MINISTERUL EDUCAȚIEI ȘI CERCETĂRII

UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA

PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student	Cigan Oliviu-David	GRUPA:	30132	Nota	

Brain Tumor Detection System

Autor: Cigan Oliviu-David

Grupa**: 30132**

AN UNIVERSITAR: 2022-2023

PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

Cuprins

1. Scopul Proiectului 3
   a. Obiective 3
   b. Specificații 4
2. Studiu bibliografic 6
3. Analiză, proiectare, implementare 7
4. Concluzii 13
   a. Rezultate obținute 13
   b. Direcții de dezvoltare 29

PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

1. Scopul Proiectului

Brain-Tumor-Detection-System este un proiect inovator in industria medicala care are ca scop imbunatatirea procesului de diagnosticare a tumorilor cerebrale. Prin colectarea si analizarea imaginilor de tip radiografie a creierului, acest sistem ajuta medicii sa determine cu aproximatie prezenta unei tumori.

Acest proiect este extrem de important deoarece diagnosticarea timpurie a tumorilor cerebrale poate face diferenta dintre viata si moarte pentru pacienti. Cu cat o tumora este descoperita mai devreme, cu atat sansele de a trata si vindeca pacientul sunt mai mari.

Brain-Tumor-Detection-System utilizeaza tehnologie de ultima ora pentru a procesa si analiza imaginile de radiografie a creierului, oferind astfel un nivel ridicat de precizie si fiabilitate. Sistemul este proiectat sa identifice anomalii in imaginile radiologice, astfel incat sa poata fi identificate zonele de interes si sa se ofere o evaluare precisa.

In plus, Brain-Tumor-Detection-System poate fi utilizat in timp real, ceea ce inseamna ca medicii pot avea acces instantaneu la rezultatele analizelor, permitand astfel diagnosticarea rapida si precisa. Acest sistem poate contribui semnificativ la reducerea timpului necesar pentru diagnosticarea tumorilor cerebrale, ceea ce poate avea un impact major asupra sanatatii si bunastarii pacientilor..

a. Obiective

Obiectul acestui proiect, este sa ofere medicilor o unealta utila pentru a intelege si analiza mai usor radiografiile la creier, contribuind astfel la imbunatatirea procesului de diagnosticare a tumorilor cerebrale.

Este important de mentionat ca acest proiect nu are ca scop inlocuirea medicilor sau analizatorilor de radiografii, ci sa fie un suport pentru acestia. Sistemul este proiectat sa ofere informatii suplimentare si evaluari preliminare ale imaginilor radiologice, astfel incat medicii sa poata lua decizii mai informate si sa ofere un diagnostic mai precis.

Brain-Tumor-Detection-System foloseste tehnologie avansata pentru a procesa imaginile radiologice ale creierului, identificand anomalii si zone de interes. In plus, sistemul poate fi utilizat in timp real, astfel incat medicii sa poata accesa informatiile

necesare imediat, ajutand astfel la reducerea timpului necesar pentru diagnosticarea tumorilor cerebrale.

Este important sa subliniem faptul ca acest proiect este conceput pentru a fi o unealta utila pentru medici, oferindu-le un suport suplimentar si ajutandu-i sa ia decizii mai informate in procesul de diagnosticare a tumorilor cerebrale. Nu are ca scop inlocuirea lor sau a analizatorilor de radiografii, ci sa fie un instrument de sprijin pentru a imbunatati procesul de diagnosticare si tratament pentru pacienti.

b. Specificații

In specificațiile lucrării detaliați cerințele. Descrieți ce intentionați să obtineți. Vă puteți referi la funcțiile aplicației, interfață, nivele de performanță, elemente, calitate, limitări,

In cadrul acestui proiect, s-a dezvoltat un sistem automat pentru colectarea imaginilor si diagnosticarea tumorilor cerebrale. In prima etapa a procesului, se face colectarea automata a imaginilor de radiografii ale creierului, care sunt redimensionate la o dimensiune standard de 300px x 300px pentru a putea fi prelucrate mai eficient.

In etapa urmatoare, imaginile sunt prelucrate intr-un format numeric, prin intermediul unui proces de preprocesare. Acest proces are ca scop eliminarea zgomotului si a altor artefacte din imagine, pentru a permite o analiza mai precisa a datelor.

Odata prelucrate, imaginile sunt trecute prin trei algoritmi diferiti de machine learning: LogisticRegression, SVM si RandomForestClassifier. Fiecare dintre aceste algoritmi este evaluat si comparat pentru a determina care dintre ele ofera cea mai buna acuratete in diagnosticarea tumorilor cerebrale.

Dupa ce algoritmul de machine learning a fost selectat, sistemul poate analiza trei tipuri diferite de tumori cerebrale: pituitary_tumor, meningioma_tumor si glioma_tumor. In functie de datele colectate si analizate, sistemul poate oferi o concluzie cu privire la prezenta sau absenta cancerului. De asemenea, performanta sistemului este destul de inalta, chiar si luand in considerare dataset-ul destul de complex utilizat in acest proiect. Sistemul raspunde foarte rapid si eficient la cerintele utilizatorilor, oferind o acuratete buna in diagosticarea tumorilor cerebrale.

In final, sistemul poate afisa o vizualizare a imaginii, impreuna cu concluzia oferita de sistem. Aceasta poate fi deosebit de utila pentru medicii care se ocupa de diagnosticarea tumorilor cerebrale, oferindu-le o metoda precisa si rapida pentru a determina daca pacientii lor sunt sau nu afectati de aceasta afectiune grava.

Acest sistem de detectare a tumorilor cerebrale a fost creat folosind Python si Jupyter Notebook. Pentru a dezvolta acest sistem, s-au utilizat o serie de librarii importante, inclusiv Sklearn, numpy, pandas si matplotlib, care au fost esentiale in procesul de colectare si preprocesare a datelor.

PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

2. Studiu bibliografic

Conține o analiză a ceea ce s-a realizat/studiat anterior. Arătați că ați studiat materiale bibliografice și că ați înțeles ceea ce ați citit.

Puteti include diferite puncte de vedere asupra problemei pe care o rezolvați în lucrare.

Nu uitați să citați corespunzător autorii oricărei idei extrase dintr-o sursă bibliografică.

PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

3. Analiză, proiectare, implementare

Logistic Regression

Logistic Regression este un model matematic folosit in statistica si invatarea automata (machine learning) pentru a prezice valoarea unei variabile binare in functie de mai multe variabile de intrare (predictori). Mai exact, logistic regression se refera la regresia logistica binara, unde variabila tinta poate avea doar doua valori posibile, de exemplu "1" sau "0", "adevarat" sau "fals".

In mod obisnuit, logistic regression este utilizat pentru a face predictii in cazurile in care variabila dependenta este de tip binar, cum ar fi prezicerea daca un pacient va dezvolta sau nu o anumita boala, daca un client va cumpara sau nu un anumit produs sau daca un email este spam sau nu. Modelul calculeaza o probabilitate a evenimentului binar in functie de variabilele de intrare si apoi transforma aceasta probabilitate intr-o valoare de 0 sau 1.

In mod concret, modelul logistic regression calculeaza o functie sigmoidala a sumei ponderate a valorilor predictori, iar aceasta functie sigmoidala reprezinta probabilitatea de a avea evenimentul binar. In cazul unei probleme de clasificare binara, se alege o valoare prag, cum ar fi 0.5, pentru a separa cele doua clase.

Functia de activare Sigmoida: (Sigmoid activation)

Math

$$S(z) = \frac{1}{1 + e^{-z}}$$

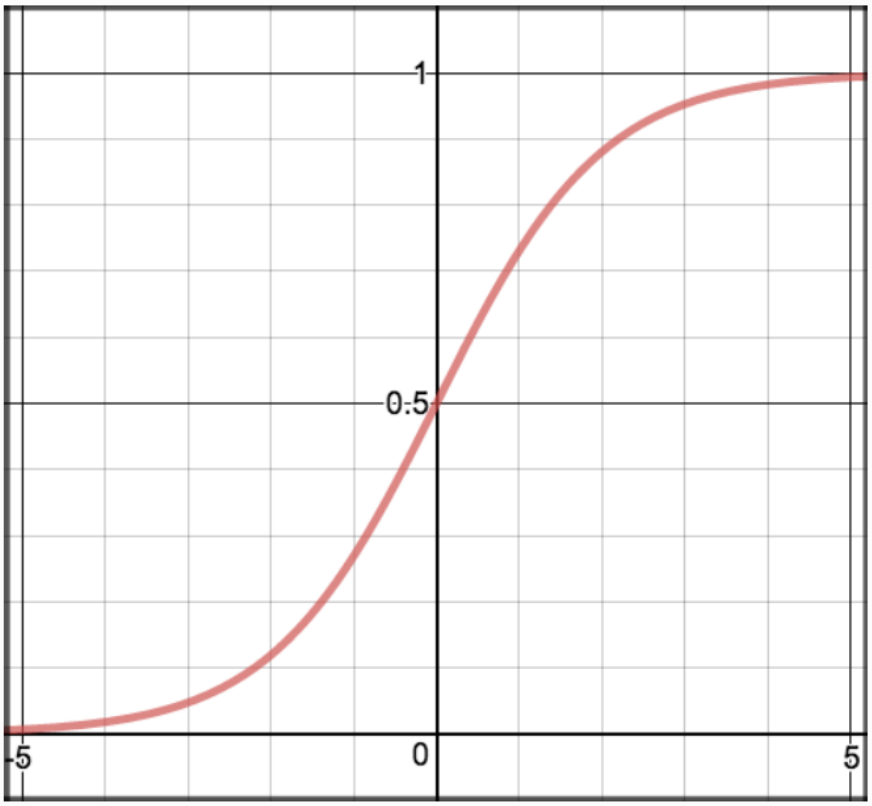
Note

- $s(z)$ = output between 0 and 1 (probability estimate)
- z = input to the function (your algorithm's prediction e.g. mx + b)
- e = base of natural log

PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

Grafic si posibila implementare in Python

Graph



Code

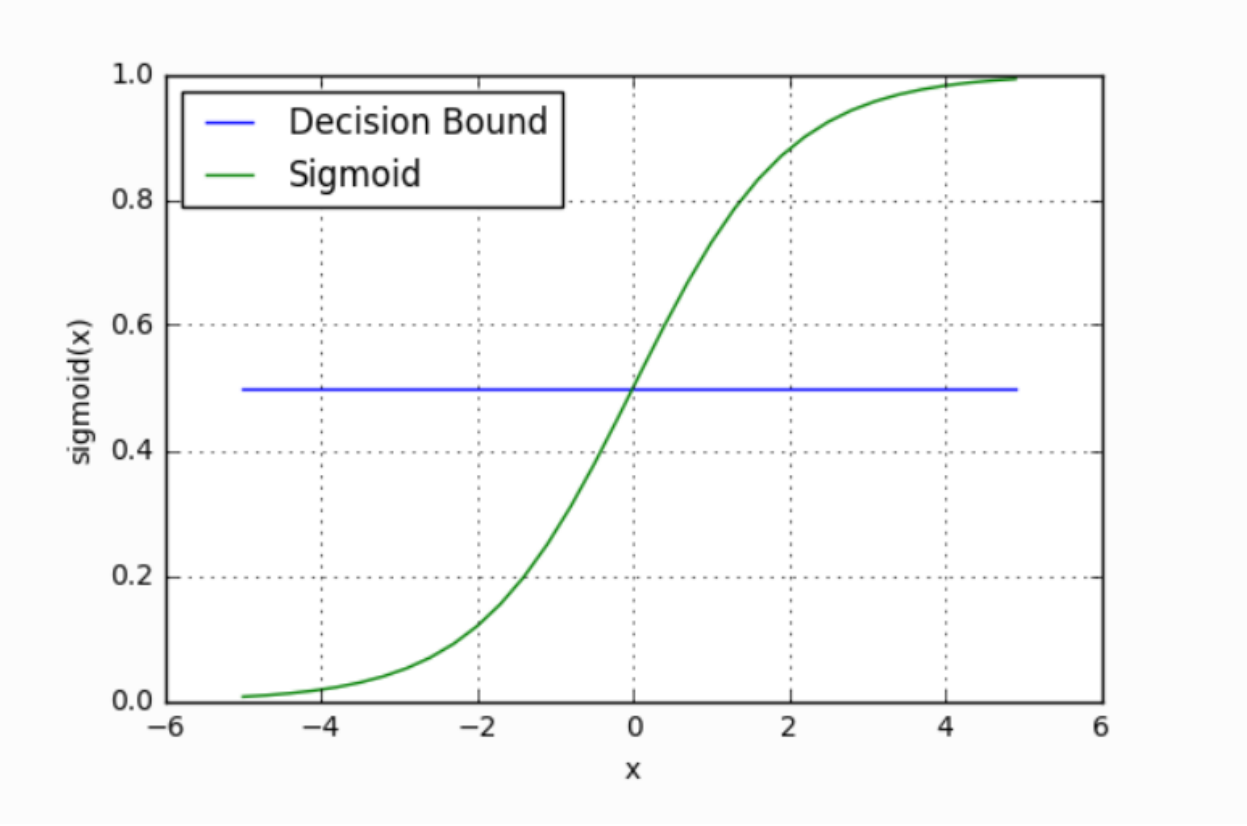
```
def sigmoid(z):
  return 1.0 / (1 + np.exp(-z))
```

PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

Decision Boundary

$$p \geq 0.5, class = 1$$
$$p < 0.5, class = 0$$

Un posibil Graf



PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

Cost function - Functia de cost este o functie matematica folosita in algoritmi de machine learning pentru a evalua cat de bine se potrivesc modelele la datele de antrenament. Scopul principal al functiei de cost este sa determine parametrii modelului astfel incat sa minimizeze eroarea dintre predictiile modelului si valorile reale.

Matematic:

$$MSE = \frac{1}{2N}\sum_{i=1}^{n}(y_i - (W_1x_1 + W_2x_2 + W_3x_3))^2$$

In esenta, functia de cost calculeaza distanta intre predictiile modelului si valorile reale din setul de date de antrenament. In functie de algoritmul de machine learning folosit, functia de cost poate fi diferita.

Gradient descent

Gradient descent este o metoda de optimizare a unei functii de cost prin iteratii succesive pentru a ajunge la valoarea minima a functiei respective. Ideea din spatele gradient descent este de a urmari gradientul functiei de cost pentru a gasi directia cea mai rapida de coborare (in cazul minimizarii functiei de cost) si apoi de a face un pas in directia respectiva. Acest proces se repeta pana cand functia de cost ajunge la o valoare minima sau cand un anumit criteriu de oprire este atins.

$$f'(W_1) = -x_1(y - (W_1x_1 + W_2x_2 + W_3x_3))$$
$$f'(W_2) = -x_2(y - (W_1x_1 + W_2x_2 + W_3x_3))$$
$$f'(W_3) = -x_3(y - (W_1x_1 + W_2x_2 + W_3x_3))$$

PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

Modul de implementare in proiect si parametrii de functionare:

```
[8]: from sklearn.linear_model import LogisticRegression

[9]: classifier = LogisticRegression(C=0.1)
     classifier.fit(xTrain, yTrain)
```

```
C:\Python310\lib\site-packages\sklearn\linear_model\_lo
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale t
    https://scikit-learn.org/stable/modules/preprocessi
Please also refer to the documentation for alternative
    https://scikit-learn.org/stable/modules/linear_mode
  n_iter_i = _check_optimize_result(
```

```
[9]: LogisticRegression
     LogisticRegression(C=0.1)
```

```
[10]: classifier.score(xTest, yTest)

[10]: 0.8481012658227848
```

Parametrul C reprezintă un parametru de regularizare. Regularizarea este o tehnică utilizată pentru a controla overfitting-ul în modelele de machine learning.

Parametrul C inversează forța de regularizare, ceea ce înseamnă că o valoare mică a lui C va duce la o regularizare puternică, în timp ce o valoare mare a lui C va duce la o regularizare mai slabă. Prin urmare, o valoare mai mică a lui C poate duce la un model mai simplu, care poate avea o performanță mai bună pe datele de testare, în timp ce o valoare mai mare a lui C poate duce la un model mai complex, care poate avea o performanță mai bună pe datele de antrenament.

PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

Cativa parametri default folositi de LogisticRegression si care merita mentionati:

max_iter default ~= **100**

multi_class default ~= **auto**

solver default ~= **Limited-memory BFGS**

Lista cu parametrii care ii poate lua LogisticRegression:

- penalty (default='l2'): specifică tipul de penalizare folosit pentru a controla overfitting-ul. Poate fi 'l1', 'l2', 'elasticnet', sau None.
- dual (default=False): specifică dacă să folosească formularea primală sau duală a problemelor de optimizare.
- tol (default=1e-4): specifică toleranța pentru criteriul de oprire.
- C (default=1.0): specifică puterea inversă a regularizării. A fost discutat mai sus.
- fit_intercept (default=True): specifică dacă să se potrivească cu un intercept (bias) pentru regresie.
- intercept_scaling (default=1): specifică scara interceptului, dacă fit_intercept este setat la True.
- class_weight (default=None): specifică greutatea asociată fiecărei clase. Poate fi None sau 'balanced'.
- random_state (default=None): specifică seed-ul generatorului de numere aleatoare.
- solver (default='lbfgs'): specifică algoritmul folosit pentru optimizarea problemei. Poate fi 'newton-cg', 'lbfgs', 'liblinear', 'sag', sau 'saga'.
- max_iter (default=100): specifică numărul maxim de iterații pentru solverul ales.
- multi_class (default='auto'): specifică strategia de gestionare a problemelor multi-clasa. Poate fi 'ovr' (one-vs-rest) sau 'multinomial'.
- verbose (default=0): specifică nivelul de verbositate al solverului.
- warm_start (default=False): specifică dacă să se folosească soluția anterioară ca punct de plecare.
- n_jobs (default=None): specifică numărul de nuclee procesor disponibile pentru a rula în paralel. Poate fi None sau un intreg.

4. Concluzii

a. Rezultate obţinute

In cadrul proiectului de detectare a tumorilor cerebrale, s-au utilizat trei algoritmi de machine learning: LogisticRegression, SVM si RandomForestClassifier. Pentru a evalua performanta acestor algoritmi, s-a folosit o imagine standard de marime 300x300.

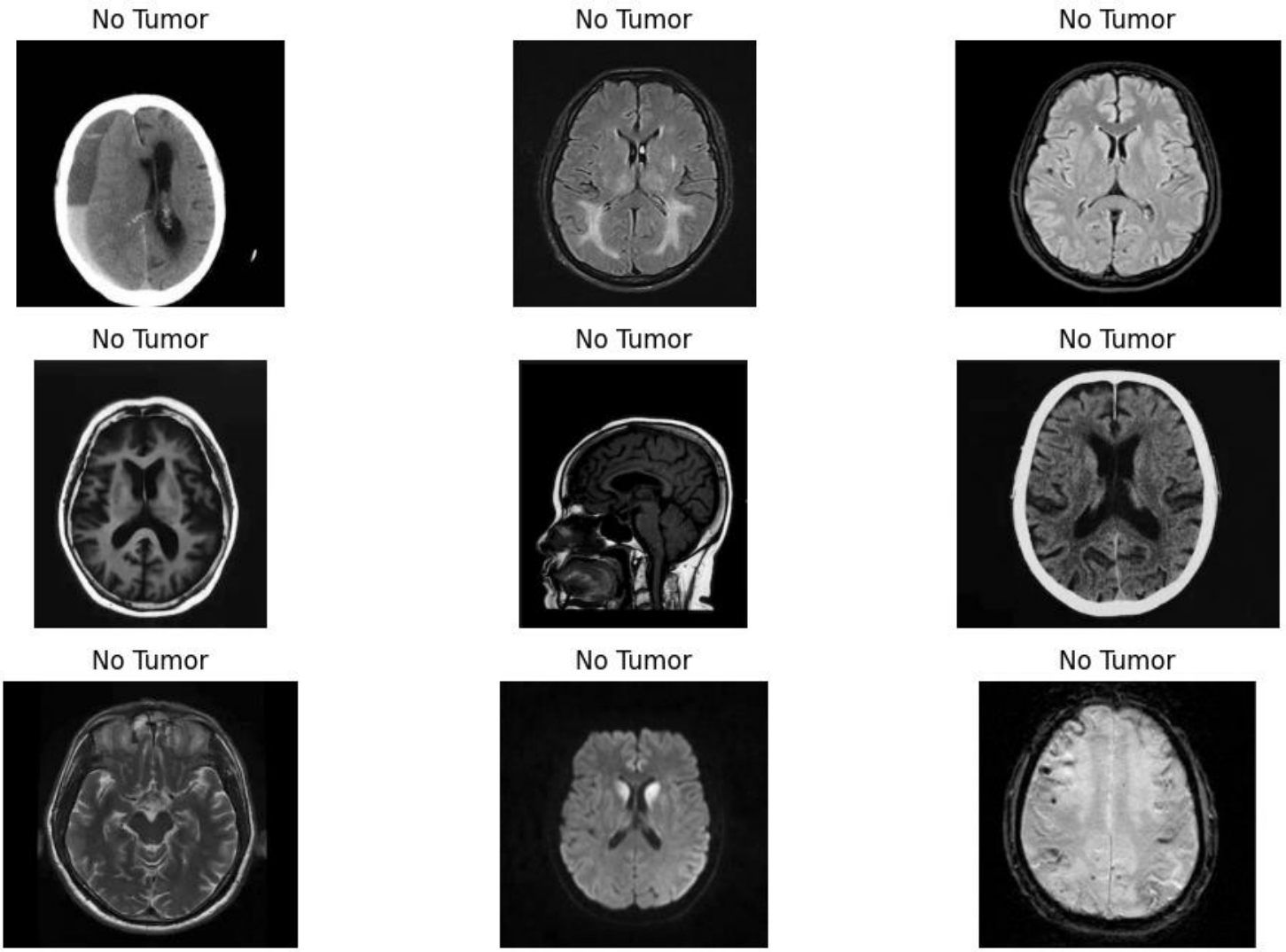
Dupa analiza datelor, s-a observat ca LogisticRegression a avut o performanta de 0.8481012658227848, ceea ce reprezinta o acuratete de 85%. RandomForestClassifier a avut o performanta de 0.810126582278481, ceea ce reprezinta o acuratete de 81%. In schimb, SVM a obtinut o performanta impresionanta de 0.9367088607594937, adica aproape 94% acuratete.

Aceste rezultate arata ca SVM a fost superior in analiza imaginilor si in detectarea tumorilor cerebrale, o performanta remarcabila poate fi atribuita capacitatii acestui algoritm de a gestiona date complexe si de a detecta modele mai precise. In consecinta, SVM este considerat a fi algoritmul ideal pentru a fi utilizat in cadrul sistemului de detectare a tumorilor cerebrale, deoarece ofera o acuratete buna si o performanta superioara in comparatie cu ceilalti doi algoritmi testati.

PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

No Tumor - Logistic Regression:

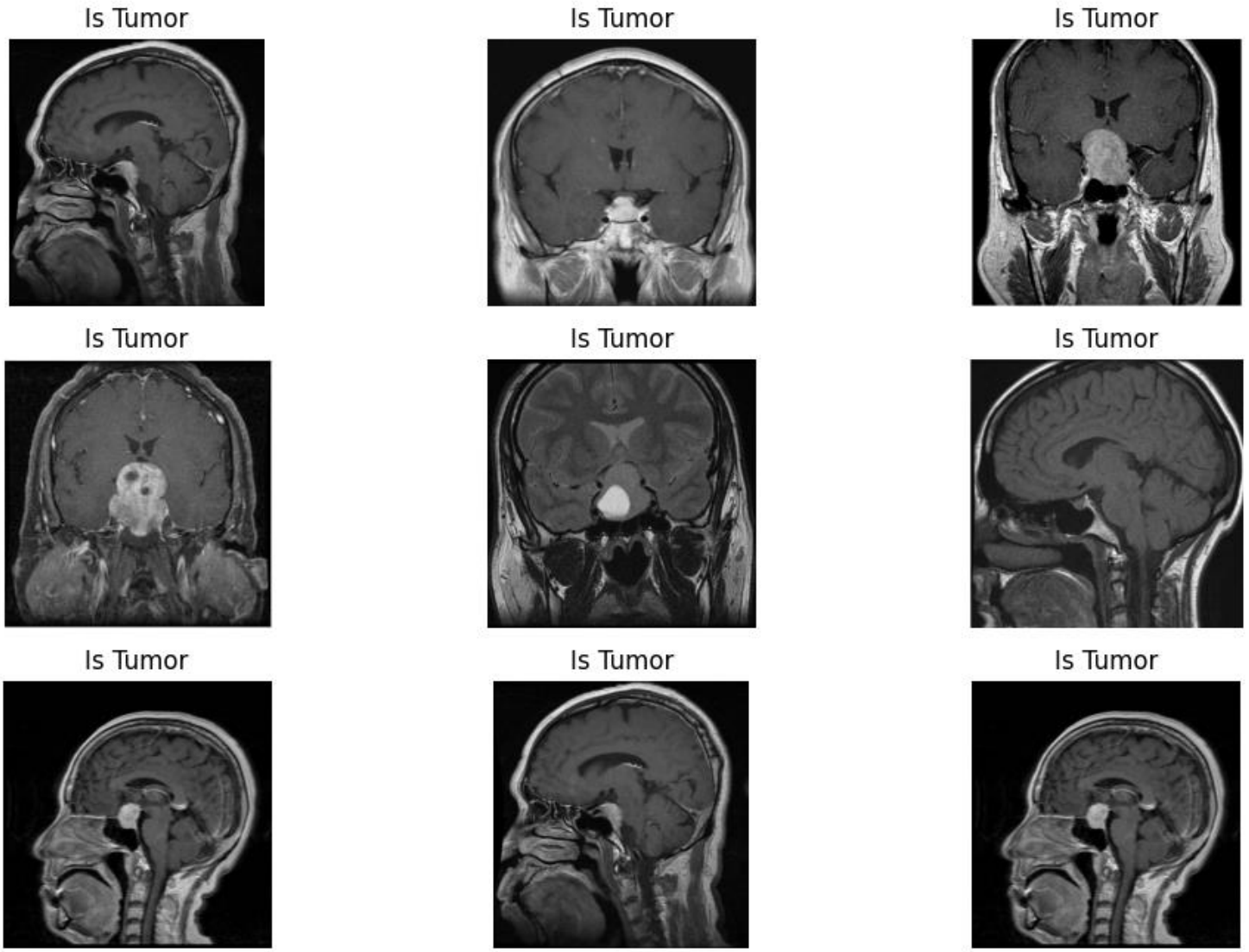
```
[11]: test_based_on_images(classifier, 'no_tumor/')
```



PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

Pituitary Tumor Detection - Logistic Regression:

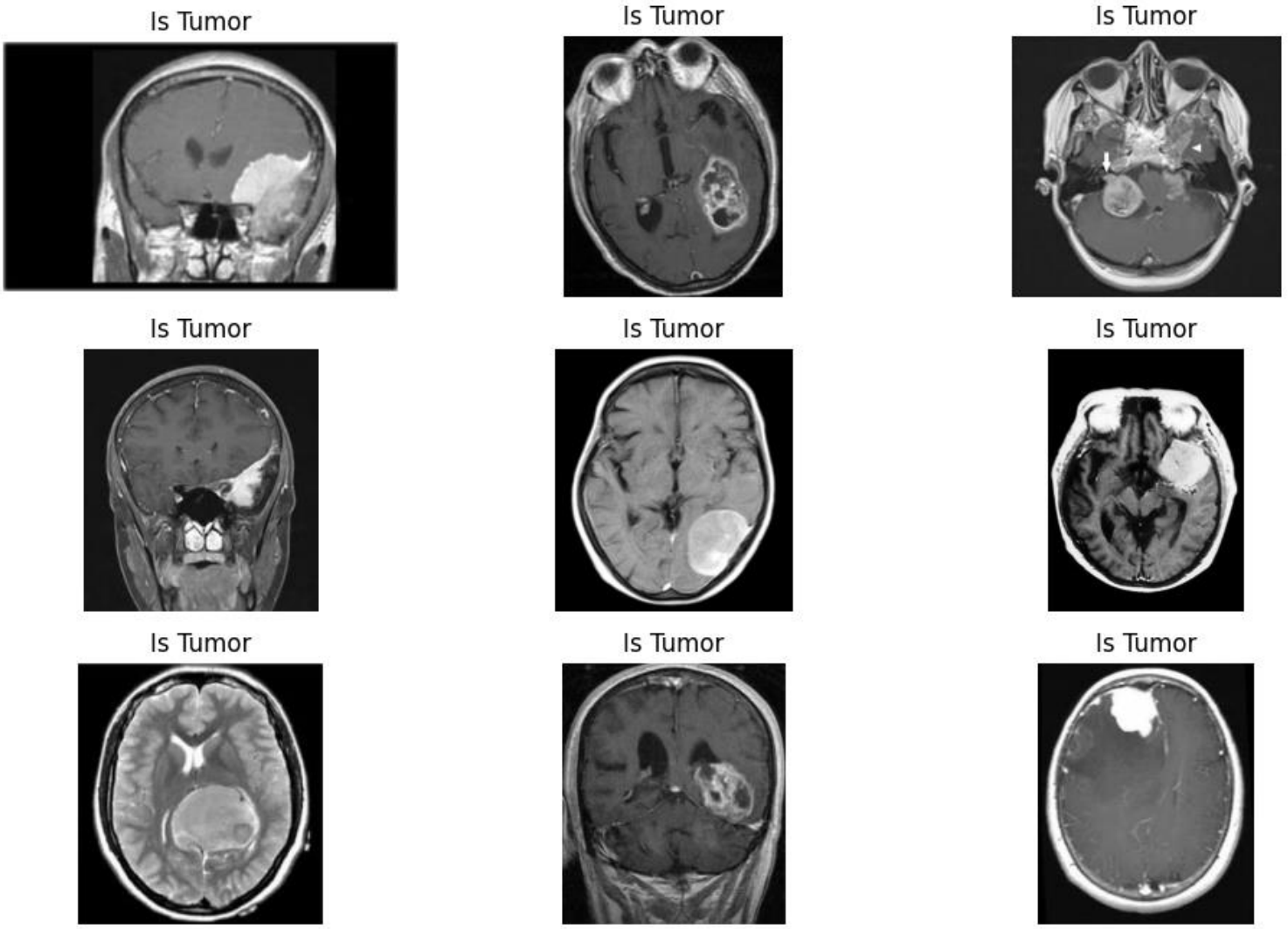
```
[12]: test_based_on_images(classifier, 'pituitary_tumor/')
```



PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

MeningiomaTumor Detection – Logistic Regression

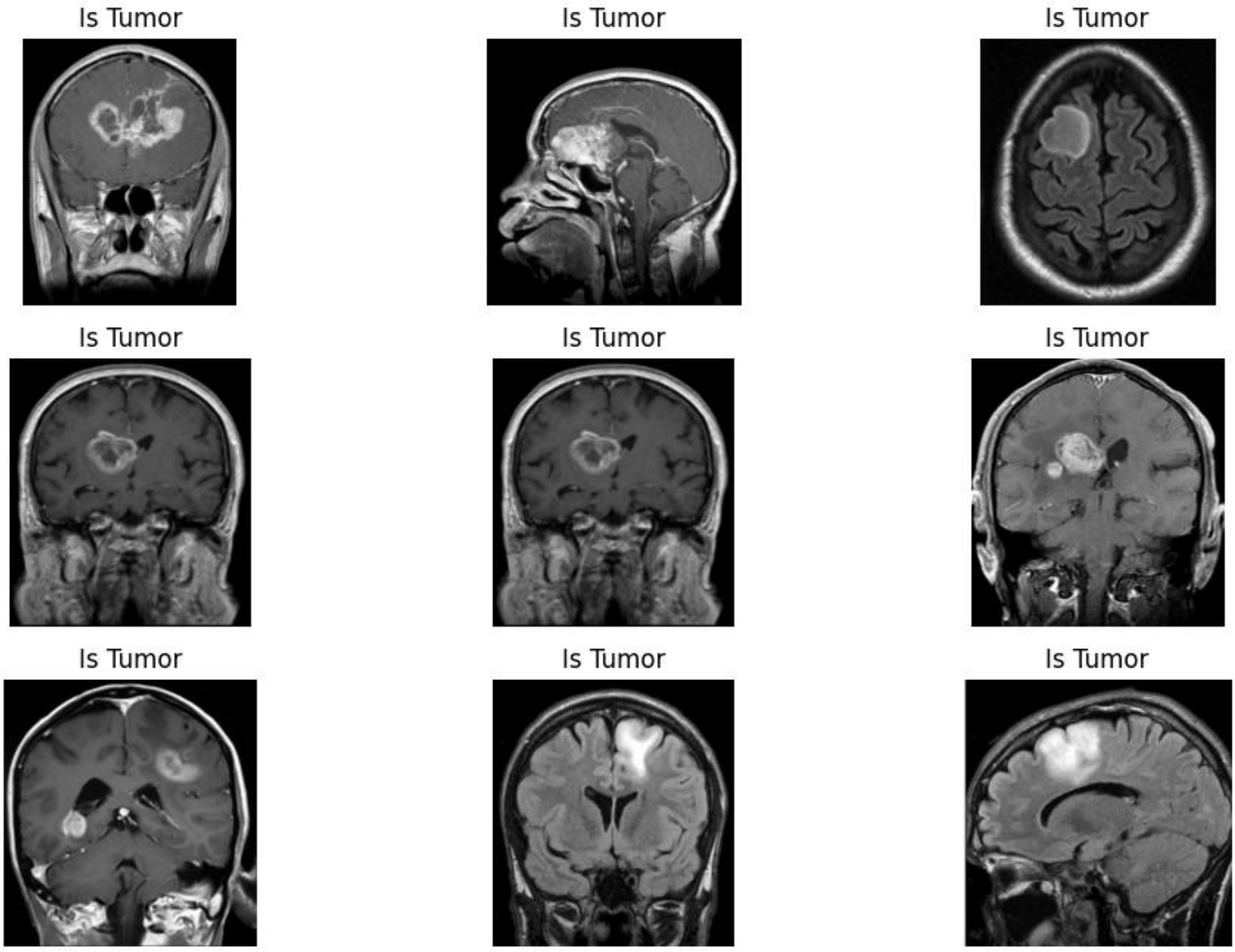
```
[13]: test_based_on_images(classifier, 'meningioma_tumor/')
```



PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

Glioma Tumor Detection - Logistic Regression

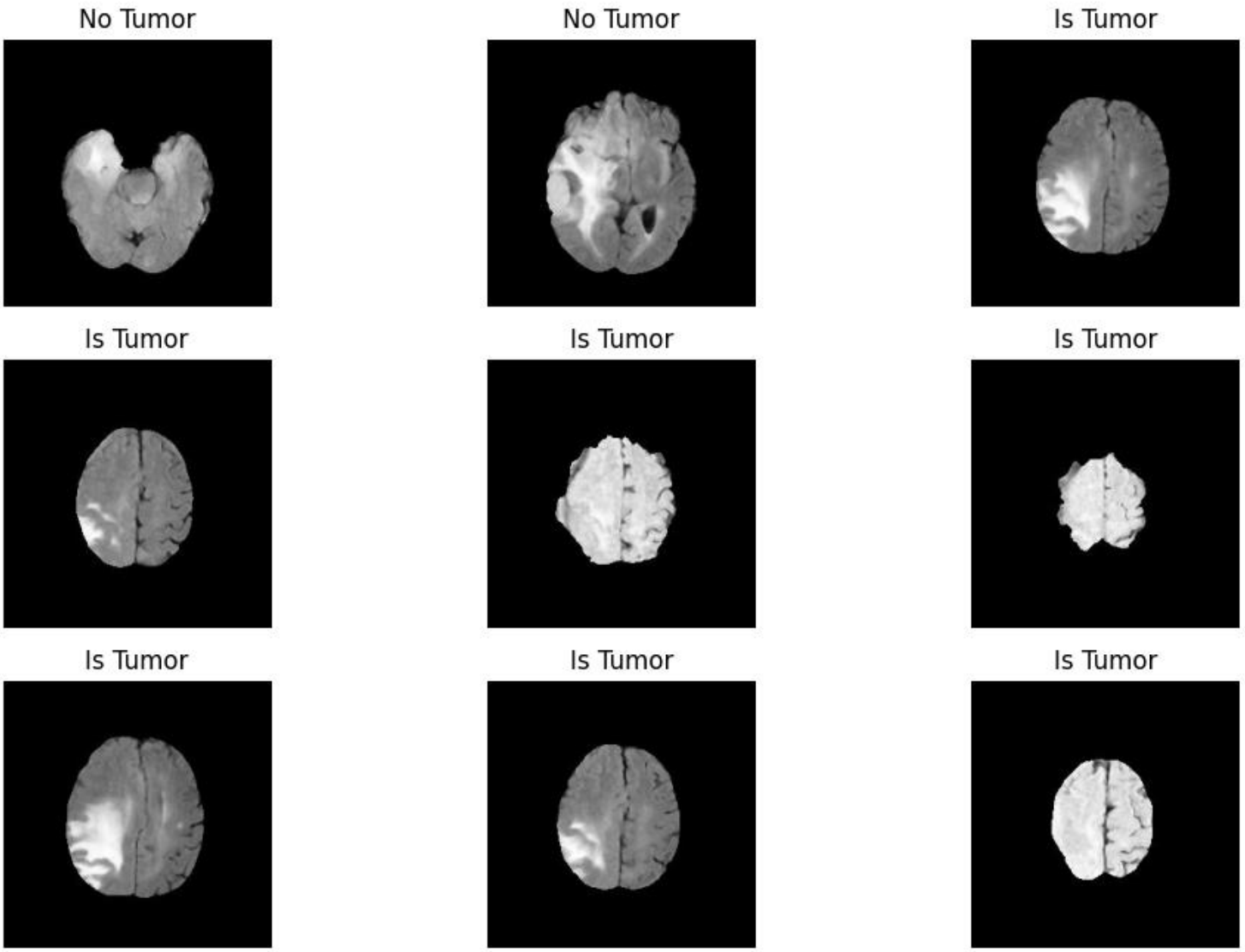
```
[14]: test_based_on_images(classifier, 'glioma_tumor/')
```



PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

New Tumor Dataset Detection - Logistic Regression

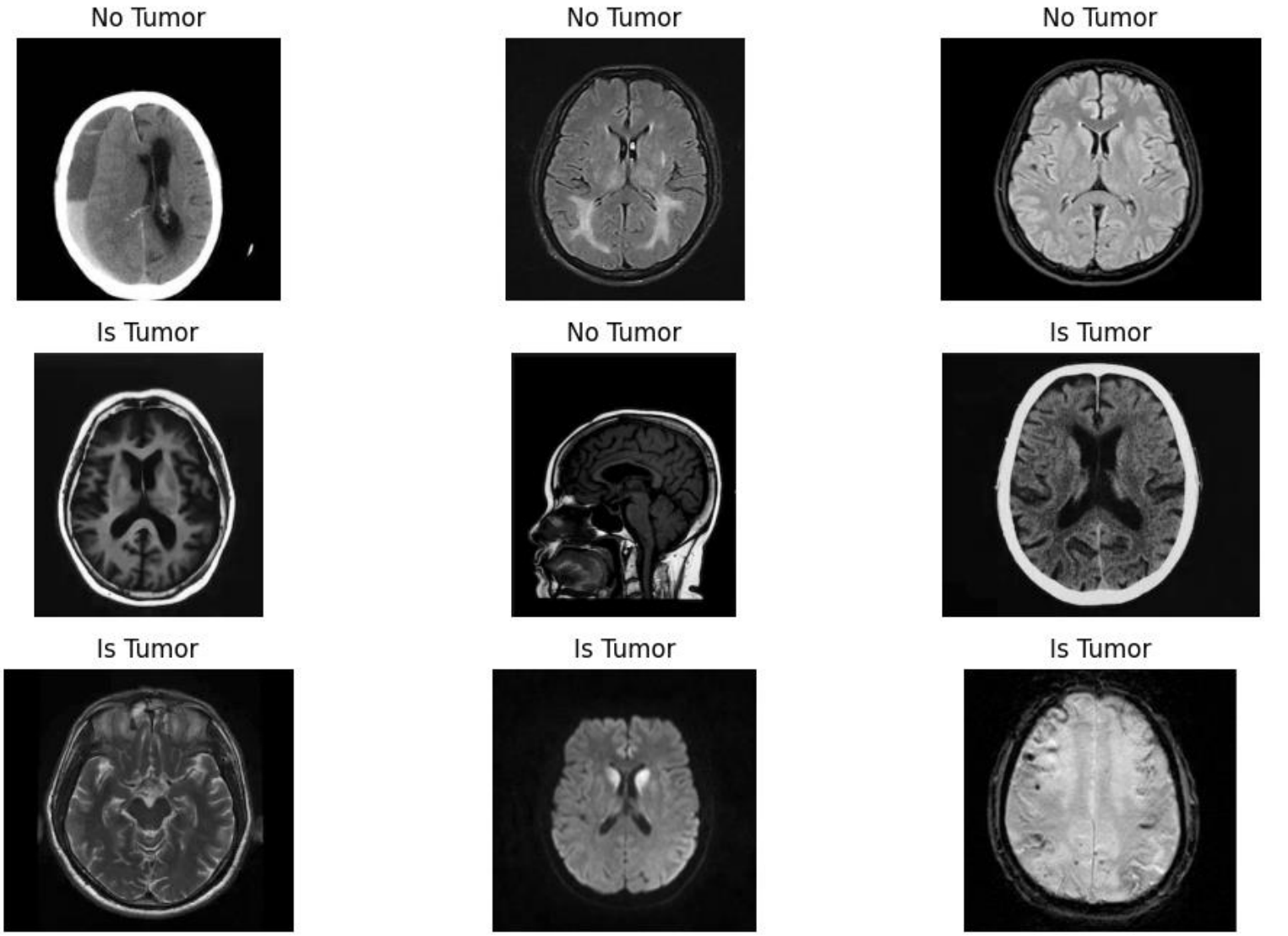
```
[15]: test_based_on_images(classifier, 'brain_tumor/', 400)
```



PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

No Tumor - Random Forest Classifier:

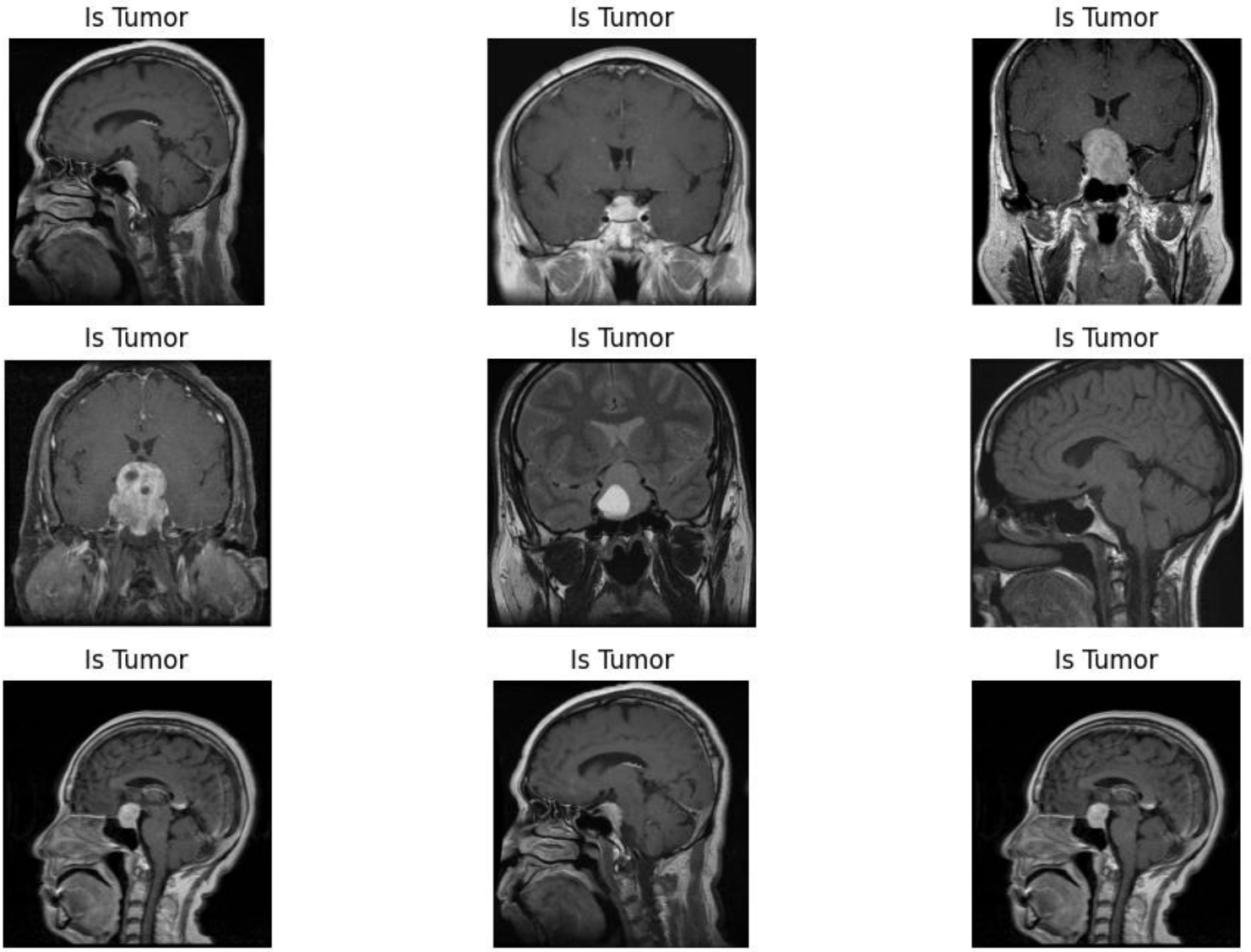
```
[11]: test_based_on_images(classifier, 'no_tumor/')
```



PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

Pituitary Tumor Detection - Random Forest Classifier:

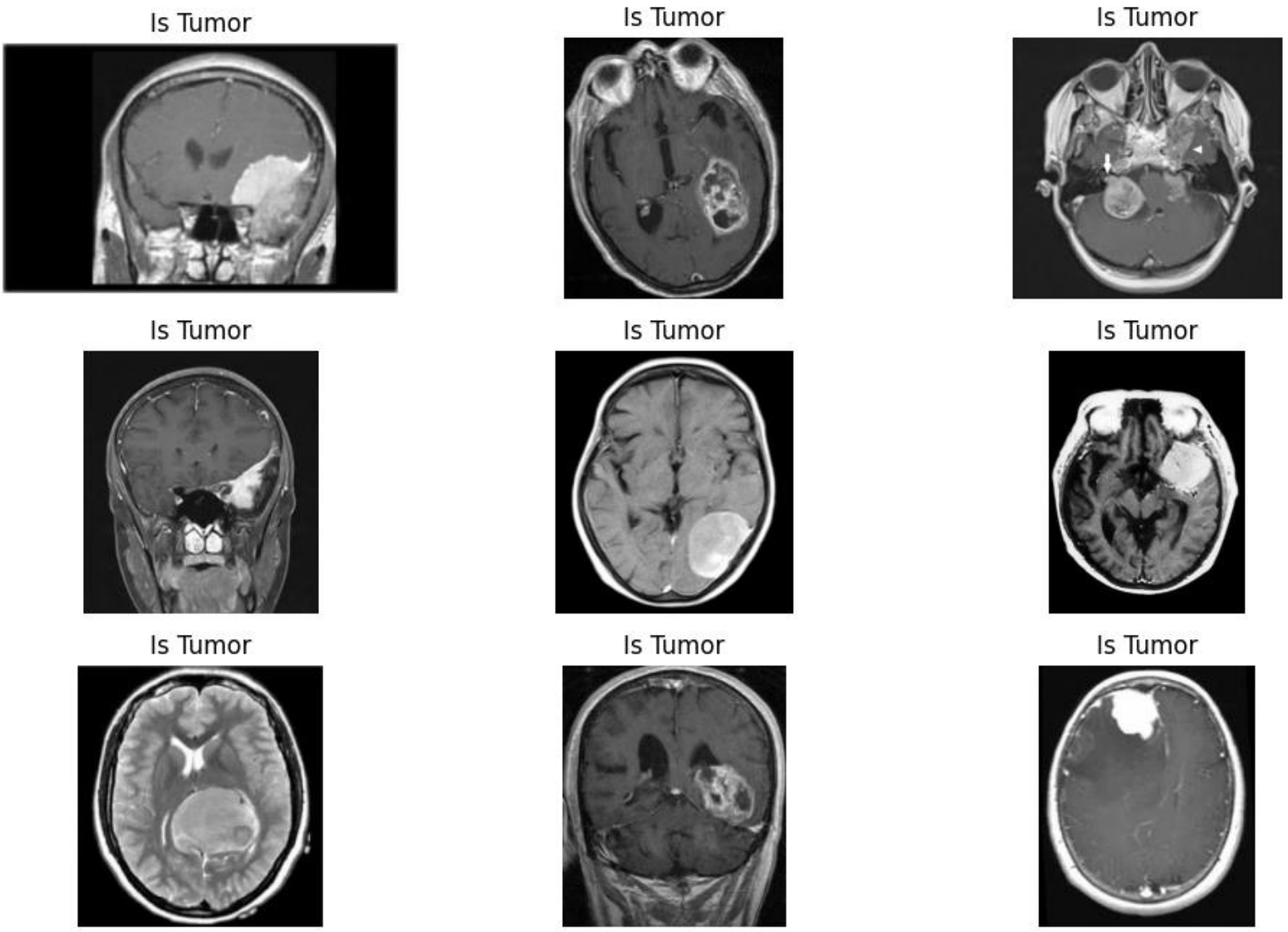
```
[12]: test_based_on_images(classifier, 'pituitary_tumor/')
```



PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

Meningioma Tumor Detection – Random Forest Classifier

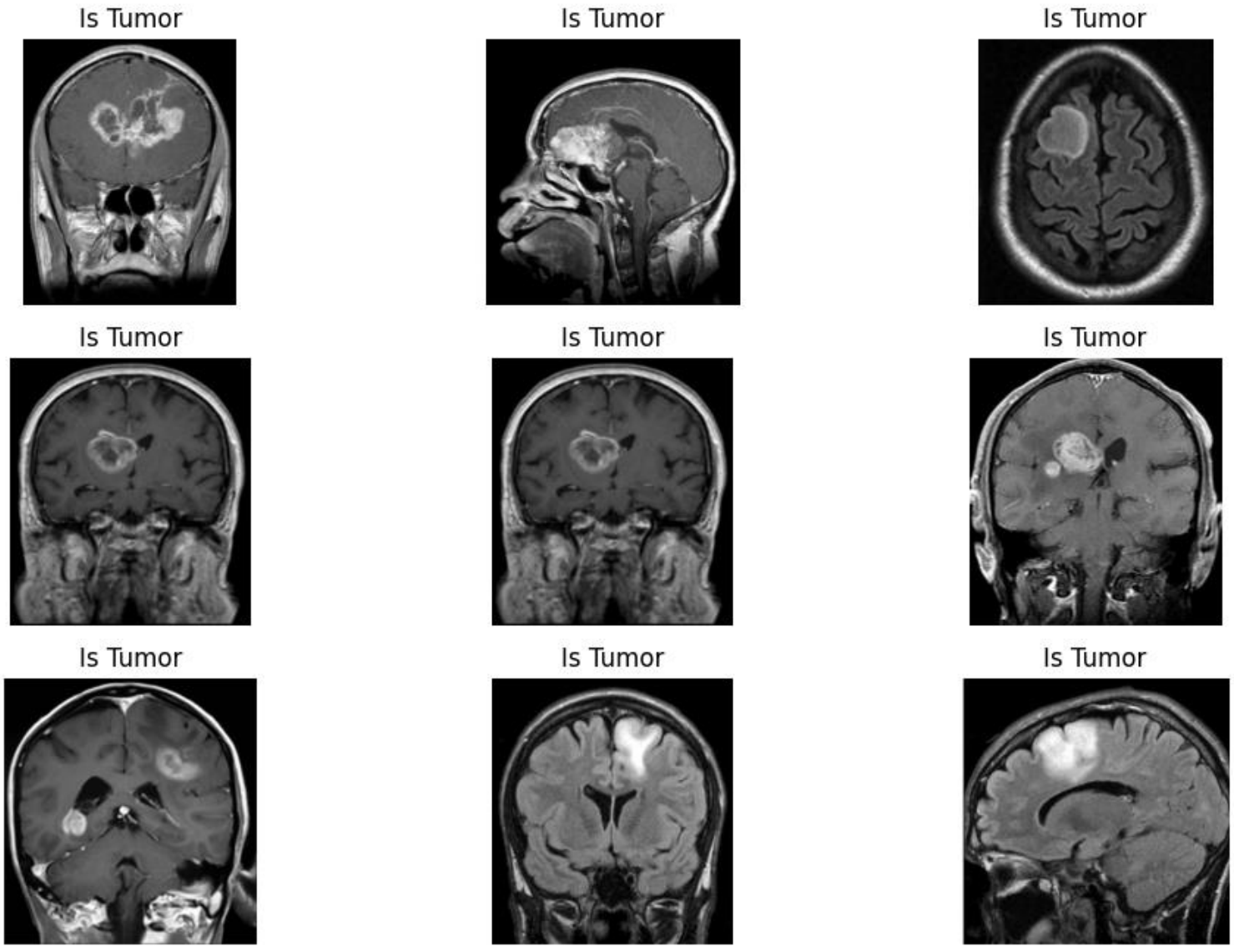
```
[13]: test_based_on_images(classifier, 'meningioma_tumor/')
```



PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

Glioma Tumor Detection - Random Forest Classifier

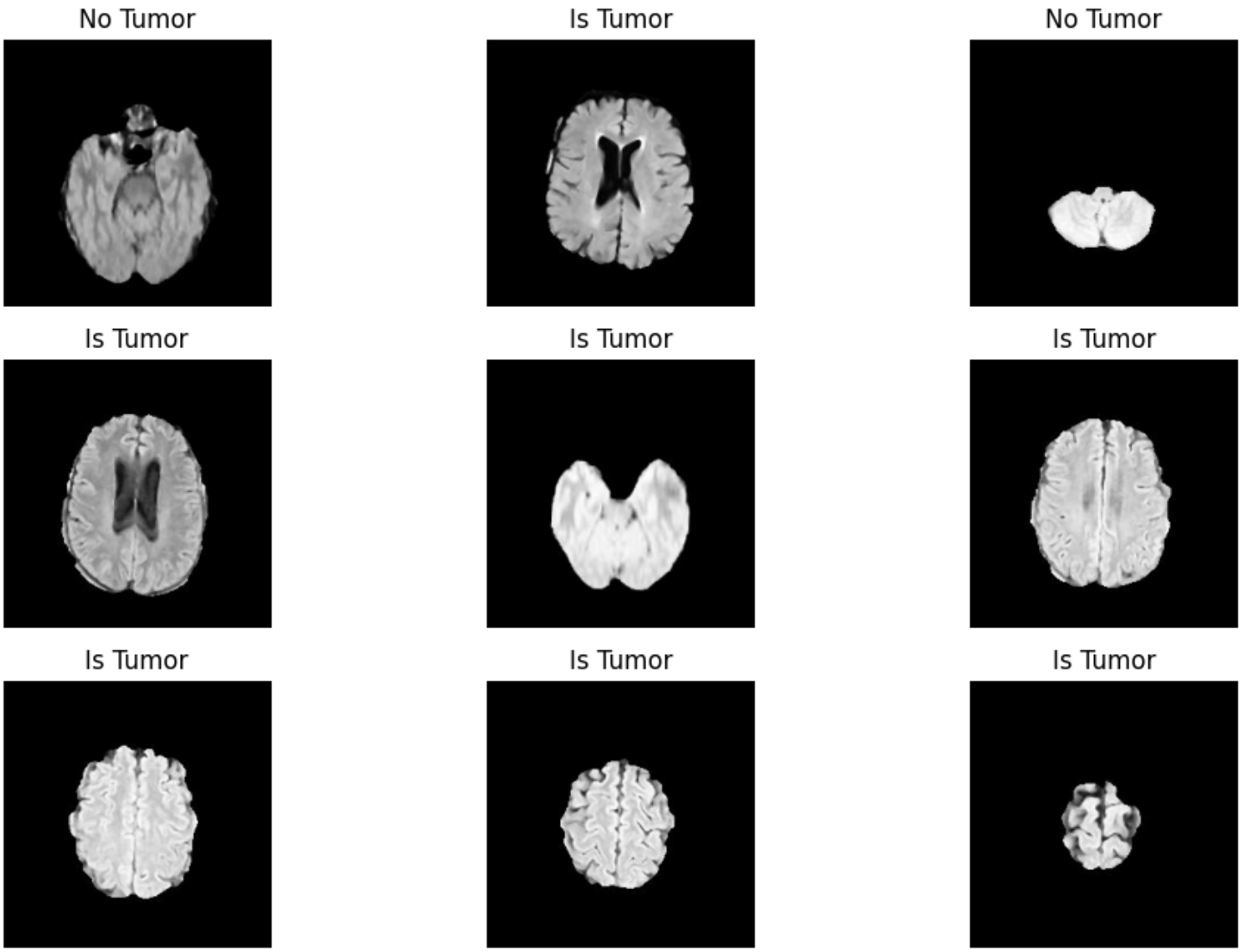
```
[14]: test_based_on_images(classifier, 'glioma_tumor/')
```



PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

New Tumor Dataset Detection – Random Forest Classifier

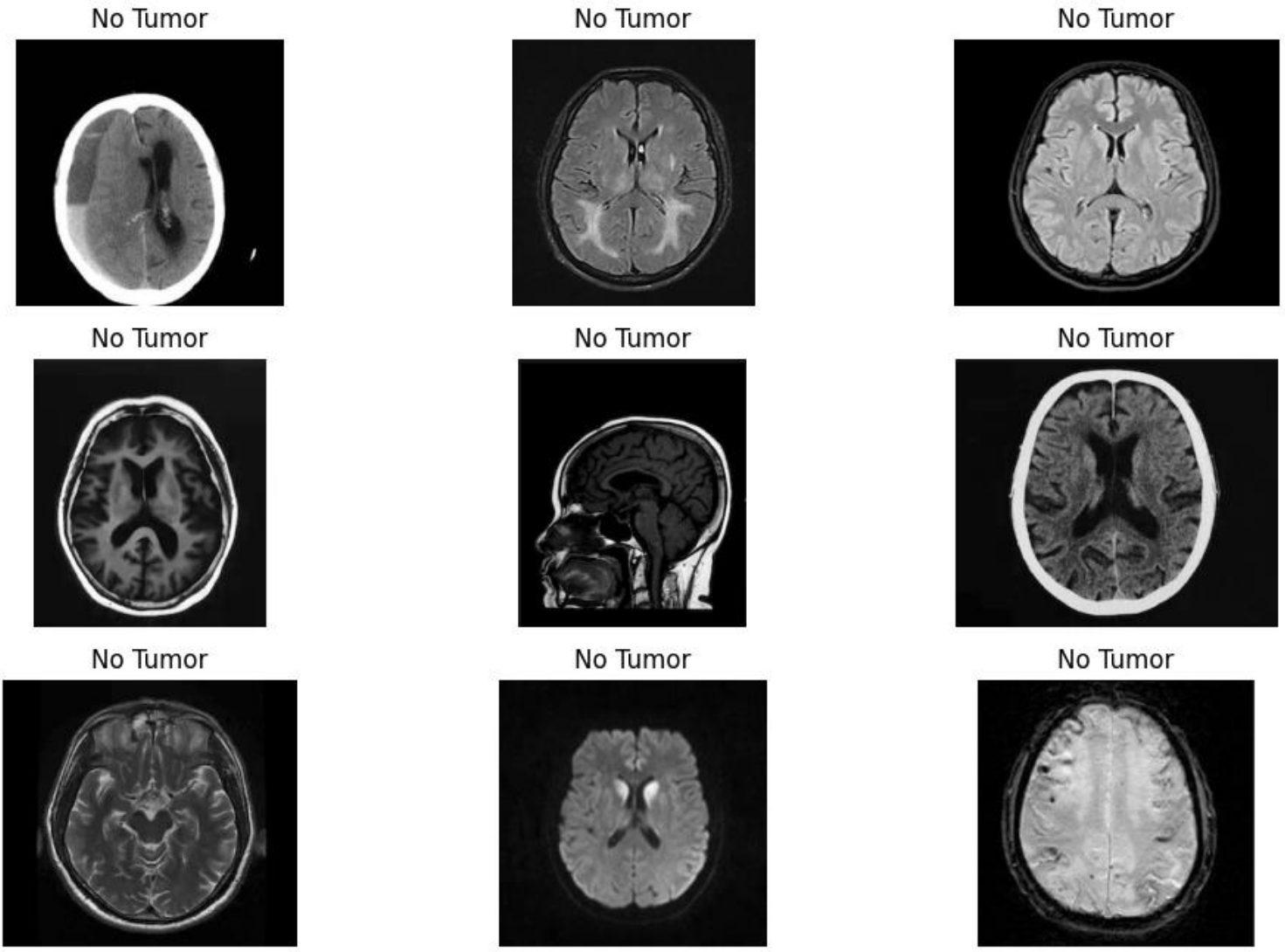
```
[15]: test_based_on_images(classifier, 'brain_tumor/', 20)
```



PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

No Tumor – Support Vector Machine – Support Vector Classifier:

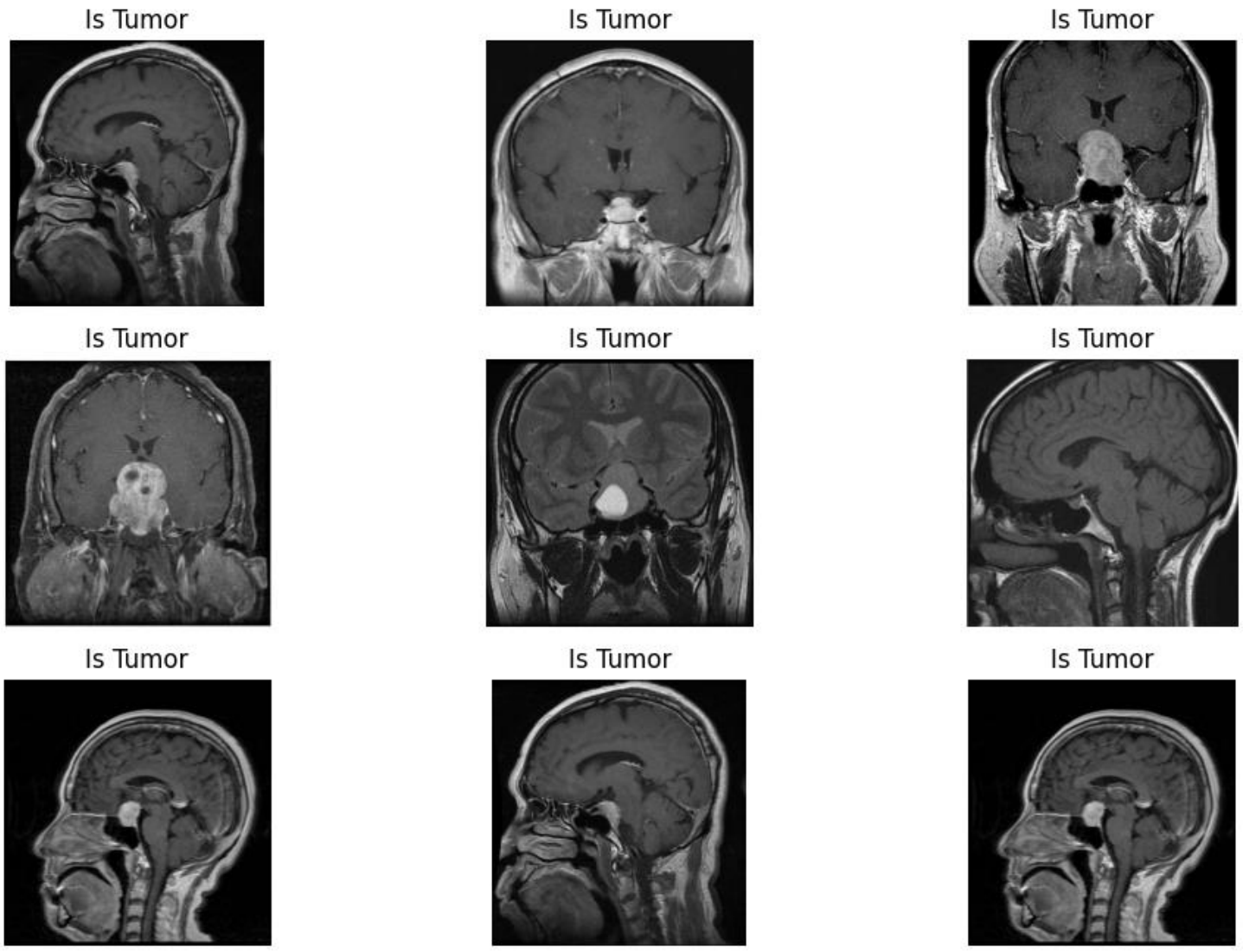
```
[11]: test_based_on_images(classifier, 'no_tumor/')
```



PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

Pituitary Tumor Detection – Support Vector Machine – Support Vector Classifier:

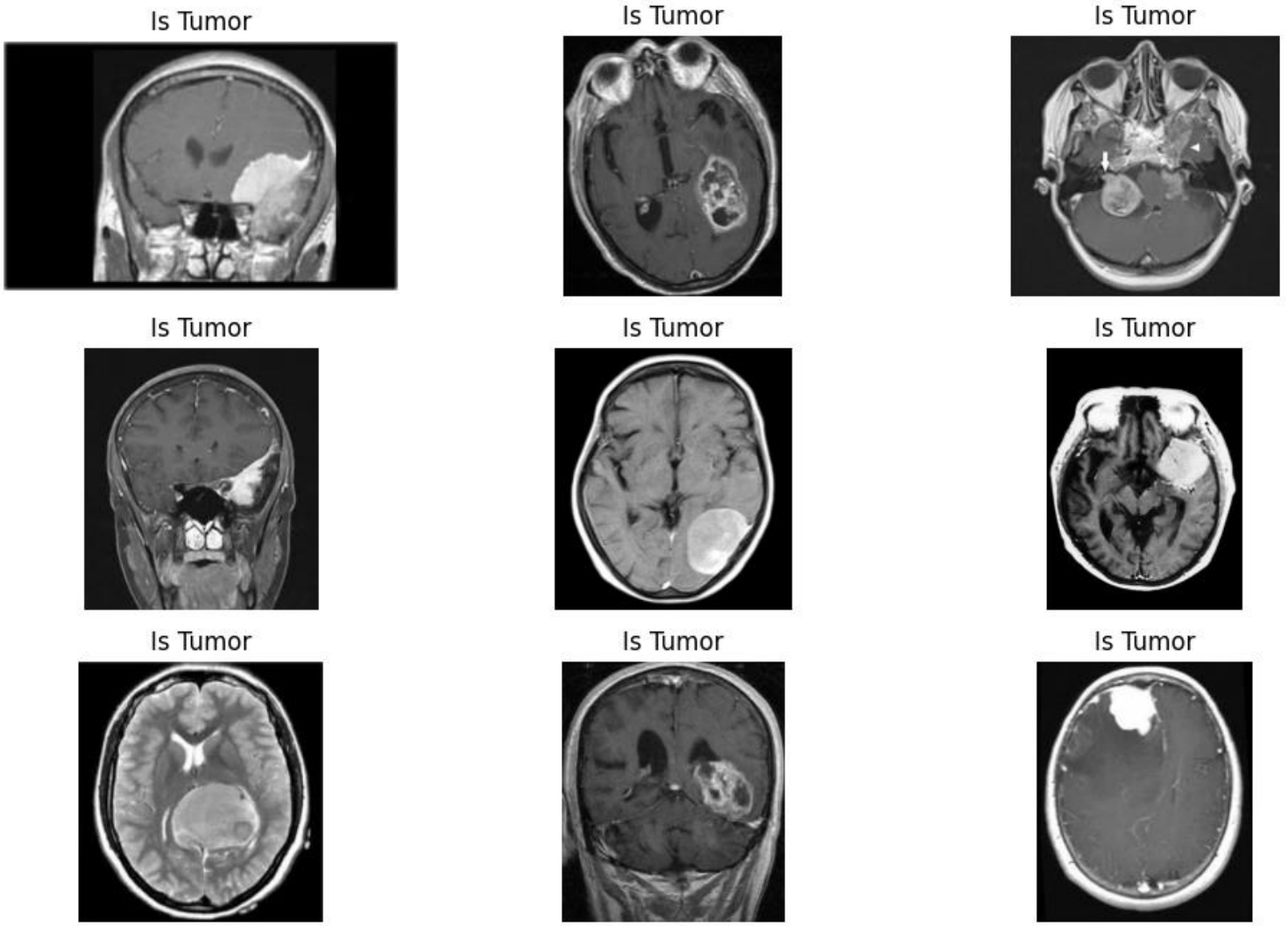
```
[12]: test_based_on_images(classifier, 'pituitary_tumor/')
```



PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

Meningioma Tumor Detection – Support Vector Machine – Support Vector Classifier:

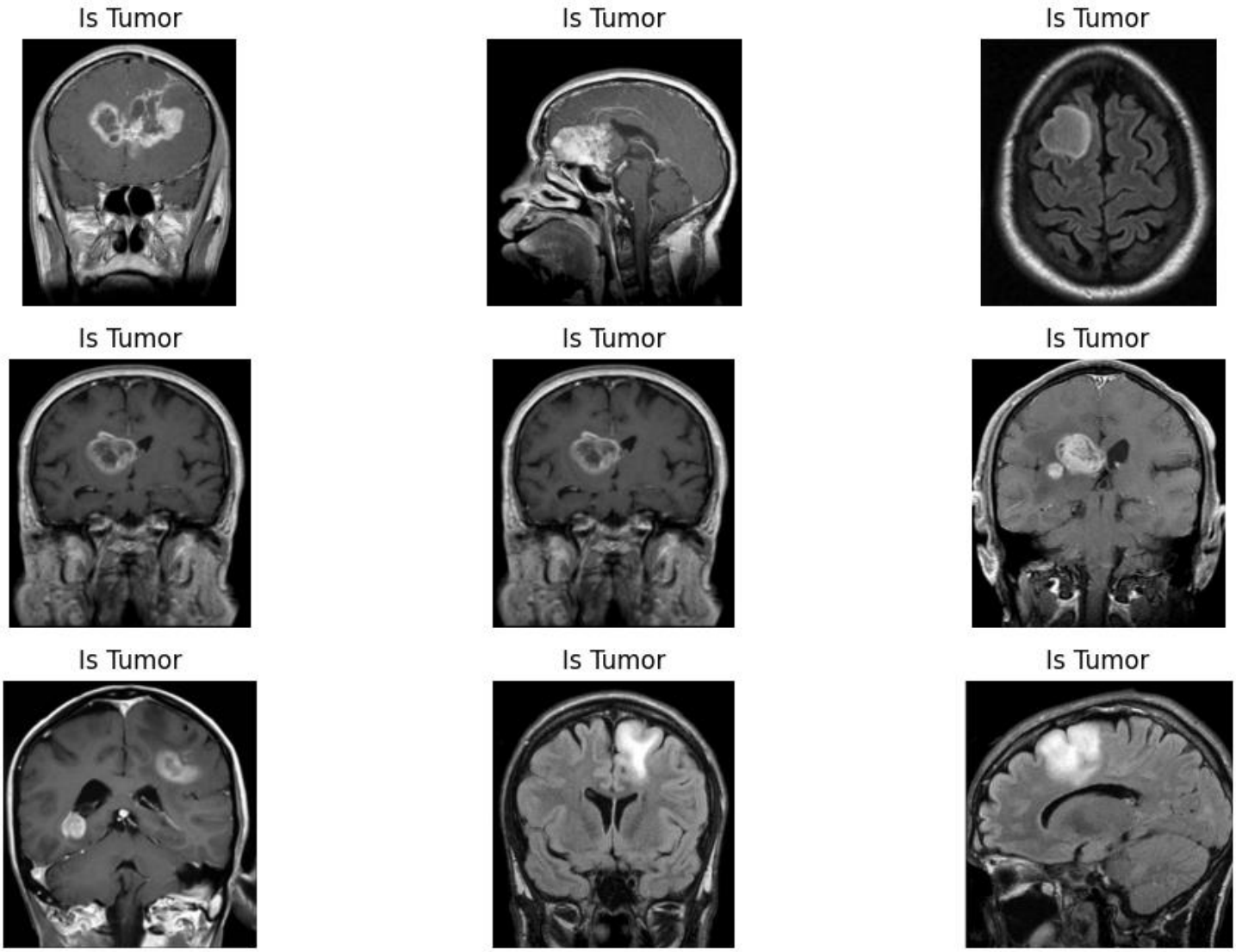
```
[13]: test_based_on_images(classifier, 'meningioma_tumor/')
```



PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

Glioma Tumor Detection – Support Vector Machine – Support Vector Classifier:

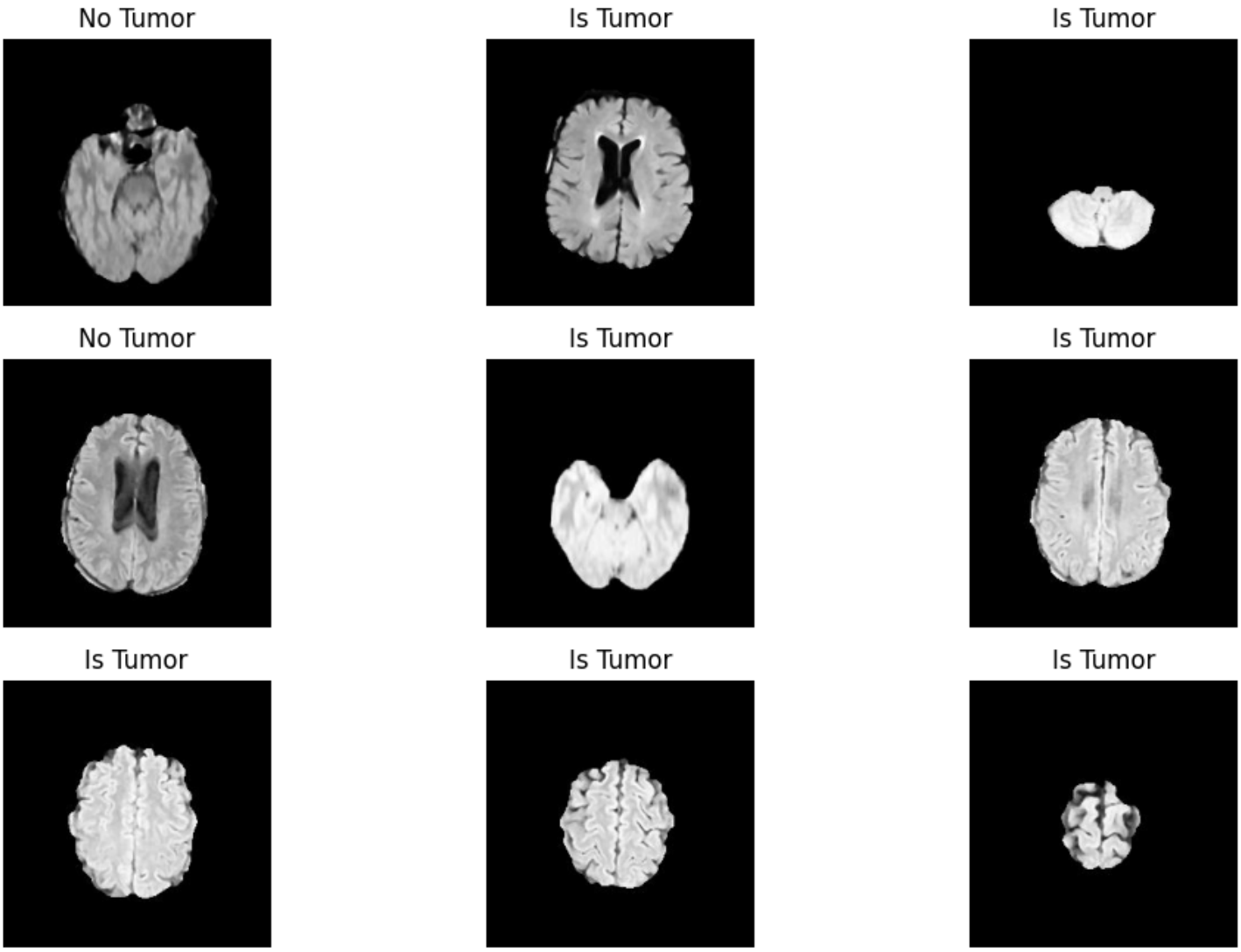
```
[14]: test_based_on_images(classifier, 'glioma_tumor/')
```



PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	

New Tumor Detection Dataset – Support Vector Machine – Support Vector Classifier

```
[15]: test_based_on_images(classifier, 'brain_tumor/', 20)
```



b. Direcţii de dezvoltare

Sistemul dezvoltat in cadrul acestui proiect pentru detectarea anomaliilor cerebrale reprezinta o solutie promitatoare pentru diagnosticarea si tratamentul acestora. Cu toate acestea, exista inca oportunitati de cercetare care pot imbunatati performanta sistemului si pot face posibila detectarea mai precisa a anomaliilor.

Una dintre aceste directii de cercetare este antrenarea sistemului cu date noi si mai complexe. Radiografiile din diferite unghiuri si perspective pot fi incluse in dataset-ul de antrenament pentru a imbunatati performanta sistemului in detectarea anomaliilor. Aceasta ar putea creste acuratetea sistemului si ar face posibila detectarea anomaliilor mai mici si mai putin vizibile.

De asemenea, exista oportunitati de cercetare in procesul de preprocesare a imaginilor. Sistemul poate fi imbunatatit prin identificarea si utilizarea unor metode mai precise de preprocesare a imaginilor, astfel incat sa se asigure ca input-ul oferit algoritmului de machine learning este cel mai potrivit posibil. Aceste cercetari ar putea include identificarea celor mai bune tehnici de preprocesare pentru reducerea zgomotului sau a distorsiunilor, eliminarea artefactelor si imbunatatirea calitatii imaginii.

In concluzie, directiile viitoare de cercetare pentru sistemul de detectare a anomaliilor cerebrale includ imbunatatirea dataset-ului de antrenament si procesul de preprocesare a imaginilor. Aceste cercetari ar putea imbunatati semnificativ performanta sistemului si ar face posibila detectarea mai precisa a anomaliilor. Prin urmare, acest proiect reprezinta doar un prim pas in dezvoltarea unor solutii mai precise si mai eficiente pentru diagnosticarea si tratamentul anomaliilor cerebrale.

PROIECT INGINERIA REGLARII AUTOMATE II					
NUME student		GRUPA:		Nota	