{}
(/home)

sunnyliang5

Applicant Success Predictor API

1.0.0 OAS3

Predicts whether a student will be high-quality and likely to succeed.

Servers

https://virtserver.swaggerhub.com/sunnyliang5/Microservices-API/1.0.0 - SwaggerHub API Auto Mocking

predict Predict application success

POST /predict Predict a student's success by ID.

Using a ML model based on the parameters `studytime`, `failures`, `absences`, `G1`, and `G2`, predict the quality of the student, specifically `G3`. A student predicted to have a `G3` grade of 15 or higher is high-quality.

Parameters

Try it out

Name	Description
studytime * required integer($int32) (query) minimum: 1 maximum: 10	weekly study time (numeric: 1 -<2 hours, 2 - 2 to 5 hours, 3 - 5 to 10 hours, or 4 - >10 hours) studytime
failures * required integer($int32) (query) minimum: 1 maximum: 4	number of past class failures (numeric: n if 1<=n<3, else 4) failures
absences * required integer($int32) (query) minimum: 0 maximum: 93	number of school absences (numeric: from 0 to 93) absences
G1 * required integer($int32) (query) minimum: 0 maximum: 20	first period grade (numeric: from 0 to 20) G1
G2 * required integer($int32) (query) minimum: 0 maximum: 20	second period grade (numeric: from 0 to 20) G2

Responses

Code	Description	Links
200	successful operation Media type: application/json Controls Accept header. Example Value Schema { "studytime": 0, "failures": 0, "absences": 0, "G1": 0, "G2": 0 }	*No links*
400	Invalid values supplied	*No links*
404	Values not found	*No links*

default

POST /train Train the student success predicting ML model.

Train the ML model over a historical dataset (/data/student-mat.csv) of student performance. Our ML model uses 5 inputs from the dataset: `studytime`, `failures`, `absences`, `G1`, and `G2`. The target performance metric of our model is `G3`. The data is in JSON format.

Parameters

Try it out

No parameters

Responses

Code	Description	Links
200	successful operation Media type: application/json Controls Accept header. Example Value Schema { "studytime": 0, "failures": 0, "absences": 0, "G1": 0, "G2": 0 }	*No links*

POST /clean/ Clean the dataset in preparation to train the ML model.

Before training our ML model on a dataset, we should clean the data. Remove corrupt, incorrect, or unnecessary data from the dataset (/data/student-mat.csv), so that our ML model will have greater success.

Parameters

Try it out

No parameters

Responses

Code	Description	Links
200	successful operation Media type: application/json Controls Accept header. Example Value Schema { "studytime": 0, "failures": 0, "absences": 0, "G1": 0, "G2": 0 }	*No links*

POST /test/ Test how the ML model performs.

Run tests to test the classifier accuracy of our ML model.

Parameters

Try it out

No parameters

Responses

Code	Description	Links
200	successful operation Media type: application/json Controls Accept header. Example Value Schema { "studytime": 0, "failures": 0, "absences": 0, "G1": 0, "G2": 0 }	*No links*

Schemas

Predict {
studytime integer($int32)
failures integer($int32)
absences integer($int32)
G1 integer($int32)
G2 integer($int32)
}