

Project Report HY-360

Team Members:

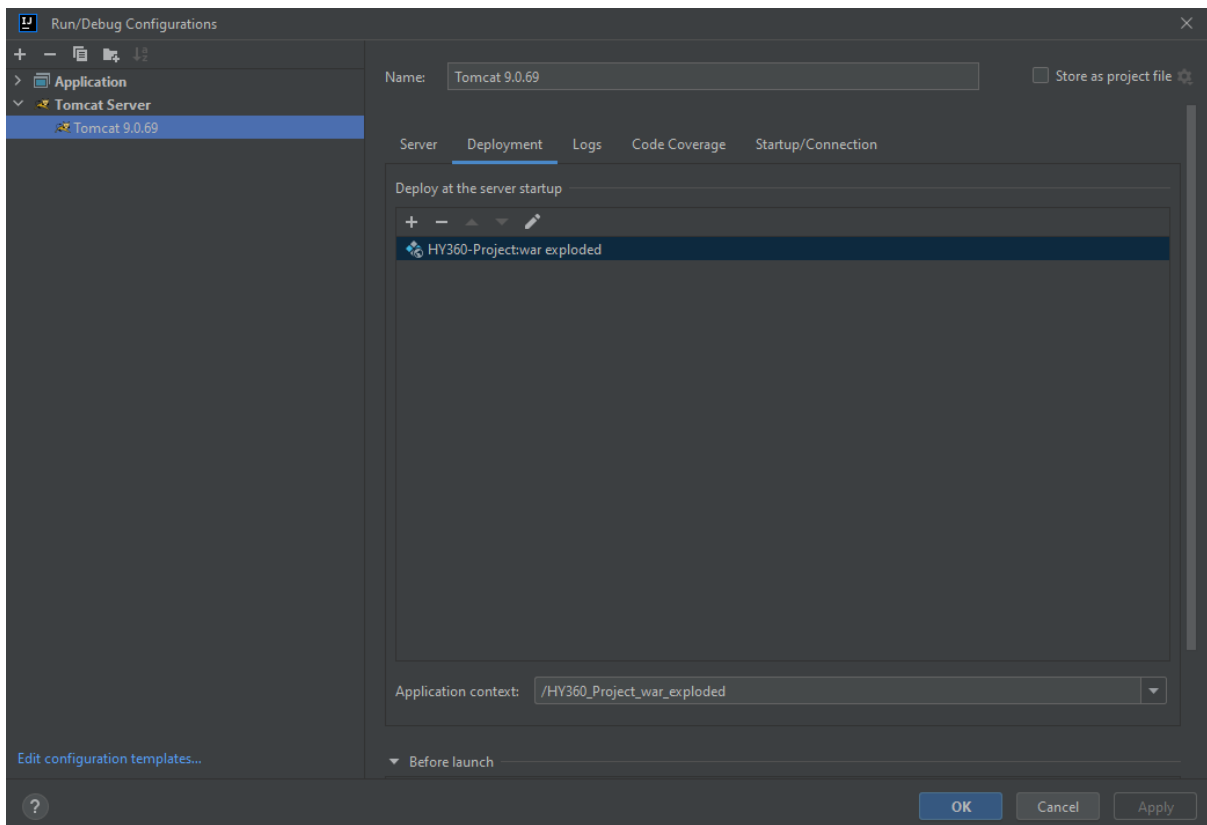
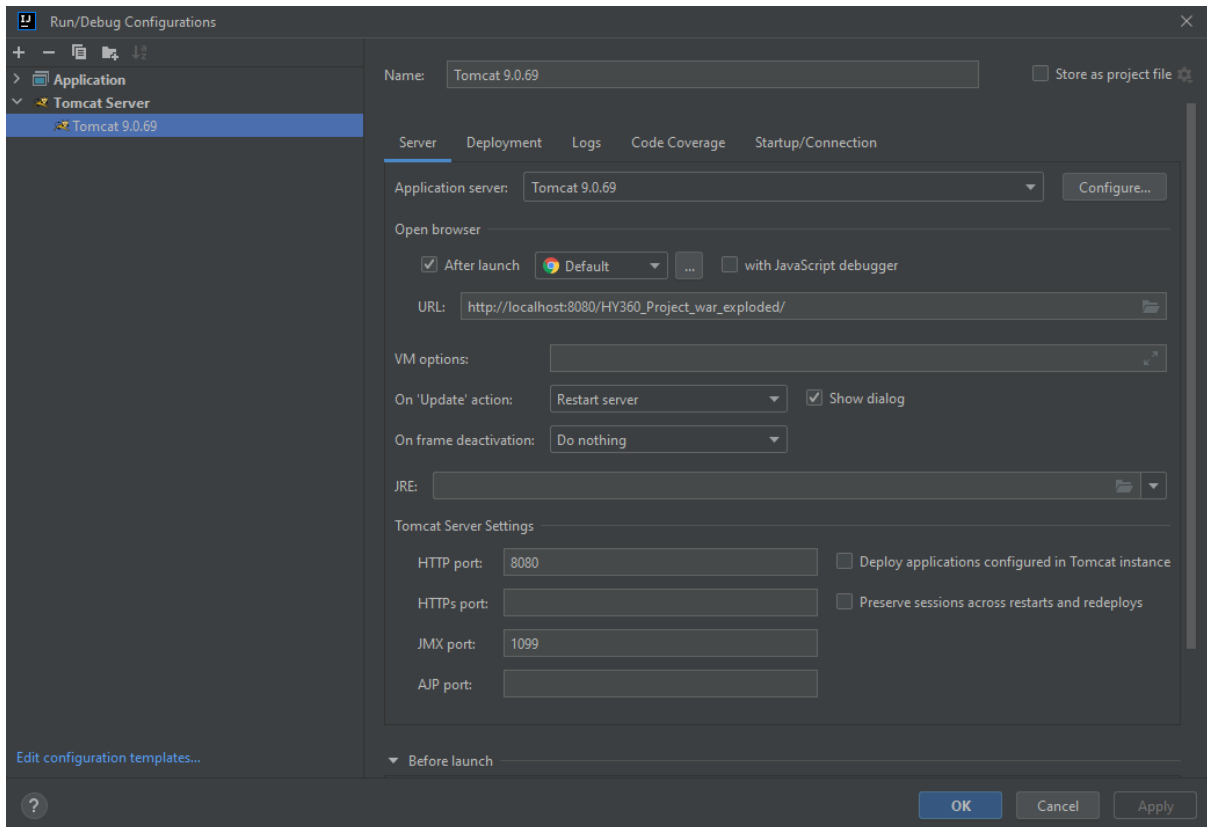
- Antonios Marinos Kostopoulos csd4446
- Michail Samaritakis csd4529
- Emmanouil Tzormpatzakis csd4479

Setup to Test

To setup your computer to test the project in Windows 10 (It is the only OS that the project to) you have to install the following:

- XAMPP : <https://www.apachefriends.org/download.html>
- IntelliJ IDEA Ultimate :
<https://www.jetbrains.com/idea/download/#section=windows>
- Tomcat apache, version: 9.0.69
- JDK Temurin 11 for Windows 10 :
<https://adoptium.net/temurin/releases/?variant=openjdk11>

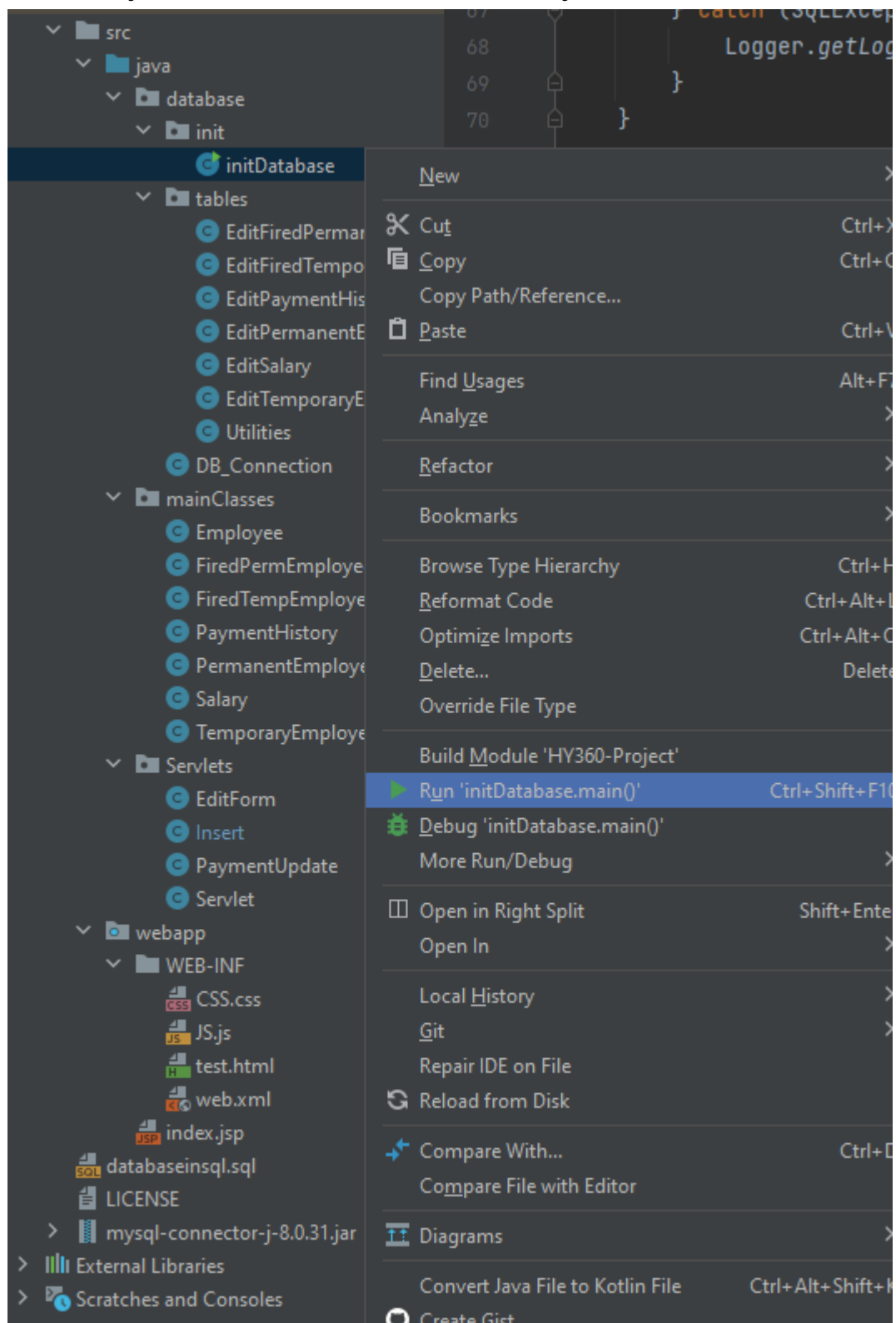
After the downloads above, you have to create a JetBrains account and activate the full licence to install IntelliJ IDEA Ultimate, after the installation is complete install XAMPP and the JDK and extract the Tomcat apache folder. Then insert the JDK into your IntelliJ settings and open the project, now create a configuration with the settings below:



Where the artifact name is the name of the folder which the project is in.

Then startup XAMPP and click start on Apache and MySQL.

Then find the `initDatabase();` main which is located at `src/main/java/database/init/InitDatabase.java` as shown here:



Now choose the Tomcat configuration which you created earlier and then hit run.

The project has now started running and is ready to be tested.

Classes and Visualization of the Implementation

We mainly used servlets(Java) to create the implementation of our project, so that we could connect the graphic interface made in HTML with our database that was created using SQL.

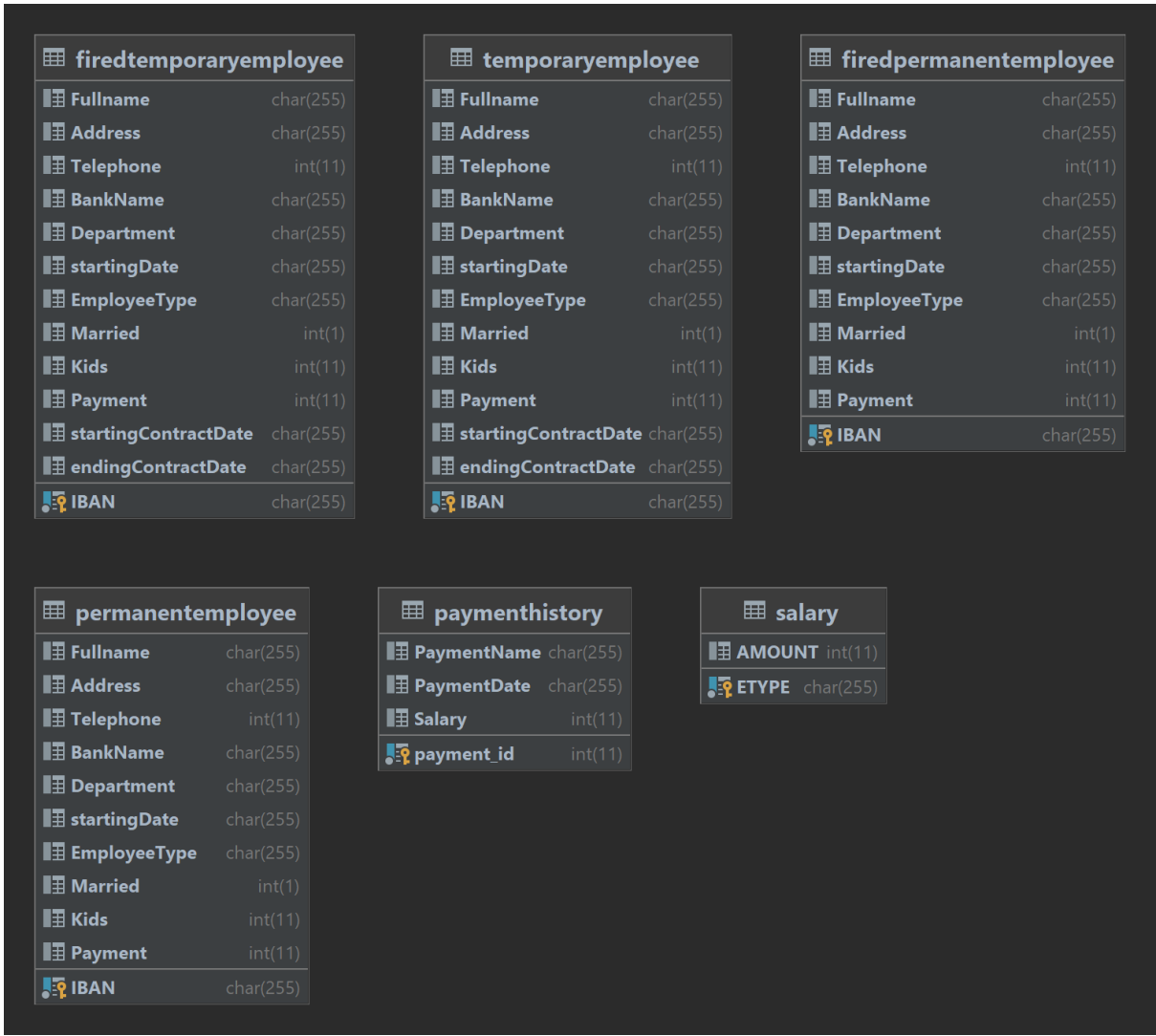
Each servlet has a designated function and is invoked whenever a corresponding button of "submit" type is pressed.

1. In the files : allsqlcommands.sql and databaseinsql.sql you can find all the SQL commands that were used throughout our code.

More specifically in allsqlcommands.sql: are all of the SQL commands. While in databaseinsql.sql: are only the ones that were needed to CREATE the database.

2. Here are the classes and methods that were used for this project:

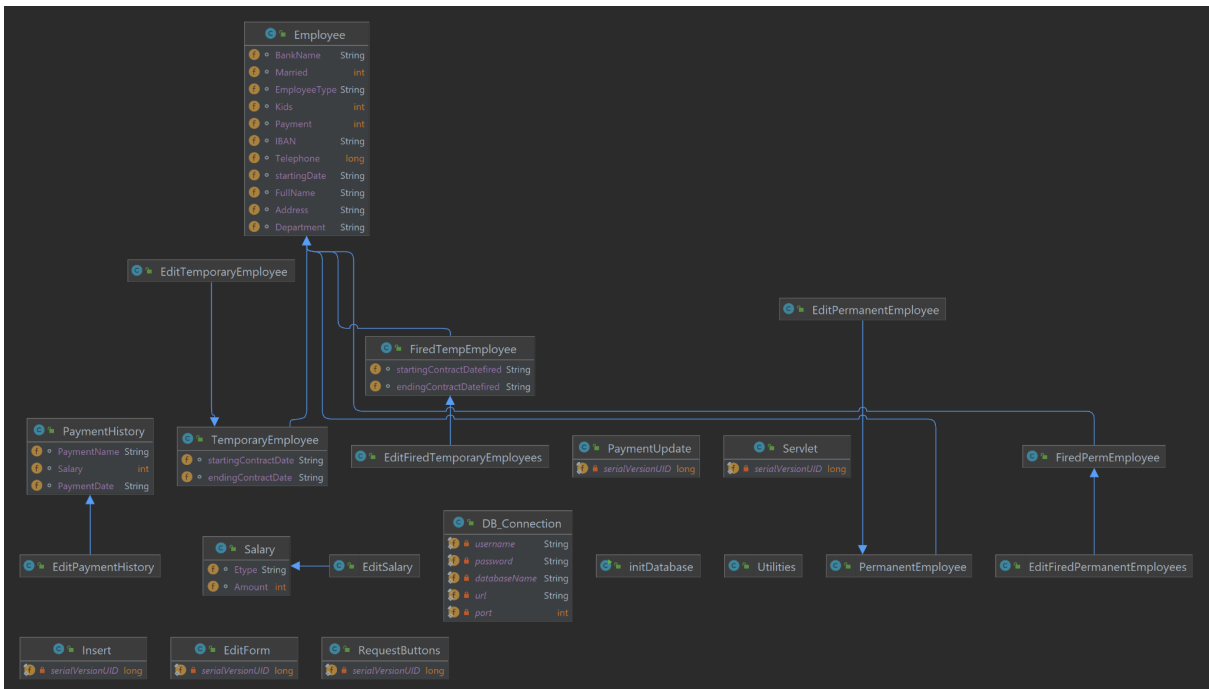
Visualization of our database:

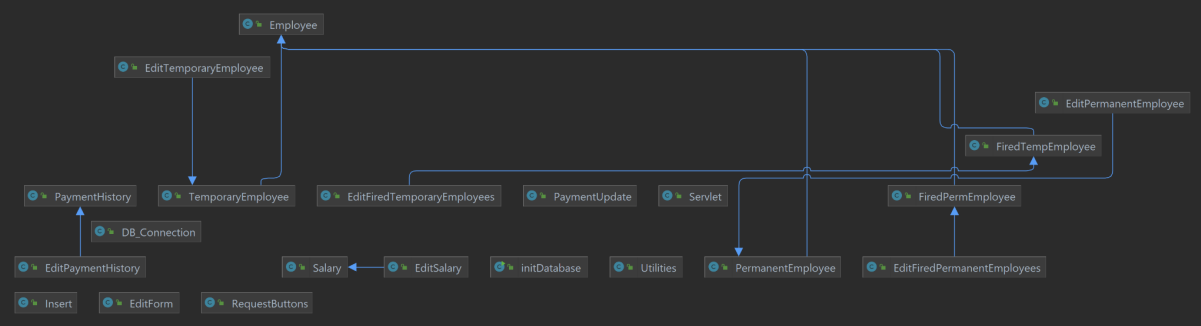


UML Diagram of the classes and methods:

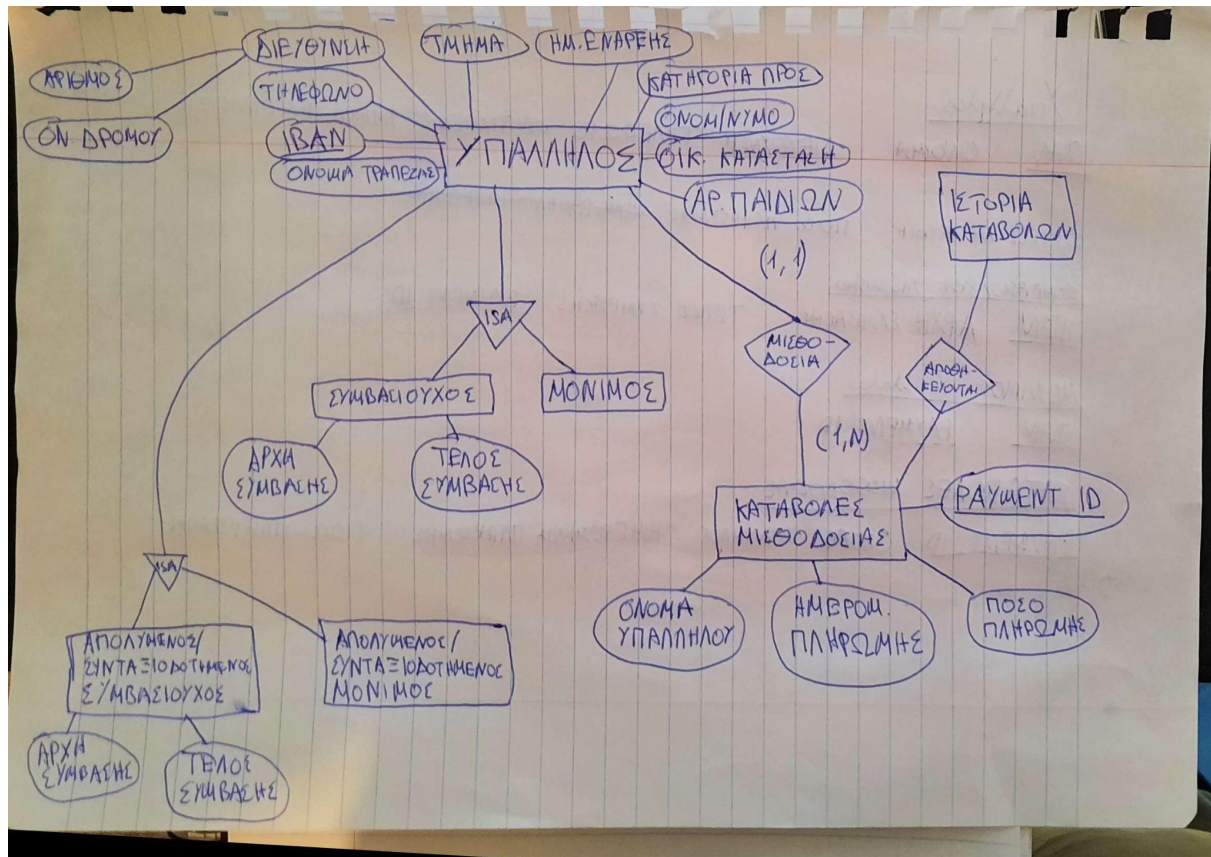


And here are its simpler versions:





Σχεδιάγραμμα Οντοτήτων Σχέσεων E-R



Πίνακες Οντοτήτων Σχέσεων E-R

Υπάλληλος	
IBAN	ΟΝΟΜΑΤΙΜΟ ΔΙΕΥΘΥΝΣΗ ΤΑΞΕΩΣ ΟΝΤΡΑΠΕΣΑΣ ΤΗΛΕΦΩΝΟ ΑΡΧΗ ΕΛΕΓΧΟΥ ΚΑΤ.ΠΡΟΣΩΠΟΥ ΟΙΚΟΓΑΤΑΛΗ ΜΙΣΘΟΣ
Συμβασιμότητα Υπάλλητου	
IBAN	ΑΡΧΗ ΣΥΜΒΑΣΗΣ ΤΕΛΟΣ ΣΥΜΒΑΣΗΣ PAYMENT ID
Μόνο Υπάλλητος	
IBAN	PAYMENT ID
ΚΑΤΑΒΟΛΕΣ ΠΙΣΤΩΣΕΩΣ	
PAYMENT ID	ΟΝΟΜΑΤΙΜΟ ΥΠ. ΗΜΕΡΟΜΗΝΙΑ ΜΗΡΩΜΑΤΟΣ ΠΟΣΟ ΠΛΗΡΩΜΗΣ ΤΥΠΟΣ ΠΟΣΟΥ
Αποζημιώσεις/Ενταξιοδοτήσεις Υπάλλητου	
IBAN	ΑΡΧΗ ΣΥΜΒΑΣΗΣ ΤΕΛΟΣ ΣΥΜΒΑΣΗΣ PAYMENT ID
Αποζημιώσεις/Ενταξιοδοτήσεις Μόνο Υπάλλητου	
IBAN	PAYMENT ID

Σχέσεις Εξαρτησέων και Μορφή 3NF

Functional Dependencies για τον πίνακα Υπάλληλοι:

Τηλέφωνο \rightarrow Όνομα (αφ' ότου ξέρουμε το τηλέφωνο ^{είναι το} ξέρουμε
και το όνομα του υπαλλήλου)
~~Διεύθυνση \rightarrow Όνομα (δίδονται)~~

Functional Dependencies για τον πίνακα Καταθ.Μισθοδοσίας:

Δεν έχει

Ο πίνακας Υπάλληλοι πρέπει να γίνει 1NF αφού δώσουμε ^{αφ'}
στοιχείο

Υπάλληλος

ΙΔΑΝ ΟΝΟΜΑΤΕΠΩΝΥΜΟ (ΑΡΙΘΜΟΣ ΟΝ. ΔΡΟΜΟΥ) ΟΝ. ΤΡΑΠΕΖΑΣ

ΤΜΗΜΑ ΗΜ. ΕΝΑΡΞΗΣ ΚΑΤΗΓΟΡΙΑ ΠΡΟΣ. ΟΙΚΟΓΕΝ. ΚΑΤΑΣΤΑΣΗ

ΜΙΣΘΟΣ

Είναι και 2NF αφού δεν υπάρχει σχέση της μορφής

~~Μη πρωτεύον \rightarrow Μη πρωτεύον~~ Πρωτεύον (βη. κλειδί) \rightarrow Μη πρωτεύον

Δεν είναι 3NF επειδή υπάρχει σχέση της μορφής
Μη πρωτεύον \rightarrow Μη πρωτεύον. ~~αλλά~~

Functional Dependencies για Νίκατα Ποσει:

Δεν έχει τη περιττή σχέση αμοι-νάρχει
ποσο το σελίδι και ένα από τα τη προέλευση ασυνέχεια