

# Szoftvertchnológia és -technikák 2. házi feladat

*Meteorológiai adatbázis*

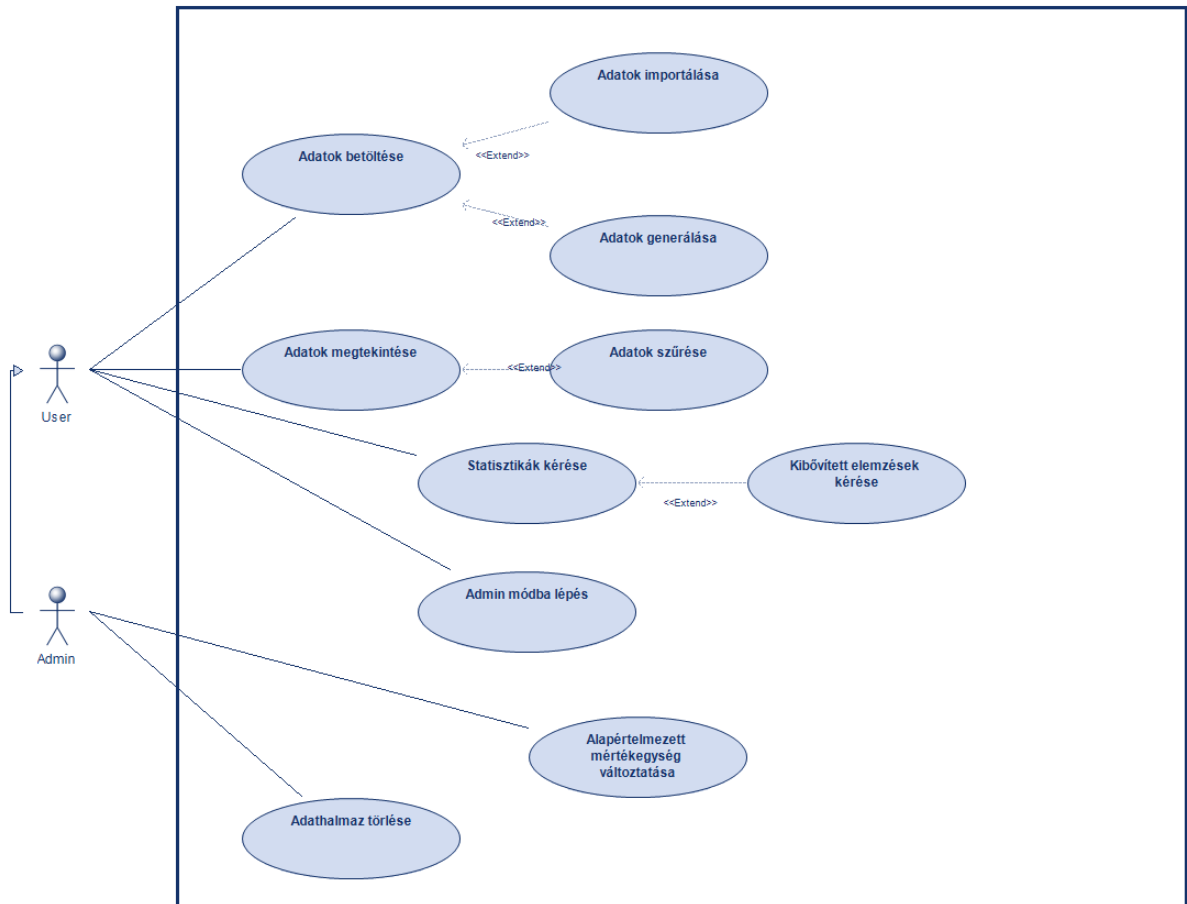
Készítette:

*Jánovszki András (GUFP6A)*

*Ládi Kornél (H0HBHV)*

# Követelményspecifikáció

## Use-case diagram



## Szerepkörök és használati esetek leírása

### User

A *User* aktor a programban be tudja tölteni az adatokat, vagy importálással, ha léteznek szenzoradatok, vagy pedig a program által generáltatva. A betöltött adatokat meg tudja tekinteni, valamint szűrni azokat bizonyos időtartamra. Az adatokból, továbbá, tud statisztikai kimutatásokat generáltatni a programmal: vagy napi szintű, vagy pedig az összes betöltött adatpontból készített kimutatást.

### Admin

Az *Admin* aktor képes a *User* aktor minden cselekvését végrehajtani. Ezeken felül tudja törölni a programba korábban betöltött adatállományt, valamint a program által használt, és a kimutatásoknál megjelenített alapértelmezett mértékegységeket.

## Funkcionális és nem funkcionális követelmények

### Funkcionális követelmények

Azonosító	Leírás	Use-case	Komment
R00	Parancssori környezetben meg tud szöveget jeleníteni.		A program használatához végig szükséges
R01	Parancssori környezetben tud beolvasni standard inputról szöveget.		A program használatához végig szükséges
R02	Tud XML formátumú fájlt megnyitni és olvasni.	Adatok importálása	
R03	Létezik <i>User</i> felhasználótípus.		A program használatához végig szükséges
R04	Létezik <i>Admin</i> felhasználótípus.		A program használatához végig szükséges
R05	<i>User</i> felhasználó tud fájlnev megadásával fájlt beolvasatni a programmal.	Adatok beolvasása, Adatok importálása	
R06	<i>User</i> felhasználó meg tudja jeleníttetni a programmal a beolvasott adatokat.	Adatok megjelenítése	
R07	<i>User</i> felhasználó tud a programmal adatokat generáltatni.	Adatok beolvasása, Adatok generálása	
R08	<i>User</i> felhasználó két időpont, maximum és minimum értékek megadásával tudja szűrni az adatokat.	Adatok szűrése	
R09	<i>User</i> felhasználó tud generáltatni statisztikai kimutatást a programmal.	Statisztikák kérése	
R10	<i>User</i> felhasználó tud generáltatni napi statisztikai kimutatást a programmal.	Day-by-day elemzések kérése	

R11	A felhasználó megfelelő jelszó megadásával <i>Admin</i> felhasználóba léphet.	Admin módba lépés	
R12	<i>Admin</i> felhasználó tudja törölni az összes betöltött adatot.	Adathalmaz törlése	
R13	<i>Admin</i> felhasználó tudja változtatni a program által használt alapértelmezett mértékegységeket.	Alapértelmezett mértékegység változtatása	

### Nem funkcionális követelmények

A program futtatásához szükséges a .NET xx verziója. Szükséges továbbá egy legalább Windows 10 operációs rendszer telepítésére és futtatására képes hardverrel rendelkező személyi számítógép.

A program használatához szükséges egy, a személyi számítógéphez valamilyen interface-en (pl. HDMI, VGA, DVI-D, DisplayPort stb.) csatlakoztatott kijelző, valamint egy csatlakoztatott billentyűzet vagy egér.

# Rendszerterv

---

## A tervezett alkalmazás szöveges leírása

A *Meteorológiai adatbázis* feladatot megvalósító program parancssori környezetben fog futni, a felhasználó azon keresztül tud interaktálni vele. A programon belül, a használni kívánt felhasználó kiválasztása után egy menüstruktúrával fog működni, előre meghatározott parancsokkal lehet majd navigálni és a kívánt tevékenységeket elvégezni.

A szenzorok adatai XML formátumban állnak majd rendelkezésre, a program ennek beolvasására, valamint a fájlból az adatok importálására lesz képes. Amennyiben nem állnak rendelkezésre adatok, vagy más okból nem akarunk importálni adatokat, a program képes lesz generálni dummy adatokat paraméterek megadásával.

A felhasználó kiválasztásakor *User* jogosultságokat kapunk, amivel végrehajthatjuk a korábban említett generált vagy fájlból beolvasott adatok betöltését. Ezen kívül ezzel a felhasználótípussal további tevékenységekre is képesek leszünk.

Az adatok betöltése után meg tudjuk ezeket jeleníteni a parancssori környezetben. Itt feltüntetésre kerül az adatpont időpontja, értéke, illetve a hozzá tartozó mértékegység.

A betöltött adatokat tudjuk szűrni is, különböző paraméterek megadásával: időben kezdő- és végpont megadásával, valamint érték alapján maximum és minimum határok megadásával tudjuk majd az általunk kívánt adatokat megjeleníteni.

Az adatokból a programmal tudunk majd statisztikai kimutatásokat is generáltatni, ahol az átlagos minimum, maximum értékeket, adatpontok számát kérheti le. Továbbá lehetősége lesz ezeket az adatokat napi lebontásban is megtekinteni.

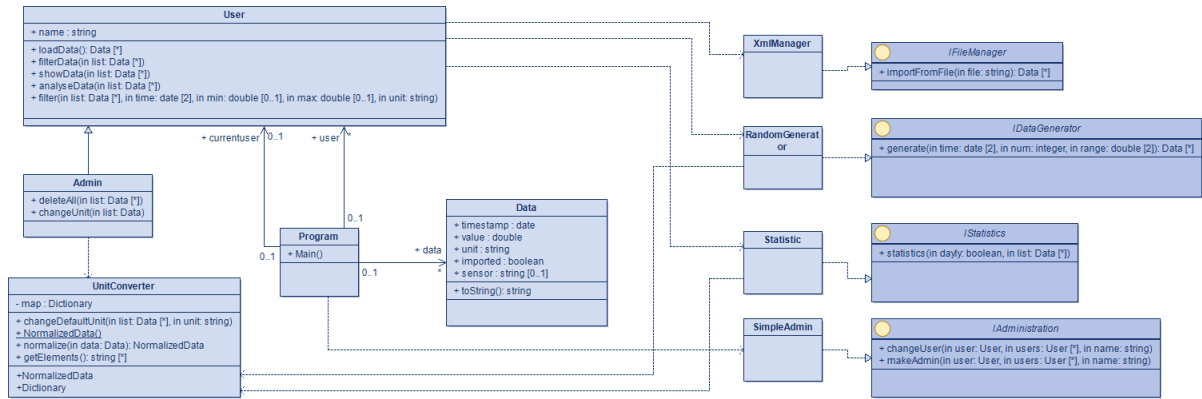
Lehetőség lesz a felhasználók adminisztrációjára. Itt lehetőség lesz felhasználót váltani, valamint *Admin* jogosultságot szerezni az adminisztrátori jelszó megadásával.

Az *Admin* jogosultságok megszerzése után a felhasználó el tudja végezni a korábban említett összes tevékenységet, ezeken felül pedig lehetősége lesz az alapértelmezett mértékegység megváltoztatására, valamint a betöltött adathalmaz törlésére.

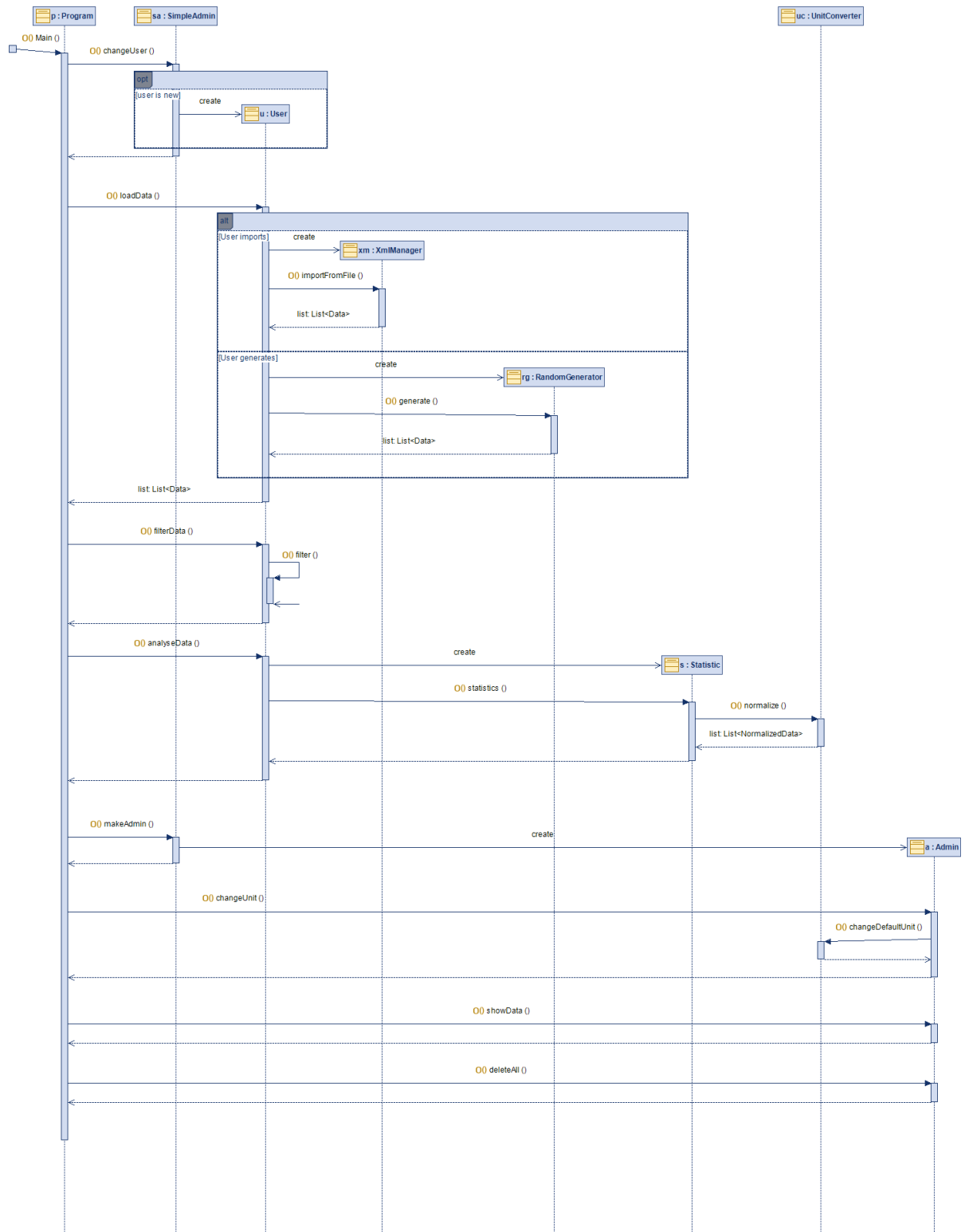
Minden menüpontban lehetőség lesz segítséget kérni (a lehetséges parancsokat kilistázni).

A parancsok megadására a parancssori környezet standard inputja lesz használható, illetve minden parancs eredménye a parancssori környezetben fog megjelenni szövegesen.

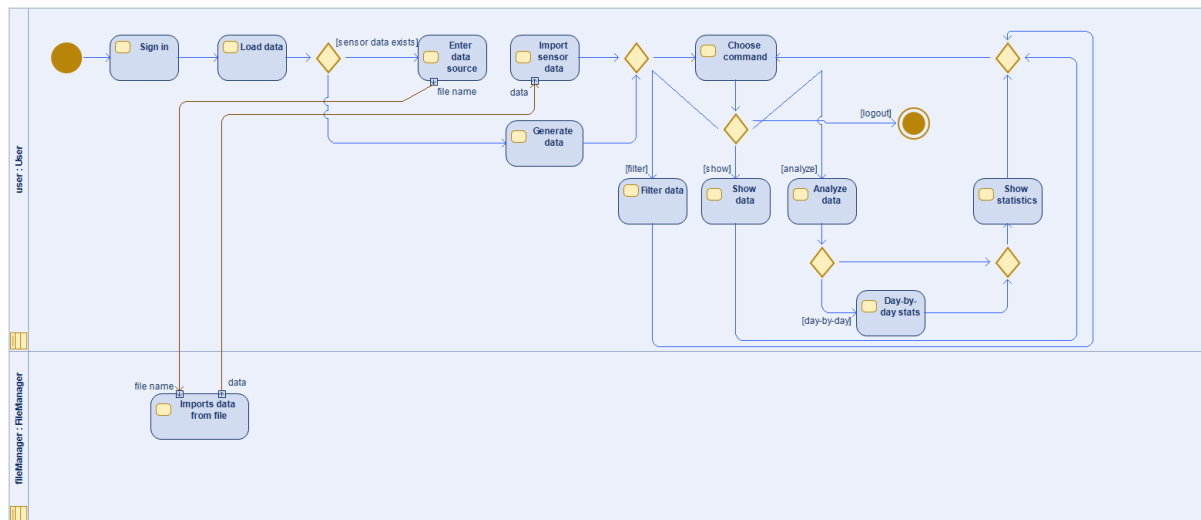
## Áttekintő osztálydiagram



## Egy tipikus lefutás szekvenciadiagramja



## Aktivitás diagram



## Modulok

A fejlesztés során törekszünk a SOLID elvek betartására. Ennek eredményeképp a különböző felelősségi köröket külön osztályokban, az egyes osztályok kiegészítését pedig interface-ekben implementáltuk.

### Vezérlés és megjelenítés

A vezérlés és megjelenítés nagy részét a *Program* osztály valósítja meg. A program futtatásakor ez jeleníti meg és olvassa be a felhasználó nevét, valamint a menüben való navigálást is ebben az osztályban találjuk. Itt tárolja a program listában az adatokat, valamint a felhasználókat, az aktuális felhasználót, valamint itt vannak meghívva a különböző osztályok metódusai.

### Felhasználók és elérhető tevékenységeik

A felhasználók által elérhető tevékenységek metódusai a *User* osztályban vannak megvalósítva (pl. a beolvasás, szűrés, adatok megjelenítése). Ennek kiegészítő interface-e az *IAdministration*, ami a felhasználóváltást, illetve az adminisztrátori módba lépést teszi lehetővé.

Az *Admin* osztály valósítja meg az adminisztrátori módban elérhető parancsokat (adathalmaz törlése, mértékegység váltása).

Az *IStatistics* interface a statisztikák elkészítését és megjelenítését valósítja meg.

Az *IDataGenerator* interface az adatok betöltésénél elérhető generálás opció metódusát valósítja meg.

### Adatok

Az adatok megvalósítását a *Data* osztály tartalmazza; a különböző attribútumok és a megjelenítéshez szükséges, attribútumok emberi fogyasztásra alkalmas szöveggé alakítása található itt.



Az adatokhoz hozzátartozik azoknak mértékegysége is. Ezeket a *UnitConverter* osztály valósítja meg. Itt található a mértékegységváltás logikája is.

Adatok generálásakor és beolvasásakor a *Data* osztály van példányosítva.

### Fájlkezelés

Az adatok fájlból való importálásának fő logikáját az *IFileManager* interface valósítja meg.

Az adatok importálásakor a *User* osztályban megvalósított metódus példányosítja ezt az interface-t, és így hívja meg annak importáló metódusát.

# Tesztelés

---

## Unit tesztek

### ReadFileTest():

- Beolvasunk egy létező fájlt és megnézzük helyesen töltődött-e be.
- Beolvasunk egy nem létező fájlt és teszteljük a helyes kezelést.
- Beolvasunk egy rossz adattal rendelkező fájlt és teszteljük a helyes hibás adat kezelést.

### DataGenerationTest():

- Generálunk adatokat és teszteljük megfelelő számú adat jött-e létre.
- Végig megyünk a generált adatokon és mindegyikre teszteljük helyes értékek generálódtak-e, megfelelő tartományban vannak.
- Teszteljük a 0 és negatív számú generálást.
- Vizsgáljuk a null érték kezelést.

### ChangeUserTest():

- Teszteljük az új felhasználó létrehozását. Valóban létrejön-e.
- Megnézzük helyesen működik-e több felhasználóval.
- Ha már meglévő felhasználó nevet adunk ne hozzon létre új felhasználót.
- Teszteljük az empty string, és null érték kezelést .

### ToAdmin():

- Teszteljük az adminná válás folyamatát.
- Ha helyes a jelszó admin lesz-e a felhasználóból.
- Ha helytelen nem szabad, hogy admin legyen.

### StatisticsExceptionTest():

- Teszteljük üres listával működik-e a statisztika készítése.
- Nem dob-e exceptiont a programunk.

### DeleteAllDataTest():

- Teszteljük helyesen törlődik-e a lista.
- Null listára nem szabad, hogy megálljon a program.

### ChangeUnitTest():

- Teszteljük alapértelmezett mértékegység váltásnál csak a megfelelő mértékegység változott-e, és hogy helyesen változott hozzá az érték.
- Megnézzük hogy az összes megfelelő adat változott-e.
- Teszteljük a hibás és null bemenetet.

### `DateTimeFilter()`:

- Teszteljük a szűrő datetime metttől-meddig bemenete összes iterációjának helyes működését.

### `ValueFilter()`:

- Teszteljük a szűrő value mennyitől-meddig bemenete összes iterációjának helyes működését.

### `UnitTest()`:

- Teszteljük, hogy a unit szűrő helyesen működik-e.

### `CombinedFilter()`:

- Teszteljük a 3 szűrő összes iterációját egyszerre.

### `NullTest()`:

- Teszteljük mi történik csupa null bemenetnél. Helyes null kezelés.

## Terheléses tesztek

### `ImportTest()`:

- Teszteljük 10000 adat importálását. Ne legyen túl lassú

### `GenerateTest()`:

- Teszteljük 10000 adat generálását. Ne legyen túl lassú

# Telepítési útmutató

---

## Rendszerkövetelmények

A program futtatásához szükséges egy Windows 10 vagy 11 operációs rendszerrel rendelkező, arra futtasára képes számítógép.

Szükséges továbbá a .NET SDK legalább 8.0-s verziója, valamint Visual Studio 2022 vagy újabb IDE.

## Letöltés, buildelés és futtatás

A program forráskódja a [https://github.com/settlers2000/szt\\_hazi\\_2](https://github.com/settlers2000/szt_hazi_2) linken elérhető publikus GitHub repository-ban található. Ennek letöltése után a solution fájlt megnyitva a fejlesztőkörnyezetben buildelni szükséges a projektet. Ezután futtatható a fejlesztőkörnyezetből. Az adatokat tartalmazó fájlnak a program mappájában kell lennie!

# Felhasználói dokumentáció

---

A program indítása után, és a futása során bármikor a *help* paranccsal tudjuk kilistázni az éppen elérhető parancsokat.

Az indítás után elsőként bejelentkeznünk szükséges. Ezt a *login* paranccsal, majd a felhasználónevünk megadásával tudjuk megtenni.

Hogy rendeltetésszerűen tudjuk használni a programot, első körben ajánlott az adatok betöltése. Ez lehetséges fájlból való importálással, vagy generálással. A *load* parancs kiadása után választhatunk a két opcióból: *generate* vagy *import*.

Az *import* opció választásakor meg kell adnunk a fájl nevét, kiterjesztéssel együtt. Ekkor a program beolvassa a benne található adatokat.

A *generate* opció választásakor meg kell adnunk az időtartamot, amin belül szeretnénk az adatokat generálni, valamint egy intervallumot a generált adatpontok értékeinek.

Miután betöltöttük az adatokat, lehetőségünk van azokat pontonként megtekinteni a *list* paranccsal. Ekkor a program a parancssori környezetben kilistázza az egyes adatpontok minden attribútumát.

A *filter* paranccsal szűrhetjük a betöltött adatokat. Itt meg kell adnunk kezdő- és végpontjával azt az időtartamot, amin belüli adatokat szeretnénk megtekinteni. Meg kell adnunk, továbbá, egy minimum, egy maximum értéket és hogy milyen mértékegységű adatokat kívánunk megtekinteni. Ezután a program kilistázza a feltételeknek megfelelő pontokat.

Statisztikát is kérhetünk a betöltött adatokból. Ezt az *analyse* paranccsal tudjuk megtenni. Itt lehetőségünk van *day-by-day*, tehát napi lebontásban kérni az elemzést. A kimutatás a teljes időtartamra vagy az egyes napokra megmutatja, hogy milyen mértékegységű adatból mennyi van, mi az adatok minimuma, maximuma és átlaga.

Lehetőségünk van felhasználót váltani, valamint admin jelszót megadni. Ezt az almenüt a *user* paranccsal érjük el, itt pedig a *change* és *password* opciók közül választhatunk.

A *change* parancs kiadása után kijelentkeztet a program, és újra beléphetünk egy másik felhasználóba nevének megadásával.

A *password* parancs kiadása, valamint az admin jelszó megadása után aktuális felhasználónk adminisztrátori jogosultságokkal ruházódik fel. Itt a korábban elért parancsok továbbra is elérhetőek maradnak.

Adminisztrátorként lehetőségünk van törölni a betöltött adathalmazt a *delete* paranccsal. Miután kiadjuk ezt a parancsot, törlődnek a betöltött adatpontok.

Adminisztrátori kiváltság továbbá az alapértelmezett mértékegységek megváltoztatása is, amit a *unit* paranccsal tudunk megtenni. A parancs kiadása után megadhatjuk, hogy mit szeretnénk az új alapértelmezett mértékegységnek, amit a program a továbbiakban használni fog.

A tevékenységünk befejeztével az *logout* paranccsal tudunk kijelentkezni az aktuális felhasználóból, majd az *exit* paranccsal tudjuk bezárni a programot.