Óbudai Egyetem



RFID óra időmérés Labview alkalmazássávaL

projektum  
Intelligens Rendszerek tárgyból

témavezető: Dr.Simon János hallgató: Kovács Árpád   
 főiskolai tanár Neptun kód: BPJZ56

Szabadka, 2020

Tartalom

[Bevezető 3](#_Toc56681868)

[1. Projektfeladat 3](#_Toc56681869)

[2. Elméleti alapok 3](#_Toc56681870)

[Kisméretű modulok családja, amelyek ESP32 chipet és számos több kulcsfontosságú komponenst tartalmaznak, beleértve a kristály oszcillátort és MIFA antennát. Ez megkönnyíti az ESP32 alapú eszközök használatát, amely így készen áll a végtermékekbe való beszerelésre. Ezek a modulok fejlesztési platformként is használhatók, mivel számos egyéb összetevőt is tartalmaznak, például programozási felületet, a passzív alkatrészeket és tüskesort. 3](#_Toc56681871)

[3. A LabVIEW szoftver bemutatása 7](#_Toc56681872)

[A felhasznált rövidítések 10](#_Toc56681873)

[Irodalom 10](#_Toc56681874)

# Bevezető

A dokumentum az intelligens rendszerek tárgy dokumentálásának céljából jött létre.

# Projektfeladat

Az én projekt feladatom az volt, hogy egy adatbázis segítségével mérjem az időt az rfid kártyák lecsippantásával. Ehhez a feladathoz ESP32 mikrovezérlőt használtam fel.

# Elméleti alapok

Labview Program:

A labview az NI által fejlesztett ún. Virtuális műszerek létrehozására, és grafikus programozásra használt szoftver.

ESP32

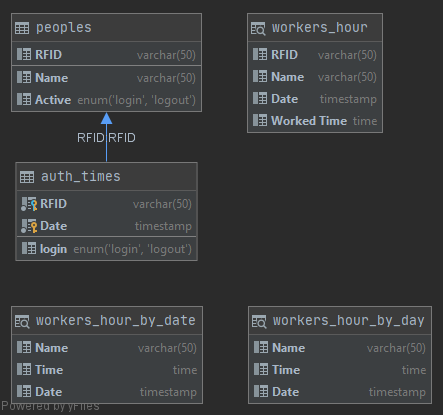
Kisméretű modulok családja, amelyek ESP32 chipet és számos több kulcsfontosságú komponenst tartalmaznak, beleértve a kristály oszcillátort és MIFA antennát. Ez megkönnyíti az ESP32 alapú eszközök használatát, amely így készen áll a végtermékekbe való beszerelésre. Ezek a modulok fejlesztési platformként is használhatók, mivel számos egyéb összetevőt is tartalmaznak, például programozási felületet, a passzív alkatrészeket és tüskesort.



A felhasznált arduino program:.

|  |
| --- |
| #include <Arduino.h> #include <WiFi.h> #include <WiFiMulti.h> #include <MFRC522.h> #include <HTTPClient.h> #include <Arduino\_JSON.h>  #define USE\_SERIAL Serial #define SS\_PIN 21 #define RST\_PIN 22  WiFiMulti wifiMulti; JSONVar my\_elements = JSON.parse("{\"RFID-code\" : \"valami hexa\"}"); MFRC522 rfid(SS\_PIN, RST\_PIN); // Instance of the class MFRC522::MIFARE\_Key key; byte nuidPICC[4]; String s\_rfid="";   void setup() {  USE\_SERIAL.begin(9600);  USE\_SERIAL.println();  USE\_SERIAL.println();  USE\_SERIAL.println();  for(uint8\_t t = 4; t > 0; t--) {  USE\_SERIAL.printf("[SETUP] WAIT %d...\n", t);  USE\_SERIAL.flush();  delay(1000);  }  wifiMulti.addAP("TP-Link", "asdfghjkl123#");  SPI.begin(); // Init SPI   rfid.PCD\_Init(); // Init MFRC522  for (byte i = 0; i < 6; i++) {  key.keyByte[i] = 0xFF;  } } void loop() {  if ( ! rfid.PICC\_IsNewCardPresent()) return;  if ( ! rfid.PICC\_ReadCardSerial()) return;   Serial.print(F("PICC type: "));  MFRC522::PICC\_Type piccType = rfid.PICC\_GetType(rfid.uid.sak);  Serial.println(rfid.PICC\_GetTypeName(piccType));  if (piccType != MFRC522::PICC\_TYPE\_MIFARE\_MINI &&  piccType != MFRC522::PICC\_TYPE\_MIFARE\_1K &&  piccType != MFRC522::PICC\_TYPE\_MIFARE\_4K) {  Serial.println(F("Your tag is not of type MIFARE Classic."));  return;  }  for (byte i = 0; i < 4; i++) nuidPICC[i] = rfid.uid.uidByte[i];    Serial.println(F("The NUID tag is:"));  Serial.print(F("In hex: "));  s\_rfid="";  for(byte i=0;i<rfid.uid.size;i++)  {  s\_rfid+=String(rfid.uid.uidByte[i],HEX);  }  Serial.println(s\_rfid);  //printHex(rfid.uid.uidByte, rfid.uid.size);  Serial.println();  //setClock();  // wait for WiFi connection  if((wifiMulti.run() == WL\_CONNECTED)) {   HTTPClient http;   USE\_SERIAL.print("[HTTP] begin...\n");  // configure traged server and url  //http.begin("https://www.howsmyssl.com/a/check", ca); //HTTPS  http.begin("http://46.40.46.94:81/timers.php?json="+s\_rfid); //HTTP   USE\_SERIAL.print("[HTTP] GET...\n");  // start connection and send HTTP header  int httpCode = http.GET();   // httpCode will be negative on error  if(httpCode > 0) {  // HTTP header has been send and Server response header has been handled  USE\_SERIAL.printf("[HTTP] GET... code: %d\n", httpCode);   // file found at server  if(httpCode == HTTP\_CODE\_OK) {  String payload = http.getString();  USE\_SERIAL.println(payload);  }  } else {  USE\_SERIAL.printf("[HTTP] GET... failed, error: %s\n", http.errorToString(httpCode).c\_str());  }   http.end();  }   // Halt PICC  rfid.PICC\_HaltA();   // Stop encryption on PCD  rfid.PCD\_StopCrypto1(); } |

MySQL adatbázis:.



A szoftverhoz felhasznált adatbázis kezelő rész.

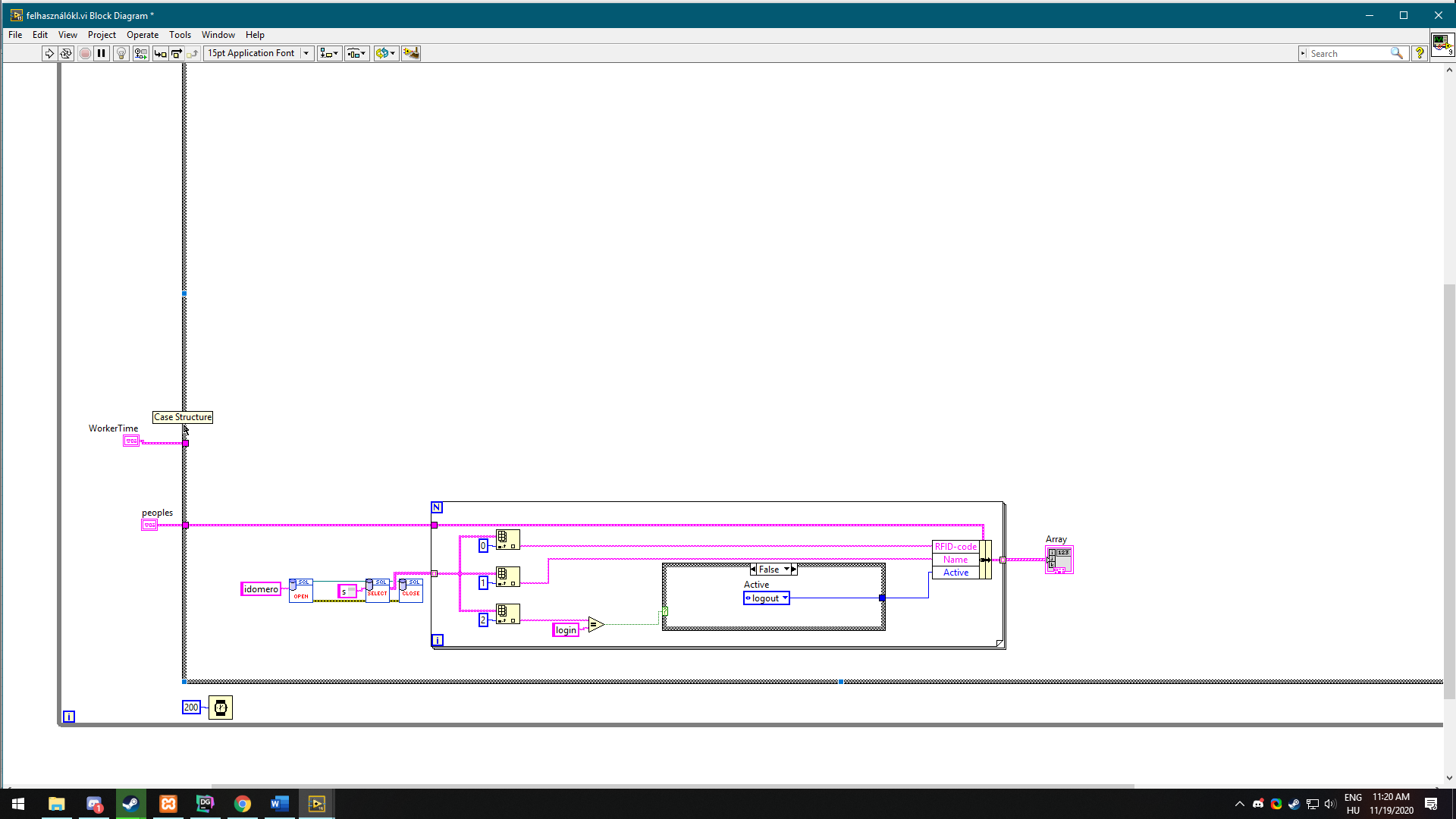
|  |
| --- |
| drop procedure if exists *auth\_times\_pre*; delimiter $$ create procedure *auth\_times\_pre*(in RFID varchar(50))  begin  declare c enum('login','logout');  select Active into c from peoples where peoples.RFID=RFID;  -- select c;  if isnull(c) then  insert into peoples values (RFID,'Temporary Joe',1);  end if;   if c='login' then  -- select 'login';  insert into auth\_times values (RFID,NOW(),'logout');  update peoples set Active='logout' where peoples.RFID=RFID;  else  -- select 'logout';  insert into auth\_times values (RFID,NOW(),'login');  update peoples set Active='login' where peoples.RFID=RFID;  end if;   end $$  DELIMITER ; |

Adatbázisba való adatbevitel.

|  |
| --- |
| <?php $servername = "localhost"; $username = "root"; $password = ""; $conn = new mysqli($servername, $username, $password,"idomero");  //var\_dump($\_GET); if(isset($\_GET["json"])) {  $json=$\_GET["json"];  $res=mysqli\_query($conn,"call auth\_times\_pre('{$json}')");  //var\_dump($res); } |

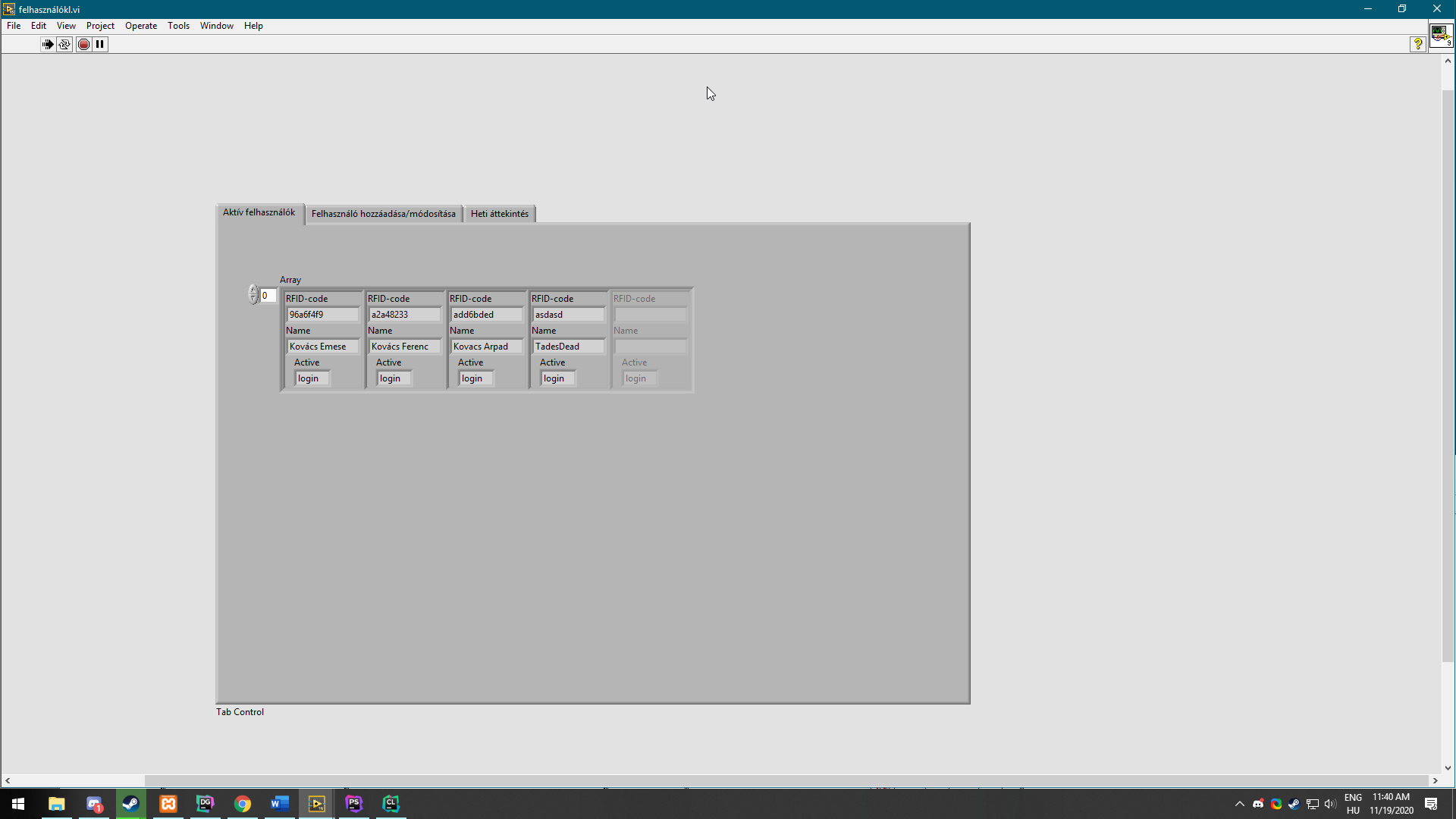
# 3. A LabVIEW szoftver bemutatása

Tartalmazza a szimulációs séma részeit.

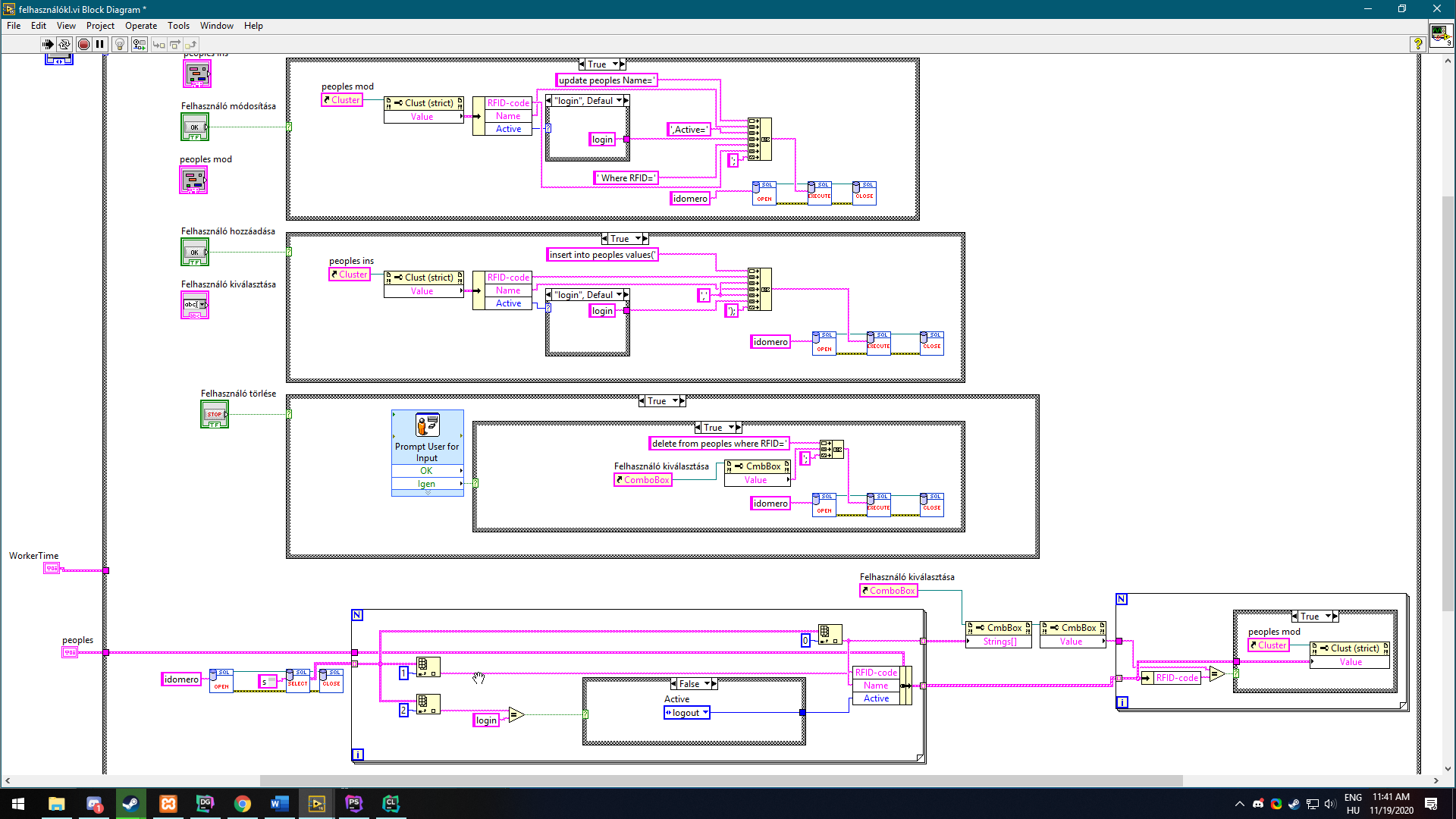


ábra felhasználók áttekintése

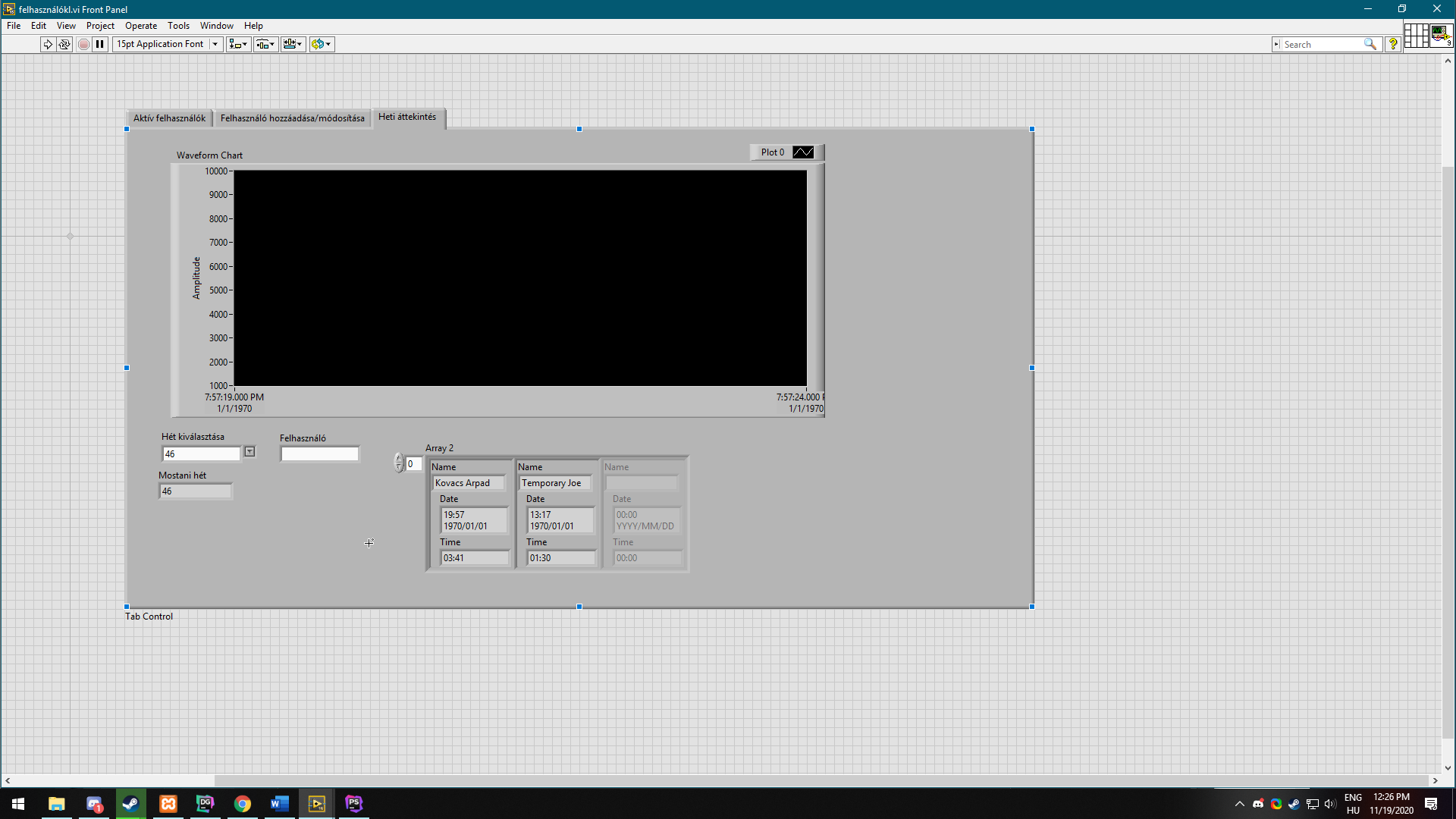
A labview front panelje:.

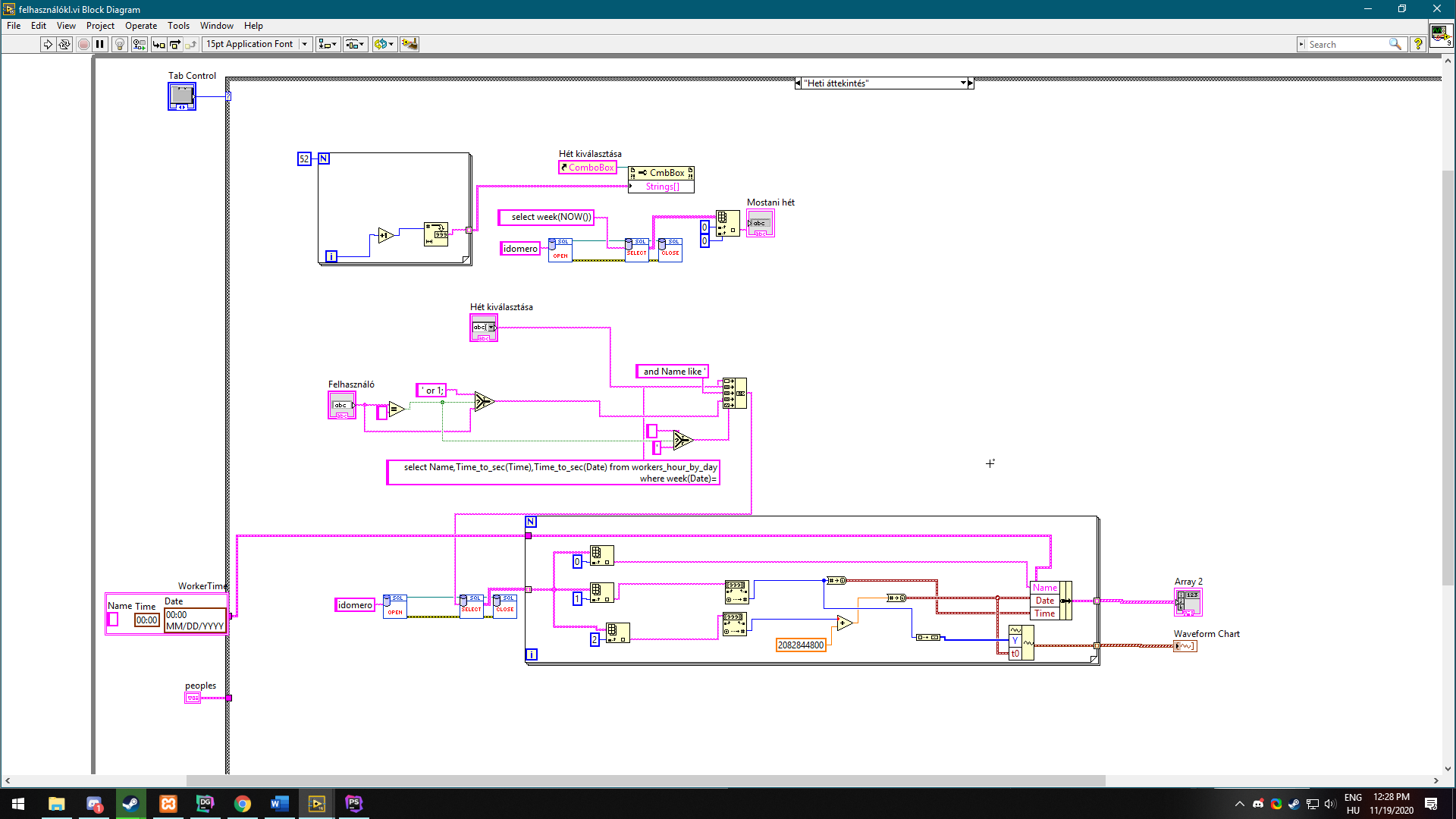


A felhasználók bevitele/módosítása:.



A felhasználók heti áttekintése



A felhasználók heti áttekintése sémája:. 

|  |
| --- |
| drop view if exists workers\_hour; create view workers\_hour as  select a.RFID,p.Name,a.Date,timediff(b.Date,a.Date) as 'Worked Time' from auth\_times a inner join peoples p on a.RFID = p.RFID  right join (select RFID,Date,login from auth\_times aub order by Date) as b on a.RFID=b.RFID where a.Date<b.Date and day(b.Date)=day(a.Date) and a.login='login';  Kijelentkezés és bejelentkezés közötti idő mérés:.  drop view if exists workers\_hour\_by\_day; create view workers\_hour\_by\_day as  select Name, SEC\_TO\_TIME(SUM(TIME\_TO\_SEC(`Worked Time`)))as Time, Date from workers\_hour group by Name; |

# A felhasznált rövidítések

A használt rövidítések jegyzéke és azok jelentése.

DMA Direct Memory Access

# Irodalom

<https://www.w3schools.com/php/php_mysql_intro.asp>

<https://people.vts.su.ac.rs/~simon/bp2/Baze_Podataka2_Prirucnik_SR_2015.pdf>