

RENware Software Systems

Propunere tehnica

Cuprins:

- [Propunere tehnica](#)
 - [Aria de cuprindere](#)
 - [Considerente generale de securitate](#)
 - [Considerente generale privind bazele de date proprii sistemelor](#)
 - [Considerente generale privind auditarea informatiilor](#)
- Client: n/a - not public
- Data: 2023-Noiembrie



Codificarea documentelor

- codificarea numelor documentelor si a proceselor este facuta in conformitate cu metodologia [RENware SDEVEN](#)

Cuprins:

- [Propunere tehnica](#)
 - [Aria de cuprindere](#)
 - [Considerente generale de securitate](#)
 - [Considerente generale privind bazele de date proprii sistemelor](#)
 - [Considerente generale privind auditarea informatiilor](#)

Aria de cuprindere

Solutiile propuse prin aceasta propunere tehnica sunt:

- **INVOICEtoROefact** (code-name `api_to_roefact`) integrare Sistemul National de Facturi Emise RO e-Factura [descriere si cerinte aici](#)
- **PayValidaBoa** (code-name `payments_validation_board`) Flux aprobare facturi primite pentru ordonantare la plata [descriere si cerinte aici](#)

In continuare se prezinta o serie de considerente generale valabile pentru toate sistemele din aria de acoperire.

Considerente generale de securitate

- **(RSEC-01)** fisierele de configurare a sistemelor (fiind format text `UTF-8`) vor avea ca `owner` un utilizator dedicat sistemului respectiv sau utilizatorul `root`. Numai acesti doi utilizatori pot avea acces `RW` la aceste fisiere
- **(RSEC-02)** toate documentele de provenienta externa sistemelor vor fi "purtatoare" ale unui certificat digital ce **atesta validitatea documentelor**. Acest certificat va fi de preferinta de tip "*semnatura electronica*" dar nu obligatoriu calificata. Este suficient un simplu certificat (cheie) tip `RSA` generat intern si distribuit utilizatorilor autorizati sa emita documentele respective. O copie a certificatului (sau a certificatelor daca se vor emite mai multe) ce atesta validitatea unui document va sta pe server in locatii ce sunt conforme cu *RSEC-01*

Considerente generale privind bazele de date proprii sistemelor

- **(DBS-01)** bazele de date vor contine o cheie primara "*real primara*" (adica avind toate caracteristicile tehnice pentru `PK` in sensul uzual cunoscut din teoria bazelor de date). Aceasta cheie va fi de tip `Char(32)` reprezentind tipul `uuid4` (cunoscut si ca `guid`) convertit la sir de caractere `UTF-8` si reversibil ca transformare din `string` in `uuid4`. Aceasta cheie va fi generata automat si intretinuta de sistem deservind scopuri pur tehnice de *referentiere si relationare* a datelor. Modificarea manuala nu este permisa putind genera situatii de hazard.
- **(DBS-02)** bazele de date vor contine si o alta "*cheie primara uman recongnoscibila*" (`AK` in teoria bazelor de date) utilizata in scop de **recunoastere si regasire** a informatiei de catre utilizatori. Aceasta cheie va avea urmatoarele catacterisrici:
 - va fi *unica*, tip `Char(10)` (limitarea lungimii se va aplica la introducerea datelo si nu in baza de date)
 - *agnostic case*, nu se va face diferenta intre litere mari sau mici (pentru a evita confuziile)
 - *obligatorie* iar daca utilizatorul "nu o doreste" se va default-a la `PK-u1` anterior
- **(DBS-03)** bazele de date vor fi intr-unul din formatele: **(a) relational** sau **(b) JSON standard**. Pentru bazele de date in format relational va fi preferata o solutie de SGBD tip open source matura, intretinuta in urmatoarea ordine de aplicare:
 - 1. `SQLite` pentru baze de date ce nu vor depasi 10,000 de inregistrari
 - 2. `PostgreSQL` pentru baze de date ce se esttimeaza ca vor depasi 10,000 de inregistrari
 - 3. `MariaDB` pentru baze de date ce se esttimeaza ca vor depasi 10,000 de inregistrari
 - prima varianta va fi preferata datoritra "portabilitatii datelor"
 - a treia varianta este enumerata ca optiune preferata a utilizatorului la varianta 2.
- **(DBS-04)** bazele de date vor folosi numai cimpuri formate standard, clasice si elemetare:
 - sir de carectere (`CHAR` sau `VARCHAR`)
 - numere intregi cu semn (`INTEGER`)

- numere reale cu semn (`Float`)
- numere combinate a caror valoare poate fi intreg sau real (`Number`)
- valori logice sub forma intreg cu semn astfel: `1` pentru TRUE si `0` sau `NULL` pentru FALSE
- valori logice sub forma de caracter astfel: prima litera din lista `[Y, y, D, d, T, t]` pentru TRUE si orice altceva inclusiv `NULL` pentru FALSE
- **(DBS-05)** in cazul bazelor de date relationale, integritatile referentiale vor fi evitate la maximum prin intretinerea datelor numai cu ajutorul aplicatiei sau in cazull necesitatii modificarii manuale a datelor, aceasta modifica re sa fie efectuata numai de personal calificat
- **(DBS-06)** informatiile de tip data-timp (data, ora, etc...) vor fi stocate de preferinta sub forma de `String` in formatul ISO: `YYYY-MM-DD HH:MM:SS.nnnnn` .
- **(DBS-07)** informatii de data-timp vor fi stocate avind valori agnostice de "Time Zone" adica vor fi considerate `UTC` lucru care va permite comparabilitatea acestora indiferent de locatia /zrna de timp de unde au fost generate.

Considerente generale privind auditarea informatiilor

- Cimpurile de audit ce indica utilizatori:
 - **(AUD-01)** pentru informatiile `CONSTIENT` `GENERATE DE UTILIZATORI` (adica generate prin activarea unor controale vizuale, prin lansarea manuala a unei aplicatii, etc), aceste cimpuri vor contine **numele tip `username` al utilizatorului folosit pentru autentificarea in sistem**
 - **(AUD-02)** pentru informatiile `GENERATE DE SISTEM` la rulari automate, periodice, de verificare, de validare, etc, aceste cimpuri vor contine textul `system` (pentru a evita confuzii cu utilizatori reali la nivel de sistem de operare)
- **(AUD-03)** Cimpurile de audit ce indica date calendaristice vor respecta standardul ISO fiind in formatul maximal `YYYY-MM-DD hh:mm:ss`

Last update: February 3, 2024