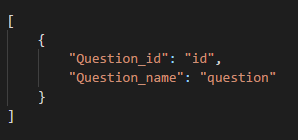
תרגיל3.1 **- API**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Explanations** | **Returns** | **Parameters** | **HTTP Method** | **Method Name** | **ID** |
| נשתמש ב-POST מכיוון שנשמור נתונים חדשים ב-DB. במידה ופרטי ההתחברות לא נכונים תוצג הודעת שגיאה, אחרת פעולה ההתחברות הצליחה. | {  "success": true,  "token": "token"  } | username, password | POST | /users/login | 1 |
| נשתמש ב-POST כיוון שהמידע המועבר פרטי ובנוסף מבוצע שינוי של הDB. | {  "username": "username",  "password": "password"  } | Firstname, Lastname, City, Country, Email, Category1, Category2, Category3, Category4, QuestId1, QuestId2 , Ans1, Ans2 | POST | /users/register | 2 |
| נשתמש ב-GET מכיוון שבבקשה זו מבקשים לקבל מידע מהשרת ואין כוונה לשנות מידע בשרת. במידה והתשובה שהזין המשתמש אינה נכונה תוצג הודעת שגיאה. | {  "success": true,  "password": "password"  } | username, questId, ans | GET | /users/retrivePassword | 3 |
| נשתמש ב-GET מכיוון שבבקשה זו מבקשים לקבל מידע מהשרת ואין כוונה לשנות מידע בשרת. בפונק' זו נקבל את נקודות העניין המועדפות שהמשתמש שמר. במידה והמשתמש לא שמר נקודות עניין תוצג הודעה מתאימה. | PointOfIntesrt []  (תיאור ה-json נמצא בסוף הטבלה) | - | GET | poi/validation/FavoritePointsOfInterest | 4 |
| נשתמש ב-GET מכיוון שבבקשה זו מבקשים לקבל מידע מהשרת ואין כוונה לשנות מידע בשרת. בפונק' זו נחזיר שלוש נקודות עניין פופולריות ורנדומליות. | pointOfIntrest[3]  (תיאור ה-json נמצא בסוף הטבלה) | - | GET | /poi/RandPopularPointsOfInterst | 5 |
| נשתמש ב-GET מכיוון שבבקשה זו מבקשים לקבל מידע מהשרת ואין כוונה לשנות מידע בשרת. בפונק' זו נחזיר שתי נקודות עניין פופולריות מהקטגוריות בהן מתעניין המשתמש. | pointOfIntrest[2]  (תיאור ה-json נמצא בסוף הטבלה) | - | GET | /poi/validation/RecomendedPointsOfInterest | 6 |
| נשתמש ב-GET מכיוון שבבקשה זו מבקשים לקבל מידע מהשרת ואין כוונה לשנות מידע בשרת. בפונק' זו נחזיר את כל נקודות העניין הקיימות במערכת. | pointOfIntrest[]  (תיאור ה-json נמצא בסוף הטבלה) | - | GET | /poi/AllPointsOfInterst | 7 |
| נשתמש ב-GET מכיוון שבבקשה זו מבקשים לקבל מידע מהשרת ואין כוונה לשנות מידע בשרת. בפונק' זו נחזיר את רשימת נקודות העניין בקטגוריה המבוקשת. | points of interest: PointOfIntesrt []  (תיאור ה-json נמצא בסוף הטבלה) | category | GET | /poi/PointsOfInterstByCategory/:category | 8 |
| נשתמש ב-GET מכיוון שבבקשה זו מבקשים לקבל מידע מהשרת ואין כוונה לשנות מידע בשרת. במידה ולא נמצאה נקודת העניין תוצג הודעה מתאימה. | point of interest: PointOfIntesrt poi  (תיאור ה-json נמצא בסוף הטבלה) | name | GET | /poi/PointOfInterstInfoByName/:name | 9 |
| נשתמש ב-GET מכיוון שבבקשה זו מבקשים לקבל מידע מהשרת ואין כוונה לשנות מידע בשרת. במידה ולא נמצאה נקודת העניין תוצג הודעה מתאימה. | point of interest: PointOfIntesrt poi  (תיאור ה-json נמצא בסוף הטבלה) | id | GET | /poi/PointOfInterstInfoById/:id | 10 |
| נשתמש ב-PUT כיוון שנקודות העניין **מתעדכנות** בבסיס הנתונים של השרת. נקודות העניין ישמרו לפי הסדר בו הן מופיעות במערך. | {  "success": true,  "message": "message”  } | poisId: pointOfInterestId[] | PUT | /poi/validation/updateFavoritePointsOfInterest | 11 |
| נשתמש ב-POST כיוון שיש הוספה של מידע ב-DB. | {  "success": true,  "message": "message”  } | poiId, rank | POST | /poi/validation/rankPointOfInterest | 12 |
| נשתמש ב-POST כיוון שיש הוספה של מידע ב-DB. | {  "success": true,  "message": "message”  } | poiId, review | POST | /poi/validation/reviewPointOfIntrest | 13 |
| נשתמש ב-PUT כיוון שנקודות העניין **מתעדכנות** בבסיס הנתונים של השרת. נעדכן את מספר הצופים בנקודת העניין ע"י הגדלת ערך זה ב-1. | {  "success": true,  "message": "message”  } | - | PUT | /poi/updateNumberOfViewers | 14 |
| נשתמש ב-GET מכיוון שבבקשה זו מבקשים לקבל מידע מהשרת ואין כוונה לשנות מידע בשרת. בפונק' זו נחזיר את רשימה הקטגוריות בהן המערכת תומכת. | categories: Categories []  (תיאור ה-json נמצא בסוף הטבלה) | - | GET | /Categories | 15 |
|  | Countries:  Country[]  (תיאור ה-json נמצא בסוף הטבלה) | - | GET | /Countries | 16 |
|  | Questions:  Question[]  (תיאור ה-json נמצא בסוף הטבלה) | - | GET | /Questions | 17 |
| נשתמש ב-GET על מנת לאחזר את שאלות האימות של משתמש ספציפי | Questions:  Question[] | - | GET | /users/userQuestions/:username | 18 |

ה-json של pointOfIntrest:



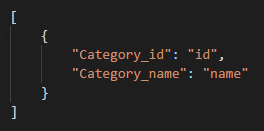
ה-json של question:



ה-json של country:



ה-json של category:



המודולים שיצרנו:

users- login, register, retrivePassword

poi- getRandPopularPointsOfInterst, getFavoritePointsOfInterest, getRecomendPointsOfInterest, getAllPointsOfInterst, getPointsOfInterstByCategory, getPointOfInterstInfoByName, getPointOfInterstInfoById, updateFavoritePointsOfInterest, rankPointOfInterest, reviewPointOfIntrest, updateNumberOfViewers

general- getCategories, getCountries, getQuestions

הערות:

* בתהליך ההרשמה (register) השרת מחולל עבור כל משתמש שם משתמש וסיסמה ייחודיים ומחזיר למשתמש.
* לאחר ההתחברות, בכל בקשת HTTP שנשלחת לשרת מצורף ב-header של הבקשה הTOKEN שהתקבל לאחר ביצוע login.

העיר בה בחרנו היא Rio de Janeiro בברזיל.

קטגוריות נקודות העניין בהן נתמקד: חיי לילה, קניות, אתרים וציוני דרך, סיורי מזון ומשקאות.

נקודות עניין בקטגוריית חיי לילה:

* Rio Scenarium
* Carioca de Gema
* Bar Bukowski
* Lapa 40 Graus
* Barra Music

נקודות עניין בקטגוריית קניות:

* Lapa e Arte
* Shopping Village Mall
* Barra Shopping
* Rio Antigo Fair
* Shopping Nova Amrica

נקודות עניין בקטגוריית אתרים וציוני דרך:

* Corcovado Christ the Redeemer
* Sugarloaf Mountain
* Maracana
* Favela
* Santa Teresa

נקודות עניין בקטגוריית סיורי מזון ומשקאות:

* Eat Rio Food Tours
* Urban Adventures
* Strawberry Tours
* Rio4Fun
* Secret Food Tours Rio de Janerio

