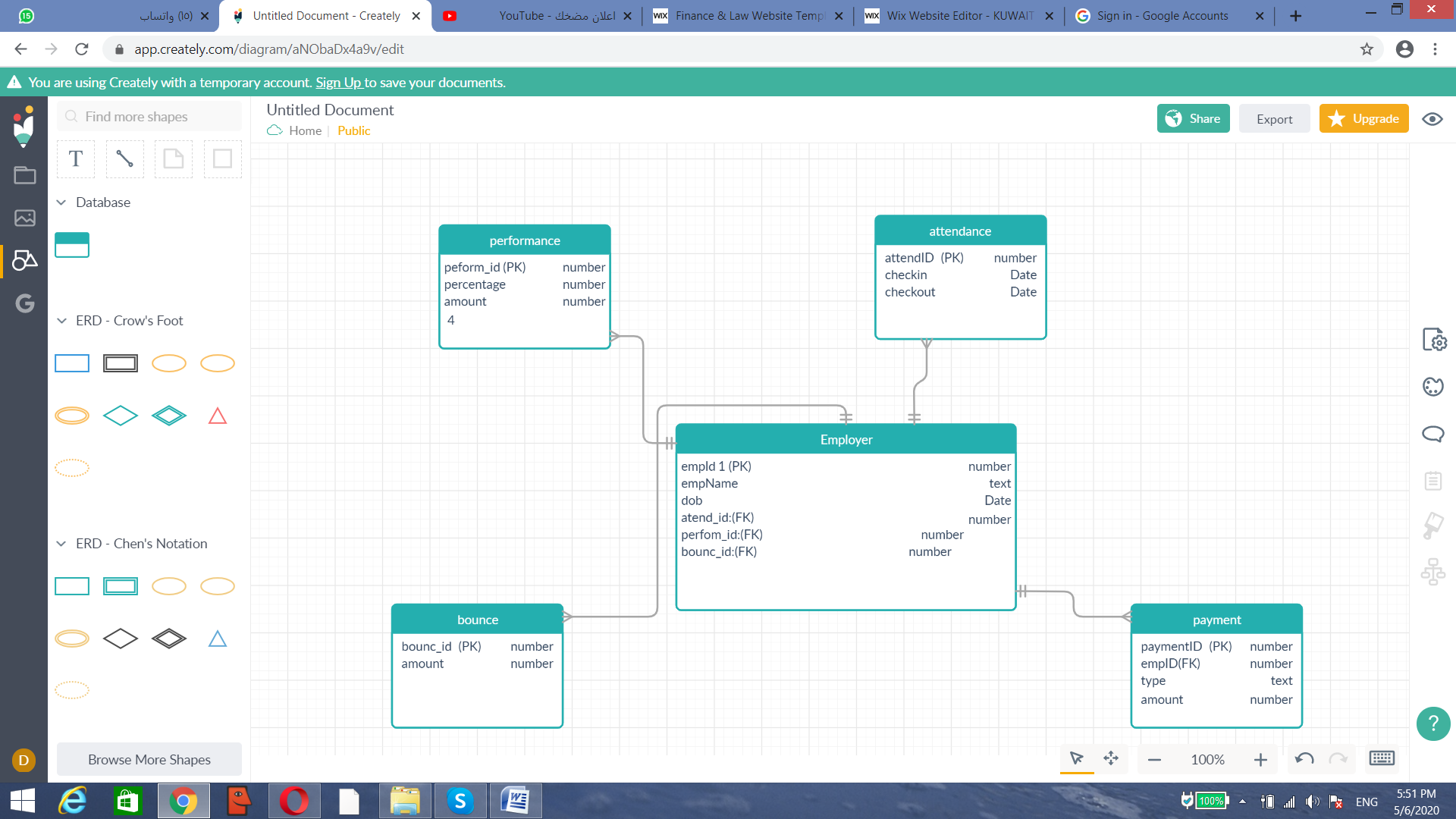


**Figure 8 Class diagram A**



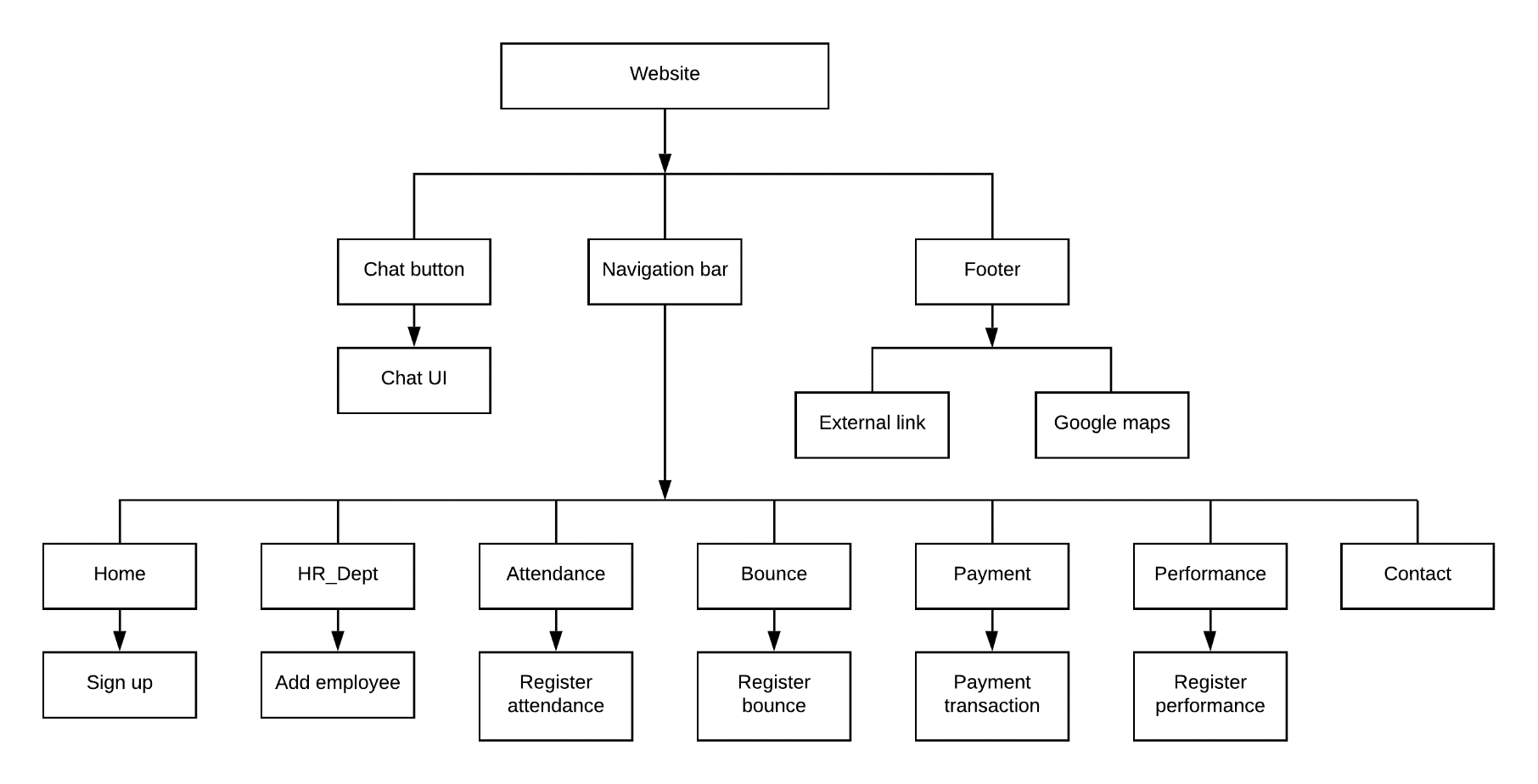
**Figure 9 Class diagram B**

# Entity Relationship diagram

****

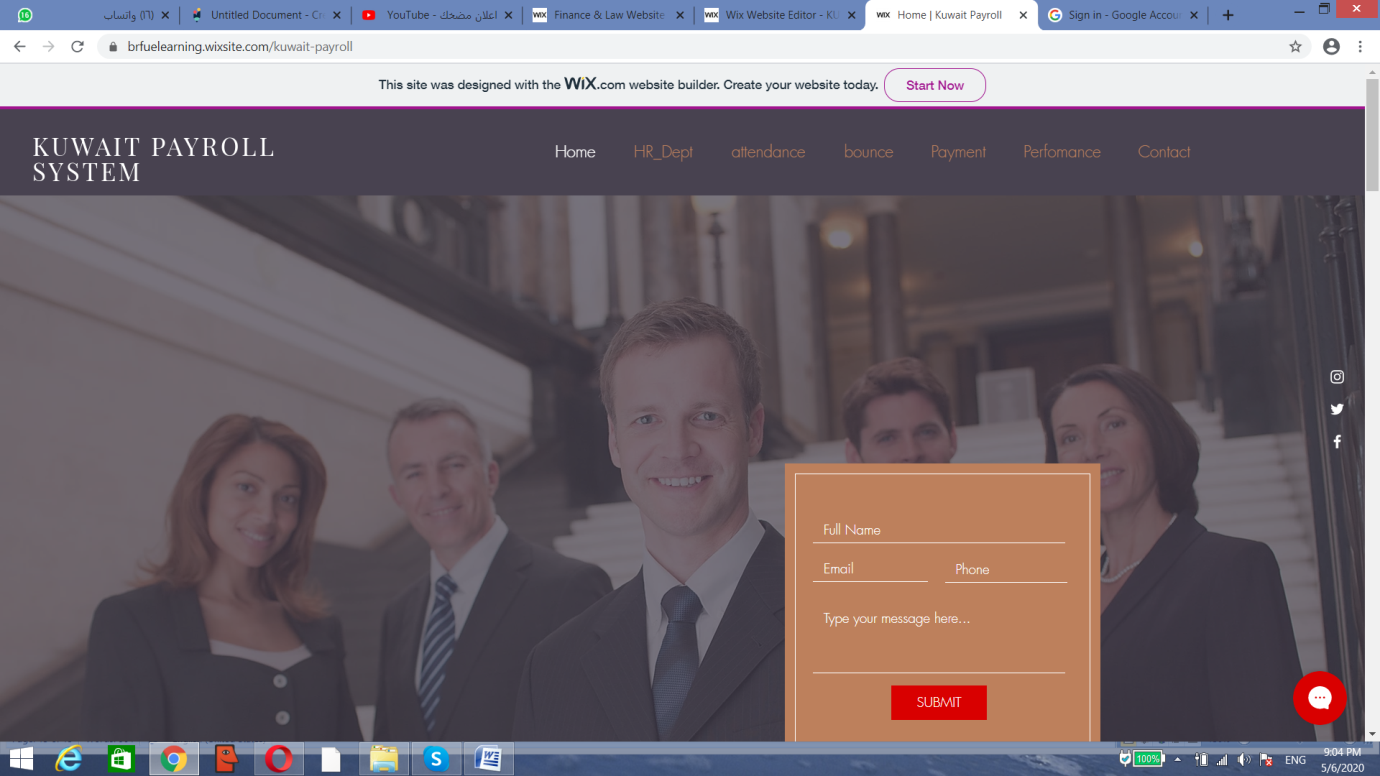
**Figure 10 ERD**

# Navigation diagram

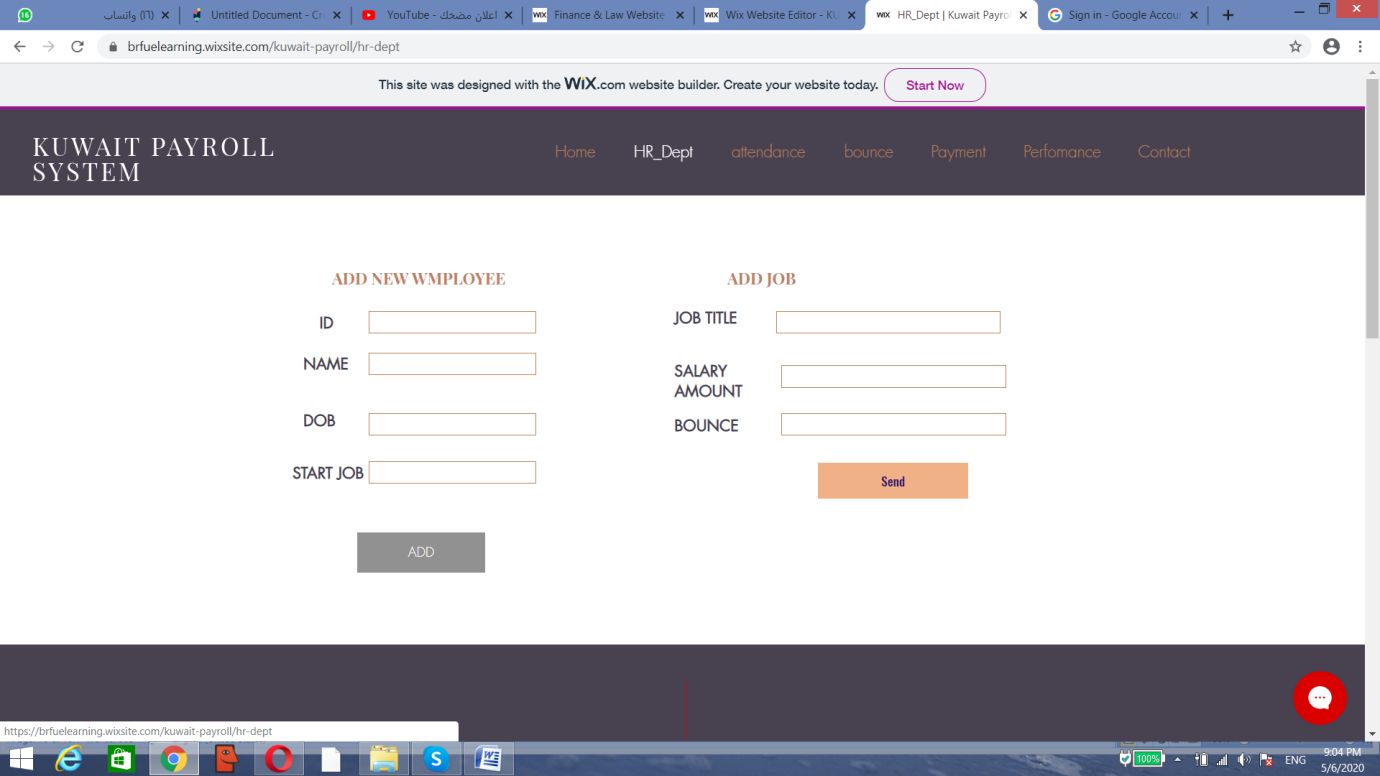


**Figure 11 Navigation diagram**

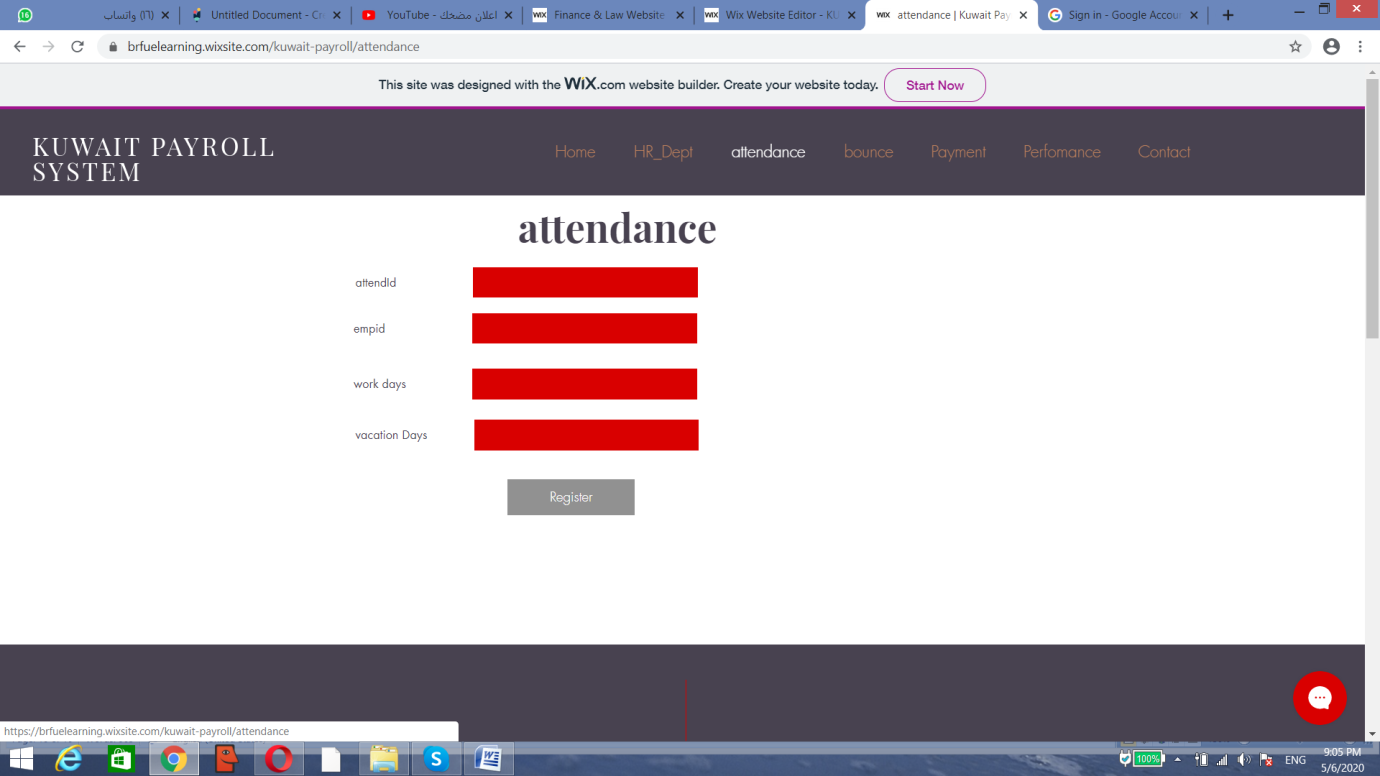
# Website’s frontend



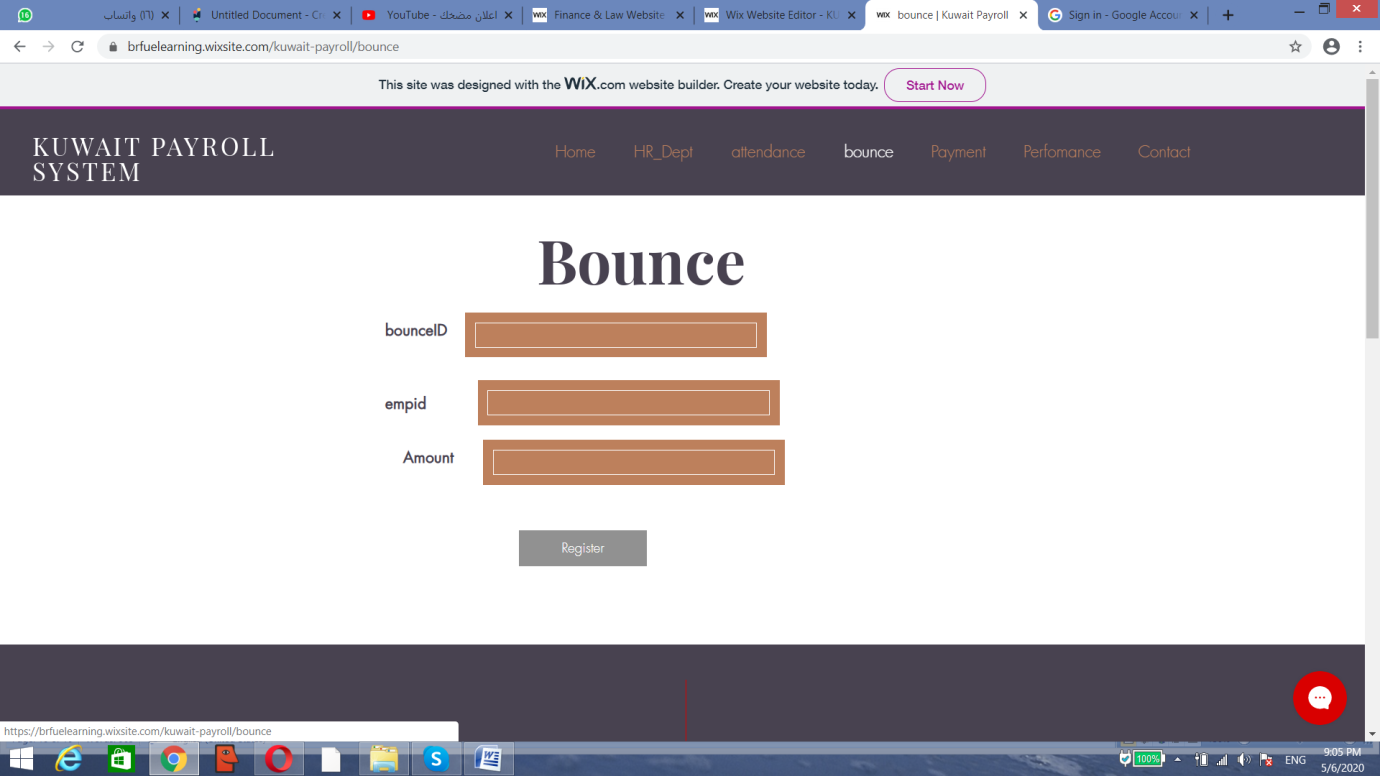
**Figure 12 Image A**



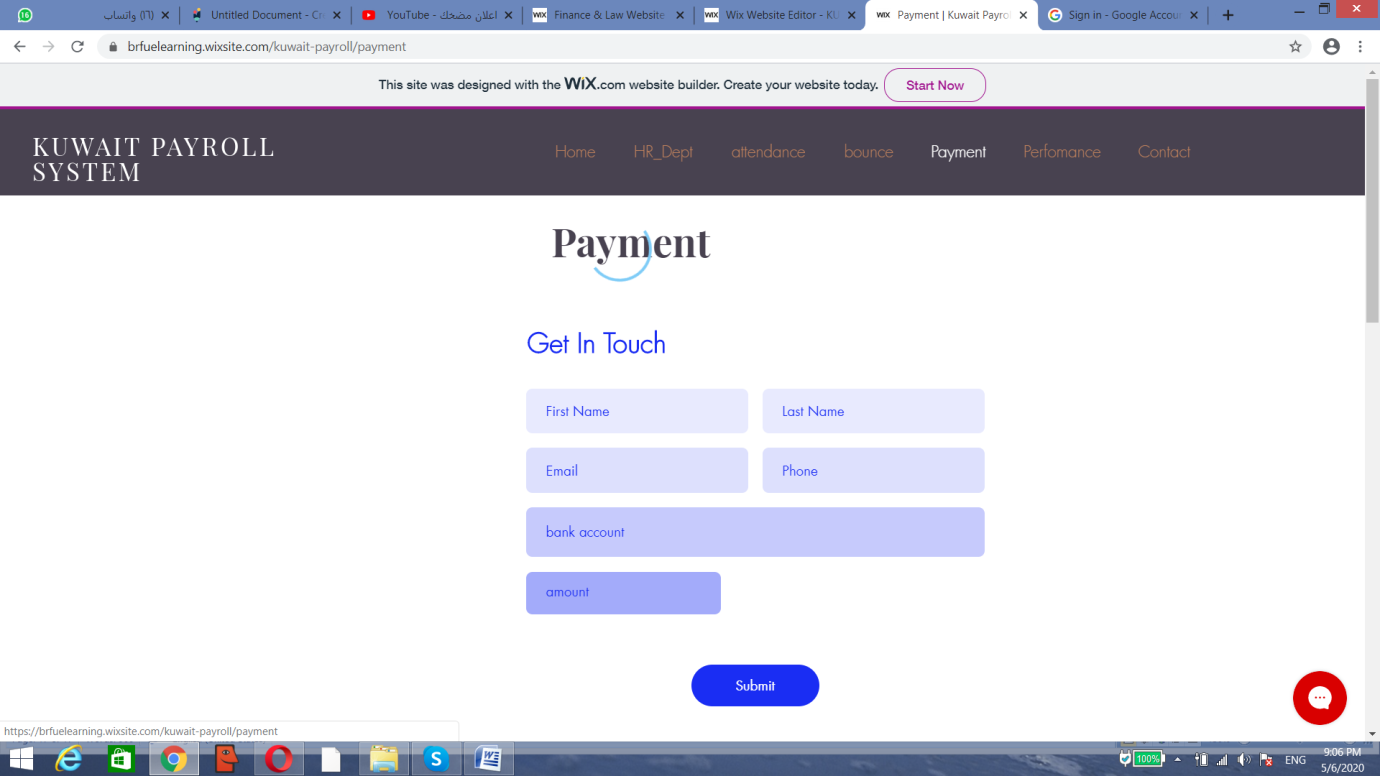
**Figure 13 Image B**



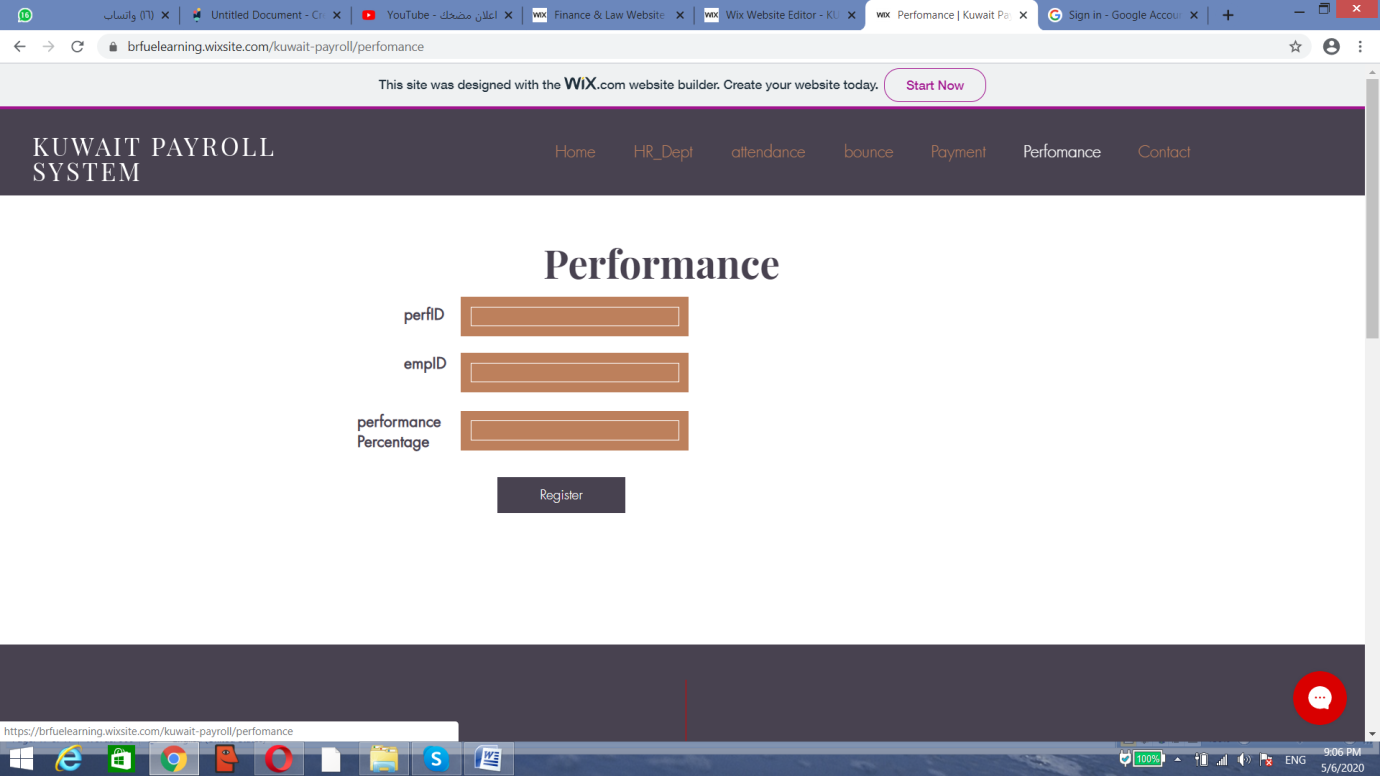
**Figure 14 Image C**



**Figure 15 Image D**

****

**Figure 16 Image E**

****

**Figure 17 Image F**

# WEB LINK

<https://brfuelearning.wixsite.com/kuwait-payroll/perfomance>