



UNIVERSITATEA TEHNICĂ "GHEORGHE ASACHI" IAȘI
FACULTATEA DE AUTOMATICĂ ȘI CALCULATOARE
SPECIALIZAREA CALCULATOARE ȘI TEHNOLOGIA INFORMAȚIEI

PROIECT: BAZE DE DATE

Gestiunea conturilor de League of Legends

Coordonator,
Asist.drd.ing. Cătălin Mironeanu

Student,
Harpa Andrei-Alexandru

Iași 2020

Titlu proiect: Gestiunea conturilor de League of Legends

Analiza, proiectarea și implementarea unei baze de date și a aplicației aferente care să modeleze elementele de bază ale mai multor conturi de League of Legends (rank, campioni, skin-uri, club și resurse)

Descrierea cerințelor și modul de organizare al proiectului:

Scopul acestei teme este de a crea o bază de date în care vor fi introduse detaliile principale ale jucătorilor de League of Legends. Fiecare user va avea informații legate de numele și adresa de email cu care și-a creat contul, deși în realitate o persoană își poate crea mai multe conturi cu aceeași adresă de email, în cadrul acestui proiect am decis ca un jucător să poată avea un singur cont per adresă email. Pe lângă numele necesar creării contului, mai este necesar un nume (summoner name) pentru identificarea jucătorului în joc. Inițial toate conturile vor începe de la nivelul 1 și vor putea ajunge la un nivel de peste aproximativ 3000 (încă nu se știe nivelul maxim pe care-l poate atinge un jucător). Pe lângă nivel, jucătorii sunt diferențiați și prin intermediul unor rank-uri, cele două rank-uri de baza sunt solo-duo și flex, la rândul lor, aceste rank-uri sunt împărțite pe mai multe divizii (cea mai mică divizie este Bronze și are 5 subdivizii (5 – 1, mai avem Silver, Gold, Platinum și Diamond care au tot 5 subdivizii, și mai rămân diviziile Master, Grand-Master și Challenger, care nu sunt împărțite pe subdivizii).

Pentru a putea juca jocul și avansa în rank, jucătorii au nevoie de campioni. Acești campioni pot fi cumpărați cu RP (riot points) sau BE (blue essences), bineînțeles prețurile campionilor sunt împărțite în câteva categorii. În afara jocului, jucătorii au posibilitatea de a vedea detaliile de bază ale campionilor (atac, armura etc.), aceste detalii pot suferi ajustări în cazul în care un campion este prea puternic sau prea slab.

Deasemenea, jucătorii vor avea posibilitatea de a colecționa skin-uri pentru campioni, aceste skin-uri au la rândul lor un preț în RP și unul în OE (orange essences) dar și Gemstones (skin-urile de tipul Mythic pot fi cumpărate doar cu gemstones), prețurile sunt împărțite pe categorii, aceste categorii sunt în strânsă legătură cu tipul skin-ului (Standar, Deluxe, Ultimate etc.). De menționat faptul că un campion poate avea mai multe skin-uri.

Mai există și posibilitatea ca un jucător să facă parte dintr-un club, chiar dacă în realitate jucătorul nu este obligat să facă parte dintr-un club, baza de date este modelată astfel încât un jucător trebuie să facă parte dintr-un club. Acest club are un nume unic și poate avea un număr variabil de jucători.

Fiecare jucător are un inventar, în această bază de date inventarul va conține doar cutii de campioni, cutii de skin-uri, chei cu ajutorul cărora vom putea deschide cutii, și fragmente de chei, prin combinarea a trei fragmente de chei se obține o cheie normală.

Tehnologii utilizate pentru crearea aplicației

1. Limbajul de programare: Python
2. Sistemul de gestionare a bazelor de date Oracle
3. Module utilizate:
 - i. cx_Oracle (pentru a realiza conexiunea la baza de date)
 - ii. Tkinter (pentru realizare interfetei grafice)

Conexiunea aplicației la baza de date s-a realizat astfel:

```
host = 'bd-dc.cs.tuiasi.ro'
port = '1539'
user = 'bd021'
password = 'lab05'
service = 'orcl'
```

```
def connect(self):
    conn_str = self.user + '/' + self.password + '@' + self.host \
               + ':' + self.port + '/' + self.service
    cx_Oracle.init_oracle_client(lib_dir=r"C:\oracle\instantclient_19_9")
    self.conn = cx_Oracle.connect(conn_str)
    self.cursor = self.conn.cursor()
```

Descrierea funcțională a aplicației

- Achiziția de campioni
- Achiziția de skin-uri
- Adăugarea unor noi campioni în joc
- Adăugarea unor noi skin-uri în joc
- Adăugarea de resurse

Descrierea detaliată a entităților și a relațiilor dintre tabele:

Tabelele din această aplicație sunt:

- Lol_user
- Resources
- Inventory
- Club
- User_champion
- Champion
- User_skin
- Skin

În proiectarea acestei baze de date s-au identificat tipurile de relații 1:1, 1:n, n:1 și n:n.

Între tabelele **lol_user** și **resources** există o relație de tip 1:1, deoarece un user are propriile sale resurse(RP, BE, OE), chiar dacă există posibilitatea ca doi useri să aibă aceleași cantități, atunci când un user achiziționează un campion sau skin, resursele se vor modifica doar la user-ul care a realizat tranzacția.

Între tabelele **lol_user** și **inventory** există o relație de tip 1:1, întrucât este aceeași situație ca și la **lol_user - resources**.

Între tabelele **lol_user** și **club** există o relație de tipul n:1, deoarece există posibilitatea ca mai mulți jucători să facă parte din același club, dar nu este posibil ca un jucător să facă parte din mai multe cluburi în același timp.

Între tabelele **champion** și **skin** există o relație de tipul 1:n. Fiecare campion are posibilitatea de a deține mai multe skin-uri, dar este imposibil ca un skin să aparțină mai multor campioni.

Între tabelele **lol_user** și **champion** există o relație de tipul n:n. Această relație este evidentă, deoarece, un campion poate fi deținut de mai mulți jucători, iar un jucător poate avea mai mulți campioni. În urma acestei relații va rezulta un nou tabel, **user_champion**, acest tabel va conține doar tuple de tipul (user_id, champion_id).

Inițial, între tabelele **lol_user** și **skin** exista doar o relație de n:n, în urma căreia rezulta tabela **user_skin**, care conținea tupla (user_id, skin_id), problema este că nu era suficient de clar pe ce campioni deține un user skin-urile, așadar am descompus relația astfel: 1:n între **lol_user** și **user_skin**, n:1 între **user_skin** și **champion** și relația n:1 între **user_skin** și **skin**, prin urmare tabela **user_skin** va conține tuple de tipul (user_id, champion_id, skin_id), această tuplă va conține suficiente informații referitoare la ce skin-uri are un user și pe ce campioni.

Constrângeri utilizate:

Lol_user:

- **name** și **summoner_name** trebuie să aibă o lungime de cel puțin 5
- **email** trebuie să fie sub o formă standard, și să fie unic
- **lvl** trebuie să fie cel puțin 1
- **rank_flex** și **rank_solo_duo** vor putea avea valori doar dintr-o gamă prestabilită(B1 – B5, S1 – S5, G1 – G5, P1 – P5, D1 – D5, M, GM, C, U)

Resources:

- **riot_points, blue_essence** și **orange_essence** nu pot avea valori negative

Inventory:

- **champion_chests, skin_chests, chest_keys, key_fragments** și **gemstones** nu pot avea valori negative

Club:

- **name** trebuie să aibă o lungime de cel puțin 4
- **number_of_members** va fi cel puțin 1
- **create_date** trebuie să fie mai mică decât SYSDATE

Champion:

- **champion_name** trebuie să fie de lungime cel puțin 3, și să fie unic
- **price_rp, price_bluesence, health, armor, magic_resist** și **attack_damage** vor putea avea valori doar dintr-o gamă prestabilită de valori
- **release_date** trebuie să fie mai mică decât SYSDATE

Skin:

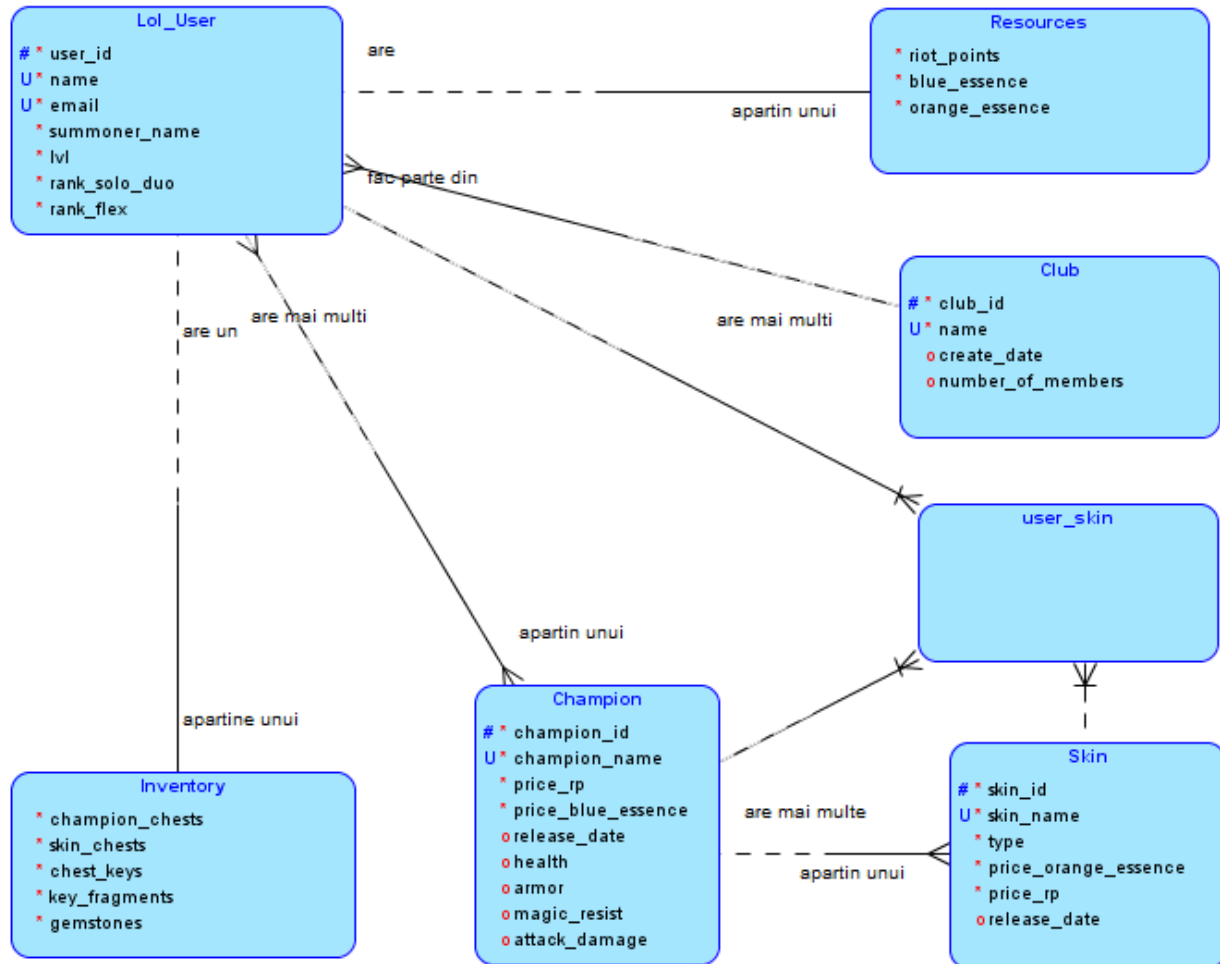
- **skin_name** trebuie să fie unic
- **type, price_rp** și **price_orange_essence** trebuie să aibă valori dintr-o gamă prestabilită
- **release_date** trebuie să fie mai mică decât SYSDATE

Autoincrement:

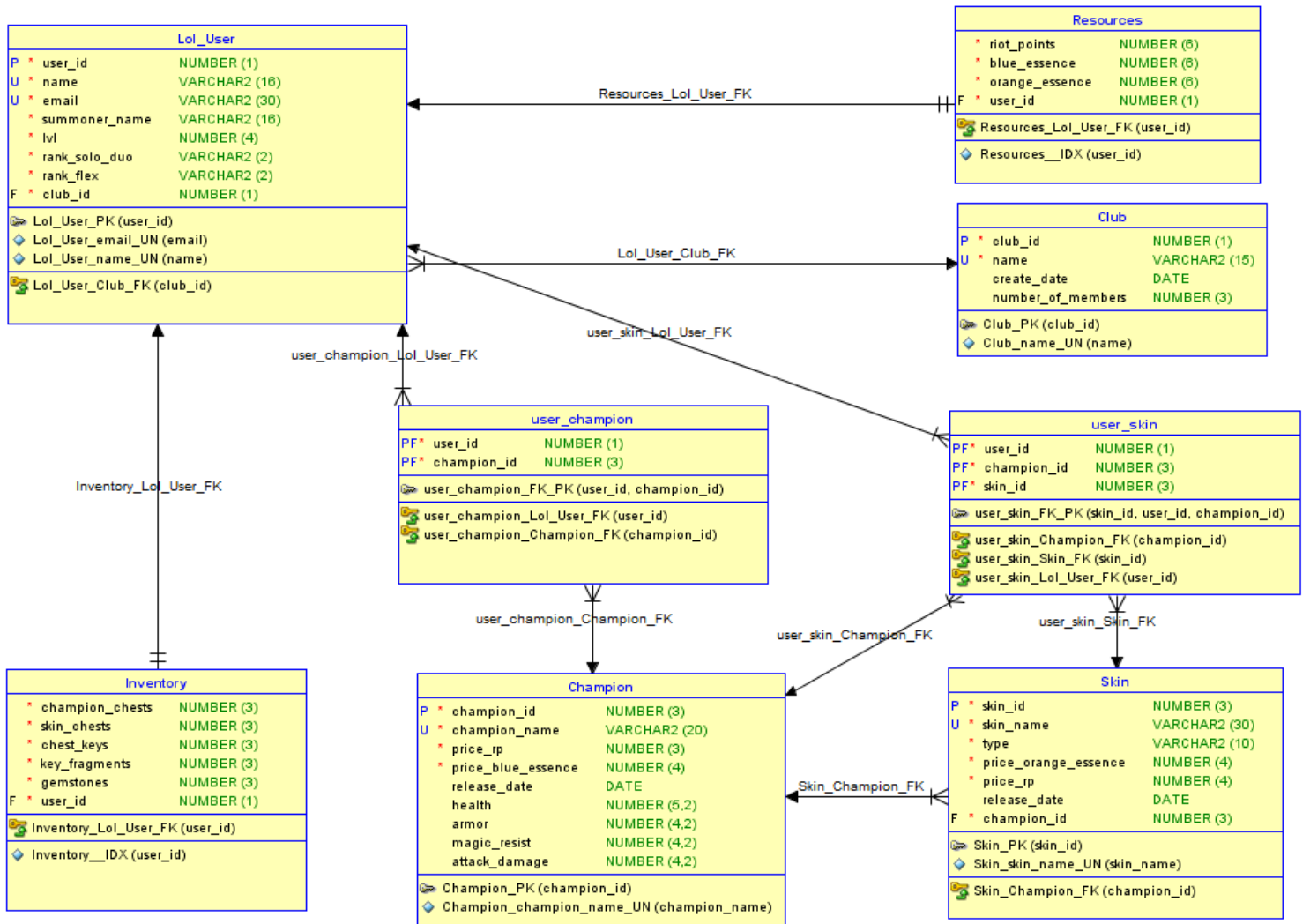
Această opțiune a fost utilizată pentru generarea automată a câmpurilor:

- **user_id, club_id, champion_id** și **skin_id**

Diagramă Model Logic:



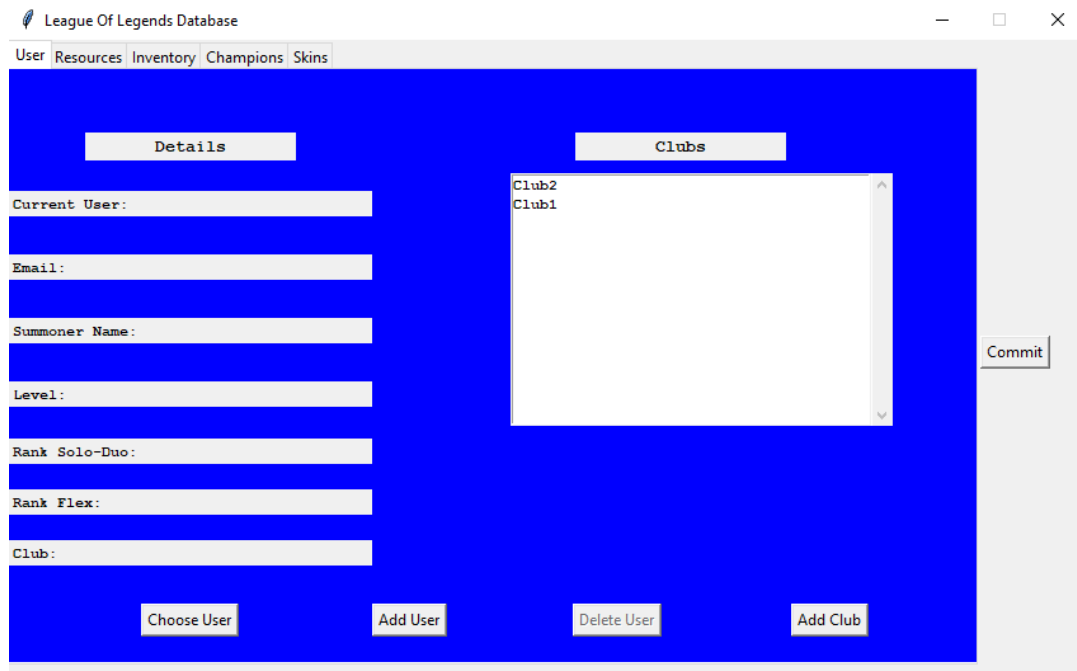
Diagramă Model Relațional:



Prezentarea aplicației(front-end):

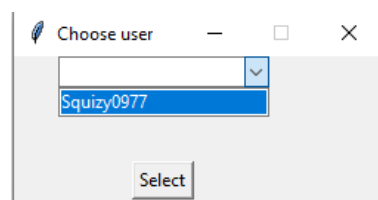
Partea de front-end a fost realizată cu ajutorul modulului tkinter și constă într-o serie de tab-uri.

De exemplu, vom detalia tab-ul User, întrucât celelalte tab-uri au implementate aceleași principii de funcționare

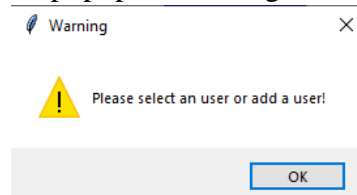


Prin intermediul tab-ului User, se pot vizualiza detaliile user-ului curent, lista de cluburi înregistrată în baza de date, dar și o serie de butoane cu ajutorul cărora putem realiza diferite operații:

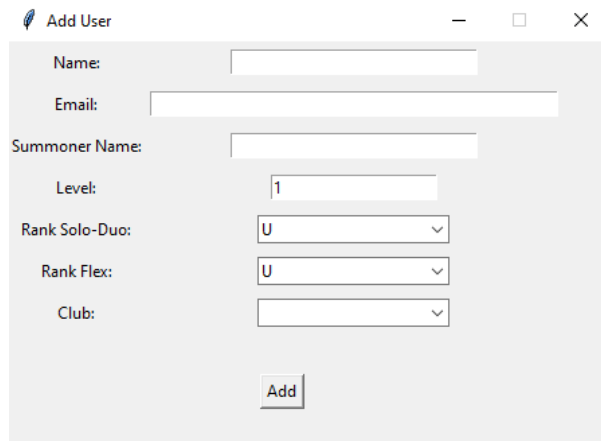
- **Choose User:** are rolul de a selecta user-ul curent



Dacă se apasă butonul select, dar nu avem selectat niciun user, va apărea un popup de warning:



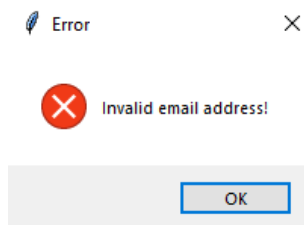
- **Add User:** în urma apăsării acestui buton, va apare un popup ce conține o listă de câmpuri ce reprezintă detaliile necesare pentru crearea unui nou user:



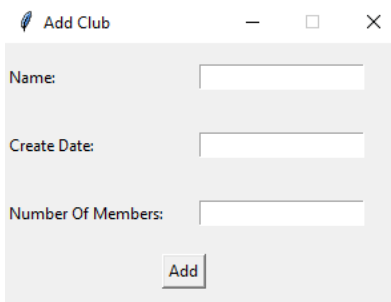
The 'Add User' dialog box features the following fields and controls:

- Name:** Text input field.
- Email:** Text input field.
- Summoner Name:** Text input field.
- Level:** Text input field with the value '1'.
- Rank Solo-Duo:** Dropdown menu with 'U' selected.
- Rank Flex:** Dropdown menu with 'U' selected.
- Club:** Dropdown menu.
- Add:** Button at the bottom center.

Dacă cel puțin un câmp nu este completat conform constrângerilor existente, va apare un mesaj de eroare, de exemplu introducerea unei adrese de email invalidă:



- **Delete User:** dacă este selectat un user, acest buton devine activ, și prin apăsarea lui vom sterge un user din baza de date
- **Add Club:** prin apăsarea acestui buton, va apărea un popup similar cu cel de la 'Add User':



The 'Add Club' dialog box features the following fields and controls:

- Name:** Text input field.
- Create Date:** Text input field.
- Number Of Members:** Text input field.
- Add:** Button at the bottom center.

Tab-ul Resources:

Prin intermediul acestui tab vom putea observa detaliile legate de resursele user-ului curent:

The screenshot shows a web application window titled "League Of Legends Database". At the top, there is a navigation bar with tabs: "User", "Resources", "Inventory", "Champions", and "Skins". The "Resources" tab is currently selected. The main content area has a pink background and a header labeled "Resources". Below the header, there are three input fields, each with a value of "0" and a corresponding "Add" button: "Riot Points:" with "Add riot points", "Blue Essences:" with "Add blue essence", and "Orange Essences:" with "Add orange essence". On the right side of the form, there is a "Commit" button.

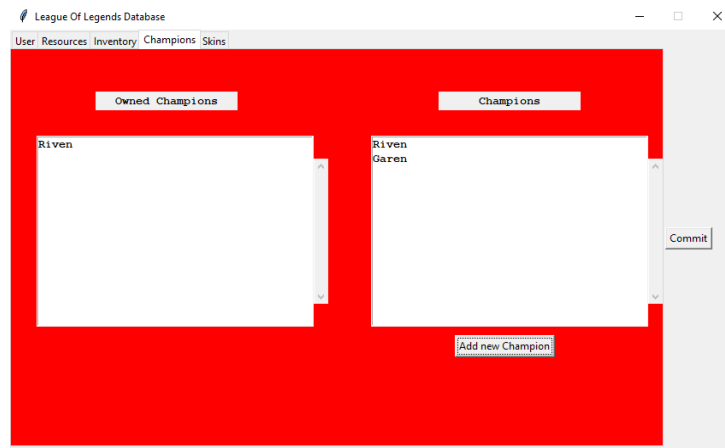
Tab-ul Inventory:

Prin intermediul acestui tab vom putea observa detaliile legate de inventarul user-ului curent:

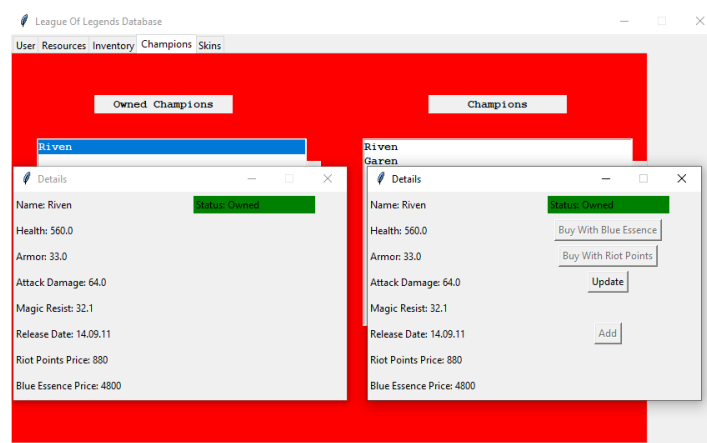
The screenshot shows the same web application window, but now the "Inventory" tab is selected. The main content area has a purple background and a header labeled "Inventory". Below the header, there are five input fields, each with a value of "0" and a corresponding "Add" button: "Champion Chests:" with "Add Champion Chests", "Skin Chests:" with "Add Skin Chests", "Chest Keys:" with "Add Chest Keys", "Key Fragments:" with "Add Key Fragments", and "Gemstones:" with "Add Gemstones". On the right side of the form, there is a "Commit" button.

Tab-ul Champions:

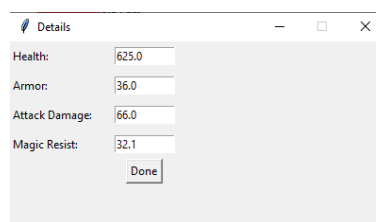
Prin intermediul acestui tab, vom putea vizualiza o listă cu toți campionii înregistrați în baza de date, dar și lista de campioni pe care îi deține user-ul curent, dacă nu este selectat un user, lista respectivă va fi goală:



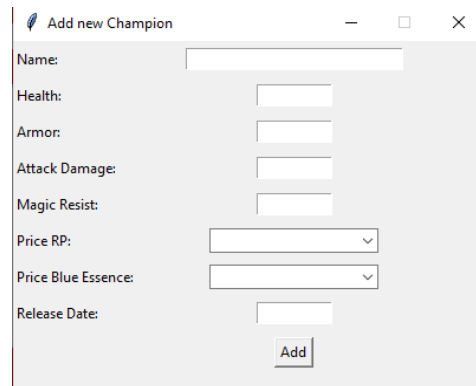
Pentru a vizualiza detaliile unui campion, putem acționa cu un dublu click asupra numelui campionului, în funcție de lista asupra caruia vom da dublu click, vom avea două popup-uri cu câteva diferențe:



Dacă selectăm un campion de pe lista generală, popup-ul va conține și o serie de butoane prin intermediul cărora vom putea adăuga un campion la lista user-ului curent, sau pentru a da un update atributelor campionului:

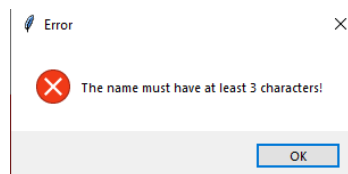


Mai există și buton ‘Add New Champion’, dacă acționăm asupra acestui buton, va apărea un popup similar cu cel de la butonul ‘Add User’:



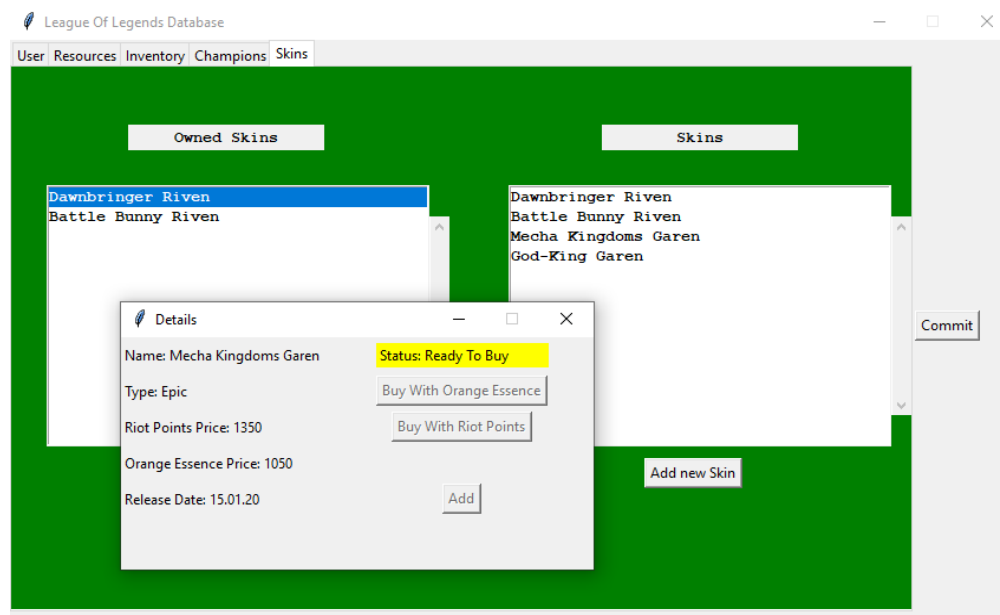
A dialog box titled 'Add new Champion' with a feather icon and standard window controls. It contains the following fields: Name (text input), Health (text input), Armor (text input), Attack Damage (text input), Magic Resist (text input), Price RP (dropdown menu), Price Blue Essence (dropdown menu), and Release Date (text input). An 'Add' button is located at the bottom right.

Deasemenea, nerespectarea constrângerilor existente va duce la apariția unui mesaj de eroare, ca de exemplu:



Tab-ul Skins:

Acest tab este similar cu tab-ul Champions, doar că prin intermediul lui vom putea gestiona skin-urile:



Finalizarea tranzacției:

Pentru a putea finaliza o tranzacție am adăugat butonul 'Commit', acesta este disponibil din orice tab, și are rolul de a modifica definitiv (corespunzător acțiunilor realizate prin intermediul aplicației) datele din baza de date.

The screenshot displays the 'League Of Legends Database' application window. It features a tabbed interface with 'User', 'Resources', 'Inventory', 'Champions', and 'Skins'. The 'User' tab is active, showing a 'Details' section on the left and a 'Clubs' section on the right. The 'Details' section contains several text input fields for user information: 'Current User' (Squizy0977), 'Email' (andy_alexandru23@yahoo.com), 'Summoner Name' (Squizy09), 'Level' (1), 'Rank Solo-Duo' (U), 'Rank Flex' (U), and 'Club' (Club1). The 'Clubs' section shows a list of clubs: 'Club2' and 'Club1'. A 'Commit' button is highlighted with a red box on the right side of the interface. At the bottom, there are four buttons: 'Choose User', 'Add User', 'Delete User', and 'Add Club'.