CS395

INTERNSHIP REPORT

*Göktuğ Gençkaya - 27888*

**Doping Technology**

Supervisor: Mert MUTLU

Start and End Date: 04.07.2022 – 02.09.2022

Submission Date: 2.10.2022

Sabanci University

Faculty of Engineering and Sciences

Computer Science and Engineering

ABSTRACT

Doping Technology is a company that has been in the market since 2011. They first got into market with their first product ‘Doping Hafıza’ which is an online education platform that offers educational materials in a fun way and facilitates learning my memory techniques. In time they created more products and develop ‘Doping Hafıza’ every year. This year they decided to have a transformation project on this first product so that they it will look like a brand-new platform and work more efficiently with newer technologies. Within the scope of this project, questions that are already been asked in the platform should be connected to learning objectives. Learning objectives are short descriptions of outcomes of each lesson which are published in MEB’s website every year. Moreover, since they have a cumulated and messy database, new database should be created and all content from the old database should be migrated to new database. Last but not least, user interface will be changed and developer technologies, coding programs will all be changed. As an intern, we were given the task to get those learning objectives from the website and put them to the database in proper structure. Furthermore, we also helped with the migration of old database. Other than that, we had learning tasks on clean code, Java, multiservice architecture, SQL and Python. ‘Online Education Transformation’ didn’t finish during the time I had my internship but we had a great start and everything happened just in time. It went really beneficial for me to learn theoretically and practically.

TABLE OF CONTENTS

1. Introduction
2. Company Information
3. Project Background
4. Internship Project
5. Internship Experience
6. Conclusion
7. Recommendations
8. References

1. INTRODUCTION

‘Doping Hafıza’, the first product of Doping Technology is Turkey’s biggest online education platform. With over 500.000 users every year, company offers another way for students to have education and prepare for high school and university exams without going to private courses. They were having a transformation project so that the program will be written again with newer technologies and would adapt more to the developing technology. In this report, there will be information about the company, Doping Techology, its products but mainly Doping Hafıza, why company wanted to start this project in the first place, what they are planning to do for the project, what we did as a part of this project. Through the end of the report, relation of my internship with the school education and my personal thoughts about this internship is mentioned.

2. COMPANY INFORMATION (3 pages maximum)

Doping Hafıza, which first came out in 2011 as a startup project from two entrepreneurs Ali Karakuş and Semih Hakyemez became the biggest online education platform in Turkey. With over 200 employees they transformed the startup project into a technology company, Doping Technology. In order to enhance teaching tools and provide instructional materials to thousands of students, they design modern programs with their unique technology aiming to provide a holistic and systematic experience for its users. Over time, they did not only limit themselves with their successful product Doping Hafıza. 3 other products are introduced into the market. These are ‘Şimdi Anladım’, ‘Çözücü’ and ’KoçumYanımda’. All of them are mobile applications with different functionalities aiming to help students on their education. ‘Şimdi Anladım’ connects students with qualified teachers who help them understand the subjects that they have difficulty with. Moreover ‘Çözücü’ application helps delivering solutions to questions that students cannot solve on their own through video narrative. Lastly, ‘Koçum Yanımda’ provides guidance and real time support through live conversations between students and counselors on the system.

Full Title: Doping Technology

Adress: Maslak Mah. AOS 55 Sok. 42 Maslak B Blok No:4/588 Sarıyer/İstanbul

https://dopingtech.net/

3. PROJECT BACKGROUND

3.1 Department information (1 page maximum)

Technology department is divided into two compartments. On one side, software engineers responsible with all 4 products of the company, development tools used within the company. On the other side, game developers and animators create the animated content for Doping Hafıza videos, newly launched Dopiverse educational game and etc. I worked as a back end developer intern in the technology department. There were full stack developers, front end developers and back end developers, data engineers in my department. Other than the software engineers, there were two business analysts who were responsible of communicating with product department which collect requests from users. They were preparing the diagrams, pseudocodes, tasks for engineers

Mert Mutlu – Software Engineer (Supervisor) - mert.mutlu@dopingtech.net

Remzi Alperen Can – Head of Technology - alperen.can@dopingtech.net

Ceren Özkan – Business Analyst - ceren.ozkan@dopingtech.net

Naz Başaran – Senior Business Analyst - nazburcu.basaran@dopingtech.net

Çağrı Duru – Data Engineering Manager - cagri.duru@dopingtech.net

Beril Talay – Data Scientist - beril.talay@dopingtech.net

3.2 Status of the project and/or the problem at the beginning (1 page max, excluding figures)

They were having a transformation project on ‘Doping Hafıza’. Aim was to recreate the product using newer technologies. Although the old program was built on monolithic architecture, the new design was on multiservice architecture, so that the product can benefit from all the advantages of that architecture. In case of a need of any update in one of the microservices, then they would just need to redeploy that microservice. Since these microservices are self-contained, their start-up and deployment times are relatively less. Moreover, it is very easy for a new developer to onboard the project as they need to understand only a particular microservice providing the functionality they will be working. Another great advantage of it is that each microservice can use different technology based on its business requirements. Last but not least, if any microservices goes down as a result of some bug, then it doesn’t affect other microservices. This way whole system remains intact and other functionalities are provided to users.

As every technology firm, innovation and the tracking of new technologies are a must thing. Before this ‘Online Education Transformation Project’ that I participated in, all tasks were just to save the day. There were no proper records on what information is stored where in the database. Since it created a real mess in the database called ‘onlinee’, new database should be created and migration should be done from old database to new database.

In the department they were using a project tracking software Jira. With the help of this program they create tasks before starting a project, marking them as ‘currently working on it’ or ‘done’. Also we were having daily meetings at 9.30 AM every morning to discuss what we have accomplished yesterday and what we’ll be doing that day. Moreover, every Thursday, Grooming meetings were arranged. At those meetings, analysts and engineers grade the upcoming tasks according to how much time they may consume during the completion of those. Furthermore, department was using Slack as a messenger application and instantaneous audio meetings, Discord for daily meetings, Zoom for the scheduled meetings.

3.3 Motivation and/or problem definition (1 page maximum)

Since 2011, almost all software engineers of Doping technology were working on the main product ‘Doping Hafıza’. Since it was the first and most successful product of the company, it was used to called as ‘flagship’ among employees. Although there has been a plenty of updates in Doping Hafıza, it started to get a little bit old in many ways.

First of all, user interface for different pages were pretty basic compared to new apps and programs. Technology is improving at the speed of light and company had to evolve at that speed. Instead of complicated and basic UI design, they aim to change its design to a more sophisticated and simpler look. Secondly, tutors working for the company came up with questions throughout the company history. Although many questions cumulated over the years, some questions were written more than once causing users to see the same questions multiple times. Through collecting feedback from clients, business analysts were able to identify the problem. They had to clean up the messy and enormous database. Singularization of questions were needed in the newer version. For these reasons, company started a transformation project. Instead of updating the existing code and existing online education software, they started to re-code the whole software for good.

3.4 Related literature

Before we even get started to help senior engineers with coding, they highlight the importance of clean code. Since there were 30 engineers working on DH, it was very crucial for our code to be readable and understandable. They gave us the password for their Udemy account. They requested to have the Clean Code lecture of Akın Kaldıroğlu who is a consultant and trainer for SWE, OOP & FP, Java & Go. It was almost a 15 hour lecture including the topics such as the principles of clean code, Object-Oriented Programming, Introduction to design patterns, defensive programming and test-driven development.

In the upcoming days, they wanted us to understand the basics of monolithic architecture and multiservice architecture. I prepared a Word document, putting together the knowledge I gain from my research for other interns.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

4. INTERNSHIP PROJECT

4.1 Project objective

Every year MEB, ministry of education, publishes a curriculum for each lesson and for every grade. These curriculum documents involves a collection of learning objectives. Learning objectives are couple of sentences that summarizes the information that has to be transferred to students in an understandable form. For example, for Math lesson, one of the most important 9th grade topic is divisibility rules. One learning objective of this topic is “Students are able to solve problems about divisibility rules with integers.” Curriculum cannot involve any more information than these objectives. Thus, any question that is asked in the examination should be able to solve-able by this cumulative information in learning objectives.



One of the main aims of the transformation project was to capture these learning objectives and match each question to one or more objective. This way, system will be able to understand the inadequacy of students. To elaborate more on that, if a specific student had a mistake on a question within the system, system would learn that that question lacks the information contained in learning objective that is connected to that question. By establishing the missing knowledge of student, system could offer topic description videos or direct user to study more on those topics. After that system should be able to ask different questions that are also connected to the same LO that user lacks. This way, that LO would marked as learned.

Company had an enormous database having different questions, different solution videos, users information and many more. We have to carry all the old information from old database to newly created database. In other words, there was a migration from old database to new database. Furthermore, some rows of data can be duplicated, some questions could be entered multiple times to the system. So one another task was to clean the already existing data.

4.2 My responsibilities

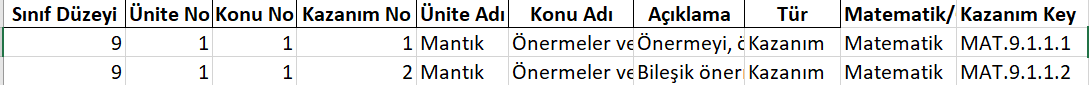
In the learning objective part, company need a different excel table for all lessons. Curriculum document were uploaded to MEB’s website. I had to write a program to download the document from the website, capture only the learning objective parts and put these objectives to a fixed structure in a dataframe object. Then I would be able to output this dataframe object as an Excel file. Learning objectives were in kind of special form such as the picture below. In this form, first number gives information about the grade of the LO, second number is the unit number, third number is the topic number and the last number is the LO number. These numbers were really important since they give knowledge of grade, unit and topic of every learning objective.



Moreover, unit names and topic names were also listed in these LO documents. They were slightly different than objectives. Unit names only have two numbers in front of the string while topic names have 3 numbers and objectives have 4. This means that, program should detect the count of numbers in front of the string description and compartmentalize according to that count as a learning objective, topic or a unit.



Moreover, chief of technology requested one more column that includes the original number of the learning objective with first three letters of the name of the lesson. An example is shown below.



After getting excel tables as an output of my python program and having some more arrangements on excel, these tables had to be imported to newly created database of company. For this task, they wanted us to work with SQL queries for some time, try to learn more about existing database. For the purpose of practicing and having experience on writing SQL queries, they wanted us to get a certification of SQL from the HackerRank website. After training with SQL statements through that website, we were able to have a test within the website and get a HackkerRank certificate.

metin içeren bir resim

Açıklama otomatik olarak oluşturuldu

Furthermore, there were too many different tables containing different information about the product Doping Hafıza, tables that connects solution videos with questions, questions with answers and many more. Additionally, there were different tables for different lessons, which again makes the database even more messy. Starting with SQL statements, they wanted us to merge all questions and solution videos of different lessons on one table separately. Having that table exported as csv, we had to import it to new database.

Last but not least, through the end of my summer internship, Data Scientist Beril whom I worked with for the Migration task wanted me to learn how to have a DB connection through Python. With this connection, SQL scripts can be executed with Python codes. With the help of this, it was easier to get an output of an SQL script and import that table to a brand new database.

4.3 Methodology / tools

For the purpose of capturing the objectives, I thought that Python would solve this problem easier than any other programming language. As we already used in the university lectures, I prefer t work on Google Colab to run python programs. I only needed PyPDF2 and pandas libraries for all of them. For some LO documents I also use re library. Additionally, to connect Python with PostGres database, I had to use psycop2 library functions.

Our data engineering manager Çağrı created a postgres database as a new database of company. All the information that old database contain had to be migrated to this new database. To get the data from old database and create a new database and have some amount of manipulation on imported data, we had to write SQL queries.

On the software part, they were using Java most of the time. None of the interns had great experience with Java coding. Thus, they wanted us to learn Java independently. At the times where we don’t have any task assigned to us, we try to learn Java using different websites such as HackerRank, Codeacademy etc. As the end of the intern was approaching, we weren’t able to have a hands on experience with Java. Our supervisor hadn’t assign any tasks on Java.

4.4 Expected outcome and deliverables

First of all, company need a full list of learning objectives within a table as they will be using all these LO’s by connecting them to questions, solution videos and lecture videos. I wrote different python programs for different LO documents. LO documents were published on MEB’s website. Although some of them were really similar, such as chemistry, biology and physics; some of them were really different. Thus, code should need little modification. At the end of the program, my code outputs an xlsx file with parsed learning objectives. There were ‘Sınıf Düzeyi’, ‘Ünite No’, ‘Konu No’, ‘Kazanım No’, ‘Ünite Adı’, ‘Konu Adı’, ‘Açıklama’, ‘Tür’, ‘Kazanım Key’ columns on the table.

After getting all these tables in desired format, we wrote SQL scripts to create new database. This new ‘mufredat’ database was composed of 4 different tables. ‘dn\_kazanim’ table, which contains all the objectives with their related grade, topic name and unit name information; ‘unite’ table with primary key id, contains unit name information with related lesson information; ‘konu’ table contains the topics of all the lessons that have topics (Some lessons just have units but don’t have topics under units). Lastly, ‘kazanim’ table contains all learning objectives with the related ‘konu\_id’ and ‘unit\_id’ which are used to connect learning objectives with topics and units. ‘id’ column of ‘unite’ table was a foreign key to ‘konu’ table’s unit\_id column and ‘id’ column of ‘konu’ table was a foreign key to konu\_id of ‘kazanım’ table.

Moreover, there was a ‘Migration’ task that need to be handled. This migration task includes singularization of content and migration of all the data to new database in a much neater structure. To clear up the old database, we wrote SQL queries that aims to collect all the information that is scattered through different tables in the old database. Moreover, we had to do the same task for different content such as solution videos and lecture videos etc.

4.5 Details

First task was to derive the learning objectives from the PDF file that is downloaded from the MEB ‘s website. MEB publishes these curriculum files once in a year upon the completion of each education year. First of all, we need to capture the part of pdf file where learning objectives are present. Then we would be able to get those pages on a page object and extract pages’ text to an input string object. Below, there’s a sample code for the 9th grade curriculum of Geography lesson.

import PyPDF2

# creating a pdf file object

pdfFileObj = open('kazanimm.pdf', 'rb')

# creating a pdf reader object

pdfReader = PyPDF2.PdfFileReader(pdfFileObj)

# printing number of pages in pdf file

print("Total number of pages: " + str(pdfReader.numPages))

# creating a page object

pstart\_9 = int(input("Start Page for 9th Grade: "))

pend\_9 = int(input("End Page for 9th Grade: "))

pstart = int(input("Start Page for rest: "))

pend = int(input("End Page for rest: "))

ptotal = pend - pstart + 1

ptotal\_9 = pend\_9 - pstart\_9 + 1

psubbed = pstart - 1

pageObj = [pdfReader.getPage(psubbed)]

psubbed\_9 = pstart\_9 - 1

pageObj\_9 = [pdfReader.getPage(psubbed\_9)]

for i in range(1, ptotal):

    pageObj.append(pdfReader.getPage(psubbed + i))

for i in range(1, ptotal\_9):

    pageObj\_9.append(pdfReader.getPage(psubbed\_9 + i))

# extracting text from page

a = pageObj[0].extractText()

s = str(pstart)

for x in range(1,ptotal):

    s = str(pstart)

    a= a.replace(s,"")

    a = a+pageObj[x].extractText()

    a = a.replace("\n","")

    pstart += 1

s = str(pstart)

a = a.replace(s,"")

a\_9 = pageObj\_9[0].extracttext()

s\_9 = str(pstart\_9)

for x in range(1,ptotal\_9):

    s\_9 = str(pstart\_9)

    a\_9= a\_9.replace(s\_9,"")

    a\_9 = a\_9+pageObj\_9[x].extractText()

    a\_9 = a\_9.replace("\n","")

    pstart\_9 += 1

s\_9 = str(pstart\_9)

a\_9 = a\_9.replace(s\_9,"")

input\_str9 = a\_9

input\_str = a

# closing the pdf file object

pdfFileObj.close()

After getting the objectives as a string object, we had to parse this string into different columns of information. What is expected from us was to divide the learning objectives to grade, unit number, topic number (if there were any), objective number and the description after them. Numbers were separated with a dot and there were keywords parts which contains unnecessary information. To be able to parse this file, keywords parts should be ignored and all these numbers and description after that can be obtained by splitting according to dot. Below code will output a list that contains grade number, unit number, topic number, objective number description consecutively. Important thing about this output is that if there is no topic number and objective number, this means that this description tells the name of that specific unit with that grade number and unit number. Congruently, if there is no objective number (equivalent to having just 3 numbers before the upcoming description string element in the list), this means that description tells the name of that topic.

listt =input\_str9.split(".")

numb = len(listt[0])

removeword= "Anahtar Kavramlar"

i=0

while (i<len(listt)):

  if (any(map(str.isdigit, listt[i])) and not (listt[i].isdigit())): #any(map()) checks if there is any number in a string

    currentword=listt[i]

    lengthh= len(currentword)

    numm= currentword[lengthh-numb:]

    alpp= currentword[0:(-1\*numb)]

    listt.remove(currentword)

    listt.insert(i,alpp)

    listt.insert(i+1,numm)

  i+=1

for i in range(0,len(listt)):

  if removeword in listt[i]:

    indnum = listt[i].find(removeword)

    newstring= listt[i][0:indnum]

    listt[i]=newstring

lenn= len(listt)-1

for i in listt:

  if is\_abcd(i): #check if it is a "a" or "b" or such

    j= listt.index(i)

    while (j<len(listt)) and (not is\_number(listt[j])):

      listt.pop(j)

for i in range(0,len(listt)):

  x = re.findall(r"\b[a-z][)]", listt[i])

  if x:

    indx = listt[i].find(")")

    listt[i]=listt[i][:indx-1]

for i in range(0,len(listt)):

  if ((i<len(listt)-1) and (not is\_number(listt[i])) and (not is\_number(listt[i+1]))): #check if there is two consecutive learning objectives, if there is, they are probably part of the same objective, concatenate

    newlist =[listt[i], listt[i+1]]

    newstr= ".".join(newlist)

    listt.insert(i,newstr)

    listt.pop(i+1)

    listt.pop(i+1)

i = 0

while (i<len(listt)): #remove all empty elements of the listt

  if ((listt[i].isspace()) or (listt[i]=="") or (listt[i]==" ")):

    listt.pop(i)

  else:

    i+=1

for i in range(0,len(listt)):

  if ( (i<len(listt)-1) and(not is\_number(listt[i])) and (not is\_number(listt[i+1]))):

    listt.pop(i+1)

Moreover, this list has to be divided into different columns such as grade, unit number, topic number and objective number and description. I choose to have a dictionary keeping 5 different lists for 5 different column. After that, this dictionary object have to transform into a dataframe object. As the desired output of this task was an excel file, I then convert this dataframe object to excel file and output that file. Below code was the first try of it for the Geography lesson. Since Geography learning objectives file doesn’t include topic number, I divide the file to 4 columns.

import pandas as pd

tabledata = {'Sınıf Düzeyi': [], 'Ünite No': [],'Kazanım No': [], 'Açıklama': [], 'Tür':[]}

newlistt = []

newlisttlength=0

for elem in actuallist:

  newlistt.append(elem)

  newlisttlength+=1

  if (not is\_number(elem)) :

    tabledata['Sınıf Düzeyi'].append(newlistt[0])

    tabledata['Ünite No'].append(newlistt[1])

    #if clauses

    if (newlisttlength ==3):

      tabledata['Kazanım No'].append(None)

      tabledata['Açıklama'].append(newlistt[2])

      tabledata['Tür'].append("Ünite İsmi")

    if (newlisttlength ==4):

      tabledata['Kazanım No'].append(newlistt[2])

      tabledata['Açıklama'].append(newlistt[3])

      tabledata['Tür'].append("Kazanım")

    #if clauses end

    newlistt.clear()

    newlisttlength=0

df = pd.DataFrame(tabledata)

df.to\_excel('cografya\_deneme.xlsx', index=False)

At this point, there were some little modifications among the codes for different lessons. However, outputs of these files were successfully obtained in desired format. At that stage of the project our CTO asked about few more things with the structure of learning objectives excel tables. Instead of having unit names and topic names in description column, he wants us to have just learning objective descriptions in the description part and unit names, topic names were supposed to be new columns. Moreover, he wants us to add a column to specify the lesson name since there were too many different files for every lesson. Last of all, there were a new learning objective key column which is was like a summary of a row. It was composed of first three letters of the lesson name, grade number, unit number, topic number (if any) and objective number consecutively separated with a dot. All of these modifications on the excel file was handled using Excel functions and utilities. Below there is a picture of mathematics excel table format for learning objectives.

tablo içeren bir resim

Açıklama otomatik olarak oluşturuldu

For the need of importing these files to the database, we also converted them to csv files manually from different helper websites. At this point, we were supposed to import these excel files to database. Our Data Engineering Manager creates a new database in front of us. As mentioned in other sections above, there were 4 different tables. ‘dn\_kazanim’ table contains the basic learning objectives as in the same format with the excel files. ‘unite’ and ‘konu’ tables were collecting the unit names and topic names with their matching id’s. Lastly, ‘kazanım’ table contains the collection of all learning objectives. Below there are SQL codes for this purpose.

**CREATE** **SCHEMA** mufredat **AUTHORIZATION** dopi\_dwh\_admin;

**CREATE** **TABLE** mufredat.dn\_kazanim (

id **int4** **NOT** **NULL** **GENERATED** **ALWAYS** **AS** **IDENTITY**,

kazanim\_key **varchar**(50) **NOT** **NULL**,

sinif **varchar**(10) **NOT** **NULL**,

ders **varchar**(100) **NOT** **NULL**,

unite\_no **int4** **NOT** **NULL**,

unite\_adi **varchar**(400) **NOT** **NULL**,

konu\_no **int4** **NULL**,

konu\_adi **varchar**(400) **NULL**,

kazanim\_no **int4** **NOT** **NULL**,

kazanim\_adi **varchar**(400) **NOT** **NULL**

);

**CREATE** **TABLE** mufredat.unite (

id **int4** **NOT** **NULL** **GENERATED** **ALWAYS** **AS** **IDENTITY**,

sinif **varchar**(10) **NOT** **NULL**,

ders **varchar**(100) **NOT** **NULL**,

unite\_no **int4** **NOT** **NULL**,

unite\_adi **varchar**(400) **NOT** **NULL**,

**CONSTRAINT** unite\_pkey **PRIMARY** **KEY** (id)

);

**CREATE** **TABLE** mufredat.konu (

id **int4** **NOT** **NULL** **GENERATED** **ALWAYS** **AS** **IDENTITY**,

unite\_id **int4** **NOT** **NULL**,

konu\_no **int4** **NULL**,

konu\_adi **varchar**(400) **NULL**

);

**CREATE** **TABLE** mufredat.kazanim (

id **int4** **NOT** **NULL** **GENERATED** **ALWAYS** **AS** **IDENTITY**,

unite\_id **int4** **NOT** **NULL**,

konu\_id **int4** **NULL**,

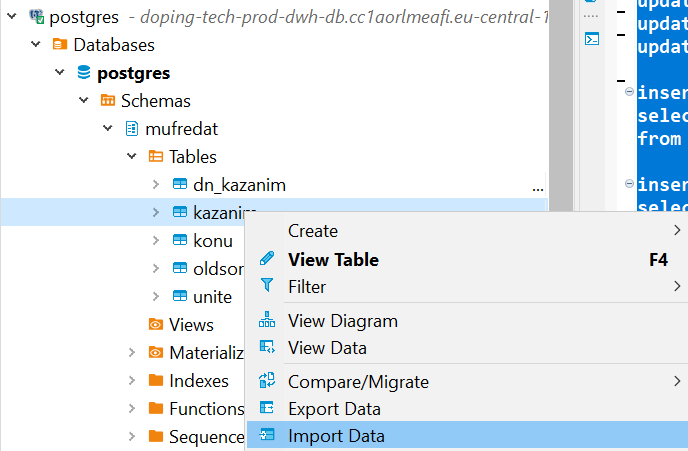
kazanim\_key **varchar**(50) **NOT** **NULL**,

kazanim\_no **int4** **NOT** **NULL**,

kazanim\_adi **varchar**(400) **NOT** **NULL**

);

After creating the tables and the schema for the new database, I imported all of the learning objectives csv files separately. I was using DBeaver program to reach to the database and write SQL scripts. I also use this program’s ‘Import Data’ feature to properly import Excel tables to the database as shown below.



After importing all learning objectives tables of all lessons to the ‘dn\_kazanim’ table, we noticed that some of the data rows had empty spaces in front of the string or end of the string. As this may cause a problem in the future, we decide to get rid of all these empty spaces with the trim() function of SQL. Now, we had to fill the ‘unite’, ‘konu’ and ‘kazanım’ tables consecutively. For first of them, selectin only the desired columns which are grade level, lesson name, unit no and unit name from dn\_kazanim was enough to complete the table. Since we created this table with ‘id **int4** **NOT** **NULL** **GENERATED** **ALWAYS** **AS** **IDENTITY**’ expression, we don’t have to specify id from any other place since it will be generated by itself. This id will become the ‘unite\_id’ of ‘konu’ table which contains all topics with their unit id’s, topic numbers and names. Again topic id is generated by itself. This time, this table is created by joining units table and ‘dn\_kazanim’ table since we had to get unit table’s unit id’s. Last of all and the most challenging of all, we had to fill the ‘kazanim’ table which includes all learning objectives with corresponding unit id and topic id that were generated by topic table and unit table. For this purpose, we had to join both ‘dn\_kazanım’, ‘unite’ and ‘konu’ tables. We wrote nested queries and first we left join unit table with konu table and then join this table with ‘dn\_kazanim’. Another thing to highlight in this query was that there were some learning objectives that don’t belong to any topics since some curriculum don’t have topic entity. So, when joining ‘dn\_kazanim’ table with joined unit and topic tables, instead of joining them on the condition of just matching topic names, we add that topic name can be null or can be matching in two of these tables. Below, there are the final SQL queries that we used for filling all the tables as I explained them above.

**update** dn\_kazanim **set** konu\_adi = **trim**(konu\_adi);

**update** dn\_kazanim **set** unite\_adi = **trim**(unite\_adi);

**update** dn\_kazanim **set** sinif = **trim**(sinif);

**update** dn\_kazanim **set** konu\_adi = **trim**(konu\_adi);

**update** dn\_kazanim **set** kazanim\_adi = **trim**(kazanim\_adi);

**update** dn\_kazanim **set** kazanim\_key = **trim**(kazanim\_key);

**insert** **into** unite(sinif,ders,unite\_no,unite\_adi)

**select** **distinct** sinif , ders, unite\_no , unite\_adi

**from** dn\_kazanim dk;

**insert** **into** konu(unite\_id, konu\_no,konu\_adi)

**select** **distinct** u.id, dk.konu\_no, dk.konu\_adi

**from** unite u, dn\_kazanim dk

**where** dk.unite\_adi = u.unite\_adi **and** dk.sinif = u.sinif **and** dk.ders = u.ders **and** dk.unite\_no = u.unite\_no

**insert** **into** kazanim(unite\_id, konu\_id, kazanim\_key, kazanim\_no, kazanim\_adi)

**select** **distinct** unite\_unite\_id, konu\_id ,kazanim\_key,kazanim\_no ,kazanim\_adi **from**(

**select** \* **from** (

**select** \* , k.id **as** konu\_id , u.id **as** unite\_unite\_id

**from** unite u

**left** **join** konu k

**on** k.unite\_id = u.id ) **as** uk

**join** dn\_kazanim dk **on** dk.unite\_adi = uk.unite\_adi **and** dk.sinif = uk.sinif **and** dk.ders = uk.ders **and** dk.unite\_no = uk.unite\_no

**and** (uk.konu\_adi **is** **null** **or** uk.konu\_adi = dk.konu\_adi)) **as** hee

Furthermore, old content of the company which were held in ‘onlinee’ database need to be migrated to the new database of the company. Our supervisor especially asked us to collect all questions from the database. In the old database, all questions were separated across different tables and column names were a bit different too. We came together and have a conference meeting with interns, our supervisor, CTO, Data Engineering Mananager and Data Scientist to discuss the new names of the columns of new and unified table. According to those names, we wrote a basic SQL query that unifies question from different tables: ‘Akilli Test Sorulari’, ‘Tarih Akilli Test Sorulari’, ‘Tarih Testi Sorulari’, ‘test\_soru’,’deneme\_soru’, ‘math\_test\_sorular’, ‘rt\_test\_sorular’ and ‘rfen\_test\_sorular’.

select ats.id as Table\_id,

'Akilli Test Sorulari' as Table\_name,

ats.uniteid as unit\_id,

ats.konuId as konu\_id,

ats.testid as test\_id,

ats.video as solution\_video\_id,

ats.sira as 'Order',

ats.soru as 'Content'

from akilli\_test\_sorular ats

union all

select tats.id as Table\_id,

'Tarih Akilli Test Sorulari' as Table\_name,

tats.uniteid as unit\_id,

tats.konuId as konu\_id,

tats.testid as test\_id,

tats.video as solution\_video\_id,

tats.sira as 'Order',

tats.soru as 'Content'

from tarih\_akilli\_test\_sorular tats

union all

select tts.id as Table\_id,

'Tarih Testi Soruları' as Table\_name,

'-' as unit\_id,

'-' as konu\_id,

'-' as test\_id,

tts.video as solution\_video\_id,

'-' as 'Order',

tts.soru as 'Content'

from tarih\_test\_soru tts

union all

select

ts.id as Table\_id,

'Test\_soru' as Table\_name,

'-' as unit\_id,

'-' as konu\_id,

ts.testId as test\_id,

ts.video as solution\_video\_id,

'-' as 'Order',

ts.soru as 'Content'

from test\_soru ts

union all

select

ds.id as Table\_id,

'deneme\_soru' as Table\_name,

'-' as unit\_id,

'-' as konu\_id,

ds.testId as test\_id,

ds.video as solution\_video\_id,

ds.sira as 'Order',

ds.soru as 'Content'

from deneme\_soru ds

union all

select

mts.id as Table\_id,

'math\_test\_sorular' as Table\_name,

mts.uniteId as unit\_id,

mts.konuId as konu\_id,

mts.testId as test\_id,

mts.video as solution\_video\_id,

mts.sira as 'Order',

mts.soru as 'Content'

from math\_test\_sorular mts

union all

select

rtts.id as Table\_id,

'rt\_test\_sorular' as Table\_name,

rtts.uniteId as unit\_id,

rtts.konuId as konu\_id,

rtts.testId as test\_id,

rtts.video as solution\_video\_id,

rtts.sira as 'Order',

rtts.soru as 'Content'

from rt\_test\_sorular rtts

union all

select

rfts.id as Table\_id,

'rfen\_test\_sorular' as Table\_name,

rfts.uniteId as unit\_id,

rfts.konuId as konu\_id,

rfts.testId as test\_id,

rfts.video as solution\_video\_id,

rfts.sira as 'Order',

rfts.soru as 'Content'

from rfen\_test\_sorular rfts

Similar to this, we were also asked to unify the lecture videos that are again spread across different tables in the database. Exactly same operation is done again and we come up with the column names in a conference meeting and collect the information inside the ‘egitim\_dersler’, ‘math\_m\_egitim\_konular’, ‘rfen\_m\_egitim\_konular’, ‘rt\_m\_egitim\_konular’ tables.

select \*

from (select eu.id, eu.ad as unite\_ad, ed.ad as ders\_ad

from egitim\_dersler ed, egitim\_uniteler eu

where ed.id = eu.dersId) ders\_unite, (select ek.id as content\_id,

'egitim\_konular' as Table\_name,

0 as eders\_id,

ek.dersId as ders\_id,

ek.uniteId as unit\_id,

ek.sira as 'Order',

ek.aciklama as aciklama,

ek.resim1 as resim1,

ek.resim2 as resim2,

ek.video as video\_id

from egitim\_konular ek

union all

select mmek.id as content\_id,

'math\_m\_egitim\_konular' as Table\_name,

mmek.edersId  as eders\_id,

mmek.dersId as ders\_id,

mmek.uniteId as unit\_id,

mmek.sira as 'Order',

mmek.aciklama as aciklama,

mmek.resim1 as resim1,

'-' as resim2,

mmek.video as video\_id

from math\_m\_egitim\_konular mmek

union all

select rmek.id as content\_id,

'rfen\_m\_egitim\_konular' as Table\_name,

rmek.edersId as eders\_id,

rmek.dersId as ders\_id,

rmek.uniteId as unit\_id,

rmek.sira as 'Order',

rmek.aciklama as aciklama,

rmek.resim1 as resim1,

'-' as resim2,

rmek.video as video\_id

from rfen\_m\_egitim\_konular rmek

union all

select rmek2.id as content\_id,

'rt\_m\_egitim\_konular' as Table\_name,

rmek2.edersId as eders\_id,

rmek2.dersId as ders\_id,

rmek2.uniteId as unit\_id,

rmek2.sira as 'Order',

rmek2.aciklama as aciklama,

rmek2.resim1 as resim1,

'-' as resim2,

rmek2.video as video\_id

from rt\_m\_egitim\_konular rmek2) uni\_sorgu

where ders\_unite.id = uni\_sorgu.unit\_id

Moreover, we export this query output as a csv file aiming to import that to the new database of the company. I was asked to write the create statements for the table that will serve as a base for old questions. Below, there is the code for ‘old\_sorular’ table where I imported the unified questions query’s output into.

CREATE TABLE mufredat.oldsorular (  
 Table\_id int4 NOT NULL,  
 Table\_name varchar(100) NOT NULL,  
 unit\_id int4 NOT NULL,  
 test\_id int4 NOT NULL,  
 solution\_video\_id int4,  
 orderr int4 NOT NULL,  
 contentt varchar(20000) NOT NULL);

As an extra task for the sake of learning more, Data Scientist Beril wants me to learn how to connect to database with python and execute SQL command on the database with python. I didn’t write the exact password to the code below for the reasons of safety. mufredat\_connection is for establishing the connection between python and the database of the company. It was supposed to connect with the new database which was called ‘postgres’. If autocommit were to be false, we had to write mufredat.connection.commit() after work is done. Moreover, we had to close the connection after committing the changes in DB by mufredat\_connection.close(). Since we execute a ‘SELECT’ command in the below code, we have to benefit from what cursor class provides. After executing the sql command, our cursor, mcursor provides 3 different functions to fetch the result. fetchall() function retrieves all rows in the result set of a query and returns them as a list of tuples. Moreover, if we execute this after retrieving few row, it returns remaining ones. fetchone() fetches the next row in the result of a query and fetchmany() fetches the next set of rows. After fetching, python program is ready to print the output query to the console.

import psycopg  
mufredat\_connection = psycopg2.connect(  
 host="[doping-tech-prod-dwh-db.cc1aorlmeafi.eu-central-1.rds.amazonaws.com](http://doping-tech-prod-dwh-db.cc1aorlmeafi.eu-central-1.rds.amazonaws.com/)",  
 database="postgres",  
 user="junior",  
 password="\*\*\*")

mufredat\_connection.autocommit = True  
mcursor = mufredat\_connection.cursor()

sql\_query = '''SELECT \* FROM mufredat.dn\_kazanim'''  
mcursor.execute(sql\_query)  
result = mcursor.fetchall()  
print(result)

Last of all, when we don’t have any tasks we were supposed to search the internet for Docker or study Java programming. Docker is computer program that provides a way to package software so it can run on any hardware. It has a structure that is build upon containers, docker images and docker files. Container is a running process and Docker Image is the template for running docker containers. Docker file contains the code to build docker images and serves as a blueprint for building a docker image. As for studying Java, I choose to learn it from the website Codecademy. I started with basic concepts such as printing a line, comments, main() method and compiling. What seemed different from me was that in Java, each individual class is converted into a .class file (byte code) when we compile a program. Java Virtual Machine is used to run that byte code. It was a bit hard to understand javac filename.java compiling command and java filename execution command. After key concepts, I moved to learning variables such as Boolean, int, char, strings. Again different from what I learned before in coding, final keyword was used to declare that a variable cannot be changed and any attempt at doing so will result in an error message. Then, I moved to learning the object-oriented nature of Java. In this section I learn the objects’ state and behavior within classes, creating a new class instance, constructors and many more. Last of all, I get to finish the conditionals and control flow part of the course where if and else statements were explained. Nested conditional statements, AND, NOT and OR operators were also introduced in this chapter. The website allows user to both learn and practice at the same time and user cannot skip to the next episode without passing the practice coding parts which I found very useful. I was able to see if I can truly understand the concepts and can implement what I understood.

4.6 Results

‘Online Education Transformation’ project started at the same time with my internship. It was planned to fully complete in May. It was the biggest project that company was currently working on but it was not the only project. Also it encompasses creating a education platform from scratch which means that engineers had to create the same but improved ‘Doping Hafıza’ program that have been developing since 2011. All the cumulative labor were to be done again within a year. Thus, project hadn’t been completed when my internship ended. On the other hand, almost all the parts and task that were assigned to me were done. Learning objectives were downloaded from the MEB’s website and put into Excel tables in desired format. New database is created and these objectives were then imported into this new database. As for the simplicity and practicability of the DB, we also created different tables for units, topics. For migrating old questions into the new database, I created an old questions table in new database. With the help of the SQL query that unifies all questions that were spread over the old database, using ‘UNION ALL’ statements, we exported the table with all existing questions and import it back to the new database. Unfortunately, we had to singularize all the questions since there were multiple versions of same question in the database. We weren’t able to come up with a solution to that singularization problem although we were able to get a hash code for all questions with md5() function of SQL. In continuation of the ‘Online Education Transformation’ project, all of these learning objectives will be connected to questions. This task will be done by instructor working at Doping Technology but engineers will create a recommendation, suggestion pool of learning objectives per question by some Natural Language Processing utilities. Project is planned to be completed in May, so that it can be put on the market for June, July for the upcoming education semester.

5. INTERNSHIP EXPERIENCE (3 pages max)

5.1 Learning

Although I have plenty of experience with python programming, there is always new thing to learn about any program or any concept in Computer Science. I learned some libraries that I don’t know and I learned some libraries that I know but haven’t got a chance to work with it. For opening the pdf files of curriculum, I had to learn PyPDF2 library and its functions. For DB connection, I learned how to use psycopg2 library, connect and cursor function and other features of cursor class. Moreover, we started the internship by preparing a report on monolithic architecture and multiservice architecture. It is a great way to start a programming project to select one of them to proceed. In advance, although we don’t actually take part in Java programming within the company, I started to learn it so that I can practice more in future. Docker learning was also very useful since it is not a thing that is teached in school but rather engineers have to search for new developer tools and have to be familiar with new technologies to be able to choose to work with them in new projects.

I learned that it is important for an engineer to learn how to learn in order to stay up to date. There’s a certain system of learning new technologies through the internet and a developer should spend some time on the internet to understand this system. When a new technology is put to the market, communities are formed to learn and teach this new technology. Forums, videos, tutorials are all different ways to update your current knowledge. People ask questions and other people reply to those questions and this learning process never ends in this job I assume. As I observe my supervisor and all other engineers in my department, most of their time goes to surfing on the internet, searching for better ways to implement what is needed to be implemented. Searching about the errors, searching about any problem in the flow of the program consume a lot more time than coding I observe.

Moreover, I got a chance to practice SQL deeply. All of the programming languages are only learned with hands-on experience in my opinion. Having a real database open under DBeaver and trying queries all the time was really beneficial for me. In the case of an error or an unwanted situation, I had to write a lot more queries to understand the problem. This way all of the concepts of database design got a lot more concrete in my mind. Similar to that, I was able to observe how Data Scientist in the company design the database from scratch so that it will be practical to use them. Seeing how they are planned and designed in the first place and having a chance to implement them played a great role in understanding Databases and key concepts a lot more clearly.

Last of all, command window was used frequently in my internship for a couple of reasons. We were able to download packages, compile and run Java files, download programs etc. I learned that having a good knowledge in commands is also a really important skill for a developer.

5.2 Relation to undergraduate education

First of all, I get to use almost all of the knowledge I learn from CS306 Database Management Systems course. Luckily, I had plenty of notes from that class scanned in my computer and even other interns benefited from them. In that class, we were able to learn SQL and had to create a DBMS project from scratch. I was able to talk during conference meetings and state my opinion since I have background information on that topic. Moreover, I was able to recall the concept names or syntax names when our supervisor introduce us a new task. I could see the difference between different interns from different universities with respect to the knowledge we have. Also, I took CS307 Operating Systems course which made me familiar with command windows and different commands. I can say it is also really helpful for me because learning with no background information in any internship can be hard since we had to ask too many questions and our supervisor may not have that much time. Being familiar with concepts make it much easier to improve myself and I was able to efficiently work in tasks.

Secondly, starting from IF100, we used Python for too many projects and had many experience with Python. Even in the company, I was one of the most informed ones in Python. I was able to understand what will work or what will not in Python without starting to task. It’s comparably a new technology and not many old developers had time to learn it deeply. I thought that it’s a great chance to know Python in depth since it can be used for a lot of things. Having little Excel information was not a problem for me since I was able to accomplish everything with Python. Furthermore, in different courses like ENS208, BIO310 we were implementing what we learned in Python and had to learn more about it in every week. Thus, I already knew how to learn new things, functions and libraries or how to use the new knowledge on Python.

Lastly, CS310 Mobile Application Development course project was really similar to multiservice architecture. Since it is a really abstract and theatrical subject, it cannot be easily learned in my opinion. But I was lucky to imagine what we had in that class when I was studying these architectures. On the other hand, I thing we were able to learn many things about software engineering in that class. I wouldn’t be able to picture front-end and back-end of a project if I didn’t have the knowledge I gained from that course.

5.3 Difficulties

First difficulty I face was when we were trying to get learning objectives to an Excel file. On MEB’s website, curriculum documents for different lessons were a little bit different. For example, in some of the documents there were keywords between unit names and learning objective descriptions which we need to get rid of. Since these keywords were written differently in documents such as capitalization, I had to change the code for all PDF files in the website. Moreover, sometimes there were extra punctuation in between the learning objectives that will cause my program to fail. I had to identify what was causing the error and then maybe manually handle it or write some extra code to handle it in some of them.

Second difficulty I face was with SQL statements when I was trying to write a query to join ‘dn\_kazanim’, ‘unite’ and ‘konu’ tables in order to fill the table for learning objectives. I couldn’t write the correct query and I don’t want to ask my supervisor because I thought that he may doubt about my skills and knowledge. Then, I realize that wasting time is a lot worse than hesitating to ask. In the beginning of the internship I don’t want to ask anything and solve all problems by myself but this may not always be the best and most efficient method. When I started to ask questions about tasks, my effort and labor becomes a lot more visible. We start to discuss how to solve that problem and I could tell about what I know and I felt more confident with the knowledge I have. This way I was able to show how capable I am of learning and growing.

Last but not least, third and maybe the greatest difficulty I face was that the other interns weren’t doing anything. When a task is introduced to us, I immediately start to work on that specific task. If needed, I start to learn more about the topic or programming language first and then get started on the task. However, other interns didn’t have any roadmap at those times. I was the one telling them what to do. Since I couldn’t predict when they will be able to do their part of the task, most of the time I was also handling their parts. I don’t want to speak to my supervisor about the topic although he was pretty aware of the situation. Since company and my supervisor were suitable with this situation, I keep on doing more than my parts and learned much more about it because I was also explaining what we are doing or what we are supposed to do to other interns. This way, I get to learn more about concepts that I was currently working on.

5.4 A typical day

Once a week we were supposed to go to the company. As for every day, we were having meeting at 9.30 which we call ‘Daily’. In those meetings, everyone describes what they did yesterday and what will they be doing that day. Little feedbacks were given in the meetings and business analysts ask further questions about tasks and make reminders about upcoming meetings on that day or nearby deadlines of tasks. Rest of the day differs according to the task we got. If we are in the phase of learning, most of the time we had meetings with supervisors. If we are working on a task, we usually don’t have any meetings if we don’t have specific questions or difficulties. We send the output or the code to our supervisor when we are done with it and asks his feedback in return. If he’s busy at that time, we used to study Java or any other language that we don’t think we are fluent in.

6. CONCLUSIONS (1 page max)

In summary, my internship experience contains both learning new things and experience to implement this new knowledge and develop my existing skills and having experience with these skills. We started with learning about multiservice and monolithic architecture. Our first task was to get the learning objectives from the curriculum. With the help of Python, I wrote several programs to output these learning objectives in a desired table format. Then with these Excel tables, we opened a new database and imported these tables. With the help of SQL queries, we also created different tables and did some little modifications of raw data. Moreover we unify the questions and videos in the old database which is too messy and is needed to be properly collect up. We also imported these old questions to a newly made table in new database. Besides of these, we learn about new technologies and tools, study some theatrical stuff, work on Java programming.

In school, I used to think that I am not a good developer and I didn’t believe that I will ever become a good developer or a coder. I had many doubts about becoming a software engineer as I don’t think I will be able to spend that much time for programming. I couldn’t find motivation to improve myself. During the internship, I had time and motivation to study for myself and get a chance to develop myself. Having this leisure time, getting to see how a project is implemented in a broader view made me see that I can develop myself in a much greater scale. In time, when I get more familiar with the programs, I get the sense that I know what I am doing and the feeling of usefulness becomes very fulfilling. Programming and coding takes too much time to practice and I started to enjoy this time eventually.

7. RECOMMENDATIONS

I think every person has different routines for working. Working from 8 AM to 6 PM might not be very effective for everyone. Some people would benefit from working at nights or working task based rather than strict employment periods. It is important for oneself to know what is working with them to work much more efficiently. This also applies to having a remote, hybrid or a face-to-face internship. I recommend that students will choose their internship according to their desired life-style.

Secondly, I used to think that it’s not a good thing to ask questions to my supervisor. I don’t want anyone to think that I am less qualified than I actually am. I was afraid if my questions cause company to think that I am not adequate. I was wrong about this since asking questions demonstrate the enthusiasm and eagerness to learn. When I get to discuss my problems with other engineers, my work and my knowledge became more visible. My suggestion is that no one should hesitate to ask questions to their supervisor.

Last of all, I recommend people to do their internship at comparably little companies. In Doping Technology, there was a really dynamic environment. There were no strict job descriptions so I get to learn about every process that company has been going through. I get so see different engineers’ work although I was a Back End Intern. In more organized and big companies, everything is settled and you would probably have limited access to many things.

8. REFERENCES

Raman\_257 (2022). Monolithic vs Microservices architecture. Geeksforgeeks. Retrieved September 30, 2022, from <https://www.geeksforgeeks.org/monolithic-vs-microservices-architecture/>

Gnatyk R. (2018). Microservices vs Monolith: which architecture is the best choice for your business? N-ix. Retrieved September 30, 2022, from <https://www.n-ix.com/microservices-vs-monolith-which-architecture-best-choice-your-business/>