מערכות מבוזרות – פרוייקט סוף סמסטר

[ פיתוח שירות WEB מבוסס REST ]

|  |  |  |
| --- | --- | --- |
| **זוהדי חאסקיה** | **311231898** | **Zohdee1@gmail.com** |
| **פואד נאצר אלדין** | **201211570** | **Fuadsami5@gmail.com** |
| **סיף אדין פרחאן** | **315301580** | **zenseifalden@gmail.com** |

מבוא :

בקורס למדנו על ארכיטקטורת REST בפרויקט הזה נרצה לממש את הארכיטקטורה על מנת לבנות שירות WEB ברצונינו לבנות צד שרת וצד לקוח שמתקשרים ביניהם על פי סטנדרטים שנרצה.

בשירות שנספק הוא חיפוש של מדינה בעולם לפי מספר המדינה

למשל אוסטרליה המספר שלה הוא 040.

המשתמש מכניס את הקוד ויוצא לו שם המדינה שהקוד שייך לה , והשרת שולף את המידע הרצוי ומחזיר תשובה. הבקשה והתשובה שתיהן בפורמט XML.

מטרת הפרוייקט

להתנסות בפיתוח בסביבת WEB SERVICE עם REpresentational State Transfer - REST ֹֹֹֹֹֹ- ושימושיו .

תכנות הפרויקט

לאחר חיפוש קטן באינטרנט ראינו ששתי הדרכים הכי נפוצות למימוש שירות כזה הן שימוש ב JAVA או ב- PHP , שתי השפות מספקות ממשק פיתוח WEBSERVICE וגם ספריות לפיתוח שרת-לקוח .

ולכן בחרנו ב- PHP

כלים

על מנת לפתח ב –PHP היינו צריכים את הכלים הבאים :

MySQL Database

Apache WebService

שהתקנו על המחשב הפרטי דרך XAMPP שהוא שרת וירטואלי אשר הופך את מחשבך לשרת מדומה ובו אתה יכול להתנסות ולפתח מערכות ב-PHP ו-MySQL

שלבי ביצוע

1. היכרות עם הממשק REST
2. תיכון קוד.
3. כתיבה וניסויים.
4. היכרות עם הממשק REST :

REST הוא סגנון ארכיטקטורה לעיצוב יישומים ברשת. הרעיון הוא שבמקום להשתמש במנגנונים מורכבים כגון CORBA, RPC או SOAP לחיבור בין מכונות, HTTP פשוט משמש לביצוע שיחות בין מכונות - (ליצור נתונים/לעדכן/ל'קור/למחוק.

הוא מבוסס על שימוש במונחים ובמאפינים קיימים בתוך פרוטוקול HTTP להגדרת אינטראקציה בין שרת ולקוח, כגון: [headers](https://he.wikipedia.org/w/index.php?title=Header&action=edit&redlink=1) לפניות (Requests) ולתגובות (Responsess), ופורמטים של קבצים באינטרנט (INTERNET MEDIA TYPES)

1. תיכון הקוד :

ברצונינו לממש שרת ולקוח שמתקשרים אחד עם השני וממשק למשתמש. המשתמש מזין מספר קוד של מדינה ומקבל בחזרה את שם המדינה.

1. כתיבה וניסויים :

חיפשנו באינטרנט על דף XML שמכיל את הקודים של המדינות , הימרנו אותו בעזרת PHP והכנסנו ל- DATABASE

תכנתנו את השרת הלקוח לקבל ולהחזיר קריאות בניהם. והקמנו את ממשק המשתמש שמעביר ומקבל בקשות ותשובות מהלקוח.

ניסויים:

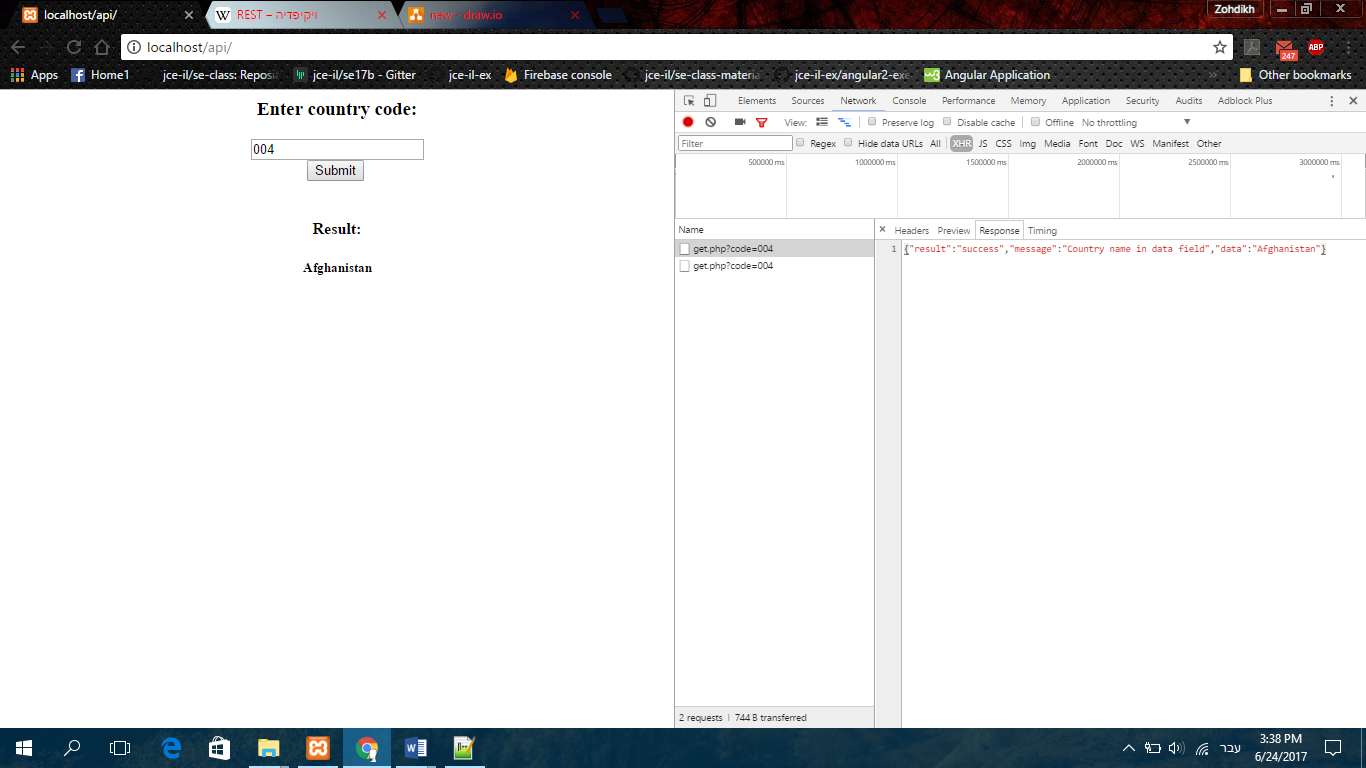
1. עבור הניסוי הראשון כתבנו פונקציה שמעבירה ארגומנט אחד. מצד הלקוח מיקוד המדינה ומהשרת את השם.

Request=> הכנסנו קוד ב LABEL ובלחיצה ONCLICK על הכפתור submit הוא מעביר בקשה לשרת דרך function load() שבקובץ index.html.

<= Response קבלת התשובה מפנה אותנו ל function process() שבקובץ index.html.

צורת הכתיבה לשרת היא דרך JSON

מסקנות וממצאים

קיים ויכוח בקהילה המקצועית בתחום המחשוב ביחס לשימוש ב-[SOAP](https://he.wikipedia.org/wiki/SOAP) או ב-REST במימוש [Web Services](https://he.wikipedia.org/wiki/Web_Service). בשנות האלפיים המוקדמות, הייתה העדפה ברורה לשימוש ב-SOAP. בשנים האחרונות השתנתה מגמה זו. ההבדל המהותי בין SOAP ל-REST הוא ש-REST היא ארכיטקטורה ו-SOAP הוא פרוטוקול.

