**Záró IKT feladat**

**FELADAT**: Készítsünk üzenetküldő alkalmazást.

A feladatunk, hogy egy minden napra elkészítsük az elküldendő üzeneteket. Az üzenetek *json* típu állományban kell eltárolni a minta szerint:

*[*

*{*

*"system": "windows",*

*"firstName": "Cally",*

*"lastName": "Sprowell",*

*"mobileNumber": "771-488-9158",*

*"message": "Nunc rhoncus dui vel sem. Sed sagittis. Nam congue, risus semper porta volutpat, quam pede lobortis ligula, sit amet eleifend pede libero quis orci. Nullam molestie nibh in lectus. Pellentesque at nulla."*

*},*

*{*

*"system": "ios",*

*"firstName": "Ingaberg",*

*"lastName": "Beasleigh",*

*"mobileNumber": "935-876-5089",*

*"message": "Nulla mollis molestie lorem. Quisque ut erat. Curabitur gravida nisi at nibh. In hac habitasse platea dictumst. Aliquam augue quam, sollicitudin vitae, consectetuer eget, rutrum at, lorem."*

*}*

*]*

A programunkat a felhasználó egy menün keresztül tudja vezérelni:

1 – Új üzenet hozzáadása. Ekkor egy teljessen új üzenetet kell rögzíteni egy megadott dátumhoz. Ezt az üzenetet a data/*messages\_{megadott dátum}.json* állományba kell elmenteni. A megadott dátum nem lehet korábbi mint a aktuális nap. Ha nem létezik file ezzel a névvel, akkor létre kell hozni.

2 – Üzenet törlése telefonszám alapján egy, a felhasználó álltal megadott napból. Ez a nap nem lehet későbbi mint az aktuális nap.

3 – Üzenetek küldése. Mindig az aktuális napra vontakozó üzeneteket kell elküldeni. Ha nincs ilyen, akkor jelezni kell a felhasználónak, hogy az aktuális napra nincsennek üzenetek.

4 – Jelentés megjelenítése egy, a felhasználó álltal megadott napra. A nap nem lehet későbbi mint az aktuális nap. E jelentés adat forrása a *report\_{ megadott dátum }.json*.

Az üzeneteket REST API kersztül kell elküldeni a szerver felé, mely továbbítja majd az üzeneteket. (mellékelve megaptuk a tesztelésre szolgáló REST API-t)

A program írásakor figyelembe kell venni, hogy mely rendszerre akarjuk az üzenetet küldeni, ugyanis 3 típusú rendszerre tudunk üzenetet küldeni: iOS, Android és Windows. Minden egyes rendszer üzenet küldése más és más végponton (*endpoint*) történik mivel minden egyes rendszer más és más *string* tartalmú üzenetet vár.

Az üzenet fomátumai:

* **iOS**: “key”: “value” | “key”: “value” | …
* **Android**: “key”: “value” # “key”: “value” # …
* **Windows**: “key”: “value” \* “key”: “value”

A elküdött üzenet eredméyéről a szerver választ küld, melyben jelzi, hogy az üzenet küldése sikeres volt e, ha nem a hiba üzenetét és a probálkozás időpontját.

A válaszokat ki kell gyüjteni különboző szöveges állományokba. A sikertelenül elküldött üzeneteket a *logs/not-delivered\_{üzenet küldés napja}.txt* szöveges állományba, még a sikeressen elküldött üzeneteket a *logs/delivered\_{üzenet küldés napja}.txt* szöveges állományba.

A *report\_{ megadott dátum }.json* állományba készítsünk statisztikát. A filenak tartalmaznia kell hogy, a hibára futott üzeneteknél mely hibák fordultak elő és hányszor, valamint a sikeressen elküldött üzenetek számát is. Minta:

{

“errors”: [

{

“reason”: “Sikertelen uzenet”,

“count”: 33

},

{

“reason”: “Nem a mefelelo a megadott rendszer”,

“count”: 1

},

{

“reason”: “Az uzenet nem megfelelo formatumu”,

“count”: 33

},

],

“success”: 10,

“date”: 2024. 04. 24. 22:39:48

}

segédanyag:

[Factory Method in C# / Design Patterns (refactoring.guru)](https://refactoring.guru/design-patterns/factory-method/csharp/example)

[The Factory Method Design Pattern in C# | by Charles | Medium](https://medium.com/@cizu64/the-factory-method-design-pattern-in-c-6ee65139238d)

[How to serialize JSON in C# - .NET | Microsoft Learn](https://learn.microsoft.com/en-us/dotnet/standard/serialization/system-text-json/how-to)

[Work with files and directories in a .NET app - Training | Microsoft Learn](https://learn.microsoft.com/en-us/training/modules/dotnet-files/)

[C# Files (With Examples) (programiz.com)](https://www.programiz.com/csharp-programming/files)

[C# Files & Directories (tutorialsteacher.com)](https://www.tutorialsteacher.com/csharp/csharp-file)

[The nameof expression - evaluate the text name of a symbol - C# | Microsoft Learn](https://learn.microsoft.com/en-us/dotnet/csharp/language-reference/operators/nameof)