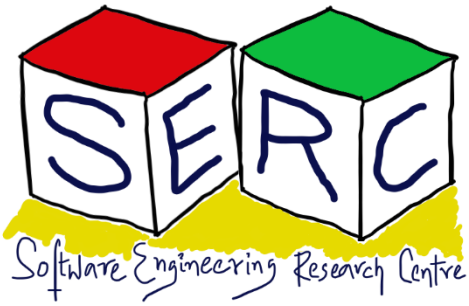


Internship Project Report

Submitted by

Krupal Shah

Umetrix

&

Guidance Tool

Under the guidance of

Prof. Y. Raghu Reddy	Supervisor
Sai Anirudh Kaari	Mentor
Vivek Pareek	Mentor
Neeraj Mathur	

Context

All the mobile and smart apps developed are targeted to serve a maximum possible audience. User-friendliness of the UI and functionalities of apps are one of the parameters which determine acceptance and market penetration of the apps. A Better interface and intelligently placed components can attract a larger userbase. Companies like Neilson & Norman does research and provide guidelines which are expected be followed to have a better app interface.

Problem Statement

Apps need to be tested to verify their user-friendliness. Currently, this task is done manually which consumes a lot of time and resources because the app needs to undergo the production stage to get the final app over which manual evaluation can be done. The project 'Umetrix' aims at automating the task of manual evaluation for some of the guidelines over usability. This can be achieved by searching the source code for some particular code snippets which implement particular on the UI. Guidelines are written in text form. Task is to convert them into the required XAML format.

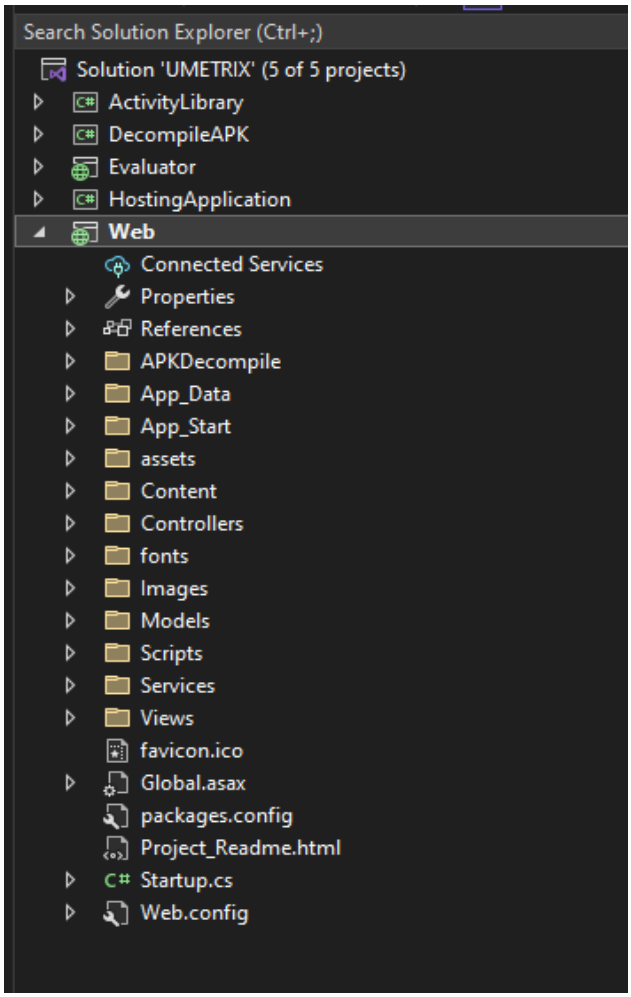
There is a need to store and efficiently retrieve back code snippets which implement particular guidelines and UI features. It helps speed up the task where there is a need to type repetitive codes. The Guidance tool stores and retrieves back the code snippets related to particular keyword tags with additional information.

Contribution to developing the solution

1. Got the Umetrix tool online over the internet. Umetrix uses code snippet form of UI guidelines to check across for the snippets in source code. I designed and fed in more instructions for the validator database making more types of apps supported. The instructions were made using Windows Workflow Foundation and submitted in form of .XAML files. Instructions were referred to from MS Thesis by Neeraj Mathur.
2. Made the whole Guidance Search Tool which is basically a tool used to store code snippets with respect to certain tags or keywords. Functionalities were provided to post or add code snippets to store them in database and retrieve them based on search tags from search bar after applying some filters if required.

Understanding the workflow of the Guidance Search tool

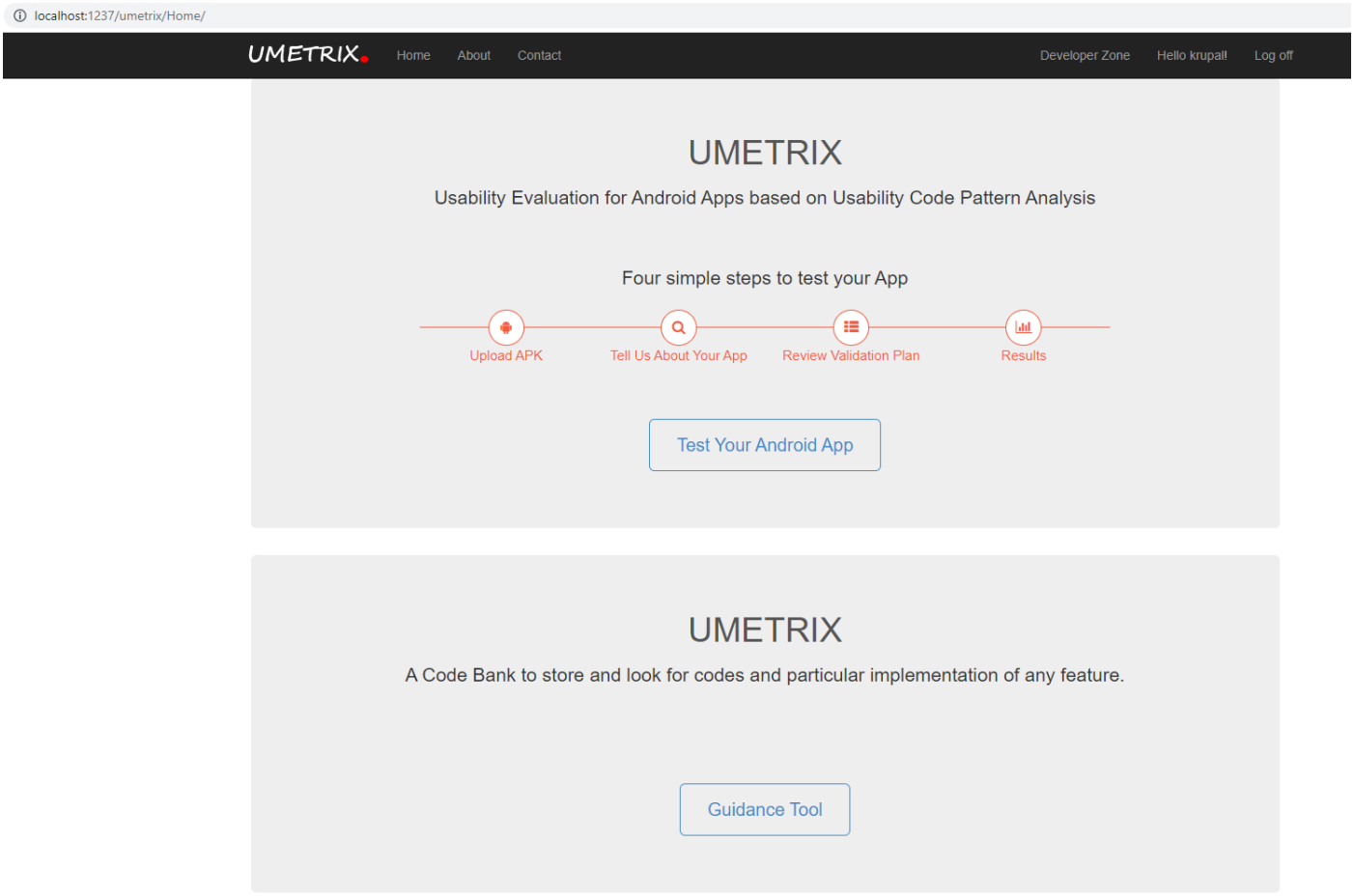
This tool is based on C# MVC project. Following is the project structure.



- The Views provide with front-end designing, each page of the website has an independent view.
- The Controllers are written in C# and it contains the functions. These functions are used to control all the rendering and functional changes.

Data is stored in MongoDB to serve the user requests. Django is used to write a service that will make API calls to MongoDB and serve the data to C# MVC project.

Entry Point to the Website: */Home/*



The Guidance Tool is developed by me while the Test Your Android App button leads to the original Umetrix project which is used to determine the user-friendliness of the apps from the APK file of the app by decompiling it and using its source code. That was developed by Neeraj sir and I have added more guideline checks to its validator database.

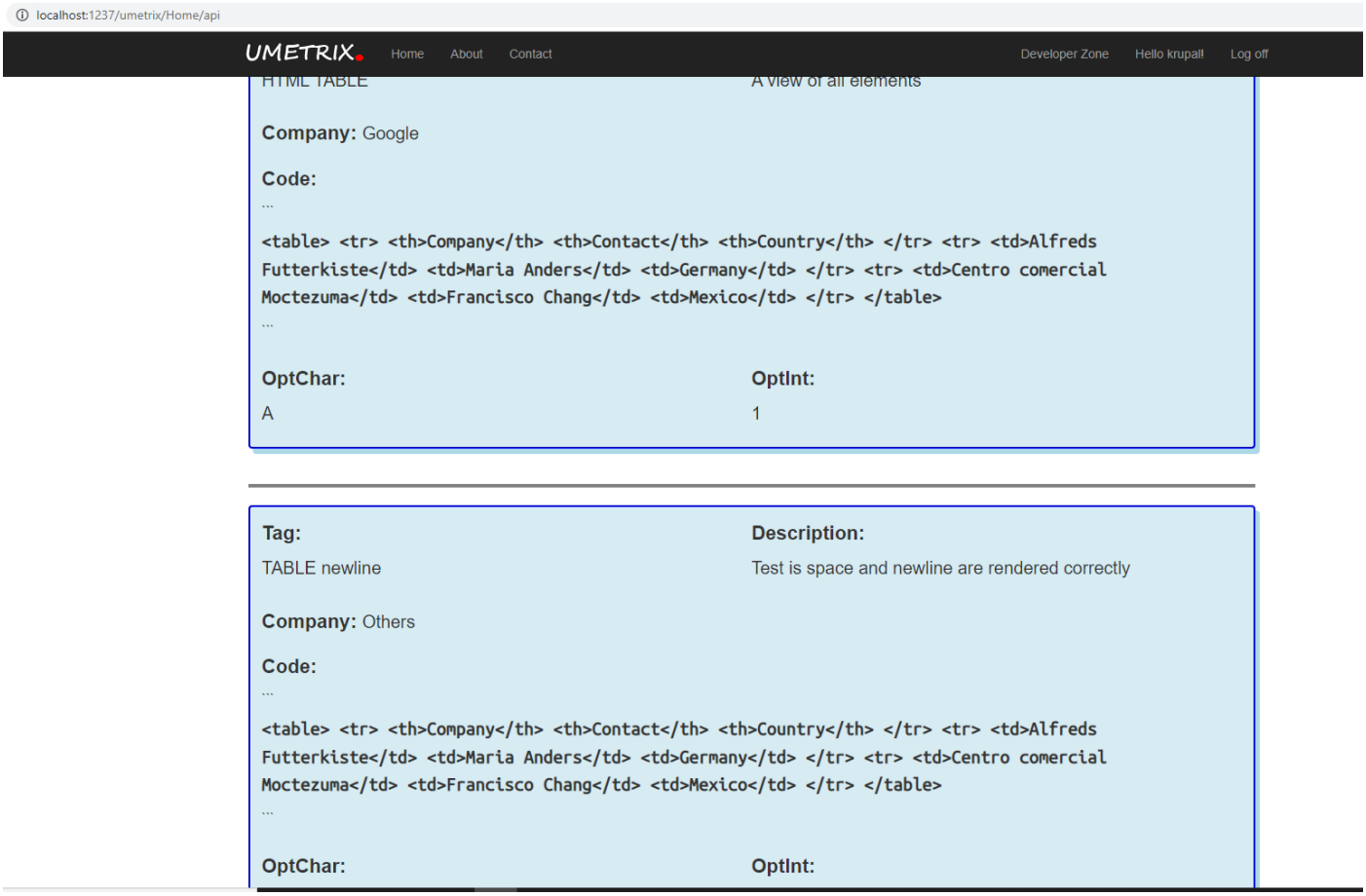
Search bar: */Home/ProductivityTool*

When entering the Guidance tool, this will be the search bar that can be used to search for the required tag or keyword of the code snippet required. It will display the related code snippets.

The search results can be filtered using the Select Company dropdown. This can help select the company giving out the guidelines – Apple, Neilson Norman, Google, and others.

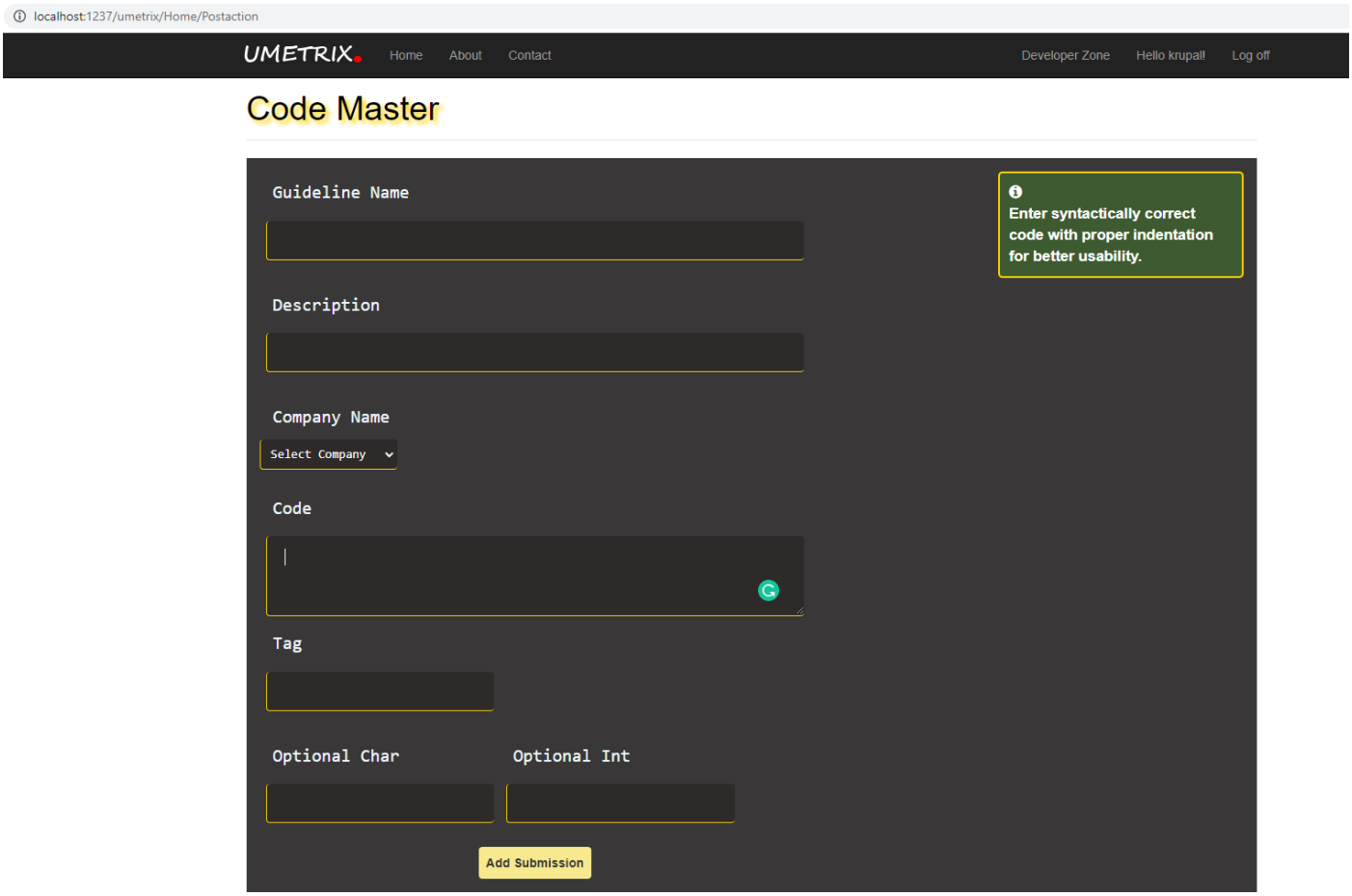
The search and filter work by sending a GET request with proper information from C# controller to the Django service. This service will parse the request and search for the required data in database and perform the filtering. Then the data will be returned to MVC project which will display it in required format.

Data Display Page or Search Results page: */Home/api*



Each data table of codes will be shown with full details available. CSS is simple and basic and can be changed easily as per the requirements.

ADD CODE: */Home/Postaction*

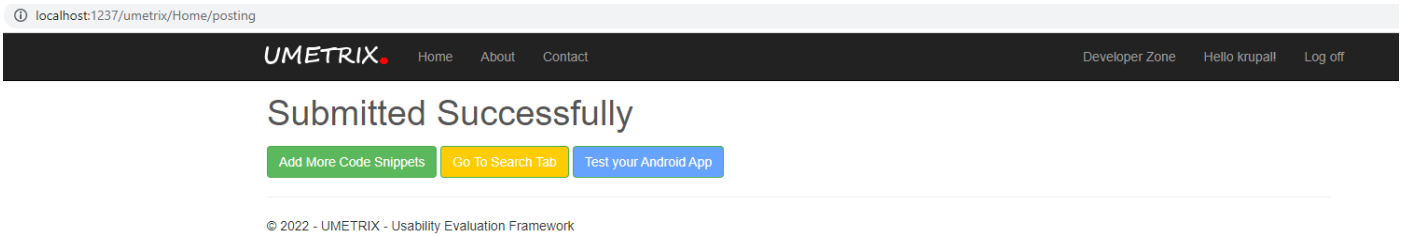


This page facilitates the user to input the code to database. Fields are self-explanatory. Clicking on field to enter data activates the field and provides the user with an information pop-up on what to fill up and the correct format for entering the data.

MULTI - TAGGED data storage - Tag is the keyword holder which will be used for searching in the database. It provides an option to provide multiple tags to the same information of code. Users can search for any of them and still get the required data.

NOTE - Only if the given data is correct as per the requirement, the submission will be accepted. Otherwise, user will not be able to click on the “Add Submission” button.

Successful submission: */Home/posting*



Successful submission will lead you to the page as shown above. This page will appear only if the submission is successful.

Three buttons provided can lead users to three different pages.

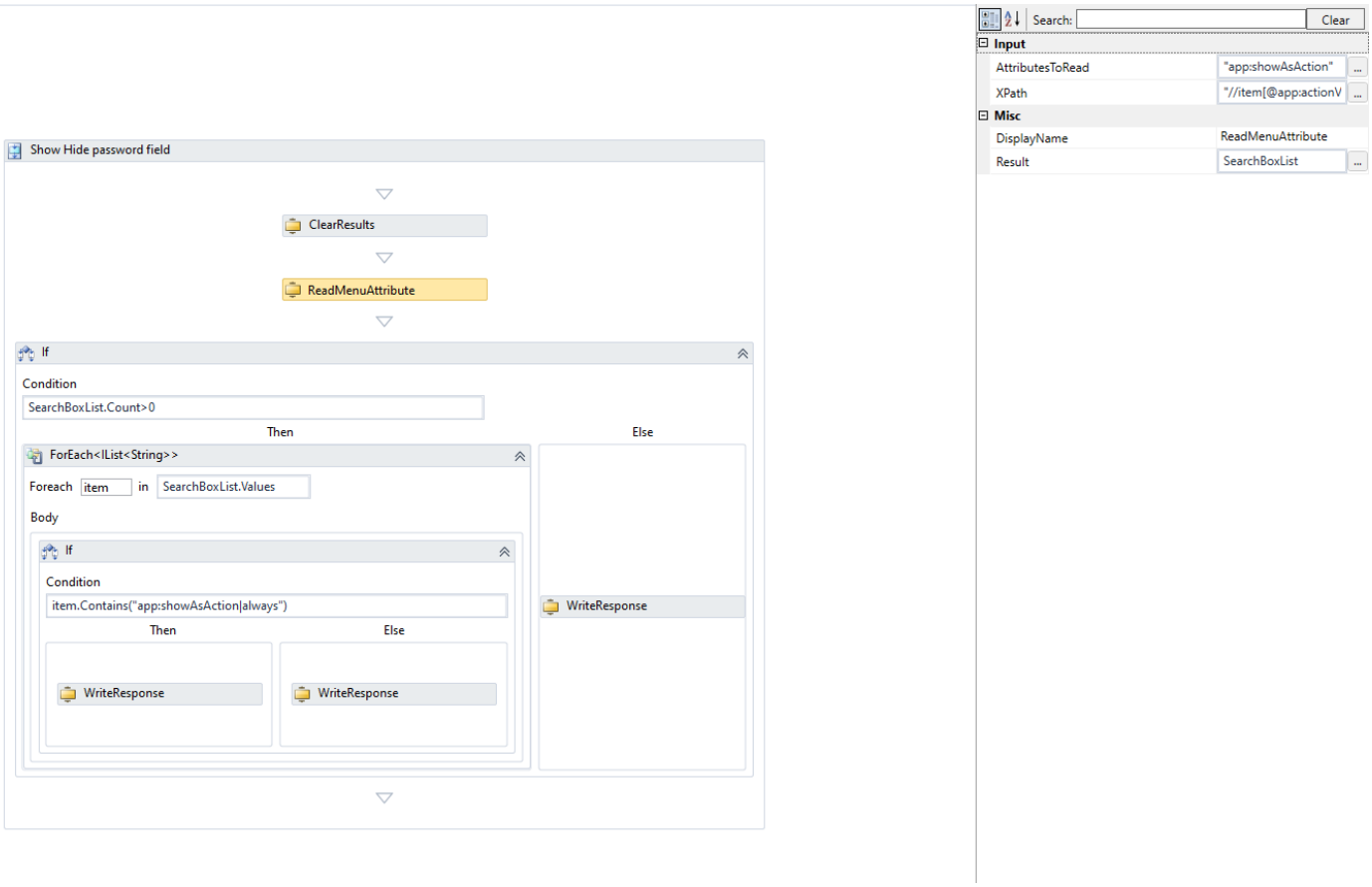
1. Add more code snippets
2. Searching page
3. Test the android app i.e., the Umetrix android testing app.

Installations required

1. C# MVC project
2. Python
3. Django
4. Django-Rest-Framework
5. MongoDB compass
6. SQL Express server
7. IIS Manager
8. Microsoft SQL Server Management Studio [SSMS]
9. MongoDB Compass
10. Visual Studio

About the Umetrix Testing app

Apps needs to have certain features for better usability. Manual testing takes time and money. Some of the tests are automatable by searching for a particular type of code from the entire source code. This work is done by Umetrix. It looks into the .XAML and .java files and searches for some code snippets which is pre-fed by the developer. Then it gives percentage approval accordingly. Certain code snippets are used to implement specific options or functions and the system searches for the occurrence of these codes. I was tasked to make more guidelines to check against and add to the database.



The above picture is of the Windows Workflow Manager guideline designer tool. It takes in conditions in flowchart type and converts them into corresponding .XAML files.

Other works

Setup the server to host Umetrix and the required Django API to search for the data in MongoDB database. This server was used to make Umetrix online.

Experience

Skills developed – Python, Django, C# and MongoDB. It can be said that almost all the things I did in this project was new for me. I learnt about networking and protocols to put Umetrix website on intranet and internet.

This project helped me improve my development skills and provided me an experience with the functioning and process of software development. Biggest skill learnt here would be efficient debugging of codes.

Major Problems and Roadblocks

- Starting up Umetrix tool and getting it functioning took a long time. Neeraj sir helped out and some versions of programs had to be degraded to get compatible with internal code used by Neeraj Sir earlier.
- Initially, I used C# WebAPI to connect to Database but it was not easily portable and created version problems. So, I wrote a microservice using Django-rest-framework to communicate with database from C# MVC.
- I faced difficulty in creating URLs for different requests as the number of parameters passed varied. Two independent URLs and controllers were made to solve this problem.

Upstanding Problem

The code entered is properly indented and in new lines while the return of code on using search button provides back the code in a single string format with no indentation and new line taken care of.

Example:

Input	Output
<html> ABCD	<html> ABCD

Conclusion

I have implemented the final approach about the Guidance Search Tool thought of by me and Sai sir. Guidance tool is an independent extension of Umetrix. My goal was to make it search for any particular Tag or company, return respective data and take data from user. More guidelines were added to Umetrix and tried to take it forward. Both the tools are up and running.

Codes

https://iiitaphyd-my.sharepoint.com/personal/saianirudh_karri_research_iiit_ac_in/_layouts/15/onedrive.aspx?ga=1&id=%2Fpersonal%2Fsaianirudh%5Fkarri%5Fresearch%5Fiiit%5Fac%5Fin%2FDocuments%2FInternVideoRecordings%5FSummer2022%2FKrupal%20Umetrix%20Intern

https://github.com/VHIL-interns-hub/UMETRIX_summer2022

Main codes are in IIIT OneDrive cloud provided by Sai sir. Work-logs and certain reference documents are present in VHIL GitHub link

References

neerajmathur/UMETRIX

https://www.youtube.com/watch?v=eeo0e1ajNnM&ab_channel=NeerajMathur

Working With Html.BeginForm() and Ajax.BeginForm() in MVC 3

Django routing APPEND_SLASH error auto-complete "/" and trailing_slash solutions - Fear Cat

What Is NoSQL? NoSQL Databases Explained | MongoDB

https://stackoverflow.com/questions/62427146/django-rest-framework-get-field-name-from-model-definition

https://www.webforefront.com/django/introrestservices.html

Setup

1. Follow the installation guidelines by Neeraj Mathur for the Umetrix app using the Umetriz-master.7z.
   Reference: https://github.com/neerajmathur/UMETRIX/tree/master/Installer
   - Download Umetix_Binaries folder and store it in D drive.
   - Grant Read,Write,Execute permission to “Everyone” for “D:\Umetrix_Binaries\Web\APKDecompile” and “D:\Umetrix_Binaries\Web\App_Data” folders
   - Create Junction for decompilation requirement D:>mklink /j D:\U "\APKDecompile" Example: D:>mklink /j D:\U "D:\Umetrix_Binaries\Web\APKDecompile" NOTE: if change link path from “D:\U” and do change it in web.config file as well
   - <appSettings>
     <add key="DecompilerPath" value="D:\U\" />
     </appSettings>
   - Open Umetrix\DB_Script\UMETRIX_DB_Script.sql and SSMS. Click on create new query in the top toolbar and paste the code copied from the .sql file opened. And execute it. If this does not work double click on the UMETRIX_DB_Script.sql file.

    - Open IIS and create a website there. This can be opened in http://localhost:8080
2. Open the Umetrix.sln file in Visual Studio and change the start-up project to Web to see the website in browser.
3. Download or update Python and pip.
4. Download Django and Django-rest-framework, you might need to drop down the version of Django to 4.0.5 due to project dependencies.
5. Download Djongo, it helps use MongoDB in backend without making single changes in Django model.
6. Download MongoDB Compass.
7. Go to the folder DjangoAPIService and open it in Visual Studio.
8. To run the Django service that connects C# front-end to the MongoDB, there is a need to python virtual environment. To create a virtual environment.
    a. In the Developer Powershell, Go to CreateVirtualENV/Scripts using CD command and execute the activate file. Confirm if the virtual environment is created.
    b. In the same terminal which has created the virtual environment, run the following lines for the first time:
        i. python manage,py makemigrations
        ii. python manage.py migrate
        iii. python manage.py runserver.
9. Run the above 3 commands when there is some change in the model or database. Else run it once. Then only runserver command is required to run the Django service.

I have tried to solve and code all the changes in the files. So, it is expected that not much errors would creep in.

Hints to Debugging or solve errors

Packages required:

jQuery by jQuery Foundation, Microsoft.AspNetCore.Http.Abstractions, Microsoft.AspNetCore.Http.Features, Newtonsoft.JSON, Microsoft.AspNet.Mvc

To change Database name, go to settings.py file.

If there is host related problem, there is an allowed host function in settings.py file. Enter the IP address of system on which you want to run it.

Earlier only smaller files were allowed to be uploaded. This was due to maximum connection string size limit.

<system.webServer>

<security>

<requestFiltering>

<requestLimits maxAllowedContentLength="524288000" />

Making the above change in web.config will solve the issue if occurred again.

There will always be a need to give all permissions to all the folders. Go to properties, edit permissions and add ‘everyone’ as user name. Give all the permissions to everybody user.

To host the website on internet or intranet, there is a need to publish the web folder of the project and put it in the inetpub. This will be used to host the website in the IIS.

There is also an allowed host setting in Django service that needs to be taken care of.

Add IIS APPPOOL as verified user in Databases/Umetrix/Security/Users in SQL express server in SSMS. Grant all the check boxes except the denydatareader and denydatawriter boxes.