Documentatie Backend

**Informatii generale:**

* Toate requesturile primesc parametrii prin POST.
* Variabila status va fi intoarsa de toate requesturile si va avea semnificatia:
  + status = -1 Eroare de server
  + status = 0 Eroare cauzata de sender(parametrii lipsa etc.)
  + status = 1 Rulare cu succes, restul valorilor intoarse vor avea valori corecte
* Raspunsul este un obiect de tip JSON
* **checkemail.php**

**Serverul primește o adresa de email și verifica existenta acesteia în baza de date a utilizatorilor.**

Parametrii:

1. *user-email:* Email-ul utilizatorului

Raspuns:

1. *email-used*: **0**(Emailul nu este folosit); **1** (Emailul este folosit)

* **login.php**

**Serverul primește o adresa de email și o parola și verifica dacă acestea sunt asociate unui utilizator existent. În cazul în care sunt returnează numele și prenumele utilizatorului.**

Parametrii:

1. *user-email:* Email-ul utilizatorului
2. *user-password:* Parola utilizatorului

Raspuns:

1. *correct-credentials:* **1** (Autentificare cu succes); **0** (Email-ul sau parola sunt greșite)
2. *UID* – Codul utilizatorului **(Pentru correct-credentials=1**)
3. *user-firstname -* Numele utilizatorului **(Pentru correct-credentials=1**)
4. *user-lastname -* Prenumele utilizatorului (**Pentru correct-credentials=1**)
5. *user-type –* Tipul utilizatorului (**Pentru correct-credentials=1**)

* **register.php**

**Serverul primește o adresa de email, o parola, numele și prenumele unui utilizator și încearcă crearea unui cont nou. Daca account-created = 1 nu conteaza daca emailul este trimis sau nu, contul este creat.**

Parametrii:

1. *user-email --* Email-ul utilizatorului
2. *user-password --* Parola utilizatorului
3. *user-firstname --* Numele utilizatorului
4. *user-lastname* -- Prenumele utilizatorului

Raspuns:

1. *account-created*: **1** (Contul a fost creat); **0** (Contul nu a putut fi creat)
2. *email*-sent: **1** (Email trimis); **0** (Eroare email)

* **addanimal.php**

**Serverul primește numele, descrierea, tipul(caine/pisica ...) si rasa animalului si il adauga la animale + il leaga pe animal de utilizator ca (pet – owner) si creeaza un request de dare spre adoptie ! Aici va trebui probabil sa modificam pentru a putea adauga animalul si pentru cazare.**

Parametrii:

1. *security\_code* – 8981ASDGHJ22123 **(Doar pentru Android)**
2. *pet\_name –* Numele animalului
3. *pet\_description --* Descrierea
4. *pet\_type –* Tipul animalului ( **int --**  **Vezi ANEXA 1**)
5. *pet\_breed* – Rasa animalului
6. *pet\_age* – Varsta animalului
7. *UID* – Codul utilizatorului care face adaugarea ( **Doar pt Android )**

Raspuns:

1. *status: -* Daca status e 1 a fost adaugat, daca status e -1 eroare

* **getanimals.php**

**Serverul primeste tipul de request si trimite animalele care se potrivesc acelui request.**

**Intoarce animalele in un vector denumit *animals.***

***!!Cu galben am specificat parametrii specifici intoarcerii animalelor unui utilizator!!***

Parametrii:

1. *security\_code* – 8981ASDGHJ22123 **(Doar pentru Android)**
2. *request\_type –* 0 Întoarce toate animale disponibile spre adoptie
   1. *pet\_type –* Tipul animalelor care vor fi intoarse **(Vezi ANEXA 1)**
3. *request\_type –* 1 Întoarce toate animale unui utilizator
   1. *UID –* Codul utilizatorului (**Doar pt Android**)

Raspuns:

1. *nr\_animals: -* Numarul de animale aflate in vector
2. *animals:* - Vector de animale(are pozitii de la 0 la *nr\_animals*-1)

*Variabile disponibile:*

* 1. *PID*
  2. *name*
  3. *birthdate*
  4. *state*
  5. *description*
  6. *image*
  7. *food*
  8. *type*
  9. *breed*
  10. *has\_request –* 0/1 Daca animalul are o cerere activa
  11. *request\_type –* Tipul de cerere activa (**Vezi ANEXA 2**)
  12. *request\_state –* Starea cererii active

Ex: [’animals’][5][’name’]

* **animalrequest.php**

**Serverul primește o adresa de email și o parola și verifica dacă acestea sunt asociate unui utilizator existent. În cazul în care sunt returnează numele și prenumele utilizatorului.**

Parametrii:

1. *security\_code* – 8981ASDGHJ22123 **(Doar pentru Android)**
2. *UID:* Codul utilizatorului **(Doar pentru Android)**
3. *PID* – Codul animalului
4. *request\_type –* Tipul cererii(**Vezi ANEXA 2**)

Raspuns:

1. *status:* **1** (Succes); **0** (Greseala a utilizatorului); **-1** (Eroare server)

**--------------------Ce e sub aceasta linie e in curs de implementare!!-------------------**

* **ping.php**

**Serverul returnează starea curenta a sistemului.**

Parametrii:

1. *security-code -* **8706A97035**

Raspuns:

1. *status --* **-1** (Eroare de server); **1** (All good)
2. *maintenance --* **true** (Serverul este în mentenanță); **false** (Serverul nu este în mentenanță)

* **resetpassword.php**

**Serverul primește o adresa de email și generează un cod (dacă aceasta aparține unui utilizator) pe care îl trimite pe acea adresa de email pentru ca utilizatorul sa își poată reseta parola contului.**

Parametrii:

1. *security-code -* **8706A97035**
2. *user-email --* E-mailul utilizatorului

Raspuns:

1. *status --* **-1** (Eroare de server); **1** (All good)
2. *registered-email --* **true** (Utilizatorul exista in baza de date); **false** (Nu exista)
3. *code-generated--* **true** (Codul a fost generat și emailul trimis către user); **false** (Codul nu a putut fi generat)

* **newpassword.php**

**Serverul primește un cod și resetează parola utilizatorului (dacă acel cod este corect).**

Parametrii:

1. *security-code -* **8706A97035**
2. *reset-code --* Codul de resetare
3. *User-new-password --* Noua parola

Raspuns:

1. *status --* **-1** (Eroare de server); **1** (All good)
2. *correct -code --* **true** (Codul exista); **false** (Codul nu exista)
3. *password-reset--* **true** (Parola a fost restata cu succes; **false** (Parola nu a putut fi resetata)

ANEXE

1. ANEXA 1

|  |  |
| --- | --- |
| Coduri Animale | |
| 0 | Toate Animalele |
| 1 | Pisica |
| 2 | Caine |
| 3 | Rozator |
| 4 | Reptila |
| 5 | Pasare |
| 6 | Acvatic |